Journal of Engineering and Sustainable Development
Vol. 30, No. 03, May 2026
https://jeasd.uomustansiriyah.edu.iq/index.php/jeasd
ISSN 2520-0917
Research Article
https://doi.org/10.31272/jeasd.3602

Design and Test an Educational Model of a Robotic Arm on an Experimental Platform

Ammar Fadhil Hussein AL-Maliki[1], Ehsan Sabah Al-Ameen[1*], Rouzbeh Nouhi Hefzabad[2], Abeer K. Abd[3]

[1]Mechanical Engineering Department, College of Engineering, Mustansiriyah University, Baghdad, Iraq
[2]School of Railway Engineering, Iran University of Science and Technology, Tehran, Iran
[3]Iraqi Ministry of Transport, Baghdad, Iraq

[*] Email: apehsanameen@uomustansiriyah.edu.iq

Article Info

Received 05/08/2025
Revised 02/03/2026
Accepted 03/03/2026

Abstract

A robotic arm with five degrees of freedom has been developed for use in learning laboratories. The arm's movements were designed and simulated in SOLIDWorks and LabVIEW. The LabVIEW Softmotion was used to impose motion constraints on the model. An Arduino Mega microcontroller was used to manipulate the servo motors in the joints. After the design and simulation, the system was implemented on a laboratory setup. The following steps were then specified to verify the results of the LabVIEW-designed model with the real physical laboratory. These tests were the transfer of mass between two predetermined points along a certain direction. Its limitations were actuator torque and arm stability limitations. Lastly, the comparison between the physical setup and the model showed high accuracy and performance in the execution of the movements. This study's findings show a qualitative enhancement in education laboratory equipment and extend to the establishment of an integrated learning system that connects the simulated model to the real laboratory systems.

Keywords: Educational Robot, Experimental Platform, LabVIEW, Manipulator, Robotic Arm

1. Introduction

Robotic arms, along with control equipment, are used for tasks such as welding, cutting, painting, and packaging in industries including automotive, medical, agricultural, and pharmaceutical sectors. These robotic arms replace high-risk human activities in these fields. Robotics encompasses a wide range of scientific disciplines and enhances skills such as reasoning, problem-solving, generalization, communication, and collaboration [1], [2]. Setting up a robotic system requires knowledge of mathematics, computer science, mechanical engineering, and electrical engineering. With the expanding application of robotics in various industries, the demand for education in this field has increased. Education in robotics cannot rely solely on theoretical discussions; it must be complemented by practical activities and the integration of both theory and practice [3]. Consequently, robots have been incorporated into educational environments, transforming traditional teaching methods. Educational robots have generated significant interest and gained increasing attention, contributing to the development of a unique generation of artificial intelligence [4]. Open source platforms such as Raspberry Pi and Arduino have become more accessible, allowing users -both novice and expert- to create projects ranging from simple electronic devices like LED displays to advanced robotic systems. As a result, educational platforms have diversified and become commercially available, including Lego Mindstorms and VEX Robotics, which can be integrated into existing curricula at various academic levels [5], [6]. Additionally, there are open-source designs related to the structures of robotics and controllers for robotic platforms [7]-[11]. To overcome financial challenges and promote Maker Culture, many studies have demonstrated that learning is most effective when students construct an entire project from the ground up, enabling them to understand how each movement occurs and how to control it. This work proposes an experimental platform for educational purposes, aimed at training students on industrial robotic manipulators in specialized colleges. Due to the rapid and significant development and increasing importance of industrial manipulators in the production process—given their substantial impact on production speed—a programming system was developed to control a 5 DOF robotic arm and simulate its movement using the Laboratory Virtual Instrument Engineering Workbench (LabVIEW) and SolidWorks software. The robotic arm in the College of Engineering laboratories, originally manufactured by Lab-Volt, was replaced with another arm capable of performing similar tasks. An Arduino Mega was used as the main controller to connect the actuators to the microcontroller without the need for additional circuits. Numerous published research papers present various methods for controlling robotic arms and address this field from multiple perspectives; some of these will be reviewed herein.

Hao and Ru [12] developed a laboratory setup featuring an open 5-degree-of-freedom robot arm and its control algorithms. This system provides easy access to both software and hardware components, with control implemented through MATLAB algorithms and a DSP hardware controller. Consequently, it serves as a unique experimental platform for research purposes. Neto et al. [13] modernized and industrialized a 5-degree-of-freedom SCARA-type robot arm for educational use and inventory management applications. Their approach focused on optimizing performance and cost. This model introduces an innovation by enhancing the conventional fixed-base arm structure with horizontal linear motion, thereby reducing the workload. The system also includes the development of the mechanism and firmware programmed in the C language. Jin et al. [5] proposed a technique for designing a robotic arm system, encompassing movement analysis and control algorithm development. The method presented in this study supports robotics enthusiasts, particularly beginners, in exploring and practicing robot-building techniques using simple materials. Oluwajobi and Oridate [1] suggested the development and design of an educational robotic arm with 5 DOFs. In their research, servo motors were used to operate all arm joints and perform inverse kinematics. To visualize joint movements, a simulation of the robotic arm was conducted using the MATLAB Robotics Toolbox. A servo controller managed the operation, and the control software was developed in C#. Zamarron et al., [4] explained how to design a 4 DOFs robotic arm with CAD, CAM, and Robotics Toolbox of MATLAB and other electronics to resolve any kinematics issues. This cheap system is applicable to research in research laboratories and may be applied in research, including design, manufacturing, electronics, and mechatronics research. The system improves the real-world experience of the students with their theoretical skills. Mirand-Flores et al. [14] came up with an innovative approach to designing and implementing a manipulative kit. The kit provides the assembly with immense flexibility, which can be configured to different structures to impart the concepts of rotation, homogeneous transformation matrices, as well as forward and inverse kinematics. The kit of manipulation is created through 3D printing, which enables the students to have it. Wang et al. [15] carried out research on the effectiveness of educational robots in enhancing the learning outcomes of students. They reviewed randomized trials of learning robot interventions with a concentration on learning outcomes. On the basis of such analysis, the effectiveness of robots in education was proved. Murzynski et al. [16] developed and built a low-cost and reconfigurable robotic arm to teach robotics and automation in different industries. Such an arrangement made students highly usable. Vega and Perez [17] created an educational ROS2 based robotic arm. In this system, low-cost components that have high teaching abilities are employed in teaching programming and control. Zeng et al. [18] introduced a six-degree-of-freedom model of a robotic arm to be used in education. The effectiveness of this model with a set of students was measured by using tests which tested the improvement of their computation skills. Zajc et al. [19] designed a cost-effective, open-source educational platform that will be used to ease the learning and teaching of STEM concepts at different levels of education. The user testing showed that there was a considerable increase in motivation, learning, and conceptual understanding. This research field has also been covered in many other studies employing novel methods and approaches [20]- [23].

The platform suggested in this paper is intended to be used in the laboratories of engineering schools as an open workspace, where the students will have an opportunity to gain skills and learn to operate with the robot hardware and software systems. The main benefit of the presented work is the creation of a platform which is easy to design and use, where the courses could be customized to the educational requirements of the corresponding fields of science. This work provides a qualitative improvement to the past attempts in the field by combining theoretical knowledge with practical skills, hence equipping students better to meet the requirements of a modern workforce in the areas of automation and smart manufacturing. In that vein, the 2nd and 3rd Sections outline the process of modeling and implementation of the robotic arm, and the assessment of the methodology of the project. The 4th section contains the conclusions.

2. Methodology

The aim of this study is to develop a low-cost, efficient tool for training students in specialized industrial robot laboratories. This tool is expected to significantly enhance the creative abilities of students and other interested individuals. The research methodology and its fundamental framework are outlined below.

2.1. Materials and Methods

This setup utilizes a LabVIEW-based system for training, testing, measurement, and automation applications that depend on graphical programming. LabVIEW is renowned for its clarity and simplicity. Unlike traditional text-based scripting languages, it does not require a textual syntax. The platform offers a comprehensive suite of functions for design, numerical analysis, data acquisition, and processing. Additionally, LabVIEW provides various development tools, including artificial intelligence features for managing cloud logic. It supports programming in a single language across multiple development environments, such as computers, programmable logic controllers, and mobile devices. Furthermore, LabVIEW includes libraries for robotic arms that can calculate the instantaneous position of the arm based on motor angle values and determine the necessary motor angles to reach a specific point. In this work, SolidWorks software was used to design the robotic arm, which was controlled through LabVIEW using the SoftMotion library.

2.2. Governing Research Tools

In this work, several programs were integrated to achieve the desired final movement. LabVIEW 2018 was used to control the constructed design and move it to specific locations. Subsequently, LabVIEW 2019 was employed to control the Arduino chip connected to the robotic arm. Through this setup, the coordinates to which the robotic arm is intended to move are fed into the LabVIEW 2018 program. The complete implementation plan for the educational robotic arm model is illustrated by the block flow diagram shown in Fig.1.

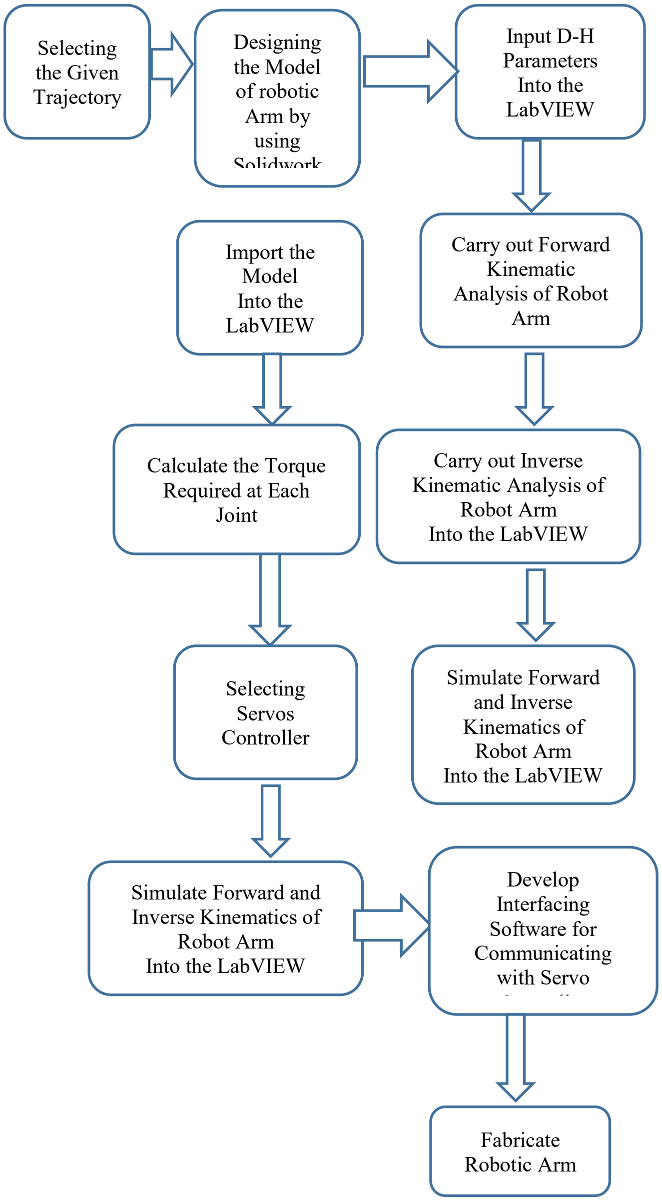

Figure 1. Block diagram illustrating the implementation of the robotic arm.

A comprehensive presentation on LabVIEW programming will now be provided for both forward kinematics and inverse kinematics.

2.2.1. Forward kinematics

Firstly, the user interface was designed using LabVIEW and includes the following components:

- **Serial Port:** An input field to specify the serial port to which the Arduino chip is connected.
- **Joint Position:** A table was created containing five fields. Each field holds a specific value representing the actuator angle corresponding to that field number (for example, the first field corresponds to the first actuator). It is important to note that the first actuator controls the base.
- **Knob:** The knob controls the degree to which the handle at the end of the arm is closed.
- **End Effector Transform:** It consists of a table displaying the current coordinates of the arm.
- **Stop:** The button used to stop the control.

Fig.2 illustrates the user interface for forward kinematics.

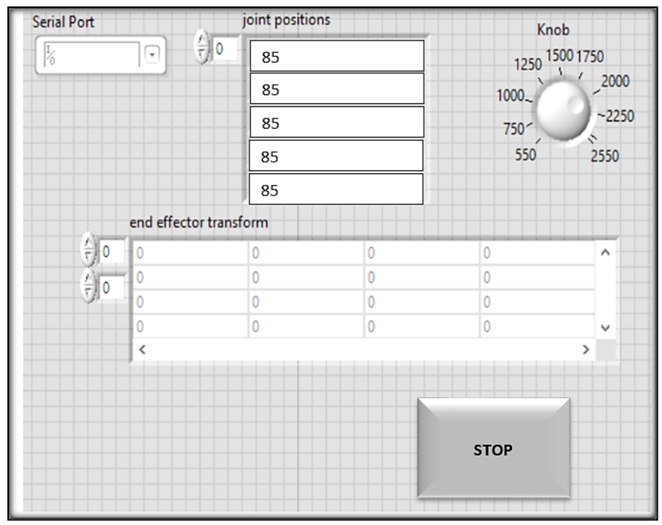

Figure 2. User interface of forward kinematics.

After the final programming interface is designed, the data for the serial arm is entered into the LabVIEW environment. This data includes both the distances between the joints and the directions of motor rotation.

Twist angle: This is the angle of twist of the joint. Its value is 90 degrees when the joint axis is parallel to the vertical Y-axis, and zero when the joint axis is parallel to the horizontal X-axis. Length: This refers to the length of the segment connecting two joints, or between a joint and the base, or between a joint and the end-effector. Rotation angle: This is the joint's rotation angle. Offset distance: This represents the angle of deviation. Fig. 3 shows the programming interface for forward kinematics.

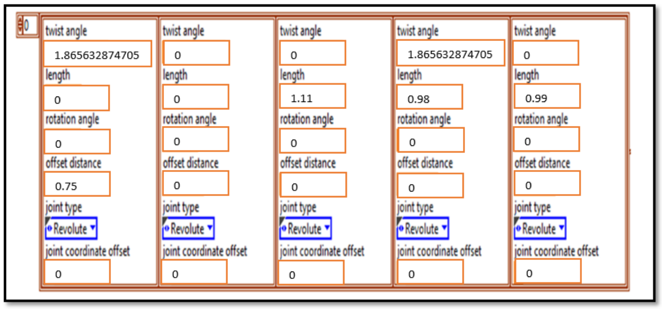

Figure 3. Program interface of forward kinematics.

As the next step, the predefined data is entered into the serial arm construction box, as shown in Fig. 4.

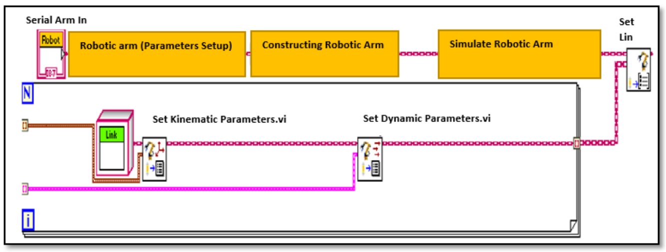

Figure 4. Programming the serial robotic arm construction.

Next, the program (Pick and Place) that the robot will execute is written, as shown in Fig. 5.

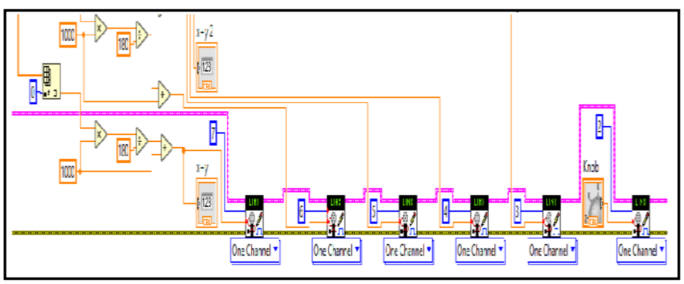

Figure 5. Programming part for converting angle values into values suitable for servo motors.

2.2.2. Inverse kinematics

The user interface for the reverse kinematics is designed as illustrated in Fig. 6. This interface includes the following components:

- **Start Position:** An input to specify the coordinates from which the movement will begin.
- **End Position:** An entry used to specify the current coordinates where the movement will end.
- **Current Position:** Display of the current position.
- **Max. Step and Number of Trajectory Points:** This section specifies the variables related to the number of steps.
- **Stop:** Button used to stop the motion.

The kinematic and dynamic equations of the robotic arm with five degrees of freedom are presented in Appendix A.

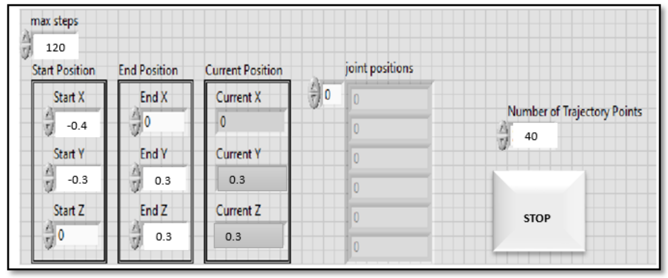

Figure 6. User interface of inverse kinematics.

After defining the arm as a serial manipulator and entering all the necessary data, a program was developed for inverse kinematics. This program calculates and displays the actuator angle values required to reach the coordinates of a specified point. These values are then applied to the actuators. The process begins by converting the coordinates of the starting point and the target point into two matrices. The program then attempts to find a common solution between these matrices, which corresponds to the actuator angle values. The programming part for converting user inputs into a matrix form and for determining the actuator angles is shown in Fig. 7 and Fig. 8, respectively.

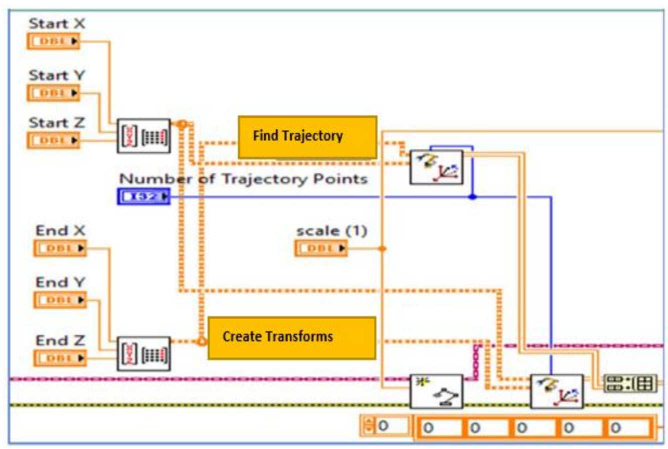

Figure 7. The programming part for converting user inputs into matrix form.

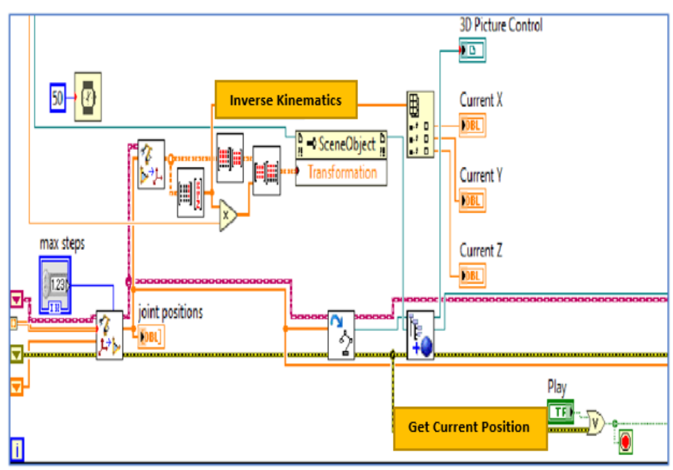

Figure 8. The programming part for determining the actuator angle values.

2.3. Simulation of the Robotic Arm

In this work, a robotic arm was designed using SolidWorks software. The designed robotic arm closely resembles the implemented model. Each component of the robotic arm was created individually and then assembled using the constraints within the SolidWorks environment. The integration of the LabVIEW program with SolidWorks relies on a motion analysis library that enables monitoring of the three-dimensional design's movement in space. Fig. 9 shows the robotic arm designed in SolidWorks.

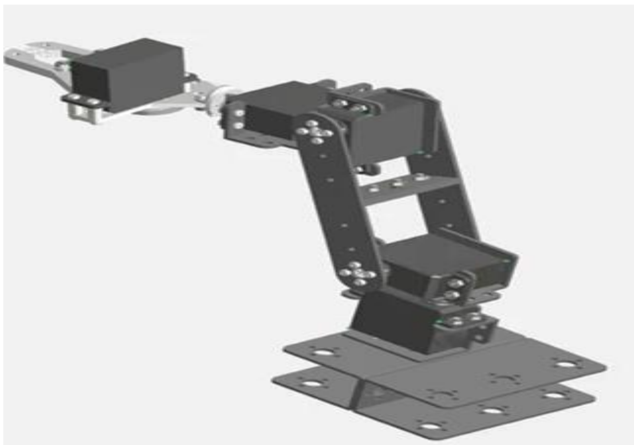

Figure 9. The robotic arm designed using SolidWorks.

To send movement commands from LabVIEW 2019 to LabVIEW2018, the Virtual Serial Port Tools program was used. This software enables the creation of virtual COM ports on the personal computer with high accuracy, where these ports function as bridge-type communication channels.

3. Test the Designed Model

The experimental work in this study involved conducting two tests on the designed model, outlined below:

3.1. The First Test

In this test, the model's ability to accurately determine the effective head position of the structure was verified when the actuator angle values were input by the user. The final position of the structure, designed in virtual reality, was also compared with the final position of the structure constructed in the real world. Fig.10 shows the interactive interface in the LabVIEW program.

3.2. The Second Test

The second test was conducted to investigate the robotic arm's ability to pick up an object at a specific location (a point with known coordinates) and then move it to another location (another point with known coordinates). Additionally, a comparison was made between the movement in the real environment and the movement in the model designed in the virtual environment. The procedure for this test is illustrated in Fig. 11, which depicts the test sequence from the initial object pickup to its release, followed by the arm returning to repeat the process. The primary specifications of the robotic arm are detailed in Appendix B.

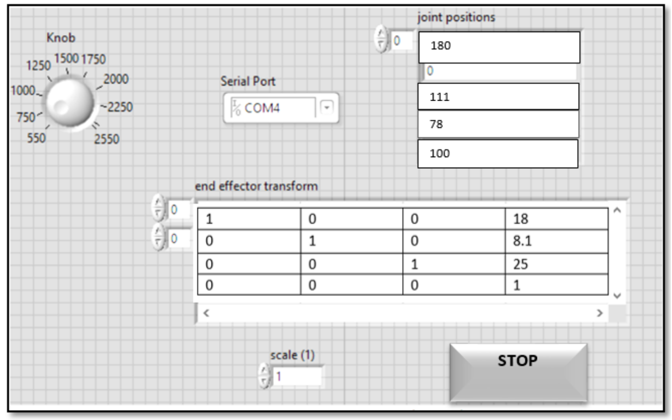

Figure 10. Communicative interface for entering actuator angle values and displaying the coordinates of the arm's current position.

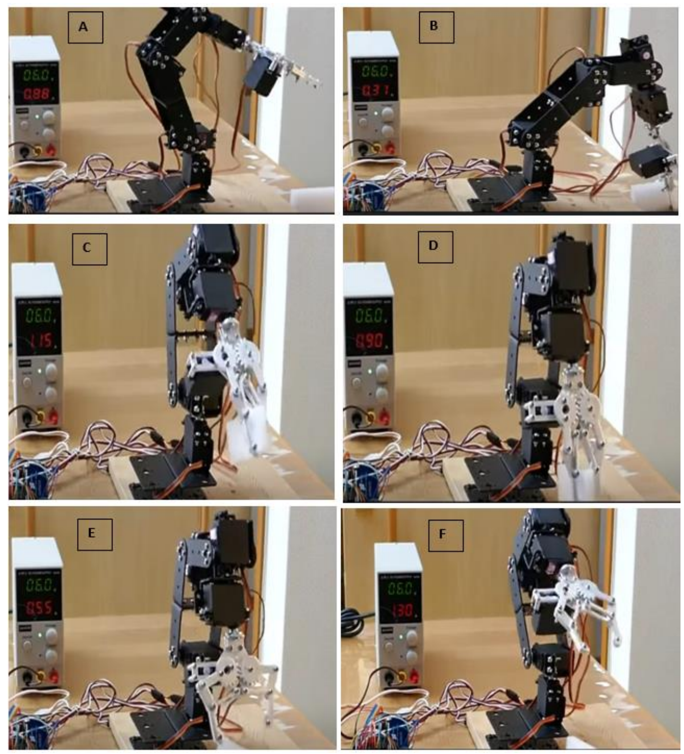

Figure 11. Positions of the robotic arm during testing in the real environment.

Fig.12 illustrates the stages involved in capturing and releasing an object within virtual reality.

4. Results and Discussion

Students may develop automation, intelligent manufacturing, operation, programming, and control system integration skills through professional educational models. Robot learning systems are high-tech devices that have revolutionised the learning institutions, especially science laboratories. These tools are very effective in terms of deep understanding, critical thinking, and interactive prospects of engaging the learners to put into practice what they have been taught. The flexibility of the performance of the robot is remarkable, as it has smooth performance and accuracy in both simulation and experimental conditions. Mass transfer between two given positions running along a particular path was put to the test. Some of the constraints were limited actuator torque and robotic arm stability. Comparison of the setup and the model showed that it was very accurate and performed well in performing movements. With appropriate setting of torque limits, the robot is able to generate smooth and accurate control of its movements, particularly in those jobs that demand delicate balance, e.g., grasping objects or handling delicate tools. Over-torque causes energy wastage and strain to the motor.

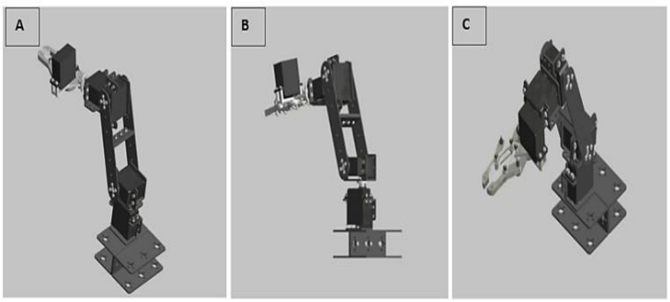

Figure 12. Positions of the robotic arm during testing in the simulation environment.

Modification of torque limits enhances the effectiveness of the system and energy savings. Another important consideration that is vital in guaranteeing safety and accuracy in the operation of the robot is to control vibrations during robot operation, especially at high speeds. Vibration control enables the stabilization of dynamic behavior and enables the robot to work faster without compromising accuracy. The findings indicated that there was high agreement between the results of the simulation and experimental data, albeit with a slight deviation in the final position of the end-effector. Such deviations are attributable to the existence of the fact that execution of the path between two points is dependent on the angular displacement of arm links coded to operate the robot, which influences the angular velocities and acceleration of the links. Also, vibrations caused by the actuators at both ends of the joint along the path affect the robotic arm. These effects may be said to be negligible, considering the high correlation between the practical and the simulation results. This is mostly because the lightweight nature of the links in the educational robots is a major factor that affects the accuracy of the course execution, even for a mass of 70 grams or less. A number of procedural considerations were carried out in this work, including balancing the mass carried and the torque necessary to move it, particularly when it was lifted off the axis of rotation. Acceleration and speed limits were also put in place because control of movement or abrupt stops was not easy with objects that had high inertia. Precision in location is significant for robotic tasks. The stability of the robot during movements guarantees precision in the path execution and accuracy of the task execution. With this in mind, path design and control algorithms took into consideration the kinematic characteristics of the robot (inertia, load, friction), acceleration and velocity limits, and, lastly, damping parameters to minimize vibrations. Calibration is also an important factor in obtaining desirable results since a slight error in calibration may lead to inaccuracies in the final position. Software integrity checks should be done continuously in order to identify coding mistakes that may cause unintended limitations. Besides this, constant check-ups are performed to detect and rectify wrong responses as they occur in the process. An experimental model presented in Fig. 11 and the virtual reality model presented in Fig. 12 exhibit that the virtual design provides a true replication of the experimental system. This can be seen in the motion mechanics as shown in the two figures, and the striking similarity between the two figures adds to the validity of the model designed.

The research offers a secure learning setting to trial ideas without the threats and expenses of hands-on labs. The technology has been a major contributor to the quick improvements in artificial intelligence and engineering, and has become a potent workstation in learning through the use of laboratories. It brings a perfect chance to receive theoretical ideas through a realistic presentation. Instead of focusing on theoretical knowledge, students will have a chance to program a robot, monitor its performance, analyze the results, and be more engaged in the learning process. Based on these benefits, and as the current curricular models are based on hands-on and active involvement, educational robots have become the main elements of educational institutions. Such high-tech educational resources introduce technology in the educational process through scientific experimentation and research. They also equip students with the digital workforce by equipping them with core skills in engineering and smart industries, including deep learning, automation, artificial intelligence, and the Internet of Things. Finally, this tool can promote a smart model of education that is aimed at accommodating adaptive learning and facilitating the elaboration of integrated curricula.

5. Conclusions

This work proposes an educational platform designed to train students on industrial robotic manipulators in specialized colleges. A 5 DOFs robotic arm was modeled specifically for educational purposes. The robotic arm was modeled using the SolidWorks software, accompanied by a simple and user-friendly control algorithm and a high-level programming interface developed with LabVIEW. This integration enabled precise control of the servo motors through commands sent via an Arduino microcontroller. The implementation and performance testing of the robotic arm demonstrated its suitability for laboratory use in training students to program and execute various movements, such as picking, editing, spot welding, and painting surfaces. We believe this platform will broaden opportunities for educational projects across multiple disciplines, benefiting laboratory systems in particular and the educational system in specialized colleges and scientific centers in general.

Acknowledgements

The authors may express their appreciation to Mustansiriyah University for the academic support for their research.

Contribution of the Authors

Ammar Fadhil Hussein AL-Maliki and Ehsan Sabah Al-Ameen proposed the research problem.
Rouzbeh Nouhi Hefzabad and Abeer K. Abd developed the theory and performed the computations.
Ammar Fadhil Hussein AL-Maliki performed simulation analyses and wrote the article and supervised the findings of this work.
All authors discussed the results and contributed to the final manuscript

Conflict of interest

The authors declare that there are no conflicts of interest regarding the publication of this manuscript.

References

[1] A. O. Oluwajobi and A. A. Oridate, “Design and Development of an Educational 5-DoF Robotic Arm,” *International Journal of Robotics and Automation Technology*, vol. 6, pp. 55–65, 2019. doi: https://doi.org/10.31875/2409-9694.2019.06.7.

[2] F. B. V. Benitti, “Exploring the educational potential of robotics in schools: A systematic review,” *Computers and Education*, vol. 58, no. 3, pp.978–988,2012. doi: https://doi.org/10.1016/j.compedu.2011.10.006.

[3] M. S. Khine and S. Areepattamannil, Eds., *Robotics in STEM Education: Redesigning the Learning Experience*. Cham, Switzerland: Springer, 2017. doi: https://doi.org/10.1007/978-3-319-57786-9.

[4] D. S. Zamarrón, N. I. Arana de las Casas, E. García Grajeda, J. F. Alatorre Ávila, and J. U. Naciff Arroyo, “Educational robot arm development,” *Computación y Sistemas*, vol. 24, no. 4, pp. 1387–1401, 2020. doi: https://doi.org/10.13053/cys-24-4-3165.

[5] T. Jin, W. Xie, J. Ma, and K. Ye, “Design Method and Example of a Simple Educational Robot,” in *2018 International Conference on Mathematics, Modeling, Simulation and Statistics Application (MMSSA 2018)*, Atlantis Press, pp. 19–24, 2019. doi: https://doi.org/10.2991/mmssa-18.2019.5.

[6] R. Parameshwaran, S. P. Kumar, and K. Saravanakumar, “An integrated fuzzy MCDM-based approach for robot selection considering objective and subjective criteria,” *Applied Soft Computing*, vol. 26, pp. 31–41, 2015. doi: https://doi.org/10.1016/j.asoc.2014.09.025.

[7] S. Evripidou, L. Doitsidis, G. Tsinarakis, Z. Zinonos, and S. A. Chatzichristofis, “Selecting a robotic platform for education,” in *2022 IEEE International Conference on Consumer Electronics (ICCE)*, IEEE, pp. 1–6, 2022. doi: https://doi.org/10.1109/ICCE53296.2022.9730568.

[8] I. I. Basith and B. Marroquin, “Byoe: Design and development of a simple robotic arm,” in *2020 ASEE Virtual Annual Conference Content Access*, 2020. doi: https://doi.org/10.18260/1-2--34245.

[9] K. Jahnavi and P. Sivraj, “Teaching and learning robotic arm model,” in *2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*, IEEE, pp. 1570–1575, 2017. doi:https://doi.org/10.1109/ICICICT1.2017.8342804.

[10] R. L. Avanzato, “Development of a MATLAB/ROS Interface to a Low-cost Robot Arm,” in *2020 ASEE Virtual Annual Conference Content Access*, 2020. doi: https://doi.org/10.18260/1-2--34447.

[11] D. Rivas *et al.*, “BRACON: Control system for a robotic arm with 6 degrees of freedom for education systems,” in *2015 6th International Conference on Automation, Robotics and Applications (ICARA)*, IEEE, pp. 358–363, 2015. doi: https://doi.org/10.1109/ICARA.2015.7081174.

[12] L. Hao and L. Ru, “An open 5-DOF manipulator platform and its control applications,” in *Proceedings 2013 International Conference on Mechatronic Sciences, Electric Engineering and Computer (MEC)*, IEEE, pp. 824–827, 2013. doi:https://doi.org/10.1109/MEC.2013.6885172.

[13] M. B. da S. Neto, J. M. C. de Mendonça, and A. P. C. de Sena, “Development and control of a prototype manipulator SCARA type as teaching tool,” *IFAC-PapersOnLine*, vol. 48, no. 19, pp. 209–213, 2015. doi:https://doi.org/10.1016/j.ifacol.2015.12.035.

[14] J. R. Miranda-Flores, L. F. Luque-Vega, E. López-Neri, L. E. González-Jiménez, and M. Saad, “Design and implementation of a novel robot manipulator kit for Industry 4.0 through educational mechatronics,” in *Proceedings of the 22nd International Conference on Engineering and Product Design Education (E&PDE 2020)*, Herning, Denmark, Sep. 2020. doi: https://doi.org/10.35199/EPDE.2020.67.

[15] K. Wang, G.-Y. Sang, L.-Z. Huang, S.-H. Li, and J.-W. Guo, “The effectiveness of educational robots in improving learning outcomes: A meta-analysis,” *Sustainability*, vol. 15, no. 5, p. 4637, 2023. doi: https://doi.org/10.3390/su15054637.

[16] C. J. Murzynski, O. Ashour, and A. Sammoud, “Constructing Reconfigurable and Affordable Robotic Arm Platform to Teach Robotics and Automation,” in *2024 ASEE Annual Conference & Exposition*, 2024. doi: https://doi.org/10.18260/1-2--47069.

[17] J. Vega and V. Pérez, “G-ARM: An open-source and low-cost robotic arm integrated with ROS2 for educational purposes,” *Multimed Tools Appl*, vol.84, pp.40683–40705, 2025. doi: https://doi.org/10.1007/s11042-025-20748-8.

[18] C. Zeng, H. Zhou, W. Ye, and X. Gu, “iArm: design an educational robotic arm kit for inspiring students’ computational thinking,” *Sensors*, vol. 22, no. 8, p. 2957, 2022. doi: https://doi.org/10.3390/s22082957.

[19] L. Č. Zajc, A. Rezelj, and D. Skočaj, “Low-cost open-source robotic platform for education,” *IEEE Transactions on Learning Technologies*, vol.16,no.1,pp.18–25,2023. doi:https://doi.org/10.1109/TLT.2022.3224227.

[20] E. Lopez-Neri, L. F. Luque-Vega, L. E. González-Jiménez, and H. A. Guerrero-Osuna, “Design and Implementation of a Robotic Arm for a MoCap System within Extended Educational Mechatronics Framework,” *Machines*,vol.11,no.9,p.893,2023.doi:https://doi.org/10.3390/machines11090893

[21] M. Esmat and A. Atta, “Using Robotic Arm as Sidekick to the Teacher in Classroom,”Fusion: Practice and Applications, vol. 18 , no. 1 , pp: 24-34, 2025.doi:https://doi.org/10.54216/FPA.180102.

[22] E. Markvicka, J. D. Finnegan, K. Moomau, A. S. Sommers, M. S. Peteranetz, and T. A. Daher, “Designing Learning Experiences with a Low-Cost Robotic Arm,” in *2023 ASEE Annual Conference & Exposition*, 2023.doi:https://doi.org/10.18260/1-2--42983.

[23] Y. Jadeja and B. Pandya, “Design and development of 5-DOF robotic arm manipulators,” *International Journal of Scientific and Technology Research*, vol. 8, no. 11, pp. 2158–2167, 2019. [Online]. Available: https://www.ijstr.org/final-print/nov2019/Design-And-Development-Of-5-dof-Robotic-Arm-Manipulators-.pdf.

Appendix (A)

For the 5 Dof robotic arm:

1. **Forward kinematics:**

Assume the base yaw θ_1, three pitch joints θ_2, θ_3 , θ_4 and wrist roll θ_5. The link length l_1 (base offset), the arm lengths: l_2, l_3, l_4, while the tool offset : l_5.Let:

$\alpha = \theta_2 + \theta_3 + \theta_4$ (A.1)

$r = l_2 \cos\theta_2 + l_3 \cos(\theta_2+\theta_3) + l_4 \cos\alpha$ (A.2)

$z_p = l_2 \sin\theta_2 + l_3 \sin(\theta_2+\theta_3) + l_4$ sn α (A.3)

For the end-effector, the positions:

$p_x = r\cos\theta_1 + l_5 \cos\theta_1 \cos\alpha$ (A.4)

$p_y = r\sin\theta_1 + l_5 \sin\theta_1 \cos\alpha$ (A.5)

$p_z = l_1 + z_p - l_5 \sin\alpha$ (A.6)

The Orientation:

$$R_0^5 = \begin{bmatrix} c_a c_b & -s_a c_g + s_a s_g c_g & s_a s_g + s_b c_a c_g \\ s_a c_b & s_a s_b s_g + c_a c_g & s_a s_b c_g - s_g c_a \\ -s_b & s_g c_b & c_b c_g \end{bmatrix} \quad (A.7)$$

Where:

c_a: cos a, s_a: sin a, a = θ_1, b = α , g = θ_5

2. **Dynamics Equations of Motion**

For the rigid link 5 Dof arm:

$$M(\Theta)\ddot{\Theta} + C(\Theta,\dot{\Theta})\dot{\Theta} + G(\Theta) = D(\Theta)\,\tau \quad (A.8)$$

Where:

Θ: $[\theta_1\ \theta_2\ \theta_3\ \theta_4\ \theta_5]^T$, M(Θ): is the inertia matrix, C(Θ, $\dot{\Theta}$): the coriolis and centrifugal term, (Θ): the gravity term, D(Θ): the matrix of torque distribution as follows:

$$D = \begin{bmatrix} 1\,0\,0\,0\,0 \\ 0\,1\,0\,0\,0 \\ 0\,0\,1\,0\,0 \\ 0\,0\,0\,1\,0 \\ 0\,0\,0\,0\,1 \end{bmatrix} \quad (A.9)$$

Appendix (B)

Robotic Arm Specifications	
DOF	5
Material	Lightweight aluminum sheets
Dimensions	40 cm when fully extended
Weight	1.5 kg (without load)
Payload Capacity	300 g
Motors	Servo motors MG995