A VIRTUAL INSTRUMENT FOR REAL SYSTEMS IDENTIFICATION AND MODELING

C. Fosalau, M. Cretu, O. Postolache, C. Donciu, B. Baraboi

Faculty of Electrical Engineering, "Gh.Asachi" Technical University of Iasi, Iasi, Romania
Phone +40 32 278680 Fax +40 32 237627 e-mail: cfosalau@ee.tuiasi.ro

Abstract – *The paper describes a virtual instrument designed to fast and accurately identify systems working in a large range of frequencies. It comprises both the hardware aimed on input-output data generation and acquisition and the software dedicated to signal processing and all the calculus ended with the model estimation. The instrument offers all the facilities provided by the LabVIEW environment, being also interfaced with the powerful toolboxes of Matlab.*

Keywords – virtual instrumentation, data acquisition, system identification

1. INTRODUCTION

In practice, knowing the actual characteristics and parameters that define the dynamic behavior of a process especially when it is intended to be controlled is of great importance. This aim is assumed to correctly calculate the regulator with the end of obtaining the best performances upon the controlled parameter and also for obtaining a mathematical model of the system as close to the reality as possible, for computer simulation purposes.

As known, process identification represents the group of operations performed to determine the mathematical model corresponding to the process, starting from the input-output measured data.

The present paper evolved as a normal consequence of our desire to obtain accurate models with minimal efforts for some processes studied in the Control and Automation Laboratory of the Technical University of Iasi, in absence of a device able to concomitantly acquire the input-output data, to calculate and validate the model, to interpret and save the results. Carrying out a single virtual instrument using the versatility and the flexibility of the LabVIEW environment and of some standard hardware fulfilled all these tasks.

From a point of view, our attempt appears as another approach for solving the problem of on-line system identification using the virtual instrumentation than that described in [1], in which a solution using SigLab package is presented. Even if SigLab allows the facility of utilizing the powerful toolboxes for signal processing both in time and frequency domains offered by Matlab, the devices needed to acquire data are rather difficult to handle, and the offered palette of drivers for data acquisition hardware is poor compared to the large diversity of such devices existing on the market.

The virtual instrument presented in this paper is conceived to utilize on one hand all the functions found in Process Identification Toolbox offered by Matlab, and on the other hand the powerful libraries of signal analysis and statistical processing that are included in the LabVIEW program.

However, the essential feature of this instrument is that it can work with various devices for data acquisition and generation without much concern on these devices from the user part. It is also able to auto-detect the device attached to the computer by using the Windows plug-and-play scanning functions and to return its main inputs: channels, maximum scan rate, input limits, etc. The only condition to fulfil this duty is that the driver must be included in the LabVIEW library. Fortunately, the later versions of LabVIEW have in their library drivers for instrumentation and devices produced by almost all the important manufacturers met on the market.

One can also note that the frequency span of the system to be identified and modeled is very large, varying from mHz to MHz. For low system dynamic, a general purpose data acquisition board (DAQ) is enough. For high frequency input-output signals generation and acquisition (in the case of electric and electronic circuit testing), function generators and digital oscilloscopes controlled by means of the GPIB interface can be utilized. Neither in this case the instrument type and producer do account, since these instruments are driven by universal SCPI commands. One thus obtains a large flexibility as regards the hardware involved.

2. INSTRUMENT FUNCTIONS

As we have already shown, the instrument is conceived for simultaneous data acquisition and identification of a real system. Nevertheless, the user has to know previously some approximate data about the system, at least with the aim of establishing the input test signals.

Thus, it is recommended to know from the beginning the process nature (i.e. industrial, electromagnetic or electronic) and the order of magnitude of the time constants involved. Once the preliminary data have been established, when the instrument starts, it is able to accomplish the following functions:

- input test signal generation (binary random sequence, impulse, step, sinusoidal with linear or logarithmic swept frequency);
- simultaneous acquisition of system excitation and response;

- computing and graphical displaying of the non-parametric model (impulse response, step response, Bode diagrams);
- computing and numerical displaying the parametric model (coefficients in time and frequency domains, zeroes-poles representation, state variables model)
- model validation and errors assessing.

For slow systems, identification process (model computing) is performed concomitantly with the input-output data acquisition, using iterative methods [2]. In this case, the system input is represented by binary random sequences. First, the prediction error is fixed to a certain value and an initial model is established. Then, as soon as a new data set is available, the model is corrected. The test advances till the chosen validation criterion provides an error value that does not exceed a given interval. Two major advantages are thus obtained: i) the model estimation is performed on-line so that the two stages, input-output measurements and coefficient computation are not separated in time and ii) the redundant measurements are eliminated since the identification process is stopped when the result meets the validation criterion.

The instrument is conceived for working also with off-line data, previously saved in separated files.

3. INSTRUMENT DESCRIPTION

The virtual instrument is composed by a hardware part comprising standard instruments and devices linked to the host computer and the software part, which drives and controls the whole process.

3.1 Hardware

Fig.1 shows the basic hardware arrangement in the case of slow systems, whilst in Fig.2 the same arrangement is presented for fast systems.

In the case in which the system bandwidth does not exceed about 10 kHz, for input test signal generation and output acquisition, a general-purpose digital acquisition card (DAQ) endowed with analog input-outputs has been employed. When the system under test is too complex, the number of analog input-outputs needed might exceed what DAQ can provide. In this case, a multiplexer controlled by the digital outputs of the board can be utilized to expand its capabilities, but care has to be taken when assessing phase shifts, due to interchannel delays. By means of analog

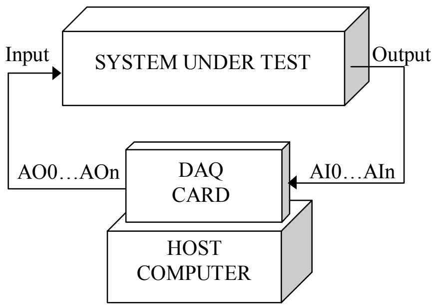

Fig.1 - Block diagram of the instrument for low variable systems identification

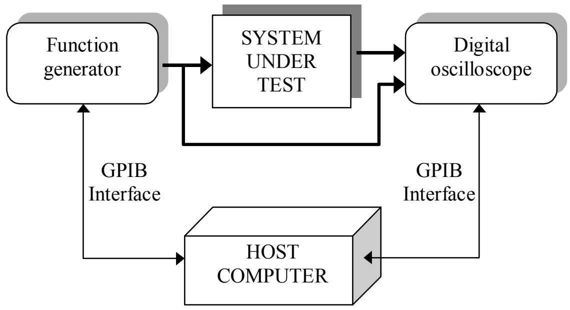

Fig.2 - Block diagram of the instrument for high speed systems identification

outputs we can simulate both the control and disturbance inputs, making possible to study the system operation under various regimes.

If the system bandwidth is too large, the test signals are generated by a function generator whose operation is remotely controlled by the host computer through the GPIB communication protocol. The system response to the test signal is acquired by a digital oscilloscope having the real sampling frequency as high as at least 10 times the maximum frequency applied to the system. The acquired data are also send via GPIB to the host computer.

Even if this arrangement seems to satisfy also the case of slow systems, it is more expensive and the number of inputs is limited to the number of generator outputs, which usually does not exceed two outputs. However, it is very suited to test and model electrical quadripoles.

3.2 Software

The entire program that deals with hardware control, data processing and results displaying is conceived in LabVIEW. Thus, all the facilities regarding file operations, default information and preferences provided by LabVIEW user interface are accessible.

The flow diagram of the program is presented in Fig.3. In this figure we used the following abbreviations: BRS-binary random sequence, SR-step response, IR-impulse response, FR-frequency response, SS-state space, TF-transfer function. The model estimation is performed using binary random sequences as test signals. The sequence is software generated according to the previous knowledge about the system, as discussed in section 2. The algorithms used for identification are based on general linear input-output models ([3]-[5]) with the special cases used in Matlab: ARX, ARMAX, Output Error and Box-Jenkins. They are implemented in LabVIEW through a Matlab script structure, which allows the user to utilize also other functions provided by Matlab.

In order to trace the real frequency characteristics of the system, the input is fed by a sinusoidal signal having the frequency swapped linearly or logarithmically in a given interval. The DAQ or the signal generator, depending on the frequency span, generates this signal. The characteristics are automatically traced step by step. All other responses are traced by feeding the input(s) with appropriate signals. Once the model was established, all the responses are simulated

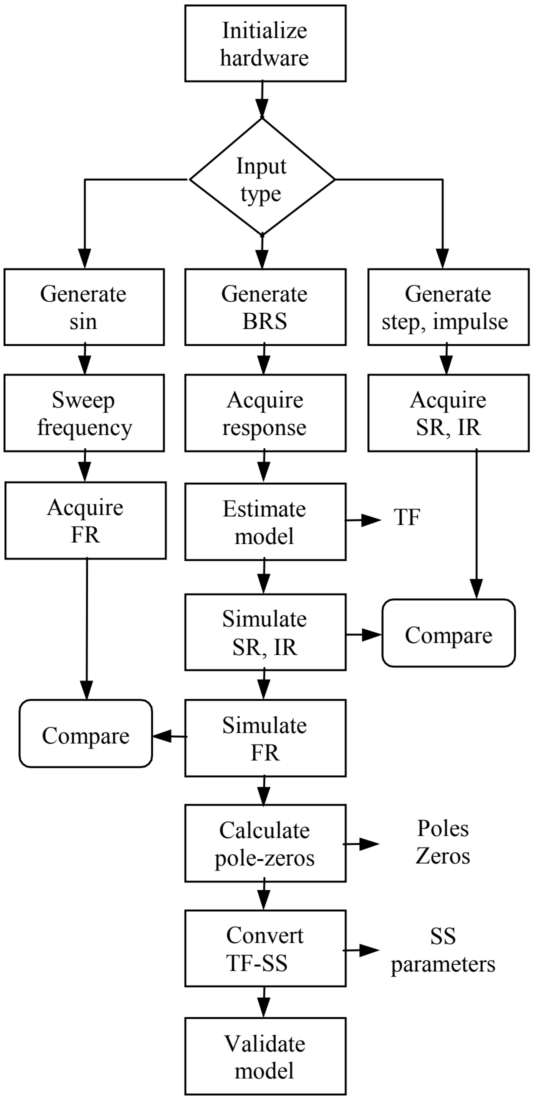

Fig.3 – Flow diagram of the identification process

and the results compared with the real ones. The mean square fit between the measured output and the simulated one is provided in numerical format and residuals are graphically displayed.

The front panel of the instrument represents the graphical interface that allows the user to establish a series of parameters related to hardware settings, shape and amplitude of the test signals, identification algorithm, and to visualize the system responses, residuals and model parameters. Three forms of the model are available as output: transfer function coefficients, poles-zeros model and state space model. The model output is available on the same window in both parametric and non-parametric format, in graphical and numerical forms.

4. EXPERIMENTAL RESULTS

In this section, we shall give an example to show how to utilize the instrument for estimating the model of a dc motor of 10 W, whose speed is controlled by using a classical loop, taking as parameter the speed measured with an analogue tachogenerator (Fig.4). A PD regulator whose parameters can be manually adjusted performs the control. The experiments have been carried out for different time constants of the regulator. As outputs, the signals that drive every block have been considered. Thus, the overall process along with its component parts was able to identify. The prescribed speed delivered by one of the analog outputs of the DAQ card was considered as input. The test signal utilized for identification process was a binary random sequence built with the aid of the random number generator provided by the LabVIEW library. The time step was 40 ms for all experiments.

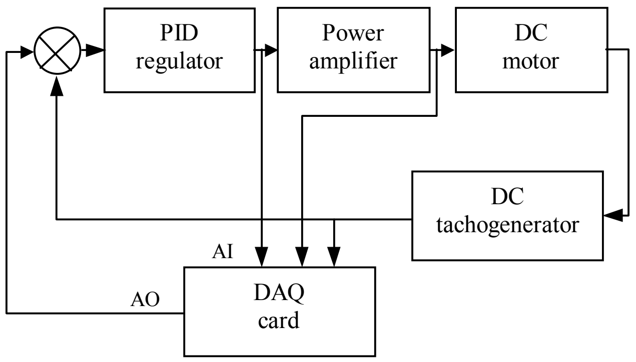

Fig.4 – Experimental arrangement for instrument testing

The DAQ card employed was a National Instruments device type AT-MIO-16E-10, plugged into an IBM compatible host computer.

In Fig. 5, four characteristics traced for the same system under various control conditions are presented. They are referred only to the step response, but the same conclusions can be drawn when considering other non-parametric characteristics. In these examples, the prescribed speed was considered as input, while the magnitude of the signal delivered by the tachogenerator was the output. The measurements were performed under various load couple values applied to the motor shaft.

The model which satisfied the best our needs was a [2 2 1] model, involving 2 zeros, 2 poles and 1 delay. In the table 1, the values of b_k and a_k coefficients of the transfer function along with their average covariance are presented, for several values of K_p and τ_d, where K_p is the proportional parameter of the regulator and τ_d is its derivative time constant.

Table 1 – Experimental results for various values of the regulator parameters

K_p	τ_d	b_0	b_1	a_0	a_1	a_2	cov
0.45	0	0.17	0.08	1	-1.06	0.33	0.009
0.45	0.04	0.19	0.08	1	-0.95	0.94	0.012
0.45	0.06	0.17	0.13	1	-0.76	0.089	0.021
0.9	0.07	0.18	0.11	1	-0.69	0.007	0.025
0.9	0.08	0.18	0.12	1	-0.57	-0.11	0.031
0.9	0.1	0.17	0.14	1	-0.46	-0.16	0.038

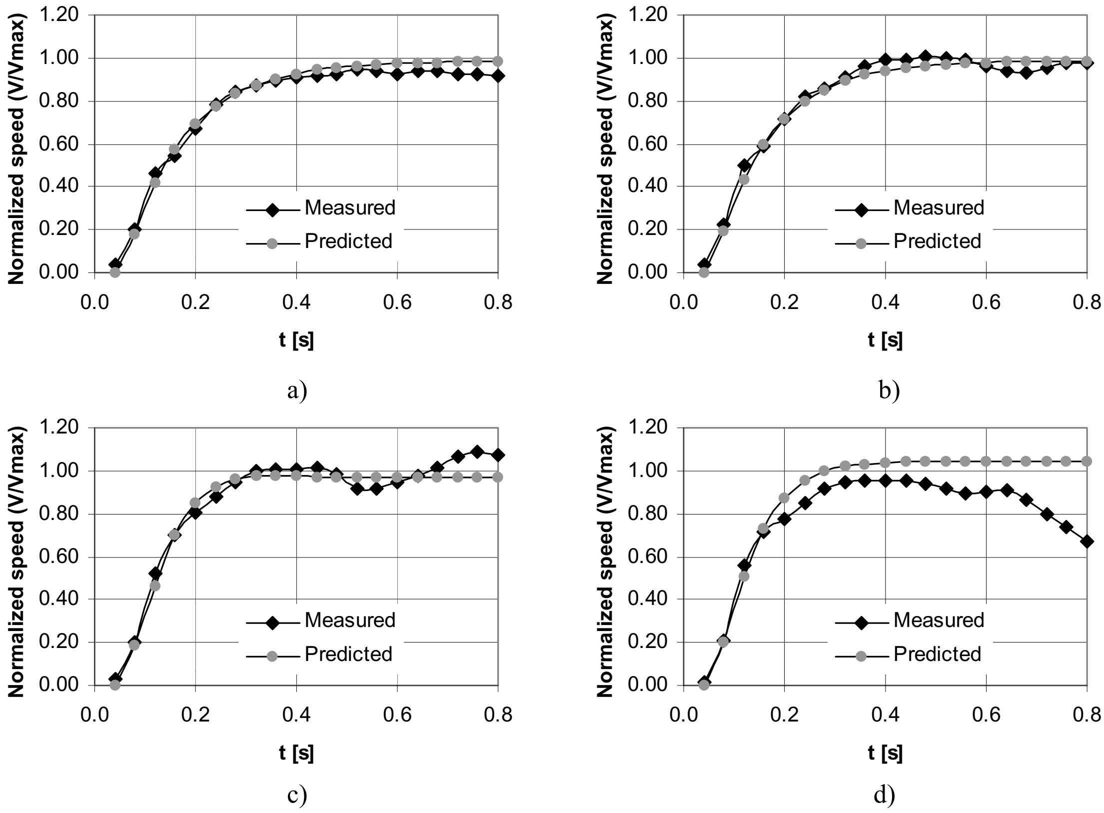

Fig.5 – Measured and predicted step responses for a controlled dc motor when: a) K_p=0.45,τ_d=0; b) K_p=0.45,τ_d=0.04; c) K_p=0.9,τ_d=0.06; d) K_p=0.9,τ_d=0.08;

As can be observed from table 1, the average covariance increases as τ_d grows, that is as the derivative component is more pregnant. This could be explained by the fact that the system works in oscillating regime and the number of points acquired in a period is not enough for correct assessing the prediction error in the identification algorithm. This problem can be corrected by diminishing the time step.

It should be noted that the experimental characteristics traced with test signals other than BRS are purely informative and does not account in the identification process.

5. CONCLUSIONS

An effective and easy-to-built instrument, very helpful in applications that require fast and accurate dynamic process identification and modeling was presented in the paper. Without much concern on the hardware part, the user is allowed to set all the parameters related to preliminary information about the system under test, identification algorithm and data processing by using a friendly graphical interface. On the same panel, the results in graphical and numerical forms are provided, allowing the user to observe the behavior of the system under various input signal tests and conditions.

REFERENCES

[1] J.G. Nemeth, B.Vargha and I. Kollar, "Online frequency domain system identification based on a virtual instrument", *IEEE Trans. on Instr. Meas.* vol. 49, no.6, pp. 1260-1263.

[2] L. Dugard, I.D. Landau, "Recursive Output Error Identification Algorithms: Theory and Evaluation", *Automatica,* No. 16, 1980, pp. 443-462.

[3] I.D. Landau, *Identification et commande des systemes*, Ed. Hermes, Paris, 1993.

[4] K. Ogata, *Modern Control Engineering*, Prentice Hall, New Jersey, 1990.

[5] L. Ljung, *System Identification - Theory for the user*, Prentice Hall, New Jersey, 1987.

[6] M.L. Chugani, A.R. Samant, *LabVIEW - Signal Processing*, Prentice Hall, New Jersey, 1998.