Article

Utilisation of Initialised Observation Scheme for Multi-Joint Robotic Arm in Lyapunov-Based Adaptive Control Strategy

Mohammad Soleimani Amiri [ID] and Rizauddin Ramli *[ID]

Department of Mechanical and Manufacturing Engineering, Faculty of Engineering and Built Environment, Universiti Kebangsaan Malaysia, Bangi 43600, Selangor, Malaysia
* Correspondence: rizauddin@ukm.edu.my

Abstract: In this paper, we present a modelling, dynamic analysis, and controller tuning comparison for a five-degree-of-freedom (DoF) multi-joint robotic arm based on the Lyapunov-based Adaptive Controller (LAC). In most pick-and-place applications of robotic arms, it is essential to control the end-effector trajectory to reach a precise target position. The kinematic solution of the 5-DoF robotic arm has been determined by the Lagrangian technique, and the mathematical model of each joint has been obtained in the range of motion condition. The Proportional-Integral-Derivative (PID) control parameters of the LAC have been determined by the Lyapunov stability approach and are initialised by four observation methods based on the obtained transfer function. The effectiveness of the initialised controller's parameters is compared by a unit step response as the desired input of the controller system. As a result, the average error (AE) for Ziegler–Nichols is 6.6%, 83%, and 53% lower than for Pettit & Carr, Chau, and Bucz. The performance of LAC for the robotic arm model is validated in a virtual 3D model under a robot operating system environment. The results of root mean square error by LAC are 0.021 (rad) and 0.025 (rad) for joint 1 and joint 2, respectively, which indicate the efficiency of the proposed LAC strategy in reaching the predetermined trajectory and the potential of minimizing the controller tuning complexity.

Keywords: multi-joint robotic arm; Proportional-Integral-Derivative; controller tuning; Lyapunov approach

MSC: 93C40

1. Introduction

The advancements in robotic and autonomous systems involve various types of robots in our daily lives and in industry [1]. Therefore, the multi-joint robotic arm represents an essential role in the automotive, agriculture, and bio-medical sectors because of its satisfactory performance, flexibility, and accuracy [2–4].

The robotic arm is one of the most common types of robots that is used in several industrial applications [5]. For example, Xie et al. [6] developed an obstacle avoidance and path planning algorithm for a multi-joint manipulator equipped with a spacecraft based on forward and backward inverse kinematics. Pavlovcic et al. [7] utilised a six-degree-of-freedom (DoF) robotic arm for simultaneous laser profilometry and hand–eye calibration in an industrial application. In another study, Jeong et al. [8] presented brain–machine interfaces for robotic arm applications. They developed an electroencephalogram, worn by humans to acquire signals for implementation as desire tracking for a robotic arm.

Proportional-Integral-Derivative (PID) is one of the classical controllers, and it has been widely used in different industries due to its simplicity, flexibility, adequate results [9,10], ease of implementation, and excellent performance [11]. In order to increase the precision and robustness of the controller, its parameters are tuned by various methods, such as classical observation and optimisation techniques [12,13]. Belkadi et al. [14] presented a

swarm optimisation to tune the parameters of the controller by minimizing the steady-state error. They verified their controller in a simulation model and compared it with conventional methods by numerical analysis. Phu et al. [15] combined optimisation with sliding mode control based on the Bolza–Meyer criterion to increase the performance of the controller by minimising and eliminating disturbances such as the vibration effect. Another example of the usage of a PID controller in different applications is the study by Wang et al. [16] that developed an optimal controller for vibration active control systems. In another study, Suhaimin et al. [17] used a PID controller for a 5-DoF robotic arm and controlled its joints for point-to-point trajectory tracking of the end-effector.

Adaptive control systems have been used in various works for different applications [18,19]. Pezzato et al. [20] presented an active inference controller that used an adaptive controller for industrial robots. They compared their proposed controller with a state-of-the-art model reference adaptive controller for an experimental setup of a 7-DoF robotic arm for the pick-and-place cycle. Tavoosi et al. [21] introduced an adjusting PID parameter through model predictive control. The parameters of the PID are tuned in real time by type-2 fuzzy-logic systems for a tank reactor application. Ma et al. [22] presented a trajectory regulating model reference adaptive controller to increase the stability and robustness of adaptation for the joint trajectory of a robotic arm. In our work, the PID controller's parameters are tuned by a Lyapunov-based Adaptive Controller (LAC) and are initialised by observation methods. In general, the contributions of this paper are as follows:

- We tuned the controller parameters in real time based on stability analysis of the non-linear control system;
- The LAC's parameters are initialised by using various observation methods and they are used to initialise the controller's parameters.

The contributions are validated in a 5-DoF robotic arm due to its serial structure, flexible action, accessible operation, and satisfactory load capacity in virtual simulation; it is extensively used in many robotic fields and industries [23].

The rest of the paper is organised as follows: Section 2 addresses the dynamic models of the 5-DoF robotic arm, developed using the Lagrangian method. Section 3 presents the LAC strategy determination. The results and discussion are presented in Section 4. The conclusions are given in Section 5.

2. Structure and Dynamics of the Robotic Arm

In this paper, a 5-DoF robotic arm, consisting of a base, four links, a wrist, and grippers, is selected as an example of a serial manipulator. The links are connected to each other by joints, which are powered by a motor and encoder sensor to move and capture the angular trajectory. The robotic arm is composed of two rotating and three swinging joints, among which the first and fifth joints are rotating and joints 2, 3, and 4 are swinging. Figure 1 shows the structure of the robotic arm.
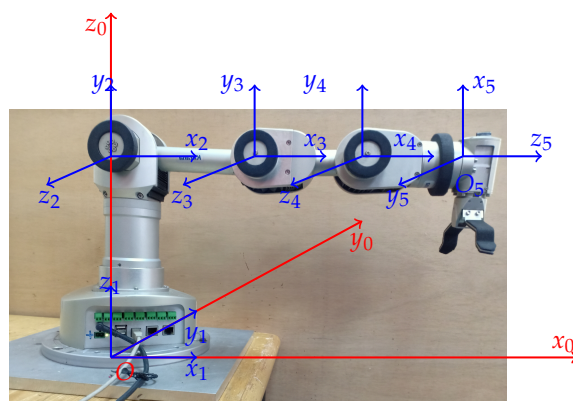

Figure 1. Structure of the 5-DoF robotic arm.

In this work, we employ the energy-based Lagrangian method to establish the relation between the torque and angle of joints [24]. The energy-based equation is expressed as follows:

$$L = E_k - E_p \tag{1}$$

$$\tau_i = \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_i}\right) - \left(\frac{\partial L}{\partial \theta_i}\right) + B_i(\dot{\theta}_i) \tag{2}$$

where L represents the Lagrangian function; E_k and E_p are the total kinetic and potential energies, respectively; B_i is the joint friction coefficient; τ_i is the torque of each joint, with $i = 1, 2, 3, 4, 5$; θ_i and $\dot{\theta}_i$ are the angular trajectory and velocity, respectively. The equations of E_p and E_k are given as follows [25]:

$$E_p = \sum_{i=1}^{5} m_i g z_{di} \tag{3}$$

$$E_k = \sum_{i=1}^{5} \left[\frac{1}{2} m_i (\dot{x}_{di}^2 + \dot{y}_{di}^2 + \dot{z}_{di}^2) + \frac{1}{2} I_i \dot{\theta}_i^2\right] \tag{4}$$

g is the gravity acceleration; m_i and I_i are the mass and inertia of each link, respectively; and $(\dot{x}_{di}, \dot{y}_{di}, \dot{z}_{di})$ is the time derivative of the centroid position of each linkage. According to the geometric relation, the centroid position (x_{di}, y_{di}, z_{di}) of every linkage is represented as

$$X_{d_i} = \sum_{j=1}^{i-1} (R_{z_j}^j X) + R_{z_i}^i X_d \tag{5}$$

Here, $^j X \in \Re^{3 \times 1}$ is the position of joint $(i-1)_{th}$ according to the reference frame; $X_{d_i} \in \Re^{3 \times 1}$ is the position of the centroid point of link i_{th} relative to the reference frame; and $^i X_d \in \Re^{3 \times 1}$ represents the centroid position of the link i_{th} regarding the coordinate system located in the joint i_{th}. Moreover, $R_{z_i} \in \Re^3$ is the rotation matrix around the z-axis according to the coordinate system placed in the i_{th} joint, given as follows:

$$R_{z_i} = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) & 0 \\ \sin(\theta_i) & \cos(\theta_i) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{6}$$

By substituting E_k and E_p into the Lagrangian function, the dynamic equation is determined as follows:

$$\tau = M\ddot{\theta} + V(\dot{\theta}, \theta) + G(\theta) \tag{7}$$

where $\tau \in \Re^{5 \times 1}$ represents the torque vector; $\theta \in \Re^{5 \times 1}$ and $\ddot{\theta} \in \Re^{5 \times 1}$ are the angular rotation and acceleration; $V \in \Re^5$ is the centrifugal, coriolis and friction matrix, and $G(\theta) \in \Re^5$ represents the gravity matrix, expressed as follows:

$$V(\dot{\theta}, \theta) = B_i \cdot I_{5 \times 5} \tag{8}$$

$$G(\theta) = e_{g_i} \cdot I_{5 \times 5} \tag{9}$$

where $I_5 \in \Re^5$ is the identity matrix and e_{g_i} denotes the elements of mass and gravity matrices, represented as follows:

$$e_{g_i} = (l_i \sum_{i=i+1}^{5} (m_i) + l_{c_i} m_i) g \sin(\theta_i) \tag{10}$$

where g represents gravitational acceleration. $M \in \Re^5$ is a matrix containing mass and inertia elements, which is shown as follows:

$$
M = \begin{bmatrix}
e_{m_1} & 0 & 0 & 0 & 0 \\
0 & e_{m_2} & 0 & 0 & 0 \\
0 & 0 & e_{m_3} & 0 & 0 \\
0 & 0 & 0 & e_{m_4} & 0 \\
0 & 0 & 0 & 0 & e_{m_5}
\end{bmatrix} \tag{11}
$$

where $e_{m_i} \quad i = 1, 2, 3, 4, 5$ represent the mass and inertia elements, expressed as follows:

$$
e_{m_1} = I_1; \quad e_{m_i} = l_i^2 \sum_{i=i+1}^{5} (m_i) + I_i + m_i l_{c_i}^2 \tag{12}
$$

where l_{c_i} is the length of the centroid position for each link and l_i is the length of the links. Table 1 shows the physical characteristics of the robotic arm's links.

Table 1. Characteristics of 5-DoF robotic arm.

Link	$l_i(\mathbf{m})$	$l_{c_i}(\mathbf{m})$	$m_i(\mathbf{kg})$	I_i	B_i
$i = 1$	0.3	0.15	0.748	0.0013	0.72
$i = 2$	0.19	0.095	0.8020	0.0043	0.83
$i = 3$	0.14	0.07	0.792	0.0023	0.95
$i = 4$	0.15	0.075	0.691	0.0015	0.88
$i = 5$	0.04	0.02	0.2562	0.0012	0.83

The mathematical model of the robotic arm has non-linear characteristics. Thus, to simplify the model for further analysis, methods such as the range of motion (RoM) condition, where one joint is moving and the other ones are fixed, have been applied. The transfer functions of each joint are represented as follows:

$$
G_1 = \frac{\theta_{a1}}{\tau_1} = \frac{1}{I_1 s^2 + B_1 s} \tag{13}
$$

$$
G_2 = \frac{\theta_{a2}}{\tau_2} = \frac{1}{a_{21} s^2 + B_2 s + a_{23}} \tag{14}
$$

$$
G_3 = \frac{\theta_{a3}}{\tau_3} = \frac{1}{a_{31} s^2 + B_3 s + a_{33}} \tag{15}
$$

$$
G_4 = \frac{\theta_{a4}}{\tau_4} = \frac{1}{a_{41} s^2 + B_4 s + a_{43}} \tag{16}
$$

$$
G_5 = \frac{\theta_{a5}}{\tau_5} = \frac{1}{a51 s^2 + B_5 s + a_{53}} \tag{17}
$$

where

$$
a_{21} = l_2^2 m_5 + l_2^2 m_4 + l_2^2 m_3 + l_{c_2}^2 m_2 + I_2 \tag{18}
$$

$$
a_{23} = g l_2 m_4 + g l_2 m_3 + l_{c_2} g m_2 \tag{19}
$$

$$
a_{31} = l_3^2 m_5 + l_3^2 m_4 + l_{c_3}^2 m_3 + I_3 \tag{20}
$$

$$
a_{33} = g l_3 m_5 + g l_3 m_4 + l_{c_3} g m_3 \tag{21}
$$

$$
a_{41} = l_4^2 m_5 + l_{c_4}^2 m_4 + I_4 \tag{22}
$$

$$
a_{43} = g l_4 m_5 + l_{c_4} g m_4 \tag{23}
$$

$$
a_{51} l_{c_5}^2 m_5 + I_5 \tag{24}
$$

$$
a_{53} = l_{c_5} g m_5 \tag{25}
$$

3. LAC Strategy for Robotic Arm Joints

PID control has been used by many researchers to move the joints' trajectory in the robotic arm, because of its potential in providing acceptable results and ease of operation [26,27]. The input and output of the control system are the desired and actual trajectory angle of each joint, respectively. In this paper, the PID controller's parameters are determined by stability analysis and initialised by observation tuning methods, i.e., Ziegler–Nichols (Z-N), Pettit & Carr, Chau, and Bucz [28,29].

Figure 2 shows the block diagram of the PID closed-loop control system, in which the parameters of the controller are tuned by a combination of observation methods and LAC based on the steady-state error. The process of observation methods is carried out by the mathematical model of the robotic arm and LAC is the real-time tuner of the controller.
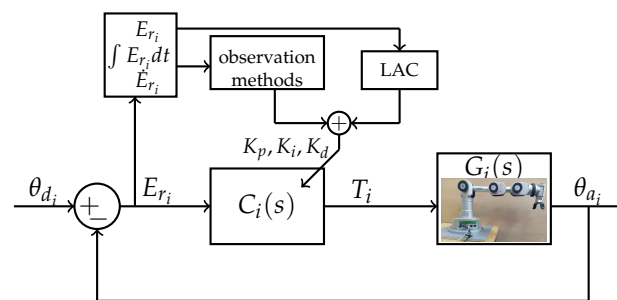

Figure 2. Block diagram of LAC strategy.

The steady-state error is the difference between the desired and actual trajectory, given as follows:

$$E_{r_i} = \theta_{d_i} - \theta_{a_i} \qquad i = 1, 2, 3, 4, 5 \tag{26}$$

where $\theta_{a_i}(s)$ and $\theta_{d_i}(s)$ represent the actual and desired angular trajectories for each joint, respectively.

Lyapunov-Based Adaptive Controller

The controller law for the Lyapunov-based adaptive controller is defined as follows:

$$u = K_p E_r + K_i \int_0^t E_r dt + K_d \frac{dE_r}{dt} \tag{27}$$

where it is assumed that $\dot{Z} \triangleq E_r$; therefore, $Z \triangleq \int_0^t E_r dt$. Equation (27) is rewritten as follows:

$$u = K_p E_r + K_i Z + K_d \dot{E}_r \tag{28}$$

Theorem 1. *Consider a general non-linear dynamic system as follows:*

$$\dot{X} = -AX + Bu \tag{29}$$

$$y = f(X) \tag{30}$$

where $X \in \Re^n$ is the state vector, $y \in \Re^n$ is the controller output, u is the control input vector. $f : \Re^n \to \Re^n$ is a Lipschitz function. Consider a function $V : \Re^n \to [0, \infty]$, which is positive definite, denoted by V_x. Therefore, there is a Lyapunov function for the non-linear dynamic system, if the following expression is valid.

$$\dot{V} := V_x(x) \cdot f(x) < 0 \quad \forall x \neq 0 \tag{31}$$

Proof. The non-linear dynamic system can be shown to be asymptotically stable if such a Lyapunov function exists [30–33]. It is assumed that for a given positive-definite matrix Q, there is a positive-definite solution P such that $PA + A^T P = -Q$, where P is the

positive-definite matrix that is used to simplify the algebra without any loss of generality. The Lyapunov candidate function is defined as follows:

$$V = X^T P X + \tilde{K}_p^T \Gamma^{-1} \tilde{K}_p + \tilde{K}_i^T \Gamma^{-1} \tilde{K}_i + \tilde{K}_d^T \Gamma^{-1} \tilde{K}_d \tag{32}$$

where Γ is the adaptive rate, and $\tilde{K}_p$, $\tilde{K}_i$, and $\tilde{K}_d$ are the differences between the actual and estimated controller parameters, $\hat{K}_p$, $\hat{K}_i$, and $\hat{K}_d$ respectively, given as follows:

$$\tilde{K}_p = K_p - \hat{K}_p \tag{33}$$

$$\tilde{K}_p = K_i - \hat{K}_i \tag{34}$$

$$\tilde{K}_p = K_d - \hat{K}_d \tag{35}$$

Differentiating the Lyapunov candidate function over time, we have

$$\dot{V} = \dot{X}^T P X + X^T P \dot{X} + \dot{K}_p^T \Gamma^{-1} \tilde{K}_p + \tilde{K}_p^T \Gamma^{-1} \dot{K}_p$$
$$+ \dot{K}_i^T \Gamma^{-1} \tilde{K}_i + \tilde{K}_i^T \Gamma^{-1} \dot{K}_i + \dot{K}_d^T \Gamma^{-1} \tilde{K}_d + \tilde{K}_d^T \Gamma^{-1} \dot{K}_d \tag{36}$$

where $\dot{X}^T$ is the transpose of Equation (29) and $\dot{K}_p^T \Gamma^{-1} \tilde{K}_p = \tilde{K}_p^T \Gamma^{-1} \dot{K}_p$, $\dot{K}_i^T \Gamma^{-1} \tilde{K}_i = \tilde{K}_i^T \Gamma^{-1} \dot{K}_i$, and $\dot{K}_d^T \Gamma^{-1} \tilde{K}_d = \tilde{K}_d^T \Gamma^{-1} \dot{K}_d$. Therefore, the derivative of the Lyapunov candidate function is written as follows:

$$\dot{V} = X^T A^T P X + B^T U P X + X^T P A X + X^T P B U$$
$$+ 2\dot{K}_p^T \Gamma^{-1} \tilde{K}_p + 2\dot{K}_i^T \Gamma^{-1} \tilde{K}_i + 2\dot{K}_d^T \Gamma^{-1} \tilde{K}_d \tag{37}$$

$$\dot{V} = X^T A^T P X + B^T u P X + X^T P A X + X^T P B u$$
$$+ 2\dot{K}_p^T \Gamma^{-1} \tilde{K}_p + 2\dot{K}_i^T \Gamma^{-1} \tilde{K}_i + 2\dot{K}_d^T \Gamma^{-1} \tilde{K}_d \tag{38}$$

By substituting Equation (28) in Equation (38), the derivative of the Lyapunov function is given as follows:

$$\dot{V} = -X^T Q X + \tilde{K}_p (B^T P X E_r + X^T P B E_r + 2\dot{K}_p^T \Gamma^{-1} \tilde{K}_p)$$

$$+ \tilde{K}_i (B^T P X Z + X^T P B Z + 2\dot{K}_i^T \Gamma^{-1} \tilde{K}_i)$$

$$+ \tilde{K}_d (B^T P X \dot{E}_r + X^T P B \dot{E}_r + 2\dot{K}_d^T \Gamma^{-1} \tilde{K}_d). \tag{39}$$

Therefore, $\dot{K}_p$, $\dot{K}_i$, and $\dot{K}_d$ are expressed as follows, to have $\dot{V} = -X^T Q X$, which is negative-definite and satisfies Theorem 1.

$$\dot{\tilde{K}}_p = \dot{K}_p = \frac{-\Gamma B^T P X E_r - \Gamma X^T P B E_r}{2} \tag{40}$$

$$\dot{\tilde{K}}_i = \dot{K}_i = \frac{-\Gamma B^T P X Z - \Gamma X^T P B Z}{2} \tag{41}$$

$$\dot{\tilde{K}}_d = \dot{K}_d = \frac{-\Gamma B^T P X \dot{E}_r - \Gamma X^T P B \dot{E}_r}{2} \tag{42}$$

$\square$

Therefore,

$$K_p = \int_0^t \dot{K}_p dt + \acute{K}_p \tag{43}$$

$$K_i = \int_0^t \dot{K}_i dt + \acute{K}_i \tag{44}$$

$$K_d = \int_0^t \acute{K}_d dt + \acute{K}_d \tag{45}$$

where $\acute{K}_p$, $\acute{K}_i$, and $\acute{K}_d$ represent the initial values of PID parameters that are determined by conventional observation methods. The closed-loop transfer function of the control system is given as

$$\frac{\theta_{a_i}(s)}{\theta_{d_i}(s)} = \frac{C(s)G_i(s)}{1 + C(s)G_i(s)} \tag{46}$$

where $C(s)$ is the transfer function of the controller, and $G_i(s)$ is the transfer function of each link. The controller $C(s)$ is given as follows:

$$C(s) = \acute{K}_p + \acute{K}_i \frac{1}{s} + \acute{K}_d s \tag{47}$$

There are various PID tuning methods, such as pole assignment, Cohen–Coon, amplitude, and phase margin methods. In this study, the parameters of PID are tuned by conventional observation methods of Z-N, Pettit & Carr, Chau, and Bucz. For tuning the controller by conventional methods, the values of $\acute{K}_i$ and $\acute{K}_d$ are set as zero to convert the PID into a proportional controller. Initially, the gain of the proportional controller is tuned. Subsequently, $\acute{K}_i$ and $\acute{K}_d$ are obtained based on the tuned proportional gain, and the proportional controller is established as follows:

$$G_p(s) = \frac{K_u G(s)}{1 + K_u G(s)} \tag{48}$$

where K_u is the proportional gain, which increases gradually until the output of the control system starts to oscillate by equal amplitude. This parameter is the ultimate gain for K_u and its period is selected as the oscillation period, T_u. Figure 3 represents the unit step response to the proportional controller for joint 3.
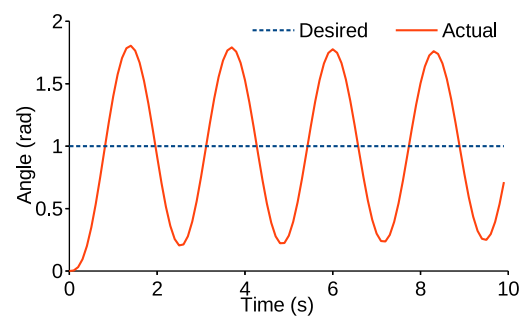

Figure 3. Unit step response for proportional controller for joint 3, in which $T_u = 1.5$ s and $K_u = 7$.

The tuning of controller parameters with various conventional methods is shown in Table 2. Although, in the observation methods, there is a lack of selection of parameters and excessive overshoot in the time response, they are still well-known and convenient to use. These methods are adopted to minimise the rise time, overshoot, and settling time.

Table 2. Conventional observation methods for tuning PID.

	$\acute{K}_p$	$\acute{K}_i$	$\acute{K}_d$
Z-N	$0.6K_u$	$\frac{1.2K_u}{T_u}$	$0.075K_u T_u$
Pettit & Carr	$0.67K_u$	$\frac{0.67K_u}{T_u}$	$0.1002K_u Tu$
Chau	$0.2K_u$	$\frac{0.3636K_u}{T_u}$	$0.066K_u T_u$
Bucz	$0.28K_u$	$\frac{0.1944K_u}{T_u}$	$0.1005T_u K_u$

Algorithm 1 expresses the overall pseudo-code of the LAC strategy. K_s is the proportional step to increase K_u gradually until the ultimate gain is obtained.

Algorithm 1 Pseudo-code of LAC.

1: Start;
2: Determine $\acute{K}_p$, $\acute{K}_i$, and $\acute{K}_d$;
3: Set $\acute{K}_i = 0$, $\acute{K}_d = 0$, and $\acute{K}_p = K_u$;
4: Initialise K_u;
5: Set step value for as K_s;
6: **while** Observe stable oscillation **do**
7:     Set value for K_u;
8:     $K_u = K_u + K_s$;
9: **end while**
10: Set $K_u = K_{cr}$;
11: Calculate oscillation period as T_u ;
12: Calculate $\acute{K}_p$, $\acute{K}_i$, and $\acute{K}_d$;
13: Set $K_p = \int_0^t \dot{K}_p dt + \acute{K}_p$, $K_i = \int_0^t \dot{K}_i dt + \acute{K}_i$, and $K_d = \int_0^t \dot{K}_d dt + \acute{K}_d$
14: End.

The ultimate gains, K_u, for joints 1 to 5, are 1, 44.5, 7, 66, and 25, respectively. Similarly, T_u is 0.003, 0.5, 1.5, 0.3, and 0.6 for joint 1 to 5, respectively. Table 3 illustrates the PID parameters tuned for each joint.

Table 3. PID parameters tuned by conventional observation methods.

		Joint 1	Joint 2	Joint 3	Joint 4	Joint 5
	$\acute{K}_p$	0.6	26.7	4.2	39.6	15
Z-N	$\acute{K}_i$	400	106.8	5.6	264	100
	$\acute{K}_d$	0.0002	1.66	0.78	1.48	0.56
Pettit	$\acute{K}_p$	0.67	29.82	4.69	44.22	16.75
&	$\acute{K}_i$	223.33	59.63	3.12	147.4	55.84
Carr	$\acute{K}_d$	0.0003	2.22	1.05	1.98	0.75
	$\acute{K}_p$	0.2	8.9	1.4	13.2	5
Chau	$\acute{K}_i$	121.19	32.36	1.7	79.92	30.3
	$\acute{K}_d$	0.0002	1.48	0.69	1.31	0.5
	$\acute{K}_p$	0.28	12.46	1.97	18.48	7
Bucz	$\acute{K}_i$	64.8	17.3	1.903	42.8	16.2
	$\acute{K}_d$	0.0003	2.23	1.05	1.2	0.75

4. Results and Discussion

We compare the performance of the tuned parameters by the conventional observation methods between Z-N, Pettit & Carr, Chau, and Bucz, in which the determined parameters are implemented in a closed-loop control system for each joint transfer function. Figure 4 compares the unit step response results of each joint in the closed-loop control system based on the transfer function of each link.
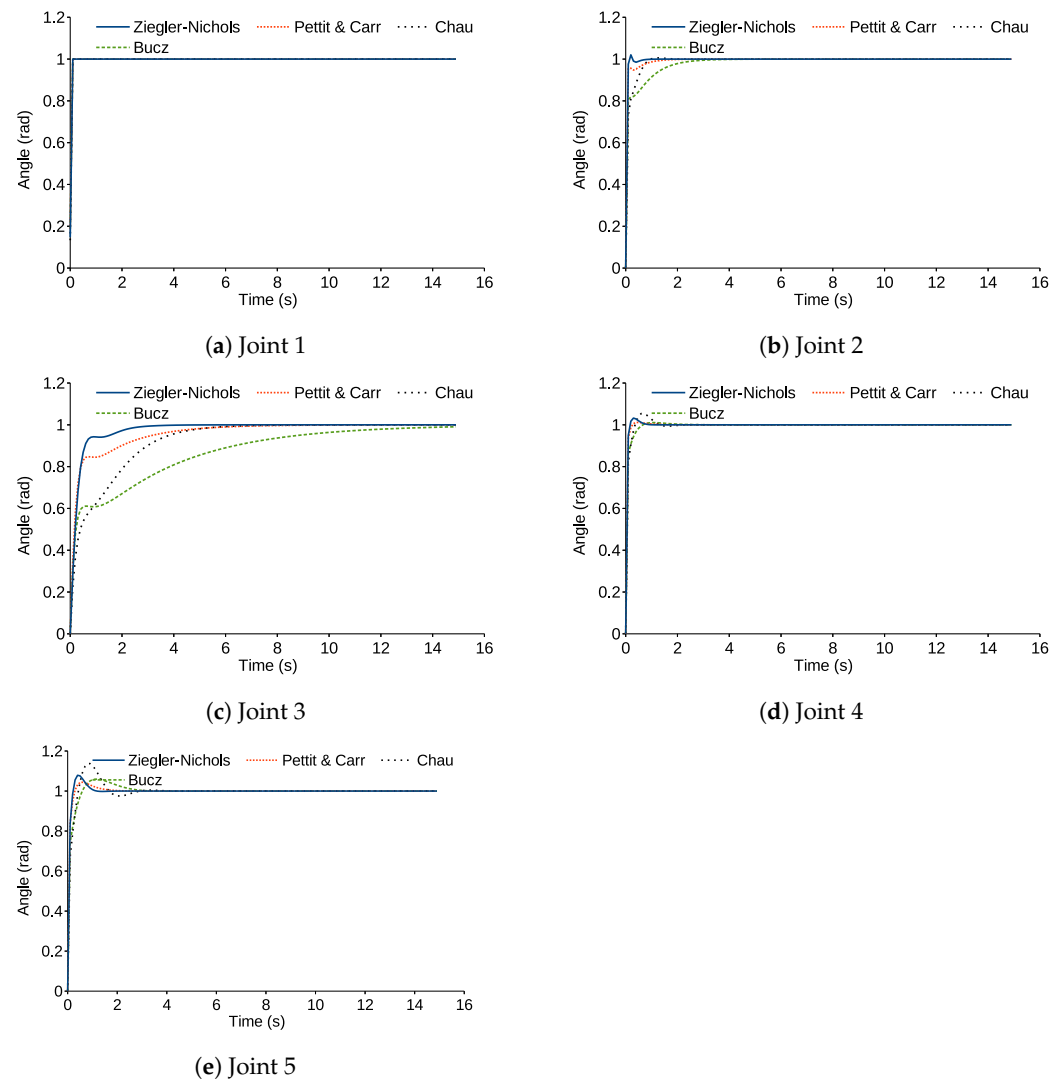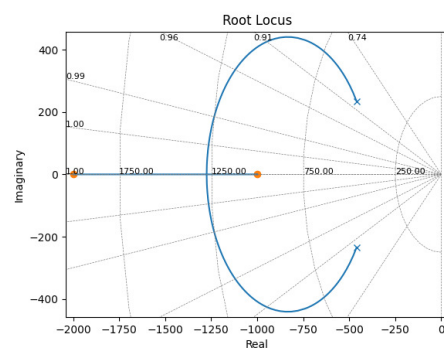
Figure 4. Unit step response of tuned closed-loop control system.

Table 4 compares the average error (AE) in radian and settling time (ST) in seconds determined by the tuning methods for each joint.
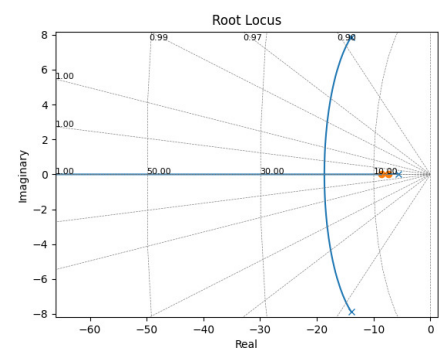
In Table 4, it can be observed that the AE and ST for Z-N show lower values in all joints. In addition, each joint followed the desired trajectory, with lower overshoot by the tuned Z-N parameters than other methods. For example, the AE for Z-N is 6.6%, 83%, and 53% lower than the AE for Pettit & Carr, Chau, and Bucz. Therefore, parameters Z-N are selected for the initialization of LAC due to its lowest AE and ST. To monitor the stability of the closed-loop control system tuned by Z-N, the root locus method is analysed for each joint based on the mathematical models and tuned controller. Figure 5 represents the root locus graph for the PID close-loop control system tuned for each joint.

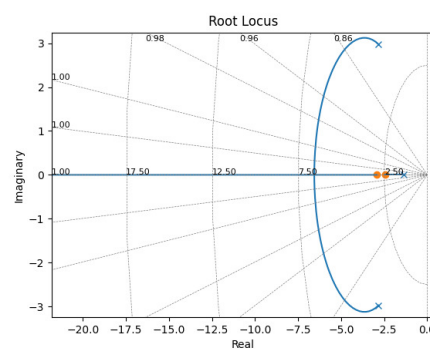**Table 4.** Analysis of unit step response by tuning methods for each link.

		Joint 1	Joint 2	Joint 3	Joint 4	Joint 5
Z-N	AE	0.003	0.011	0.038	0.01	0.015
	ST	0.1	0.3	2.3	0.4	0.6
Pettit and Carr	AE	0.009	0.013	0.064	0.011	0.016
	ST	0.11	0.9	4.8	0.5	0.8
Chau	AE	0.004	0.019	0.114	0.016	0.028
	ST	0.13	0.8	5	0.8	1.6
Bucz	AE	0.008	0.03	0.102	0.014	0.023
	ST	0.115	2.12	5.1	0.9	1.5

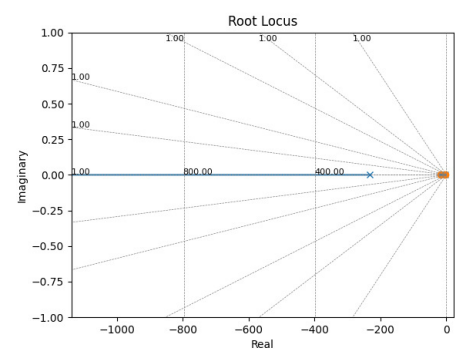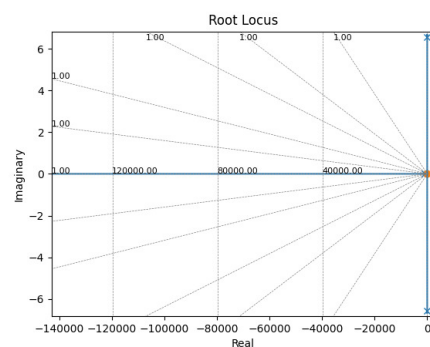(**a**) Joint 1

(**b**) Joint 2

(**c**) Joint 3

(**d**) Joint 4

(**e**) Joint 5

Figure 5. Root locus stability analysis.

In Figure 5, all the roots are on the left side of the graph, i.e., the stable region of the root locus. To validate the performance of the tuned controller, a 3-D model of a robotic arm was created in a virtual environment integrated with ROS, in which the physical characteristics are as defined in Table 1. The virtual environment and the model are designed to analyse the performance of the robotic arm, which resembles the actual one [24,34]. In addition, the model is designed and simulated with similar DoF and structure to the analysed robotic arm. Figure 6 shows the performance of the LAC strategy initialised by the Z-N method for the robotic model in the virtual environment, while the step response actual trajectory is applied for each joint.
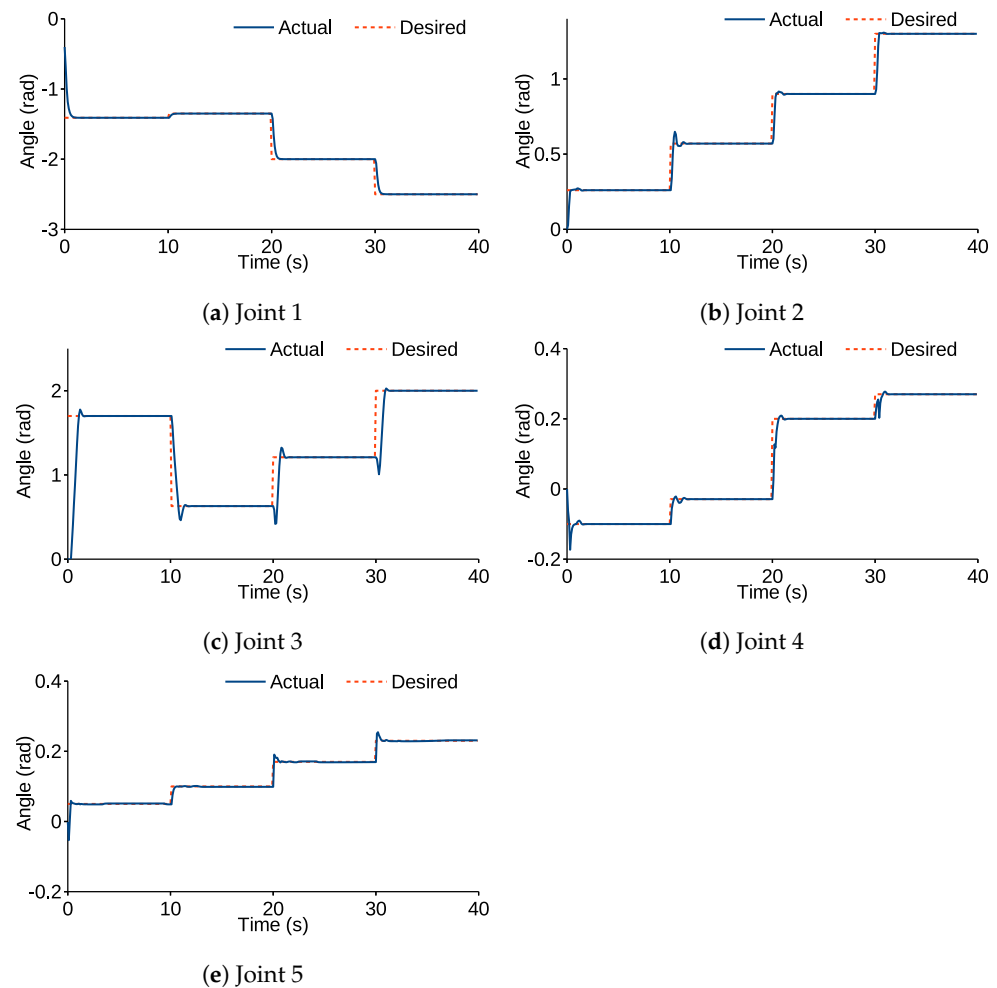

(**a**) Joint 1

(**b**) Joint 2

(**c**) Joint 3

(**d**) Joint 4

(**e**) Joint 5

Figure 6. Performance of LAC for robotic model in virtual environment.

The experimental results in Figure 6 show that the robotic arm has good performance for various step response trajectories. There is an overshoot when there is a change in steps of the desired trajectory because the designed LAC needs to overcome the disturbances and converge the error. In the other experiment, the LAC strategy tuned by Z-N is validated in a 3D model in RoM condition, as shown in Figure 7.
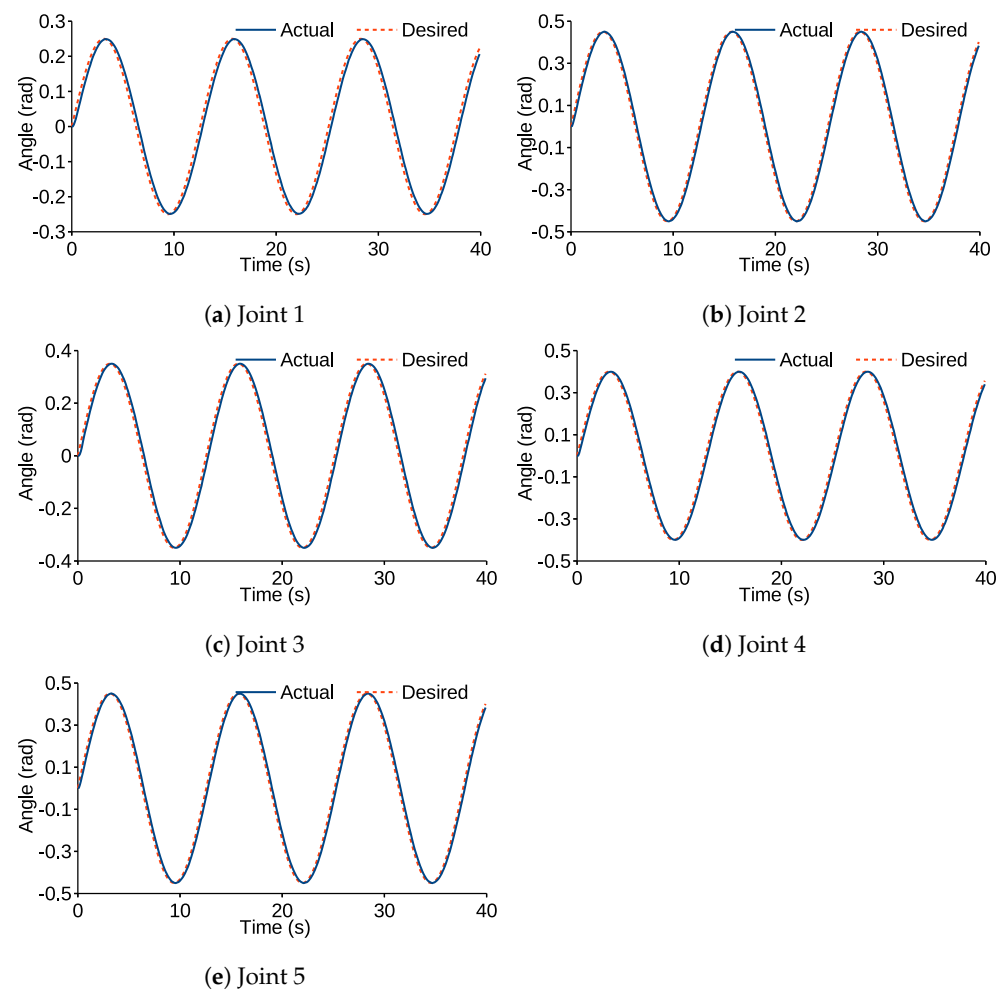
Figure 7. Validation of LAC for robotic model in virtual environment in RoM condition.

The experimental results are shown in Figure 7, in which the control system input is a periodic signal. The results represent satisfactory performance within an acceptable range of error. Table 5 illustrates the AE and root mean square (RMS) for each joint in RoM condition.

Table 5. Statistical analysis of LAC's performance for each joint in RoM condition.

	Joint 1	Joint 2	Joint 3	Joint 4	Joint 5
AE	0.0193	0.022	0.0215	0.0217	0.0228
RMS	0.0215	0.025	0.0239	0.0241	0.0254

AE and RMS are measured in radians and they are in the acceptable range. The statistical analysis shows that AE and RMS did not exceed 0.05 (rad), which represents satisfactory performance [35]. It is concluded that the proposed control method and tuning based on the mathematical model in RoM condition have satisfactory tracking performance.

5. Conclusions

This paper presented the modelling and analysis of an LAC strategy for a 5-DoF multi-joint robotic arm. The Lagrangian technique has been utilised to determine the mathematical model of a robotic arm based on its structure. The transfer functions of each joint were obtained in RoM condition. The LAC's parameters were determined based on the Lyapunov stability approach. The LAC parameters were initialised based on four

observation methods, including Z-N, Pettit & Carr, Chau, and Bucz. The control strategy was validated in a 3D model of the robotic arm in a virtual environment. The results show that the controller with Z-N based on the mathematical model of each joint has better performance than other observation methods. For instance, the AE for Z-N in the unit step response was 6.6%, 83%, and 53% lower than the AE for Pettit & Carr, Chau, and Bucz. In addition, validation results in the virtual environment showed that the AE and RMS were in an acceptable range that was below 0.05 (rad).

The method can be employed for any multi-joint robotic arm to control its end-effector and track its angular trajectory. The limitation of this work is that we did not obtain the desired trajectory of each joint based on tracking the end-effector. In addition, this method was not validated in an actual robotic arm in the presence of disturbances. Future work will be devoted to combining inverse kinematic and visual algorithms to be developed for practical applications. Moreover, the proposed method is applicable in industries such as welding, material handling, and thermal spraying, or any other industrial applications.

Author Contributions: Conceptualisation, M.S.A. and R.R.; methodology, M.S.A. and R.R.; software, M.S.A.; validation, M.S.A. and R.R.; formal analysis, M.S.A. and R.R.; investigation, M.S.A. and R.R.; resources, M.S.A.; data curation, M.S.A. and R.R.; writing—original draft preparation, M.S.A.; writing—review and editing, M.S.A. and R.R.; visualisation, M.S.A.; supervision, R.R.; project administration, R.R.; funding acquisition, R.R. All authors have read and agreed to the published version of the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Wei, H.; Bu, Y.; Zhu, Z. Robotic arm controlling based on a spiking neural circuit and synaptic plasticity. *Biomed. Signal Process. Control* **2020**, *55*, 101640. [CrossRef]
2. Jahnavi, K.; Sivraj, P. Teaching and Learning Robotic Arm Model. In Proceedings of the International Conference on Intelligent Computing, Instrumentation and Control Technologies, Kerala, India, 6–7 July 2017; pp. 1570–1575. [CrossRef]
3. Quiros, A.R.F.; Abad, A.C.; Dadios, E.P. Object Locator and Collector Robotic Arm using Artificial Neural Networks. In Proceedings of the International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM), Cebu, Philippines, 9–12 December 2016; pp. 200–203. [CrossRef]
4. Ye, H.; Wang, D.; Wu, J.; Yue, Y.; Zhou, Y. Forward and inverse kinematics of a 5-DOF hybrid robot for composite material machining. *Robot.-Comput.-Integr. Manuf.* **2020**, *65*, 101961. [CrossRef]
5. Montalvo, W.; Escobar-Naranjo, J.; Garcia, C.A.; Garcia, M.V. Low-Cost Automation for Gravity Compensation of Robotic Arm. *Appl. Sci.* **2020**, *10*, 3823. [CrossRef]
6. Xie, Y.; Zhang, Z.; Wu, X.; Shi, Z.; Chen, Y.; Wu, B.; Mantey, K.A. Obstacle Avoidance and Path Planning for Multi-Joint Manipulator in a Space Robot. *IEEE Access* **2020**, *8*, 3511–3526. [CrossRef]
7. Pavlovčič, U.; Arko, P.; Jezeršek, M. Simultaneous Hand–Eye and Intrinsic Calibration of a Laser Profilometer Mounted on a Robot Arm. *Sensors* **2021**, *21*, 1037. [CrossRef]
8. Jeong, J.H.; Shim, K.H.; Kim, D.J.; Lee, S.W. Brain-Controlled Robotic Arm System Based on Multi-Directional CNN-BiLSTM Network Using EEG Signals. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2020**, *28*, 1226–1236. [CrossRef] [PubMed]
9. Mahmud, M.; Motakabber, S.; Zahirul Alam, A.H.; Nordin, A.N. Adaptive PID Controller Using for Speed Control of the BLDC Motor. In Proceedings of the IEEE International Conference on Semiconductor Electronics (ICSE), Kuala Lumpur, Malaysia, 28–29 July 2020; pp. 168–171. [CrossRef]
10. Misaghi, M.; Yaghoobi, M. Improved invasive weed optimization algorithm (IWO) based on chaos theory for optimal design of PID controller. *J. Comput. Des. Eng.* **2019**, *6*, 284–295. [CrossRef]
11. Castillo-Zamora, J.J.; Camarillo-Gómez, K.A.; Pérez-Soto, G.I.; Rodriguez-Resendiz, J. Comparison of PD, PID and Sliding-Mode Position Controllers for V-Tail Quadcopter Stability. *IEEE Access* **2018**, *6*, 38086–38096. [CrossRef]
12. Demirel, B.; Ghadimi, E.; Quevedo, D.E.; Johansson, M. Optimal Control of Linear Systems with Limited Control Actions: Threshold-Based Event-Triggered Control. *IEEE Trans. Control Netw. Syst.* **2018**, *5*, 1275–1286. [CrossRef]
13. Ma, Z.; Yan, Z.; Shaltout, M.L.; Chen, D. Optimal Real-Time Control of Wind Turbine During Partial Load Operation. *IEEE Trans. Control. Syst. Technol.* **2015**, *23*, 2216–2226. [CrossRef]
14. Belkadi, A.; Oulhadj, H.; Touati, Y.; Khan, S.A.; Daachi, B. On the robust PID adaptive controller for exoskeletons: A particle swarm optimization based approach. *Appl. Soft Comput.* **2017**, *60*, 87–100. [CrossRef]

15. Phu, D.X.; Mien, V.; Tu, P.H.T.; Nguyen, N.P.; Choi, S.B. A New Optimal Sliding Mode Controller with Adjustable Gains based on Bolza-Meyer Criterion for Vibration Control. *J. Sound Vib.* **2020**, *485*, 115542. [CrossRef]

16. Wang, L.; Liu, J.; Li, Y. The optimal controller design framework for PID-based vibration active control systems via non-probabilistic time-dependent reliability measure. *ISA Trans.* **2020**, *105*, 129–145. [CrossRef]

17. Suhaimin, S.C.; Azmi, N.L.; Rahman, M.M.; Yusof, H.M. Analysis of Point-to-Point Robotic Arm Control using PID Controller. In Proceedings of the 7th International Conference on Mechatronics Engineering (ICOM), Putrajaya, Malaysia, 30–31 October 2019; pp. 5–10. [CrossRef]

18. Huang, H.; Shirkhani, M.; Tavoosi, J.; Mahmoud, O. A New Intelligent Dynamic Control Method for a Class of Stochastic Nonlinear Systems. *Mathematics* **2022**, *10*, 1406. [CrossRef]

19. Sun, W.; Lin, J.W.; Su, S.F.; Wang, N.; Er, M.J. Reduced Adaptive Fuzzy Decoupling Control for Lower Limb Exoskeleton. *IEEE Trans. Cybern.* **2021**, *51*, 1099–1109. [CrossRef]

20. Pezzato, C.; Ferrari, R.; Corbato, C.H. A Novel Adaptive Controller for Robot Manipulators Based on Active Inference. *IEEE Robot. Autom. Lett.* **2020**, *5*, 2973–2980. [CrossRef]

21. Tavoosi, J.; Shirkhani, M.; Abdali, A.; Mohammadzadeh, A.; Nazari, M.; Mobayen, S.; Asad, J.H.; Bartoszewicz, A. A new general type-2 fuzzy predictive scheme for PID tuning. *Appl. Sci.* **2021**, *11*, 392. [CrossRef]

22. Ma, C.; Lam, J.; Lewis, F.L. Trajectory Regulating Model Reference Adaptive Controller for Robotic Systems. *IEEE Trans. Control Syst. Technol.* **2019**, *27*, 2749–2756. [CrossRef]

23. Kang, S.; Chou, W. Kinematic Analysis, Simulation and Manipulating of a 5-DOF Robotic Manipulator for Service Robot. In Proceedings of the 2019 IEEE International Conference on Mechatronics and Automation (ICMA), Tianjin, China, 4–7 August 2019; pp. 643–649. [CrossRef]

24. Amiri, M.S.; Ramli, R. Intelligent Trajectory Tracking Behavior of a Multi-Joint Robotic Arm via Genetic–Swarm Optimization for the Inverse Kinematic Solution. *Sensors* **2021**, *21*, 3171. [CrossRef]

25. Amiri, M.S.; Ramli, R.; Ibrahim, M.F. Genetically optimized parameter estimation of mathematical model for multi-joints hip–knee exoskeleton. *Robot. Auton. Syst.* **2020**, *125*, 103425. [CrossRef]

26. Gheisarnejad, M.; Khooban, M.H. An Intelligent Non-Integer PID Controller-Based Deep Reinforcement Learning: Implementation and Experimental Results. *IEEE Trans. Ind. Electron.* **2021**, *68*, 3609–3618. [CrossRef]

27. Amiri, M.S.; Ramli, R.; Tarmizi, M.A.A.; Ibrahim, M.F.; Danesh Narooei, K. Simulation and Control of a Six Degree of Freedom Lower Limb Exoskeleton. *J. Kejuruter.* **2020**, *32*, 197–204. [CrossRef]

28. Díaz-Rodríguez, I.D.; Han, S.; Keel, L.; Bhattacharyya, S. Advanced Tuning for Ziegler-Nichols Plants. *IFAC-PapersOnLine* **2017**, *50*, 1805–1810 [CrossRef]

29. Amiri, M.S.; Ramli, R.; Ibrahim, M.F. Initialized Model Reference Adaptive Control for Lower Limb Exoskeleton. *IEEE Access* **2019**, *7*, 167210–167220. [CrossRef]

30. Li, X.; Yang, X. Stability analysis for nonlinear systems with state-dependent state delay. *Automatica* **2020**, *112*, 108674. [CrossRef]

31. Liu, S.; Liberzon, D.; Zharnitsky, V. Almost Lyapunov functions for nonlinear systems. *Automatica* **2020**, *113*, 108758. [CrossRef]

32. Afaque Khan, M.A.; Swamy, M.N. Modified MRAC based on Lyapunov theory for improved controller efficiency. In Proceedings of the International Conference on Automatic Control and Dynamic Optimization Techniques, Pune, India, 9–10 September 2016; pp. 989–995. [CrossRef]

33. Aliman, N.; Ramli, R.; Haris, S.M.; Amiri, M.S.; Van, M. A robust adaptive-fuzzy-proportional-derivative controller for a rehabilitation lower limb exoskeleton. *Eng. Sci. Technol. Int. J.* **2022**, *35*, 101097. [CrossRef]

34. Amiri, M.S.; Ramli, R.; Ibrahim, M.F.; Wahab, D.A.; Aliman, N. Adaptive Particle Swarm Optimization of PID Gain Tuning for Lower-Limb Human Exoskeleton in Virtual Environment. *Mathematics* **2020**, *8*, 2040. [CrossRef]

35. Wu, J.; Gao, J.; Song, R.; Li, R.; Li, Y.; Jiang, L. The design and control of a 3DOF lower limb rehabilitation robot. *Mechatronics* **2016**, *33*, 13–22. [CrossRef]