

Simulation Time Analysis of MATLAB/Simulink and LabVIEW for Control Applications

Civan Artun Cansalar

Dept. of Electrical and Electronics Engineering TOBB ETU
Systems Test and Evaluation Dept. ROKETSAN
Ankara, Turkey
cacansalar@etu.edu.tr

Ertan Maviş

Dept. of Electrical and Electronics Engineering TOBB ETU
Systems Test and Evaluation Dept. ROKETSAN
Ankara, Turkey
emavis@etu.edu.tr

Coşku Kasnakoğlu

Dept. of Electrical and Electronics Engineering
TOBB ETU
Ankara, Turkey
kasnakoglu@etu.edu.tr

Abstract— Nowadays advanced software programs are actively used in the field for simulation and modeling. LabVIEW and MATLAB/Simulink are both graphical programming languages, allowing a more efficient method to perform and analyze control systems. Therefore, they are the most commonly used software programs for these purposes. Moreover; in some applications - like hardware in the loop systems, flight simulations etc. - simulation time can be vital to achieve more realistic simulation. Therefore, in addition to programming style, simulation performance can be affected by used platform's abilities. In this paper both tools' abilities on some of control applications will be presented with benchmark problems and their performances on simulation of control systems will be analyzed and compared.

Keywords—LabVIEW; MATLAB; Simulink; controller; simulation; performance comparison; HWIL

I. INTRODUCTION

LabVIEW and MATLAB/Simulink are two commonly used graphical programming platforms unlike conventional text based programming languages such as C, VHDL, Assembly etc. programming is made using blocks and creating flowcharts. In addition to fast programming, graphical programming languages creates more flexible and observable test and measurement environment for control applications [6].

Both softwares are widely used by engineers, scientists, and students in various applications. Easy and fast programming is one of the reasons that these softwares are used frequently in the field. In addition, they have extensive libraries that most of users can easily find what they need. Because these two platforms are used by many people, sample codes can be obtained via internet.

Many applications can be performed on these platforms. Thus, there are some surveys in literature which compares LabVIEW and MATLAB/Simulink from different aspects [1]. If the application consists of just mathematical functions, calculated results will be the same in both platforms. However, we may face some applications such that they have

different requirements. For example; if the application is time critical, analysis for simulation time is a major factor to choose the platform.

In this study, a test procedure is developed to analyze timing performances of both platforms for three types of controllers, namely, P-controllers, PI-controllers and PID-controllers. A simple DC motor is modelled in both platforms as a system to be controlled.

Moreover; exact same controllers are developed in LabVIEW [2] and MATLAB/Simulink [3]. All tests are conducted on the same computer while same programmes are running. In this way the effects of hardware and operating system are negligible.

The rest of this paper is organized as follows: Section 2 gives briefly information about LabVIEW and MATLAB/Simulink. Simple DC motor used in this study is described and modelled in section 3. P, PI and PID controllers for the DC motor are developed and timing performances for created simulations are analysed in both platforms in section 4. Our analysis and results are presented in section 5. Finally, discussions and conclusions are provided in section 6.

II. LABVIEW AND MATLAB/SIMULINK

A. LabVIEW

LabVIEW is a software developed by National Instruments. LabVIEW is commonly used by engineers, scientists and students for data acquisition, instrument control and industrial automation. LabVIEW has support for Microsoft Windows, MAC OS X and some versions of Linux. In this study, LabVIEW 2013 SP1 is used on a PC which runs Windows 7 64 bit on Intel Xeon W3550 processor with 8GB RAM.

B. MATLAB/Simulink

Simulink is an add-on application of MATLAB which is developed by MathWorks. Simulink is also a graphical programming language for simulating and analyzing systems and a cross-platform software which can run on Microsoft

Windows, MAC OS X, and some versions of Linux. In order to avoid effects of hardware and effects of operating system, Simulink 7.3 and MATLAB R2009s is used on the exact same PC with LabVIEW which runs Windows 7 64 bit on Intel Xeon W3550 processor with 8GB RAM.

III. DC-MOTOR MODEL

Generally, DC motor models are used in control design examples because of their simplicity and controllability. In order to make our benchmarks more understandable and independent of complexity of the system, DC motor model [4] is used as the main system which controllers are designed for [5]. DC Motor whose transfer function is used for controller models in LabVIEW and MATLAB/Simulink is shown in Fig 1. After arranging formulas related with DC Motor model, state space of the DC Motor system is calculated as (1) and parameters used in the model are shown in Table 1.

$$\begin{bmatrix} \dot{i} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} -\frac{R}{L} & -\frac{K_b}{L} \\ \frac{K_m}{J} & -\frac{K_f}{J} \end{bmatrix} \begin{bmatrix} i(t) \\ w(t) \end{bmatrix} + \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} V_a(t) \quad (1)$$

$$y(t) = [0 \quad 1] \begin{bmatrix} i(t) \\ w(t) \end{bmatrix} + [0]V_a(t)$$

Transfer function can be found as in (2) by using parameters in Table 1 and space equations given in (1).

$$TF = \frac{10}{s^2 + 14s + 41} \quad (2)$$

IV. CONTROLLER DESIGNS AND COMPUTATION TIMES

In this section, first of all, P, PI, PID controllers are designed in both platforms. "Control Design and Simulation Module" of LabVIEW and "Control System Toolbox" of MATLAB/Simulink are used. Then, after choosing suitable parameters, simulation results and average simulation times are gathered.

In LabVIEW, all simulations are repeated 1000 times using a for loop. In addition to the for loop; stacked sequence and tick count blocks are used to get simulation time for every loop and mean block is used to calculate average of simulation times. In MATLAB/Simulink, simulation time is calculated by defining tic for "StartFcn" and "simulation_time= toc" for "StopFcn". Then; Simulink model is called in a MATLAB script for 1000 times and average of simulation times is calculated using mean function. Measurement methods for both platforms are represented in Fig 2.

TABLE I. DC MOTOR PARAMETERS

Resistance	R = 2.0 ohm
Inductance	L = 0.5 H
Armature Constant	K _m = 0.1 Nm/A
EMF Constant	K _b = 0.1 V/rad/s
Friction Constant	K _f = 0.2 Nms
Rotor Inertia	J = 0.02 kg.m ²

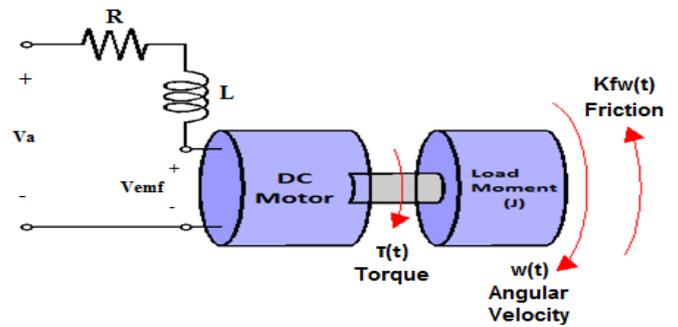


Fig. 1. DC motor scheme

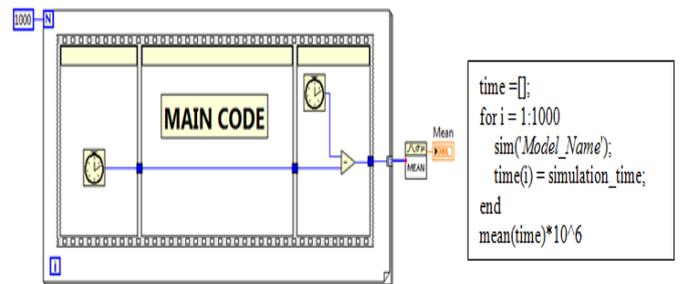


Fig. 2. Measurement Method of Simulation Time for LabVIEW and Simulink

A. P-controller

P-controller is a type of a linear control system. In P-controllers, controller output is proportional to error signal and its transfer function is K_p. For the benchmark application, K_p=15 is chosen.

In both MATLAB/Simulink and LabVIEW, P-controller is designed for DC motor model as shown in Fig 3. Fixed step size "ode4 Runge-Kutta" solver is chosen and all other simulation parameters are set the same values for both platforms. Simulation results can be seen in Fig 4.

System is simulated 1000 times for several step sizes and each of two different simulation times (2s and 60s). Averages of simulation times for both platforms are shown in Table 2.

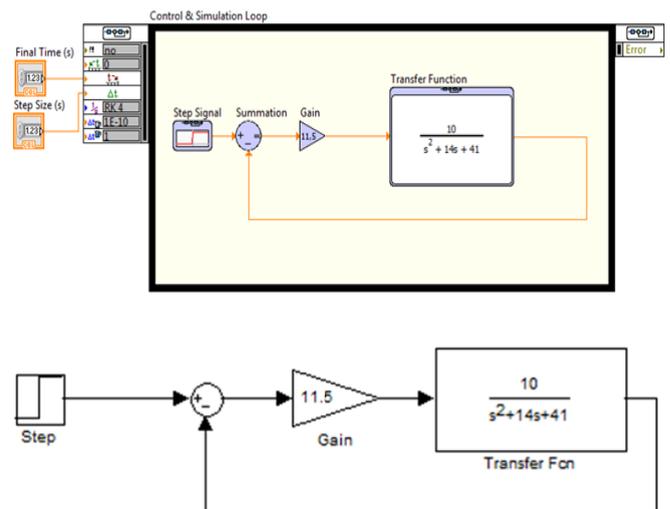


Fig. 3. P Controller Design of LabVIEW and Simulink

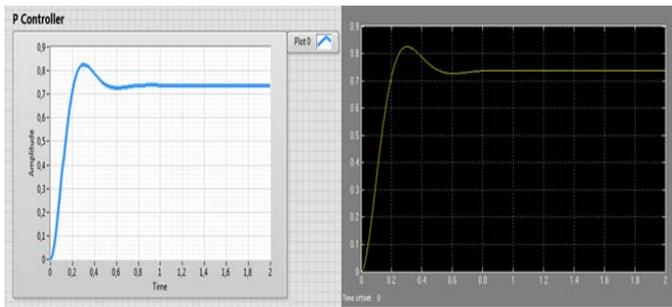


Fig. 4. Simulations Results of LabVIEW and MATLAB/Simulink

TABLE II. BENCHMARK RESULTS OF THE SYSTEM WITH P-CONTROLLER

Sim Time (s)	Step Size (s)	Amount of loops	MATLAB/Simulink (µs avg. of 1000 simulations)	LabVIEW (µs avg. of 1000 simulations)
2	0.001	2000	3940	11600
2	0.01	200	932	1206
2	0.1	20	419	160
60	0.001	60000	91262	344453
60	0.01	6000	9781	34777
60	0.1	600	1561	3616

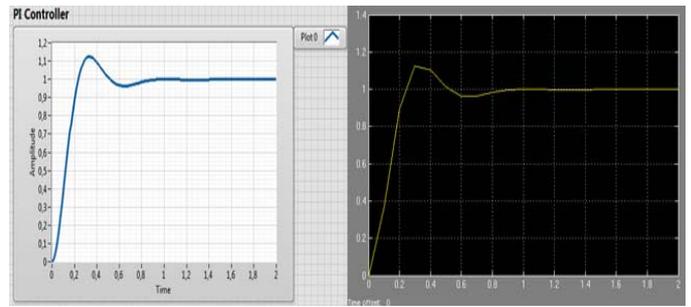


Fig. 6. Simulations Results of LabVIEW and MATLAB/Simulink

TABLE III. BENCHMARK RESULTS OF THE SYSTEM WITH PI-CONTROLLER

Sim Time (s)	Step Size (s)	Amount of loops	MATLAB/Simulink (µs avg. of 1000 simulations)	LabVIEW (µs avg. of 1000 simulations)
2	0.001	2000	5011	33303
2	0.01	200	1106	3322
2	0.1	20	694	379
60	0.001	60000	119900	984330
60	0.01	6000	12849	99261
60	0.1	600	1859	9895

B. PI-controller

PI-controller is a special type of PID-controller. In PI-controllers, derivative of error signal is not used therefore calculations of PI-controller are generally easier than PID controller. Transfer function of a PI controller is $K_p + K_i/s$. For the benchmark application, $K_p=11.5$ and $K_i=37$ are chosen.

System is simulated using both platforms, MATLAB/Simulink and LabVIEW. Models and results can be seen in Fig 5 and Fig 6.

System is again simulated 1000 times for several different step sizes and for each of two different simulation times. Averages of computation times for both platforms are shown in Table 3.

C. PID-controller

PID-controller is a feedback mechanism generally used in industrial applications. Three different terms are involved in PID-controller, namely, proportional, integral and derivative. Transfer function of a PID-controller is $K_p + K_i/s + K_d s$. For the benchmark application, $K_p = 11.5$, $K_i = 37$ and $K_d = 0.05$ are chosen.

In the same way, system is simulated 1000 times for several step sizes and for each of two different simulation times using both platforms, MATLAB/Simulink and LabVIEW. Models and results can be seen in Fig 7 and Fig 8. Averages of computation times for both platforms are shown in Table 4.

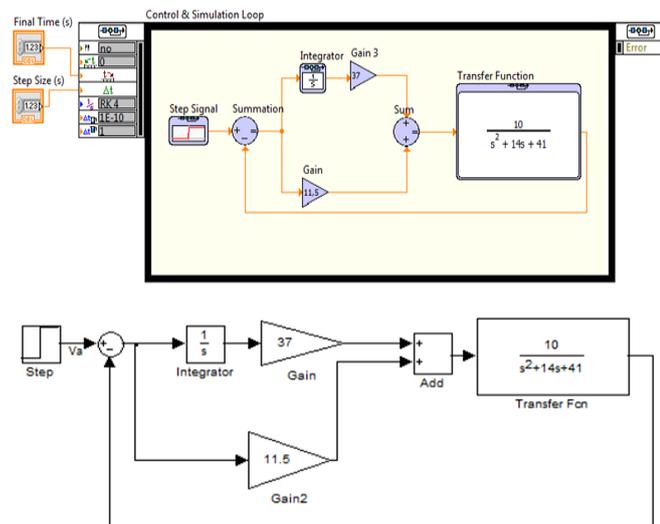


Fig. 5. PI Controller Design of LabVIEW and MATLAB/Simulink

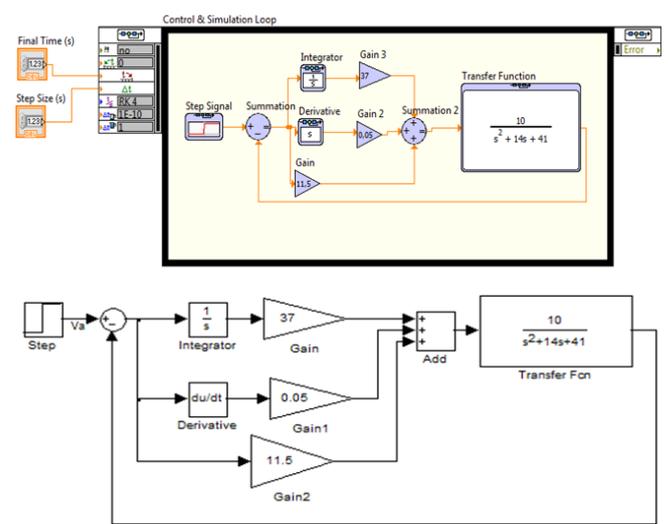


Fig. 7. PID Controller Design of LabVIEW and MATLAB/Simulink

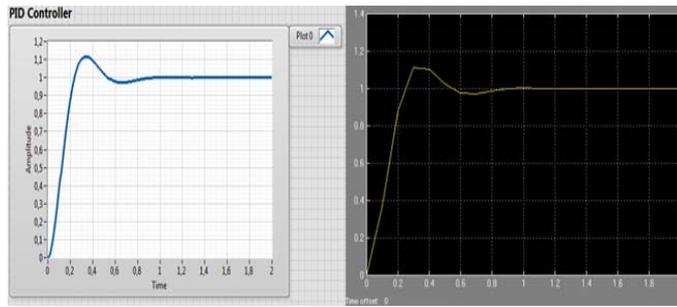


Fig. 8. Simulations Results of LabVIEW and MATLAB/Simulink

TABLE IV. BENCHMARK RESULTS OF THE SYSTEM WITH PID-CONTROLLER

Sim Time (s)	Step Size (s)	Amount of loops	MATLAB/Simulink (μ s avg. of 1000 simulations)	LabVIEW (μ s avg. of 1000 simulations)
2	0.001	2000	5692	38885
2	0.01	200	1167	4106
2	0.1	20	713	449
60	0.001	60000	140620	1176089
60	0.01	6000	14503	116675
60	0.1	600	2058	11986

V. RESULTS

First of all, effect of controller type on simulation time is inspected. As expected, it is observed that the time wasted for simulation increases with increasing complexity of controller. Although it increases in both platforms, rate of increment in simulation time is higher in LabVIEW as it can be seen in Table 5. However, it is observed that LabVIEW does not spend too much time for derivation process.

Secondly, effect of amount of loops in one simulation is examined. To compare simulation time performances, time per loop is calculated by dividing total simulation time to amount of loops for all controllers and results are shown in Fig 9. It is obvious that, for very small number of loops, LabVIEW's performance of simulation time is better than MATLAB/Simulink's. However, when the number of loops increases, MATLAB/Simulink provides better simulation time performance.

VI. DISCUSSIONS AND CONCLUSIONS

While developing an application, choosing which platform to use depends on several factors namely; price of platform, reachability, customer support, user-friendliness, performance etc. However; when the application is time-critic, performance of platform becomes more important in this decision. For this purpose, simulation time is analyzed for control simulations using MATLAB/Simulink and LabVIEW.

TABLE V. EFFECT OF CONTROLLER TYPE

Sim Time = 2s Step Size = 0.001	MATLAB/Simulink	LabVIEW
P \rightarrow PI Controller	5011 / 3940 = 1.272	33303 / 11600 = 2.871
PI \rightarrow PID Controller	5692 / 5011 = 1.136	38885 / 33303 = 1.168
P \rightarrow PID Controller	5692 / 3940 = 1.445	38885 / 11600 = 3.352

Three different controllers are developed in both platforms. After that, results of simulations are gathered. Results are examined with two different approaches. One of them is the effect of complexity of controller and the other one is effect of simulation time per loop on platforms' performance.

After making simulations of P, PI and PID controllers, as expected that increasing complexity of controller increases simulation time. However, MATLAB/Simulink shows better performance than LabVIEW. Considering simulation time per loop; in spite of the fact that LabVIEW is faster than MATLAB/Simulink for very small number of loops, MATLAB/Simulink shows better performance with increasing number of loops.

There may be some other parameters which affect the performance of platform but by using the same hardware for all the benchmarks, effects of hardware are kept negligible. As a result of this study, it is clear that when simulation time is the main consideration of the application to be developed MATLAB/Simulink is one step in front of LabVIEW.

Acknowledgment

The authors want to thank to ROKETSAN Missiles Company for their support on this study.

References

- [1] T. TAŠNER, D. LOVREC, F. TAŠNER and J. EDLER, "Comparison Of LabVIEW and MATLAB for scientific research", Annals of Faculty Engineering Hunedoara - International Journal of Engineering, pp. 389-394, 2012.
- [2] National Instruments, Getting Started with LabVIEW, 2007.
- [3] MathWorks, "Getting Started Guide", MathWorks Inc., 2014.
- [4] C. Kasnakoglu, "ELE515 – MATLAB ile Bilgisayarlı Kontrol, 2013–2014/II," [Online]. Available: <http://kasnakoglu.files.wordpress.com> [Retrieved: 15 11 2014].
- [5] K. Ogata, Modern Control Engineering, Prentice Hall, 2009.
- [6] A. NAGANI, M. NASIRUDDIN and T. GOSKULA, "Simulation of Control System with LabVIEW" IOSR Journal of Electrical and Electronics Engineering, pp. 30-35, 2014.

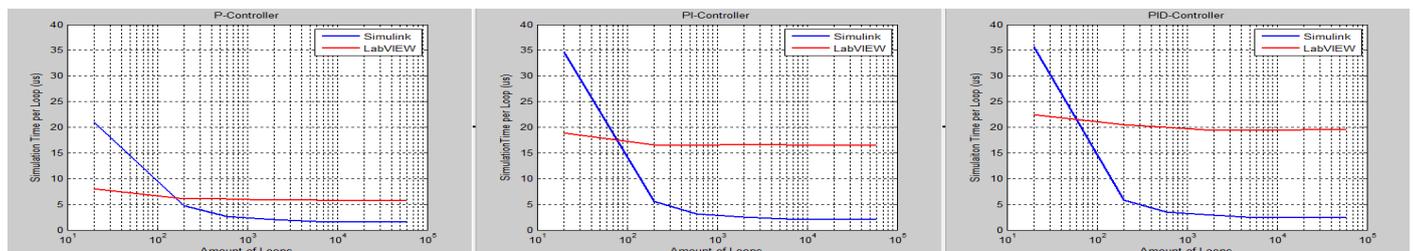


Fig. 9. Graphs of Amount of Loops vs Simulation Time per Loop