

Comparative Study Regarding the Optimization and Analysis of Automatic Systems using PID, LQR and PI-MIMO

PAUL CIPRIAN PATIC, LUMINITA DUTA

Automatics, Informatics and Electrical Engineering Department

Valahia University of Targoviste

18-24 Unirii Boulevard, 130056, Targoviste

ROMANIA

patic@valahia.ro, luminid@yahoo.com

Abstract: - Automatic modeling system using Matlab-Simulink software package applies the Cohen-Coon method for determining the closed loop PID parameters and plotting the system response curve. Implementing the same system in closed loop PID adjustment using LabVIEW software package, requires, at first, an additional procedure, namely, the use of transfer functions in Z domain, that means the use of sampling signal. The study from this paper intends to determine the parameters of a PID regulator using two software packages - MATLAB-Simulink and LabVIEW. As a mathematical pattern a DC machine has been used. To exemplify regulators PID, LQR (Linear Quadratic Regulator) and PI-MIMO (PI – Multiple Input Multiple Output) were used. The application is easy to be implemented, the inverted pendulum being used to emphasize the results. The closed loop system and its evolution over time are easily implemented. This software package offers an easy user interface, which means that is convenient to introduce the program functions.

Key-Words: - Inverted pendulum, PID regulator, LQR regulator, PI-MIMO regulator.

1 Introduction

MATLAB is a tool with high performance in technical estimations. It integrates the calculation, visualization and programming in an easy to use environment where problems and solutions are expressed in familiar mathematical notation. Using MATLAB software package has the following main purposes: mathematical calculation, the development of algorithms, data acquisition, modeling, simulation and prototype development, visualization, exploration and data analysis, scientific graphics and engineering, application development graphical user interface. MATLAB is an interactive system whose basic element to define data is the string that do not require sizing [1], [3].

With LabVIEW virtual instruments are built (VI), having the appearance of instruments or physical systems. Virtual instruments have an interactive user-interface front panel - and one part for the programmer - block diagram. For identification and use in other applications, each virtual instrument is an icon with a specific entries and exits.

Automatic systems modeling using Matlab-Simulink software package applied the Cohen-Coon method for determining the closed loop PID parameters and plotting the system response curve. The limit of the over adjustment was found to be 20%, the timing of the transitional regime - 3

seconds, and maximum response time is 10 seconds. Also, one represented the chart of the evolution over time, in closed loop system, to unit step input.

This software package offers an easy user interface, which means that is convenient to introduce the program functions. The application has implemented the conversion done from a field of definition of transfer functions to another (ex. obviously $c2d$ function transforming the transfer function of s field in Z domain, using a sampling period by the user), and includes Simulink tool that greatly facilitates user's work as the user wants to consider in detail the operation of each block, and many other features in the field of automation and theory systems.

2 Example using PID Regulator

At first, the problem of control design it is shaped by the transfer application as having a single input and a single output (Single Input Single Output, SISO) of third order, where $a_1 = 43$ and $a_1=3$ with an approximation of ± 0.8 and a non-linear saturation ± 2 .

In addition, due of design tolerances, the current dynamics presents suggestive variations from the base model. Specifically, the denominator coefficient a_2 varies between 40 and 50 and

coefficient a_1 between half and 1.5 times the nominal value of 3 [4] and [15].

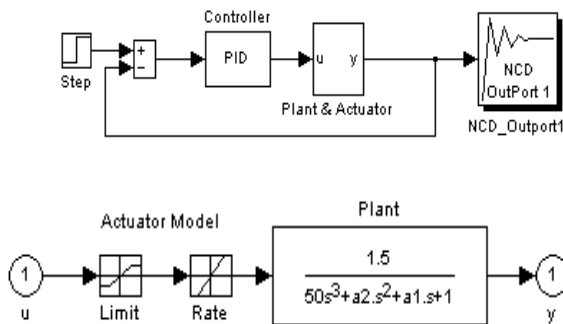


Fig. 1 NCD block attached to the result of the plant

$$G(s) = \frac{1.5}{50s^3 + a_2s^2 + a_1s + 1} \quad (1)$$

A PID regulator is designed so that closed-cycle system to meet the following specifications for tracking:

- Maximum Oscillation 20%;
- Up to 10 seconds for the propagation;
- Maximum response time of 30 seconds [4], [15].

It results that the answer of the closed cycle be resistant to uncertainty in installation's dynamics.

The SIMULINK *ncddemo1* system contains the application and the control structure. It is noted that non-linearity and saturation speed are included in the model plant. An input step drives the system. NCD block is attached to the outlet installation (Figure 1) because is the signal which will be restricted. Checking System's Parameters dialog box is noted that each simulation lasts 100 seconds.

Tunable and uncertain variables are initialized. The uncertain variables a_1 and a_2 are initialized at nominal values at 40 and 3, respectively. Tunable parameters K_p , K_i and K_d are initialized at 0.6323, 0.0493, respectively at 2.0272. These values result from the use of Ziegler-Nichols method for PID regulators.

Ziegler-Nichols method for PID regulators grant can be summarized as:

- The entire amplification and derivative is established to zero and the amplify is increased proportionally, until the system becomes unstable;
- This amplification is defined as K_u ; the measure of the oscillation period is P_u ;
- The coefficients are established one by one

$$K_p = \frac{3 * K_u}{5}, K_i = \frac{6 * K_u}{5 * P_u}, \text{ si } K_d = \frac{3 * K_u * P_u}{50} \quad (2)$$

Then, the limitations of time are defined. Upper and lower restriction limits define oscillation, propagation time and response time [4], [15].

Note that the uncertainty in the parameters a_1 and a_2 is defined and that the restrictions are applied to the installation only nominal. After running optimization, the time, the cost function evolution and the final values for tunable parameters vary depending on computer's performance. Optimization should provide a regulator to satisfy any restrictions [22], [23], [24] and [25].

The lower and superior limits are restricted and the optimizing process is started with uncertainty. One can consider that these restrictions can not be satisfied, but the result shows a maximum violation of the restriction under the value of 0.01.

One can experiment by moving the limits of restriction to achieve better performances of the system. For example, one can reduce the time spread or may decrease the restrictions of oscillation (Figure 2.)

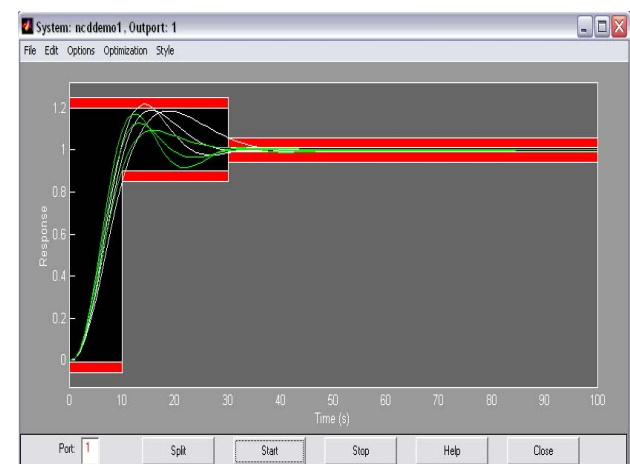
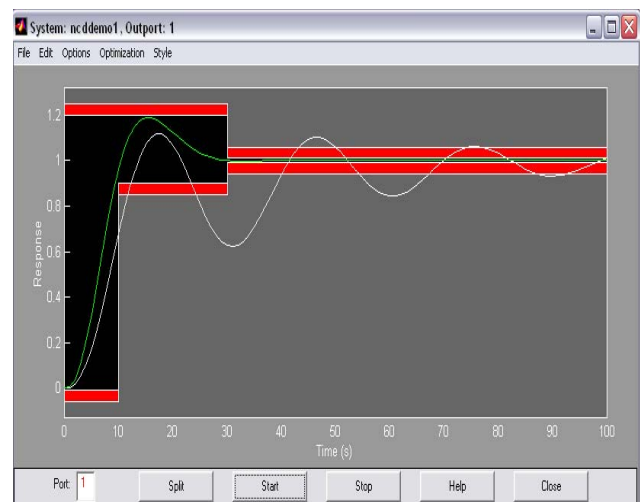


Fig. 2 Moving the limits of restriction to achieve better performance of the system

3 Example using LQR Regulator (linear quadratic rules) with Feed-forward Controller

The second problem requires Control System Toolbox since it is an extended version of the problem from SIMULINK demo *lqgdemos* file.

The SISO application can be modeled as a linear ranking 4 system with an enlarging saturation by ± 5 and a nonlinear limit of error of by ± 10 .

The equations are [4], [15], [22], [23], [24] and [25]:

$$\dot{x} = \begin{bmatrix} -1.0285 & 0.9853 & -0.9413 & 0.0927 \\ -1.2983 & -1.0957 & 2.8689 & 4.7950 \\ 0.1871 & -3.8184 & -2.0788 & -0.9781 \\ 0.4069 & -4.1636 & 2.5407 & -1.4236 \end{bmatrix} x + \begin{bmatrix} 0 \\ 6.6389 \\ 0 \\ 0 \end{bmatrix} u = Ax + Bu$$

$$y = [-1.7786 \quad 1.1390 \quad -1.029] x$$

(3)

They define the nominal plant model. One allows the installation's Matrix to vary between half to two times of its nominal value.

Using LQG / LTR techniques, one can design a Kalman state estimator and a k amplifier regulator for the linear system. Add an integrator to ensure a zero steady error.

To achieve an increased time response one adds a feed-forward amplifier (FF).

In the demo SIMULINK *lqgopt* system, the control parameters k and FF are granted by "the method of the least squares".

These parameters can be accorded using the NCD Blockset.

In particular the control parameters K and FF are granted, so that the closed-cycle system to meet the following specifications:

- Maximum oscillation - 20%;
- Propagation time - one second;
- Response time - three seconds [4], [15], [22], [23], [24] and [25].

SIMULINK system contains the application and the control structure below.

After starting the system, it is noted that the non-linearity error (± 10) and the saturation (± 5) are included in the model's installation.

Using the *From Workspace* block, one introduces a step that goes from zero to one in one second. NCD block is attached to the result of the installation because there is a restricted signal. Checking the *System's Parameters* one observes that each simulation lasts, always, 10 seconds.

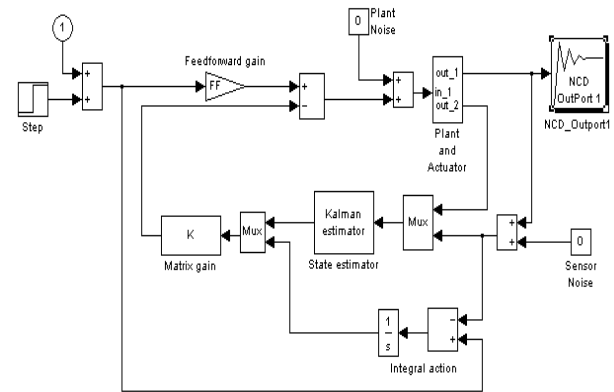


Fig. 3 NCD block attached to the result of the plant

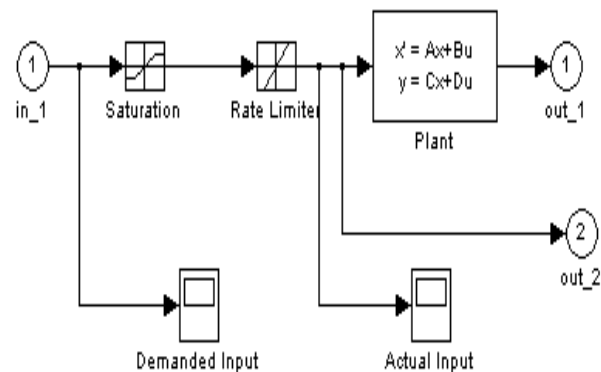


Fig. 4 Plant & Actuator

Tunable and uncertain variables are initialized. Domain restrictions for this demonstration are defined. Upper and lower restriction limits of the oscillation, propagation time and response time are defined. As described above, an initial design regulator using the LQG / LTR methods is designed starting from a linear application. For non-linear control optimization the feed-forward amplifier FF and regulatory matrix amplifier k are tunable [4], [15], [22], [23], [24] and [25].

Note how the installation (reaction) matrix A is defined and also that the optimization restrictions are applied only on nominal installation (reaction). Now starts the optimization.

We monitor the response evolution in time. The optimization time, the cost function evolution and final values for tunable parameters vary depending on computer.

Optimization should produce a regulator which satisfies any restrictions.

Now, we restrict the upper and lower limits. One starts the optimization with uncertainty. Considering that these restrictions can be satisfied, the result shows a maximum violation of the restriction under 0.01 (1%).

One can experiment, by moving the limits of restriction to achieve better performances of the system; for example, reducing the propagation time or using less restrictions of oscillation (Figure 5).

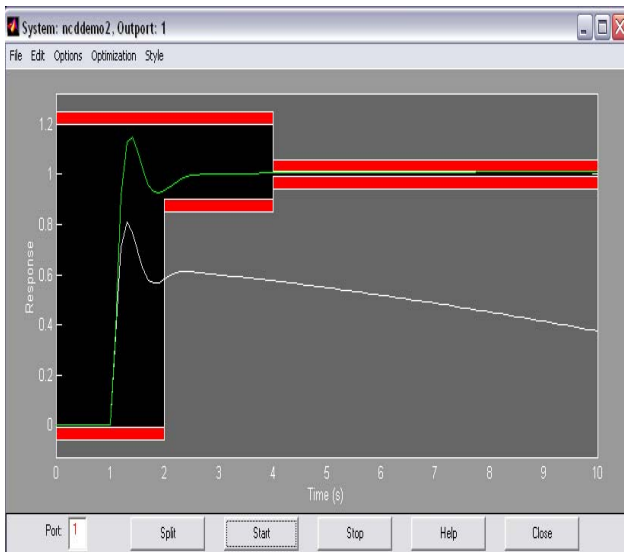


Fig. 5 Moving the limits of restriction to achieve better performances of the system

3.1 Results and Simulation after Using LQR

Simulation using the optimal LQR has been implemented. Figure 6 shows the unit step system response. As shown from the figure the displacement reaches its final value in 3×10^{-6} sec., gives zero tracking error and the system has better stability. Comparing, that, with the results have been obtained previously. It is quit clear using LQR has much better response [9] and [17].

Figure 7 and figure 8 show unit ramps and unit acceleration system response. As shown from the figure the displacement reaches its final value in few microseconds, and gives zero tracking error, which means much better sensitivity and accuracy for the micro accelerometer compared to previous approaches.

The speed of reaching the final values depends on choosing the values of matrix Q , as choosing high values of Q as having faster response for any input signal and having better stability.

Finally, for all inputs, the mass speed when using the LQR is faster and almost has no oscillations, for example, for unit step input the average speed is 1.824×10^{-3} m/sec, compared to 1.7111×10^{-4} m/sec after some authors. That makes, the mass reaches its final position faster with no oscillations.

Using LQR has increased and improved the accelerometer bandwidth, stability, accuracy and response time [7] and [9].

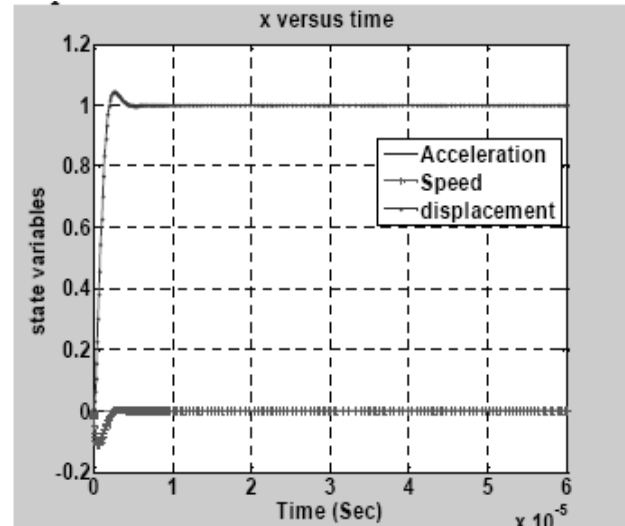


Fig.6 State variables versus time for unit step input tracking

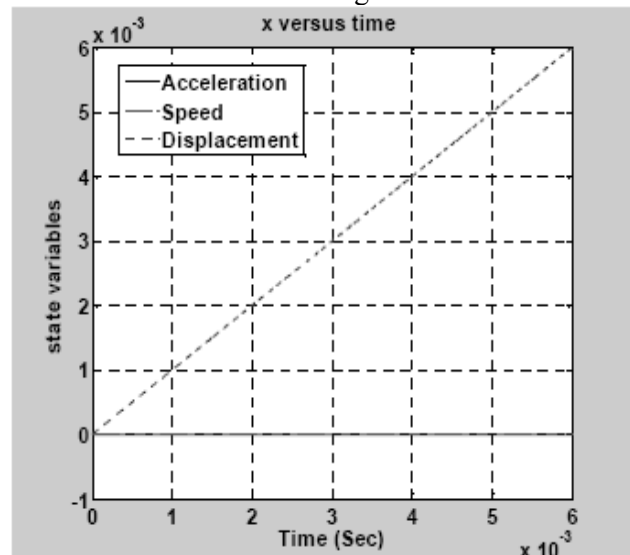


Fig.7 State variables versus time for unit ramp input tracking

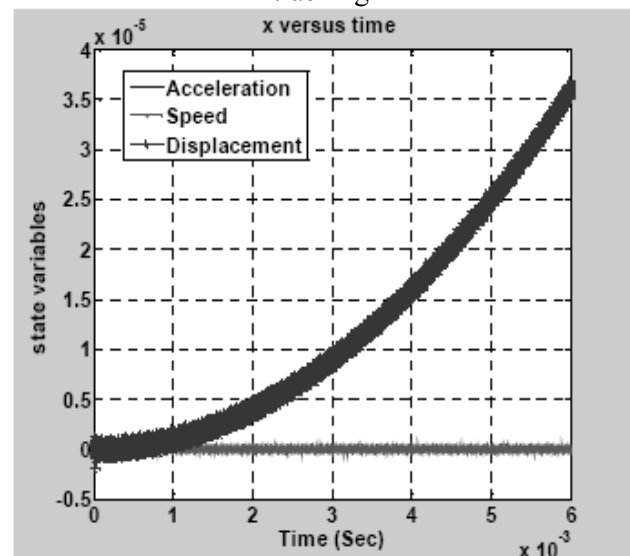


Fig.8 State variables versus time for unit acceleration input tracking

4 Example using PI-MIMO Regulator (Multiple Input Multiple Output)

The third problem of control design involves designing a centralized PI-MIMO regulator for a turbine engine of LVI00 fuel. One models the application as a system with two inputs, two outputs and a minimum of five phase states. Inputs are fuel flow and variable area turbine nozzle. The outputs are speed coil generator fuel and temperature. The five stages are speed spool generator, the output power, temperature, driving level of the fuel flow and the driving level of the turbine nozzle. A system model is given by the next equation [4], [15], [22], [23], [24] and [25]:

$$\dot{x} = \begin{bmatrix} \dot{N}_g \\ \dot{N}_p \\ \dot{T}_6 \\ \dot{W}_f \\ \dot{V}_{ATN} \end{bmatrix} = \begin{bmatrix} -14122 & -00522 & 0 & 42536 & 63087 \\ 00927 & -01133 & 0 & 42204 & -07581 \\ -78467 & -02555 & -33333 & 0 & 167-44894 \\ 0 & 0 & 0 & -25000 & 0 \\ 0 & 0 & 0 & 0 & -33333 \end{bmatrix} \begin{bmatrix} N_g \\ N_p \\ T_6 \\ W_f \\ V_{ATN} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \end{bmatrix}$$

$$y = \begin{bmatrix} N_g \\ T_6 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} x \quad (4)$$

To compare with previous results, none nonlinearities is not modeled in this problem. Saturation nonlinearities exist in the system as limited efforts and maximum temperatures.

These nonlinearities can be included in the problem formulation as in previous examples.

Also, for the demonstration plant will exaggerate uncertainty of installation.

More precisely, one permits a variation of a matrix A between half and twice of its nominal value.

We want to design a centralized controller 2-by-2 PI for the application so that the system closed cycle track to meet the following specifications:

- Maximum propagation delay time of one second;
- 0 oscillation in the first channel and less than 10% in one second;
- less than 5% across the channel coupling.

SIMULINK system contains software and control structure below (Figure 9).

To open the system can be written in MATLAB prompt or double-click on the NCD block 3 Demo SIMULINK system.

One model the PI regulator as a state-space system with the value 0 for matrix A and the identical matrix B.

The C and D matrix are variable tunable K, and Kp, for a total of eight tunable variables.

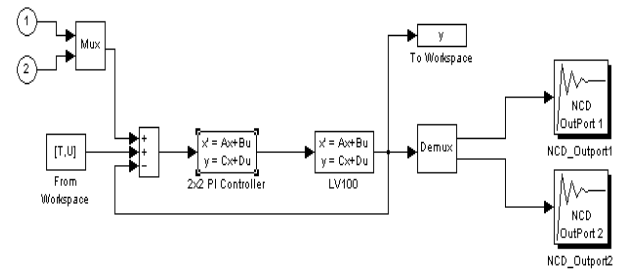


Figure 9 Control structure of the system

$$T = \begin{bmatrix} 0 \\ 0,001 \\ 5 \\ 5,001 \\ 10 \\ 10,001 \\ 15 \end{bmatrix} \quad U = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} \quad (5)$$

Double click on the *ncd3init* block is executed to load data installation, the signal inputs, the initial values of tunable parameters and previous values of regulators obtained by other methods. With double click on the *ncd3init* block one defines the restrictions of time domain responses of this demonstration. Note that there are two optimization NCD blocks that can be displayed simultaneously.

The approach suggested for MIMO regulator design requires the sequencing of input steps. When the first channel entering, the first result should follow up the step and other channels must reject signal. When the second channel entering, the second signal must follow up the step and the other channels must eject the signal etc. [4], [15], [22], [23], [24] and [25].

Note that we used the From Workspace block to enter the sequential steps inside the system. Double-click on NCD blocks to open the restriction NCD images and to display all the other restrictions. It is noted that restrictions for the first result define the limits step response as shown above. Meanwhile, the output restrictions limits make that the signal stay near to ± 0.05 value.

Before starting the optimization, the Optimization Parameters dialog box opens by selecting Parameters ... from Optimization menu and it noticed that how is defined the Parameters Optimization. It is opened, also, the Uncertain Variables and one observe how is defined the A matrix installation and also notes that optimization restrictions are only nominal applied on installation.

One press Start or one can hold down the acceleration key and t to start optimizing process. It follows the time response evolution during the optimizing process. Optimization time, cost function evolution and final values for tunable parameters vary depending on the computer. Optimization should produce a regulator that satisfies all restrictions [4], [15], [22], [23], [24] and [25].

Now, one return to Uncertainty Variables dialog box and one restrict the lower and upper limits. Press Start to begin optimizing with uncertainty. We can consider that these restrictions can not be satisfied, since the result appears to show a maximum restraint violation less than 0.01.

We can experiment by moving the restriction limits to achieve a higher system performance. For example, one reduces the propagation time or less oscillation restrictions (Figure 10).

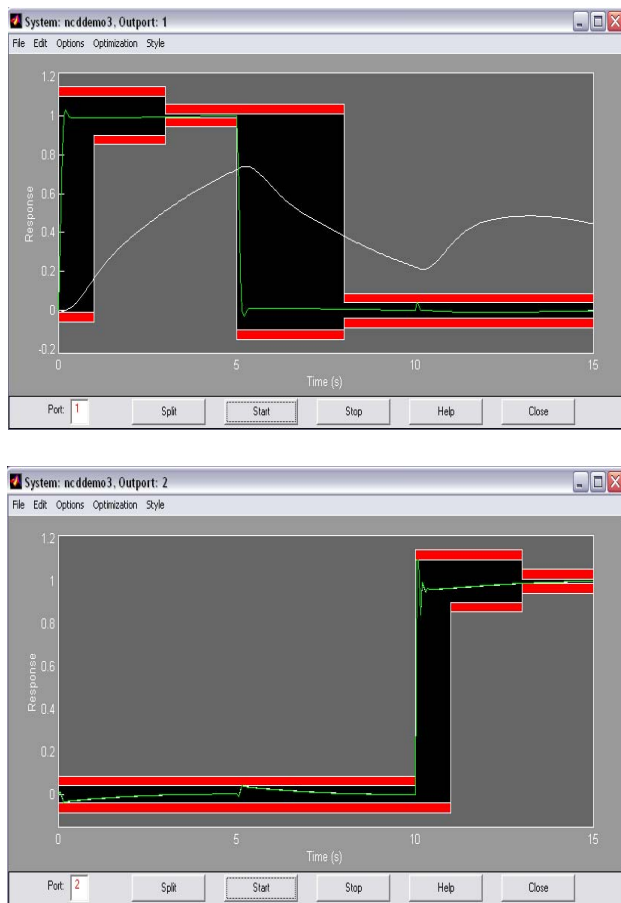


Fig. 10 Moving the limits of restriction to achieve better performances of the system

5 Application – Inverted Pendulum

Inverted pendulum can be described as a cylindrical metal rod attached to a device controlled by an engine powered to revolve only around an axis. The

device follows a linear track to create a stabilized problem (Figure 11).

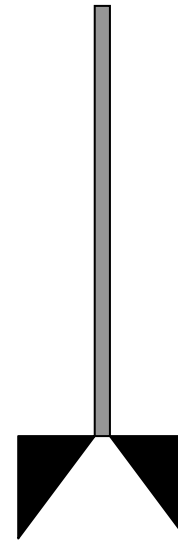


Fig. 11 Inverted pendulum

The inverted pendulum is a classic problem in dynamics and control theory and is widely used as a benchmark for testing control algorithms (PID controllers, neural networks, fuzzy control, genetic algorithms, etc). Variations on this problem include multiple links, allowing the motion of the cart to be commanded while maintaining the pendulum, and balancing the cart-pendulum system on a see-saw. The inverted pendulum is related to rocket or missile guidance, where thrust is actuated at the bottom of a tall vehicle. The understanding of a similar problem is built in the technology of Segway, a self-balancing transportation device. The largest implemented use is on huge lifting cranes on shipyards. When moving the shipping containers back and forth, the cranes move the box accordingly so that it never swings or sways. It always stays perfectly positioned under the operator even when moving or stopping quickly.

Another way that an inverted pendulum may be stabilized, without any feedback or control mechanism, is by oscillating the support rapidly up and down. If the oscillation is sufficiently strong (in terms of its acceleration and amplitude) then the inverted pendulum can recover from perturbations in a strikingly counterintuitive manner. If the driving point moves in simple harmonic motion, the pendulum's motion is described by the Mathieu equation.

In practice, the inverted pendulum is frequently made of an aluminum strip, mounted on a ball-bearing pivot; the oscillatory force is conveniently applied with a jigsaw.

The equation of motion is similar to that for a uninverted pendulum except that the sign of the angular position as measured from the vertical unstable equilibrium position:

$$\ddot{\theta} - \frac{g}{l} \sin \theta = 0 \quad (6)$$

When added to both sides, it will have the same sign as the angular acceleration term:

$$\ddot{\theta} = \frac{g}{l} \sin \theta \quad (7)$$

Thus, the inverted pendulum will accelerate away from the vertical unstable equilibrium in the direction initially displaced, and the acceleration is inversely proportional to the length. Tall pendulums fall more slowly than short ones.

The inverted pendulum with small parametric forcing is considered as an example of a wider class of parametrically forced Hamiltonian systems. The qualitative dynamics of the Poincare map corresponding to the central periodic solution is studied via an approximating integrable normal form. At bifurcation points we construct local universal models in the appropriate symmetry context, using equi-variant singularity theory. In this context, structural stability can be proved under generic conditions [26].

The upper equilibrium of a pendulum can be stabilized by a vertical oscillation of the suspension point within a specific range of excitation frequencies and amplitudes. This follows from classical perturbation theory applied to the linearized equation of motion, e.g., according with van der Pol and Strutt, Stoker and Hale.

The corresponding bifurcation is determined by the non-linear dynamics. Our aim is to understand this dynamics in a qualitative way, with special interest in persistence.

Here the symmetries of the system are first maintained. However, we consider a system that is slightly more general, but still in the 1 and 1/2 degree-of-freedom Hamiltonian setting [26].

We study the corresponding Poincare map, following the approach of Broer and Vegter.

Normal form theory yields a planar Hamiltonian vector field which gives an integrable approximation of this map, valid for every angular displacement and small velocity of the pendulum. The relation between the Poincare map and its approximation is briefly discussed in terms of perturbation theory [26] and [27].

At each bifurcation point of the approximating vector field a model is constructed that is locally equivalent to this approximation. by performing

small perturbations that respect the symmetries, and conjugating these perturbations to the model by symmetry-preserving local morphisms [26].

The study of the Inverted Pendulum could be a collection of MATLAB functions & scripts, and SIMULINK models, useful for analyzing Inverted Pendulum System and designing Control System for it.

The Inverted Pendulum is one of the most important classical problems of Control Engineering. Broom Balancing is a well known example of nonlinear, unstable control problem. This problem becomes further complicated when a flexible broom, in place of a rigid broom, is employed. Degree of complexity and difficulty in its control increases with its flexibility. This problem has been a research interest of control engineers. In this submission, however, we have analyzed Inverted Pendulum only with rigid broom [27].

The aim of the study is to stabilize the Inverted Pendulum such that the position of the carriage on the track is controlled quickly and accurately so that the pendulum is always erected in its inverted position during such movements.

The inverted pendulum (IP) is among the most difficult systems to control in the field of control engineering. Due to its importance in the field of control engineering, it has been a task of choice to be assigned to Control Engineering students to analyze its model and propose a linear compensator according to the PID control law [27].

For device, a stabilized runway requires the existence of the initial LQR stabilizer. The generation of this regulator rules is realized starting with writing the nonlinear equations which are defining the inverted pendulum. Ignoring the dynamics of the engine, the nonlinear equations of motion, for the inverted pendulum system, are [4], [15], [22], [23], [24] and [25]:

$$\ddot{\theta} = \frac{\frac{f}{m} + l\theta \sin \theta - g \sin \theta \cos \theta}{\frac{M}{m} + \sin^2 \theta} \quad (8)$$

$$\ddot{\theta} = \frac{-\frac{f}{m} \cos \theta + \frac{M+m}{m} g \sin \theta - l\theta^2 \sin \theta \cos \theta}{l \left(\frac{M}{m} + \sin^2 \theta \right)} \quad (9)$$

Where:

- f is the force applied to the cart by the engine in Newton (N);
- m is the position of the cart in meters;
- y is the vertical angle of the pendulum in radians;
- θ is the mass of the cart (0.455 kg);
- M is the pendulum mass (0.21 kg);
- l is the distance from the mass center of the pendulum (half of its length of 0.61 m);
- g is gravitational acceleration (m/s).

It is necessary that those equations be linear in the operating point $y=0$ and $\theta=0$ to obtain the linear system:

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{gm}{M} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{g(M+m)}{lM} & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 11 \\ M \\ 0 \\ -\frac{1}{lM} \end{bmatrix} f = Ax + bx \quad (10)$$

Using MATLAB command:

$$Klqr = lqr(A,b,diag([0.25 \ 0 \ 4 \ 0],0.003)$$

one obtains the gain stability:

$$K1qr = (-28.8675 \ -28.5632 \ -145.00 \ 14 \ -14.8601)$$

Besides the obvious non-linearity of the system's equations, the voltage limit applied to the engine gives a restriction of the action saturation of 1 N [4], [15], [22], [23], [24] and [25].

The sensors measure the position and angle pendulum device.

In addition, with the pendulum stability follows that the device has a commanded reference signal. Specifically, one has to design a controller for the system to meet the following specifications of a closed loop specifications:

- up to 4 sec. for propagation time;
- up to 6 sec for the response time;
- zero oscillation;
- less than 0.2 radians deviation from vertical.

It is noted that the saturation non-linearity included in the process model and hidden pendulum block contain the motion system with nonlinear equations.

One commands the input device position with a signal-type unit. NCD blocks are attached to the pendulum angle and device position signal (Figure

12 and Figure 13). It is noted that each simulation lasts 15 sec.

The control structure contents finite estimated status differences for the speed of the device and angular velocity of the pendulum.

As part of an internal control cycle, the estimations for the speed and angular velocity are multiplied by the amplification, collected and then introduced into the engine [4], [15], [22], [23], [24] and [25].

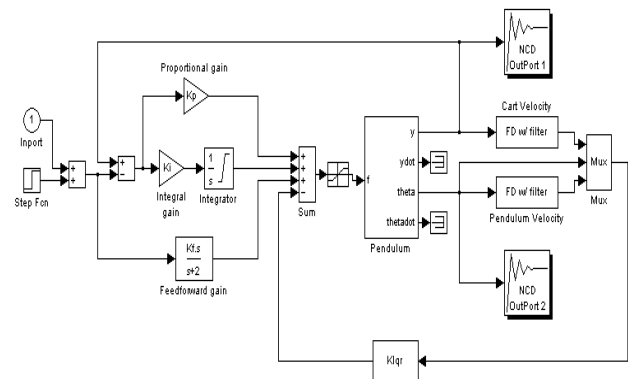


Fig. 12 NCD block attached to the result of the inverted pendulum

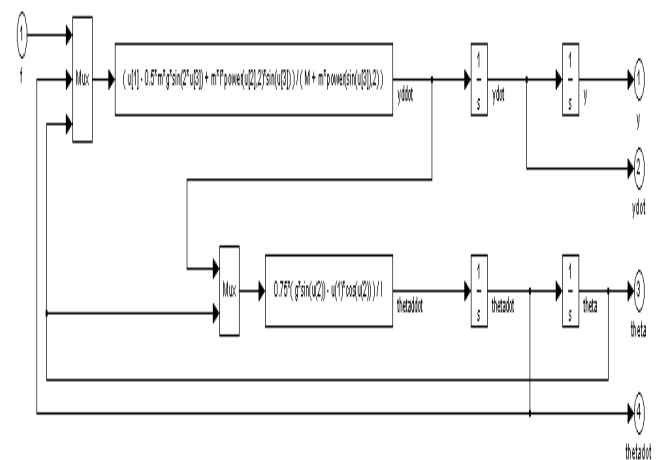


Fig. 13 The internal scheme of a pendulum block

The gain is initialized from 1 at 3 with: $Klqr = Clqr(2:4)$, where $Clqr$ is the solution from 1 to 4 at LQR. In a series of external control (used to allow the device to follow a commanded signal) a feed-forward amplifier Kf is initialized by $Kf=Clqr(1)$ and an integral amplifier, Ki is initialized as zero. It is noted that, in the absence of a commanded signal, these baseline controllers reduce the control structure at LQR amplification described in the previous section [1], [3], [4] and [15].

Tunable variables are initialized and, in accordance, restrictions of time domain response are defined. The configuration of these restrictions

should be known like a step response. Pendulum angle channel contains restrictions that define a disturbance rejection problem of perturbations. In other words, while device moves to the controlled position, the pendulum must remain in balance.

The optimization begins. The optimization time, the cost function evolution and final values for tunable parameters vary depending on computer. Optimization should produce a controller to meet any restrictions.

It is noted that this optimization runs slower than others. This is due to the estimating finite-state involve frequent updates during the simulation (Figure 14).

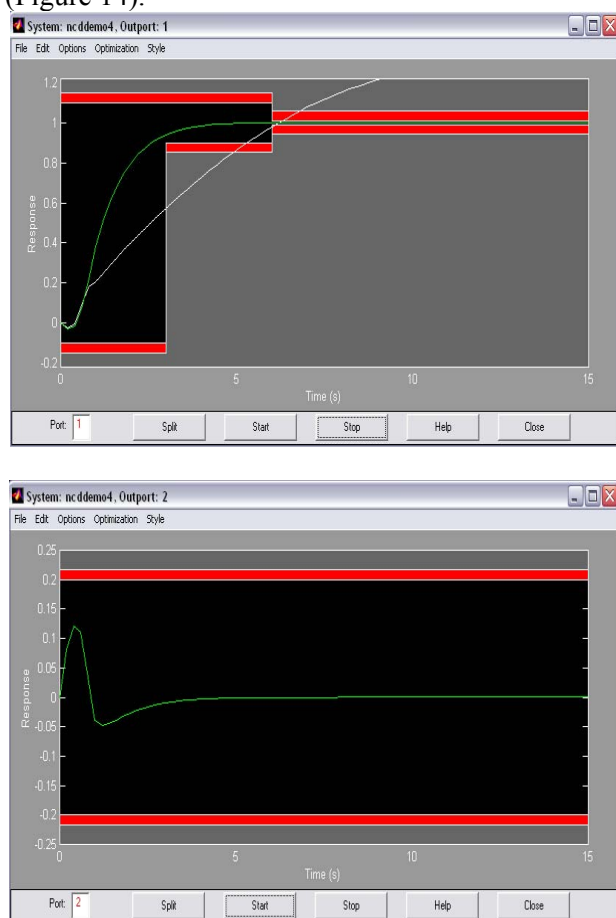


Fig. 14 Moving the limits of restriction to achieve better performance of the inverted pendulum

6 Conclusion

The work from this paper intended to determine the parameters of a PID controller using two software packages, namely MATLAB-Simulink and LabVIEW. As a mathematical model, a DC machine was used.

Automatic modeling systems using Matlab-Simulink software packages applied the Cohen-

Coon method for determining the closed loop PID parameters and plotted the system response curve. The limit of the over adjustment was found to be of 20%, the timing of the transitional regime - 3 seconds, and maximum response time was 10 seconds. Also, one represented the chart of the evolution over time, in closed loop system, to unit step input. The use of this software packages offers an easy way of work, which means that is convenient to utilize the program functions.

Implementing the same system in a closed loop PID adjustment using LabVIEW software package, one requires, at first, an additional procedure, namely, the use of the transfer functions in Z domain. For displayed example the sampling period is 60 μ s and thus can be determined the transfer function graph, sampled to the entry with a unitary step.

The use of the graphical features of LabVIEW software package for the adjustment of the PID controller, one recorded the best performance, since they include zero stationary error due to the integrative component for minimum stabilization time, due to anticipatory component derivative etc. Using the LabVIEW package to study the reaction systems involved additional procedures.

A comparative analysis of the two programs using Matlab-Simulink and LabVIEW to determine the same automatic performances of the system indicates that Matlab-Simulink software is more efficient having different features of its libraries ready implemented, with coverage areas of automation and systems theory. LabVIEW is a software package dedicated to particular virtual instrumentation and graphical applications in time.

References:

- [1] Andrei H, Brezeanu I, Enciu G, Enciu M – „*Elements of adjustment theory*”, Ed. Macarie, Târgoviște, 1999.
- [2] Călin S., Tertîşco M., Dumitrache I., Popeea C., Popescu D. – „*Optimisation in industrial automatisation*”, Ed. Tehnică, Bucharest, 1979.
- [3] Călin S, Brezeanu I, Popescu M., Ispas G. – „*Automatic tuning system theory. Matlab Application*”, Ed. Biblioteca, Târgoviște, 2002.
- [4] Franklin, Gene F., J. David Powell, and Abbas Emami-Naeini, *Feedback Control of Dynamic Systems*, Addison-Wesley Publishing Company, 1987.
- [5] Dumitrache I. – „*State-space method for linear systems automatic analysis, continuous and discrete*”, U.P.B. Litography, Bucharest, 1971.
- [6] Gibson J. – „*Linear automatic system, English translation*”, Ed. Tehnică, Bucharest, 1967.

- [7] Ibrahim H.E.A., *LQR Optimal Control for Micro-machined Tunneling Accelerometer*, Proceedings of the 9th WSEAS International Conference on Signal Processing, Robotics and Automation, ISPRA '10, February 20-22, Cambridge, U.K., p.40-45, 2010.
- [8] Ionescu VI. – „*Introduction in structural theory of a linear systems*”, Academy Edition., Bucharest., 1975.
- [9] Jalili-Kharaajoo Mahdi, Dehestani Alireza, *Active Queue Management: Comparison of Sliding Mode Controller and Linear Quadratic Regulator*, WSEAS Transactions on Circuits and Systems, Issue 3, Volume 2, July 2003, ISSN 1109-2734, p. 594-599.
- [10] Mahesh L. Chugani, Abhay R. Samant, Michael Cerna – *Labview Signal Processing*, Pearson, 1998.
- [11] Mejhed N.H., Hmamed A., Benzaouia A. *Continuous Time Regulator for Linear Systems with Constrained Control*, WSEAS Transactions on Systems Issue 5, Volume 4, May 2005, ISSN 1109-2777, p. 537-545.
- [12] Nițu C., Jora B., Popeea C., Georgescu E. – „*Automatic regulators. Laboratory guide*”, UPB, Bucharest, 1974.
- [13] Oppelt W. – „*Automatic tuning technique*”, Ed. Tehnică, Bucharest, 1965.
- [14] Papadache I. – „*Choosing and according of the regulators*”, Ed. Tehnică, Bucharest, 1975.
- [15] Potvin, A.F. *A Unified Solution to Constrained Configuration Control Law Design*, Master's Thesis, MIT EECS Dept., 1991.
- [16] Robert Bishop – *Labview 2009 Student Edition*, Prentice Hall, 2010.
- [17] Roshan Nader, *Application of Linear Quadratic Regulator (LQR) in Displacement Control of an Active Mass Damper*, 4th WSEAS International Conference on Systems Theory And Scientific Computation (ISTASC'04), December, 17-19 2004, Tenerife, Spain.
- [18] Savant C. J. – „*Automatic system calculus*”, Ed. Tehnică, Bucharest, 1967.
- [19] Sergiu Călin – „*Automatic regulators*”, Ed. Didactică și Pedagogică, Bucharest, 1970.
- [20] Sprânceană N., Dobrescu R., Borangiu Th. – „*Discrete automation in industry*”, Ed. Tehnică, Bucharest, 1978.
- [21] Tunsoiu Gh. – „*Electric drive systems and automation*”, Ed. Didactică și Pedagogică, Bucharest, 1968.
- [22] <http://zone.ni.com/devzone/conceptd.nsf/webmain/2F266590B802FEFD8625686500609588/AN039.pdf>
- [23] <http://zone.ni.com/dzhp/app/main>
- [24] http://www.mathworks.com/access/helpdesk_r13/help/toolbox/ncd/case.html
- [25] *Nonlinear Control Design Blockset, for use with Simulink – User's Guide* – The MathWorks, 2003.
- [26] <http://dissertations.ub.rug.nl/FILES/faculties/science/2001/m.van.noort/c2.pdf>.
- [27] <http://www.mathworks.com/matlabcentral/fileexchange/3790-inverted-pendulum>.