Control Theory (RMS-1-612-s ; RIMA-1-612-s)

Laboratory no. 4 - Control System Design for Inverted Pendulum on a Cart

(updated: 18th May 2018)

Issue date: 24th May 2021
Due date: 10th June 2021

Phong Ba Dao, dr hab. inż., prof. AGH
Department of Robotics and Mechatronics
AGH University of Science and Technology

Keywords: inverted pendulum, feedback control systems, linear time-invariant (LTI) systems, transfer functions, PID controllers, automatic PID tuning, state-space models, state feedback controller, pole placement, fuzzy logic control, modelling and simulation, Matlab/Simulink, Control Systems Toolbox, PID Tuner GUI, Fuzzy Logic Toolbox.

Recommended reading: Chapter 10 of [1] and documents provided in "Course materials > Fuzzy Logic Control".

[1] Roland S. Burns, *Advanced Control Engineering*, Butterworth-Heinemann, 2001, ISBN 978-0-7506-5100-4.
[2] Control Theory (RMS-1-612-s), Lecture 07. State-space methods for control system design.
[3] Control Theory (RMS-1-612-s), Lecture 10. Fuzzy logic control.
[4] Control Theory (RMS-1-612-s), Exercises: Part 1 – Modelling of Dynamic Systems and Linearization of Nonlinear Systems.
[5] Control Theory (RMS-1-612-s), Exercises: Part 2 – State-Space Methods for Control System Design.

Remarks:

(1) Short instructions and final results are provided to support you in solving problems. Based on the instructions provided you should elaborate and discuss in the report your solutions with Matlab codes, Simulink models and results obtained in detail.
(2) Before using Matlab commands (or functions) you should first check the syntax, descriptions, input arguments, output arguments, and illustrated examples by typing "help command".
(3) The report must be typed on a word-processor and submitted as a single PDF file via **UPeL platform**.

1. Introduction

Linearization of the nonlinear model of the inverted pendulum on a cart at the "upright" position (i.e. an equilibrium point) – performed in Exercises (Part 1. Modelling of Dynamic Systems and Linearization of Nonlinear Systems) – resulted in a linearized model in terms of linear state-space equations with the state matrices (A, B, C, D)

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 42.51 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -2.943 & 0 & 0 & 0 \end{bmatrix}; \; B = \begin{bmatrix} 0 \\ -3.333 \\ 0 \\ 1 \end{bmatrix}; \; C = \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}; \; D = \begin{bmatrix} D_1 \\ D_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{1}$$

Two transfer functions were derived from the state-space model:

$$G_1(s) = \frac{\theta(s)}{U(s)} = \frac{-3.333s^2}{s^4 - 42.51s^2} \quad \text{and} \quad G_2(s) = \frac{X(s)}{U(s)} = \frac{s^2 - 32.7}{s^4 - 42.51s^2} \tag{2}$$

where

$G_1(s)$ is the transfer function from the input external force $u(t)$ to the output rod angle $\theta(t)$;

$G_2(s)$ is the transfer function from the input external force $u(t)$ to the output cart position $x(t)$.

From Equations (1) and (2) it is easy to know that the linearized model of the inverted pendulum on a cart has four (open-loop) poles, which are two poles locating at the origin ($p_1 = p_2 = 0$), one negative pole ($p_3 = -6.52$) and one positive pole ($p_4 = 6.52$), and therefore the linearized system is clearly unstable.

The impulse and step responses were respectively plotted in Figures 1 and 2 to investigate the dynamic characteristics of the linearized model of the inverted pendulum on a cart in case the system is not controlled. Based on the behaviour observed in Figures 1 and 2, some comments could be given, i.e. because of the fact that the linearization was performed only for small displacements around the "upright" position (i.e. $\bar{x} = 0$ and $\bar{\theta} = 0$) with an assumption that the external force acting on the cart is equal to zero (i.e. $\bar{u} = 0$). Hence, on running the simulation model, after a certain (short) time (e.g. t = 1 [s]) under the impact of input signals (e.g. impulse or unit step signals) the inverted pendulum has no longer stayed at the "upright" position (or in other words, it departed from this equilibrium) and therefore the linearized model at the "upright" equilibrium position is not correct anymore. For this reason, a controller is highly needed to keep the rod staying "upright". In other words, we are going to design a stable closed-loop control system for the unstable inverted pendulum on a cart.

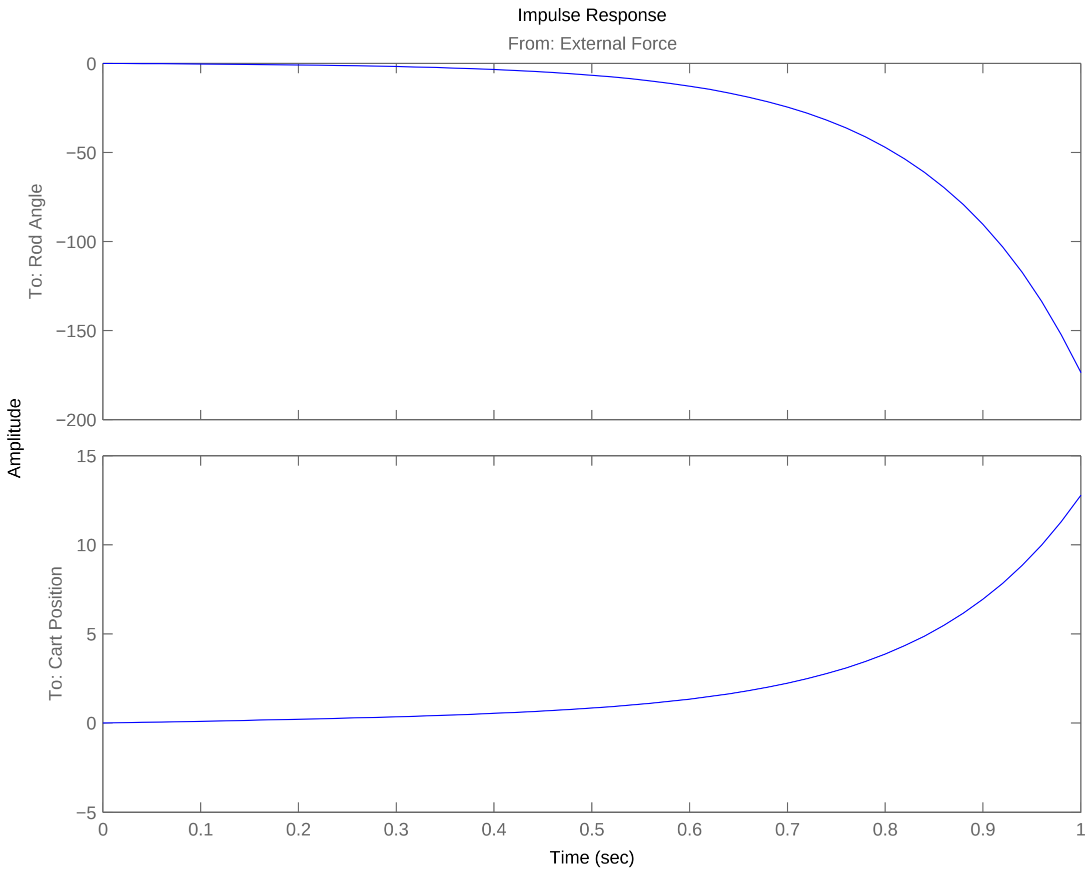

Figure 1. Impulse response of the linearized system in case without a controller.

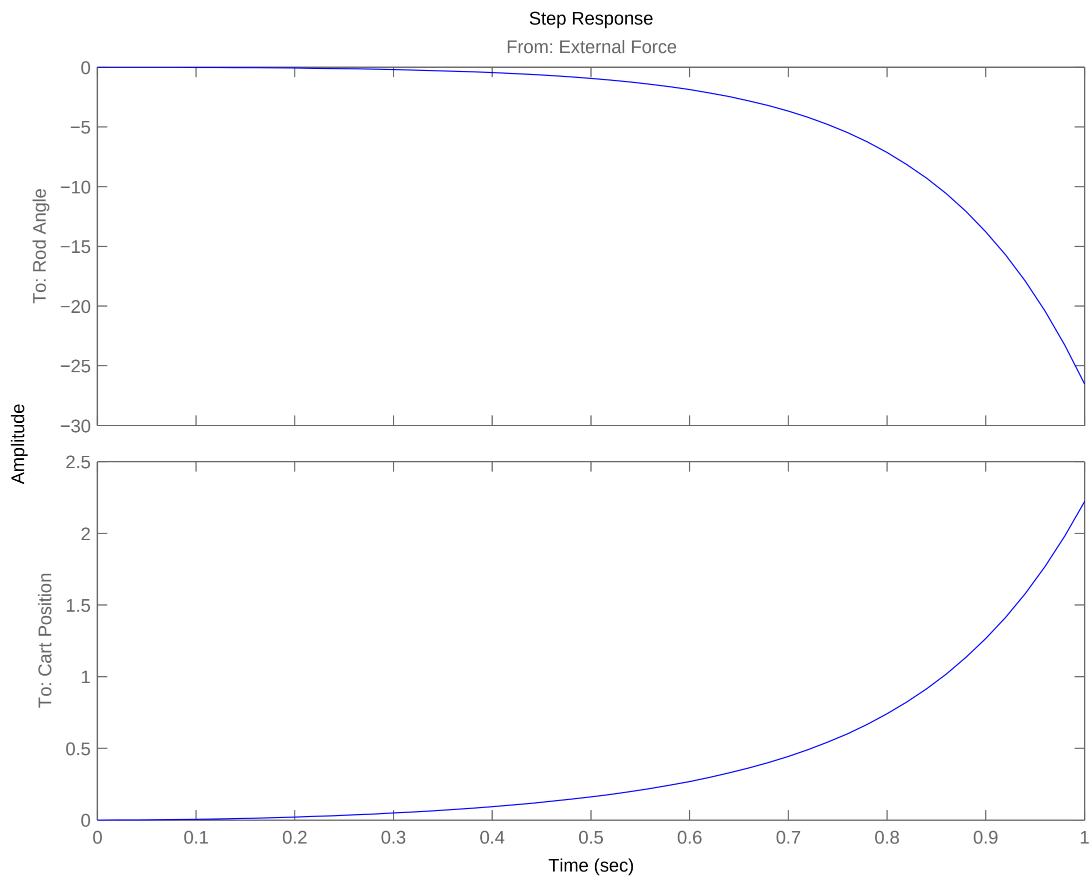

Figure 2. Step response of the linearized system in case without a controller.

2. Problems

Based on the linearized model of the inverted pendulum on a cart you are required to design closed-loop control systems (using PID controller, state feedback controller, and fuzzy logic controller) for the system so that the pendulum remains in the vertical position (or at the "upright" position) from an initial condition of the rod angle $\theta = 0.1$ radians. More specifically, you are required to solve the following four tasks:

1. Design a PID controller using the automatic PID tuning tool in Simulink (i.e. PID Tuner). It should be noted that this functionality requires Matlab version R2010b or later.
2. Assume that all the state variables (i.e. cart position, cart velocity, rod angle, rod angle velocity) can be measured by using sensors. Design a state feedback controller in Simulink using the state feedback gain $F = [30.044 \quad 4.759 \quad 0.997 \quad 2.346]$, which has been found in Task 2 (given by Equation (18) on page 7) of Exercises (Part 2. State-Space Methods for Control System Design).
3. Design a fuzzy logic controller using Matlab-Simulink and the Fuzzy Logic Toolbox.
4. Compare and discuss the results obtained by using these design methods.

The required performance specifications (with respect to the output rod angle $\theta(t)$) for the designed closed-loop control systems are:

- settling time under 2 seconds;
- overshoot less than 1 percent;
- zero steady-state error.

3. Instructions

To complete this assignment one should obtain the Simulink model, as shown in Figure 3 with similar results, as shown in Figure 12.

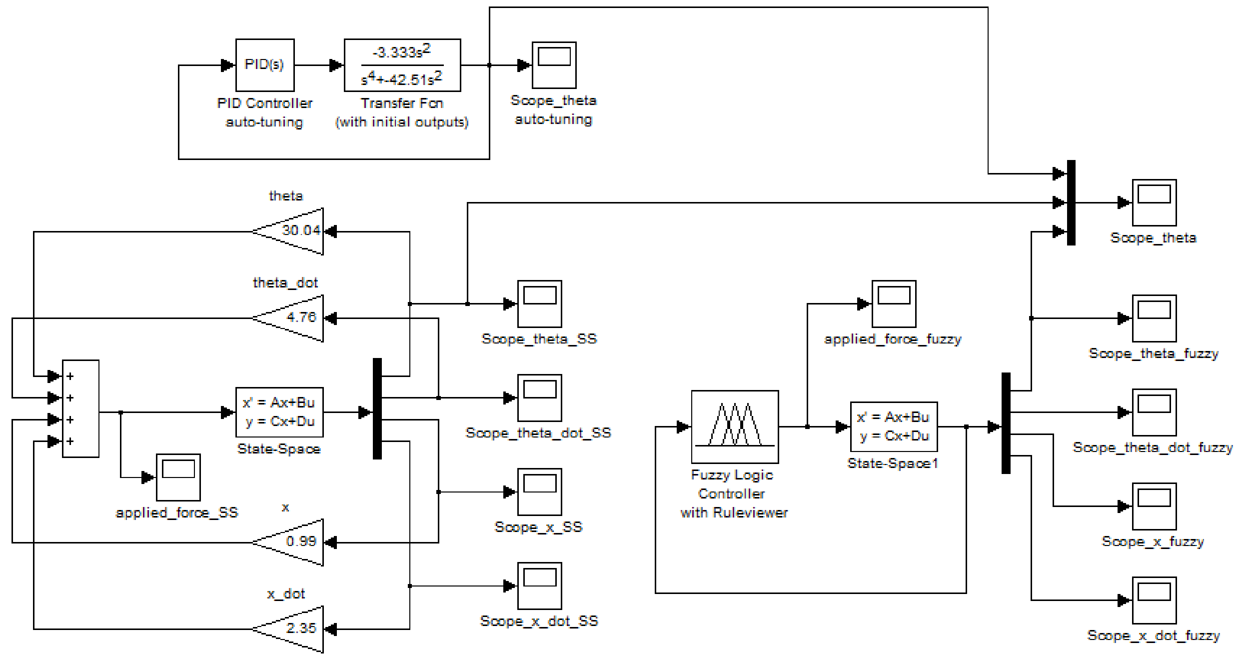

Figure 3. Simulink model of three different controllers.

As the purpose of the control system is to keep the rod staying "upright", the transfer function $G_1(s)$ from the input external force $u(t)$ to the output rod angle $\theta(t)$ will be used. The block "Transfer Fcn (with initial outputs)" should be used to build the transfer function $G_1(s)$ in Simulink, as shown in Figure 4.

Recall from Exercises (Part 1. Modelling of Dynamic Systems and Linearization of Nonlinear Systems) that the state variables have been selected as follows:

$$\begin{cases} x_1(t) = \theta(t) \\ x_2(t) = \dot{\theta}(t) \\ x_3(t) = x(t) \\ x_4(t) = \dot{x}(t) \end{cases} \Rightarrow \begin{cases} \dot{x}_1(t) = \dot{\theta}(t) = x_2(t) \\ \dot{x}_2(t) = \ddot{\theta}(t) \\ \dot{x}_3(t) = \dot{x}(t) = x_4(t) \\ \dot{x}_4(t) = \ddot{x}(t) \end{cases} \tag{3}$$

and the system has one input $u(t)$ and two outputs $y(t) = \begin{bmatrix} \theta(t) \\ x(t) \end{bmatrix} = \begin{bmatrix} x_1(t) \\ x_3(t) \end{bmatrix}$ (4)

It should be noted that in Equations (3) and (4) $x_1(t)$, $x_2(t)$, $x_3(t)$, $x_4(t)$ are state variables; whereas $x(t)$, $\dot{x}(t)$ and $\ddot{x}(t)$ are cart position, cart velocity and cart acceleration, respectively.

So when we create the linearized model of the inverted pendulum on a cart using the block "State-Space", four state variables should be "matched" with four elements of the state feedback gain $F = [30.044 \quad 4.759 \quad 0.997 \quad 2.346]$.

Since we have assumed that all the state variables can be measured by using sensors, matrix C has the form of [1 0 0 0; 0 1 0 0; 0 0 1 0; 0 0 0 1].

Figure 5 shows the design of state feedback controller.

PID controller auto-tuning tool with the controller parameters obtained are shown in Figure 6.

You are referred to some online tutorials about PID controller tuning and PID Tuner:
- Introduction to Model-Based PID Tuning in Simulink
https://www.mathworks.com/help/slcontrol/ug/introduction-to-automatic-pid-tuning.html
- Model-Based PID Controller Tuning
https://www.mathworks.com/help/slcontrol/automatic-pid-tuning.html
- PID Controller Tuning
https://www.mathworks.com/help/control/pid-controller-design.html
- PID Tuner
https://www.mathworks.com/help/control/ref/pidtuner-app.html
- Open PID Tuner
https://www.mathworks.com/help/slcontrol/ug/designing-controllers-with-the-pid-tuner.html
- PID Controller Tuning in Simulink
https://www.mathworks.com/help/slcontrol/gs/automated-tuning-of-simulink-pid-controller-block.html
- Tune PID Controller to Favor Reference Tracking or Disturbance Rejection
https://www.mathworks.com/help/slcontrol/ug/tune-pid-controller-to-balance-tracking-and-disturbance-rejection-performance.html

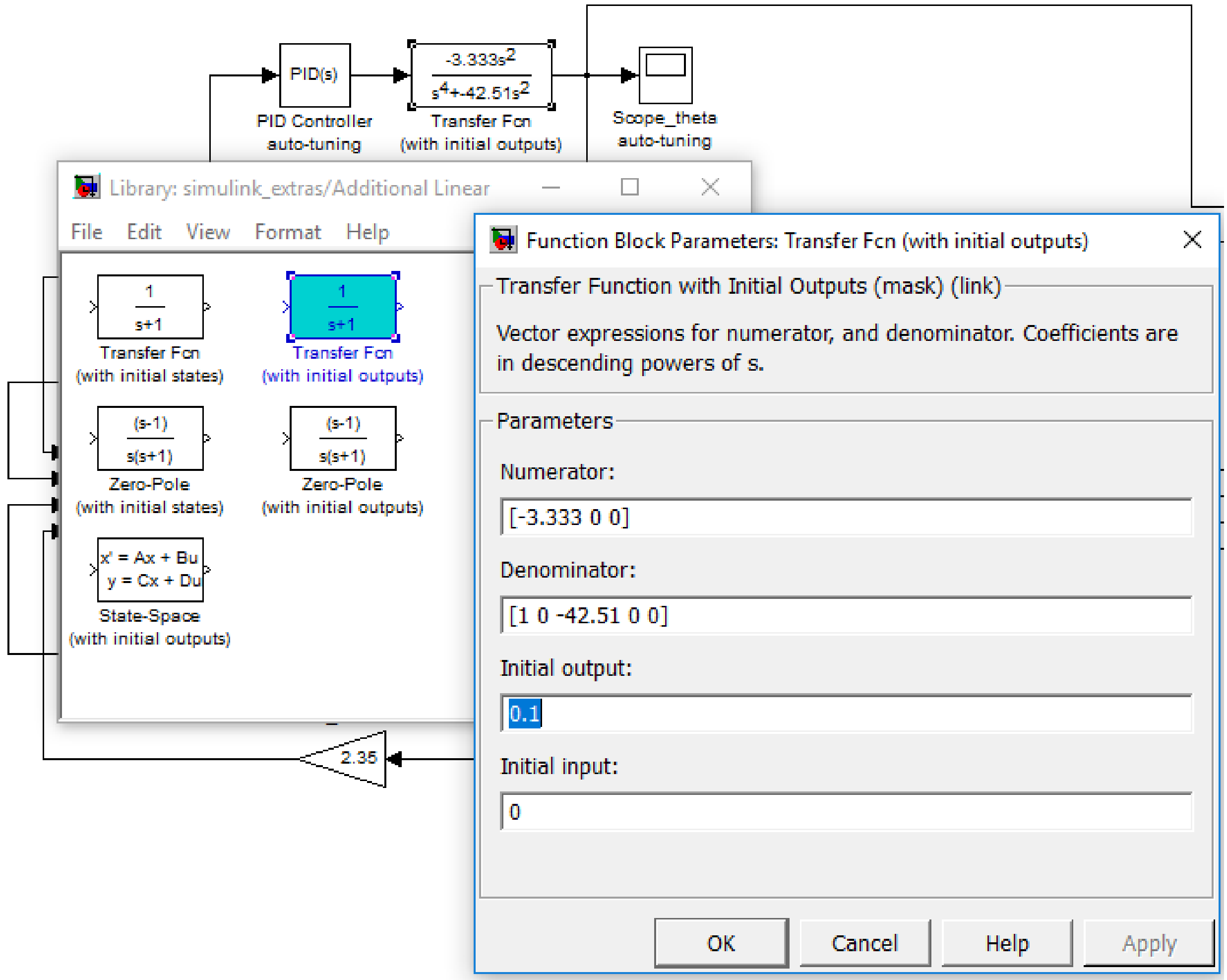

Figure 4. Creating the transfer function $G_1(s)$.

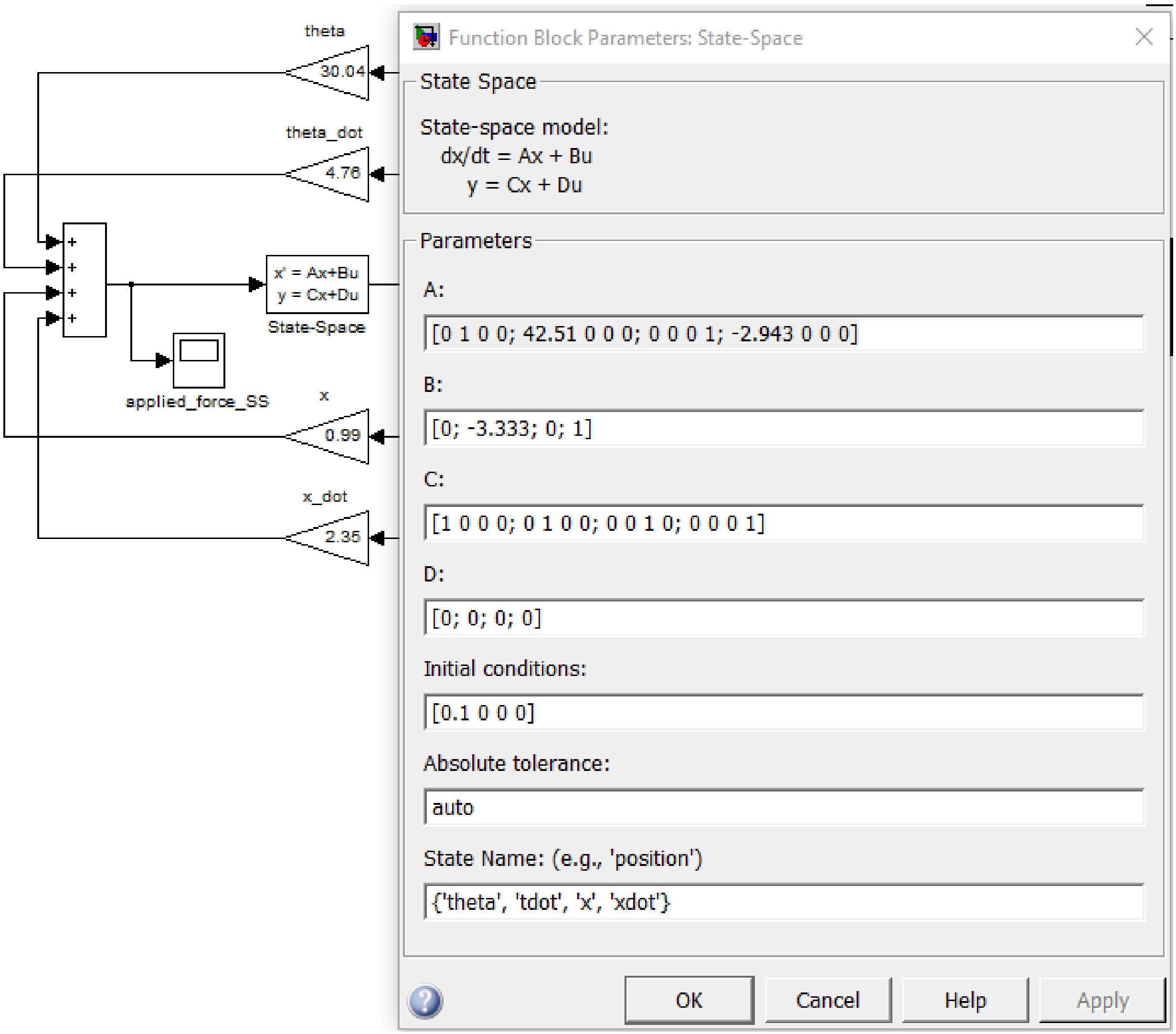

Figure 5. Creating the linearized model of the inverted pendulum on a cart using the block "State-Space".

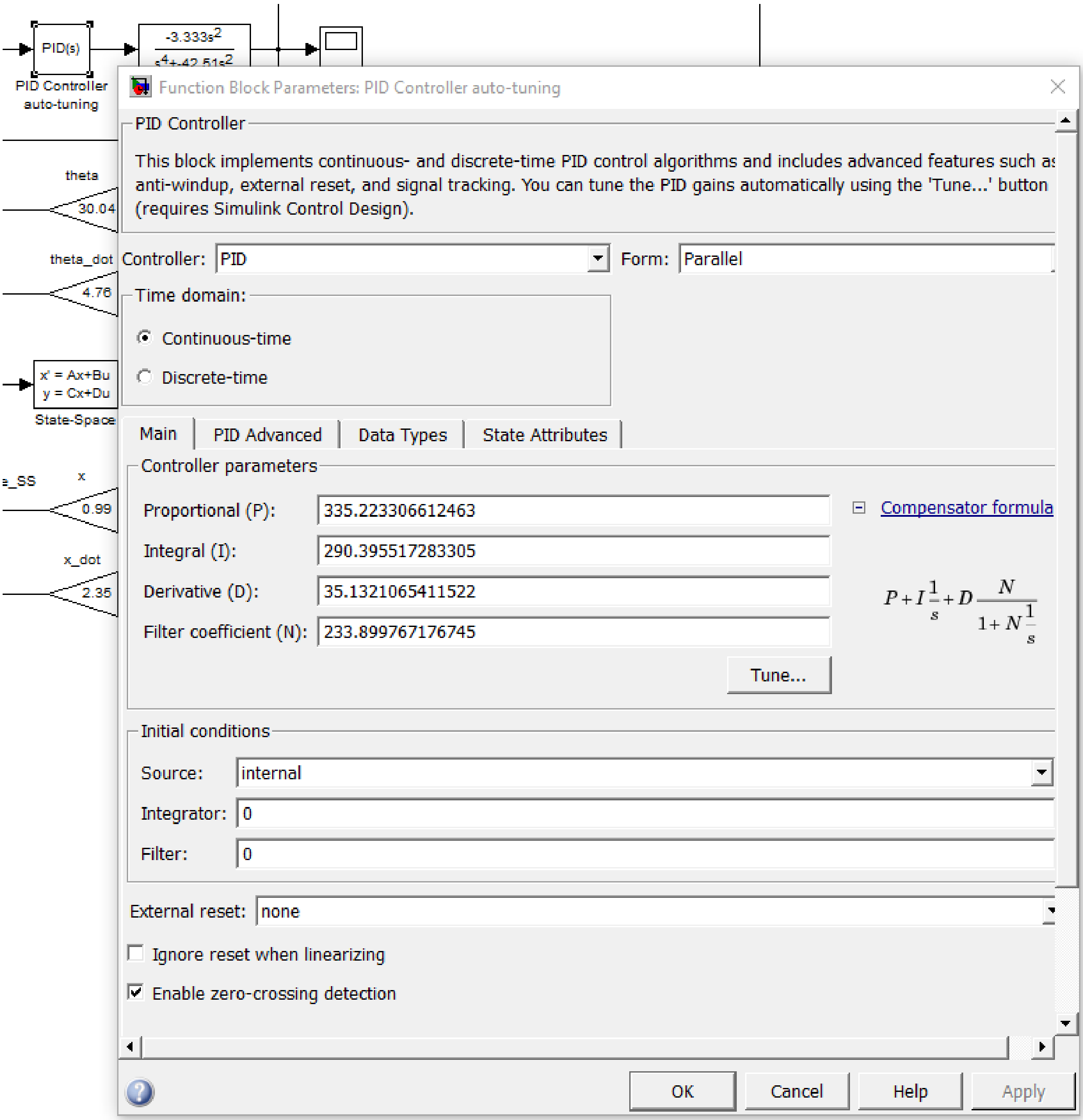

Figure 6. PID controller auto-tuning tool (PID Tuner GUI).

Regarding the design of fuzzy logic controller, you are referred to Example 10.3 on page 337 of Ref. [1]. The design process can be described through Figures 7-11.

Finally, Figure 12 shows the inverted pendulum angle response for three control strategies.

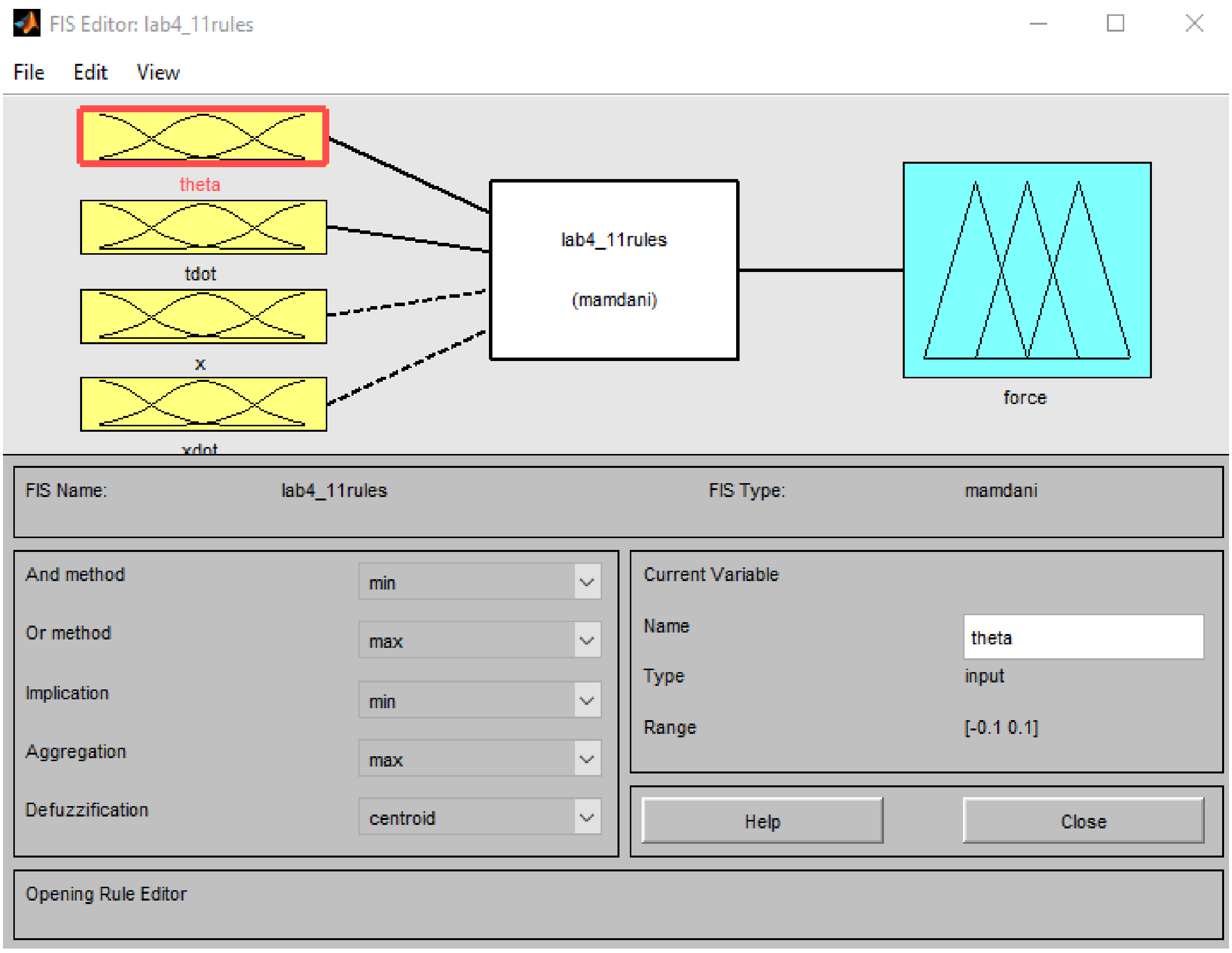

Figure 7. MATLAB Fuzzy Inference System (FIS) Editor.

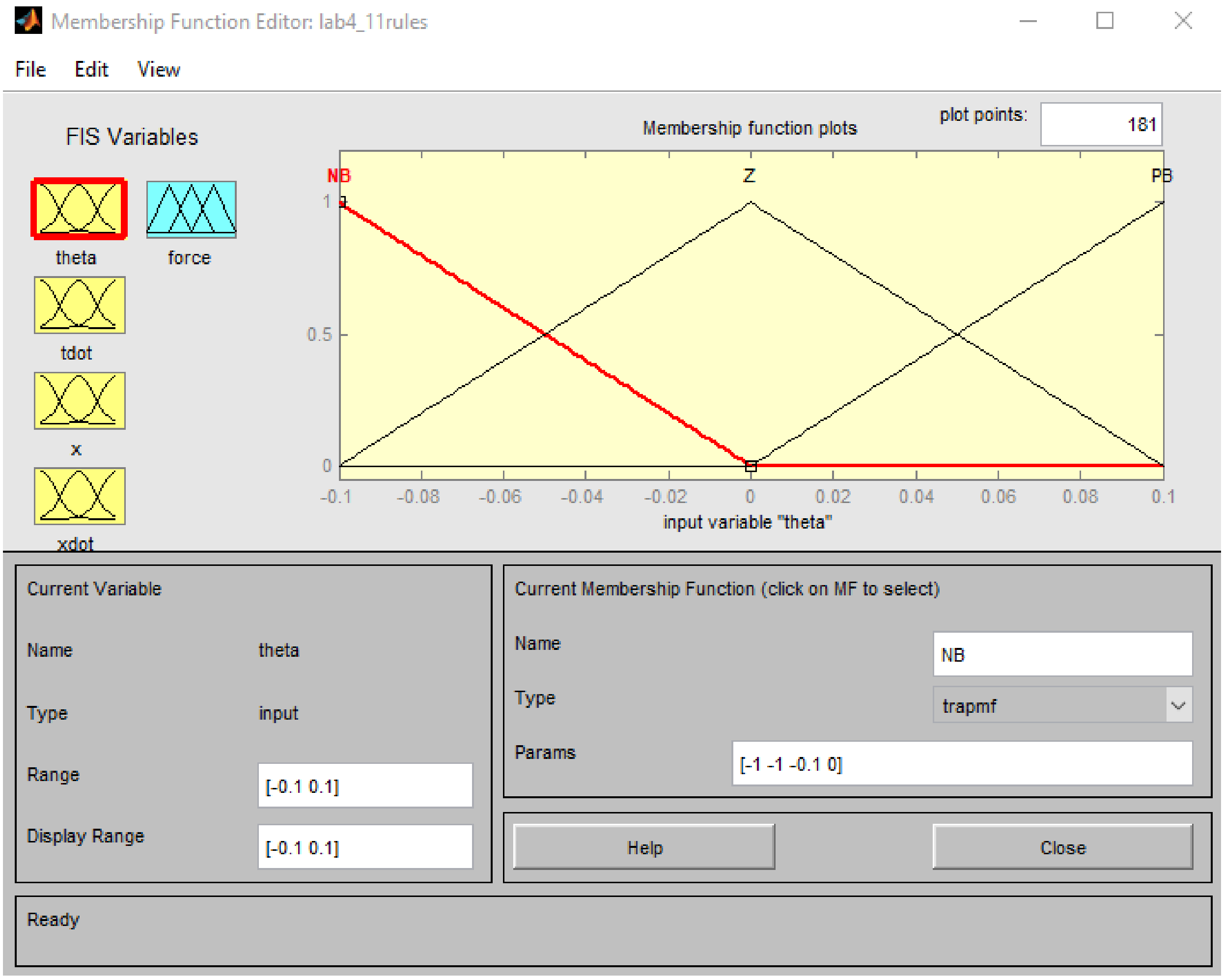

Figure 8. Membership Function Editor.

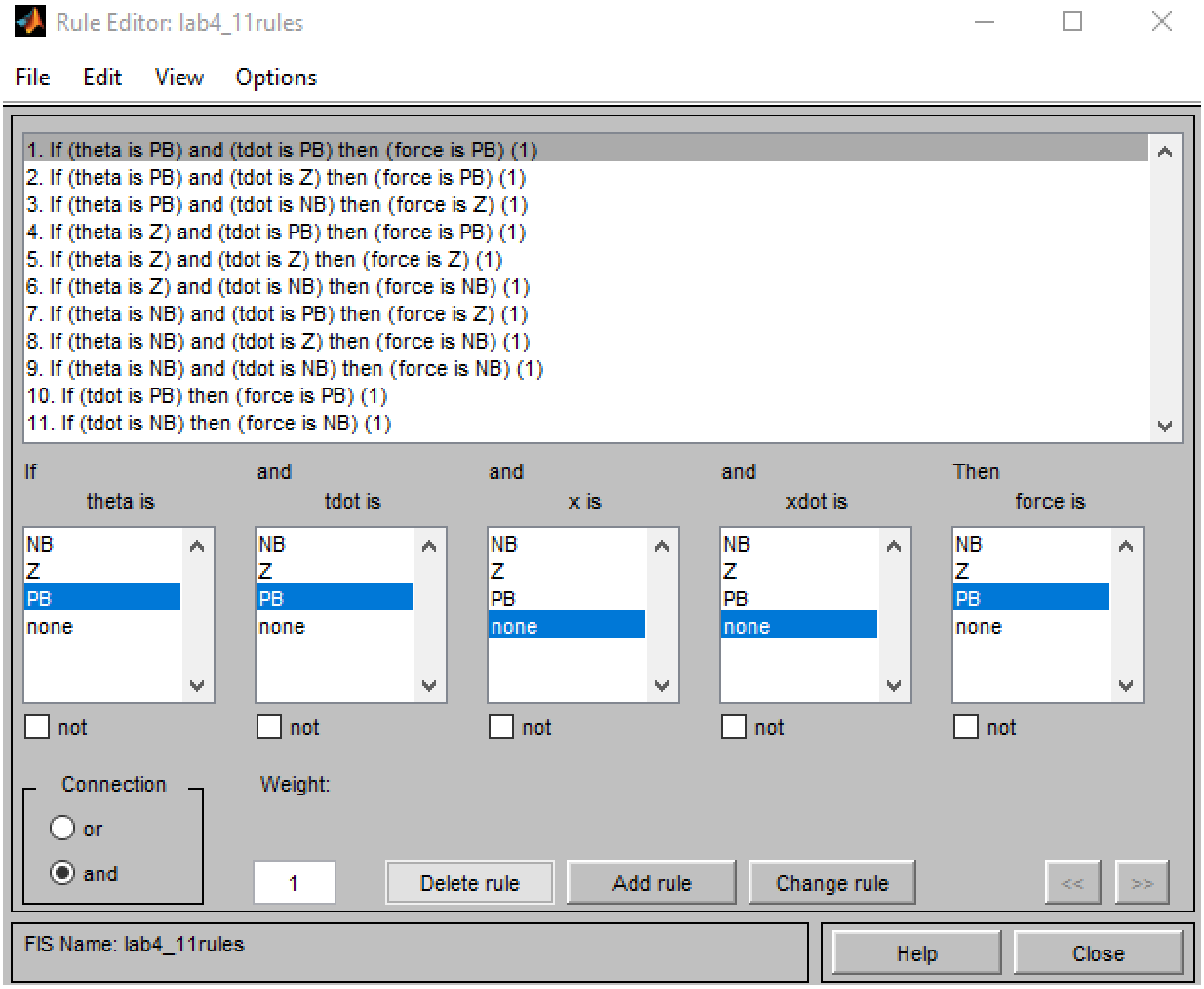

Figure 9. Rule Editor.

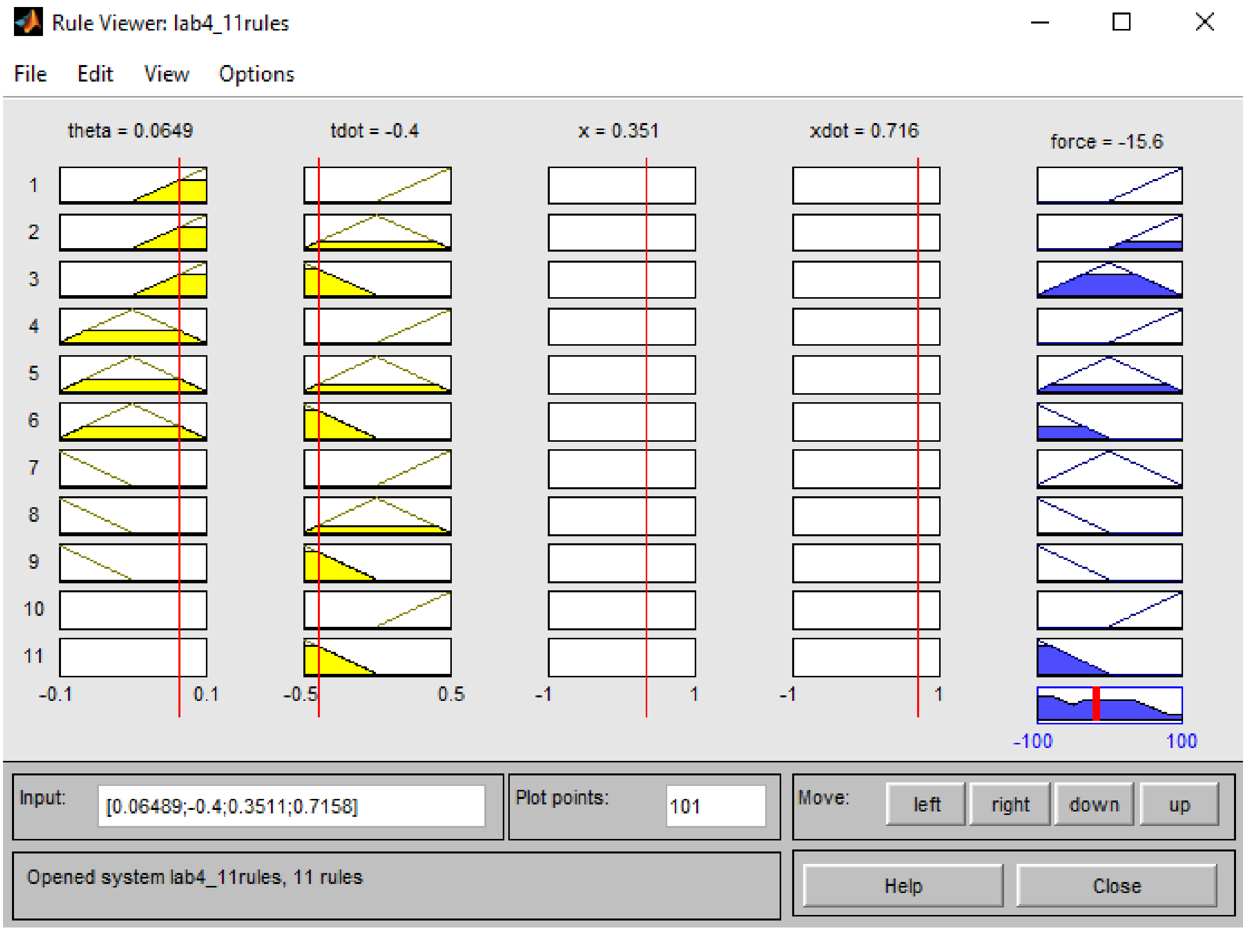

Figure 10. Rule Viewer.

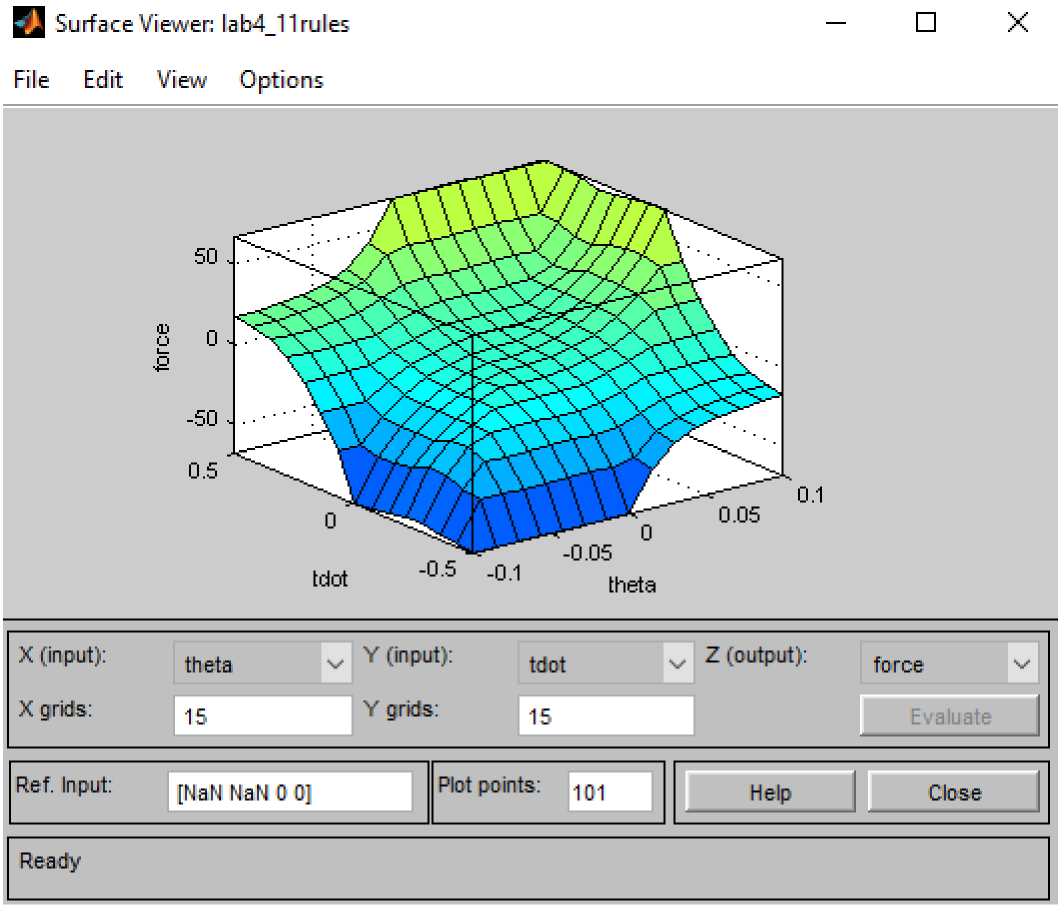

Figure 11. Surface Viewer.

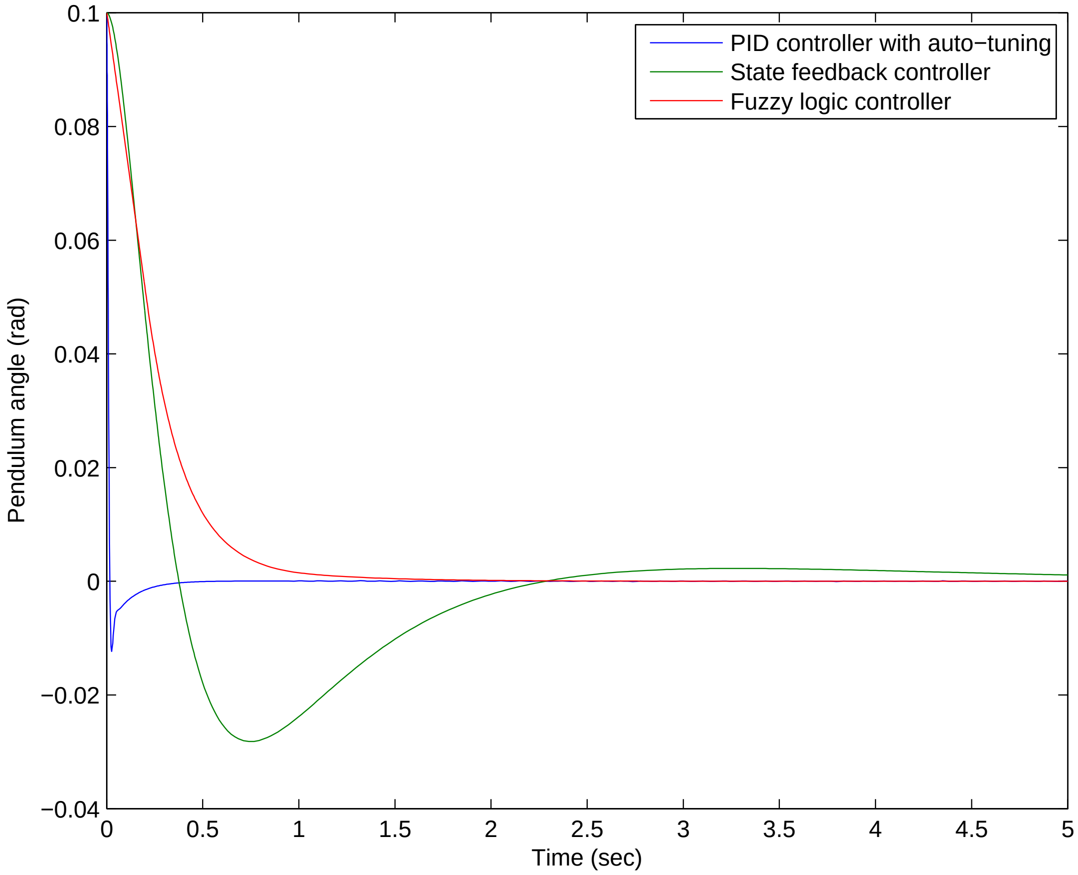

Figure 12. Inverted pendulum angle response for three control strategies.