Conference contribution :
Rick, W., Philip, A., Serban, P., Zhao, G., Robert, S. & Jonathan, C. (2013). *Force-Directed Parallel Coordinates.* 2013 17th International Conference on Information Visualisation, Institute of Electrical & Electronics Engineers (IEEE).

http://dx.doi.org/10.1109/iv.2013.101

Force-Directed Parallel Coordinates

Rick Walker[1], Philip A. Legg[2], Serban Pop[1], Zhao Geng[2], Robert S. Laramee[2] and Jonathan C. Roberts[1]
[1]School of Computer Science, Bangor University, UK
[2]School of Computer Science, Swansea University, UK

Abstract

Parallel coordinates are a well-known and valuable technique for the analysis and visualization of high dimensional data sets. However, while Inselberg emphasizes that the strength of parallel coordinates as a methodology is rooted in exploration and interactivity, the set of interaction techniques is currently limited. Axes can be re-ordered and brushing (simple, angular or multi-dimensional) can be performed. In this paper, we propose a force-directed algorithm and related interaction techniques to support the exploration of parallel coordinate plots through a physical metaphor. Our parallel-coordinates visualization offers novel user interaction beyond the standard techniques by allowing the user to rotate the axis according to force-directed polylines. The new interaction provides the user with a more immersive experience for data exploration that results in greater intuition of the data, especially in cases where many polylines overlap. We demonstrate our approach, then present the results of a qualitative evaluation of the system.

1 Introduction

Parallel coordinates [16] remain a fundamental technique for information visualization. In particular, parallel coordinate plots (PCPs) allow multidimensional data sets to be presented in a natural manner without loss of information, and have been gainfully applied to many such problems. The problems of occlusion when displaying large data sets and of determining an appropriate axis order to highlight multivariate relations have been the subject of much recent research [7]. In applying parallel coordinates to real world problems, though, Inselberg stresses that interactivity is essential [15].

In exploratory analysis, the role of visualization is often not to answer specific questions, but in determining what questions might have interesting or useful answers. Direct and indirect manipulation techniques are vital in this process, but the set of such interaction techniques applicable to PCPs is currently limited. Axes can be brushed for selection or filtering, as can the spaces between axes, and the order of axes can be changed, algorithmically or through user interaction. While these are powerful, a richer set of interactions could improve the exploration process, and provide the user with a better understanding of the underlying data.

Another motivation behind our work lies with interpretation. New users often have difficulty interpreting parallel coordinate visualizations and understanding their benefits when seeing them for the first time. The model we present helps facilitate interpretation. One obvious visual comparison for a PCP is with a bridge. This is a useful analogy for our work because a cable-stayed bridge consists of cables connected to towers to support a road. This is analogous to a PCP where the axes of the plot correspond to the towers of the bridge, and the cables to the projected data points. Bridges, though, are much more familiar objects: the stresses and tensions in the towers and cables that compose these structures are well-understood.

In this paper, we present force-directed parallel coordinates, a technique that uses a familiar visual metaphor to represent data in PCPs by modelling the plot as a physical system, together with some physical interactions with the user. We contribute:

- A novel visual metaphor for parallel coordinate plots as a physical system to aid intuitive understanding of the underlying data
- Three new interaction techniques that act in concert with this metaphor to aid with exploration and the occlusion problem

We demonstrate the benefit of this novel model and interaction on a number of both synthetic and real world data sets, and conduct a user study to evaluate the technique.

The remainder of this paper is organised as follows: work from a number of related fields is reviewed in Section 2. In Section 3 we describe the mathematical model for the physical system formed from the plots. Novel interactions with the data through this system are described in Section 4 and demonstrated in Section 5. The methodology and results of a user study on usability is presented in Section 6 and these results are discussed together with suggestions for future work Section 7.

2 Related Work

While there is a vast amount of literature on parallel coordinates in general [15], we concentrate on work that focuses on interaction with PCPs. Interaction techniques are summarised in work by Kosara et al. [20] for information visualization general and Siirtola [24] for parallel coordinates in particular. We consider recent research in two areas specific areas of PCP interaction — brushing and axis re-ordering — before moving on to examine other uses of force-directed techniques in information visualization.

Brushing, the act of selecting of a set of polylines within a PCP on which to perform an operation such as zooming, deleting, highlighting or masking, has been referred to as the fundamental interaction with parallel coordinate plots [24]. It can occur on or between axes - Inselberg [15] refers to these as *Interval* and *Pinch* respectively. By combining brushing on multiple axes with logical operators brushes can act in multidimensional fashion [21], as in the XmdvTool system [28]. Polyline segments can also be selected based on their correlation, since in PCPs correlation is depicted as the slope between axes [9]. This allows a more nuanced selection than selection on a single axis and facilitates identification of outliers.

Allowing selection directly on the plot, rather than providing a separate brush interface, can enhance the process by providing continuous feedback [23], and modern hardware may enable a more natural interface for this direct manipulation [18]. Brushing can be used to increase display resolution for the selected region [30] and has also been demonstrated for large data sets [2]

Axis re-ordering is an important element of exploring data sets using parallel coordinates, since correlations or relationships are only visible between adjacent axes. While for a parallel coordinate plot of n dimensional data there are $n!$ possible axis permutations, only $\lfloor (n+1)/2 \rfloor$ are needed to ensure adjacency of every pair of axes [29]. However, multidimensional correlations are still easily overlooked and hence determination of useful and appropriate axis orderings is an active research field.

Typically, axis ordering algorithms operate by defining a metric for desirability of layout, and then using an algorithm to find axes orderings that optimise this metric. Dimension arrangement has been shown to be an NP-complete problem [1], so heuristic algorithms such as the ant system algorithm [1], random swapping, nearest neighbor, greedy [22] or branch-and-bound [4] are used. Similarity [1, 25], clutter [22] and screen space [4, 17] have all been used as metrics for these algorithms, which may produce a range of possible useful orderings.

Force-directed algorithms have been widely used in graph layout [6, 5] to produce aesthetically pleasing representations by performing simplified simulations of physical systems. A system called RadViz [13] uses Hooke's Law to layout points based on spring forces. Hierarchical edge bundling [14] uses a bundling strength parameter as a measure of stiffness, while Tominski et al. [26]'s Bring Neighbor Lens uses an attraction approach to show neighbors to a given node or set of nodes.

The force-directed approach has some significant challenges: high computational complexity ($O(n^3)$)and lack of predictability of layout for similar graphs [12]. The algorithm may never reach a stable state or may take a long time to do so. There may also be several stable states for the system, and local minima may prevent it reaching the most globally stable one. We address this challenge primarily with interaction in our work.

3 Physical Modelling

In this section, we describe the physical model by considering line segments as springs and axes as solid rods fixed on a pivot, together with a Verlet integration scheme.

3.1 Spring Forces

Consider one spring, with end points $\vec{a}$ and $\vec{b}$, spring constant k and rest length l. The length of the spring, d, is then $|\vec{b} - \vec{a}|$. The force exerted by the spring is, by Hooke's Law:

$$F = -k(\frac{d-l}{d}) \tag{1}$$

Since the spring is attached at both ends, $F/2$ acts on each endpoint, and hence on each axis-rod. The rest length for each spring is set to match the inter-axis spacing.

3.2 Axis forces

As the rod is fixed at a pivot point, $\vec{p}$, the turning force (torque, τ) on the rod from each spring is dependent on both the distance from the fulcrum and the angles of both the rod and the spring, since only the component of the spring force acting perpendicular to the rod has an effect. This component, r, is given by:

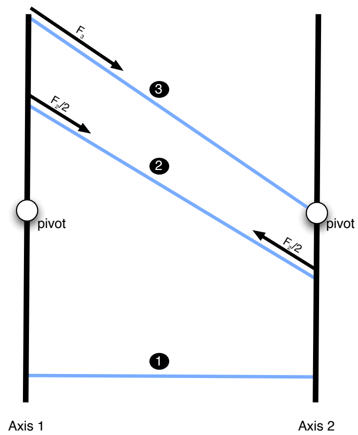
$$\vec{r} = v - \frac{\vec{v} \cdot \vec{u}}{\parallel \vec{u} \parallel^2} \vec{u} \tag{2}$$

where $\vec{v} = \vec{a} - \vec{p}$ is the vector from p to a and $\vec{u} = F(\vec{b} - \vec{a})$ is the force at point a.

The torque τ on the axis-rod from this component is then simply

$$\tau = \parallel \vec{r} \parallel \parallel \vec{v} \parallel \tag{3}$$

The resultant torque is calculated by summing over all springs on each side of the pivot. Figure 1 shows an example: spring 1 has $d = r$ and hence exerts no force via 1. Spring 2 exerts a force on both axes, while spring 3 exerts force only on the first axis, since for the second axis $\parallel \vec{r} \parallel = 0$.



[h]

Figure 1: By considering the axes as rods, the lines between as springs and fixing a pivot point, we can determine the forces on each axis from each spring and hence the resultant angular acceleration of the axes. In this diagram, spring 1 is at its rest length and hence exerts no force on either axis. Spring 2 exerts a force on each axis proportional to the length of the spring and the distance from the pivot. Accordingly, spring 3 exerts a force on axis 1 but no force on axis 2, since the distance from the pivot is 0.

3.3 Numerical Integration

Since we are concerned more with stability and convergence than accuracy, a Verlet scheme [27] is used to perform the integration. We determine angular acceleration α via the rotational form of Newton's Second Law:

$$\tau = I\alpha \tag{4}$$

where I is the moment of inertia, and then apply:

$$x' = 2x - x^* + \alpha \cdot \Delta t^2 \tag{5}$$

$$x^* = x \tag{6}$$

where x^*, x and x' are the previous, current and new positions of the end point of the rod and Δt is the time step.

3.4 Pivot Position

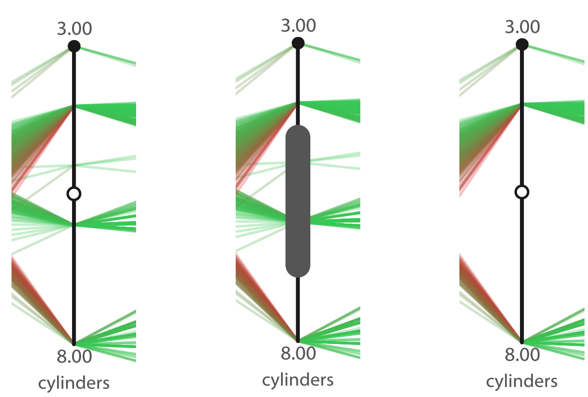
The pivot for each axis can be placed at an arbitrary point on the length of the rod. Placing it at either extreme (top or bottom) means the axis rotation is in an inward direction with respect to the next axis. Placing the pivot in the center of the rod gives a system most akin to a cable-stayed bridge. However, placing the pivot at the mean value for that axis gives some interesting results: for a data value mapped to a location x on the rod, the distance to pivot is then $x - \bar{x}$. This is equivalent to mean-centering the data. For this paper, we typically set the pivot for each axis at the mean but also allow the user to choose to use the mid-point of the axis instead, to match more closely with our bridge metaphor.

Axis rotation occurs only when the resultant torque (clockwise vs counter-clockwise) is non-zero. By mean-centering the data, the rotation that occurs in the system without user interaction can be minimized. Figure 2 shows some examples of correlated data displayed in three different forms: scatterplots, PCPs and force-directed PCPs. Even for skewed, highly correlated data, little axis rotation is observed. Since rotation is directly observable by the user, this reinforces the physical nature of the system while avoiding distracting the user (and complicating interactions) by having continual axis rotation.

The situation with more than two axes is more difficult to interpret. Small rotations in each preceding axis can result in a large rotation for axes in the center of the plot. We address this through interactions as described in the following section, but initial axis order still plays an important role. While we do not address in this work the problem of an appropriate axis order, we support axis rearrangement through user interaction using a simple drag-and-drop metaphor.

4 Interactions

As a static image, a force-directed PCP shows advantages over a standard PCP. Through interacting with the system, a user can gain an even better understanding of the underlying data. In this section, we present three new interaction techniques for force-directed parallel coordinates and give details of their implementation in our system.



(a) Before (b) During selection (c) After

Figure 4: Cutting is a filtering operation in our system that works by removing springs within the brushed region. The related springs linking other axes are also removed, which may trigger a shift in orientation of the axis. Here, springs are colored based on the force they exert, from low (green) to high (red).

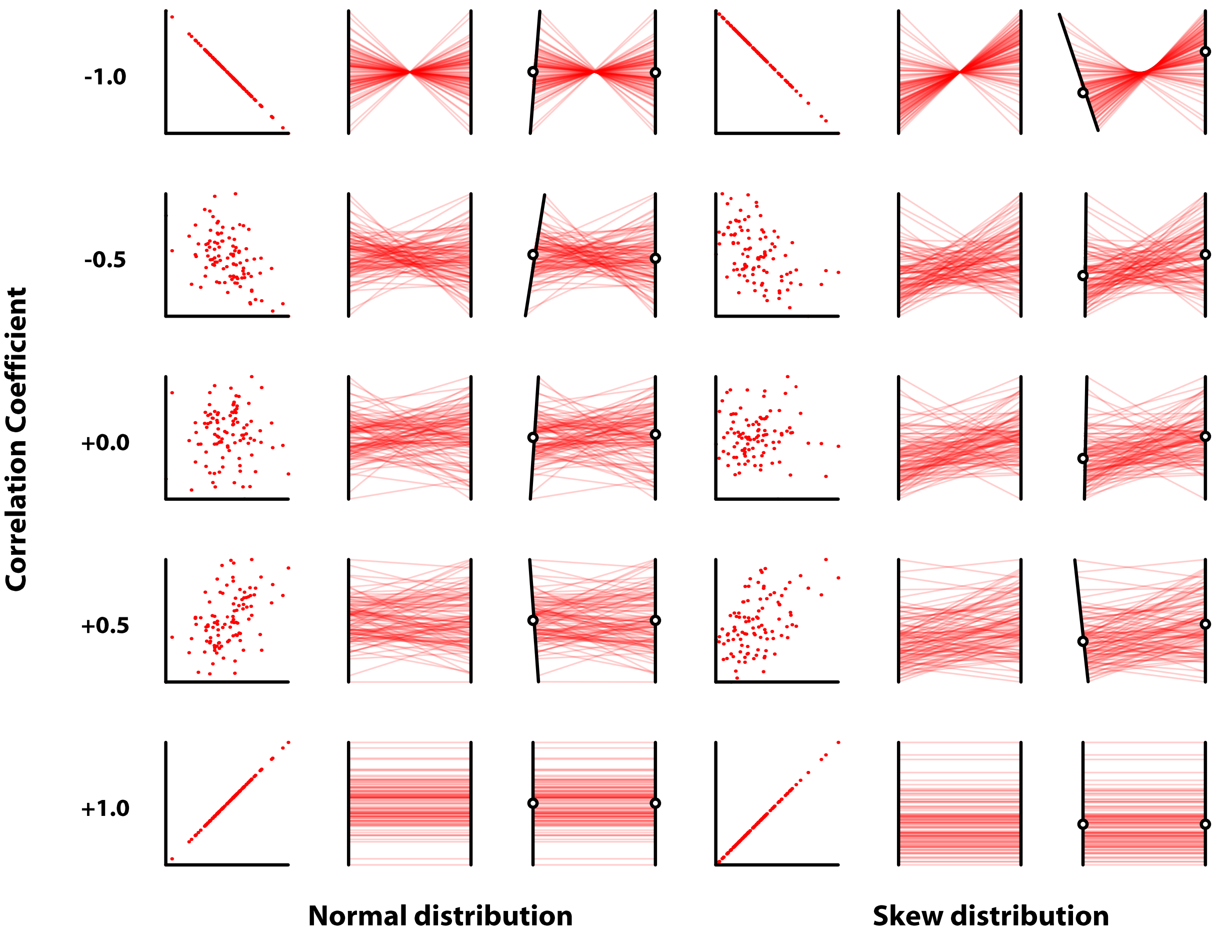


Figure 2: Comparisons between scatterplots, PCPs and force-directed PCPs for two dimensions using synthetic data generated with known skew and correlation. The first three columns show data with the given correlation where both variables follow a normal distribution, while the last three columns show data for which one axis has a skew-normal distribution and the other is normally distributed. In neither case is there a large movement of the axes, due to our choice of pivot position, and this places the focus for exploration on user interaction.

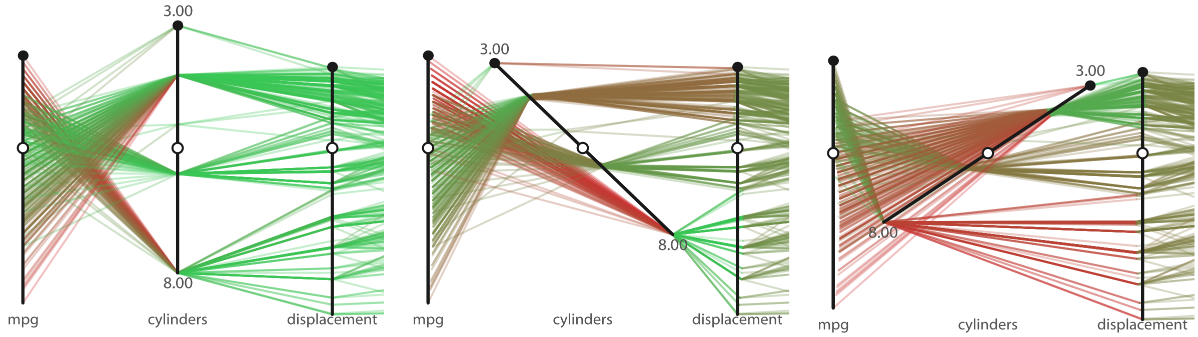
4.1 Cutting

Brushing on a PCP can be a filtering operation as well as a selection operation: polylines can be removed instead of simply colored. The natural analogue of this operation on a force-directed PCP is cutting. Selection on an axis can be used to determine springs to cut. These springs (and the complete polyline of which they form part) are then removed from the physical system, as shown in Figure 4. To return to our bridge metaphor, this is equivalent to cutting cables, and will result in a change in the forces on the axis and hence possibly a rotation. However, in our system, since the whole polyline (multiple springs) is removed, more than one axis may change orientation. These changes give information about both the data remaining and the data removed, and the animated transition in axis orientation may improve the perception of changes between the states [11].

Our implementation for this interaction is straightforward. First, the transformed start and end values for the selection are calculated. Next, springs in the system with a start or end point in this range are identified, and by following the chain of springs across axes in both directions, a list of springs to remove is compiled. Finally, these springs are all removed from the system.

4.2 Axis Swinging

The most natural interaction with any spring system is to allow pushing or pulling directly. Most systems, once released, will return to their original state. In our sys-



(a) Before (b) Swinging counter-clockwise (c) Swinging clockwise

Figure 3: Axis swinging allows the user to change the angle of any axis in the system interactively, by dragging it around. The initial state of the system is shown in Figure 3a, with springs again colored by force low-high as green-red. Swinging the axis counter-clockwise as in Figure 3b changes these the lengths of springs and hence the forces they exert. The same is true for Figure 3c for the other direction. In addition, the process of manipulating the axis gives some insight into the data represented by overlapping lines, as they change angle and color. Since color is mapped relative to the maximum and minimum forces in the current frame, the colors of springs joining other axes may also change.

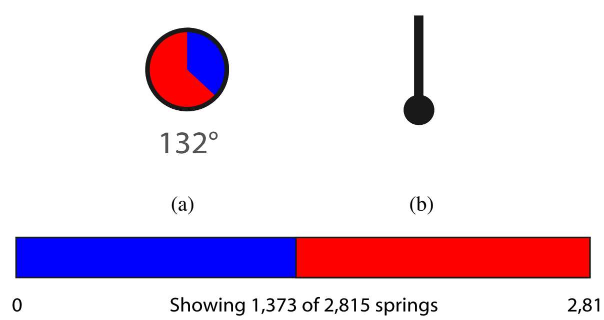
tem, pushing and pulling an axis to rotate manually accomplishes two tasks. First, it gives the user a better indication of the positions of springs connected to that axis, since they will appear to slide over each other (and, in our implementation, perhaps change color, if the forces change). This is similar in some respects to the idea of jittering in scatterplots. Second, it gives the opportunity for the system to return to a different state: if, as in the strong negative correlation examples in Figure 2, there is more than one stable state for the system, the user can force the system out of the local minima into the other state.

There are two cases within our implementation of axis swinging: with the physics engine active and with it paused. With the engine running, we simply calculate the new angle for the axis the user wishes to swing, and force it to that position. Releasing the mouse button releases the axis back into the system at its new position. With the physics system paused, we fix all the other axes in their current orientation, and update only the positions of springs connected to the axis under manipulation. These two cases were suggested by informal user feedback on our system: having the ability to force axes to angles with the system in motion was perceived as a very different operation to manipulating the system with all axes at rest, but both are considered useful.

4.3 Axis Pinning

For exploring the system, changes made to the axis can potentially affect another non-adjacent axis in an unexpected way. Perhaps a cutting operation results in an axis settling at a right-angle to its neighboring axis — the worst possible case for occlusion in our system. While we do not tackle the axis re-ordering problem directly, we offer a partial solution to this problem through interaction by giving users the ability to pin or freeze an axis in position.

This action splits the physical system into two separate systems at the point of the pinned axis. It can be repeated as many times as required to form smaller and smaller systems, giving the ability to examine purely local effects between adjacent axes without changing orientations elsewhere. Implementation is simple: the end points of the rod are fixed, which fixes the end points of connected springs also. Pinned axes are shown visually as translucent.



(a) (b) (c)

Figure 5: Instrumentation for force-directed PCPs. (a) shows the rotation angle for a particular axis, (b) denotes the low-value end of an axis and (c) indicates the number of springs currently shown against the original spring count.

5 Example

Having introduced the technique and interactions, in this section we present an example of using force-directed PCPs to examine a multidimensional data set. The focus

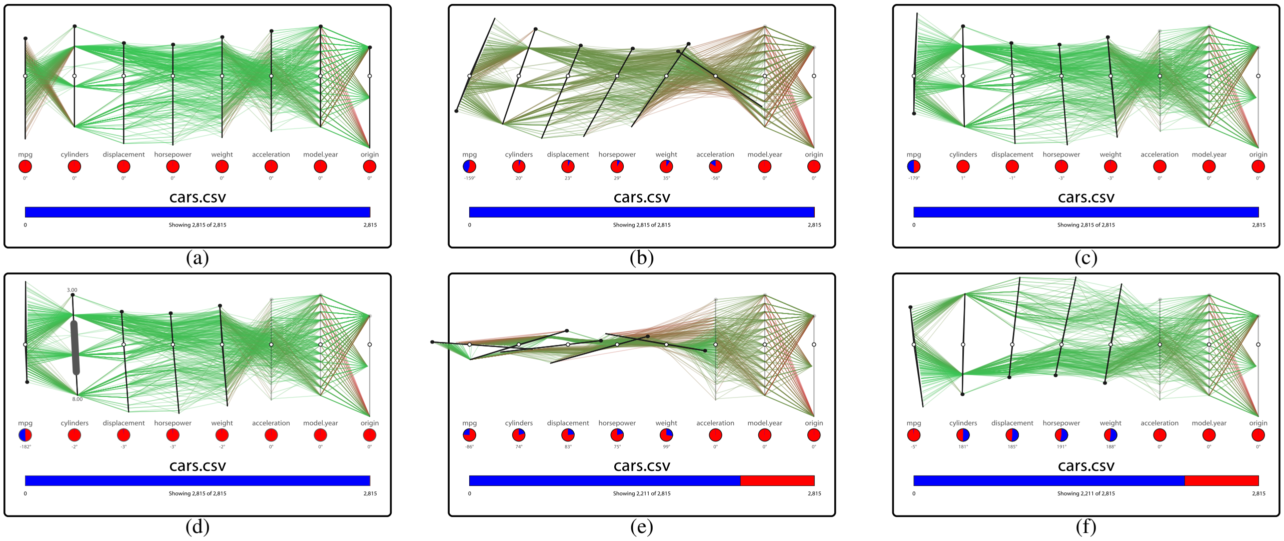


Figure 6: Analysis of the cars data set using a force-directed PCP. (a) shows the initial state, (b) the stable state (c) the result of swinging and pinning interactions, (d) a cutting operation (e) a transitional state before arriving to (f) the stable state.

of this example is less on the insights gained than on how the techniques described above can be used to explore the data set.

To facilitate comparison with existing work, we use the cars dataset from the American Statistical Association Data Exposition of 1983. As with all interaction techniques, they are difficult to fully illustrate on paper, therefore we highly encourage the reader to view the accompanying video. To help the user track the results of axis manipulation and cutting operations, we instrument the display of our plots with an angle tracker (Figure 5a) and spring count (Figure 5b), and denote the lower end of an axis as in Figure 5c.

5.1 Cars

The cars data set consists of 406 rows of 9 dimensional data on cars tested by the Consumer Reports magazine between 1971 and 1983. Of the 9 variables per car — make and model, fuel economy, cylinders, displacement, horsepower, weight, acceleration, model year and origin — we exclude make and model from our consideration. While PCP-like techniques can deal with categorical data, either by simply mapping each category to a numerical value or by other extensions [19], it poses a different set of problems to force-directed parallel coordinates, because the arithmetic average used to position the pivots is meaningless for a category variable mapped to a number range. For the origin dimension, which falls into this category, we avoid the problem by pinning the axis in our tool.

The starting point for our analysis is shown in Figure 6a. Axes are arranged in the default order, and some correlations are visible: MPG and cylinders seem to show some negative correlation, and displacement and horsepower are also correlated. Allowing the axes to swing freely gives a state as in Figure 6b. The MPG axis has inverted itself automatically. Its new orientation shows positive correlation with cylinders, but the color scheme is largely dictated by the springs in the weight-acceleration-model.year set. In fact, the acceleration axis settles at an angle approaching 90 degrees to its neighboring axes — the worst possible case for occlusion.

Here, we begin to interact with the data. We can swing the acceleration and model.year axes to an upright position and pin them there (Figure 6c). This gives a stable state for the system, but we decide to investigate further — we filter on the cylinder axis to remove cars with $4 < cylinders < 8$ by cutting (Figure 6d). This changes the forces on the weight axis sufficiently for it to invert, and this in turn triggers inversion of the axes to the left of it, passing through the state in Figure 6e before arriving in the state described in Figure 6f, which shows the remaining data split into two groups with a minimal amount of overlap. Our final conclusions from this state — that, looking at cars with less than 4 or eight cylinders, mpg is inversely correlated to cylinders which is directly correlated to displacement, horsepower and weight. The process described illustrates deeper exploration of attribute relationships.

6 User Study

We conducted a usability evaluation of our system using the same methodology as Claessen and van Wijk [3], with a total of 13 participants between the ages of 18 and 34.

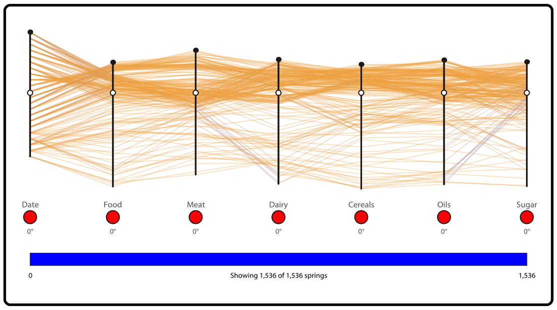


Figure 7: FAO Food Price Indices since 1990 data set, as used for the data exploration phase of the user study described in Section 6, springs colored by force low-high as orange-blue. While some patterns are visible in this image, such as the decline in food prices in the early 2000s, other, more complex patterns can be discovered through interaction with the plot.

The gender split was 12 male to 1 female, all participants had normal color vision and 10 were already familiar with parallel coordinate plots.

Our experiment consisted of six phases: an introduction, discussing the ideas behind parallel coordinates, the differences between PCPs and FDPCPs and the purpose of the experiment; a demonstration of the new interaction techniques using the cars data set; a user exercise on the Iris data set; a second demonstration using this set; user exploration on a previously-unseen data set; and finally a survey to gather opinions and comments on this exploration.

In the user exercise, participants were asked to answer a set of seven questions:

1. What is the range of values of attribute Sepal width?
2. Is there a correlation between Sepal length and Sepal width?
3. Which value occurs most often for Sepal width?
4. What is the average value for Sepal width?
5. Is there a correlation between Sepal length and Petal length?
6. Is there a correlation between Sepal length and Petal width?
7. Which attribute(s) lend themselves for classification?

It should be noted that question 3 (determining the mode for sepal width) is not easily answerable using FDPCPs - this was included as a test of their understanding of the plots, and discussed during the second demonstration. Question 3 aside, participants were able to answer all questions correctly. The user exploration phase was performed as a 'think-aloud' exercise: participants explored a data set of world food prices since 1990 — overall index for food, and separate breakdowns for dairy, cereal, meat, oils and sugar, as shown in Figure 7 — while describing their use of interactions and the expected result. The role of the experimenter was to watch and record their statements and activities. This data set includes a number of interesting patterns, such as the 2007–2008 world food price crisis.

During the experiment, we observed a number of approaches to the exploration task. Several participants preferred to work primarily with the physical system turned off, and enable it only to see the affect of significant cutting or axis rotations on other axes. Color was used as a cue to identify data points of possible interest for further exploration. Brushing was used more often than cutting: participants recognized that cutting in the current system is an irreversible operation, and preferred to explore thoroughly before removing data in this fashion.

The survey on completion of the exploration task contained a number of statements on which participants could offer an opinion using a 5-point Likert scale. The results of this survey are shown in Table 1.

	++	+	o	-	- -
FDPC					
– system is easy to use	4	5	2	2	
– technique is easy to understand	6	6	1		
Helped to understand the data					
– Watching movement of axes	3	6	3	1	
– Moving axes manually	6	5	2		
– Cutting springs	10	2	1		
– Pinning axes	6	6	1		
– Marking springs	6	3		3	1
– Coloring springs by length	5	2	1	3	2
Added value					
– moving over fixed axes	4	6	3		
– FDPCPs over PCPs	6	6	1		

Table 1: Summary of survey responses from user study, with the encodings Strongly agree (++), Slightly Agree (+), Neutral (o), Slightly Disagree (-), Strongly Disagree (- -). Participants were generally positive about the new interactions with the plots.

In the comments sections, participants considered the strength of FDPCPs to be in identification of correlations and trends (mentioned explicitly by 6 participants). Other comments concerned the usefulness of interactions - participants felt that the natural movement of axes in the system was less useful than rotation driven by the user. The ability to toggle the physical system on and off was also mentioned as a strength: while participants were in general agreement that the physical model and interactions helped gain understanding, these interactions were complicated if the system was constantly in motion.

With respect to weaknesses, we received much useful feedback concerning elements of the interface. Participants wanted the ability to adjust pivot points and disliked that cutting springs was a one-way operation with no way to reverse the process without resetting the system. The interactions required very precise mouse input, and this interfered with the naturalness of the exploration.

In summary, the participants were positive about both the technique and the prototype system, and suggested a large number of useful improvements to the software.

7 Discussion

This paper introduces force-directed parallel coordinates, a model and set of related interactions for multidimensional data based on a physical interpretation of parallel coordinate plots. We have defined and demonstrated cutting, axis swinging, and axis pinning for these plots, and our user study indicates that users feel that these interaction techniques are of benefit in exploring a data set. While these techniques are described here, we strongly recommend that the reader view the accompanying video to see them in operation.

7.1 Limitations

While these examples demonstrate the usefulness of our interaction techniques, force-directed PCPs have some limitations. The initial state of the system is dependent on axis ordering, axis scaling and pivot position. Given an unfavorable axis ordering, these interaction techniques may reveal little concerning the data. However this is true for parallel coordinates visualizations in general and not specific to our case only. While our choices of pivot position and axis scaling are made to produce initially stable states and minimize occlusion, they may not be the best choice for other data sets with different characteristics.

The current implementation is also unsuited to large data sets, due to both axis ordering as above and also computational complexity. The chief consequence of the change in complexity is that interactivity is quickly lost: cutting and pinning can act on a paused system, but axis swinging is far less useful as an operation when each movement requires lengthy computation before an update.

7.2 Future Work

There is much interesting future work to be completed on this system. Integration of our system with other existing techniques to address its limitations in areas such as axis re-ordering would be beneficial, as would interactions like angular cutting and perhaps other physical interactions, such as user-adjustment of pivot position by sliding axes up or down. Our implementation of the physical system could be optimized by performing many of the mass-spring calculations on the GPU [8], which would enable consideration of larger data sets. It may be possible, through careful configuration of the system, to mimic statistic measures in a physical fashion. From a computational perspective this would be inefficient, but having a physical representation of, say, principal component analysis expressed as a set of axis rotations could be valuable to user understanding. We are also currently engaged in a larger scale, crowdsourced user study [10] comparing force-directed PCPs to standard PCPs, which should help us to quantify possible benefits for a number of tasks and data sets.

Acknowledgements

This work was supported by RIVIC (Wales Research Institute of Visual Computing) funded by Higher Education Funding Council for Wales (HEFCW).

References

[1] M. Ankerst, S. Berchtold, and D.A. Keim. Similarity clustering of dimensions for an enhanced visualization of multidimensional data. In *infovis*, page 52. Published by the IEEE Computer Society, 1998.

[2] J. Blaas, C. Botha, and F. Post. Extensions of parallel coordinates for interactive exploration of large multi-timepoint data sets. — *IEEE Transactions on Visualization and Computer Graphics*, pages 1436–1451, 2008.

[3] Jarry H.T. Claessen and Jarke J. van Wijk. Flexible linked axes for multivariate data visualization. *Visualization and Computer Graphics, IEEE Transactions on*, 17(12):2310 –2316, dec. 2011.

[4] A. Dasgupta and R. Kosara. Pargnostics: Screen-Space Metrics for Parallel Coordinates. *Visualization and Computer Graphics, IEEE Transactions on*, 16(6):1017–1026, 2010.

[5] G. Di Battista, P. Eades, R. Tamassia, and I.G. Tollis. Algorithms for drawing graphs: an annotated bibliography. *Computational Geometry-Theory and Application*, 4(5):235–282, 1994.

[6] T.M.J. Fruchterman and E.M. Reingold. Graph drawing by force-directed placement. *Software: Practice and Experience*, 21(11):1129–1164, 1991.

[7] Z. Geng, Z. Peng, R.S. Laramee, J.C. Roberts, and R. Walker. Angular histograms: Frequency-based visualizations for large, high dimensional data. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2572–2580, Dec 2011.

[8] J. Georgii and R. Westermann. Mass-spring systems on the GPU. *Simulation Modelling Practice and Theory*, 13(8):693–702, 2005.

[9] H. Hauser, F. Ledermann, and H. Doleisch. Angular brushing of extended parallel coordinates. In *Information Visualization, 2002. INFOVIS 2002. IEEE Symposium on*, pages 127–130. IEEE, 2002.

[10] J. Heer and M. Bostock. Crowdsourcing graphical perception: using mechanical turk to assess visualization design. In *Proceedings of the 28th international conference on Human factors in computing systems*, pages 203–212. ACM, 2010.

[11] J. Heer and G. Robertson. Animated transitions in statistical data graphics. *IEEE Transactions on Visualization and Computer Graphics*, pages 1240–1247, 2007.

[12] I. Herman, G. Melançon, and M.S. Marshall. Graph visualization and navigation in information visualization: A survey. *Visualization and Computer Graphics, IEEE Transactions on*, 6(1):24–43, 2000.

[13] P. Hoffman, G. Grinstein, K. Marx, I. Grosse, and E. Stanley. DNA visual and analytic data mining. In *Proceedings of the 8th conference on Visualization'97*, pages 437–ff. IEEE Computer Society Press, 1997.

[14] D. Holten. Hierarchical edge bundles: Visualization of adjacency relations in hierarchical data. *IEEE Transactions on Visualization and Computer Graphics*, pages 741–748, 2006.

[15] A. Inselberg. *Parallel coordinates: visual multidimensional geometry and its applications*. Springer-Verlag New York Inc, 2008.

[16] A. Inselberg and B. Dimsdale. Parallel coordinates: a tool for visualizing multi-dimensional geometry. In *Proceedings of the 1st conference on Visualization'90*, pages 361–378. IEEE Computer Society Press, 1990.

[17] J. Johansson and M. Cooper. A screen space quality method for data abstraction. In *Computer Graphics Forum*, volume 27, pages 1039–1046. Wiley Online Library, 2008.

[18] R. Kosara. Indirect multi-touch interaction for brushing in parallel coordinates. In *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 7868, page 7, 2011.

[19] R. Kosara, F. Bendix, and H. Hauser. Parallel sets: Interactive exploration and visual analysis of categorical data. *IEEE Transactions on Visualization and Computer Graphics*, pages 558–568, 2006.

[20] R. Kosara, H. Hauser, and D. Gresh. An interaction view on information visualization. *State-of-the-Art Report. Proceedings of EUROGRAPHICS*, 2003.

[21] A.R. Martin and M.O. Ward. High dimensional brushing for interactive exploration of multivariate data. In *Proceedings of the 6th Conference on Visualization'95*, page 271. IEEE Computer Society, 1995.

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

[23] H. Siirtola. Direct manipulation of parallel coordinates. In *Information Visualization, 2000. Proceedings. IEEE International Conference on*, pages 373–378. IEEE, 2000.

[24] H. Siirtola and K.J. Räihä. Interacting with parallel coordinates. *Interacting with Computers*, 18(6):1278–1309, 2006.

[25] A. Tatu, G. Albuquerque, M. Eisemann, J. Schneidewind, H. Theisel, M. Magnork, and D. Keim. Combining automated analysis and visualization techniques for effective exploration of high-dimensional data. In *Visual Analytics Science and Technology, 2009. VAST 2009. IEEE Symposium on*, pages 59–66. IEEE, 2009.

[26] C. Tominski, J. Abello, F. van Ham, and H. Schumann. Fisheye Tree Views and Lenses for Graph Visualization. In *Information Visualization, 2006. IV 2006. Tenth International Conference on*, pages 17–24. IEEE, 2006.

[27] L. Verlet. Computer "Experiments" on Classical Fluids. I. Thermodynamical Properties of Lennard-Jones Molecules. *Physical Review*, 159(1):98–103, 1967.

[28] M.O. Ward. Xmdvtool: Integrating multiple methods for visualizing multivariate data. In *Proceedings of the Conference on Visualization'94*, pages 326–333. IEEE Computer Society Press, 1994.

[29] E.J. Wegman. Hyperdimensional data analysis using parallel coordinates. *Journal of the American Statistical Association*, 85(411):664–675, 1990.

[30] P.C. Wong and R.D. Bergeron. Multiresolution multidimensional wavelet brushing. In *Proceedings of the 7th conference on Visualization'96*, pages 141–ff. IEEE Computer Society Press, 1996.