A Scalable Method for Visualising Changes in Portfolio Data

Tim Dwyer

School of Information Technology,
University of Sydney,
Madsen Building F09
University of Sydney
NSW 2006
Australia
Email: dwyer@it.usyd.edu.au

Abstract

In this paper techniques from multidimensional scaling and graph drawing are coupled to provide an *overview-and-detail* style method for visualising a high dimensional dataset whose attributes change over time. The method is shown to be useful for the visualisation of movements within a large set of fund manager's stock portfolios.

Keywords: portfolio, multidimensional scaling, principal component analysis, graph, visualization, stock-market

1 Introduction

Perhaps due to the conservative nature of capital market research or possibly an unwillingness to publish profitable techniques, methods for visualising financial and capital market data tend to be unambitious. Most "state-of-the-art" stock market visualisations are:

- variations on time-series charts which show variations in price of a small number of shares over time (Tegarden 1997)
- multidimensional scalings showing correlations between share price or other attributes (Brodbeck, Chalmers, Lunzer & Cotture 1997)
- glorified pie charts showing market share of sectors or stocks (Jungmeister & Turo 1992).

This paper is an attempt to push the boundaries by showing that it is possible to visualise the changing interests of thousands of fund managers over a period of time in such a way that an analyst can potentially diagnose the state of the market, learn from the apparent behaviour or even identify illegal activities. Further more, even though the technique was conceived for visualising such portfolio data it will have application in any field where the changing attributes of a large set of high dimensional data needs to be studied.

The dataset that inspired this research was UK stock market registry data in which the changing portfolio contents of all the registered fund managers is held at a granularity of approximately one month. The data contains in the order of 3,000 portfolios which are made up of a selection of over 2,000 different stocks from 54 different market sectors. To obtain a useful picture of the movements within this dataset at least twelve months' worth of this data must be available. Obviously, presenting such a large body of data to an analyst in a visual form is going to challenge their "perceptual bandwidth". In (Dwyer & Eades 2002) I presented a graph-visualisation based method for visualising a relatively small subset of this data over a few time periods. However, the method did not scale very well to larger samples. (Card, Mackinlay & Schneiderman 1999) describe various methods for allowing users to effectively navigate through large datasets by showing both an over-all view and a detailed view. In the over-all view, or overview, large scale patterns can be observed. Users can then select a portion of the overview and "zoom-in" to obtain a more detailed view. In this paper this *overview-and-detail* concept is used to produce a more scalable visualisation method. To provide the overview for the dataset a visualisation based on Principal Component Analysis (PCA) is introduced. The key difference between this PCA based method and other multidimensional scaling approaches is that changes over time are clearly shown. This overview can then be coupled with my earlier graph visualisation based methods to allow the user to "zoom in" to the finest level of detail available.

Section 2 provides some background on multidimensional scaling and the PCA method used in this paper. In sections 3 and 4 a novel system for visualising the results of the PCA dimension reduction is introduced. Section 5 gives a formal definition of the graph used in the detailed view and section 6 describes the layout algorithm used to find an embedding of the graph for visualisation.

2 Multidimensional Scaling

As mentioned above, the dataset consists of several thousand portfolios each of which contains a selection of shares from different companies belonging to different market sectors. If there are 2,000 different companies one can think of these portfolios as being points in a 2,000-dimensional space. Alternatively, they can be placed by market sector in a 54-dimensional space. Obviously, to make an intelligible visualisation this high-dimensional space must be reduced to two or three dimensions. This is the domain of Multidimensional Scaling (Borg & Groenen 1997), the chief aim of which is to find a lower dimensional representation for a high dimensional dataset that preserves, as much as possible, the relative Euclidean distances between the data points. Typically, this is achieved by minimising a *stress* function for the entire data set. That is, multidimensional scaling seeks to find a mapping from an n-dimensional set of proximities between pairs of data points p_{ij} to a configuration of the data X in an m-dimensional visualisable space: $f : p_{ij} \rightarrow d_{ij}(X)$ where $n >> m$. Stress can then be defined as the sum of squared errors between distances in a possible X and the ideal mapping:

$$\sigma = \sum_{(i,j)} (f(p_{ij}) - d_{ij}(X))^2$$

Usually an iterative approach, such as functional majorisation, is used to find X such that σ is minimised. Obviously, for n data points such a method is going to require at least $O(n^2)$ operations per iteration to consider the distances between each pair of points. In the dataset considered in this paper there are in the order of 3,000 portfolios with a data point for each of at least 12 monthly samples. Therefore, the multidimensional-scaling approach needs to be able to scale up to around 36,000 data points. It should also have a processing time that will allow for interactive zooming. For such a large dataset where the dimensionality of the data is significantly lower than the number of data points it is more practical to use a method based on Principal Component Analysis (Borg & Groenen 1997). In PCA the complexity is based on the dimensionality of the data rather than the number of points.

PCA aims to find the axes of greatest variance (*principal components*) through our m-dimensional data. The data can then be projected onto the plane defined by two such axes to obtain a two dimensional representation which captures this variance. Of course, this does not guarantee to minimise σ but hopefully an adequate visualisation will be obtained in reasonable time.

The approach used to find the principal components is fairly standard. There are m possible stocks or market sectors, l different portfolios, k different temporal samples for each portfolio and therefore $n = k \cdot l$ data points. The first step is to place all of this data in an $m \times n$-matrix A such that the columns are the dimensions and the rows are the data points. The next step is to find the covariance matrix C of our data. This is done by transposing the data so that the barycentre of the points is at the origin:

$$B_{ij} = A_{ij} - \frac{1}{n}\sum_{j=1}^{n} A_{ij}$$

and multiplying the resulting matrix B by its transpose to find C:

$$C = \frac{1}{n}BB'$$

Next, an eigen decomposition is obtained such that $C = Q\Lambda Q'$ where Q contains the eigenvectors $q_1 \vec{.} . m$ and Λ is a diagonal matrix of the corresponding eigenvalues. The two eigenvectors with the largest eigenvalues, q_1 and q_2 are then the principal components.

Finally, we can perform the projection to find x and y coordinates of the n data points in two dimensions:

$$x_i = \vec{p_i}' q_1$$
$$y_i = \vec{p_i}' q_2$$

where $i = 1, ..., n$.

3 Displaying temporal changes by extruding into 3D

The novel visualisation of the 2D PCA projection described above involves extruding the data into 3D such that time is now represented by the 3rd dimension. The result is a mass of "worms" crawling through time. Visualisations in which the third dimension is used in a significantly different way to the other two dimensions are commonly called $2\frac{1}{2}$D and I will use this terminology in the sequel.

Each of the data points comes from one of k samples, usually taken at regular intervals in time. Each of the n data points above is now assigned a z coordinate such that $z = c \cdot (i - k/2)$ where $i = 0...(k-1)$ and c is a constant scale factor. When drawing, each pair of adjacent points representing the same portfolio is connected with a line segment to produce the worms.

It is worth noting that in any type of multidimensional scaling the position of data points in the reduced dimensional space relative to the axes is meaningless. The important thing is that clusters and outliers are clearly visible. In the "worm" visualisation an analyst can see how various portfolios move in and out of clusters over time. This also allows colour and thickness of the lines to be used to display additional attributes as they change over time.

The resulting visualisation is shown in Figure 1. The collection of dots on the right-hand side is a cross section of the $2\frac{1}{2}$D worm view. The particular time period shown in cross section is selected by moving the transparent blue disk (the "water-level") on the left. Colour and alpha-transparency of the worms and their corresponding points in the slice view are used to convey extra attributes. In the figure colour is used to indicate whether a portfolio has increased or decreased in value between time periods. The worms are shaded from dark to light green to show an increase in value or dark to light red to show loss. Solid black indicates the portfolio has broken even. The total value of the fund is indicated by transparency. In the figure transparency potentially ranges from invisibility, indicating a zero value fund, to complete opacity, indicating the fund is worth more than £100,000. The other user interface features provided by the interface are described in more detail in Section 4.

An interesting feature of the data that can be seen in Figure 1 is that fund managers generally tend to move either toward or away from a single central cluster. Currently, our analysis suggests that this cluster contains portfolios that are weighted closer to the market index. That is, more conservative portfolios that are weighted to hedge against significant volatility. The fact that outliers tend to move either towards or away from this central cluster (rather than around it) may suggest that when they are confident (possibly buoyed by a stronger market) they move towards more volatile, higher risk, stocks and when the market is less stable they move back towards the safer "blue-chip" shares in the index. Of course it may also be an artifact of PCA. Work is continuing to investigate this phenomenon in greater depth.

4 User Navigation

An essential part of any 3D (or $2\frac{1}{2}$D) visualisation is providing the user with the ability to freely rotate, zoom-in or otherwise "fly" around the 3D model. When viewing a static projection of a 3D visualisation the user has no sense of depth and the extra dimension is wasted[1]. In our system this capability is provided in a fairly standard way with mouse interaction.

However, PCA gives us the ability to navigate around the dataset in some fundamentally different ways. Using the two eigenvectors associated with the two largest eigenvalues to define the projection plane by definition captures the greatest variance in the data. However, one can just as easily use any pair of eigenvectors. Allowing the user to cycle through the largest eigenvectors to choose the projection plane provides a high-dimensional rotation which may capture different aspects of variation.

[1] Indeed it is difficult to properly show the use of 3D visualisation in a paper with only static black and white figures!

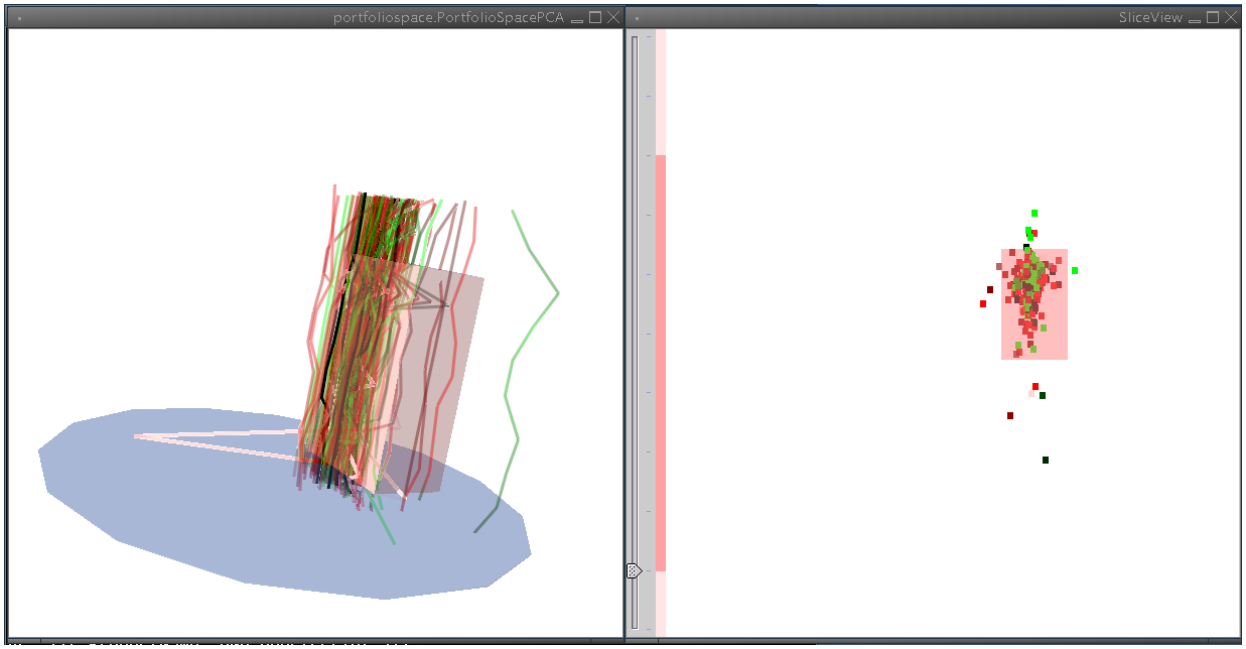

Figure 1: A *worm-view* visualisation showing the movements for the 391 largest fund managers over a 12 month period.

A zoomed view is also easily obtained by producing another PCA projection of a subset of the visible data. In Figure 1 the user is in the process performing such a zooming operation. In the cross-section on the right hand side they have selected a subset of portfolios by sweeping out a rectangular area with the mouse. They can then select a subset of the available time samples by moving the water-level up or down, thus creating the transparent box seen in the $2\frac{1}{2}$D view on the left-hand side of the figure. The selected subset of the data will then be re-projected as before and the zoomed view shown in a new window. The zoomed window is shown in Figure 2. An example where a similar PCA based rotation and zooming strategy is used to view high dimensional graph embeddings is given in (Harel & Koren 2002).

5 Detailed Graph Visualisation Based View

The "worm" visualisation described thus far provides us with an overview of our data set. In order to capture broad movements across as much data as possible the PCA based dimensional scaling was a necessary, though severe, abstraction of the underlying detail. To visualise the detailed behaviour of an individual fund-manager as they re-balance their portfolio a different approach is required.

Suppose an analyst selects an individual worm from the overview for closer inspection. Effectively, they have isolated a set of data for a single portfolio over a number of time periods. In our dataset this includes share price data and the count of shares of a particular type held in the portfolio. That is, we have two matrices in which the columns are associated with market sectors (or any other aggregation of stocks, or individual stocks) and the rows are associated with each of the sample times (the examples shown here have 12 monthly samples). In the first matrix P we have share price data. When an aggregation of shares is used this will be the average unit price across the aggregate. The second matrix Q contains the counts of shares held in the portfolio.

These matrices could be visualised by simple 3D area charts. For example, Figure 3 shows the share price data P, in this case the average share price for each of $n = 50$ market sectors for a year's worth of data. Figure 4 shows the counts of shares held in a particular fund manager's portfolio across the same time period. Figure 5 charts the total value of the portfolio across the time period, where the value x at each month j is:

$$x_j = \sum_{i}^{n} P_{ij} Q_{ij}$$

The relatively flat curve in Figure 5 shows that by re-weighting the portfolio the fund manager has managed to more or less even out the volatility in the share prices. In visualisations like that of Figures 3 and 4 we can see a lot of activity taking place. However, one must ask whether it is possible to produce a visualisation which focuses an analyst's attention more specifically on the fund manager's movement between market sectors.

In (Dwyer & Eades 2002) I proposed a graph visualisation based approach for visualising the movements of fund managers between different stocks or market sectors (in the sequel I'll refer only to sectors). The graph for fund manager movement is defined as $G = (V, E)$ consisting of a set V of vertices representing sectors and a set E of edges where each edge $e(u, v) \in E$ represents "movement" of one or more fund managers from sector $u \in V$ to sector $v \in V$.

For the matrix Q of share counts in each sector we can construct a graph in which a vertex represents each non-zero column. Beginning with the second row we compare the values in each column against that column's value in the previous row. For each column (sector) showing a decreased holding we construct an edge to all the columns with an increased holding. To allow the user to focus on more significant movements they can adjust a threshold (τ_e) in increase or decrease of stock price below which no edge is created. We continue comparing each pair of rows to create *layers of edges* (ie, a set of edges for each time period) until all rows have been examined.

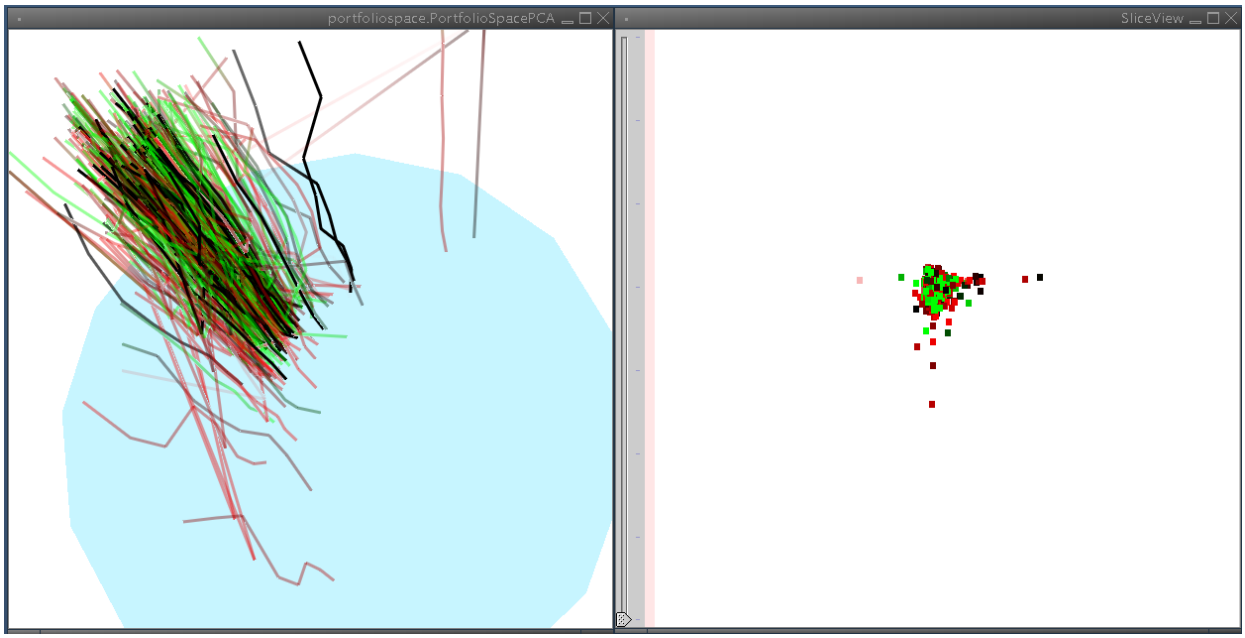

Figure 2: A zoomed view of the highlighted portion of Figure 1. 8 months are shown.

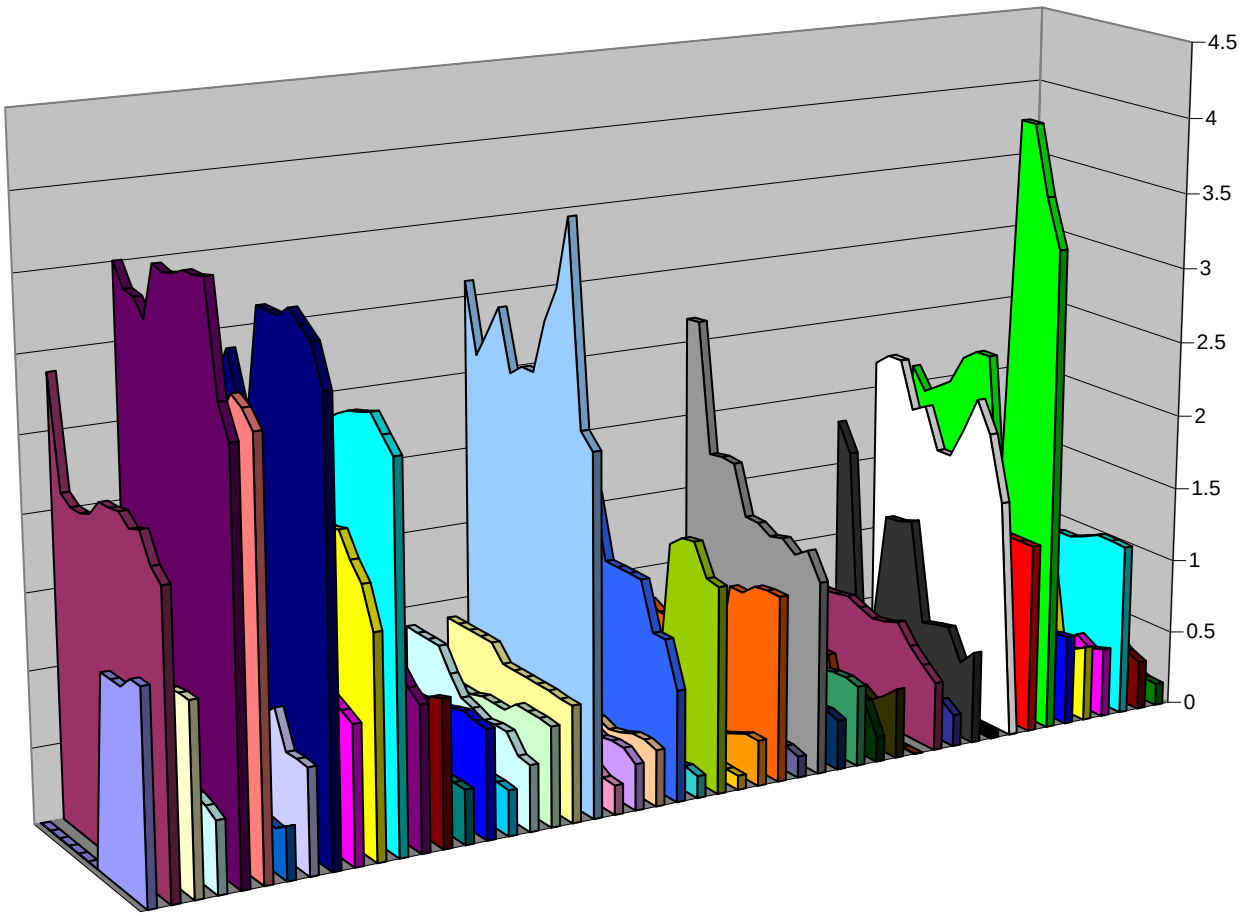

Figure 3: A 3D area chart showing the fluctuations in share price in a particular portfolio over 12 months. Each of the columns is a different stock, the depth axis is time (most recent at the front) and the vertical (labelled axis) shows share price in GBP (British Pounds Stirling).

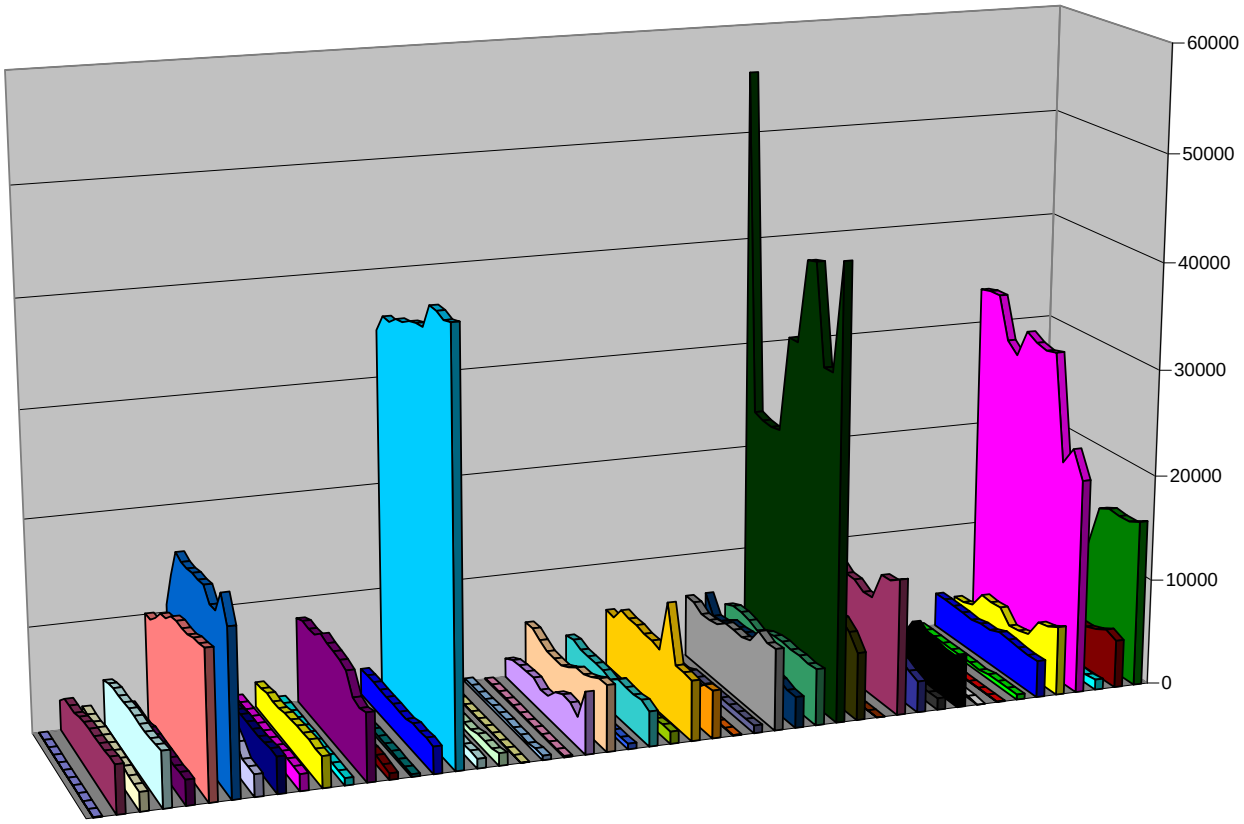

Figure 4: A 3D area chart showing the changing count of shares in the portfolio over 12 months. The axes are as in Figure 3 except the vertical axis which shows count of shares

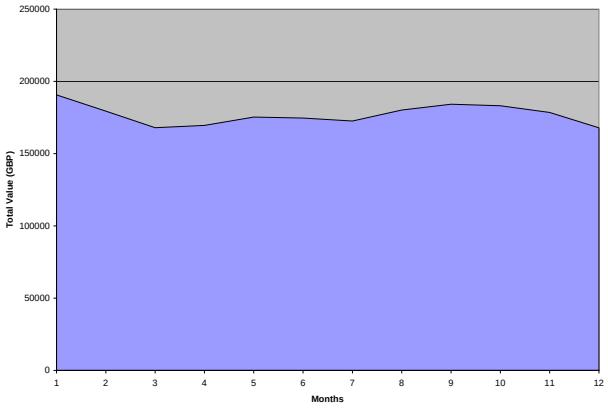

Figure 5: This chart shows the net effect on the total value of a portfolio of the share price fluctuations, shown in Figure 3, and the fund manager's re-weighting, shown in Figure 4.

To visualise this graph so that it is easy to see at what time different movements occurred it is possible to use a $2\frac{1}{2}$D paradigm similar to that followed for the worm view. That is, the graph drawing is extruded into the third dimension, see figures 8, 9 and 10. The vertices become pillars or columns parallel to the new third axis and the edges are placed perpendicular to the axis at a level dependent on the time (matrix row) at which the movement they represent occurred. The total value of a market sector at each point in time (ie, $P_{ij}Q_{ij}$) can be shown by setting the radius of the column vertex representing that market sector. Clutter in the graph may be reduced by only including vertices for which the maximum value is greater than a threshold τ_v. In other words, it is only necessary to include market sectors which make up a significant proportion of the portfolio. The changing average share price information from P can be shown by colouring each segment of the column. In the examples the columns are shaded towards green if there is an increase in average share price from one period to the next. The shading tends towards red if there is a decrease. The default colour, light grey, means there is no change.

In the extruded $2\frac{1}{2}$D view the edges can be shown as tubes or pipes between the columns. The total value of a given movement is shown by adjusting the radius of the edges. That is for an edge showing a change in holding from time samples k and $k+1$ between vertices representing columns i and j in P and Q the radius of the edge is set to:

$$radius \propto (Q_{i(k+1)} - Q_{ik})P_{i(k+1)} + (Q_{jk} - Q_{j(k+1)})P_{j(k+1)}$$

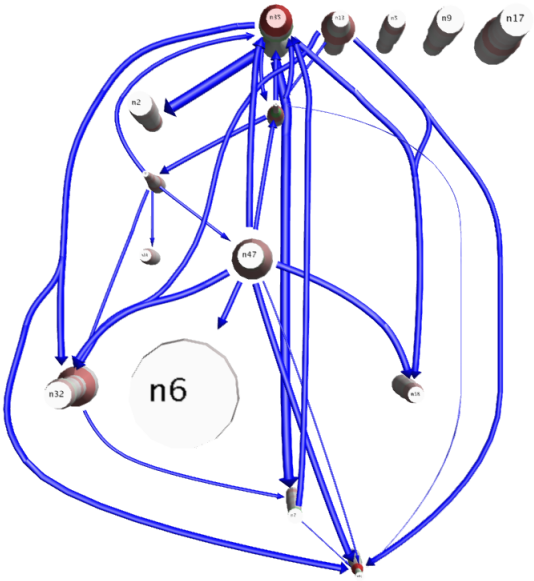

Figure 10: The stratified graph from 8 viewed from above.

6 Stratified Graph Layout

I have chosen to refer to such graphs in which edges appear in layers corresponding to a particular time period as "Stratified" graphs[2]. Visualisations of Stratified graphs have appeared in the literature from time to time but the question of how to arrange them is yet to be thoroughly examined. (Koike 1993) was possibly the earliest, using a stratified graph to show the flow of control in parallel software execution, but the author left placement of the columns representing vertices to the user. (Brandes & Corman 2002) chose to represent social group interactions as stratified graphs using the well known force-directed method simulating physical attractive and repulsive forces in an iterative algorithm that attempts to balance the opposing forces typically creating an aesthetically pleasing arrangement. In (Dwyer & Eades 2002) we used a similar force-directed method but allowed the columns to *bend* in order to further reduce the forces. We showed that this last variation was useful in highlighting clusters in the graph. A stratified portfolio graph arranged using the force-directed approach (without bendy columns) is shown in Figure 6.

In this paper another well known layout method is introduced to the problem of stratified graph layout: the Sugiyama (or Layered Graph Layout) Algorithm(Eades & Sugiyama 1990). The chief advantage of the Sugiyama algorithm over the other layout methods described is that it clearly shows *flow* in directed graphs. This is because the first stage of the algorithm creates a layering of the nodes such that net sources (nodes with mostly outgoing edges) are usually placed closer to the top and net sinks (nodes with mostly incoming edges) are placed nearer the bottom. The final arrangement thus ensures that most of the edges are downward pointing and net flow can be said to be from top to bottom.

This graph theoretic concept of flow is useful when considering layout of our portfolio movement graphs in that it roughly corresponds to the flow of money between different stocks. That is, source nodes represent stocks that are mostly being sold and sink nodes represent stocks that are more commonly being bought.

To generate the layout shown I use an implementation of the Sugiyama algorithm based on the Dot program that comes with AT&T's Graphviz package (`http://www.graphviz.org`). I have modified this algorithm to handle separate edge sets for each of the strata. Specifically this involved the following modifications:

- the edge concentration[3] method was changed so that edges on different strata would not be concentrated
- the crossing minimisation was changed such that only crossings between edges on the same strata are considered. The idea being that when edges on different strata overlap the crossing can be resolved by rotating in 3D

7 Conclusion / Further Work

I have demonstrated two visualisations for fund manager movement data which may be coupled to provide an holistic, overview and detail system. The first visualisation, the PCA worm view, compresses a great deal of information about the entire dataset into a single scene and takes advantage of the speed and flexibility of PCA to allow a user to quickly focus in on smaller regions of detail. The second visualisation paradigm, the graph based column view, brings together the most important information from all three charts shown in figures 3,4 and 5 into a single visualisation and draws an analysts attention to the features in which they are most interested. Particularly, it allows an analyst to directly see the correlation (if any) between stock price and a fund managers behaviour in re-weighting the portfolio.

In this paper a broad overview and definition of these two paradigms has been given. Work is progressing on validating the techniques by using them in a field study with industry experts.

The problem of extending existing layout algorithms such as the Sugiyama algorithm, to produce the best possible results for stratified graphs is also ongoing.

Finally, since the paradigms proposed in this paper should be applicable to any high-dimensional, multivariate dataset I hope in future to test their utility in other domains.

8 Acknowledgements

This work was conducted with the support of the Capital Markets CRC Limited. The author would also like to thank the members of the University of Sydney Information Visualisation Research Group for their comments and suggestions.

References

Borg, I. & Groenen, P. (1997), *Modern Multidimensional Scaling: Theory and Applications*, Springer Series in Statistics, Springer.

Brandes, U. & Corman, S. (2002), Visual unrolling of network evolution and the analysis of dynamic

[2] borrowing the geological term used to describe rock consisting of layers of sediment. I use this term to avoid confusion with the conventional usage of the term "layered graph drawing" associated with the Sugiyama algorithm

[3] Edge concentration involves condensing edges so that a single edge can split and branch and a single branching edge may be shared by a number of nodes

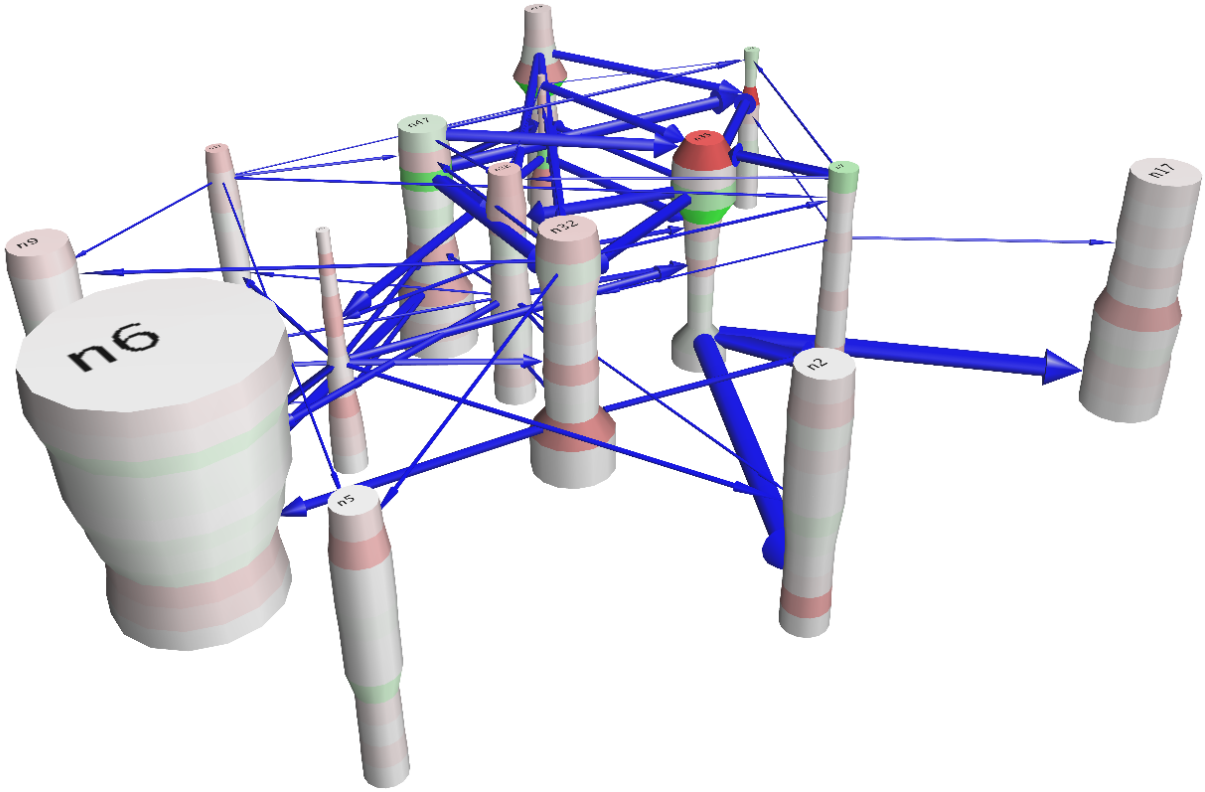

Figure 6: A stratified graph representing the movements in the portfolio shown in figures 3, 4 and 5, laid out using a basic force-directed algorithm. The thresholds are $\tau_e = \pm 5\%$ and $\tau_v = £5,000$.

discourse, *in* 'To appear in Proc. IEEE Symp. Information Visualization (InfoVis '02)', IEEE Computer Society.

Brodbeck, D., Chalmers, M., Lunzer, A. & Cotture, P. (1997), Domesticating bead: Adapting an information visualization system to a financial institution, *in* 'Proceedings of the IEEE Symposium on Information Visualization', pp. 73–90.

Card, S., Mackinlay, J. & Schneiderman, B. (1999), *Information Visualization: Using Vision to Think*, Morgan Kaufmann.

Dwyer, T. & Eades, P. (2002), Visualising a fund manager flow graph with columns and worms, *in* 'Proceedings of the 6th International Conference on Information Visualisation, IV02', IEEE Computer Society.

Eades, P. & Sugiyama, K. (1990), 'How to draw a directed graph', *Journal of Information Processing* **13**, 424–437.

Harel, D. & Koren, Y. (2002), Graph drawing by high-dimensional embedding, *in* 'Proceedings of Graph Drawing 2002', Vol. 2528, LNCS Springer-Verlag, pp. 207–219.

Jungmeister, W.-A. & Turo, D. (1992), Adapting treemaps to stock portfolio visualization, Technical Report UMCP-CSD CS-TR-2996, College Park, Maryland 20742, U.S.A.

Koike, H. (1993), 'The role of another spatial dimension in software visualization', *ACM Trans. Inf. Syst.* **11**(3), 266–286.

Tegarden, D. (1997), 'Business information visualization', *Communications of the Association for Information Systems* **1**(4).

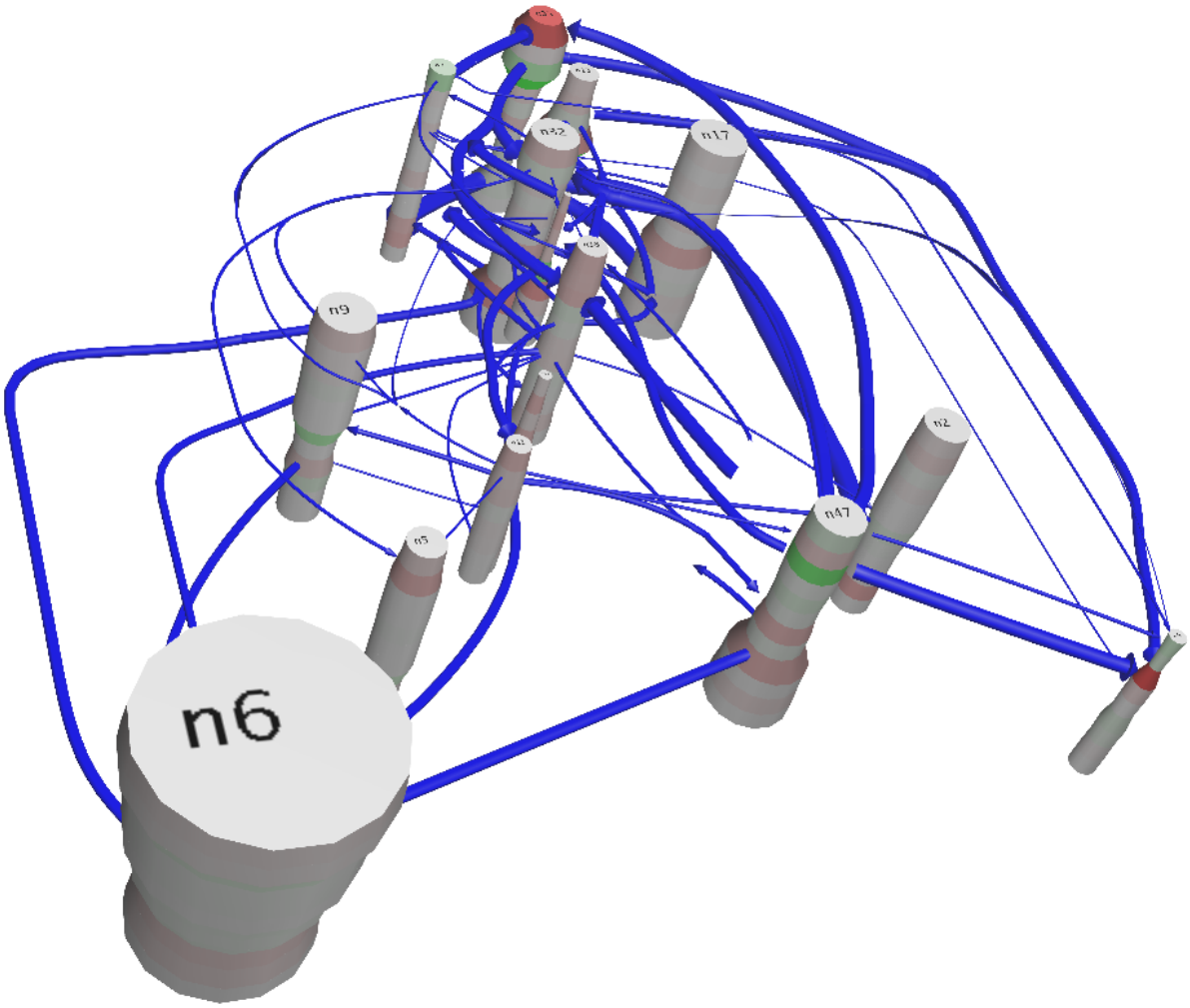

Figure 7: The stratified graph from 6 arranged using the Sugiyama algorithm.

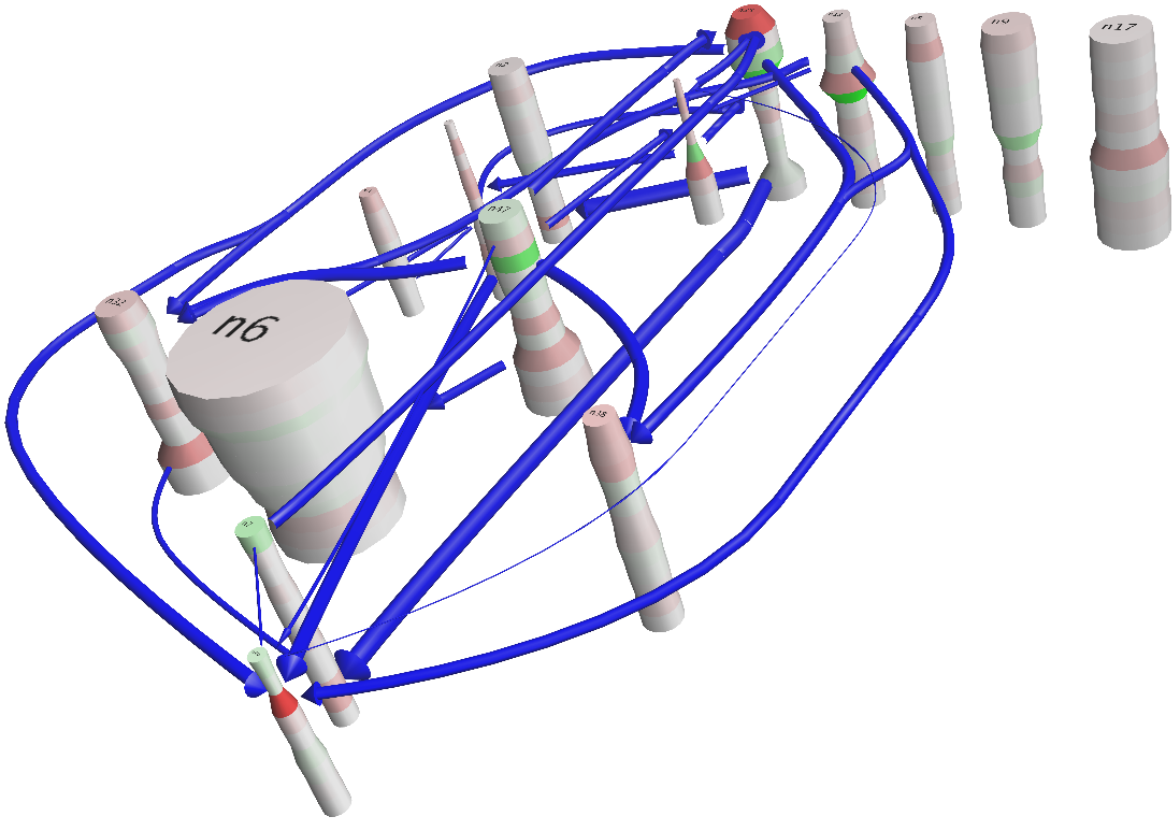

Figure 8: Another stratified graph showing the movements in the portfolio from Figure 7 in lower detail. In this figure fewer edges are visible since $\tau_e\pm$ has been raised to 8%.

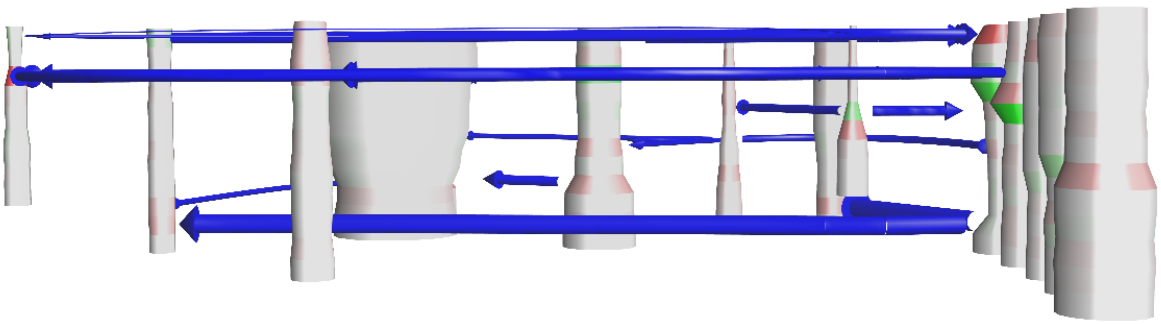

Figure 9: The stratified graph from 8 viewed from the side to better show the strata.