ISSN: 2977-0041

Review Articles

Journal of Material Sciences and Engineering Technology

State-Space Analysis and Controller Design for a Dynamic System Using MATLAB

Ehsan Shirzad[1*] and Nima Hadizadeh[2]

[1]*Department of Electrical Engineering, University of Bojnord, Iran*

[2]*Toronto Metropolitan University, Toronto, Canada*

***Corresponding author**

Ehsan Shirzad, Department of Electrical Engineering, University of Bojnord, Iran.

Received: July 14, 2025; **Accepted:** July 21, 2025; **Published:** July 28, 2025

ABSTRACT

This paper presents a comprehensive analysis and design approach for a third-order linear timeinvariant (LTI) dynamic system using the state-space framework. Key contributions include deriving transfer functions from state-space representations, verifying system stability through eigenvalue and Lyapunov-based analysis, assessing system controllability and observability, and implementing canonical realizations. Advanced control strategies such as state feedback via pole placement and observer design are applied using MATLAB. Simulation results confirm system stability and tracking accuracy.

Keywords: State-Space, Stability Analysis, Lyapunov Method, Controllability, Observability, Canonical Realization, State Feedback Control, Observer Design, MATLAB Simulation

Introduction

The state-space approach offers a powerful and systematic method for modeling, analyzing, and designing control systems. Unlike classical methods based on transfer functions, state-space representation provides the flexibility to handle multi-input multi-output (MIMO) systems, timevarying systems, and initial conditions directly. This paper focuses on the analysis and controller design of a third-order LTI system using state-space tools. The primary goals include verifying stability, checking system controllability and observability, and implementing a full-state feedback controller and observer.

In the field of modern control engineering, the ability to understand and manipulate system dynamics is vital for achieving desired performance and robustness. Classical control approaches rely heavily on transfer functions, which, although effective for single-input single-output (SISO) systems, lack generality when dealing with complex, multi-variable systems. State-space analysis, on the other hand, offers a versatile framework where internal system states are explicitly modeled, allowing for more in-depth understanding of system dynamics and enabling advanced control strategies.

The shift toward state-space techniques has been particularly significant in areas such as aerospace, robotics, power systems, and process control. These fields often involve systems with multiple interacting variables and stringent requirements on stability, response time, and energy efficiency. By representing a system in terms of its state variables, we can design controllers that directly manipulate internal states, resulting in improved control accuracy and dynamic performance.

This paper aims to demonstrate how MATLAB-based tools can be leveraged to analyze the structural properties of a dynamic system, evaluate its stability using both analytical and numerical techniques, and implement effective state feedback and observer controllers. Each phase of the system analysis and design process is grounded in theory but verified through simulation, ensuring both academic rigor and practical relevance.

Ultimately, this work serves as a step-by-step guide to employing state-space methods for control design. It integrates fundamental concepts such as controllability, observability, and canonical realizations with practical controller synthesis to address real-

Citation: Ehsan Shirzad, Nima Hadizadeh. State-Space Analysis and Controller Design for a Dynamic System Using MATLAB. J Mat Sci Eng Technol. 2025. 3(3): 1-6. DOI: doi.org/10.61440/JMSET.2025.v3.59

world engineering problems. The provided examples and results highlight the value of modern control techniques in the age of automation and digital design.

System

The state space matrices of the system under consideration are as follows

$$B = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix} \quad A = \begin{pmatrix} -23.8268 & 5.1395 & -3.4960 \\ 00150 & -30.1122 & -.8091 \\ -0.0732 & 112.4109 & -108.3716 \end{pmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \quad D = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}$$

Determining the conversion function

To obtain the system transformation function for each input, we use thess2tf command.

[n1, d1] =ss2tf (a, b, c, d,1);
g1=tf (n1, d1);

s^2 + 138.5 s + 3354

g1= -----------------------------------
s^3 + 162.3 s^2 + 6654 s + 7.991e004

We can also obtain the conversion function between the output and the selected input for the other two inputs .

Sustainability review

To check the stability of the system, we obtain the eigenvalues of the matrix A. To obtain the eigenvalues, we use the eig command.

eig(a)
answer =
-23.8227
-31.2923
-107.1956

Since all the eigenvalues of matrix A are on the left side of the jw axis , the system is therefore stable.

Plotting unit shock and unit step responses and analyzing stability and permanent error

The step response of each transfer function is plotted in the figure below:

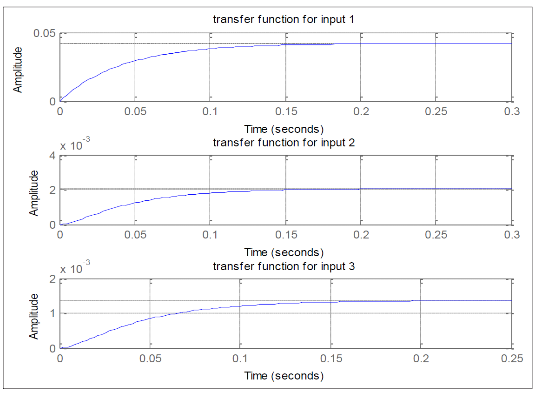


Figure 1: Step response

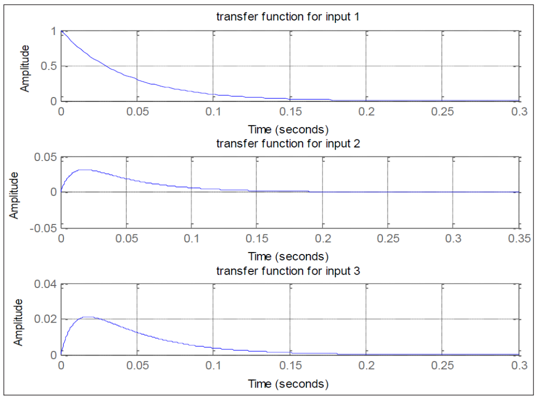


Figure 2: Impulse response in all three transfer functions

Obtaining the Rise, Delay, Landing and Maximum Jump Times as Well as the Maximum Jump Percentage

use the stepinfo command rise, delay, settle, and to obtain system specifications such as maximum jump times, as well as the maximum jump percentage.

info_g1 =

RiseTime: 0.0922
SettlingTime: 0.1642
SettlingMin: 0.0378
SettlingMax: 0.0420
Overshoot: 0
Undershoot: 0
Peak: 0.0420
PeakTime: 0.3281

info_g2 =

RiseTime: 0.0973
SettlingTime: 0.1766
SettlingMin: 0.0019
SettlingMax: 0.0021
Overshoot: 0
Undershoot: 0
Peak: 0.0021
PeakTime: 0.3281

info_g3 =

RiseTime: 0.0961
SettlingTime: 0.1748
SettlingMin: 0.0012
SettlingMax: 0.0014
Overshoot: 0
Undershoot: 0
Peak: 0.0014
PeakTime: 0.3175

Checking Controllability and Visibility

To check controllability, we first create the controllability matrix using the ctrb command, and if the rank of the matrix is complete, it indicates the controllability of the system.

ctrb_matrix=rank(ctrb(a,b))

1.0e+004 *

0.0001 0 0 -0.0024 0.0005 0.0003 0.0568 -0.0670 -0.0458 0
0.0001 0 0.0000 -0.0030 0.0001 -0.0001 0.0816 -0.0112
0 0 -0.0001 -0.0000 0.0112 0.0108 0.0011 -1.5567 -1.1654

ctrb_matrix =3

The rank of the matrix is 3 instead of 9, so the system is not controllable

And similarly, to check visibility, we first create the visibility matrix using theobsv command, and if the rank. of the matrix is complete, the system is visible

obsv_matrix=rank(obsv(a,c))
answer =

1.0000 0 0
-23.8268 5.1395 -3.4960
568.0494 -670.2080 458.0072

obsv_matrix =3

By considering one input, for example the first input, and checking the controllability condition, we will have
answer =

1.0000 0 0
-23.8268 5.1395 -3.4960
568.0494 -670.2080 458.0072

rank(ans) answer = 3
So. the system is stable, controllable, and observable for only the first input,

Investigating Stability in the Lyapunov Sense - 7
Input-output stability:
The purpose of this stability is to ensure that the output remains limited for a limited. range of input.

To demonstrate this, we applied a step input to the system and observed the output. As seen in Figure 1, the output range remains limited.

Stability in the sense of Lyapunov:
In this stability, for initial conditions and no input, the output must converge to zero. Considering the controller canonical and for zero input and initial conditions [2,3,.4], the output states are as follows.

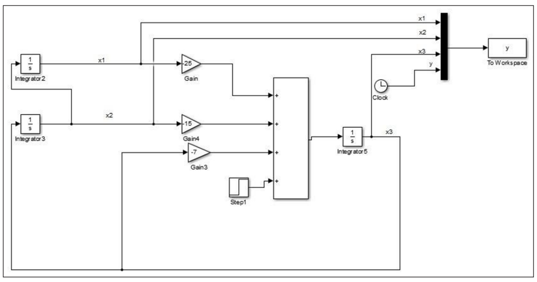


Figure 3: Block diagram for checking Lyapunov stability

As can be seen in Figure 4, the states have converged towards zero for zero input and initial. conditions, so the system is stable in the Lyapunov sense.

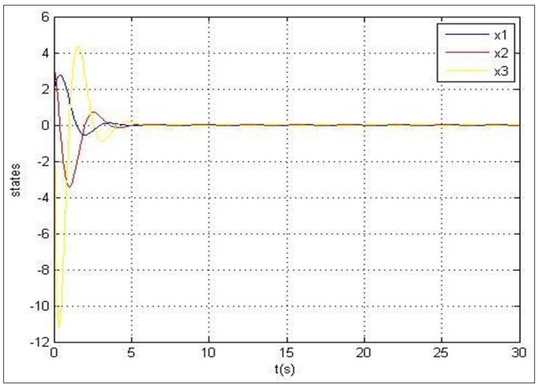


Figure 4: Output states for zero input and initial conditions

Realizations -8
Fulfillment Canonical Seer

$$\dot{x}(t) = \begin{pmatrix} 0 & 0 & -a_0 \\ 1 & 0 & -a_1 \\ 0 & 1 & -a_2 \end{pmatrix} x(t) = \begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix} u(t)$$

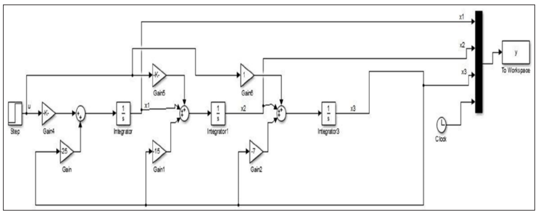
$$y(t) = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} x(t)$$

In this case, the derivative of the states is as follows.

$$\dot{x}_1 = -a_0 x_3 + b_0 u$$
$$\dot{x}_2 = x_1 - a_1 x_3$$
$$\dot{x}_3 = x_2 - a_2 x_4$$

Considering the transformation functiong, we have the following

s^2 + 138.5 s + 3354
g= -----------------------
s^3 + 7 s^2 + 15 s + 25



Block diagram Implementation Canonical Seer

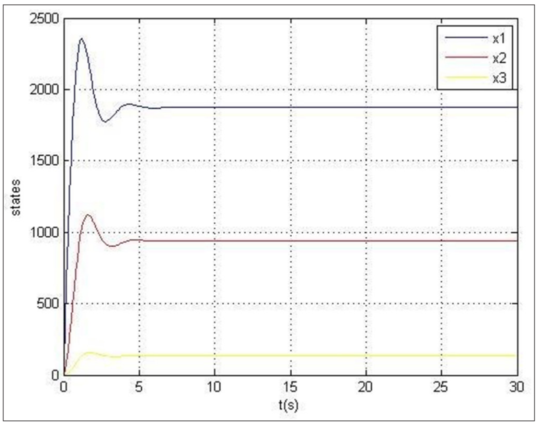


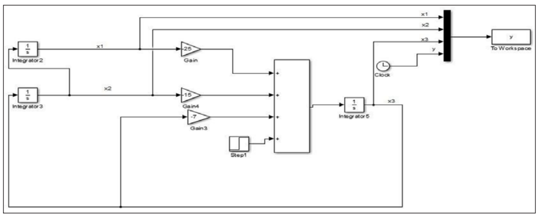
Figure 6: Output of states in Canonical Seer

Realization

Canonical controller realization

$$\dot{x}(t)=\begin{pmatrix}0 & 1 & 0\\ 0 & 0 & 1\\ -a_0 & -a_1 & -a_2\end{pmatrix}x(t)=\begin{pmatrix}0\\0\\1\end{pmatrix}u(t)$$

$$y(t)=\begin{bmatrix}b_0 & b_1 & b_2\end{bmatrix}x(t)$$



Block diagram Implementation Canonical Controller

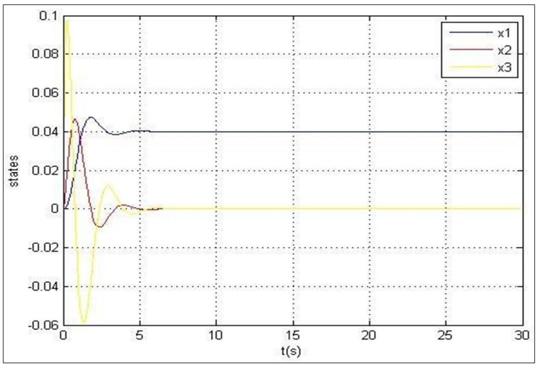


Figure 8: Output of states in Controller Canonical

Realization

Canonical realization of controllability

$$\dot{x}(t)=\begin{pmatrix}0 & 0 & -a_0\\ 1 & 0 & -a1\\ 0 & 1 & -a_2\end{pmatrix}x(t)=\begin{pmatrix}1\\0\\0\end{pmatrix}u(t)$$

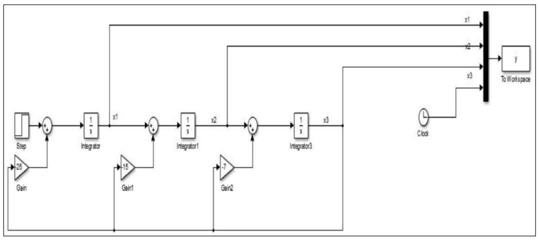
$$y(t)=\begin{bmatrix}\beta_2 & \beta_1 & \beta_0\end{bmatrix}x(t)$$

are the Markov parameters β_i where.

$\beta_3 = h_1, \beta_2 = h_2, \beta_1 = h_3$

h_i obtain to We use the following relationship.

h1=c*b =0;
h2=c*a*b=0;
h3=c*a^2*b=45;



Implementation block diagram Canonical controllability

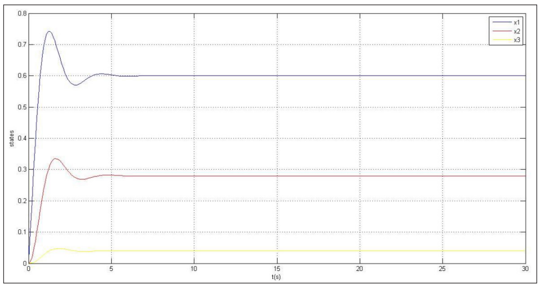


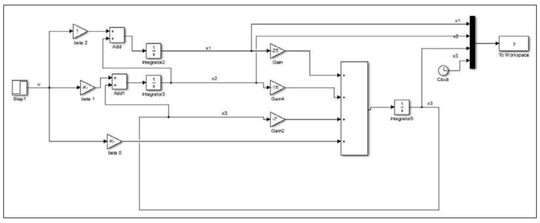
Figure 10: Output of states in controllability Canonical

Realization

Canonical realization of visibility

$$\dot{x}(t)=\begin{pmatrix}0 & 1 & 0\\ 0 & 0 & 1\\ -a_0 & -a_1 & -a_2\end{pmatrix}x(t)=\begin{pmatrix}\beta_2\\ \beta_1\\ \beta_0\end{pmatrix}u(t)$$

$$y(t)=\begin{bmatrix}1 & 0 & 0\end{bmatrix}x(t)$$



block diagram Implementation Canonical visibility

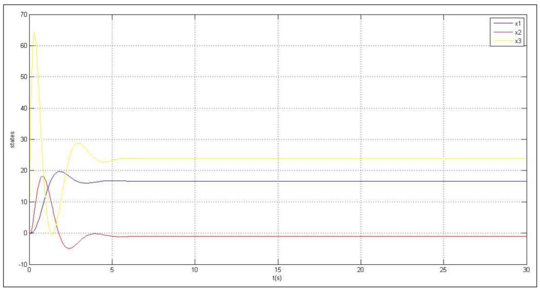


Figure 12: Output of states in visibility Canonical

Realization

Mode Feedback Controller -9

Considering that for the design of the state feedback controller, the system must be controllable and observable, and as it was observed, the system is not controllable for all 3 inputs, but for one input, for example, the first input, it was observed that the system is stable. We design the state feedback controller for the first input. We want the poles. It should be as follows

[-1+2*i -1-2*i -5]

The desired system has two dominant poles and one pole at -5, which, due to its distance, has no. effect on the dominant poles.

To obtain the gain matrixk, we use the place. command
a= [0 1 0; 0 0 1; -25 -15 -7]; b= [0;0;1]; c= [3354 138.5 1]; d=0;

k=place (a,b, [-1+2*i -1-2*i -5]);

For initial conditions of zero and step reference input, we want the controller to drive the system. output to one. The realization is as follows

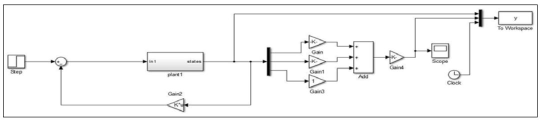


Figure 13: Block diagram of the state feedback controller.

The response form of the states and output is as follows

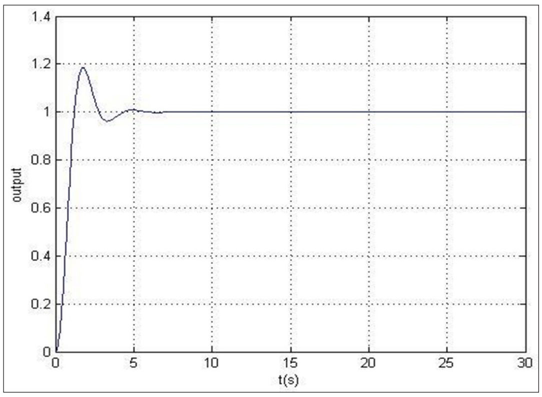


Figure 14: System output response (without initial conditions)

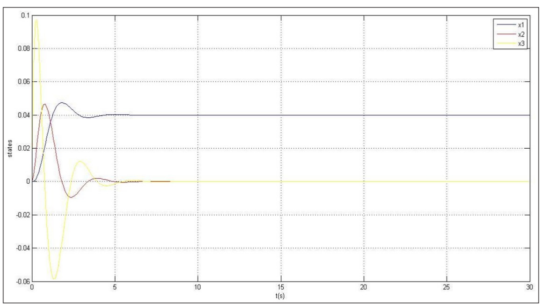


Figure 15: Output states in a state feedback controller (without initial conditions)

The shape of the states and output with the following initial conditions are specified in Figures and 17, 16

x_0= [2;3;.4]

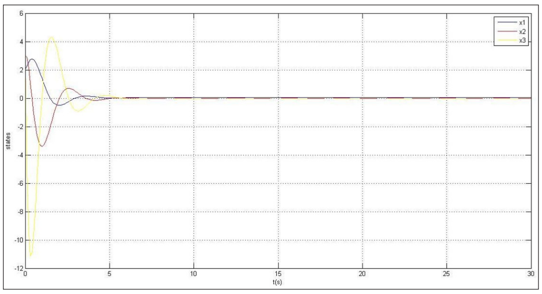


Figure 16: State output in a state feedback controller (with initial conditions)

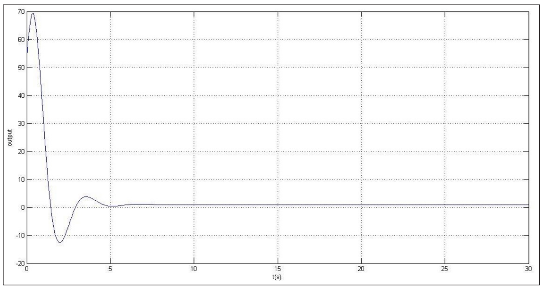


Figure 17: System output response (with initial conditions)

Design of the Pole-viewer Location Controller -10
To design the pole locating controller, we assume that the desired poles of the system are like the pole section and we place the observer poles at the points [-2 -3 -10].

For realization, we use controller realization. In this case, the matricesa, b, c, d. are as follows

a= [0 1 0; 0 0 1; -25 -15 -7]; b= [0;0;1]; c= [3354 138.5 1]; d=0;
k=place (a,b, [-1+2*i -1-2*i -5]); l=place (a',c', [-2 -3 -10])';

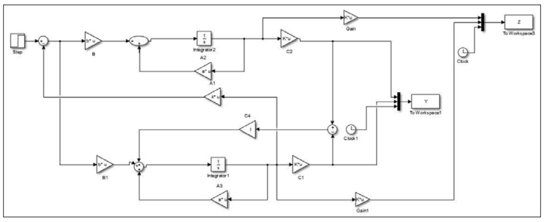


Figure 18: Block diagram of state feedback control with observer

The output result can be seen in Figure 19

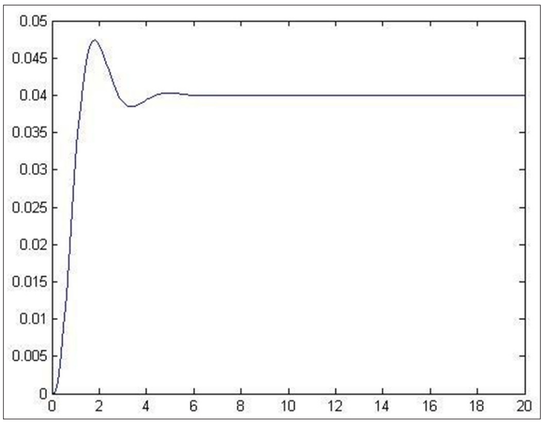


Figure 19: System output result

The system output response along with the observer output is shown in Figure 20

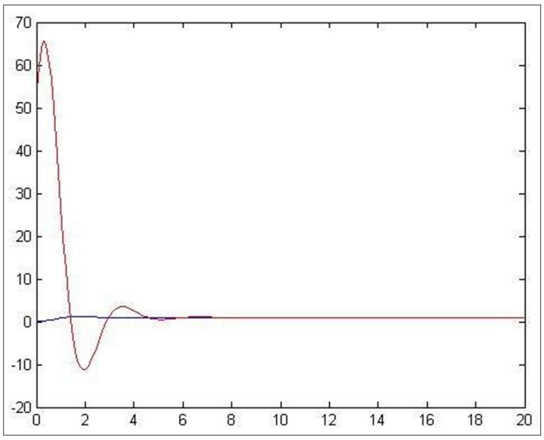


Figure 20: Output result with viewer

Conclusion

This paper presented a detailed study of a third-order LTI system using state-space methods. From modeling to simulation, and through canonical transformations to advanced controller design, all steps were implemented and validated in MATLAB. The system's controllability, observability, and stability were analyzed both theoretically and through simulations. The combination of state feedback and observer design successfully ensured robust control and estimation.

References

1. Ogata k. Modern Control Engineering, 5th ed. Upper Saddle River, NJ, USA: Prentice Hall. 2010.
2. Franklin GF, Powell JD, Emami-Naeini A. Feedback Control of Dynamic Systems, 7th ed. Boston, MA, USA: Pearson. 2014.
3. Friedland B. Control System Design: An Introduction to State-Space Methods. Mineola, NY, USA: Dover Publications. 2005.
4. Levine WS. The Control Handbook: Control System Fundamentals, 2nd ed. Boca Raton, FL, USA: CRC Press. 2011.
5. Zhou K, Doyle JC, Glover K. Robust and Optimal Control. Upper Saddle River, NJ, USA: Prentice Hall. 1996.
6. Antsaklis PJ, Michel AN. Linear Systems. Boston, MA, USA: Birkhäuser. 2006.
7. Kailath T, Linear Systems, Englewood Cliffs, NJ, USA: Prentice Hall. 1980.
8. Chen CT. Linear System Theory and Design, 4th ed. Oxford, U.K.: Oxford University Press. 2013.
9. Kwakernaak H, Sivan R. Linear Optimal Control Systems. New York, NY, USA: Wiley-Interscience. 1972.
10. MathWorks, Inc., Control System Toolbox User's Guide, Natick, MA, USA. 2024.