Jurnal Ilmiah Teknik Elektro Komputer dan Informatika (JITEKI)
Vol. 8, No. 4, December 2022, pp. 650-659
ISSN: 2338-3070, DOI: 10.26555/jiteki.v8i4.24646

Control Improvement of Low-Cost Cast Aluminium Robotic Arm Using Arduino Based Computed Torque Control

Petrus Sutyasadi
Mechatronics Dept. Faculty of Vocational Studies, Sanata Dharma University, Paingan, Maguwoharjo, Depok, Sleman, 55282, Indonesia

ARTICLE INFO

Article history:

Received August 20, 2022
Revised November 13, 2022
Published December 28, 2022

Keywords:

Robot arm;
Low-cost;
Arduino;
Computed Torque Control;
Feedback Linearization

ABSTRACT

Gravity causes non-linearity in position control of an articulated industrial robotic arm. Especially for a joint position control of a robot's shoulder and elbow that works parallel with the gravity direction. To overcome the problem, Computed Torque Control algorithm was implemented. This algorithm linearized the feedback, so a regular linear Proportional Derivative controller can be implemented. The contribution of this research is to find an effective controller to control a heavy weight low-cost robotic arm link/body using low-cost controller such as Arduino. A Computed Torque Control was implemented to control the shoulder joint of an articulated robotic arm. This joint is the most affected joint by the gravity. It works along the vertical plane, and loaded by the rest of the arm and the robot's load. The proposed controller was compared to a Proportional Integral Derivative (PID) Controller and a Cascade PID Controller. The experiment showed that the Computed Torque Controller can control the position of the arm properly both in the direction along or against the gravity. A linear PID controller could not bring the arm to the set point when it moves against the gravity, but it works well when the arm moves in the opposite direction. A Cascade PID controller has an overshot when the arm moves along the gravity. But it works properly when it moves up against the gravity. A Computed Torque Control works well in both directions even in the presence of gravity force because it includes the gravity on its algorithm.

Corresponding Author:

Petrus Sutyasadi, Mechatronics Dept. Faculty of Vocational Studies, Sanata Dharma University, Paingan, Maguwoharjo, Depok, Sleman, 55282, Indonesia
Email: peter@usd.ac.id

1. INTRODUCTION

The use of industrial robotic arms is increasingly widespread. The use of robotic arms is expected to increase production, efficiency, and also safety. Although the level of demand for the use of robotic arms is increasing, the price for an industrial robot arm is still quite expensive. Some mid-level industries and engineering schools cannot afford the expense if they want to use industrial robotic arm. Engineering schools need to add robotics to their curriculum in order to prepare workers who understand robot control and programming. One way to overcome this problem is to make a low-cost robotic arm. Some researchers tried to develop special purpose robots that handle special purpose task [1]–[6], but some also developed general purpose robotic arm [7]–[13]. Some of them used flexible manipulator instead of rigid body [14], [15]. One of popular algorithm to control the robotic arm manipulator is Proportional Integral Derivative (PID) controller. Some researchers proposed PID controller as their robot controller [16]–[22]. In order to optimize the performance of the PID controller, some hybrid PID controllers were also introduced [23]–[30]. Nonetheless, due to some uncertainties, another controller besides PID were also introduced [31]–[37].

Experiments to control a robotic manipulator using a robust H∞ controller have been carried out by Iqbal [38]. However, the research is still limited to simulation only. Hayat [39] said that the controller model before the introduction of the robust controller omitted too many dynamic parameters so that the stability and performance of the robot was not good. However, the traditional synthesis of H∞ robust controllers result in high-order controllers. So, if later combined with the order of the controlled device, it will produce a much larger order of the control system. If it will be implemented on a simple controlling device such as a microcontroller that has a low speed/clock, it will cause problems. To avoid this, the order of the controller is determined first by determining its structure. Several researchers such as Chaiya and Kaitwanidviai [40] and Olranthichachat and Kaitwanidvilai [41] tried to make a H∞ controller with a simple structure that had been determined in advance with the Particle Swarm Optimization optimization technique. The same approach using optimization Genetic Algorithm has been carried out by Kaitwanidvilai and Parnichkun [42]. Sutyasadi and Parnichkun [43] also tried the same strategy but used the Differential Evolution optimization technique. They show that the stability of the system is maintained even though there are variations in model parameters up to ±30%. The response of the controller based on Differential Evolution optimization has a faster rise time and settling time, but the controller generated from the Matlab hinfstruct command has a more consistent response to the variation of model parameters by ±30%. From these studies, it is known that using a solid controller H∞ even though there are variations in model parameters caused by changes in the dynamic model of the system, the system remains stable. Even so, there are variations in the system response which means that the robot arm will have a varied trajectory. Whereas what is needed in controlling the robotic arm is the consistency of the trajectory tracking. Sutyasadi and Parnichkun [44] showed an improvement in the trajectory tracking of a robotic manipulator using a combination of Proportional Integral Derivative controllers with Iterative Learning Control controllers. In term of the mechanical structure, Bora and Nandi [45] have developed a low-cost robotic arm but this robot is made on a very small scale and uses only a small RC Servo motor. Jahnavi and Sivraj [46] developed an articulated robotic arm for education, but the developed robot also only uses a small RC servo motor. Strong robotic arms require strong materials as well. This makes the robot arm a bit heavier. High inertia manipulator requires a larger or more powerful drive motor. In this study, the robotic arm was made using cast aluminum. This makes the arm inertia become considerably large. The large inertia of the mechanical manipulator will greatly affect the control effort to reach stability. Moreover, in handling the gravity effect that also contribute non-linearity to the system. Position control of an articulated robotic arm is a non-linear control due to the present of the gravity force. Nonetheless, some linear control like PID is used to control this type of manipulator. In order to control the system using a linear controller, the system should be linearized. In this research, inverse dynamic was applied to linearized the feedback part and form a computed torque control system. The algorithm being used is Computed Torque Control. For the performance comparison, PID and Cascade PID algorithms performance were also presented.

The contribution of this research is to present a good control algorithm to control the position of a medium weight low-cost robotic arm made form cast aluminium. Especially in handling the gravity effect that perturb a lot the manipulator vertical movement.

2. METHODS

The method or sequence of steps of this research is shown in Fig. 1. After the assembly of the system, a preparation for the low-level control was conducted. The preparation involved some tests and calibration of the sensors and the actuators. Experiment for the low level controller was conducted in three different algorithm which are: The Computed Torque Control, Proportional Integral Derivative (PID) Control, and Cascade PID Controller. After that, data analisys of the system responses were analized.

A cast aluminium of robotic arm was constructed. The arm has three degrees of freedom. The robotic arm manipulator is shown in Fig. 2. Axis 1 and axis 2 are actuated using geared dc motor, axis 3 uses geared stepper motor. The base motor does not have much problem caused by the gravity force. Therefore, the base motor will not be included into the experiment. The shoulder joint of the robot manipulator is the hardest joint to be controlled. It has the highest load compared to the other joints in the robotic arm manipulator. Three control algorithms were compared to control the joint to move up and down. The gravity force affects the load torque significantly, and make the control effort differ significantly during the way up and down.

2.1. Computed Torque Control

Controlling a robotic arm that moves vertically will get a very influential disturbance from the effects of gravity. In the process of setting up a linear controller, this will be very difficult because the power to move the robot arm in certain positions will be different. It also depends on the posture of the arm at that time. If the position and posture are in a position where the effect of gravity is high, more energy is needed to lift the arm. Rotating the shoulder joint to move the arm up needs a large of controller gain. But by using the same controller gain, moving the joint down will cause an overshot because the arm will move much faster, and it may lead to instability. For that we need a method that pays attention to the influence of gravity on a certain position or posture of the robot arm. This can be done with the Feedback Linearization algorithm, or using the Computed Torque Control algorithm. The torque provided by the controller is dynamic according to the torque value calculated by the algorithm. Computed Torque Control is also known as Inverse Dynamics. The way this controller works is based on the principle of feedback linearization which will map a non-linear model into a linear model. Therefore, we can apply a linear controller such as a PID or PD controller to control a non-linear system such as a robot arm. A dynamic equation of the robot is shown in (1).

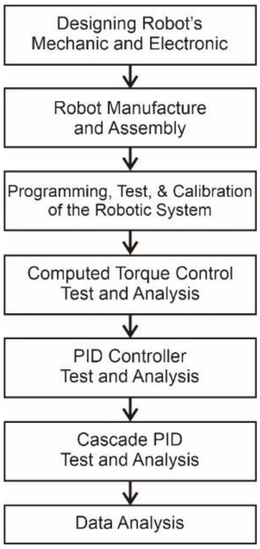


Fig. 1. Flow chart of the research methods

Fig. 2. Developed cast aluminium robotic manipulator

$$M(q)q + C(q,q) + G(q) = \tau \tag{1}$$

where q is angular position, M is inertial matrix, C is coriolis/centripetal vector, G is gavity vector, and τ is control input torque. The target of the controller is to get the right τ so that the robot can follow the reference trajectory. The inertial matrix (M) is

$$M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \tag{2}$$

$$M_{11} = m_1 l_1^2 + m_2[l_1^2 + l_2^2 + 2l_1 l_2 \cos q_2] \tag{3}$$

$$M_{12} = M_{21} = m_2 l_1 l_2 \cos q_2 + m_2 l_2^2 \tag{4}$$

$$M_{22} = m_2 l_2^2 \tag{5}$$

$$C = \begin{bmatrix} -h\dot{q}_2 & -h\dot{q}_1 - h\dot{q}_2 \\ h\dot{q}_1 & 0 \end{bmatrix} \tag{6}$$

with $h = m_2 l_1 l_2 \sin q_2$

$$G = \begin{bmatrix} G_1 \\ G_2 \end{bmatrix} = \begin{bmatrix} m_1 l_1 g \cos q_1 + m_2 g (l_2 \cos(q_1 + q_2) + l_1 \cos q_1) \\ m_2 l_2 g \cos(q_1 + q_2) \end{bmatrix} \tag{7}$$

Equation (1) can be rewritten as:

$$\ddot{q} = M^{-1}(q)(\tau - C(q, \dot{q}) - G(q)) \tag{8}$$

If the M, C, and G matrices already have parameters, then we can choose the control input as follows:

$$\tau = M(q)[\wedge] + C(q, \dot{q}) + G(q) \tag{9}$$

Where $\wedge$ depends on the type of controller selected. In Proportional Derivative (PD) Computed Torque Control, then $\wedge$ is defined as:

$$\wedge = \ddot{q}_d + K_p e + K_d \dot{e} \tag{10}$$

All the robot's parameters are shown in Table 1.

Table 1. Equation parameters

Variable	Physical representation	Robot parameter value
m_1	Mass of link 1	2kg
m_2	Mass of link 2	1.1kg
l_1	Length of link 1	0.34m
l_2	Length of link 2	0.29m
q_1	Position of joint 1	variable
q_2	Position of joint 2	variable
g	Gravity force	9.8ms^{-2}

Where $e(t)$ is the difference between the target angle and the actual position. Kp is the proportional gain, Kd is the derivative gain. The PD-CTC control block diagram is shown in Fig. 3.

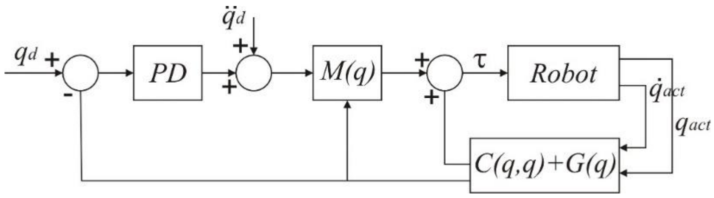
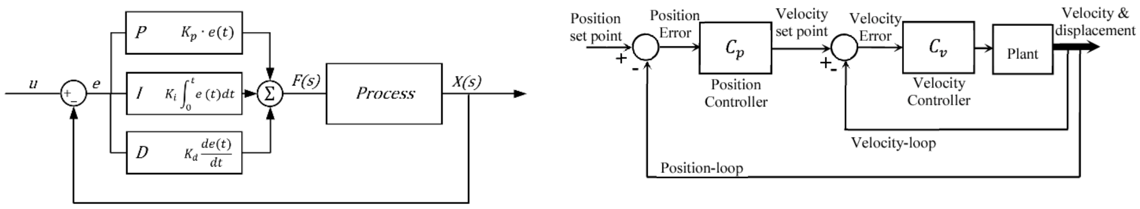


Fig. 3. PD Computed Torque Control block diagram

In the experiments, system response from the PD Computed Torque Controller was compared to PID Controller and Cascade PID Controller. The block diagram of PID and Cascade PID Controller are shown in Fig. 4. The meaning of the block diagram parameters are presented in Table 2.

Table 2. Diagram block parameters

Variable	Physical representation
Kp	Proportional Constant
Ki	Integral Constant
Kd	Derivative Constant
e	Error signal
$F(s)$	Manipulated Variable
u	Set Point/reference value
$X(s)$	Output
Cp	PID for position control
Cv	PID for velocity control



(a) (b)

Fig. 4. Block diagram of (a) PID Controller, (b) Cascade PID Controller

3. RESULTS AND DISCUSSION

The comparison of the three algorithms is presented in this chapter. Fig. 5 and Fig. 6 show the system response of the shoulder joint position control using PID controller. Fig. 5 shows the system response when the shoulder moves in upward direction. The graph shows that there is a large steady state error. It shows that a PID controller cannot be used in this case. PID gain can be set to higher value to reduce the steady state error. However, the arm will move too fast when it move down. The system moved very aggressive at the beginning; however, it stop when it reached around 40% of the setpoint. After that it seemed the system rely most on the integral action of the PID controller. Fig. 6 shows the system response when the shoulder moves in downward direction. Using the same gain configuration with the system response showed in Fig. 5, the arm can reach the set point position. It indicates that the controller gain was set when it moves in downward direction. Fig. 6 also shows that the system has settling time around 0.3 second. It indicates that the arm was pulled down by the gravity significantly.

The PID Controller was tuned based on the downward arm movement. Fig. 6 shows that the system response has satisfactory result. It has no overshoot, small rise time and settling time, and no steady state error. However, Fig. 5 shows that the system has a long rise time and settling time, also a very large steady state error. This was caused by the gravity effect. If the system is tuned when the arm is moving up, the gain will be higher to move the arm against the gravity. Nevertheless, even the gains are set to some bigger number, the arm sometimes cannot hit the setpoint. If the gains are set to a much bigger value, it will drive oscillation and sometimes cause instability. Another problem is when the arm moves downward, the system moves to fast along the gravity. It leads to overshoot and oscillation.

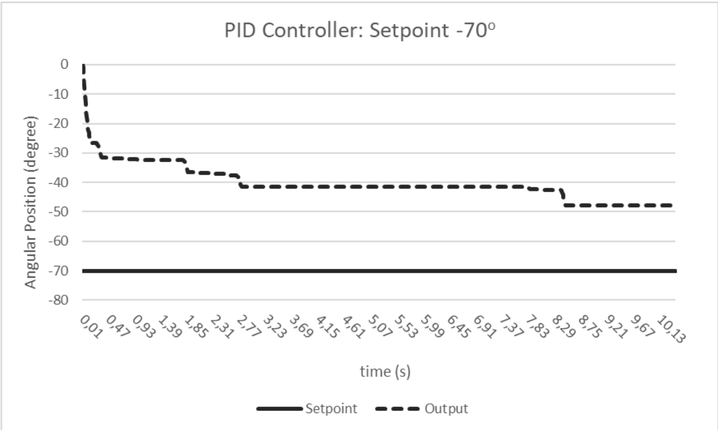


Fig. 5. Shoulder swings up response using PID controller

Fig. 7 and Fig. 8 show the system response of the shoulder joint position control using Cascade PID controller. Fig. 7 shows the system response when the shoulder moves in upward direction. While Fig. 8 shows the system response when the shoulder moves in downward direction. Using cascade controller, the controller gain can be set using the same gain for both upward or backward direction. The gain was set to be high enough to bring the arm moved in upward direction. Using the same gain, the arm can go down to the setpoint without any problem except one small overshoot and small steady state error. The gain was optimally set to get the best response for both going in upward or downward direction. Fig. 8 shows that the gravity effect still dominantly disturb the system during the movement.

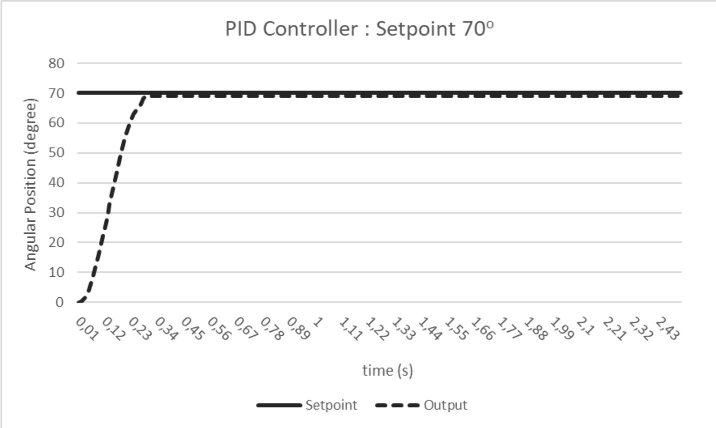


Fig. 6. Shoulder swing down response using PID controller

Cascade PID controller has two PID blocks, which are: the inner block/loop and the outer block/loop. Inner loop has faster cycle time than the outer loop. The inner loop controls the velocity of the arm, and the outer loop controls the arm position. The controller is tuned based on trial-and-error method, but still considering the general PID characteristic. Firstly, the position is tuned and then the velocity is adjusted later.

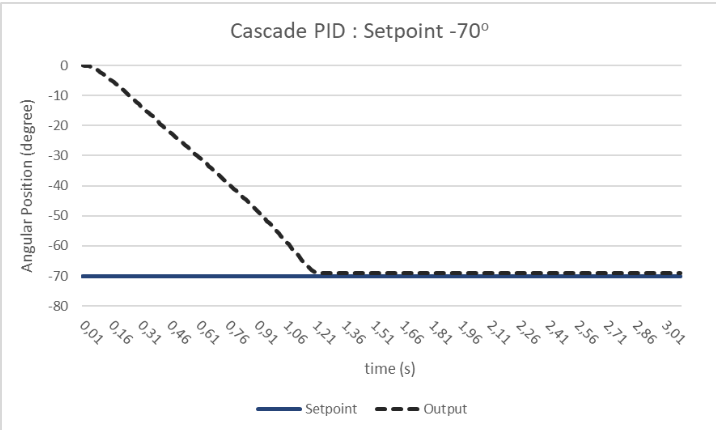


Fig. 7. Shoulder swings up response using Cascade PID controller

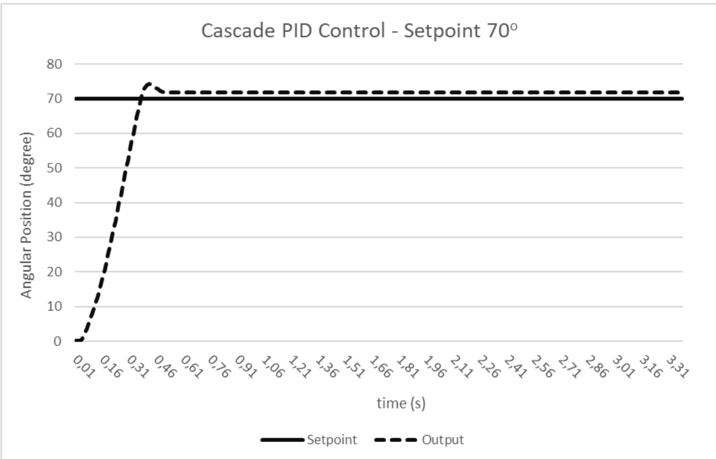


Fig. 8. Shoulder swing down response using Cascade PID controller

Fig. 9 and Fig. 10 show the system response of the shoulder joint position control using Computed Torque Control algorithm. Fig. 9 shows the system response when the shoulder moves in upward direction. Rise time when the arm moves in upward direction is smaller compared to when it moves in downward direction. This shows different behavior compared to PID and Cascade PID. But it shows that the algorithm can handle the gravity effect during the movement. While Fig. 10 shows the system response when the shoulder moves downward. Both up or down movement can reach the setpoint perfectly.

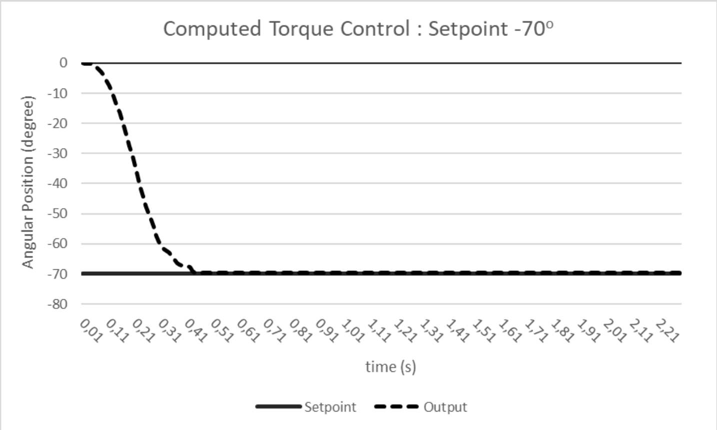


Fig. 9. Shoulder swings up response using PD Computed Torque Control

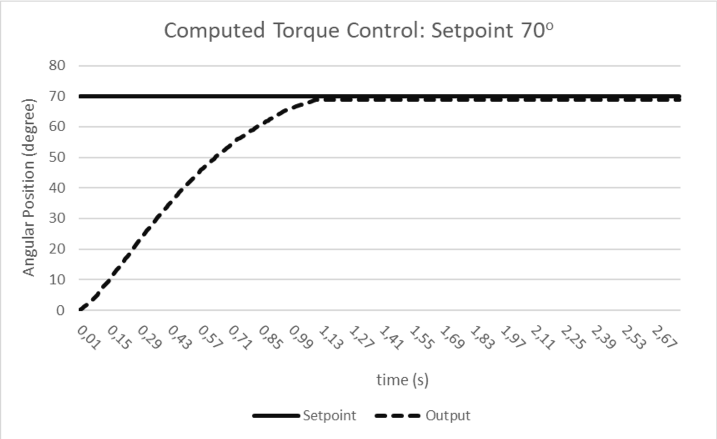


Fig. 10. Shoulder swing down response using PD Computed Torque Control

The PD Computed Torque Control gains were tuned based on trial-and-error method also. The position was set before setting the velocity, until the system reaches the most preferred response. Table 3 shows the full comparison of the three controller's system responses. All the time unit are in second, and the steady state error unit is in degree (angular position).

Table 3. System Response Comparison of The Three Controllers Performance

Controller	Move	Rise time(t)	Settling time(s)	Overshoot (%)	Steady State Error(deg)
PID	Up	∞	∞	0	22
	Down	0	1	0	2
Cascade PID	Up	1	1.15	0	1
	Down	0.31	0.4	7.14	2
PD-Computed Torque	Up	0.3	0.4	0	0.5
	Down	0.75	1	0	1

4. CONCLUSION

As mentioned in the beginning that one big problem in controlling robotic arm with high inertia is the position control in vertical direction. Usually, it is affected by the gravity, so that the the gain selection is hard to be chosen to compromise both for upward and downward direction. That non-linear situation is hardly to be solved by using a linear controller such as PID. Therefore, a linearization process was involved. The experiment shows that feedback linearization can solve the position of the robotic arm manipulator. The robotic arm was developed in the lab using cast aluminium. Therefore, it has a large amount of inertia, and potentially affected by the gravity force. Gain selection of PID controller that work for both up and down arm movement is hard to be done. When the gain was selected for down way movement, the system cannot reach the setpoint in upward direction. The best system response was given by the CTC. There was big rise time, settling time, and steady state error when we use PID only. Better response was given by Cascade controller. The worst case from Cascade controller were the steady state error at 2 degree and the 7% overshoot. System response of the Computed Torque Controller was considered as the best result among the three controller performances. The response has the smallest steady state error, without overshoot, and acceptable rise time around 0.3 second when the shoulder moves up against the gravity force, and 0.75 second when it moves down.

Acknowledgments

The author would like to thank Sanata Dharma University for providing the research funding.

REFERENCES

[1] A. Misra, A. Sharma, G. Singh, A. Kumar, and V. Rastogi, "Design and Development ofa Low-Cost CNC Alternative SCARA Robotic Arm," *Procedia Comput Sci*, vol. 171, pp. 2459–2468, 2020, doi: https://doi.org/10.1016/j.procs.2020.04.266.

[2] K. Rahul, H. Raheman, and V. Paradkar, "Design and development of a 5R 2DOF parallel robot arm for handling paper pot seedlings in a vegetable transplanter," *Comput Electron Agric*, vol. 166, p. 105014, 2019, https://doi.org/10.1016/j.compag.2019.105014.

[3] H. Seki *et al.*, "Development of assistive robotic arm for power line maintenance," *Precis Eng*, vol. 67, pp. 69–76, 2021, https://doi.org/10.1016/j.precisioneng.2020.09.006.

[4] V. Paradkar, H. Raheman, and R. K., "Development of a metering mechanism with serial robotic arm for handling paper pot seedlings in a vegetable transplanter," *Artificial Intelligence in Agriculture*, vol. 5, pp. 52–63, 2021, https://doi.org/10.1016/j.aiia.2021.02.001.

[5] X. Chen, S. Deng, M.-L. Sun, and R. He, "Robotic arm-assisted arthroplasty: The latest developments," *Chinese Journal of Traumatology*, vol. 25, no. 3, pp. 125–131, 2022, https://doi.org/10.1016/j.cjtee.2021.09.001.

[6] A. Caon, F. Branz, and A. Francesconi, "Development and test of a robotic arm for experiments on close proximity operations," *Acta Astronaut*, vol. 195, pp. 287–294, 2022, https://doi.org/10.1016/j.actaastro.2022.03.006.

[7] N. Nurnuansuwan, A. Lonklang and K. Chamniprasart, "A Prototyping of 2-DOF Robot Arm Using Feedback Control System," *2019 5th International Conference on Control, Automation and Robotics (ICCAR)*, pp. 368-372, 2019, https://doi.org/10.1109/ICCAR.2019.8813726.

[8] P. C. Sau, "Comparative study of Fractional Order Controllers for Three-link Robotic Manipulator System," in *2020 International Conference on Power Electronics & IoT Applications in Renewable Energy and its Control (PARC)*, pp. 528–533, 2020, https://doi.org/10.1109/ICCAR.2019.8813726.

[9] O. Glaufe, O. Gladstone, E. Ciro, C. C. A. T, and L. José, "Development of Robotic Arm Control System Using Computational Vision," *IEEE Latin America Transactions*, vol. 17, no. 8, pp. 1259–1267, 2019, https://doi.org/10.1109/TLA.2019.8932334.

[10] X. Li, Z. Jia, X. Cui, and L. Zhang, "Design, Analysis and Experiment of A Non-humanoid Arm Exoskeleton for Lifting Load," in *2018 IEEE International Conference of Intelligent Robotic and Control Engineering (IRCE)*, pp. 64–68, 2018, https://doi.org/10.1109/IRCE.2018.8492965.

[11] F. S. M. Alkhafaji, "Modeling and control High speed Robotic Arm for Industrial Applications," in *2021 Global Congress on Electrical Engineering (GC-ElecEng)*, pp. 121–130, 2021, https://doi.org/10.1109/GC-ElecEng52322.2021.9788417.

[12] Z. Zhang, F. Ferry, Q. Fang, and X. Gu, "Robotic arm for unsupervised stroke rehabilitation: A pilot study using PID controller," in *2013 1st International Conference on Orange Technologies (ICOT)*, pp. 19–22, 2013, https://doi.org/10.1109/ICOT.2013.6521147.

[13] A. Nugroho, E. M. Yuniarno, and M. H. Purnomo, "Cooperative Multi-agent for The End-Effector Position of Robotic Arm Based on Consensus and PID Controller," in *2019 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)*, pp. 1–6, 2019, https://doi.org/10.1109/CIVEMSA45640.2019.9071621.

[14] C. Yanga and L. Banb, "Vibration Control of Piezoelectric Flexible Manipulator Based on Machine Vision and Improved PID," in *2021 IEEE International Conference on Power Electronics, Computer Applications (ICPECA)*, pp. 886–888, 2021, https://doi.org/10.1109/ICPECA51329.2021.9362620.

[15] W. Alam, N. Ali, H. M. W. Aziz, and J. Iqbal, "Control of Flexible Joint Robotic Manipulator: Design and Prototyping," in *2018 International Conference on Electrical Engineering (ICEE)*, pp. 1–6, 2018, https://doi.org/10.1109/ICEE.2018.8566796.

[16] A. D. Shakibjoo and M. D. Shakibjoo, "2-DOF PID with reset controller for 4-DOF robot arm manipulator," in *2015 International Conference on Advanced Robotics and Intelligent Systems (ARIS)*, pp. 1–6, 2015, https://doi.org/10.1109/ARIS.2015.7158355.

[17] A. L. Luna, I. C. Vega, and J. M. Carranza, "Gain-Scheduling and PID Control for an Autonomous Aerial Vehicle with a Robotic Arm," in *2018 IEEE 2nd Colombian Conference on Robotics and Automation (CCRA)*, pp. 1–6, 2018, https://doi.org/10.1109/CCRA.2018.8588145.

[18] D. Zhang and B. Wei, “Design of a joint control system for serial mechanical arms based on PID and MRAC control,” in *2016 Asia-Pacific Conference on Intelligent Robot Systems (ACIRS)*, pp. 91–96, 2016, https://doi.org/10.1109/ACIRS.2016.7556194.

[19] J. N. Quispe, D. C. Solis, J. H. León, and L. M. Ccopa, “Performance comparison between PD and PID Controller in an upper limb Exoskeleton by analyzing an arm trajectory modeled with Image Recognition,” in *2020 IEEE Engineering International Research Conference (EIRCON)*, pp. 1–4, 2020, https://doi.org/10.1109/EIRCON51178.2020.9254082.

[20] J. S. Kumar and E. K. Amutha, “Control and tracking of robotic manipulator using PID controller and hardware in Loop Simulation,” in *2014 International Conference on Communication and Network Technologies*, pp. 1–3, 2014, https://doi.org/10.1109/CNT.2014.7062712.

[21] J. d. J. Rubio, P. Cruz, L. A. Paramo, J. A. Meda, D. Mujica, and R. S. Ortigoza, “PID Anti-Vibration Control of a Robotic Arm,” *IEEE Latin America Transactions*, vol. 14, no. 7, pp. 3144–3150, 2016, https://doi.org/10.1109/TLA.2016.7587614.

[22] S. Singh, M. S. Qureshi, and P. Swarnkar, “Comparison of conventional PID controller with sliding mode controller for a 2-link robotic manipulator,” in *2016 International Conference on Electrical Power and Energy Systems (ICEPES)*, pp. 115–119, 2016, https://doi.org/10.1109/ICEPES.2016.7915916.

[23] J. N. Quispe, D. C. Solis, J. H. León, and L. M. Ccopa, “Performance comparison between PD and PID Controller in an upper limb Exoskeleton by analyzing an arm trajectory modeled with Image Recognition,” in *2020 IEEE Engineering International Research Conference (EIRCON)*, pp. 1–4, 2020, https://doi.org/10.1109/EIRCON51178.2020.9254082.

[24] O. Djaneye-Boundjou, X. Xu, and R. Ordóñez, “Automated particle swarm optimization based PID tuning for control of robotic arm,” in *2016 IEEE National Aerospace and Electronics Conference (NAECON) and Ohio Innovation Summit (OIS)*, pp. 164–169, 2016, https://doi.org/10.1109/NAECON.2016.7856792.

[25] Z. m. omer, O. I. Al-Agha, A. o. a. bakr, and k. h. Bilal, “Performance Analysis of OWMR Directional Wheels Robot Arm based on (ANFIS+PID) controller,” in *2018 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE)*, pp. 1–5, 2018, https://doi.org/10.1109/ICCCEEE.2018.8515874.

[26] M. P. Lazarević, P. D. Mandić, B. Cvetković, L. Bučanović, and M. Dragović, “Advanced open-closed-loop PIDD2/PID type ILC control of a robot arm,” in *2018 Innovations in Intelligent Systems and Applications (INISTA)*, pp. 1–8, 2018, https://doi.org/10.1109/INISTA.2018.8466308.

[27] P. Yeptho, K. Katheria, S. Swain, and P. Gaur, “Analyzing the Impact of Degrees of Freedom on the Performance of Fractional Order PID Controllers in Robotic Manipulators with Payloads,” in *2020 IEEE 17th India Council International Conference (INDICON)*, pp. 1–6, 2020, https://doi.org/10.1109/INDICON49873.2020.9342298.

[28] S. Suvilath, K. Khongsomboun, T. Benjanarasuth, and N. Komine, “IMC-based PID controllers design for a two-links SCARA robot,” pp. 1030–1034, 2011, https://doi.org/10.1109/TENCON.2011.6129267.

[29] A. A. Mohammed and A. Eltayeb, “Dynamics and Control of a Two-link Manipulator using PID and Sliding Mode Control,” in *2018 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE)*, pp. 1–5, 2018, https://doi.org/10.1109/ICCCEEE.2018.8515795.

[30] E. O. Freire, F. G. Rossomando, and C. M. Soria, “Self-tuning of a Neuro-Adaptive PID Controller for a SCARA Robot Based on Neural Network,” *IEEE Latin America Transactions*, vol. 16, no. 5, pp. 1364–1374, 2018, https://doi.org/10.1109/TLA.2018.8408429.

[31] N. Dersarkissian, R. Jia, and D. L. Feitosa, “Control of a Two-link Robotic Arm using Fuzzy Logic,” in *2018 IEEE International Conference on Information and Automation (ICIA)*, pp. 481–486, 2018, https://doi.org/10.1109/ICInfA.2018.8812444.

[32] A. P. Acharya, P. Biswal, J. C. Pati, U. Samantsinghar, and K. C. Rout, “Performance Analysis of Stepper Actuator with Leadscrew for Robotic Arm Motion Applications using Fractional Calculus,” in *2020 International Conference on Computer Science, Engineering and Applications (ICCSEA)*, pp. 1–5, 2020, https://doi.org/10.1109/ICCSEA49143.2020.9132918.

[33] M. Elsisi, K. Mahmoud, M. Lehtonen, and M. M. F. Darwish, “An Improved Neural Network Algorithm to Efficiently Track Various Trajectories of Robot Manipulator Arms,” *IEEE Access*, vol. 9, pp. 11911–11920, 2021, https://doi.org/10.1109/ACCESS.2021.3051807.

[34] M. Manoj, S. Ananthakrishnan, P. Sriharshitha, and V. R. Pandi, “Four Axis Welding Robot Control using Fuzzy Logic,” in *2021 2nd Global Conference for Advancement in Technology (GCAT)*, pp. 1–5, 2021, https://doi.org/10.1109/GCAT52182.2021.9587770.

[35] M. Abbas, J. Narayan, and S. K. Dwivedy, “Simulation Analysis for Trajectory Tracking Control of 5-DOFs Robotic Arm using ANFIS Approach,” in *2019 5th International Conference on Computing, Communication, Control and Automation (ICCUBEA)*, pp. 1–6, 2019, https://doi.org/10.1109/ICCUBEA47591.2019.9128742.

[36] A. Parvaresh and S. A. A. Moosavian, “Modeling and Model-free Fuzzy Control of a Continuum Robotic Arm,” in *2018 6th RSI International Conference on Robotics and Mechatronics (IcRoM)*, pp. 501–506, 2018, https://doi.org/10.1109/ICRoM.2018.8657596.

[37] N. Nagpal, B. Bhushan and V. Agarwal, "Intelligent control of four DOF robotic arm," 2016 IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES), pp. 1-6, 2016, https://doi.org/10.1109/ICPEICES.2016.7853357.

[38] J. Iqbal, “Modern Control Laws for an Articulated Robotic Arm: Modeling and Simulation,” *Engineering, Technology and Applied Science Research*, vol. 9, pp. 4057–4061, 2019, https://doi.org/10.48084/etasr.2598.
[39] R. Hayat, M. Leibold, and M. Buss, “Robust-Adaptive Controller Design for Robot Manipulators Using the $\mathcal{H}_{\infty}$ Approach,” *IEEE Access*, vol. 6, pp. 51626–51639, 2018, https://doi.org/10.1109/ACCESS.2018.2870292.
[40] C. Ukrit and K. Somyot, “Fixed-Structure Robust DC Motor Speed Control,” in *Proceedings of the International MultiConference of Engineers and Computer Scientists*, vol. 2, 2009, https://www.iaeng.org/publication/IMECS2009/IMECS2009_pp1533-1536.pdf.
[41] O. Piyapong and K. Somyot, “Structure Specified Robust H_infinite Loop Shaping Control of a MIMO Electro-hydraulic Servo System using Particle Swarm Optimization,” in *Proceedings of the International MultiConference of Engineers and Computer Scientists*, vol. 2, 2011, https://www.iaeng.org/publication/IMECS2011/IMECS2011_pp879-883.pdf.
[42] S. Kaitwanidvilai and M. Parnichkun, “Genetic algorithm based fixed structure robust H∞ loop shaping of a pneumatic servo system,” *J. Robot. Mechatron.*, vol. 16, pp. 11–20, 2004, https://doi.org/10.20965/jrm.2004.p0362.
[43] P. Sutyasadi and M. Parnichkun, “Gait Tracking Control of Quadruped Robot Using Differential Evolution Based Structure Specified Mixed Sensitivity H∞ Robust Control,” *J. Control Sci. Eng.*, vol. 2016, p. 7, 2016, https://doi.org/10.1155/2016/8760215.
[44] P. Sutyasadi and M. Parnichun, “Trotting control of a quadruped robot using PID-ILC,” in *IECON 2015 - 41st Annual Conference of the IEEE Industrial Electronics Society*, pp. 4400–4405, 2015, https://doi.org/10.1109/IECON.2015.7392784.
[45] P. Bora and V. Nandi, “Low cost shadow function based articulated robotic arm,” in *2015 International Conference on Energy, Power and Environment: Towards Sustainable Growth (ICEPE)*, pp. 1–4, 2015, https://doi.org/10.1109/EPETSG.2015.7510079.
[46] K. Jahnavi and P. Sivraj, “Teaching and learning robotic arm model,” in *2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*, pp. 1570–1575, 2017, https://doi.org/10.1109/ICICICT1.2017.8342804.

BIOGRAPHY OF AUTHORS

Petrus Sutyasadi obtained a master's degree in mechatronics in 2008 and a doctorate in mechatronics in 2016 from the Asian Institute of Technology in Thailand. Currently. He is a lecturer at the Department of Mechatronics Engineering, Sanata Dharma University, Yogyakarta, Indonesia. His research interests include robust control, robotics, and frugal innovation in mechatronic engineering. He can be contacted at email: peter@usd.ac.id.