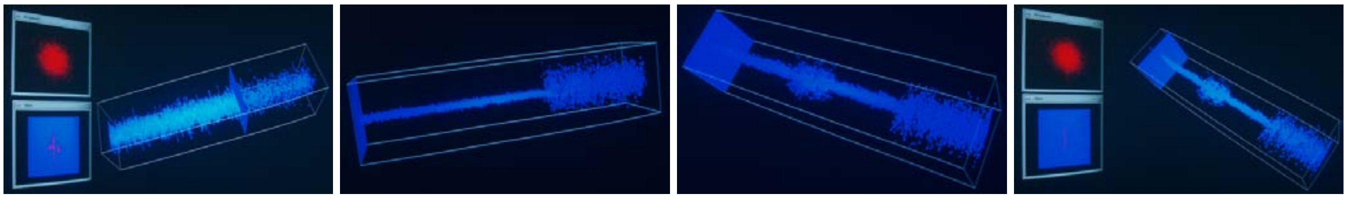
Interactive Poster: Interactive Visualization approaches to the Analysis of System Identification Data

Jimmy Johansson* Patric Ljung* David Lindgren† Matthew Cooper*

*NVIS - Norrköping Visualization and Interaction Studio, Linköping University, Sweden
† Division of Automatic Control, Linköping University, Sweden



a,b) One region with large residuals c,d) Two regions with large residuals

Figure 1: A semi-immersive display showing the relationship between the model input, the residual (the error between the model and the real system) over time. The visualization reveals that in some regions the residual is quite large, i.e. the model does not fit very well. The temporal information is crucial for the model validation which is an important step in the system identification process.

Abstract

We propose an interactive visualization approach to finding a mathematical model for a real world process, commonly known in the field of control theory as system identification. The use of interactive visualization techniques provides the modeller with instant visual feedback which facilitates the model validation process. When working interactively with such large data sets, as are common in system identification, methods to handle this data efficiently are required. We are developing approaches based on data streaming to meet this need.

CR Categories: I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction techniques; I.6.4 [Simulation and Modeling]: Model Validation and Analysis;

Keywords: System Identification, Interactive Visualization, Model Validation

1 Introduction

The purpose of system identification [4] is to find a mathematical model for a real world process, for example, a model that describes the relationship between the outflow and level in a set of connected water tanks.

The process of system identification can be broken down into three steps:

1. Select model structure.
2. Estimate model parameters given data sampled from a system and calculate the model residual – that which is left unmodelled.
3. Validate the model, see figure 2. If the residual is satisfactory then finish, else select another model structure and start again.

We show how step 3 above can be aided by the use of interactive visualization.

Denoting the model regressor, model input and model output by φ_t, u_t and y_t respectively we can define φ_t as

$$\varphi_t = [y_{t-1}, y_{t-2}, \ldots, y_{t-k}, u_t, u_{t-1}, \ldots, u_{t-l}] \tag{1}$$

and the estimated model output, $\hat{y}_t$, as

$$\hat{y}_t = f(\varphi_t), \tag{2}$$

where k and l are the number of time steps the model uses as a "memory" in order to predict the current time step, $\hat{y}_t$. In the most general case the aim of model validation is to identify the function f and the parameters k and l.

With large k and l the data to be considered in the model validation becomes highly multivariate and the relationships between these data are frequently non-linear and difficult to observe directly. This makes the model validation process difficult.

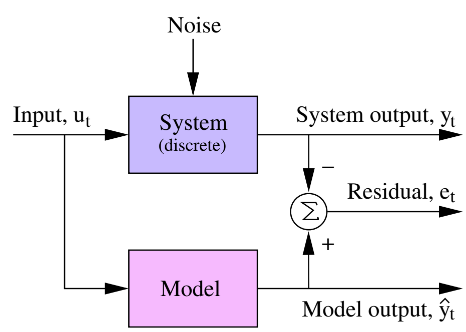


Figure 2: Model validation. The output y_t of the system, here discrete, is compared to the output of the model, $\hat{y}_t$. The difference between the two makes the residual, $e_t = \hat{y}_t - y_t, \quad t = 1, 2, \ldots, N$. If the residual is dependent on the noise only, not on the input u_t, then there is nothing more to model. Depending on the application, u_t and e_t could be either a scalar or a vector.

*e-mail:{jimjo,plg,matco}@itn.liu.se
†e-mail:david@isy.liu.se

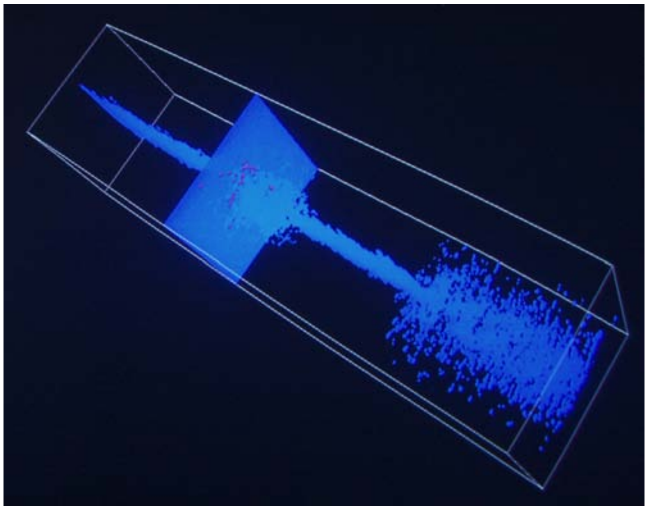


Figure 3: Our 3D visualization running on a semi-immersive display. The visualization shows the relationship between u_t, e_t and t.

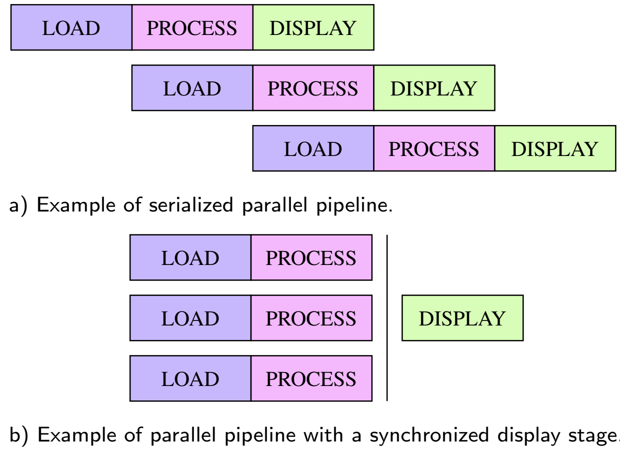


Figure 4: Representation of two different pipeline constructs.

2 Visualization of Time-Variant Dynamics

In a time-variant system, the differential equation relating u_t to y_t changes over time. We can seldom understand this change in dynamics by looking at the signals alone. Often we cannot even see when the changes occur. By using a time-invariant model, however, that describes the system well at one time instant, we can detect when new measurements do not match. This often means that something is wrong with the system, for example, overflow in a tank or a broken pipe. This will give rise to a large residual and can thus be identified. The interactive 3D visualization enables the user to dynamically visualize pair-wise combinations of e_t, u_t, y_t (or any of their time-lagged instances) against t. In figure 3 we can clearly see two different regions where the residual is large and so the model does not fit. The visualization gives the modeller the necessary temporal information to easily identify those regions.

The techniques we use in this work are well-known in their respective disciplines. Using interactive 3D visualization as a 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 [2]. The applications have mainly been in decision support [3]. For the interested reader, refer to [1] for an overview of concepts and taxonomies of visual data exploration.

Our implementation merges the rich variety of estimation methods provided by MATLAB with interactive visualization techniques offered by the AVS/Express visualization software. The developed application has also been extended from an ordinary desktop PC to a much larger semi-immersive stereoscopic display.

3 Pipelined Stream Processing

Dealing with data sets of sizes requiring unattainable amounts of internal memory prompt for new schemes of data management [6]. The nature of time-varying data sets often lend themselves to process data by streaming blocks of data through a processing pipeline instead of the entire data set at once. By formalizing the processing pipeline to work on blocks of data it becomes possible to provide progressive interactive visualization as well as to integrate external tools for processing of large data sets.

Although model fitting is usually performed on selected subsets of the data under study the validation of the fitted model is carried out on the entire data set. In order to support large data sets, blocks of data can be sent through the pipeline in sequence which reduces the memory requirement and enables reuse of memory for temporary data along the processing data path. In addition, parallel processing is also enabled, specifically useful on multi-CPU systems [5]. The pipelined processing increases the interactivity and allows for progressive visualization that also enables a user to abort and redefine a poor choice of processing at an early stage.

In figure 4 two schematic configurations are shown. The typical pipeline in subfigure *a* can apply to progressive processing and visualization as well as to time-varying presentation or animation of data. Data sets that allow for partitioned processing can be made in parallel (subfigure *b*). For this application both schemes can benefit the interactive visualization.

4 Conclusions

The understanding and experience gained from interacting with data through our application has proven to be a valuable tool in the system identification process. The use of interactive visualization seems to be a promising technique for model validation and a good complement to the more traditional techniques in use today.

References

[1] 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.

[2] Jimmy Johansson, Robert Treloar, and Mikael Jern. Integration of unsupervised clustering, interaction and parallel coordinates for the exploration of large multivariate data. In *Proceedings of the 8:th IEEE International Conference on Information Visualisation*, 2004.

[3] Rudolf Kulhavý. A developer's perspective of a decision support system. *IEEE Control System Magazine*, pages 40–49, Dec 2003.

[4] Lennart Ljung. *System Identification: Theory for the User*. Prentice-Hall, Upper Saddle River, N.J. 07458, USA, 2:e edition edition, 1999.

[5] Patric Ljung. Optimization and Parallelization of code for computing 3D Quantum Chaos. Technical report, Department of Science and Technology, Linköping University, 2003.

[6] Patric Ljung, Mark E. Dieckmann, Niclas Andersson, and Anders Ynnerman. Interactive Visualization of Particle-In-Cell Simulations. In *Proceedings IEEE Visualization 2000*, pages 469–472,592, 2000.