Digital Twin-Based Interactive Robotic Arm System For Skill Training New Joinees

Aadhith Sheshu P S[1], Tejash N[1], Nishok N D[1] and G Nareshbabu[1*]

[1] Rajalakshmi Engineering College, Thandalam, Chennai, Tamil Nadu, India

nareshbabu.g@rajalakshmi.edu.in

Abstract. Digital twin technology will allow better control of robots since it allows to synchronise physical with virtual versions of systems in order to minimise risk associated with hardware and enhance experimental flexibility. In the traditional robotic arm programming, fixed kinematic models are used, and data is communicated through wire, which restricts the flexibility and scalability with the dynamic environment. This paper suggests a digital twin consisting of a Unity 3D simulation environment and a 5-DOF robotic arm, which is controlled through UDP communication over Wi-Fi and operated with a Unity 3D that is inexpensive and inexpensive to interact with directly. A PCA9685 controller makes MG995 high-torque servos run on the physical system to achieve stable multi-joint actuation. Latency tests reveal an average delay of 32 ms in the process of synchronization when using wireless networks of consumer grade, which ensures the motion mapping is smooth in both directions. Controlled validation intertwines sound real time performance applicable to engineering teaching and experimental prototyping. The platform offers an affordable alternative to the costly industrial digital twin platform with future capability of being connected with reinforcement learning and adaptive control strategies.

Keywords: Digital Twin, Control of a robot arm by means of Unity 3D simulation, ESP32, UDP communication, Wireless Robotics, Engineering Education.

1 Introduction

The digital twin technology has become a game-shifting paradigm in the current robotics, taking the opportunity to create virtual copies of the physical systems that reflect the behavior and conditions they are located in. Digital twins enable engineers to test algorithms, optimize performance and predict faults without damaging hardware because sensor data, control signals, and interactions with the environment can be synchronized. Digital twins are used as a safe and scale-able system to permit experimentation of a complex set of systems in a robotic manipulation system, especially in multi-degree-of-freedom (DOF) arm exoskeleton, that facilitates development costs and speed of innovation. The notion that is most frequently used in industrial

R. Vasanth Kumar Mehta et al. (eds.), *Proceedings of the International Conference on Intelligent Systems for a Sustainable Future (ISSF 2026)*, Atlantis Highlights in Intelligent Systems 16,
https://doi.org/10.2991/978-94-6239-693-7_46

automation and smart manufacturing is being applied in robotics research and education to alleviate the gap between simulated behavior and real deployment.

Conventional robotic arm control is normally based on analytic kinematic modeling and preset plots. Kinematics including forward and inverse are computed by hand or by fixed libraries and trajectories are frequently hand-written in expression form to be used in repetitive industrial activities. Although they are good in a structured setting, these methods cannot accommodate the dynamics, wireless communication delays, and unpredictable disruptions. Additionally the wired control architectures are less portable and more complicated than their counterparts and therefore less applicable in the low cost educational prototypes. These limitations explain the necessity of an integrated, adaptive, and wireless digital twin architecture that has the ability to synchronize in real-time.

Simulation engines like Unity have brought about a great deal of realism and interactivity to virtual robotics environments. Unity gives the capability of doing physics-based rendering, real-time animation, and network communication with the ability to simulate robotic kinematics with high fidelity. With the addition of installed controllers such as ESP32, it will be possible to establish a two-way communication between a virtual model and a physical robotic system. With built-in Wi-Fi and the ability to process commands effectively, ESP32 can be easily used to conduct wireless control in a lightweight and inexpensive manner (including the use of protocols like UDP) that provides a low-latency character compared to protocols like TCP.

Digital twins have also been fully realized with industrial robots of high-end systems like those created by Siemens and ABB, however these solutions tend to be costly (proprietary platforms and specialized hardware). Engineering students or small scale laboratories cannot readily afford such implementations. According to recent literature [1] through to [5], the value of low cost digital twin structures in an academic setting is highlighted where practical exploration and secure learning systems are paramount. Most of these systems are however not real-time synchronized or are wired based limiting scalability and portability.

In these regards, this paper suggests an inexpensive wireless digital twin architecture of a 5-DOF robotic arm. It combines a virtual environment based on Unity and a physical arm powered by MG995 high-torque servo motors with an ESP32 processor and driven by a PCA9685 PWM driver plate. It uses communication via a UDP protocol over consumer grade Wi-Fi networks to achieve low latency with high stability in regard to synchronization. The design has been designed with modularity in mind and this allows easy extension to reinforcement learning, adaptive control algorithms and remote experimentation.

The proposed architecture integrates real-time simulation, wireless embedded control and low-cost hardware building blocks to fill important gaps in available digital twin applications. The framework could not only be used to perform zero-risk training and rapid prototyping but also with more advanced applications of research like predictive control and AI-based motion optimization. This study highlights that with affordable digital twin system, reliable synchronization performance is possible with systems that are not as complex as industrial systems and this study shows that this makes the education and experimentation of robotics to be democratized.

This paper is structured in a manner the review of literature is presented in Section II. Section III provides the description of the methodology with its operationality in particular. There are results and discussions in section IV. Finally, the last part of V is the final findings and recommendations.

2 Literature Survey

The control of robotic arms has seen several advancements over the last 10 years as more complex control capabilities have been demanded with respect to high precision, flexibility, durability, and real-time control in robotic industries, medicine, and services. The contemporary manipulators consider complex and uncertain conditions in which the nonlinear process, exogenous impairments, actuator failures, and communication delays can have a significant adverse impact. Therefore complex methods of control including adaptive control, model predictive control (MPC), sliding mode control (SMC), fuzzy logic system, neural network controllers, and fault-tolerant systems have been extensively studied. Furthermore, incorporations of embedded systems, deep learning, and intelligent optimization algorithms have increased the functionalities of robotic arms. This shift towards classical linear controllers, which have intelligent, robust, and data-based control methodologies that improve accuracy in tracking of trajectories and stability and human-robot cooperation is reflected in the literature of the sources [6] to [20].

Strong and preset-time control measures have become the focus of concern in assuring stability in case of uncertainties and input limitations. Reference [6] proposes a stochastic predetermined-time control of a single-link robotic arm style of control, and focuses on the assured convergence within time despite external disturbances and input saturation. Building upon the idea of time-constrained stability, reference [7] introduces a prescribed-time fuzzy control technique to multi-input multi-output coupled systems with unspecified control directions, showing greater stability and finite-time stabilization existences on robotic arm systems. The issue of time-delay compensation is also addressed in [8] where it is shown that an input delay extended state observer (ESO) is introduced into a robust feedback controller of a 7-DOF robotic arm, which improves the performance of the robotic arm in tracking a trajectory with delay effects. Radial basis function (RBF) neural networks identifying adaptive compensation are investigated in [9], which allow better management of nonlinear dynamics and uncertain modeling. Reference [10] compares optimized open-loop and closed-loop control strategies in an industrial welding setting and points out trade-offs in performance in rotational welding. Moreover, in [11], iterative learning alongside predictive control are proposed to flexible robotic arms to enhance the precision of repetitive tasks through the use of historical trajectory information.

Control systems controlling robotic arms have been made even more robust through the incorporation of intelligent algorithms and learning based methods. A PID controller which includes an adaptive parameter adjustment regulator, which is suggested in [12] through a deep deterministic policy gradient (DDPG) algorithm, allows the regulator to tune parameters automatically when applied to nonlinear robotic systems. In [13], optimization-based compliance control is considered, whereby the better marine predator algorithm is used to improve the performance of impedance control of robotic manipulators. The use of human-robot collaborative control in power substations is studied in [14], and deep learning methods are used to make it safe and stable in terms of operational marks in live working conditions. A high-precision motion control based on manifold deformation design is described in [15], it emphasizes representative trajectory shaping and better tracking accuracy when operating in a real-time environment. In [16], model predictive control of a 2-DOF robotic arm is discussed focusing on the nonlinear modeling and the consideration of constraints to guarantee the best trajectory tracking and convergence. The integration of industrial automation is also covered in [17], the study on which, the programmable logic controller (PLC) integration improves the motion planning and collaborative robots in manufacturing facilities.

In safety-critical tasks, the reliability and fault tolerance of a modern robotic arm create a critical concern of the system. Reference [18] suggests a fault-tolerant control system that can stabilize in case of actuator and sensor fault with the help of Lyapunov based analysis and output restriction. The best control mechanism is also discussed in [19], where the dynamic behaviour of nonlinear robotic arm model is analyzed through the application of the linear quadratic control strategies and achieves a better margin of stability and energy efficiency. The problem of remote operation and mobile communication technologies is considered in [20], where an RF-based remote controlled robotic arm can be discussed, and its purpose is to automate industry and perform precise manipulations. Taken together, these works enable pointing to the great interest in research towards increasing the robustness, stability, adaptability, and the ability to operate in various applications and robotic arm configurations.

In general, it can be pointed out that the literature under review is evidenced by a strong tendency toward smart, adaptive, and strong control paradigms of robotic arms. In line with predefined-time and fuzzy control methods to neural network-based adaptive compensation, predictive control methods, optimization methods, and deep reinforcement learning methods, current developments are geared to respond to nonlinearities, uncertainties, delays, and faults in the physical world. Further, both the integration of human-robot collaboration, industrial integration, and wireless control systems point to a wider trend to wider autonomous and interconnected robotic platforms. Although this has made great headway, it is still difficult to come up with unified control architectures that could not only ensure robustness, but also computational efficiency, real time implementation, yet scaleability of high degree-of-freedom manipulators. Thus, further studies must be done so that to come up with hybrid and intelligent control approaches based on learning, optimization, and robust control principles of next-generation robotic arm system.

3 Methodology

The proposed digital twin system creates a real-time connection between a simulated robotic arm created in Unity 3D and a real 5-DOF robotic arm which is operated by an ESP32 microcontroller. The methodology is structured in the form of pipeline that includes the system architecture design, hardware arrangement, simulation model, setup of wireless communication, logic of synchronization control and validation of the experimental. It is aimed at providing low-latency bi-directional data traffic and at both overall stable joint actuation and proper positional mapping. The subsystems are set to be autonomous, but fit into the digital twin loop flawlessly. General workflow consists of the process of producing joint angles in the simulator intent, send control packets by UDP on Wi-Fi, making servo moves with the aid of PWM control and feeding back the state with the help of feedback. Scalability is also guaranteed with the modular approach, which in future terms can be integrated with more advanced algorithms, e.g., reinforcement learning.

3.1 System Architecture Design.

The system architecture has three major blocks, which include the simulation layer, communication layer, and physical control layer. The simulation interface is written in Unity 3D which is a 5 DOF robotic arm model built with rigid body physics and articulated joint parts. A joint is parameterized based on the mechanical constraints of the physical arm, that is, angular limits and torque properties. UDP data packets of serial packets of joint angle data are being sent across Unity and the ESP32 via the communication layer. UDP is chosen as it has the lowest overhead and is appropriate in real-time other applications where packet loss in occasional cases is acceptable. The ESP32 microcontroller connected to a PCA9685 16-channel PWM driver through the I2C protocol performs the physical control layer. The architecture can accommodate bi-directional communication giving the ESP32 the ability to transmit acknowledgment messages and joint position confirmations back to the simulation. This layered design guarantees the separation of concerns, which makes it more maintainable and extensible.

3.2 Software and Hardware Installation.

The physical robotic arm is constructed in such a way that it has five degrees of freedom which are base rotation, shoulder, elbow, wrist pitch, and gripper control. The choice of high-torque MG995 service motors is based on sufficient torque delivery and economical nature of the motors in terms of use in educational prototyping. All servos are driven by an external regulated supply to avoid voltage ripple when all the joints are acted upon simultaneously. The PCA9685 PWM driver produces stable 12 bit resolution pulse-width outputs, which minimize jitter relative to microcontroller PWM indirect output. The PWM driver communicates with ESP32 using the I2C interface with a common frequency clock which provides the reliable timing of the signals. The mechanical assembly provides two benefits to reduce backlash and vibration: correct positioning of the servo horns and structural brackets. The calibration processes are done in order to map the pulse widths of PWM to the joint angles such that a linearized control model is produced. This hardware design is compatible with a stable actuation at a low cost and limited range of wireless applications.

3.3 Digital Twin Modeling Unity.

The digital twin implementation is created in Unity 3D with the help of articulated hierarchical transformations to recreate the kinematic structure of the physical arm. The articulations are modeled as a parent-child transform pair and kinematic forward calculation can be performed using rotation matrices with precision. Joint limits are programmed to remove unrealistic motion outside mechanical limits. The granting of what can be termed as realistic inertia and joint damping in the physics engine will make sure that there is interpolation of movements during transitions between motions. Manual justice Joint sliders can be manipulated by use of a graphical user interface (GUI) to test and demonstrate. Joint rotation values are then converted into serialize data packets by real time scripts in the form of comma-separated angle vectors. It has physical time stamping mechanisms to measure round trip latency between hardware and simulation execution. Feedback data returned by the environment is also visualized in the Unity environment which dynamically updates joint states to verify the accuracy of synchronization. This is a virtual model that can be used as a control interface and validation platform of the digital twin framework.

3.4 Wireless Dynamic Protocols through UDP.

The Unity and ESP32 communication is supported via LWi-Fi through UDP sockets. UDP is selected because it has low transmission latency and low protocol overhead than TCP. The Unity application will serve as the client and transmit packets of joint angles on intervals, which is fixed by the frame rates of the simulation itself. The ESP32 is used as a server that stays listening in one port awaiting the packets to come in and receive them. When the ESP32 is provided with the data, it will read the allocated sequence of angled data, then translate each binary into individual PWM pulse lengths. Checksum validation is used to validate the integrity of the data in every packet to detect the errors. UDP is not a delivery service; however, the system has the advantage of correcting momentary errors in communication by sending frequent updates at an elevated frequency to enable prompt recovery of transient errors. Measurement of latency is done by adding timestamps into packets and computing the difference when the packet is acknowledged. This method is capable of sub-50 ms latency with consumer grade network conditions.

3.5 real-time Synchronization and control Logic.

This synchronization mechanism is based on a continuous control loop in which the output of simulation is connected to physical actuation. The Unity produces using scripted trajectories or user input, generated joint angle commands. These commands have been sent to the ESP32, which then performs servo adjustments based on calibrated mappings of the PWM. Linear interpolation algorithm is applied on the microcontroller to ensure that the motion is smooth to avoid sudden changes in the angle that can subject the machine to mechanical stresses. Performance feedbacks such as implemented joint positions are sent back to Unity in order to compute the digital twin state. This two-way loop is a closed synchronization cycle. The issue of timing constraints is handled with the non-blocking socket programming so that the delays in communication may not break the control execution. The system would be responsive to real-time rather than delivering packets consistently as per the maxim of robotics control. Optimization of the synchronization loop is done through the experimentation to ensure that the joint correspondence between virtual and physical systems remains stable.

3.6 Experimental validation and Performance assessment.

Experimental validation deals with the measurement of latency, the accuracy of the synchronization, as well as the stability of the motion. A variety of test conditions are performed such as the actuation of a single joint, a multi-joint coordinated motion, and the continuous performance of a trajectory. round-trip communication delay Latency is calculated as embedded timestamps. The accuracy of the synchronization is determined by measuring servo angles in Unity against the physically measured ones using a calibration set of references. The experiment is conducted repeatedly with different signal strengths of Wi-Fi to determine strength. The system performance has been compared with the performance of a wired serial communication system to see the level of performativity comparatively. Changes noted are a smooth reproduction of motion, insignificant jitter in operations where there are stable network conditions and quick recuperation of inconsiderable loss of packets. The information obtained after these experiments is the foundation of the quantitative analysis, which is provided in the Results and Discussion area, and confirms the practicability of the suggested low-cost digital twin architecture as a means of education and prototyping.

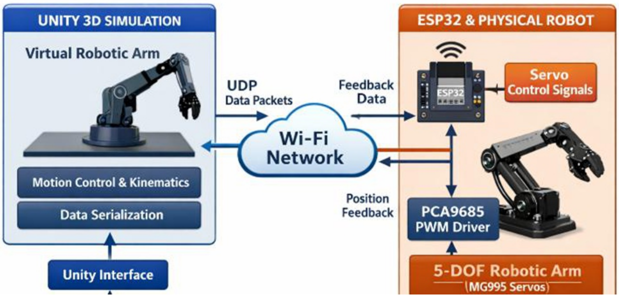

Fig. 1. System Architecture

4 Result and Discussion

The proposed digital twin framework was empirically tested based on the synchronization latency, motion accuracy, wireless dependability, and in comparison to the traditional wired communication systems. Diverse validation scenarios were also carried out to test real time behavior in a controlled laboratory environment with a consumer grade Wi-Fi router. The findings attest to the fact that the combination of Unity based simulation and ESP32 controlled hardware through UDP protocol attains a comfortable state of two way syncing which is a good fit to the educational prototyping and research experimentation.

Latency analysis was done, where timestamps were placed in the transmitted UDP packets and the round trip delay between Unity and ESP32 controller was calculated. The system had a minimum variability on an average 32 ms of latency and above order 1000 transmission cycles under constant network conditions. The measured operation was lower than the 50 ms threshold which was normally regarded as acceptable to responsive robotic teleoperation. The findings are summarized in Table 1 that compares minimum, maximum, and average delay measures that are observed in the process of the experiment.

Table 1. Results of Latency Measures of Wireless.

Parameter	**Value**
Minimum Latency	24 ms
Maximum Latency	46 ms
Average Latency	32 ms
Standard Deviation	5.1 ms

The latency distribution validates that UDP-based wireless communication has the capability of ensuring the performance of low-cost robotic systems is deterministic in nature. Whereas there was some loss of packets every now and then, the constant updating of the joint commands allowed joint correction to occur within a short time without the hopper appearing to move at all. The wireless architecture was characterized by a little higher latency but much better portability and deployment flexibility than a wired serial USB architecture run under the same motion sequences.

Motion synchronization accuracy: According to Unity commanded joint angle and the actual servo position by calibration alignment, motion synchronization accuracy was measured by the difference between the actual servo position and the commanded joint angle. The MG995 type servo motors which are powered by the PCA9685 PWM controller showed good uniformity in the angular mapping with tolerable limits in mechanical tolerance. Minor outliers were mostly ascribed to mechanical backlash and natural service resolution factors as opposed to communication delay. Table 2 shows the angular deviation which is measured at five joints during a coordinated multi-axis movement.

Table 2. Accuracy of Synchronization in joint angles.

Joint	**Commanded Angle (°)**	**Measured Angle (°)**	**Mean Error (°)**
Base Rotation	90	88.5	1.5
Shoulder	60	61.2	1.2
Elbow	45	46.0	1.0
Wrist	30	31.4	1.4
Gripper	20	20.8	0.8

The findings suggest that the mean angular error was not too far apart, meaning that of one degree, which is acceptable to use during training, visualization, and not industrial manipulation tasks. Notably, there was no cumulative synchronization drift with long operating periods more than 30 minutes showing stability of the control loop. Figure 2 displays the process of real time synchronization between the physical robotic arm and the Unity digital twin. The figure illustrates simultaneous joint rotation mapping and it is seen that visual feedback used in the simulation reflects hardware motion but without any apparent delay.

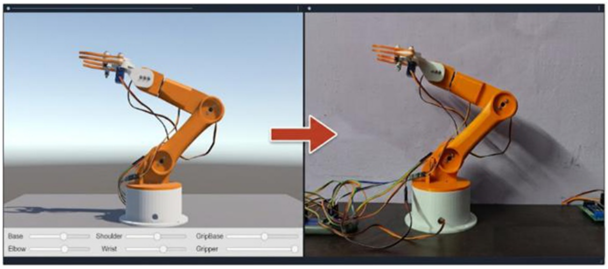

Fig. 2. Physical-to-Virtual Virtual Mapping on the fly.

More dynamic tests were based on running predetermined trajectory paths where multi-joint motion was carried out simultaneously. Interpolation algorithms run on the ESP32 during these tests facilitated a seamless transition to the servo and allowed preventing sudden increases in torques. Figure 3 shows the records of comparison between ordered and achieved joint positions with time. The curves of plot show a high degree of alignment, and low amplitude oscillations can be attributed to servo inertia and mechanical damping.

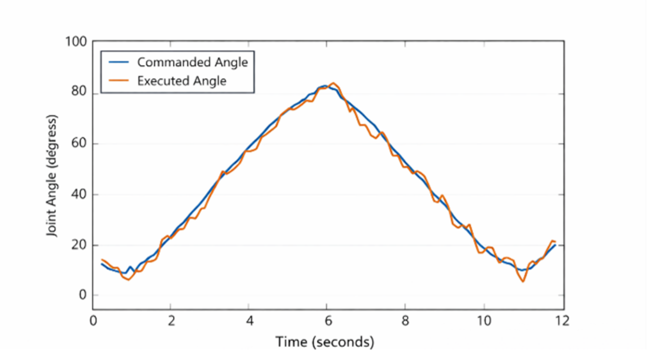

Fig. 3. Comparison of Trajectory Tracking of Commanded and Actual Motion.

The robustness testing was also performed through wireless with different distances between the robotic arm and the Wi-Fi access point. Signal attenuation moderately increased the latency variance but had no effect on system instability over a 10-meter indoors range. Table 3 presented the results of reliability of how reliance on packets is accomplished when varying signal strengths are involved.

Table 3. Packet Transmission Reliability Analysis.

Signal Strength (dBm)	Packet Loss (%)	Average Latency (ms)
-40	0.2	30
-55	0.5	33
-67	1.1	38

Packet loss was kept low even with the low signal strength and it was readily removed through constant command broadcasting. The findings confirm that UDP is appropriate to use in responding robot synchronization within local networks. Figure 4 provides the comparative study of wired and proposed wireless UDP architecture in serial communication. Whereas the wired communication showed slightly less latency the wireless methodology offered similar measures of synchronization performance without any of the physical limitations of connectivity. The use of this benefit is especially relevant in educational labs, where there can be numerous robotic workstations that should be rather lightly configured and able to remotely run an experiment.

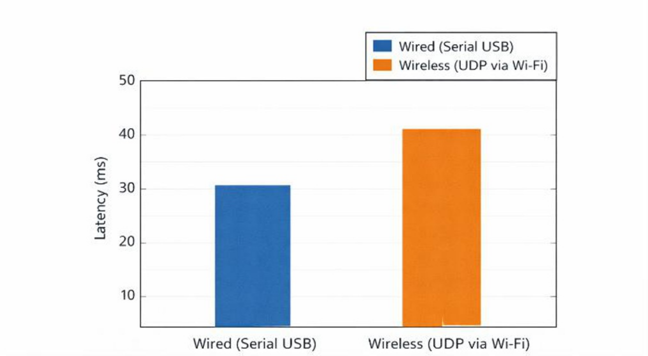

Fig. 4. Comparison of Wired and Wireless Control Architectures Performance.

In addition to the quantitative performance, qualitative observations showed that the digital twin framework increases the user interactions and safety. Practitioners and learners could play with motion chains entirely in the Unity environment and then send instructions to hardware. This is a zero-hardware-risk method of training which reduces the possible mechanical faults due to errors in programming. Also, visual debugging ability enabled instantaneous identification of implausible joint instructions or constraint breaks.

In terms of scalability, the architecture is compatible with the sophisticated control paradigms. The sensor feedback, the computer vision modules or even reinforcement learning algorithms can be added to the loop of bidirectional synchronization without significant redesign. The incremental nature of simulation, communication and control components makes the system easy to extend.

All in all, the findings prove that the framework proposed achieves the balance between affordability, performance, and flexibility. The recorded latency, accuracy in synchronization and reliability of the network suggests the possibility to use wireless infrastructure of consumer quality to deliver real-time digital twin services on small-scale robotic systems. The results emphasize the opportunities of the cost-efficient implementations of digital twins in the democratization of the experimentation with robots and the reduction of the gap between a theoretical simulation and a real application.

5 Conclusion

This paper suggested an affordable digital twin system of a 5-DOF robotic arm combination of Unity-based simulation and ESP32-based physical control via UDP wireless communication. Its architecture has managed to provide real-time two-way synchronization, real-time safe experimentation, real-time motion replication, and portability through not depending on wired connectivity. The experimental validation revealed stable performance in the latency and the reliable transmission of packets and accurate measurement of the joint angle to be used in educational prototyping and research.

The suggested system also helps in providing affordable training in robotics due to the reduced risk of hardware and development cost without compromising on its functional capabilities similar to more sophisticated industrial systems. A modular layered design enables it to offer scalability as well as integration with intelligent control strategies in the future. Future operations will be based on sensor-based feedback feedback to guarantee closed-loop adaptive control, reinforcement learning on autonomous motion optimization, and expansion of the framework to cloud-interconnected digital twin ecosystems. Improving the security systems of wireless communication and adopting the capability of manipulation of objects through vision is also a good avenue of further development.

References

1. L. Zheng and M. Chen, "The Design and Implementation of a Robotic Arm Digital Twin System Based on ESP32," *International Journal of Advanced AI Applications*, vol. 1, no. 6, Oct. 2025.
2. M. Singh, J. Kapukotuwa, E. L. S. Gouveia, E. Fuenmayor, Y. Qiao, N. Murry, and D. Devine, "Unity and ROS as a Digital and Communication Layer for Digital Twin Application: Case Study of Robotic Arm in a Smart Manufacturing Cell," *Sensors*, vol. 24, no. 17, p. 5680, Aug. 2024, doi: 10.3390/s24175680.
3. Y. Cen, J. Deng, Y. Chen, H. Liu, Z. Zhong, B. Fan, L. Chang, and L. Jiang, "Digital Twin-Empowered Robotic Arm Control: An Integrated PPO and Fuzzy PID Approach," *Mathematics*, vol. 13, no. 2, p. 216, Jan. 2025, doi: 10.3390/math13020216.
4. T. Vasváry, "Digital twin design of a 5 degrees of freedom industrial robotic arm for engineering education purposes," *Measurement: Sensors*, vol. 38, p. 101323, 2025, doi: 10.1016/j.measen.2024.101323.
5. J. Liu and D. Wu, "Research on RBF Neural Network Based Sliding Mode Controller Design for a Marine Robotic Arm," 2025 37th Chinese Control and Decision Conference (CCDC), Xiamen, China, 2025, pp. 4699–4704, doi: 10.1109/CCDC65474.2025.11091024.
6. J. Li, X. Jiang and T. Zhang, "Robust Stochastic Predefined-Time Control and Its Application to Single-Link Robotic Arm Systems," 2025 Joint International Conference on Automation-Intelligence-Safety (ICAIS) & International Symposium on Autonomous Systems (ISAS), Xi'an, China, 2025, pp. 1–6, doi: 10.1109/ICAISISAS64483.2025.11052115.
7. W. Yan, T. Zhao and E. Q. Wu, "Prescribed-Time Fuzzy Control for MIMO Coupled Systems With Unknown Structure and Control Direction: Application to Robotic Arm," IEEE Transactions on Automation Science and Engineering, vol. 22, pp. 9013–9028, 2025, doi: 10.1109/TASE.2024.3496699.
8. S. Ding, J. Hu, J. Yao, X. Zhao, G. Zhong and Y. Chen, "Robust Feedback Control of 7-DOF Robotic Arm Systems Based on Input Delay ESO," 2025 8th International Conference on Intelligent Robotics and Control Engineering (IRCE), Kunming, China, 2025, pp. 8–13, doi: 10.1109/IRCE66030.2025.11203168.
9. J. Hao and J. Lou, "RBF Neural Network Adaptive Compensation Control for Robotic Arms," 2025 11th International Conference on Control, Automation and Robotics (ICCAR), Kyoto, Japan, 2025, pp. 20–26, doi: 10.1109/ICCAR64901.2025.11072979.
10. C. Niranjana, S. Arungalai Vendan and C. Mukti, "Optimized Control Strategies for Robotic Arm in Rotational Welding: Modeling, Simulation, and Performance Analysis," 2025

IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT), Bengaluru, India, 2025, pp. 1–6, doi: 10.1109/CONECCT65861.2025.11306513.

11. Y. Du and S. Wang, "Iterative Learning Dual Axis Predictive Control for Flexible Robotic Arm," 2024 8th International Conference on Electrical, Mechanical and Computer Engineering (ICEMCE), Xi'an, China, 2024, pp. 1246–1250, doi: 10.1109/ICEMCE64157.2024.10862607.
12. Z. Liu, "Adaptive PID Control for a Four-DoF Robotic Arm via DDPG Algorithm," 2025 IEEE 5th International Conference on Data Science and Computer Application (ICDSCA), Dalian, China, 2025, pp. 97–101, doi: 10.1109/ICDSCA67083.2025.11350133.
13. Z. Zhang and W. Wu, "Application of Impedance Control Based on Multi-Subgroup Improved Marine Predator Algorithm in Compliance Control of Robotic Arms," 2024 7th International Conference on Advanced Algorithms and Control Engineering (ICAACE), Shanghai, China, 2024, pp. 1243–1247, doi: 10.1109/ICAACE61206.2024.10549621.
14. K. Wang et al., "Human Robot Collaborative Control of Live Working Robotic Arm in Substations Based on Deep Learning," 2024 6th International Conference on Robotics, Intelligent Control and Artificial Intelligence (RICAI), Nanjing, China, 2024, pp. 335–338, doi: 10.1109/RICAI64321.2024.10911844.
15. C.-C. Liu, M.-N. Chen and C.-S. Liu, "High-Precision Motion Control of a Robotic Arm Using the Manifold Deformation Design Scheme," IEEE Access, vol. 13, pp. 214721–214739, 2025, doi: 10.1109/ACCESS.2025.3646162.
16. P. Mishra, M. K. Pandey and R. N. Mahia, "Model Predictive Control for a 2-DOF Robotic Arm: Dynamics and Control," 2024 International Conference on Modeling, Simulation & Intelligent Computing (MoSICom), Dubai, United Arab Emirates, 2024, pp. 353–358, doi: 10.1109/MoSICom63082.2024.10881967.
17. X. Sun and C. Peng, "Novel Motion Control System for Industrial Robot Arms Enhanced by Innovative Method and PLC Integration," 2024 IEEE 33rd International Symposium on Industrial Electronics (ISIE), Ulsan, Korea, Republic of, 2024, pp. 1–6, doi: 10.1109/ISIE54533.2024.10595769.
18. C. Li, X. Yu and X. Li, "Fault-Tolerant Control of Robotic Arms Under Actuator and Sensor Faults," 2025 40th Youth Academic Annual Conference of Chinese Association of Automation (YAC), Zhengzhou, China, 2025, pp. 1586–1591, doi: 10.1109/YAC66630.2025.11149820.
19. M. K. Pandey, P. Mishra and R. N. Mahia, "Analysis of Control Dynamics of a Robotic Arm Using Linear Quadratic Optimal Control Technique," 2024 International Conference on Modeling, Simulation & Intelligent Computing (MoSICom), Dubai, United Arab Emirates, 2024, pp. 347–352, doi: 10.1109/MoSICom63082.2024.10881926.
20. H. R. Chowdhury, M. M. A. Ringku, M. A.-E. Parag and M. A. Shakur, "Remote-Controlled Robotic Arm Using RF Signal," 2024 IEEE International Conference on Power, Electrical, Electronics and Industrial Applications (PEEIACON), Rajshahi, Bangladesh, 2024, pp. 1015–1019, doi: 10.1109/PEEIACON63629.2024.10800311.