Interactive Analysis of Time-Varying Systems using Volume Graphics

Jimmy Johansson, David Lindgren, Matthew Cooper and Lennart Ljung

Abstract— We show how 3-dimensional volume graphics can be used as a tool in system identification. Time-dependent dynamics often leave a significant residual with linear, time-invariant models. The structure of this residual is decisive for the subsequent modelling, and by using advanced visualization techniques, the modeller may gain a deeper insight into this structure than that which can be obtained from standard correlation analysis. We present a development platform that merges a rich variety of estimation programs with state of the art visualization techniques.

I. INTRODUCTION

A. Interaction in System Identification

System Identification is inherently an interactive art. Results from preliminary model building are studied by the user. Based on such studies, decision about new model structures are taken. The studies are typically of visual nature, often simple 2-dimensional line plots of correlation functions and residuals. Visualization techniques have gone through a significant development during the past decade. It is an interesting problem to study what such new techniques may offer in terms of improved interaction in system identification. The purpose of the present contribution is to give some illustrations of what can be achieved in this way. No new identification methods and no new model indicators will be introduced, but new ways to visualize the information in a suggestive manner. The study is a result between a research group in system identification and one in scientific visualization.

B. Time Invariant Models and Time-varying Systems

Fitting a time-invariant model to data sampled from a system with time-varying dynamics is, of course, possible but often results in a large-magnitude time-varying residual. Unless we are aware from the beginning of the time-variant nature of the system dynamics the cause of the large residual magnitude will, perhaps, not be evident. Covariation and auto-covariation estimates reveal that something is wrong but give no clue to whether the bad fit is due to wrong model order, non-linearities or time-variant dynamics. At this point, it is actually *difficult to gain insight* into the model discrepancies. Volume graphics in 3 dimensions, however, offers a means to go beyond standard summary statistics measures for model validation. By assigning each

Jimmy Johansson, `jimjo@itn.liu.se`, and Matthew Cooper, `matco@itn.liu.se`, are with the Department of Science and Technology, Linköping University, Sweden

David Lindgren, `david@isy.liu.se`, and Lennart Ljung, `ljung@isy.liu.se`, are with the Division of Automatic Control, Linköping University, Sweden

data point to a semi-transparent volume in a 3-dimensional spatial/temporal space, the modeller may explore and interpret large-scale process data sets (10^4-10^5 sample points).

The process of *system identification* [9] can be broken down into 3 steps:

1) Select model structure.
2) Estimate model parameters given data sampled from a system and generate the model residual – the part of the output that is left unmodelled.
3) Validate the model. If the residual is *satisfactory* then finish, else select another model structure and start again. A satisfactory residual is, for instance, a residual small in magnitude and independent of the model/system input.

In this work we show how step 3 above can be aided by the use of interactive 3-dimensional volume graphics, in particular, how time-varying dynamics and other potential inadequacies of the model can be effectively detected.

Being able to visualize data with multiple visualization techniques in multiple views can be powerful when analysing complex data. There exist a large number of visualization techniques for 2- and 3-dimensional data. One of the simplest and most widely used techniques is the 2-dimensional scatter plot which simply shows the relationship between two variables (Fig. 1). A more sophisticated way of visualizing data is to use a volume graphics technique [4]. Volume graphics is used for visualizing sampled functions of three spatial dimensions by computing 2-dimensional projections of a coloured, semitransparent volume. One common application area of volume graphics is in the medical imaging domain where, for example, data can be constructed from the output of an X-ray Computed Tomography (CT) scanner.

However, visualization alone is usually not enough in order to fully understand the data. Of equal importance are also correct pre-processing of the data and the ability to interact and manipulate with the result. Using linked views [11], [14] is one technique, which enables the investigation of the same data in several views at the same time. This facilitates the discovery of unexpected relationships since multiple views are used. With linked views, a movement in one view is automatically propagated to all other views. In combination with different zoom levels, a setup of two or three linked views is useful to create a focus-plus-context [3] configuration. Many other combinations are possible where linked views reduce the user's cognitive load compared to the case of two independent views.
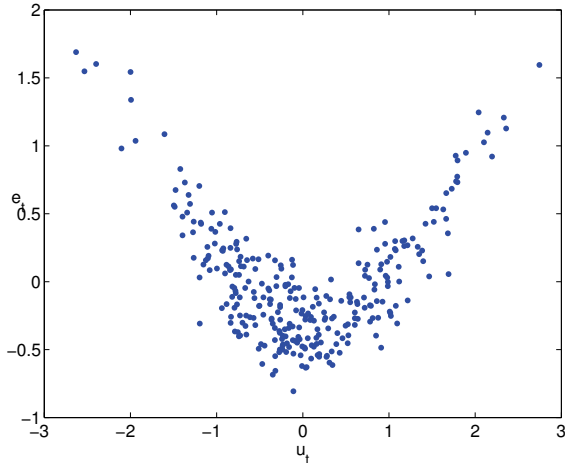
Fig. 1. A simple scatter plot showing samples from two dependent but uncorrelated random variables.

Fig. 2. Model validation. The output y_t of the system, here discrete, is compared to the output of the model, $\hat{y}_t$. The difference between the two makes the residual, e_t. If the residual is dependent on the noise only, not on the input u_t, then there is nothing more to model.

C. Related Work

The visualization and estimation techniques we use in this work are well-known in their respective disciplines. Using volume graphics as an interactive tool for system identification, however, we have not seen elsewhere. What we have seen, though, are great efforts being made to visualize complex process data, for instance, by means of scatter plot matrices and parallel coordinates, [5], [7] The applications have mainly been for decision support, see [8]. For an overview of concepts and taxonomies of visual data exploration, see [2].

II. MODEL VALIDATION

We will use a linear discrete time-invariant state space model

$$\begin{cases} x_{t+1} = Ax_t + Bu_t \\ \hat{y}_t = Cx_t. \end{cases} \quad (1)$$

Here, u_t is the input signal (common to the system and model), $\hat{y}_t$ the simulated model output and y_t the sampled output signal of the system to be identified. The model order n is the same as the dimensionality of the state space vector x_t. Given n (design parameter), a set $\{u_t, y_t\}_1^N$ with sampled data points and some assumption of the initial state x_0, techniques to estimate the model parameters A, B, C are well-known (state-space subspace system identification), see [12]. Programs for this task are available commercially, for instance the *System Identification Toolbox* [10] for use with MATLAB.

Once the model parameters are estimated and $\hat{y}$ simulated according to (1), the model residual is calculated as

$$e_t = y_t - \hat{y}_t, \quad t = 1, 2, \ldots, N. \quad (2)$$

If the model is a correct description of the sampled system, the residual will depend only on some unknown noise process, not on the input. In fact, if the residual is independent of present and past inputs, nothing more is left to be modelled and the process is complete. Fig. 2. illustrates the signal paths.
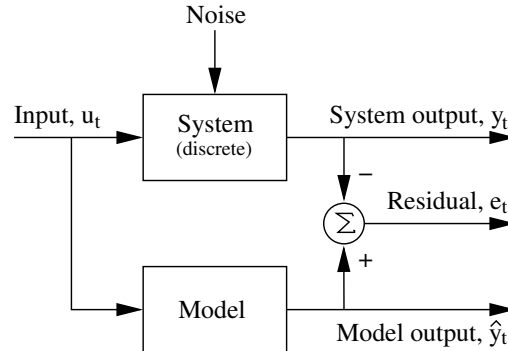
A standard measure of the dependence between two scalar, zero-mean samples is the sample cross covariance,

$$\widehat{R}_{eu}(\tau) = \frac{1}{N-\tau} \sum_{t=1}^{N-\tau} e_{t+\tau} u_t. \quad (3)$$

If some $\widehat{R}_{eu}(\tau)$ is significantly larger than 0, this indicates the model is not sufficient. Plotting $\{e_t\}$ against $\{u_{t+\tau}\}$ in different 2-dimensional scatter plots for $\tau = 0, 1, \ldots$ may however reveal dependencies not reflected by $\widehat{R}_{eu}$ at all. Fig. 1 illustrates this by plotting samples from two evidently dependent random variables that are totally uncorrelated, $\widehat{R}_{eu}(0) = 0$. To also visualize *time-varying* dependencies it will be necessary to use 3-dimensional graphics to display $e_{t+\tau}$, u_t, and t. This gives rise to a number of issues we will address in the subsequent sections.

III. IMPLEMENTATION

Our software platform exploits the system identification toolbox in MATLAB together with a powerful object-oriented visualization system, AVS/Express [1]. This system is based on individual modules and an application is built by combining these through a visual network editor. MATLAB and AVS/Express have been seamlessly integrated through a common user interface with which the user can inter-actively pre-process, analyse and visualize their data. Our application has also been extended from an ordinary desktop PC to a much larger semi-immersive stereoscopic display. This type of visualization creates the psychological illusion of being inside the computer-generated environment, rather than viewing it from the outside through a screen. The user can interact with the visualization to navigate through and control the positions of the objects. Stereoscopic vision seems to be specially promising when analysing large complex data.

A. Application Overview

A simplified work flow of our application is shown in Fig. 3. The work flow begins with loading data from a

file into MATLABs workspace. The user can choose to visualize the original data set but can also perform a number of pre-processing operations on the data. The next step in the analysis process is to choose an appropriate model structure and to perform the estimation. Through the user interface, the user can look at a preview of the estimation result in a 2-dimensional scatter plot. This process is highly interactive so the user can at any time go back and change the parameters of the model to further improve the result of the visualization. When the user has identified an area of interest, it can be visualized using a volume graphics technique. In Fig. 4 this is illustrated on data from a sampled feed-back system where the closed loop gain is increased at the end of the sequence. The axes are t, e_t and u_{t-1}. To further investigate the data, linked views can be used to analyse a cross-section of the data.

To manage the communication between MATLAB and AVS/Express, we have developed a communication module. This module handles all control messages and the entire flow of data between the two systems. Since system identification data may be large, the data is (as far as possible) handled locally by MATLAB. It is only when the user explicitly requests data that it is transferred from MATLAB to AVS/Express. Also, it is only the selected subset that is sent, never the entire data set. This makes the processing of large data sets much easier and the visualization process highly interactive. Our module can be used in all AVS/Express applications and can be downloaded for free from the International AVS Centre [6].
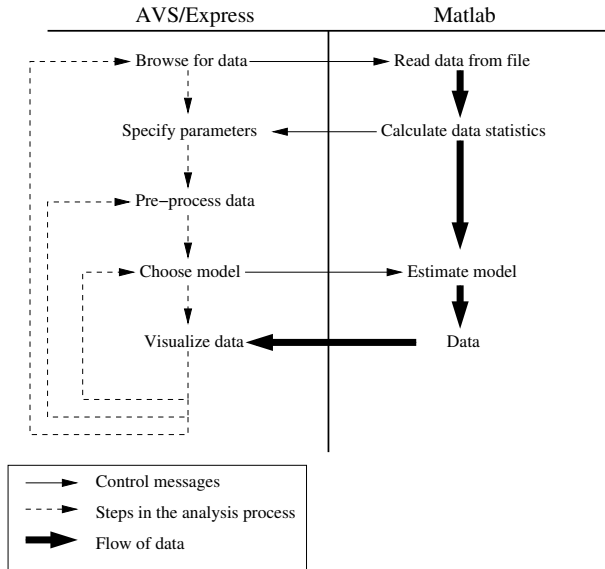

Fig. 3.   The work flow between AVS/Express and MATLAB.

B. Interactive Volume Graphics

In the analysis process it is possible to perform a volume graphics visualization of a selection of variables. This 3-dimensional visualization technique effectively reveals any time-variations and is an important part of the analysis process (Fig. 5). To some extent it can also be used to identify non-linear dependencies but more work needs to be done to systemise the process.

To make the volume graphics visualization interactive we need to re-calculate the volume data every time the user makes a change in the parameter settings. How quickly this can be done primarily depends on the volume size and on the type of platform used. The calculation, described below, and rendering of the visualization of a $128 \times 128 \times 256$ volume on a desktop PC can be done with only one or two seconds of delay. The size of the volume can dynamically be changed to speed up the calculations on a slower PC. To construct the volume data we extract a subset d of the 2-dimensional data (Fig. 6). Each plane in the volume consists of an $n \times m$ matrix of bins and the data points, scaled by a suitable factor in each dimension, are aggregated into these bins as a simple population count. The scaling factors are calculated so that

$$\begin{cases} x_i^{\text{scaled}} = \frac{m}{\max(\vec{x})} x_i \\ y_i^{\text{scaled}} = \frac{n}{\max(\vec{y})} y_i. \end{cases} \tag{4}$$

This procedure is done for every subset, which gives us a volume of size $m \times n \times \frac{t}{d}$.
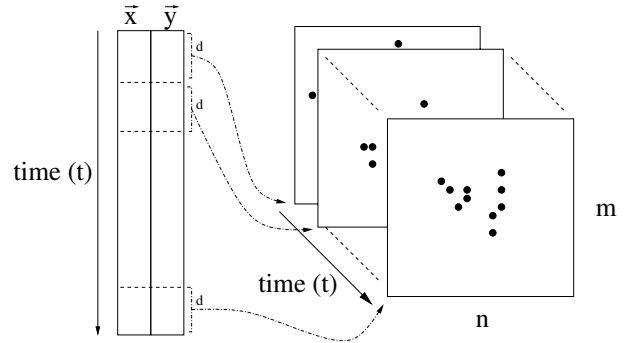

Fig. 6.   A subset of the 2-dimensional data is scaled and aggregated. This is done for every subset d, which gives a volume of size $n \times m \times \frac{t}{d}$.

C. Application Performance and Timing

To perform a complete analysis of a data set, a number of visualization modules and MATLAB libraries are used. We have identified the model estimation to be the most time consuming operation in our application. The time for estimating a third order linear state-space model of a data set with 40,000 observations may be ten times as long as loading the data, which is the second most time consuming operation.

We have conducted a series of performance tests on our model estimation library. These tests were done on a PC running Windows XP with an Intel Pentium 4 CPU (2.40 GHz) with 256 MB RAM. As test data we used a time-varying data set where we used one input and one output variable, with a growing number of observations. The results are presented in Fig. 7, where we can see that
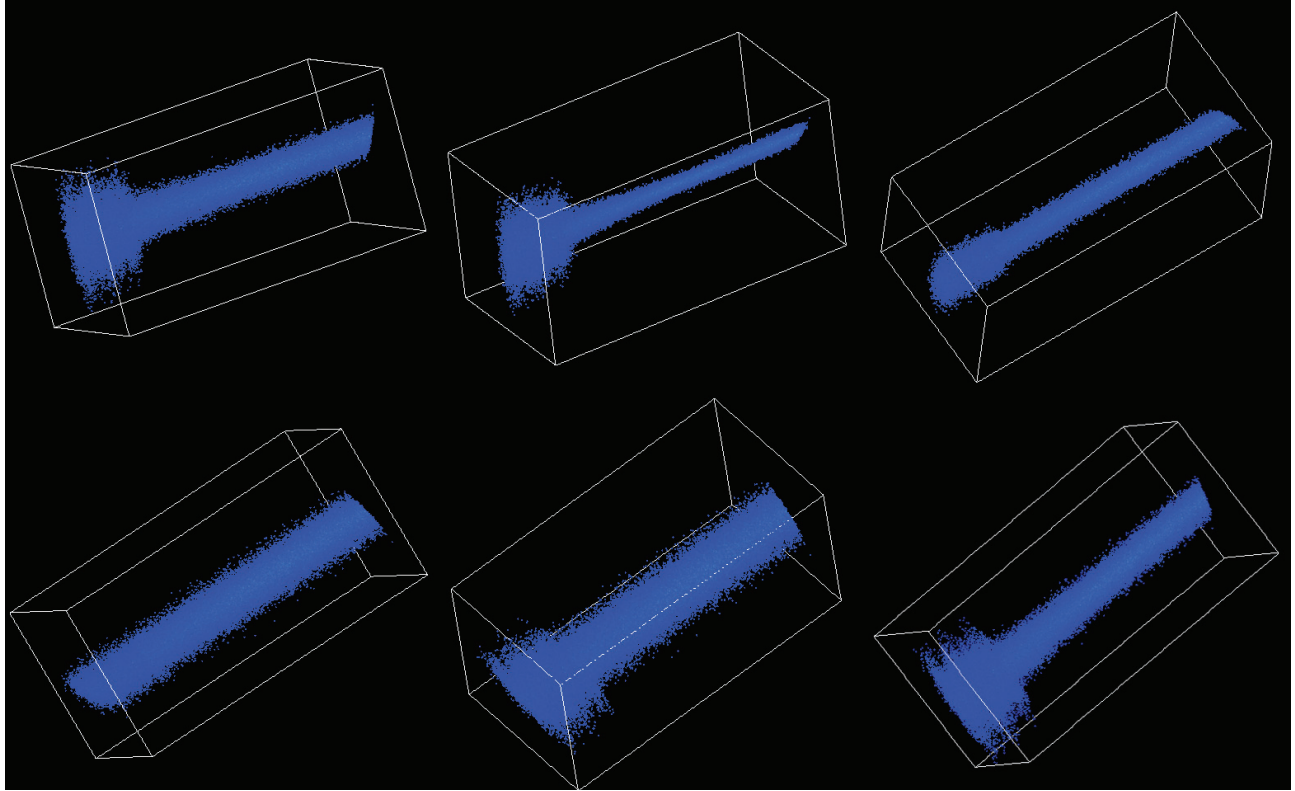
Fig. 4. Six different view angles of time-varying data visualized using a 3-dimensional volume graphics technique.
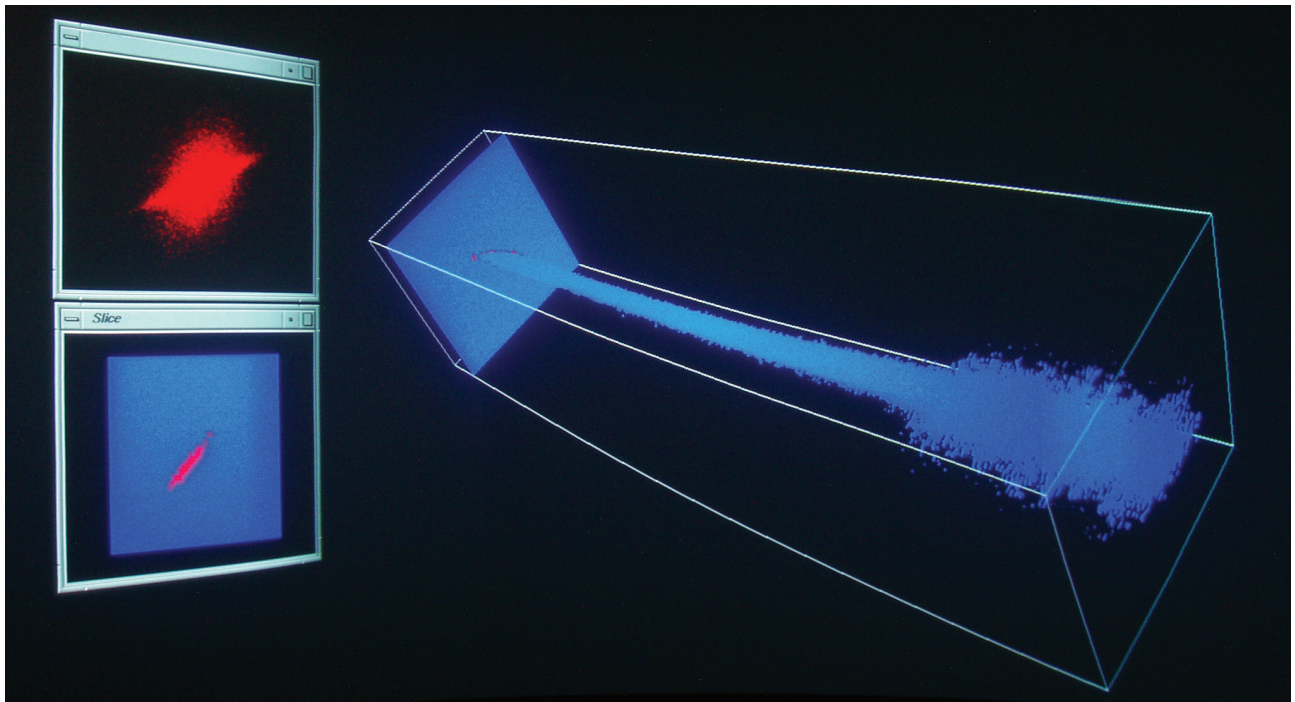

Fig. 5. Our application running on a semi-immersive display. The 2-dimensional scatter plot in the upper left view shows the relationship between u_t and e_t. From the visualization we can clearly see that there is a correlation, which implies that there are still information left to be described. There is also another region, which is less correlated and the problem is that the 2D scatter plot gives no information about where in time these separate regions are. Visualizing the same variables with a volume graphics technique (right view) gives the necessary temporal information. To further investigate the relationship between these variables, the cross-section from the clip plane is displayed in the lower left view. The plane can also be animated through the entire time sequence.

the relationship is fairly linear and it takes approximately 14 seconds to estimate a third order state-space model on a data set with 100,000 observations. This is, however, an acceptable waiting time since this operation is not performed repeatedly during the analysis process.
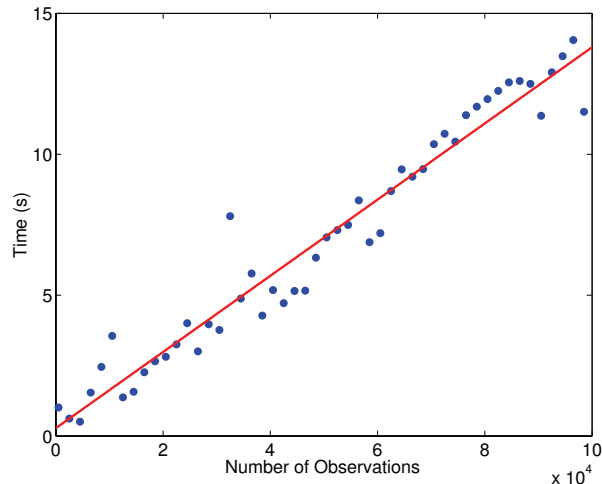

Fig. 7.   Result of performance test on our model estimation library.

IV. RESULTS AND CONCLUSIONS

A. Results

Volume graphics provides the modeller with useful insight into the structure of data sampled from a non-linear and/or time-varying system. The residual of a LTI state space model has been combined with the model input and time to construct a spatial/temporal 3-dimensional volume. Integrating this 3-dimensional volume visualization with several 2-dimensional visualizations together with interaction through linked views creates a dynamic platform for exploratory analysis of such data. The understanding and experience gained from interacting with data through this platform has proven to be a valuable tool in the system identification process.

Extending the visualization from a desktop PC to a much larger semi-immersive stereoscopic display may help the user analyse complex 3-dimensional data even more effectively.

B. Conclusions

We see a great potential for advanced visualization techniques as a support for system identification. Future plans and ideas include

- Completion of the platform with additional model structures: for instance non-linear Wiener-Hammerstein models and adaptive time-variant models.
- Study of non-linear systems, and how volume graphics and projection techniques can facilitate the pursuit of adequate model structures.

- Implementation of stereoscopic display on an ordinary PC making the new tools available to a broader public.
- Merging the interactive volume graphics with projection based methods known from the area of exploratory data analysis.

V. ACKNOWLEDGEMENTS

REFERENCES

[1] Advanced visual systems. http://www.avs.com/software/soft_t/avs-xps.html.
[2] Maria Cristina Ferreira de Oliveira and Haim Levkowitz. From visual data exploration to visual data mining, a survey. *IEEE Transactions on Visualization and Computer Graphics*, 9(3):378–394, 2003.
[3] Helmut Doleisch, Martin Gasser, and Helwig Hauser. Interactive feature specification for focus+context visualization of complex simulation data. In *Proceedings of the symposium on Data visualisation 2003*, pages 239–248. Eurographics Association, 2003.
[4] Robert A. Drebin, Loren Carpenter, and Pat Hanrahan. Volume rendering. In *Proceedings of the 15th annual conference on Computer graphics and interactive techniques*, pages 65–74. ACM Press, 1988.
[5] A. Inselberg and B. Dimsdale. Parallel coordinates: a tool for visualizing multi-dimensional geometry. In *Proceedings of the First IEEE Conference on Visualization*, pages 361–378, 1990.
[6] International avs centre. http://www.iavsc.org/.
[7] Jimmy Johansson, Robert Treloar, and Mikael Jern. Integration of unsupervised clustering, interaction and parallel coordinates for the exploration of large multivariate data analysis. In *Proceedings of the 8th IEEE International Conference on Information Visualisation IV04*, pages 52–57, 2004.
[8] Rudolf Kulhavý. A developer's perspective of a decision support system. *IEEE Control System Magazine*, pages 40–49, Dec 2003.
[9] Lennart Ljung. *System Identification: Theory for the User*. Prentice-Hall, Upper Saddle River, N.J. 07458, USA, 2:e edition edition, 1999.
[10] Lennart Ljung. *System Identification Toolbox User's Guide*. The MathWorks, Inc., Natick, MA. USA, version 6 edition, 2003.
[11] Chris North and Ben Shneiderman. Snap-together visualization: a user interface for coordinating visualizations via relational schemata. In *Proceedings of the working conference on Advanced visual interfaces*, pages 128–135. ACM Press, 2000.
[12] Peter van Overschee and Bart de Moor. *Subspace identification for linear systems : theory, implementation, applications*. Boston: Kluwer, 1996.
[13] Visimod homepage. http://www.visimod.org/.
[14] 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*, pages 110–119. ACM Press, 2000.