Visualizing Time-Varying Matrices Using Multidimensional Scaling and Reorderable Matrices

Ermir Qeli[1], Wolfgang Wiechert[2] and Bernd Freisleben[1]

[1]*Dept. of Math. and Computer Science, University of Marburg, D-35032 Marburg, Germany*
{ermir,freisleb}@informatik.uni-marburg.de

[2]*Inst. of Syst. Eng., Dept. of Simulation, University of Siegen, D-57068, Siegen, Germany*
wiechert@simtec.mb.uni-siegen.de

Abstract

In this paper, we present a novel approach to visualize time-varying matrices. This approach is based on combining multidimensional scaling and the reorderable matrix method. An adapted version of multidimensional scaling which allows the construction of similarity plots for columns/rows of time-varying matrices is proposed. In addition, we have extended the reorderable matrix method to allow the visual exploration of time-varying matrix data in a tabular form for being able to verify the results of MDS and possibly discover new patterns in data. The benefits of our approach are illustrated by showing visualizations of sensitivity matrices generated during simulations of metabolic network models.

Keywords: Visualization, Multidimensional Scaling, Reorderable Matrices, Time-Varying Matrices.

1. Introduction

Time-varying matrices are often encountered as a result of simulations or experiments, e.g. to study the sensitivity of parameters with respect to the outputs over time [22]. The possibly large dimensions of these matrices and the fact that they vary over time make their understanding difficult. However, one way to improve this situation is their visual exploration using appropriate visualization techniques.

In this paper, we present a novel hybrid approach consisting of multidimensional scaling and the reorderable matrix method to visualize time-varying matrices. Multidimensional scaling (MDS) [9, 14, 19] is a well-known non-linear dimension reduction technique. It is concerned with the construction of configurations of m points in Euclidean space using information about the distances between them. In the context of this paper, we consider N-dimensional vectors where N>3 and the results of MDS are 2D or 3D scatter plots representing the similarity between these vectors. Reorderable matrices [16, 17] allow the visualization of multidimensional data by mapping the values to colors, grayscale values or symbols. By reordering the rows and columns of matrices, different patterns in the data can be made visible. Since both methods were originally designed to work for a single point of time only, we have adequately extended them to enable time-varying matrix visualizations.

To demonstrate the benefits of our approach, it is applied to analyze time-varying sensitivity matrices generated during the simulation of metabolic networks [2, 22]. It works, however, for any kind of dynamic sensitivity matrices generated during other types of simulation or other time-varying matrices. The sensitivity matrices of our application example represent how the changes in the parameters of a metabolic network model affect the output of the model. These matrices are analyzed in order to find redundancies, which are then used to derive a simpler model with the same properties. Our approach allows to visually find similarities between rows/columns by using MDS, whereas the reorderable matrix method allows the visual extraction of patterns via interaction or with automatic reordering. The two visualization methods are synchronized with each other to allow the combined interaction with the user.

The paper is organized as follows. Section 2 gives a survey of related work in the field. Section 3 describes the individual steps of our hybrid approach to visualize time-varying matrices. Section 4 presents a case study for a set of sensitivity matrices, generated during the

simulation of a particular metabolic network model and shows how the visualization could be used to simplify the model of a metabolic network. Section 5 concludes the paper and outlines areas for future research.

2. Related Work

The related work relevant to the subject of this paper can be grouped into two categories:

- Visualization of time-varying data
- Visualization of static multi-dimensional data.

The visualization of time-varying data is commonly achieved using various approaches for visualizing time series, but these usually deal with univariate or vectorial data only. A survey of visualization techniques for time-dependent data is given in [26].

To the best of our knowledge, there are no approaches dealing particularly with the visualization of time-varying matrices. Several visualization methods for static multi-dimensional data have been proposed in the literature. Dimension reduction is one alternative to visualize multivariate data in 2D or 3D. There are two types of dimensionality reduction techniques: linear and nonlinear. *Principal component analysis* (PCA) is a linear projection method where the projection is formed as a linear combination of the input. *Multi-dimensional scaling* (MDS) and *Sammon´s mapping* are two related nonlinear projection methods, with the former method preserving large distances and the latter preserving small distances. A survey of these techniques can be found in [15]. In addition to dimension reduction techniques, there are several other approaches for visualizing multivariate data. For instance, the *reorderable matrix method* proposed by Bertin [16, 17] is a simple but robust approach to visualize tabular data. Specific permutations of rows and columns allow the user to find clusters in data. Minnotte and Webster [24] use reorderable matrices under another name, *data image,* to explore high dimensional data. Marchette and Solka [4] use the data images for outlier detection in data. *Corrgrams* proposed by Friendly [23] is an approach similar to the reorderable matrix method to visually explore correlation matrices, which are important in multivariate statistics.

Chernoff faces [11] represent multi-dimensional data by means of faces with changing attributes. Thus, the problem of finding similar vectors is converted into the problem of finding similar faces, which is somehow easier for the human eye. *Parallel coordinates* introduced by Inselberg and Dimsdale [1] allow visualizing multi-dimensional data in parallel axes. *Stardinates* proposed by Lanzerberger et al. [21] provide a similar approach where the axes are not parallel anymore, but arranged radially in a circle. *Star coordinates* is a similar approach proposed by Kandogan [8] for visualizing clusters and outliers. Siirtola in [12] combines parallel coordinates [1] with reorderable matrices [13, 16, 17] to visualize multi-dimensional data. *Andrews curves* [5] is a visualization method similar to parallel coordinates based on a transformation similar to a Fourier transformation.

3. Our Approach

As Roberts in [18] argues, single representation of data can often lead to misinterpretation of information. Furthermore, multiple visualization which complement each other help the user to see the data in different perspectives [3]. Thus, the basic idea of our approach to visually explore time-varying matrix data is to combine two visualization methods: multidimensional scaling and reorderable matrices. At first sight, these two methods do not look to be combinable at all. However, MDS allows viewing the similarity of columns/rows of input matrices and also serves as the master view of the visualization, but loses the connection to the original data. Reorderable matrices mixed with color visualization allow to visually verify the results of MDS, allowing at the same time a detailed view of the data. In the following, the relevant issues of our approach are presented.

3.1. Input Data

We assume that our input data are time-varying matrices stored in a CSV (Character Separated Value) file. The data is assumed to be appropriately normalized depending on the problem domain. This is the only preprocessing step, which is application specific. Assuming that this step has been completed, the procedure described below can be used for any kind of time-varying matrices. Currently, we use our approach to visualize matrices containing double-precision floating-point numbers, but it can be adapted to nominal data presuming that proper distance functions are used [10].

3.2. Multidimensional Scaling

Multi-dimensional scaling (MDS) is concerned with the construction of a configuration of n points in Euclidean space using information about the distances between these points. MDS is often used to project data nonlinearly from a high dimensional space to a

low dimensional one, usually a 2D or 3D space. The purpose of MDS is that the distances between points in the lower-dimensional space approximate the distances between points in the higher-dimensional space best. MDS allows the use of similarity/dissimilarity measures instead of strict distances and thus enables to flexibly view the relationships between data items.

We have implemented a modified version of the classical algorithm for *metric multi-dimensional scaling* [9, 14, 19]. This algorithm has the advantage of being robust and fast, which ensures good response times with time-varying data. The performance of the algorithm is strongly dependent on the procedure for finding the *eigenvalues* of the distance matrices (calculated in step 3). For the visualized sensitivity matrices, the time for calculating the MDS solution for one point in time ranges from tens of milliseconds to hundreds of milliseconds on a contemporary PC.

The implemented MDS algorithm consists of the following steps:

1. *Let X(t) for t=1 to Tmax represent the normalized time-varying matrices as described in section 3.1*
2. *Let the vectors $X_1(t) \ldots X_n(t)$ represent the columns of the matrix X(t) at time t.*
3. *Compute the matrix D(t) for t=1 to Tmax, where the elements d_{ij} of this matrix are calculated as follows:*
   - 3.1. *Alternative 1:* $d_{ij} = \left\| X_i(t) - X_j(t) \right\|$
   - 3.2. *Alternative 2:* $d_{ij} = \sqrt{c_{ii} - 2c_{ij} + c_{jj}}$ *where c_{ij} is the cumulative correlation coefficient between two vectors (explained below)*
4. *Construct the matrix A(t) where* $a_{ij}(t) = (-\frac{1}{2} d_{ij}^2(t))$
5. *Construct the matrix B where* $b_{ij} = a_{ij} - \bar{a}_{i\cdot} - \bar{a}_{\cdot j} + \bar{a}_{\cdot\cdot}$ *where $\bar{a}_{i\cdot}$ is the mean of row i, $\bar{a}_{\cdot j}$ is the mean of column j and $\bar{a}_{\cdot\cdot}$ is the overall mean*
6. *Compute the k largest eigenvalues $\lambda_1(t), \lambda_2(t), \ldots, \lambda_k(t)$ where k is the dimension (in our case, k=2 or k=3).*
7. *Get the corresponding k eigenvectors $v_1(t), \ldots, v_k(t)$ and normalize them by* $v_i'(t) \cdot v_i(t) = \lambda_i(t)$ *and take the normalized eigenvectors as a solution of MDS for time t*

Step 3 is the most important step. Two views of the data can be selected: normal view and cumulative view, and they define how the distance matrix is calculated. Alternative 1 (step 3.1) is normally used to visualize the matrices for the first time. However, there are cases when consecutive configurations differ significantly from each other. In these cases, to reduce this effect, the cumulative correlation should be used as a similarity measure between column vectors of the cumulative information matrix (step 3.2).

The cumulative information matrix of a set of time-varying matrices X(t) is calculated as $I(t) = \sum_{k=T\min}^{T\max} (X'(k) \cdot X(k))$. *I(t)* is converted by normalization to a correlation matrix which is then used as a similarity measure that cumulatively considers time-varying matrices. The time window from Tmin to Tmax allows more flexibility for the users so that they can take into consideration a variable number of consecutive matrices, e.g. from the beginning to time point t, or only time point t, etc. To illustrate the step by an example, suppose we have two matrices in two consecutive points of time 1 and 2, X(1) and X(2):

$$X(1) = \begin{bmatrix} 2 & 1 \\ 0 & 0 \\ 1 & 2 \end{bmatrix} \quad X(2) = \begin{bmatrix} 2 & -1 \\ 0 & 0 \\ 1 & -2 \end{bmatrix}$$

Then: $$I(1) = \begin{bmatrix} 6 & 4 \\ 4 & 6 \end{bmatrix} \quad I(2) = \begin{bmatrix} 12 & 3 \\ 3 & 12 \end{bmatrix}$$

I(1) contains information about time 1 whereas I(2) contains information about time 1 and time 2. It is important that we cumulate the sums of X'(t)*X(t), otherwise if we directly add the matrices X(1) and X(2), the second column would be zero.

3.3. The Reorderable Matrix Method

The information that is hidden in the input data, such as similarities or correlation, is difficult if not impossible to extract only by looking at numbers.

Figure 1: The Color Map Blue-White-Red

	v_10 k1	v_9 k1	v_8 k1	v_8 tot	v_7 k1	v_6 k1	v_6 tot	v_5 k1	v_4 k1	v_4 tot	v_3 a_tot	v_3 k1	v_3 k2	v_3 k3	v_3 k4	v_3 n_tot	v_2 k1	v_1 k1	v_1 ki	v_1 pot	v_0 k1
AT	-8.002	-38.68	4.8872	54.875	221.61	45.124	77.686	106.58	12.983	98.329	22.316	55.127	9.3495	-9.35	0.1872	-6.104	491.86	86.886	197.06	-137.2	-287.9
NA	2.0257	9.7651	-1.072	-13.26	-55.76	-11.42	-19.64	-27.15	-3.622	-27.17	-5.953	-14.24	-2.438	2.4382	-0.019	1.844	-125.4	-21.29	-45.75	29.498	71.97
S1	21.219	102.48	-13.03	-145.9	-583.2	-119.7	-207	-282.3	-37.06	-278.2	-61.82	-145.8	-25.14	25.144	-0.183	18.598	-1302	-224.3	-486.6	318.95	753.89
S2	-1.365	-6.591	0.7986	9.1782	36.832	7.7011	13.473	18.08	2.3941	17.92	4.0185	9.3453	1.6192	-1.619	0.0044	-1.253	82.453	14.126	30.256	-19.42	-46.65
S3	-4.144	-20	2.2831	27.417	113.22	23.37	40.448	55.223	7.3408	55.042	12.068	27.833	4.8541	-4.854	0.0336	-3.432	254.8	43.243	92.921	-59.85	-145.2
S4	9.7512	47.067	-6.037	-67.34	-267.2	-54.99	-95.32	-129.8	-18.91	-134	-29.11	-66.92	-11.66	11.655	-0.002	9.1849	-598.3	-100.7	-214.2	135.23	344.91
S5	-0.715	-3.451	0.3935	4.6774	19.281	4.0353	7.1513	8.503	1.219	9.1418	2.1407	4.8731	0.8494	-0.849	-0.003	-0.693	43.213	7.3681	15.879	-10.23	-23.79
S6_cyt	3.4417	16.553	-1.791	-22.47	-95.65	-20.41	-36.45	-46.48	-6.215	-46.66	-10.09	-24.49	-4.174	4.1741	-0.045	3.089	-215.3	-36.59	-78.72	50.966	124.59
S6_out	3.077	13.911	-1.482	-18.93	-81.8	-17.52	-31.51	-39.76	-5.286	-39.72	-8.562	-21	-3.569	3.569	-0.045	2.6163	-184.3	-31.38	-67.67	44.056	106.76

Figure 2: A Sample Matrix

Considering that the human user is more sensitive to visual stimuli, the substitution of numbers by symbols or colors is a good way to eliminate this problem. Thus, the basic idea of our approach is to transform a matrix of numerical data into a matrix of colors. In order for this color matrix visualization to be successful, the columns and rows of this matrix must be reorganized via manual permutations or by algorithms for automatic generation of the optimal permutation matrices [8, 16, 17].

We have implemented a color visualization version of the reorderable matrix method. The algorithm for transforming a matrix X(t) of numbers into a matrix C(t) of colors is based on using so-called *color maps* which determine the spectrum of colors used in the visualization. This color spectrum is defined by three colors: two border colors and the transition color, as shown in Figure 1. Spectra with more colors are not used though in specific problem domains this could be useful. The input data is firstly exposed to a row based normalization process where the maximum norm (L_∞ norm for a vector $\vec{x}$ is defined as $\|\vec{x}\|_\infty = \max_i |x_i|$) is used to normalize the data within all points in time. After normalization, the values of the data matrix lie in the segment [-1, 1]. This segment is divided in as many small segments as nuances of colors are used (we use 511 colors; 256 for red, 256 for blue, but white is common to both), and the transformation to colors is done according to Formula (1).

$$Color = ColorMap\ (Value \times 255) \quad (1)$$

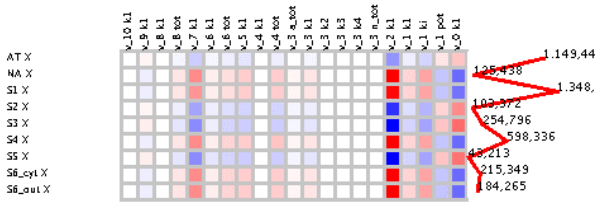

Figure 3: Reorderable Matrix of the Data of Figure 2

For example, in Figure 2 a sensitivity matrix generated for a fictitious metabolic network model at a certain point in time before normalization with the L_∞ norm is presented. The parameters of the model are shown in the columns whereas the metabolites are presented in the rows. The values in the matrix show how the change of a certain parameter affects the values of the metabolites. For instance, an increase of one unit in parameter v10_k1 would bring a decrease of 8.002 units in the metabolite AT, and an increase of 21.219 units in the metabolite S1. The result of the color visualization for this matrix can be viewed in Figure 3. Each element of the data matrix is transformed into a rectangle with an appropriate color; in the right of the visualization we see a small graph representing the maximum norm, which is calculated for all points of time and is used to normalize the input. However, the color visualization alone does not allow the easy detection of structures in data. Two methods are provided to give the user this possibility:

- Interaction with the color visualization
- Automatic calculation of the optimal permutation

The interaction method is described in the next subsection. Concerning the automatic calculation of permutation matrices, Mäkinen and Siirtola [7] present two heuristics for automatically reordering matrices. The first one is a weight-based sorting algorithm. Figure 4 shows the reordering of Figure 3 after applying the weight-based algorithm proposed in [7]. We can see in Figure 4 that similar columns are placed near to each other.

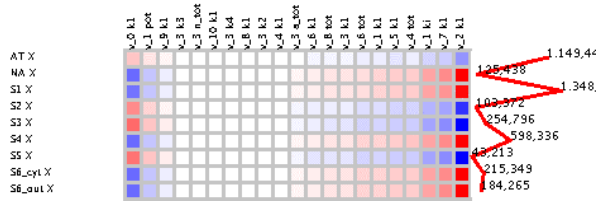

Figure 4: Reorderable Matrix of Figure 3 After Reordering

The second heuristic is based on the Sugiyama algorithm for drawing graphs [20]. Due to space reasons, these two algorithms are not discussed further. However, bearing in mind that we have time-varying matrices, these two heuristics generate one solution (i.e. one permutation) for each point in time. Considering that these permutations can vary with time, the optimal permutation over time should be found; otherwise, the user would be distracted by swapped columns possibly at each point of time.

Thus, a new discrete optimization problem is raised: we have to find the optimal permutation among all permutations over time. In our approach, this problem is solved heuristically, considering the fact that the search space is large. According to Bertin [16,17], the reorderable matrix method can be used for exploring matrices where one dimension can go up to 500. Thus, the number of possible permutations would be 500!.

Our proposed heuristic for finding the optimal permutation over time works as follows:

1. *Let P(t) for t=1,..,n be the permutations of m columns of a matrix at all time points n.*
   a. *Let* d_{P_1,P_2} *be a similarity function between two permutations, defined as the number of common values at the same positions of the two permutations. Thus, the permutations {1,2,3,4,5,6} and {2,1,3,4,5,6} have a similarity degree of 4 (similar to Hamming distance in string matching problems).*
   b. *The goal is to find the permutation* $\hat{P}$*, for which* $\sum_{t=1..n} d_{\hat{P},P(t)}$ *is minimized.*
2. *Compute the "average" of all permutations as the sum of the individual corresponding elements of all permutations each divided by n, which is considered as the "barycenter" of these permutations*
3. *Choose as* $\hat{P}$ *the permutation from P(t) for t=1,..,n which has the largest similarity value with the "barycenter".*

In this algorithm, the distance function and the solution $\hat{P}$ are computed heuristically, based on the observation that in our applications the permutations do not vary radically over the time. These choices were mainly made to achieve a fast execution of the algorithm. If other criteria are desirable in further applications, both the distance function and the "optimal" permutation could be chosen in a different way.

3.4. Interaction

To help the user during the exploration process, both visualization methods, MDS and the reorderable matrix method, are interactive. Our MDS visualization allows zooming in and out, translation, selection and export into a text format. Our implementation of the reorderable matrix method allows the user to swap columns of matrices in the case when he or she is not pleased with the found permutation. To control the time, the possibility to view the results of the visualization in an animation-like form is provided. The MDS view is the master view, and the interaction with the reorderable matrix is the subordinate view.

Both views are synchronized with each other such that actions in one view also affect the other view, as shown in Figures 5 and 6 where two selected points (labeled) in MDS in Figure 6 are also highlighted in the reorderable matrix view in Figure 5 (with frames in the middle).

4. Case Study: Visualization of Sensitivity Matrices Generated During Simulation of Metabolic Network Models

The described approach was developed as a tool to support our project partners of the biotechnology group at the Research Center Jülich, Germany, during their metabolic modeling work.

The generic model of a metabolic network is described by equation (2) [2, 22]. Here, $\vec{X}$ represents the vector of metabolite concentrations, $\vec{S}$ represents the concentration of input substances, the matrix N represents the structure of the metabolic network (similar to the adjacency matrix of a graph) and $\vec{\alpha}$ represents the vector of parameters of the model. The vector $\vec{v}$ represents the kinetic functions, which show how the reactions evolve over the time.

$$\dot{\vec{X}} = N \cdot \vec{v}(\vec{\alpha}, \vec{S}, \vec{X}), \vec{X}(0) = \vec{X}_0 \qquad (2)$$

Considering that a typical model could have tens or hundreds of parameters, a reasonable step would be to search for a new model with fewer parameters but the same properties as the original one. For this purpose, a procedure called sensitivity analysis [2, 22] is carried out in order to analyze the influence the changes in parameters have on the model. Sensitivity analysis is one of the important tools to help the modelers in their work, as the modeling process is an iterative process

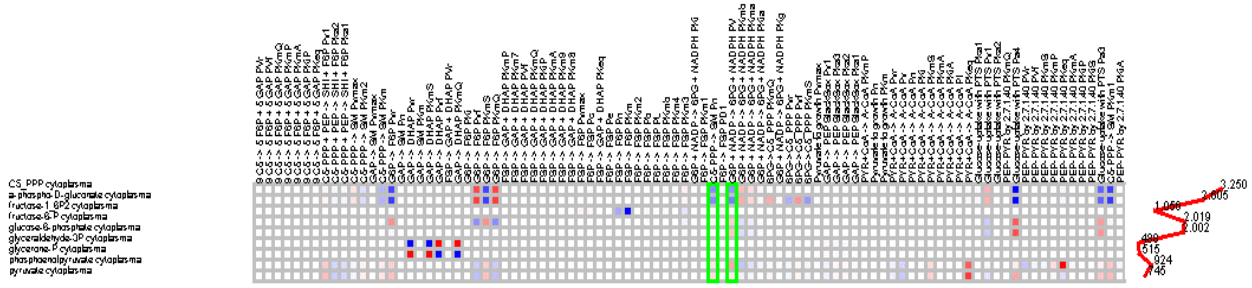

Figure 5: The Colored Reorderable Matrix of a Sensitivity Matrix of a Certain E.coli Model

where quite often the next generation model is a simplification of the old one. Sensitivity analysis in the context of this paper is concerned with computing the expression in formula (3). The results of this computation are time-dependent sensitivity matrices.

$$\frac{\partial \vec{X}}{\partial \vec{\alpha}}(t) \qquad (3)$$

The visualization helps the modeler to understand the results of sensitivity analysis, since a manual exploration is impossible. Its main purpose is to determine which parameters are important for the model, i.e. which parameters are different from the others ("outliers") and which parameters are correlated with each other and could possibly be omitted in a new model.

Figure 5 shows the reorderable matrix at a randomly selected point of time for the sensitivity matrices of a certain E.coli model, whereas Figure 6 shows the corresponding MDS visualization. From the MDS visualization, we can directly see which parameters are possible outliers (two possible outliers which are similar to each other are the parameters C5_PPP ->BM Pn and G6P + NADP -> 6PG +NADPH PV, as highlighted in both views). From the color visualization (Figure 5), we can verify the result of MDS and see whether the selected parameters are sensitive or not with respect to the corresponding metabolites. This procedure helps the modeler to simplify the model of the metabolic network. In this case, the two highlighted parameters could be possibly mixed into a single one in a new model.

5. Conclusions

In this paper, we have presented a new approach to visualize time-varying matrices. The proposed approach is based on combining multi-dimensional scaling and the reorderable matrix method. Since both methods were originally designed to work for a single point of time only, they had to be adequately extended.

The benefits of our proposal were shown for a biocomputing application concerned with the visualization of sensitivity matrices generated during the simulation of metabolic network models.

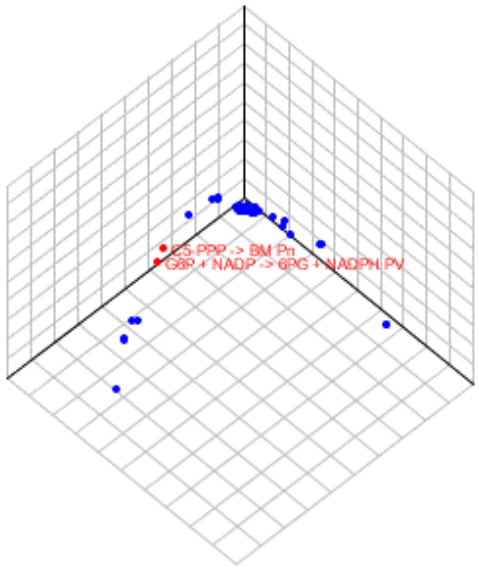

Figure 6: MDS Visualization of the E.Coli Model Shown in Figure 5

There are several areas for future work. For example, it would be interesting to investigate alternatives for the current heuristics used for dealing with time-varying reorderable matrices. Furthermore, interactive parallel coordinates in 3D or further hybrid approaches for visualizing time-varying matrices would be interesting to explore. Finally, the current MDS algorithm finds solutions for every time point. The next step would be to search for a globally optimized solution within all points of time, possibly in parallel, such that MDS is consistent globally.

6. Acknowledgements

This work is financially supported by the Deutsche Forschungsgemeinschaft (Schwerpunktprogramm 1063, Teilprojekt FR 791/8-1).

7. References

[1] Alfred Inselberg and Bernard Dimsdale. Parallel Coordinates: A Tool for Visualizing Multidimensional Geometry. In *Proceedings of IEEE Conference on Visualization '90*. IEEE, 361-378. 1990.

[2] Brian P. Ingalls and Herbert M. Sauro. Sensitivity Analysis of Stoichiometric Networks: An Extension of Metabolic Control Analysis to Non-Steady State Trajectories. In *Journal of Theoretical Biology Volume* 222, Issue 1, 23-26. 2003.

[3] Claudia Schmid, Hans Hinterberger. Comparative Multivariate Visualization Across Conceptually Different Graphic Displays. In *Proceedings of the Seventh International Working Conference on Scientific and Statistical Database Management*. IEEE, 42-51. 1994.

[4] David J. Marchette and Jeffrey L. Solka. Using Data Images for Outlier Detection. In *Computational Statistics & Data Analysis*, Vol. 43, 541-552. 2003.

[5] D.F. Andrews. Plots of High Dimensional Data. In *Biometrics*, Issue 28, 125-136. 1972.

[6] T.T. Elvins. VisFiles - Presentation Techniques for Time-Series Data. In *Computer Graphics*, Vol. 31, Issue 2, 14-16. 1997.

[7] Erkki Mäkinen and Harri Siirtola. Reordering the Reorderable Matrix as an Algorithmic Problem. In *Proceedings of Diagrams 2000*. 453-467. 2000.

[8] Eser Kandogan. Visualizing Multi-Dimensional Clusters, Trends, and Outliers Using Star Coordinates. In *Proceedings of Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 107-116. 2001.

[9] G. Young and A. S. Householder, Discussion of a Set of Points in Terms of Their Mutual Distances. *Psychometrica*, vol. 3, 19-22. 1938.

[10] Geraldine E. Rosario, Elke A. Rundensteiner, David C. Brown and Matthew O. Ward. Mapping Nominal Values to Numbers for Effective Visualization. In *Proceedings of the 2003 IEEE Symposium on Information Visualization*. 15-22. 2003.

[11] Herman Chernoff. The Use of Faces to Represent Points in k-Dimensional Space Graphically. In *Journal of the American Statistical Association*, Vol. 68, 361-368. 1973.

[12] Harri Siirtola. Combining Parallel Coordinates with the Reorderable Matrix. In *Proceedings of Coordinated and Multiple Views In Exploratory Visualization (CMV'03*). IEEE, 63-74, 2003.

[13] Harri Siirtola. Interaction with the Reorderable Matrix. In *Proceedings of the International Conference on Information Visualization 1999*. IEEE, 272-277. 1999.

[14] I.J. Schoenberg. Remarks to M. Fr'echet's article "Sur la d'efinition axiomatique d'une classe d'espaces vectoriels distanci'es applicables vectoriellement sur l'espace de Hilbert". *Annals of Mathematics*, 36:724-732, 1935

[15] I.K. Fodor. A Survey of Dimension Reduction Techniques. Lawrence Livermore National Laboratory (LLNL) Technical Report, June 2002. UCRL-ID-148494.

[16] Jacques Bertin. Semiology of Graphics, Madison, Wisconsin. University of Wisconsin Press. 1983

[17] Jacques Bertin. Matrix Theory of Graphics. In *Information Design Journal*, Vol. 10, 5-19. 2001

[18] Jonathan C. Roberts. On Encouraging Multiple Views for Visualization. In *Proceedings of the International Conference on Information Visualization*. IEEE, 8-14. 2003.

[19] K. V. Mardia, J. T. Kent, and S. M. Bibby. Multivariate Analysis. Academic Press, 1979.

[20] K. Sugiyama, S. Tagawa and M. Toda. Methods for Visual Understanding of Hierarchical System Structures. In *IEE Trans.Syst. Man Cybern*. 109-125. 1981.

[21] Monika Lanzerberger, Silvia Miksch and Margit Pohl. The Stardinates – Visualizing Highly Structured Data. In *Proceedings of the International Conference on Information Visualization 2003*. IEEE, 47-52. 2003.

[22] Marc Daniel Haunschild, Wolfgang Wiechert. Sensitivity Analysis of Metabolic Network Models. ASIM 2003, 17. *Symposium Simulationstechnik*, 415-420. 2003

[23] Michael Friendly. Corrgrams: Exploratory Displays for Correlation Matrices. In *The American Statistician*. Vol. 56 Issue 4, 316-324. 2002.

[24] Michael C. Minnotte and R. Webster. The Data Image: A Tool for Exploring High Dimensional Data Sets. In: *1998 Proceedings of the ASA Section on Statistical Graphics*, Dallas, Texas. 25-33. 1998.

[25] Michelle Q. Wang Baldonado, Allison Woodruff and Allan Kuchinsky. Guidelines for Using Multiple Views in Information Visualization. In *Proceedings of the Working Conference on Advanced Visual Interfaces (AVI'00*). 110-119, 2000.

[26] W. Müller. and H. Schumann. Visualization Methods for Time-Dependent Data: An Overview. In: *S. Chick, P.J. Sanchez, D. Ferrin, D.J. Morrice (eds.), Proceedings of 2003 Winter Simulation Conference*, 737-745. 2003.