Teaching System Identification Through Interactivity

J.L. Guzmán * **D.E. Rivera** ** **S. Dormido** *** **M. Berenguel** *

* *Dep. Lenguajes y Computación, University of Almería, Spain (e-mail: joguzman,beren@ual.es)*
** *Department of Chemical Engineering, Arizona State University, Tempe AZ 85287-6006 USA (e-mail: daniel.rivera@asu.edu)*
*** *Dep. Informática y Automática, UNED, Spain (e-mail: sdormido@dia.uned.es)*

Abstract: The paper describes the main features of an interactive software tool developed in support of system identification education. This **I**nteractive **T**ool for **S**ystem **I**dentification **E**ducation (*ITSIE*) provides two distinct functional modes that are very useful from an educational point of view. The simulation mode enables the student to evaluate the main stages of system identification, from input signal design through model validation, simultaneously and interactively in one screen on a user-specified dynamical system. The real data mode allows the user to load experimental data obtained externally and identify suitable models in an interactive fashion. The interactive tool enables students to discover a myriad of fundamental system identification concepts with a much lower learning curve than existing methods.

Keywords: System identification education, interactivity, experimental design, prediction-error estimation, model validation.

1. INTRODUCTION

System identification deals with the problem of building dynamical models of systems from experimental data, and is a key component in control engineering practice [Ljung, 1999]. Consequently, system identification education forms an essential part of any comprehensive control engineering curriculum, and as such requires flexible and simple-to-use software tools. There are many powerful software tools available for system identification [Garnier and Mensler, 2000, Ljung, 2003a,b], but these present several disadvantages when viewed from a primarily educational point view. Normally, these tools do not evaluate all stages of the system identification process (experimental design, model structure selection, parameter estimation, and validation) in an integrated fashion. Furthermore, available tools provide substantial amounts of information in many different screens, which can be quite confusing for students. Finally, system identification is naturally performed in an iterative manner, that is, it involves a refining process where subsequent stages need to be recalculated step by step when a parameter or specification is modified. Failure to accomplish these iterations in a manner transparent for the user diminishes any educational benefits since students lose the connection with theoretical ideas and become less motivated. Thus, a new generation of software tools addressing these concerns are needed in support of advancing system identification education.

Interactive software tools have been proven as particularly useful techniques with high impact on control education [Dormido et al., 2005, Guzmán et al., 2005, Guzmán, 2006, Guzmán et al., 2008]. Interactive tools provide a real-time connection between decisions made during the design phase and results obtained in the analysis phase of any control-related project. Because system identification is a field rich in visual content that can be represented intuitively and geometrically [Ljung, 2003a], it naturally lends itself to interactivity. A novel interactive software tool for system identification was developed in Guzmán et al. [2009] based on these ideas. This **I**nteractive **T**ool for **S**ystem **I**dentification **E**ducation (*ITSIE*) addresses the various issues described previously. It includes all stages of system identification in the same screen, with the different stages connected interactively in such a way that a modification in one stage is automatically visualized in the remaining stages.

The work described in Guzmán et al. [2009] represents an initial effort to develop an interactive software tool for identification, with some limitations from an educational perspective. The only plant option available is a simulated SISO fifth-order system, with no option to enable the instructor to configure his/her own simulated example that could be shared with students. It is not possible to load external data in order to identify models from real experiments, Finally, students cannot generate any reports summarizing the results obtained with the tool.

This paper presents a new improved version of this tool providing all these new features, among others. The tool consists of a graphical interface depicting the various stages of system identification. The paper emphasis is on describing the tool that examines the integrated effect of experimental design and model structure selection on prediction-error estimation. Both pseudo-random binary sequence (PRBS) and minimum crest factor multisine inputs are applied for ARX, ARMAX, Output Error (OE), Box-Jenkins (BJ), and State Space (SS) estimation of this system [Braun et al., 2002]. Experimental duration, estimation and crossvalidation data sets, input signal bandwidth and magnitude, and model structure are evaluated under varying signal-to-noise ratios, with all results computed and displayed interactively to the user. The interactive tool is coded in Sysquake, a Matlab-like language with fast execution and excellent facilities for interactive graphics [Piguet, 2004]. Executable files for the modules that do not require the Sysquake software to operate are in the public domain and available for Windows, Mac, and Linux operating systems [Guzmán et al., 2009].

The paper is organized as follows: a brief description on the theoretical background behind the tool is presented in Section 2. A summary of the tool's functionality is presented in Section 3. A series of illustrative examples are presented in Section 4. The paper concludes with a brief discussion of development plans for the future.

2. THEORETICAL BACKGROUND

The theoretical background behind *ITSIE* is presented in Guzmán et al. [2009]; we summarize here the more salient points, with emphasis on the simulation mode. In *ITSIE*, the plant to be identified consists of a discrete-time system sampled at a value specified by the user (default value $T = 1$ min) and subject to noise and disturbances according to

$$y(t) = p^*(q)\ (u(t) + n_1(t)) + n_2(t). \tag{1}$$

In (1), $y(t)$ is the measured output signal and $u(t)$ is the input signal that is designed by the user. $p^*(q)$ is the zero-order-hold-equivalent transfer function for $p(s)$, where q is the forward-shift operator. The system is subject to two stationary white noise sources (n_1 and n_2) introduced at different locations in the plant. n_1 allows evaluating the effects of autocorrelated disturbances in the data, while n_2 introduces white noise directly to the output signal.

A comprehensive system identification procedure consists of experimental design and execution, data preprocessing, model structure selection and parameter estimation, and model validation. The following are emphasized in the tool:

- *Experimental design and execution.* The success of any identification methodology hinges on the availability of an informative input/output data set obtained from a sensibly designed identification experiment. In *ITSIE*, deterministic, periodic signals relying on pseudo-random binary sequence (PRBS) and multisine inputs are considered. A PRBS is binary signal generated by using shift register modulo 2 addition. One cycle of a PRBS sequence is determined by the number of registers n_r and the switching time T_{sw}. The signal repeats itself after N_sT_{sw} units of time, where $N_s = 2^{n_r} - 1$; a_{mag} is the magnitude of the PRBS signal. Multisine signals are deterministic, periodic signals, represented in the single input case by the equation

$$u(k) = \lambda \sum_{i=1}^{n_s} \sqrt{2\alpha_i}\ cos(\omega_i kT + \phi_i) \tag{2}$$

$$\omega_i = 2\pi i/N_sT, \quad n_s \leq N_s/2$$

The power spectrum of the multisine input is directly specified through the selection of the scaling factor λ, the Fourier coefficients α_i, the number of harmonics n_s, and the signal length N_s.

Both direct parameter specification and applying time constant-based guidelines for input design are evaluated in the tool. In practice, little is known about the process dynamics at the start of identification testing, and plant operating restrictions will discourage excessively long or very intrusive identification experiments. A guideline that provides a suitable estimate of the frequency band over which excitation is required is

$$\frac{1}{\beta_s \tau_{\text{dom}}^H} \leq \omega \leq \frac{\alpha_s}{\tau_{\text{dom}}^L}, \tag{3}$$

where τ_{dom}^H and τ_{dom}^L are high and low estimates of the dominant time constant, and β_s is an integer factor representing the settling time of the process. For example, $\beta_s = 3$; specifies the low frequency bound using the 95% settling time ($T_{95\%}$) of the process. α_s, meanwhile, is a factor representing the closed-loop speed of response, written as a multiple of the open-loop response time.

Equation (3) is used in *ITSIE* to specify design variables in both PRBS and multisine inputs. Expressions for specifying T_{sw} and n_r based on (3) are developed in Rivera [1992]:

$$T_{sw} \leq \frac{2.8\tau_{\text{dom}}^L}{\alpha_s}, \quad N_s = 2^{n_r} - 1 \geq \frac{2\pi\beta_s\tau_{\text{dom}}^H}{T_{sw}} \tag{4}$$

n_r and N_s are integer values, while T_{sw} is an integer multiple of the sampling time T. Similarly, Equation (3) can also be used to specify design variables in multisine inputs, using guidelines found in Rivera et al. [1993]

$$N_s \geq \frac{2\pi\beta_s\tau_{dom}^H}{T}, \qquad n_s \geq \frac{N_sT\alpha_s}{2\pi\tau_{dom}^L} \tag{5}$$

In both cases increasing α_s and β_s will widen the frequency band of emphasis in the input signal and increase the resolution of the input signal spectrum. To reduce model variance it is beneficial to apply the highest input signal amplitudes a_{mag} or λ that operations will allow, and implement the greatest number of input cycles m possible. In practice, decisions regarding the magnitude of the input signal, spectral content, and experimental test duration are dictated by physical limitations, economics, and safety considerations Ljung [1999].

In multisine inputs, the choice of phase angles ϕ_i does not influence the power spectrum, but it does strongly influence plant-friendly metrics such as crest factor [Rivera et al., 2003]. Both the work of Schroeder [1970], who derives a closed-form formula to select the phases in (2) and the successive p-norm approach by Guillaume et al. [1991] are implemented in *ITSIE*.

- *Data preprocessing. ITSIE* data preprocessing supports mean subtraction, differencing, and substraction of baseline values; mean detrending is applied by default. Future versions of the tool will emphasize issues in prefiltering and control-relevance.
- *Model structure selection and parameter estimation. ITSIE* examines the general family of prediction-error models which corresponds to

$$A(q)y(t) = \frac{B(q)}{F(q)}u(t-nk) + \frac{C(q)}{D(q)}e(t) \qquad (6)$$

$$y(t) = \tilde{p}(q)u(t) + \tilde{p}_e(q)e(t) \qquad (7)$$

where

$$A(q) = 1 + a_1 q^{-1} + \ldots + a_{n_a} q^{-n_a}$$
$$B(q) = b_1 + b_2 q^{-1} + \ldots + b_{n_b} q^{-n_b+1}$$
$$C(q) = 1 + c_1 q^{-1} + \ldots + c_{n_c} q^{-n_c}$$
$$D(q) = 1 + d_1 q^{-1} + \ldots + d_{n_d} q^{-n_d}$$
$$F(q) = 1 + f_1 q^{-1} + \ldots + f_{n_f} q^{-n_f}$$

The five most popular PEM models shown in Table 1 are evaluated in *ITSIE*, with FIR belonging as a subset of ARX models. The tool also includes PEM estimation of state-space models.

Method	$\tilde{p}(q)$	$\tilde{p}_e(q)$
ARX	$\frac{B(q)}{A(q)}q^{-nk}$	$\frac{1}{A(q)}$
ARMAX	$\frac{B(q)}{A(q)}q^{-nk}$	$\frac{C(q)}{A(q)}$
FIR	$B(q)q^{-nk}$	1
Box-Jenkins	$\frac{B(q)}{F(q)}q^{-nk}$	$\frac{C(q)}{D(q)}$
Output Error	$\frac{B(q)}{F(q)}q^{-nk}$	1

Table 1. Prediction-error model structures evaluated in *ITSIE*.

As noted in Ljung [1999], PEM estimation involves either linear and nonlinear regression, depending on the model structure being evaluated.

$$\arg \min_{\tilde{p},\tilde{p}_e} \frac{1}{N}\sum_{i=1}^{N} e^2(i) = \arg\min_{\theta} \frac{1}{N}\sum_{i=1}^{N}\left[y - \varphi^T(t|\theta)\theta\right]^2 (8)$$

The use of Parseval's Theorem enables a frequency-domain analysis of bias effects in PEM estimation that allows deep insights into the selection of design variables for these techniques. As the number of observations $N \to \infty$, the least-squares estimation problem denoted by (8) can be written as:

$$\lim_{N\to\infty} \frac{1}{N}\sum_{i=1}^{N} e^2(t) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \Phi_e(\omega)d\omega \qquad (9)$$

where $\Phi_e(\omega)$, the prediction-error power spectrum is

$$\Phi_e(\omega) = \frac{1}{|\tilde{p}_e(e^{j\omega})|^2}\left(|p^*(e^{j\omega}) - \tilde{p}(e^{j\omega})|^2\Phi_u(\omega) + |p^*(e^{j\omega})|^2\sigma_{n_1}^2 + \sigma_{n_2}^2\right) \qquad (10)$$

Equation (10) helps explain systematic bias effects in identification, which can be readily explored in *ITSIE*. This includes issues relating to the spectral content in the input signal, bias that is introduced (or removed) by the choice of model structure (particularly the noise model), and the associated multi-objective optimization problem resulting from varying magnitudes of the noise variances $\sigma_{n_1}^2$ and $\sigma_{n_2}^2$.

- *Model validation.* In *ITSIE*, model validation consists principally of classical methods of simulation, cross-validation, residual analysis on the prediction errors, and step responses. To enhance its educational value, in the simulation mode the step response of the true plant is presented alongside that generated by the estimated models. The percent of the output variance explained by each model on the crossvalidation data set is also reported.

Leveraging the interplay between the various stages of the identification problem is readily supported in *ITSIE*. One example is ARX estimation, where model structure selection can be accomplished without substantial user intervention through the sensible use of crossvalidation. Because ARX parameter estimation consists of solving a linear least squares problem, a large number of model structures defined by ranges for n_a, n_b and n_k can be evaluated without incurring significant computational burden. The model order that minimizes the loss function over a crossvalidation data set can be obtained without iteration.

3. INTERACTIVE TOOL DESCRIPTION

This section briefly describes the functionality of the developed tool, which highlights the theoretical concepts described in the previous section. The tool is freely available through `http://aer.ual.es/ITSIE/` and does not require a Sysquake license in order to execute [Guzmán et al., 2009]. One consideration that must be kept in mind is that the tool's main feature - interactivity - cannot be easily illustrated with written text. Nonetheless, some of the features and advantages of the application are shown below. The reader is cordially invited to download the tool and personally experience its interactive features.

When developing a tool of this kind, one of the most important things that the developer needs to keep in mind is the organization of the main windows and menus to facilitate to the user an understanding of the identification technique [Dormido, 2004, Guzmán, 2006]. The tool has two different modes, a simulation mode and a real data mode, which are depicted in Figures 1 and 2, respectively. The ensuing subsections briefly describe the main features of these modes.

3.1 Simulation mode

In this mode, a user-specified simulated process is evaluated. The graphical distribution has been performed according to the most important steps in a system identification process, described as follows (see Figure 1):

- *Plant definition and simulation parameters.* The central part of the tool in this mode has a section called Simulation parameters, which allows interactively modifying the noise sources of the simulated

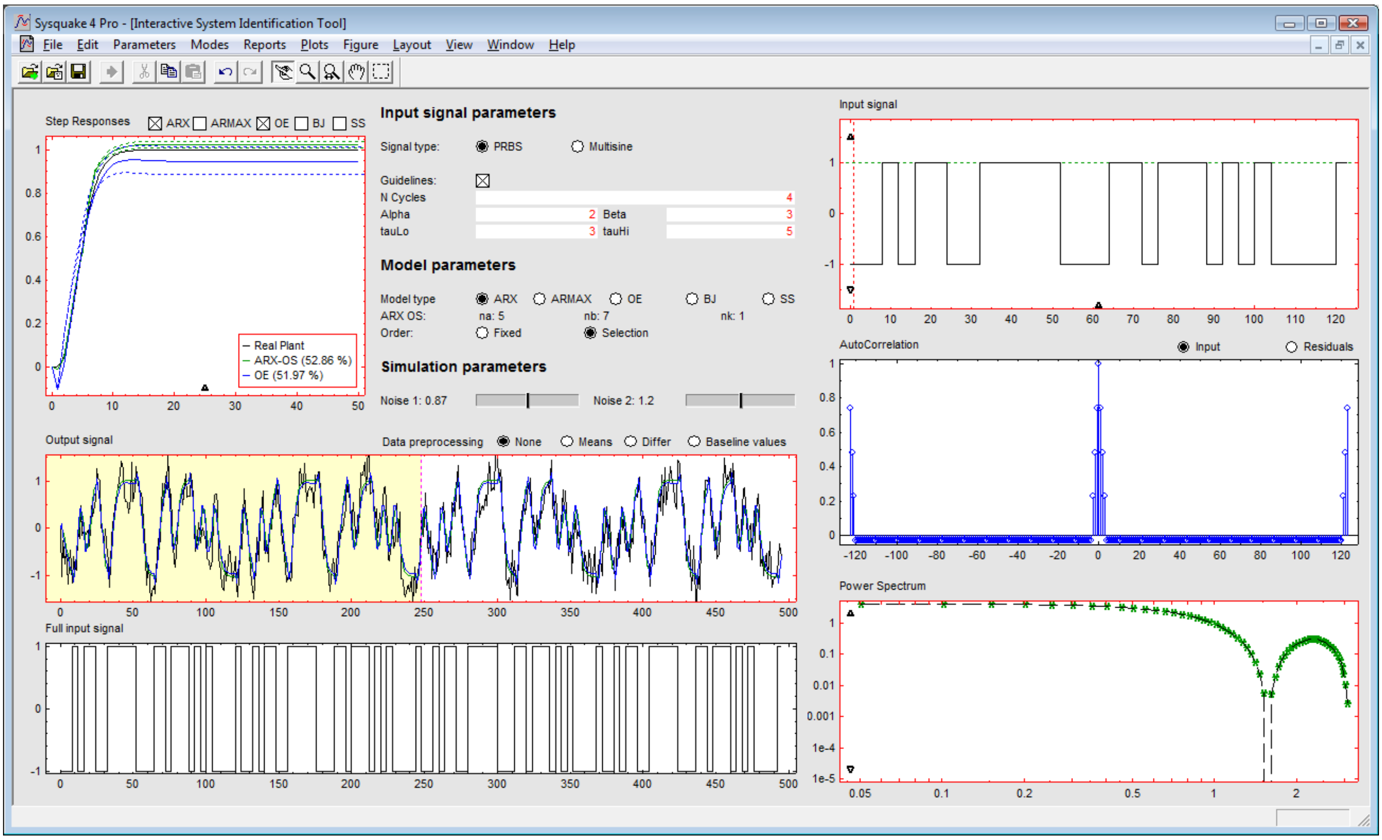

Fig. 1. *ITSIE* interactive tool user interface demonstrating four cycles of a PRBS input applied to a simulated fifth-order system. The time-constant guidelines from Section 2 are used to define input parameters. An OE-[2 2 1] model is compared with an ARX-[5 7 1] model obtained from exhaustive order selection on a crossvalidation data set.

process. On the other hand, other simulation parameters, such as sampling time, are available from an entry at the Parameters menu. Furthermore, the simulated process can be configured from the Mode→Simulation menu. The process model configuration can be also loaded and stored from files.

- *Input design.* A parameter definition section and three interactive graphics characterize the input design stage. The parameter definition section is called Input signal parameters, being located at the top of the middle section of the tool. The three graphics are located at the right-hand side of the tool, namely, Input signal, Autocorrelation, and Power Spectrum, representing one cycle of the input signal, the input signal autocorrelation, and the input signal power spectrum, respectively (see Figure 1). From the Input signal parameters area, the user can choose the type of the input signal (PRBS or multisine) and whether to use the checkbox called Guidelines to decide between specifying the input signal directly or following the guidelines mentioned in Section 2. When the user does not select the guidelines, that is, the Guidelines checkbox is not active, the input signal parameters can be interactively modified using specific sliders or dragging on the graphics.
- *Model structure selection and parameter estimation.* On top of the Step responses graphic, located on the upper left-hand side of the tool, there is a set of checkboxes allowing to active the different model structures, namely, ARX, ARMAX, OE, BJ, and SS. Once a model structure is selected, the estimation and validation results for that model are shown in corresponding parts of the tool. Below the Input signal parameters section there is an area called Model parameters showing parameters to modify the orders of the different model structures. Several radio buttons are available to choose between the different model structures. Once a model structure is selected, different sliders appear making it possible to modify the associated orders interactively. Once a input signal has been configured, the final input with all the desired cycles is shown in a graphic called Full input signal, which is located at the lower-left corner of the tool. This full input signal is applied to the simulated plant with noise in order to obtain the simulated "real data" (shown in black in the Output signal graphic), which is used as real process data in the estimation and validation process. In the Output signal graphic, an interactive magenta vertical line defines the estimation and validation data sets. The area shown in yellow (at the left of the vertical line) specifies the estimation data, whereas the white area represents the validation data (at the right side of the vertical line).
- *Model validation.* As mentioned in the previous bullet, the magenta-colored vertical line of the Output signal graphic is interactively used to define the estimation and validation data sets. The validation data is used for crossvalidation purposes. Model validation results are displayed in other three different graphics: Step

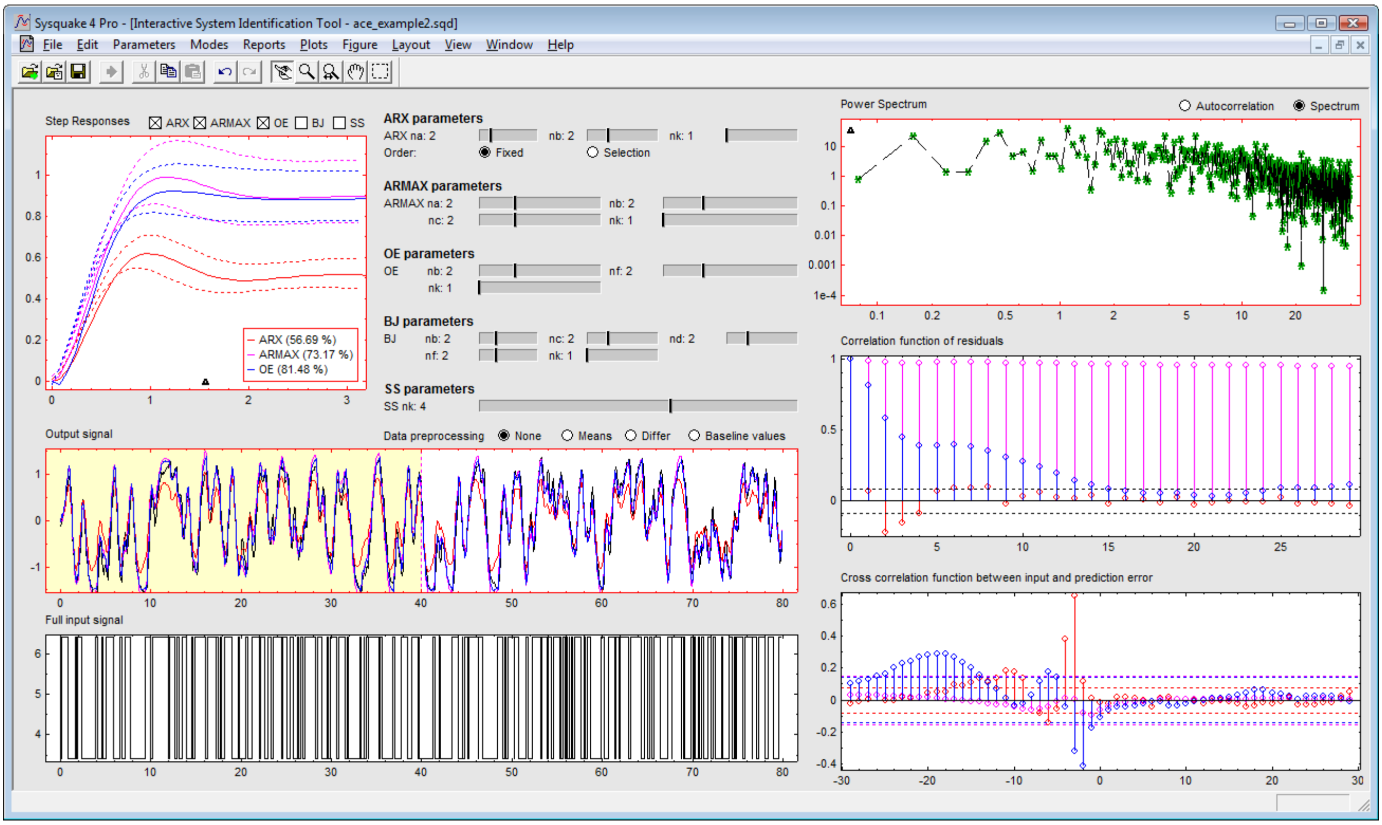

Fig. 2. Real mode of the *ITSIE* interactive tool evaluating external data corresponding to the System Identification Toobox's "hairdryer" data set in Matlab. Model estimates for ARX, ARMAX and OE estimation are shown along with residual analysis of the prediction errors.

responses, Correlation function of residuals, and Cross correlation function between input and output. For all these graphics, the same color distribution noted before is used to represet the results of each model. The Step responses graphic, which is located at the upper left-hand side of the tool, shows the step responses for the each resulting model and a legend representing its goodness of fit in %. Confidence intervals can be also shown in this graphic activating this option from the Parameters menu.

3.2 Real data mode

This mode allows to load real data from the Mode→Real data menu. The real data can be loaded in ASCII and Matlab formats [1]. For ASCII format, the data must be organized in columns with the following order: time, output, and input signals. If the Matlab format is used, the file must contain three variables called "t", "y", and "u" for the time, the output, and the input, respectively. When real data is loaded, the tool screen is changed such as shown in Figure 2. As it can be observed, those areas in the simulation mode dedicated to input design and plant definition and simulation parameters are changed. The *Model Structure Selection*, *Parameter Estimation* and *Model Validation* areas are exactly the same than in the simulation mode, but now working with real data loaded from file. In this mode, all the model parameters are always shown simultaneously on the right side of the Step responses graphic.

[1] For the Matlab format, the data must be compatible with Matlab version 4. Use -V4 option with "save" Matlab command.

3.3 Additional options for education

The tool has been complemented with some additional options to be used for educational purposes. For instance, the teacher can define his/her own process model for the simulation mode using the Mode→Simulation menu, such as mentioned above. Once the the process model is defined, the teacher can export the model into a file and share it with the students. Notice that the model is hidden for the students. On the other hand, students and teachers can obtain detailed reports of the results from the Reports menu. The reports include information about the resulting identified models, e.g., goodness of fit, model structure, model parameters, and transfer functions in Matlab format.

4. ILLUSTRATIVE EXAMPLES

We have noted that there are large numbers of possible scenarios with educational value that can be illustrated by the *ITSIE* tool. The list below is by no means exhaustive, but representative of some valuable concepts:

(1) The importance of selecting crossvalidation data, and how it impacts parameter estimation, particularly the effectiveness of automated order selection in ARX estimation.

(2) A comparison between two different input signal types (i.e., PRBS versus multisines) and the usefulness of crest factor minimization for achieving "plant-friendliness" [Rivera et al., 2003].
(3) Understanding the issue of persistent excitation, as displayed in the interrelationship between input design and model order selection. This is particularly useful when using the multisine input signal, given that the user can directly specify the number of nonzero harmonics in this signal.
(4) The importance of taking advantage of *a priori* knowledge in input design. The time-constant guidelines presented in Section 2 can be thoroughly evaluated and appreciated.
(5) The relative merits of various validation criteria. Correlation analysis on the residuals may indicate that there is still a need to refine on model structure; however, the model may still describe a large percentage of the output variance in the validation data and closely match the plant step response.

Figures 1 and 2 depict two interesting cases evaluated with the tool. Figure 1 presents the use of a PRBS signal designed using the guidelines in Section 2 for the simulated fifth-order system according to

$$p(s) = \frac{1}{(s+1)^5}, \quad T = 1 \text{ min} \tag{11}$$

Four cycles of data are generated, with two used for estimation, and two used for validation purposes. An OE-[2 2 1] model is compared to an ARX-[4 5 3] model structure obtained from systematic order selection over a range of model structures; this model has a superior fit over the crossvalidation dataset.

Figure 2 illustrates the real mode of the *ITSIE* interactive tool, evaluating the "hairdryer" data set that is used as an illustrative example in Matlab's System Identification Toolbox. Model estimates for ARX-[2 2 1], ARMAX-[2 2 2 1] and OE-[2 2 1] estimation are shown along, with residual analysis of the prediction errors. The residual analysis results indicate the need to modify the model structure in each of these estimation scenarios.

5. CURRENT USE AND FUTURE PLANS

The first official use of *lTSIE* in a classroom setting was as part of a system identification short course taught at the University of Almería in September, 2008. It was also used as part of ChE 494-598: Introduction to System Identification, a combined undergraduate-graduate level course taught at Arizona State University in spring 2009. Student response has been positive, and has provided input for further refinement and organization of the tool.

This paper has focused on the initial offerings of what we envision as a comprehensive family of novel interactive tools for system identification. Future tools will examine the interplay between input design, data prefiltering, and model structure on control-relevance, as well as tradeoffs in closed-loop identification and issues in multivariable system identification.

REFERENCES

M.W. Braun, R. Ortiz-Mojica, and D.E. Rivera. Control Engineering Practice. *Design of Minimum Crest Factor Multisinusoidal Signals for Plant-Friendly Identification of Nonlinear Process Systems*, 3(3):301–313, 2002.

S. Dormido, S. Dormido-Canto, R. Dormido, J. Sánchez, and N. Duro. The Role of Interactivity in Control Learning. *International Journal of Engineering Education*, 21 (6):1122–1133, 2005.

S. Dormido. Control learning: present and future. *Annual Reviews in Control*, 28(1):115–136, 2004.

H. Garnier and M. Mensler. CONTSID toolbox: a matlab toolbox for continuous-time system identification. In *Proc. 12th IFAC Symposium on System Identification*, Santa Barbara, USA, 2000.

P. Guillaume, J. Schoukens, R. Pintelon, and I. Kollár. Crest-factor minimization using nonlinear Chebyshev approximation methods. *IEEE Trans. on Inst. and Meas.*, 40(6):982–989, 1991.

J.L. Guzmán, K.J. Åstroṁ, S. Dormido, T. Hägglund, M. Berenguel, and Y. Piguet. Interactive learning modules for PID control. *IEEE Control System Magazine*, 28(5):118–134, 2008.

J.L. Guzmán, M. Berenguel, and S. Dormido. Interactive teaching of constrained generalized predictive control. *IEEE Control Systems Magazine*, 25(2):52–66, 2005.

J. L. Guzmán, D.E. Rivera, S. Dormido, and M. Berenguel. ITSIE: An interactive software tool for system identification education. In *Proc. 15th IFAC Symposium on System Identification*, page `http://aer.ual.es/ITSIE/`, St. Malo, France, July 2009.

J. L. Guzmán. *Interactive Control System Design*. Phd thesis, University of Almería, Spain, 2006.

L. Ljung. *System Identification: Theory for the User*. Prentice-Hall, New Jersey, 2nd edition, 1999.

L. Ljung. Educational aspects of identification software user interfaces. In S. Weiland P. van der Hof, B. Wahlberg, editor, *Proc. 13th IFAC Symp. on System Identification*, pages 1590 –1594, Rotterdam, The Netherlands, August 2003a.

L. Ljung. Version 6 of the system identification toolbox. In S. Weiland P. van der Hof, B. Wahlberg, editor, *Proc. 13th IFAC Symp. on System Identification*, pages 989 – 994, Rotterdam, The Netherlands, August 2003b.

Y. Piguet. *SysQuake 3 User Manual*. Calerga S'arl, Lausanne (Switzerland), 2004.

D.E. Rivera, X. Chen, and D.S. Bayard. Experimental design for robust process control using schroeder-phased input signals. In *Proc. of the 1993 American Control Conference*, pages 895 – 899, San Francisco, CA, 1993.

D.E. Rivera, H. Lee, M.W. Braun, and H.D. Mittelmann. Plant-friendly system identification: a challenge for the process industries. In *13th IFAC Symposium on System Identification (SYSID 2003)*, pages 917–922, Rotterdam, Netherlands, 2003.

D.E. Rivera. Monitoring tools for PRBS testing in closed-loop system identification. In *AIChE Annual Meeting*, pages 1–24, Miami, FL, Paper 131d, 1992.

M.R. Schroeder. Synthesis of low-peak-factor signals and binary sequences with low autocorrelation. *IEEE Trans. Info. Theory*, IT-16:85–89, 1970.