Research Article

Neural Networks Application for the Data of PID Controller for Acrobot

Nguyen Cong Danh [ID]

The Independent Researcher, Pailin, Cambodia

Correspondence should be addressed to Nguyen Cong Danh; sugus2101@gmail.com

Acrobots are a system that has levels of operating states in many investigated cases, and they are subjects to many events during operation due to the mechanisms of locomotion processes. These states have been investigated in specific situations. Due to the limited nature of surveying under conditions without the aid of software fine-tuning the desired output values, designers have to create a number of algorithms that control the system most appropriately in a complex working environment of this system. In this study, the author has proposed design problems to suit the working needs. The author has modeled the objects to be investigated, and at the same time, the author has combined simulation of phenomena that have been documented from the theoretical model of this system before. Control an Acrobot system, including Neural Network Application for the data of the PID controller (closed form). Mathematical models, Simulink, are also presented specifically through this study. The simulation parameters have been adjusted to match the set criteria, and the reader's perception will be more intuitive through simulation interpretations. Based on simulation data, the system performance analysis becomes more accurate than ever. The above suggestions are intended to serve vocational education and scientific research, and at the same time, they also contribute to suggesting new and unique ideas. The ANN is the most intelligent control method currently added in this study to firmly confirm its effectiveness in all problems related to artificial intelligence. Proposing control strategies for different models is also suggested by the author in the conclusion. Neural networks application is a highly applicable method because it works in sync with other methods. In other words, neural networks application can be applied to any situation based on given data. Specifically, in this study, the application of neural networks becomes more flexible and vivid thanks to the data of the PID controller. The purpose of this method is to increase the security of the system against the attack of hackers on facilities of the automatic control system. Simulation is done by Matlab.

1. Introduction

In the future, in control algorithms, including PID, the artificial intelligence mode can be applied to flexible electric motors, and electromechanical devices have yielded encouraging results. Above control methods are an important component in controlling a flexible Acrobot. Artificial intelligent (Ai) algorithms have also been implemented in labs for Acrobots, and they can appear in industry factories. Flexible robots are controlled by a combination of genetic algorithms [1] and classical algorithms: PID controller, and fuzzy mode to intelligent algorithms: Ai [2]. Broadly understood, Papoutsidakis et al. [2] included algorithms associated with neural networks. These algorithms will be applied to a small branch of robotic feet. Research on these control algorithms brings many practical applications such as self-flying fighter and self-driving aircraft carrier. In this study, the author proposed new several control methods to control a relatively complex system, Acrobot. These control methods are viewed as a direct application to Acrobot. Besides, the application of artificial intelligence has been mentioned in this topic. Acrobot systems as well as other systems that are structurally similar to inverted pendulum, Pendubot [3], and reaction wheel inverted pendulum [4], which are nonlinear systems such as single-input multi-output (SIMO). Although they have a similar structure, they

have their own ways of operating according to a particular mechanism. In terms of physical theory, the mathematical basis for forming their physical phenomena is different. They are nonlinear systems. They do not follow a linear phenomenon because their model is complex. They have many interactions between metal rods that are joined together. Therefore, the design of controllers for these systems always faces challenges with many signals at the output. Output signals are fair valid only when positions near the working point are considered in this study. Modern control methods are implemented according to this rule, including neural networks application. This system is described by 2 metal/hard plastic bars that have a joint between 2 metal/hard plastic bars. The first metal/hard plastic bar is denoted "link 1" and the 2nd metal/hard plastic bar is denoted "link 2." This coupling is regulated by the motor so that the movement of the system is smooth. This joint is a joint with flexible movement with large opening angle between two bars. The top of "link 1" is attached to the passive joint and "link 1" is moved freely around this passive joint. This passive joint is mounted on a fixed support. The Acrobot system is shown by a link between "link 1" and "link 2" (Figure 1).

2. Dynamic Equation of the Model System

All models are represented by physical phenomena. Mathematical equations reflect physical properties and phenomena. Operating states of the system are described by these mathematical equations. Through these equations, the author has established a model with the intervention of (Ai). The purpose of this is the author's wish. The author desired that the system achieves stable values under the control of the controllers. In the future, the controllers can be controlled remotely. This is useful for cases where working conditions are harsh for humans. Then, one or more controllers are connected to a system, and they are controlled by the remote control system. After successfully setting up the mathematical equations, the author has processed to calculate values: the value of the response of the PID controller and the value of neural networks application. After finding above values, the author found the control signals in the open system and closed system for the PID controller. Acrobot or an inverted pendulum can be described by mathematical equations. The two-dimensional coordinate system, Ox and Oy, is shown in Figure 1 [5]. The control method for surveys [6–8] can be applied to nonlinear models such as SIMO. Planar dictubot [9] with the artificial intelligence method is an attractive title for readers. Swing-up control [10] for nonlinear models like SIMO is a good idea. Chaotic perturbation [11] can be applied to nonlinear models such as SIMO. Hybrid control [12] is combined with the neural network to form a new research direction. Application of the ant colony algorithm [13] in optimal control of robotic foot systems is an exciting future work. Energy-based control [14] for the robot's feet through their movement is an elaborate research program. Hybrid control [15] with navigation functions for the inverted pendulum is a great mission. The robust control for stabilization of the noninertial system [16] can be
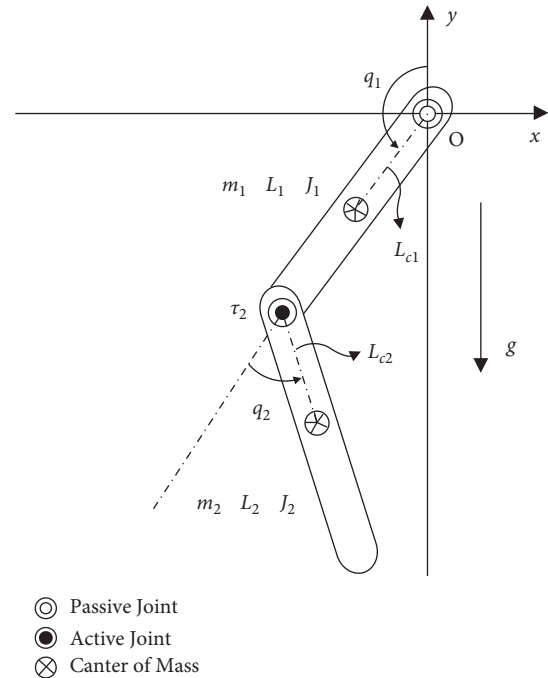

Figure 1: Mathematical model of Acrobot [25].

Table 1: Parameters of Acrobot.

Parameters	Values
m_1	0.8 kg
O_1	0.18 m
O_{C1}	0.11 m
m_2	0.2 kg
O_2	0.18 m
O_{c2}	0.09 m
J_1	0.0022 kg·m^2
J_2	0.00054 kg·m^2

applied to an inverted pendulum. A positioning control strategy [17] is intended for nonlinear systems to examine their behavior more closely. Balance control [18] for Pendubot in various forms is of great interest to the author. The inverse linear quadratic method [19] can be used for nonlinear systems. Several new algorithms have been implemented [20, 21] for controlling robots which can replace neural network applications. Approaches in the processing of signals materialized by abstract images [22] have also been studied in recent years. The development from classical algorithms to more modern algorithms [23, 24] has promoted its effectiveness.

In this study, neural network applications are designed after a linearization process is established, as given in Section 3. The linearization of this model makes the survey more convenient in the process of controlling the properties around the working point as described above. This purpose contributes to the stabilization of nonlinear structures. The challenges of the proposed strategy: the working environment of this system is relatively complex. Therefore, it is necessary to stabilize the structure properties before controlling them with any control method.

Physical parameters of Acrobot are given in Table 1. Variables of the mathematical equation are q_1, q_2, and τ_2, where q_1 and q_2 are the output signals and τ_2 is the input signal. Variables are unknown values. These variables will be solved through the establishment of a mathematical equation presented. In the process of designing controllers for these systems, readers can better understand the nature of the relationships between input signals and output signals through voltage signals, the torque of an electric motor. In Figure 1, the x-axis of the Cartesian coordinate system was chosen to be the reference level of zero potential energy. Letting $S_i = [S_i^x, S_i^y] \in R^2$ be the absolute position of the COM of the ith link gives [5]

$$S_1 = \begin{bmatrix} Oc_1 \sin q_1 \\ Oc_1 \cos q_1 \end{bmatrix}, \tag{1}$$

$$S_2 = \begin{bmatrix} O_1 \sin q_1 + O_{c2} \sin(q_1 + q_2) \\ O_1 \cos q_1 + O_{c2} \cos(q_1 + q_2) \end{bmatrix}. \tag{2}$$

The kinetic energy is $Y(q, \dot{q})$, and the potential energy is $V(q)$:

$$Y(q, \dot{q}) = \frac{1}{2} \sum_{i=1}^{2} \left[m_i S_i^2 + J_1 \dot{q}_1^2 + J_2 (\dot{q}_1 + \dot{q}_2)^2 \right], \tag{3}$$

$$Y(q, \dot{q}) = \frac{1}{2} [\dot{q}_1 \ \dot{q}_2] M(q_2) \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix}, $$

where

$$M(q_2) = \begin{bmatrix} a_1 + a_2 + 2a_3 \cos(q_2) & a_2 + a_3 \cos(q_2) \\ a_2 + a_3 \cos q_2 & a_2 \end{bmatrix}, \tag{4}$$

$$\begin{cases} a_1 = m_1 O_{C1}^1 + m_2 O_1^2 + J_1, \\ a_2 = m_2 O_{C2}^2 + J_2, \\ a_3 = m_2 O_1 O_{C2}, \\ a_4 = (m_1 O_{c1} + m_2 O_1) g, \\ a_5 = m_2 O_{c2} g. \end{cases} \tag{5}$$

Considering friction is very small, the author used Lagrangian of Acrobot; then, the dynamic equation of the mechanical system is

$$V(q) = m_1 g S_1^y + m_2 g S_2^y = a_4 \cos q_1 + a_5 \cos(q_1 + q_2), \tag{6}$$

$$\frac{d}{dt} \left[\frac{\partial O(q, \dot{q})}{\partial \dot{q}_i} \right] - \frac{\partial O(q, \dot{q})}{\partial q_i} = \dot{\tau}_i, \quad i = 1, 2, \tag{7}$$

where $O(q, \dot{q}) = Y(q, \dot{q}) - V(q)$ and $\tau_1 = 0$. Equation (7) is equivalent to

$$M(q_2)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \begin{bmatrix} 0 \\ \tau_2 \end{bmatrix}, \tag{8}$$

where

$$q = [q_1 \ q_2]^T, \tag{9}$$

$$C(q, \dot{q}) = \begin{bmatrix} -a_3 \dot{q}_2 \sin q_2 & -a_3(\dot{q}_1 + \dot{q}_2)\sin q_2 \\ a_3 \dot{q}_1 \sin q_2 & 0 \end{bmatrix}, \tag{10}$$

$$G(q) = \begin{bmatrix} -a_4 \sin q_1 - a_5 \sin(q_1 + q_2) \\ -a_5 \sin(q_1 + q_2) \end{bmatrix}. \tag{11}$$

State space equation is created as

$$x = [x_1 \ x_2 \ x_3 \ x_4]^T = [q_1 \ \dot{q}_1 \ q_2 \ \dot{q}_2]^T. \tag{12}$$

Let

$$H(q, \dot{q}) = [H_1(q, \dot{q}) \ H_2(q, \dot{q})]^T := C(q, \dot{q})\dot{q} + G(q). \tag{13}$$

Then, the dynamic equation (8) can be described as

$$\dot{x} = f(x) + g(x)\tau_2. \tag{14}$$

where

$$\begin{cases} f(x) = [x_2 \ R(1,1) \ x_4 \ R(2,1)]^T, \\ g(x) = [0 \ I(1,1) \ 0 \ I(2,1)]^T, \end{cases} \tag{15}$$

where $R(i,j)$ and $I(i,j)$ are the matrices having the ith row and j^{th} column of R and I. Therefore, R and I are determined as

$$\begin{cases} R = -M^{-1}(q_2) \begin{bmatrix} H_1(q, \dot{q}) \\ H_2(q, \dot{q}) \end{bmatrix}, \\ I = -M^{-1}(q_2) \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \end{cases} \tag{16}$$

Parameters of the model are given in Table 2.

3. The Survey of the System

The state variable equations of the system are described as follows:

$$\begin{cases} \dot{x} = Ax + Bu, \\ c = Cx + Du, \end{cases} \tag{17}$$

where

$$A = \begin{bmatrix} \dfrac{\partial f_1}{\partial x_1} & \dfrac{\partial f_1}{\partial x_2} & \dfrac{\partial f_1}{\partial x_3} & \dfrac{\partial f_1}{\partial x_4} & \cdots & \dfrac{\partial f_1}{\partial x_n} \\ \dfrac{\partial f_2}{\partial x_1} & \dfrac{\partial f_2}{\partial x_2} & \dfrac{\partial f_2}{\partial x_3} & \dfrac{\partial f_2}{\partial x_4} & \cdots & \dfrac{\partial f_2}{\partial x_n} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \dfrac{\partial f_1}{\partial x_1} & \dfrac{\partial f_1}{\partial x_2} & \dfrac{\partial f_1}{\partial x_3} & \dfrac{\partial f_1}{\partial x_4} & \cdots & \dfrac{\partial f_1}{\partial x_n} \end{bmatrix}, B = \begin{bmatrix} \dfrac{\partial f_1}{\partial u} \\ \dfrac{\partial f_2}{\partial u} \\ \cdots \\ \dfrac{\partial f_n}{\partial u} \end{bmatrix}. \tag{18}$$

With above parameters, matrices A and B of the state space model, $G_1(s)$, $G_2(s)$, are calculated:

Parameters and variables	Described
q_1	Angle of link 1
q_2	Angle of link 2
$\dot{q}_1$	Angle velocity of link 1
$\dot{q}_2$	Angle velocity of link 2
m_1	Mass of link 1
O_1	Length of link 1
O_{c1}	Distance from passive joint to the center of mass of the link 1
m_2	Mass of link 2
O_2	Length of link 2
O_{c2}	Distance from active joint to the center of mass of the link 1
J_1	Moment of inertia link 1
J_2	Moment of inertia link 2
g	Gravitational acceleration
τ_2	Torque applied to active joint

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 70.6961 & -19.6783 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -94.9903 & 130.9458 & 0 & 0 \end{bmatrix},$$

$$B = \begin{bmatrix} 0 \\ -185.7355 \\ 0 \\ 927.3017 \end{bmatrix} C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, D = 0. \tag{19}$$

The transfer function method is a mathematical description of an automated system that makes it easier to investigate the system. The transfer function of the system is formed by converting the mathematical description from the form of an equation of state to the form of a transfer function. This is done using Matlab software commands. The transfer function of the system G_1, G_2, G_3, G_4. G_1, G_2, G_3, G_4 can be examined separately using the PID controller in each specific case: G_1 is connected to G_{PID} and G_2 is connected to G_{PID} as follows:

$$G_1(s) = \frac{-185.7}{s^2 + 19.68s - 70.7},$$

$$G_2(s) = \frac{927.3s^2 - 6074s - 4.791 \times 10^4}{s^4 + 19.68s^3 - 70.7s^2},$$

$$G_3(s) = \frac{-185.7s}{s^2 + 19.68s - 70.7}, \tag{20}$$

$$G_4(s) = \frac{927.3s^2 - 6074s - 4.791 \times 10^4}{s^3 + 19.68s^2 - 70.7s}.$$

4. Controller Design Using PID Stratery

PID controllers are commonly used to regulate the time domain behavior of many different types of dynamic plants [26]. The transfer function of PID control is given by

$$G_{PID} = K_P + \frac{K_I}{s} + K_D s = \frac{K_D s^2 + K_P s + K_I}{s}$$

$$= \frac{K_D \left[s^2 + K_P/K_D s + K_I/K_D \right]}{s}, \tag{21}$$

where K_p, K_i, and K_d are the parameters of the PID controller. These parameters can be adjusted according to according to the requirements given. G_{PID} can be directly connected to the system transfer function.

5. Model Using the Artificial Neural Network (ANN)

5.1. An Artificial Neural Network. An artificial neural network (or neural network for short) can be seen as a simple mathematical model of the human brain. Neural networks consist of neurons (processing units) connected to each other by links. Each link is associated with weight, which characterizes the excitatory or inhibitory properties between neurons.

5.1.1. Input Signal. There are (m) input signals, where $(m-1)$, the excitation signal, at the input is $(x_1, x_2, \ldots, x_{m-1})$; they are taken from the output of neurons placed before this neuron or they are taken from other input signal sources. These input excitation signals are passed through a set of weights (w) that characterizes the degree of association between the front neurons is associated with it. A positive association weight corresponds to a restrained synapse. Particularly, the (m^{th}) input signal component (x_m) is called threshold with the value $x_m = +1$. The (x_m) signal is passed through the displacement component (bias) b_i: $w_m = b$.

5.1.2. Output of Neurons. The output of the neuron is given by the expression: $y = a$ (net) $= a(f)$, where a $(\cdot)$ is the symbol of the conversion function. In the conversion function, there is a document also called activation function or transfer function, which is responsible for converting the total weight (f) (or net) into an output signal (y).
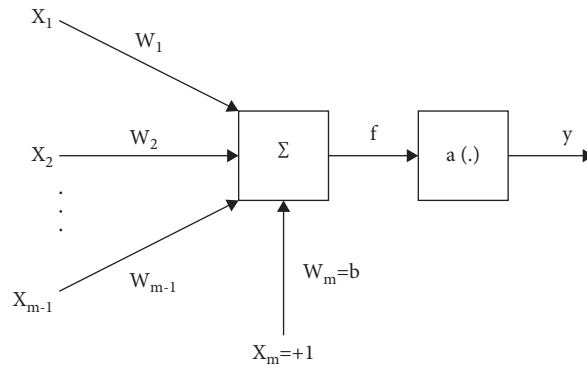
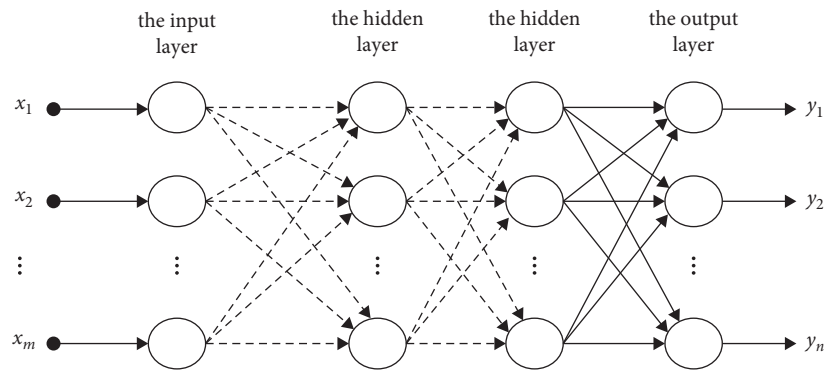FIGURE 2: Artificial neurons [27–31].

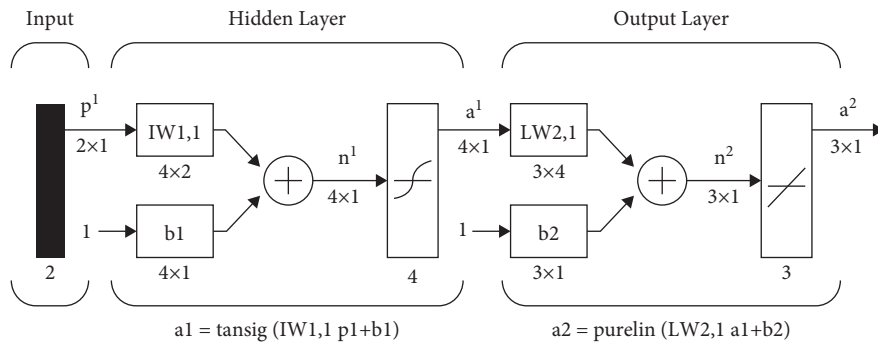FIGURE 3: Multilayer feedforward neural network [27–31].

FIGURE 4: Model of the multilayer feedforward neural network on Simulink [27–31].
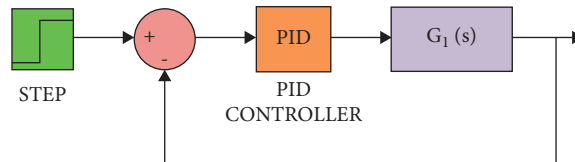

FIGURE 5: Simulink of the PID controller for transfer function $G_1(s)$.

5.2. Multilayer Feedforward Neural Network

5.2.1. Multilayer Feedforward Neural Network Architecture.
The multilayer feedforward network is a feedforward network with two or more layers of processing of neurons. The layer of neurons connected to the input is called the input layer (usually the input layer does not perform processing operations), the layer of neurons connected to the output is called the output layer, and the layer of neurons that is not directly connected to the input and the output is called the
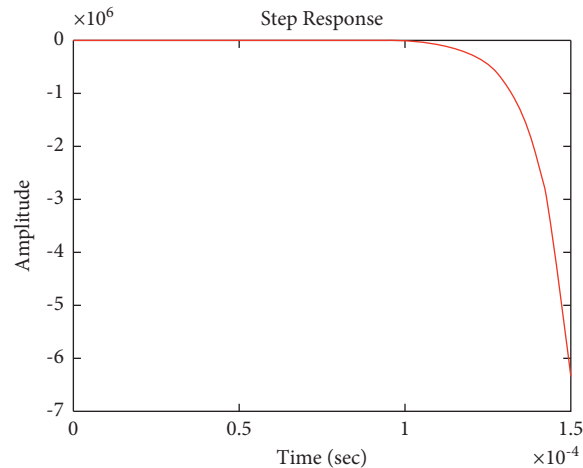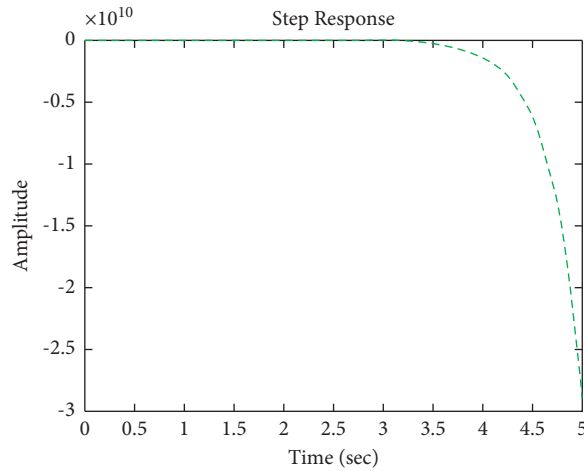
FIGURE 6: Step response of the closed form for transfer function $G_1(s)$.
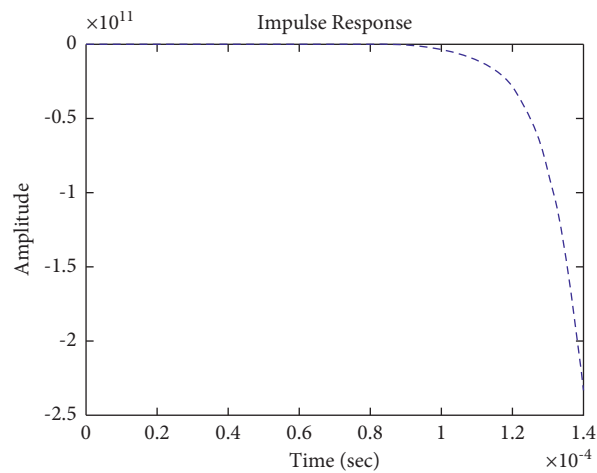
FIGURE 7: Impulse response of the closed form for transfer function $G_1(s)$.

hidden layer. Connections between neurons in layers can be complete or incomplete. The algorithm for training the multilayer feedforward network is a backpropagation algorithm.

5.2.2. Multilayer Feedforward Neural Network on Matlab. To set up the multilayer feedforward neural network below on Matlab, the author used the command "newff," which takes four input arguments, the first is the value "input" taken from the variable name "in" of "To workspace" in Simulink. This value indicates the range of the input variable, followed by arrays showing the number of neurons per layer: the first layer has fifty neurons, the output layer has one neuron, then the array showing the name of conversion functions used in each layer: the input layer conversion function is "tan-sigmoid," the output layer is layer linearity, and finally the name of the function used to train the network: the training function is "trainrp," and parameters of the network are initialized according to the preset algorithm. The command is described as follows: net = newff ([minmax (input)] (Figures 2–4), [50 1],{"tansig" "purelin"},"trainrp").

6. Simulation Results and Discussion

The diagram of the system using the PID controller and simulation results are shown Figures 5–32.

6.1. Part 1: Model Using PID Controller. Figure 7 shows the impulse response for the closed form (the signal is highlighted in blue) is worse than the open form (the signal is

FIGURE 8: Step response of the open form for transfer function $G_1(s)$.
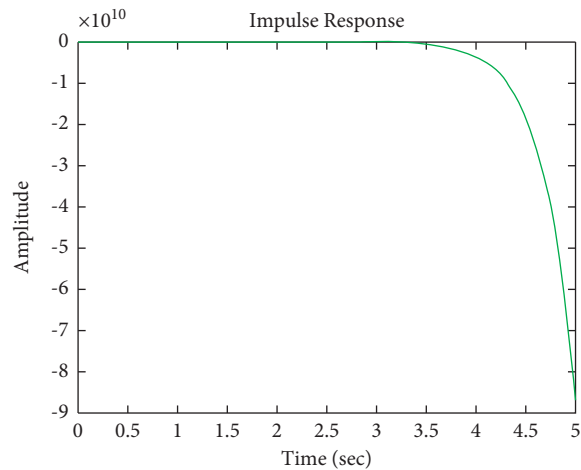

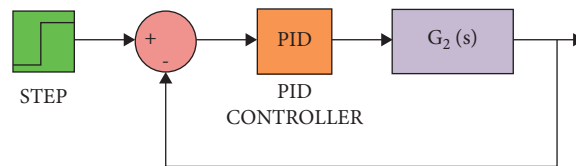FIGURE 9: Impulse response of the open form for transfer function $G_1(s)$.

FIGURE 10: Simulink of the PID controller for transfer function $G_2(s)$.

highlighted in green in Figure 9). The value of the amplitude of the closed form in this case is large, and the closed form does not reach a state of the steady. For the PID controller, the closed form does not respond well. The value of the amplitude of the open form in this case is large, and the open form does not reach a state of the steady. Figure 6 shows the step response for the closed form (the signal is highlighted in red) is better than that for the open form (the signal is highlighted in green in Figure 8). The value of amplitude of the closed form in this case is large, and the closed form does not reach a state of the steady. For the PID controller, the

closed form does not respond well. Meanwhile, the open form cannot respond well.

Figure 12 shows the impulse response for the closed form (the signal is highlighted in blue) is better than the open form (the signal is highlighted in green in Figure 14). The value of the amplitude of the closed form in this case is zero, and the closed form reaches a state of the steady. For the PID controller, the closed form responds well. The value of the amplitude of the open form in this case is large, and the open form does not reach a state of the steady. Figure 11 shows the step response for the closed form (the signal is highlighted in
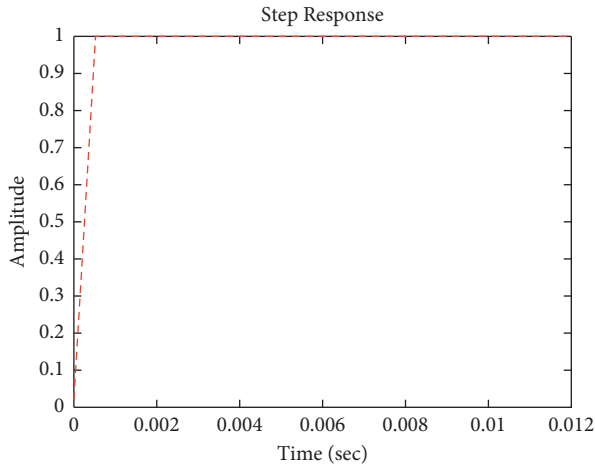
Figure 11: Step response of the closed form for transfer function $G_2(s)$.
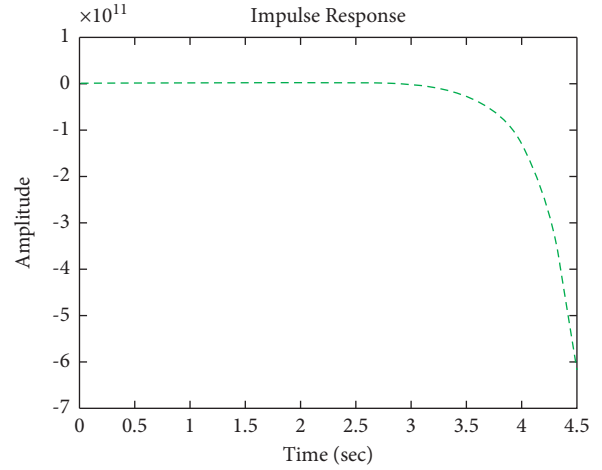

Figure 14: Impulse response of the open form for transfer function $G_2(s)$.
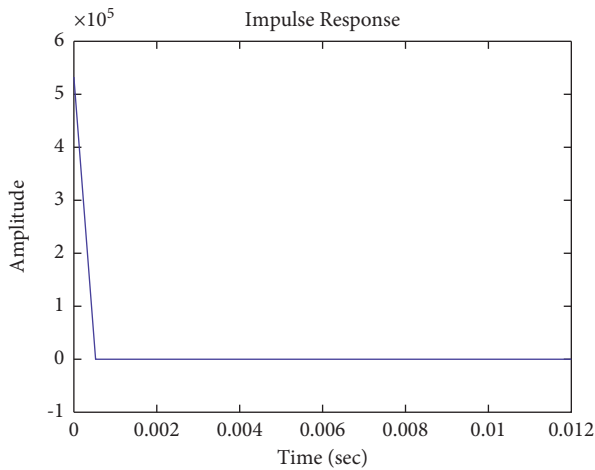

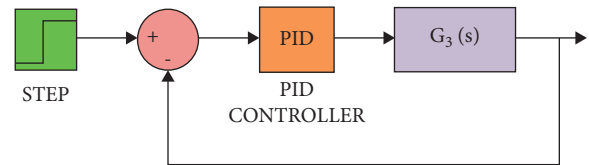Figure 12: Impulse response of the closed form for transfer function $G_2(s)$.

Figure 15: Simulink of the PID controller for transfer function $G_3(s)$.
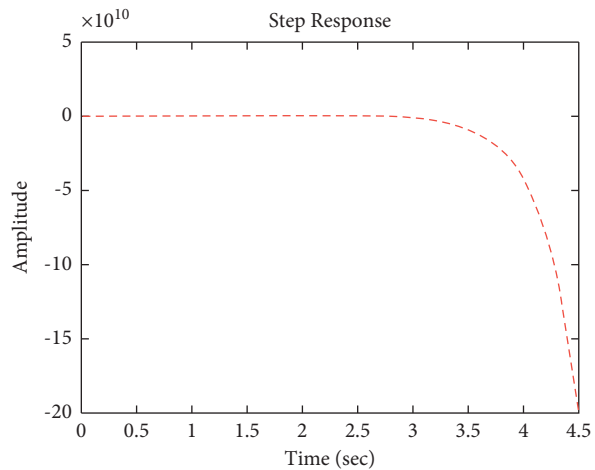

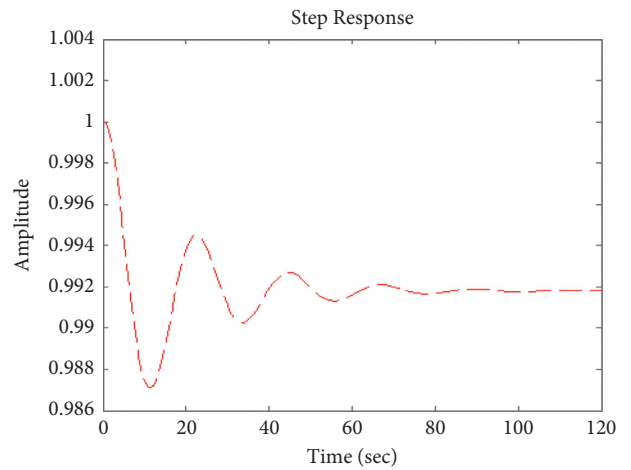Figure 16: Step response of the closed form for transfer function $G_3(s)$.

Figure 13: Step response of the open form for transfer function $G_2(s)$.

red) is better than that for the open form (the signal is highlighted in red in Figure 13). The value of amplitude of the closed form in this case is 1, and the closed form reaches a state of the steady. For the PID controller, the closed form responds well. Meanwhile, the open form cannot respond well.

Figure 17 shows the impulse response for the closed system (the signal is highlighted in green) is better than the
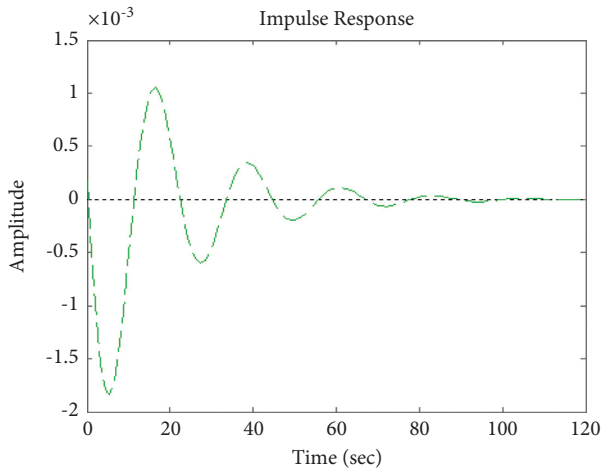
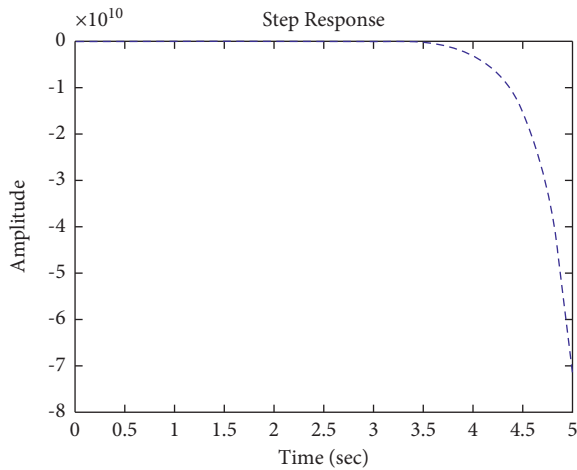FIGURE 17: Impulse response of the closed form for transfer function $G_3(s)$.

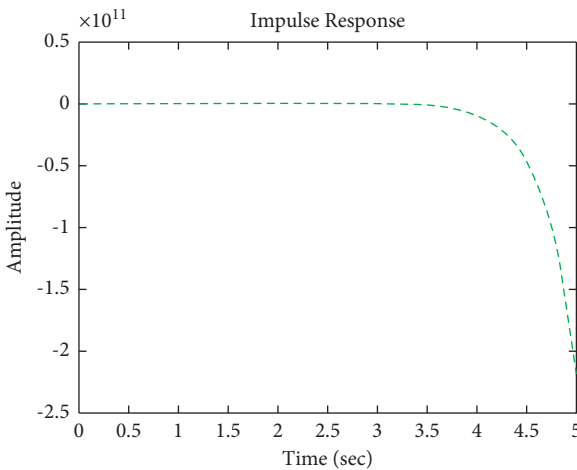FIGURE 18: Step response of the open form for transfer function $G_3(s)$.

FIGURE 19: Impulse response of the open form for transfer function $G_3(s)$.
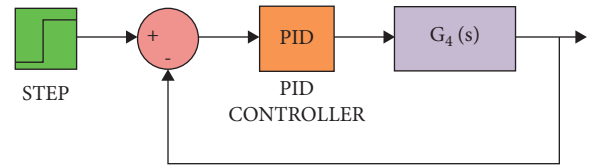

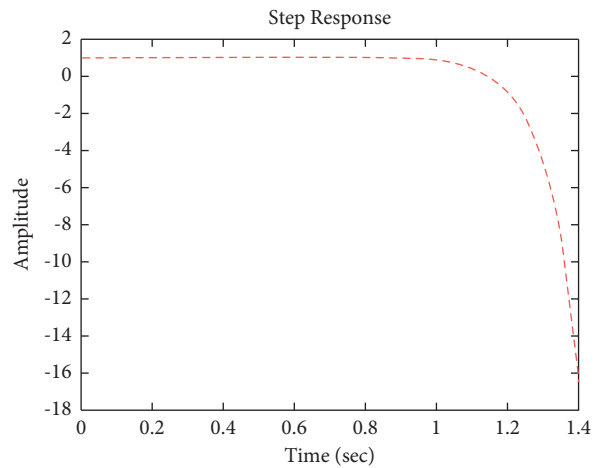FIGURE 20: Simulink of the PID controller for transfer function $G_4(s)$.

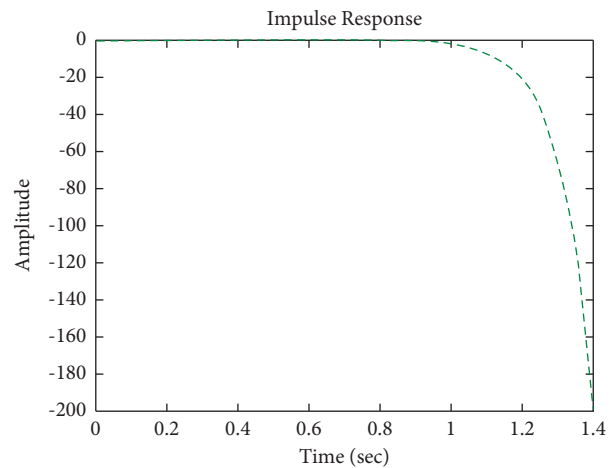FIGURE 21: Step response of the closed form for transfer function $G_4(s)$.

FIGURE 22: Impulse response of the closed form for transfer function $G_4(s)$.

open form (the signal is highlighted in green in Figure 19). The value of the closed form in this case is zero, and the closed form reaches a state of the steady. For the PID controller, the closed form responds well. The value of the amplitude of the open form in this case is large, and the open form does not reach a state of the steady. Figure 16 shows the step response for the closed form (the signal is highlighted in red) is better than that for the open form (the signal is highlighted in blue in Figure 18). The value of the amplitude of the closed form in this case is 0.992, and the closed form reaches a state of the
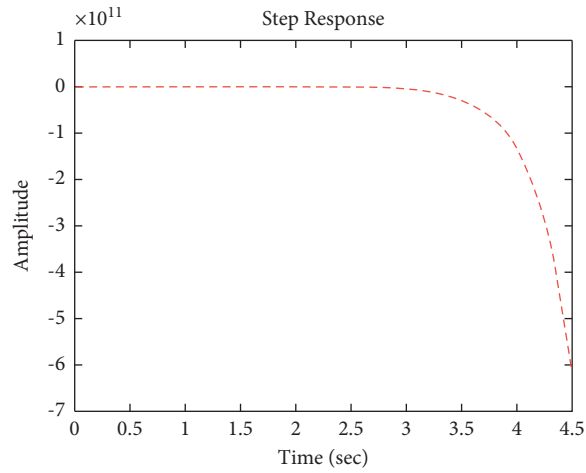
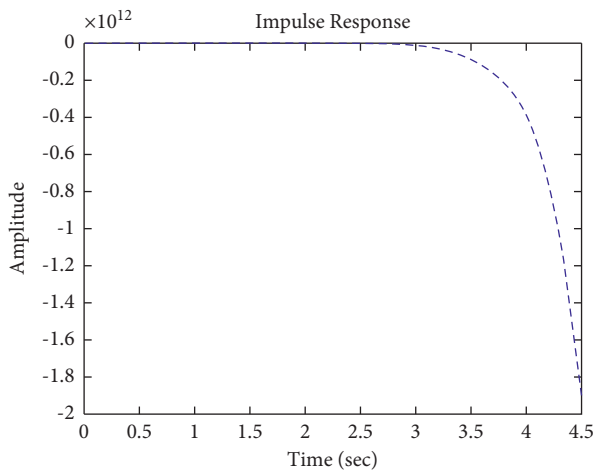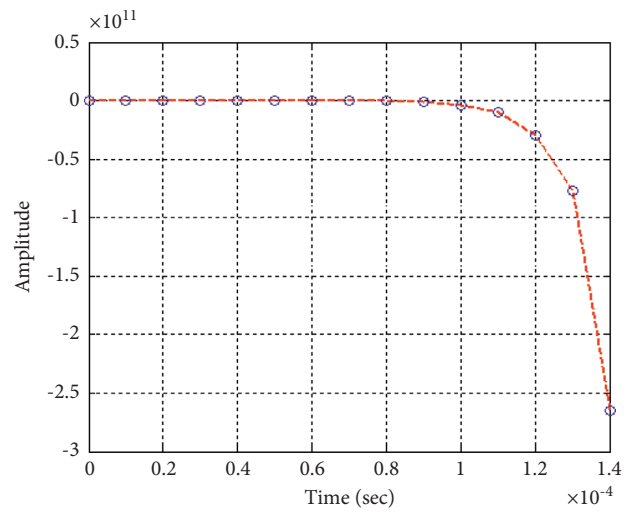FIGURE 23: Step response of the open form for transfer function $G_4(s)$.
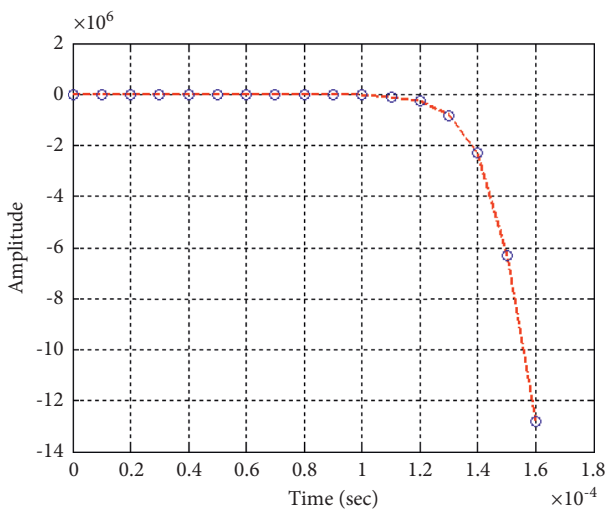

FIGURE 24: Impulse response of the open form for transfer function $G_4(s)$.
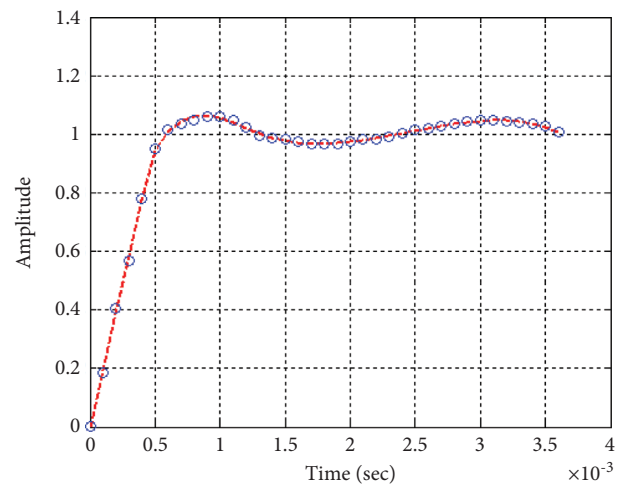

PID controller
Neural Networks Application

FIGURE 26: Impulse response of the closed form "$G_1(s)$".

PID controller
Neural Networks Application

FIGURE 25: Step response of the closed form "$G_1(s)$".

PID controller
Neural Networks Application
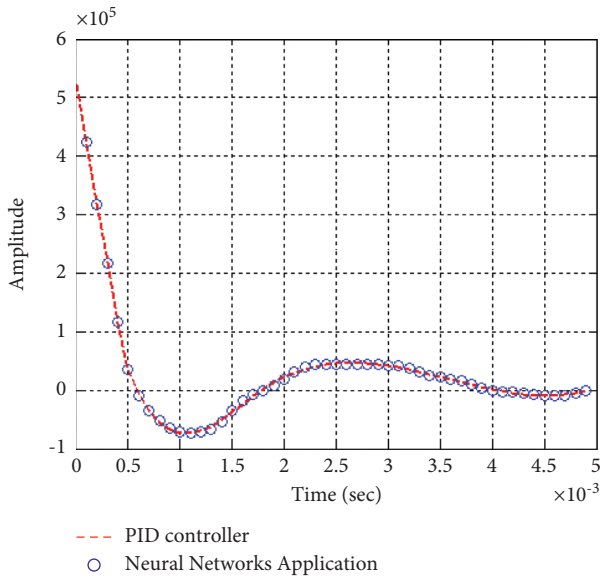
FIGURE 27: Step response of the closed form "$G_2(s)$".
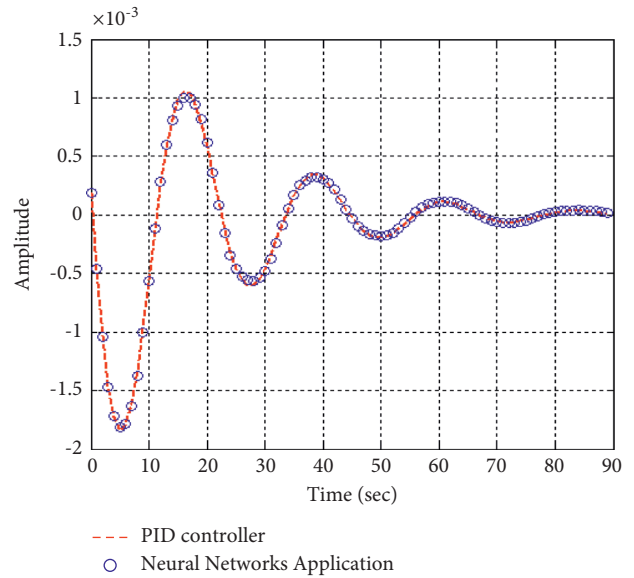
FIGURE 28: Impulse response of the closed form "$G_2(s)$".

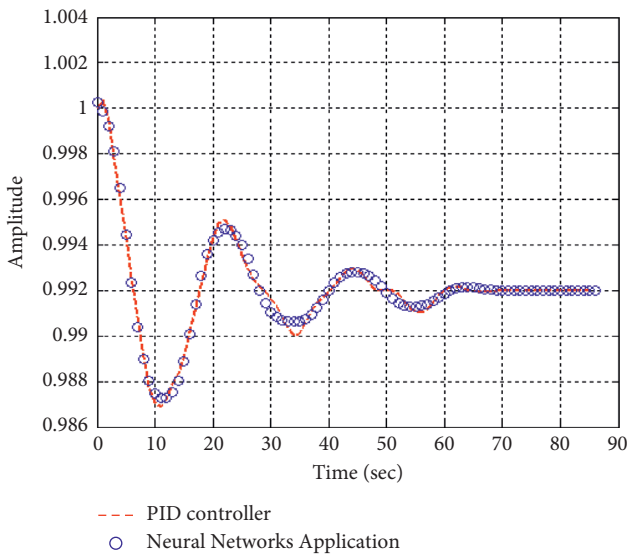FIGURE 30: Impulse response of the closed form "$G_3(s)$".

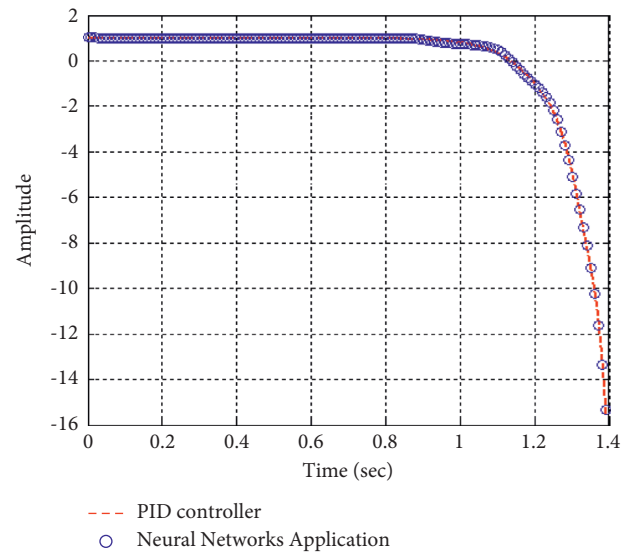
FIGURE 29: Step response of the closed form "$G_3(s)$".

FIGURE 31: Step response of the closed form "$G_4(s)$".

steady. For the PID controller, the closed form responds well. Meanwhile, the open form cannot respond well.

Figure 22 shows the impulse response for the closed form (the signal is highlighted in green) is better than the open form (the signal is highlighted in blue in Figure 24). The value of the amplitude of the closed form in this case is large, and the closed form does not reach a state of the steady. For the PID controller, the closed form does not respond well. The value of the amplitude of the open form in this case is large, and the open form does not reach a state of the steady. Figure 21 shows the step response for the closed form (the signal is highlighted in red) is better than that for the open form (the signal is highlighted in red in Figure 23). The value of the amplitude of the closed form in this case is large, and

the closed form does not reach a state of the steady. For the PID controller, the closed form does not respond well. Meanwhile, the open form cannot respond well.

6.2. Part 2: Model Using Neural Networks Application for the Data of PID Controller.
The neural network used in this case is based on the data collected from simulation results of the PID controller.

Figures 25–32 show that neural networks application for the data of the PID controller is the best choice for training the network according to a given control method. Results show that values of control methods are relatively consistent with each other, and results have achieved desired requirements.
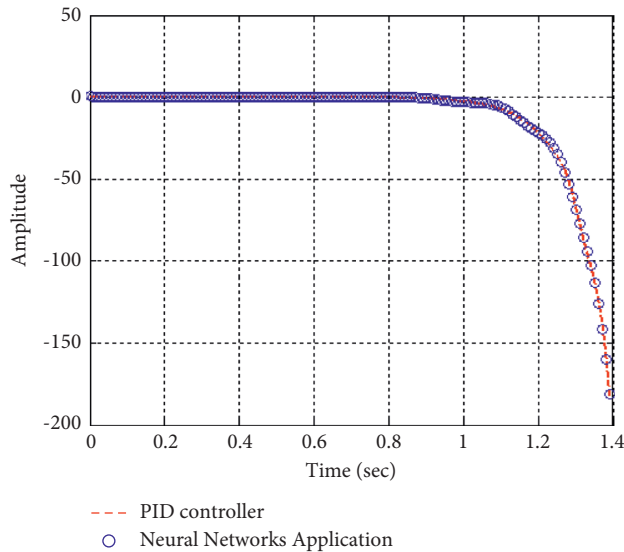
Figure 32: Impulse response of the closed form "$G_4(s)$".

7. Conclusions

To simplify and speed up the design process of control systems in general and Acrobot in particular, the author's choices for the control presented above are consistent with the strategies laid out. Simulinks for the design of closed forms are according to the design process. The analysis for the performance of the system was tested with the PID controller, neural networks application. The purpose of this study is for research and educational purposes. The analysis obtained from curves of 2 responses showed that the strategies for controlling a system were implemented on the simulation software carefully. Neural networks application can be a priority to be selected among above methods. Neural networks application has achieved satisfactory results in training the data of any controller, specifically here the PID controller. It has very good security according to plans outlined. In the process of implementing this topic, simulation results are stable for the system to be considered by the author. In the future, this system can be controlled by more modern algorithms such as the new modified multitracker optimization algorithm to suit more stringent requirements.

Data Availability

The data used to support the findings of this study are cited as references within the article.

Conflicts of Interest

The author declares that there are no conflicts of interest.

References

[1] P. V. Gopi Krishna Rao, M. V. Subramanyam, and K. Satyaprasad, "Performance comparison of PID controller tuned using classical and genetic algorithm methods," *International Journal of Applied Engineering Research*, vol. 6, no. 14, pp. 1757–1766, 2011.

[2] M. Papoutsidakis, D. Piromalis, F. Neri, and M. Camilleri, "Intelligent algorithms based on data processing for modular robotic vehicles control," *WSEAS Transactions on Systems*, vol. 13, 2014.

[3] H. T. Nguyen, M. T. Nguyen, V. D. H. Nguyen, T. T. Doan, and C. P. Vo, "Designing PID-fuzzy controller for pendubot system," *Robotica & Management*, vol. 22, no. 2, pp. 8–12, 2017.

[4] A. Ramm and M. Sjöstedt, *Reaction Wheel Balanced Robot, Design and Sensor Analysis of Inverted Pendulum Robot*, KTH Royal Institute of Technology, Stockholm, Sweden, 2015.

[5] A. Zhang, J. She, X. Lai, and M. Wu, "Motion planning and tracking control for an acrobot based on a rewinding approach," *Automatica*, vol. 49, no. 1, pp. 278–284, 2013.

[6] A.-W. A. Saif, "Strong stablization of the non linear Pendubot system," in *Proceedings of the 2015 IEEE 12th International Multi-Conference on Systems, Signals & Devices (SSD15)*, Mahdia, Tunisia, March 2015.

[7] V. T. Tran, T. T. Ha, and T. Vi Do, "Hybrid control for swing up and balancing pendubot system: an experimental result," in *Proceedings of the 2017 International Conference on System Science and Engineering (ICSSE)*, Ho Chi Minh City, Vietnam, July 2017.

[8] Xu. Luo and H. Wang, "A digital controller for the pendubot system using approximate output regulation approach," in *Proceedings of the 2010 2nd International Conference on Future Computer and Communication*, Wuhan, China, May 2010.

[9] Z. Huang and X. Lai, "Control strategy based on iterative method for planar pendubot," in *Proceedings of the 2018, 37th Chinese Control Conference (CCC)*, Wuhan, China, July 2018.

[10] T. Albahkali, R. Mukherjee, and T. Das, "Swing-up control of the pendubot: an impulse–momentum approach," *IEEE Transactions on Robotics*, vol. 25, 2009.

[11] P. Awootsopa, U. Sritheerawirojana, P. Sooraksa, and N. Komine, "Control of pendubot with chaotic perturbation," in *Proceedings of the 2006 1ST IEEE Conference on Industrial Electronics and Applications*, Singapore, May 2006.

[12] M. Zhang and T.-J. Tarn, "Hybrid control for the pendubot," in *Proceedings of the 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164)*, Seoul, Korea (South), May 2001.

[13] X. Su, Yu Yang, and B. Ge, "Application of ant colony algorithm in optimal control for the under-actuated system pendubot," in *Proceedings of the IEEE 2008 Chinese Control Conference (CCC)*, Kunming, China, July 2008.

[14] I. Fantoni, R. Lozano, and M. W. Spong, "Energy based control of the pendubot," *IEEE Transactions on Automatic Control*, vol. 45, 2000.

[15] T.-J. Tarn, M. Zhang, and F. Celani, "Hybrid control for the pendubot," in *Proceedings of the 2000 26th Annual Conference of the IEEE Industrial Electronics Society. IECON 2000. 2000 IEEE International Conference on Industrial Electronics, Control and Instrumentation.*, 21st Century Technologies, Seoul, Korea (South), May 2000.

[16] D. I. Arevalo and A. Hussain, "Robust control for stabilization of non-inertial system: pendulum-Ac-robot," in *Proceedings of the 2018 15th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE)*, Mexico City, Mexico, September 2018.

[17] Y. Luo, X. Lai, and M. Wu, "A positioning control strategy of planar Acrobot," in *Proceedings of the 30th Chinese Control Conference*, Yantai, China, July 2011.

[18] F. Xue, Z. Hou, and H. Deng, "Balance control for an acrobot," in *Proceedings of the 2011 Chinese Control and Decision Conference (CCDC)*, Mianyang, China, May 2011.

[19] N. Hannouda and H. Takami, "Control of acrobot using inverse linear quadratic method," in *Proceedings of the 2018 International Conference on Control, Automation and Diagnosis (ICCAD)*, Marrakech, Morocco, March 2018.

[20] M. Elsisi, H. G. Zaini, K. Mahmoud, S. Bergies, S. S. M. Ghoneim, and M. Ghoneim, "Improvement of trajectory tracking by robot manipulator based on a new cooperative optimization algorithm," *Mathematics*, vol. 9, no. 24, p. 3231, 2021.

[21] M. Elsisi and M. A. Ebrahim, "Optimal design of low computational burden model predictive control based on SSDA towards autonomous vehicle under vision dynamics," *International Journal of Intelligent Systems*, vol. 36, 2021.

[22] M. Elsisi, "Improved grey wolf optimizer based on opposition and quasi learning approaches for optimization: case study autonomous vehicle including vision system," *Artificial Intelligence Review*, Springer, Cham, Switzerland, 2022.

[23] M. Elsisi, "Optimal design of nonlinear model predictive controller based on new modified multitracker optimization algorithm," *International Journal of Intelligent Systems*, vol. 35, no. 6, 2020.

[24] M. Elsisi, "Optimal design of non-fragile PID controller," *Asian Journal of Control*, vol. 23, 2019.

[25] N. Cong Danh, "Adjusting LQG with noise signals, PID controller forAcrobot system," *Mathematical Problems in Engineering*, Hindawi, vol. 2021, Article ID 4164382, 2021.

[26] R. D. Doncker, D. W. J. Pulle, and A. Veltman, *Advanced Electrical Drivers: Analysis, Modeling, Control*, Springer, Berlin, Germany, 2011.

[27] S. Walczak and N. Cerpa, *Artificial Neural Networks*, IEEE, Piscataway, NJ, USA, 3rd edition, 2003.

[28] N. Jin and D. Liu, "Wavelet basis function neural networks for sequential learning,"*IEEE Transactions on Neural Networks*, vol. 19, no. 3, IEEE, 2008.

[29] F. Faggin, "Neural network hardware," in *Proceedings of the IJCNN International Joint Conference on Neural Networks*, vol. 1, IEEE, Baltimore, MD, USA, June 1992.

[30] Z. Miao and B. Yuan, "An extended BAM Neural network model," in *Proceedings of the 1993 International Conference on Neural Networks (IJCNN-93-Nagoya, Japan)*, vol. 3, IEEE, Nagoya, Japan, October 1993.

[31] G. Costantini, D. Casali, and R. Perfetti, "Associative memory design for 256 gray-level images using a multilayer neural network," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 17, IEEE, 2006.