Robotica: (2021) volume 39, pp. 367–377. © The Author(s), 2020. Published by Cambridge University Press.
doi:10.1017/S0263574720000417

Virtual Simulation and Experimental Verification for 3D-printed Robot Manipulators

Jonqlan Lin* and Kuan-Chung Lai

Department of Mechanical Engineering, Chien Hsin University of Science and Technology, Taoyuan City, Taiwan 320, Republic of China. E-mail: laign65492@gmail.com

(Accepted May 3, 2020. First published online: June 18, 2020)

SUMMARY
The objective of this work is to construct a robot that is based on 3D printing to meet the low-cost and light structures. The Computer-aided-design model is used with LabVIEW to simulate the given trajectory. Users of the simulation of such methodology can preview the simulated motion and perceive and resolve discrepancies between the planned and simulated paths prior to execution of a task. The advantages of this study are the lack of need to mount extra sensors on realistic robot to measure joint space coordinates, simplifying the hardware. These outcomes can also be used in an undergraduate robotics course.

KEYWORDS: Robot manipulators; Kinematics; End-effector; 3D printing; Path planning.

1. Introduction

Recently, robot manipulators have been widely used in industry and elsewhere. Research into and the development of robot manipulators that grasp various objects are intensive. However, numerous commercially available robot manipulators do not support extensive customization of their design features for attachment to altered manipulator platforms or the integration of additional sensors for research purposes.[1] To address these issues, a 3D printing, rapid prototyping technology has been used in the manufacture of inexpensive robot manipulators.[2–9] 3D printing technology reduces the cost of fabrication of such manipulators and speeds up their assembly. To ensure accessibility to scholars, Prof. Carbone utilized an open-source, 3D-printed, three-finger robotic gripper with a simple design and a higher payload than similarly sized tendon-driven mechanisms to develop a robot system.[4] 3D printing has been used to determine the feasibility and functionality of the mechanism solution.[5] The design of a low-cost desktop robot based on the 3D printing technology and open-source control system has been proposed elsewhere.[6] However, all of the above efforts involved the fabrication of only a robot hand using the 3D printing technology without positioning control. Another work demonstrated the design of a 3D print head for use with the ABB IRB 140 robotic arm.[7] Surgical robots have been extensively investigated for use in a wide range of procedures owing to their excellent precision, sensing capabilities, motion scaling, and tremor reduction. Desai et al.[8] discussed the use of 3D printing techniques to meet the strict requirements of surgical interventions. Nevertheless, in these works, only the end-effector of an existing robot, rather than a whole robot manipulator, has been designed.

Kinematics is the learning of motion without consideration of the interacting forces that cause the dynamics. The relevant literature includes actual kinematic analysis that is carried out using Computer-aided-design (CAD) models.[10] Unfortunately, such study analyses include only simulations of robots without real-time verification. Rathy and Balaji[11] developed a six-axis robot arm using the Arduino card, based on the LabVIEW platform. A methodology for developing offline

* Corresponding author. E-mail: jlin@uch.edu.tw

https://doi.org/10.1017/S0263574720000417 Published online by Cambridge University Press

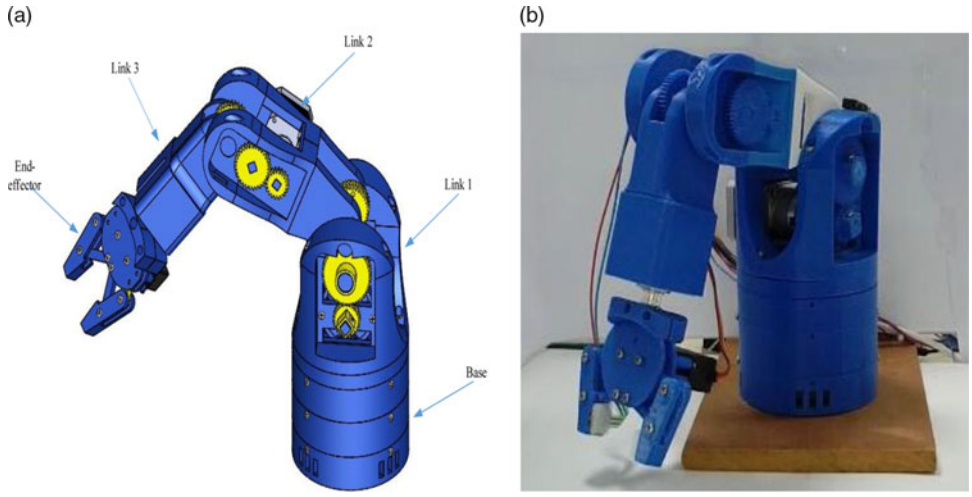


Fig. 1. The proposed 3D design model for robot manipulator system: (a) 3D drawing model and (b) real robot.

and online software for simulating custom robots (Nex Dexter Robotic Manipulator) has been developed.[12] A method for simulating human–robot collaboration in virtual reality has been proposed.[13] However, the major shortcomings of these methods are their concentration on the design of intuitive programming platforms for the robot system without a kinematic analysis to compare simulated and experimental results. Lin et al.[14] used the zero moment point with augmented reality to improve the stability of the gait motion of a biped robot, without considering kinematic calculations for serial link robot arms.

As a result of rapid development and innovation, 3D printing is known as a valuable means in the design procedure with rapid prototyping of robot mechanism, especially in primary phases of robot developments. Hence, the objective of this paper is to design a robot manipulator that is based on 3D printing that meets the demand for low cost and a light structure. The remaining of this paper is as follows: The system configuration is briefly explained in Section 2. In Section 3, the kinematic calculation is specified for the proposed robot system. Forward and inverse kinematics are used to compute the coordinates of the robot manipulator. The implementation and experimental verification is are delineated in Section 4. The SolidWorks CAD model is used with LabVIEW to simulate the assigned trajectory. Real-time control is carried out and its outcomes are compared with a simulation to verify the proposed kinematic calculations. Experimental results confirm that the design herein is reliable and has a low cost of implementation. Finally, the conclusions of the contributions in are demonstrated in Section 5.

2. System Configuration

The goal of this work is to complete the design and implementation of a robot manipulator system. A 3D software package, SolidWorks, was used for the design (Fig. 1(a)). The proposed robot manipulator has three links with an end-effector, each of which is driven by a revolute motor and mounted on a supporting base. The components of the proposed robotic system were fabricated and assembled as depicted in Fig. 1(b). The prototype was manufactured using a 3D printing technology with Acrylonitrile Butadiene Styrene plastic and rubber materials. The design features of the proposed robot manipulator system are described below.

Generally, shoulder joints require motors with high power and torque. In this work, to increase the torque, planet gear-type transmission is used on the base and link 1. The reduction ratio is designed to be 4:1 (Fig. 2(a)). Since space is limited, a two-segment gear mechanism is used between link 1 and link 2 (Fig. 2(b)) and another is used between link 2 and link 3 (Fig. 2(c)). The ratio of the speeds of link 1 and link 2 is designed to be 2.72, and that of link 2 and link 3 is designed to be 2.56. The end-effector has a two-finger-type gripper (Fig. 2(d)). This type of gripper was developed mainly as a result of the comparatively simple strategy of associating such devices with fully actuated agile artificial fingers. This under-actuated finger device exhibits the close packing of altered owing to its adaptive grasping with single actuation. The end-effector can also be embedded in sensing elements, such as tactile sensors, to provide a force feedback capability, as well as a depth camera for object recognition. These devices facilitate research into autonomous grasping.

https://doi.org/10.1017/S0263574720000417 Published online by Cambridge University Press

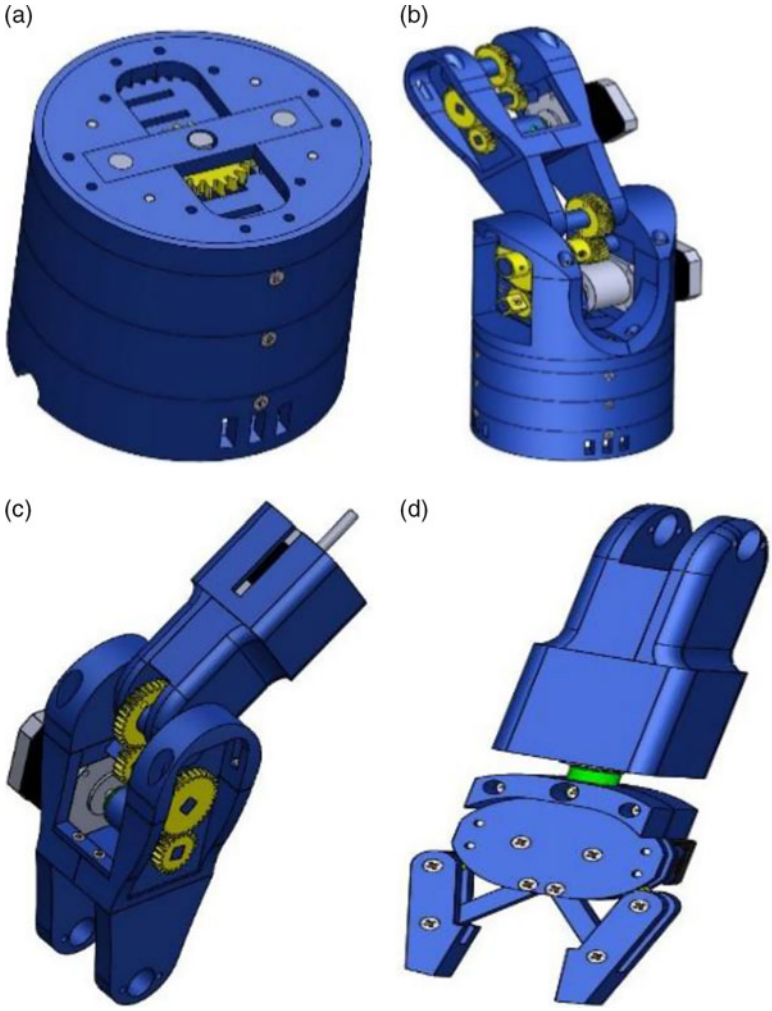


Fig. 2. Component structure between each link.

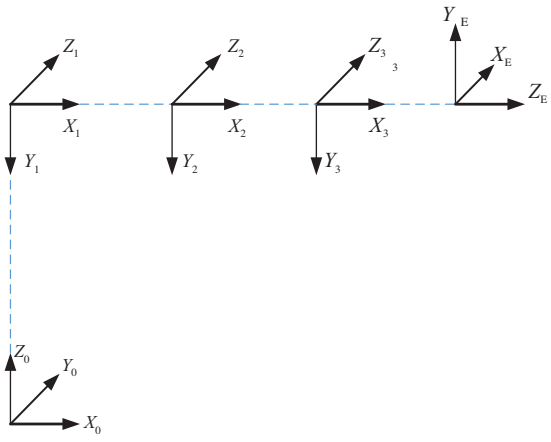


Fig. 3. D–H coordinate frames.

3. Kinematics Calculation

To move the end-effector of the robot manipulator accurately to touch an object, forward and inverse kinematics must be used to compute the orientation of the end-effector. Coordinate frames are attached to the base, links 1∼3, and the end-effector as depicted in Fig. 3. Each link *i* may be fixed

https://doi.org/10.1017/S0263574720000417 Published online by Cambridge University Press

Table I. Danevit–Hartenberg parameters of the proposed robotic system.

Joint (*i*)	***θ***	***α***	***a***	***d***
1	θ_1	-90°	0	d_1
2	θ_2	0	a_2	0
3	θ_3	0	a_3	0
E	θ_4	0	0	0

to each coordinate frame (x_i, y_i, z_i). The frame that is attached to link 0 is called the base frame. The frame that is attached to link E is the end-effector frame.

As in the Danevit–Hartenberg (D–H) scheme, the homogeneous transformation matrix A_i^{i-1} of the robot system is given by Eq. (1). Here, the link parameters are α_i, which is the twist of link i, and a_i, which is its length. The joint parameters are the joint angle θ_i and the joint offset d_i.[15] To obtain the coordinates of an end-effector in the base frame, D–H parameters of the proposed robotic system are found in Table I.

$$A_i^{i-1} = \begin{bmatrix} c\theta_i & -c\alpha_i s\theta_i & s\alpha_i s\theta_i & a_i c\theta_i \\ s\theta_i & c\alpha_i c\theta_i & -s\alpha_i c\theta_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{1}$$

To attain the coordinates of an end-effector in terms of the base frame, we may use the matrices.

$$A_{\mathrm{E}}^0 = A_1^0 \times A_2^1 \times A_3^2 \times A_{\mathrm{E}}^3 = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2}$$

where

$$\begin{aligned}
n_x &= s\theta_1 s\theta_4 + c\theta_1(c\theta_1 c\theta_2 c\theta_3 - c\theta_1 s\theta_2 s\theta_3) & n_y &= c\theta_4(c\theta_2 c\theta_3 s\theta_1 - s\theta_1 s\theta_2 s\theta_3) - c\theta_1 s\theta \\
n_z &= -c\theta_4(c\theta_2 s\theta_3 + c\theta_3 s\theta_2) & o_x &= c\theta_4 s\theta_1 - s\theta_4(c\theta_1 c\theta_2 c\theta_3 - c\theta_1 s\theta_2 s\theta_3) \\
o_y &= -c\theta_1 c\theta_4 - s\theta_4(c\theta_2 c\theta_3 s\theta_1 - s\theta_1 s\theta_2 s\theta_3) & o_z &= s\theta_4(c\theta_2 s\theta_3 + c\theta_3 s\theta_2) \\
a_x &= -c\theta_1 c\theta_2 s\theta_3 - c\theta_1 c\theta_3 s\theta_2 & a_y &= -c\theta_2 s\theta_1 s\theta_3 - c\theta_3 s\theta_1 s\theta_2 \\
a_z &= s\theta_2 s\theta_3 - c\theta_2 c\theta_3 & p_x &= A_2 c\theta_1 c\theta_2 + A_3 c\theta_1 c\theta_2 c\theta_3 - A_3 c\theta_1 s\theta_2 s\theta_3 \\
p_y &= A_2 c\theta_2 s\theta_1 + A_3 c\theta_2 c\theta_3 s\theta_1 - A_3 s\theta_1 s\theta_2 s\theta_3 & p_z &= d_1 - A_2 s\theta_2 - A_3 c\theta_2 s\theta_3 - A_3 c\theta_3 s\theta_2.
\end{aligned}$$

The location of the end-effector is specified in base coordinates by the A matrix. Also, the position of the end-effector is given in joint coordinates by providing the values of the joint variables. Therefore, the solution to the inverse kinematics problem yields the joint variables q_i that are required to achieve that orientation and position. Inverse kinematics can be used to generate the inverse matrix, as follows

$$(A_1^0)^{-1} \times A_{\mathrm{E}}^0 = A_2^1 \times A_3^2 \times A_{\mathrm{E}}^3. \tag{3}$$

By comparing the right and left hand side of Eq. (3), the joint variables $\theta_1 \sim \theta_4$ can be obtained and described as the following:

$$\theta_1 = \tan^{-1}\left(\frac{p_y}{p_x}\right), \theta_2 = \tan^{-1}\left(\frac{k_2 a_2 + k_2 a_3 c_3 - k_1 a_3 s_3}{k_1 a_2 + k_1 a_3 c_3 + k_2 a_3 s_3}\right)$$
$$\theta_3 = \cos^{-1}\left(\frac{k_1^2 + k_2^2 - a_2^2 - a_3^2}{2a_2 a_3}\right), \theta_4 = \theta_{234} - \theta_2 - \theta_3,$$

https://doi.org/10.1017/S0263574720000417 Published online by Cambridge University Press

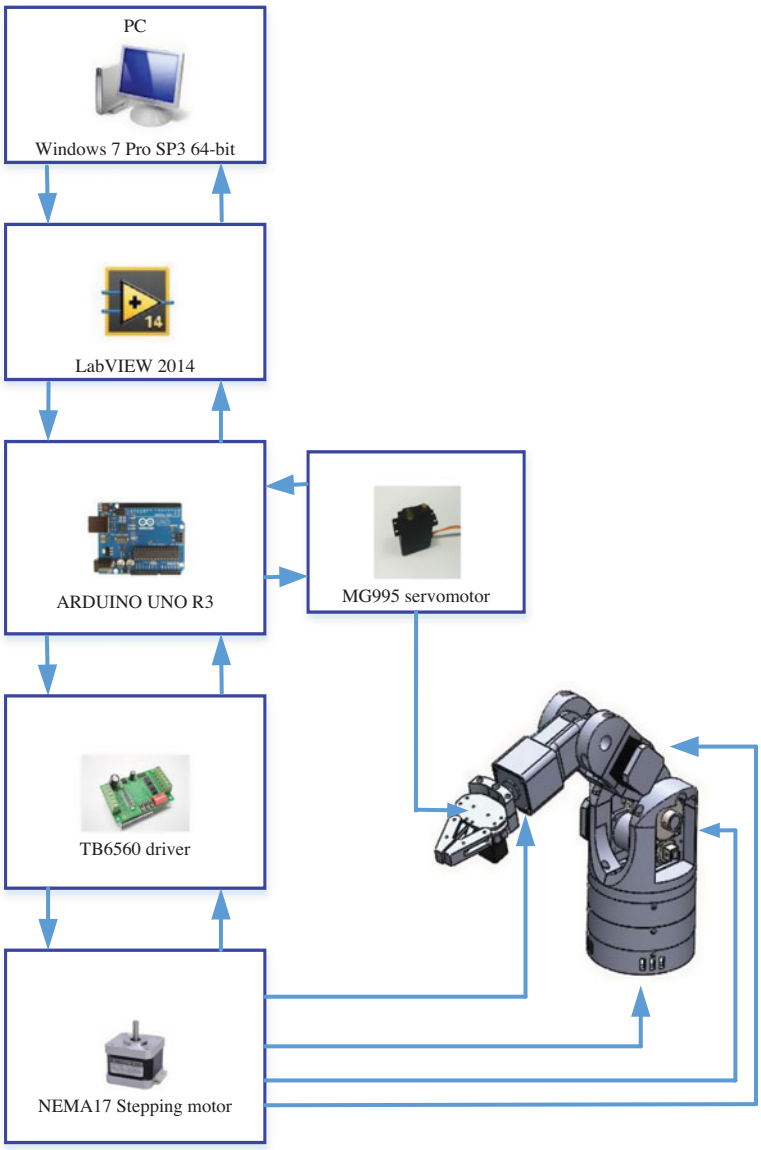


Fig. 4. Real-time implementation for the proposed robot manipulators.

where $\theta_{234} = \tan^{-1}\left(\frac{-n_z}{c_1 n_x + s_1 n_y}\right)$,

$$k_1 = A_2c_2 + A_3c_2c_3 - A_3s_2s_3,\ k_2 = A_2s_2 + A_3c_2s_3 - A_3s_2c_3,$$

where $c(^*) = \cos$, and $s(^*) = \sin$

4. Implementation and Experimentation

To obtain the exact kinematic equation for the control of the end-effector, a simulation was run using the LabVIEW software. The SolidWorks Motion toolbox, which is 3D CAD software, was used to design apparatus and articulated robot manipulators. A Virtual Simulator was used to simulate the motion of robot manipulators along various paths. Thus, a virtual prototyping interface was developed using SolidWorks as the CAD software, LabVIEW for controller programming, and Arduino (Type: Uno R3) as a control card to generate motion for the proposed robotic system. The joints of the manipulators are controlled by stepping motors (Model: NEMA17) which can supply sufficient torque (0.36 N·m) to manipulate the joint. Moreover, the gripper is operated using a servomotor (Model: MG995). Figure 4 displays the real-time implementation of the proposed robot manipulators.

https://doi.org/10.1017/S0263574720000417 Published online by Cambridge University Press

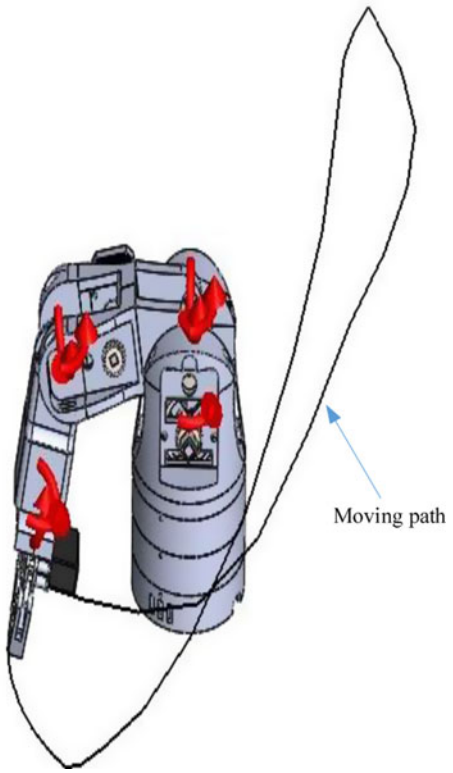


Fig. 5. Robot controlled by National Instruments *SoftMotion Axis*.

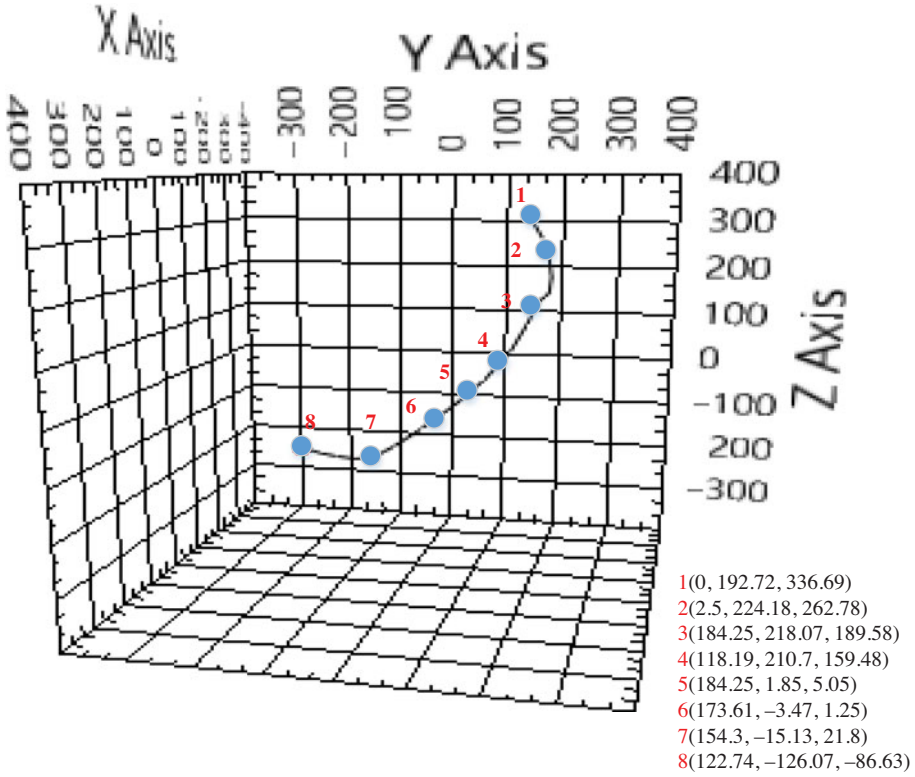


Fig. 6. Robot end-effector trajectory in National Instruments *SoftMotion Axis* for SolidWorks (I).

https://doi.org/10.1017/S0263574720000417 Published online by Cambridge University Press

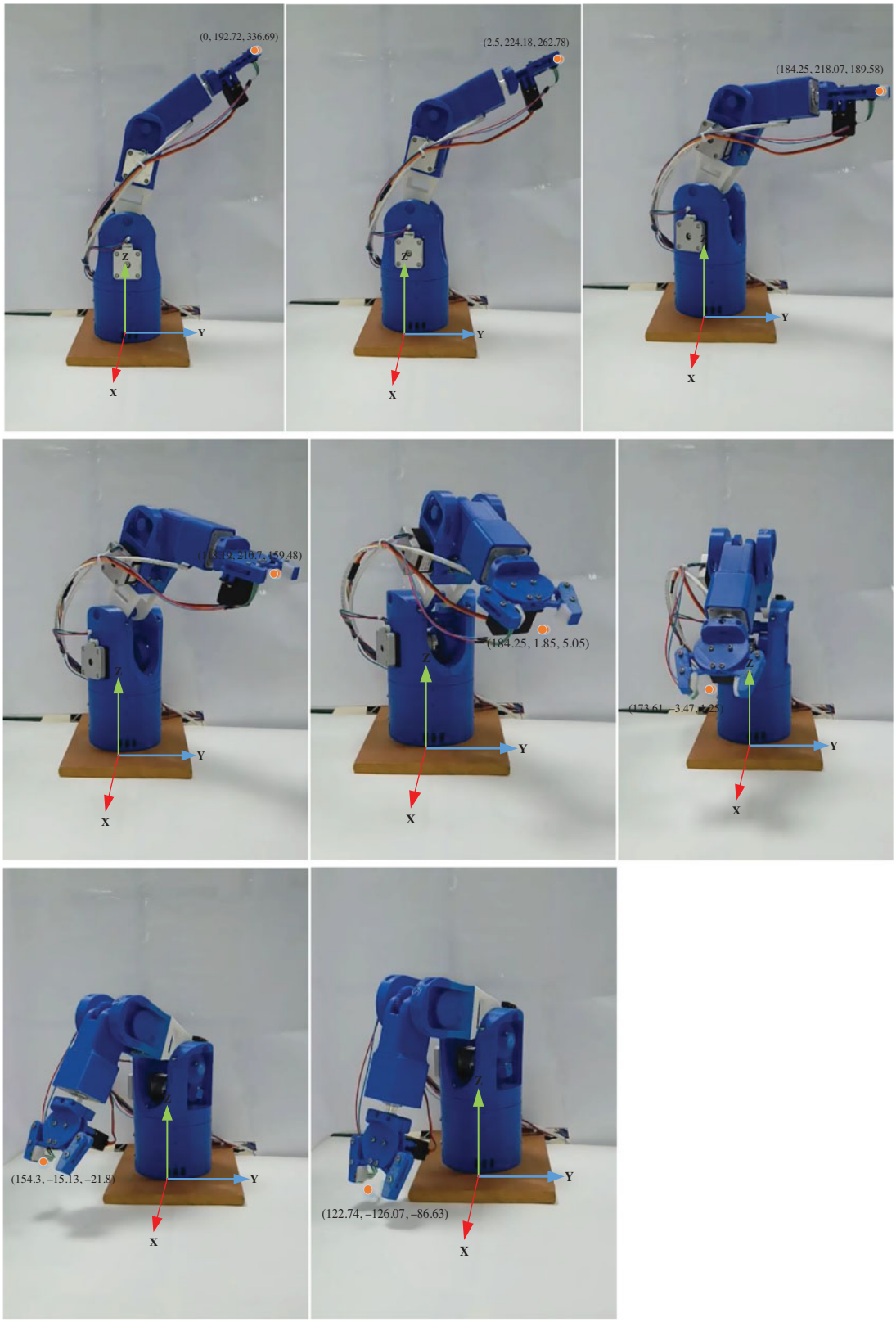


Fig. 7. Real-time robot end-effector trajectory (I).

https://doi.org/10.1017/S0263574720000417 Published online by Cambridge University Press

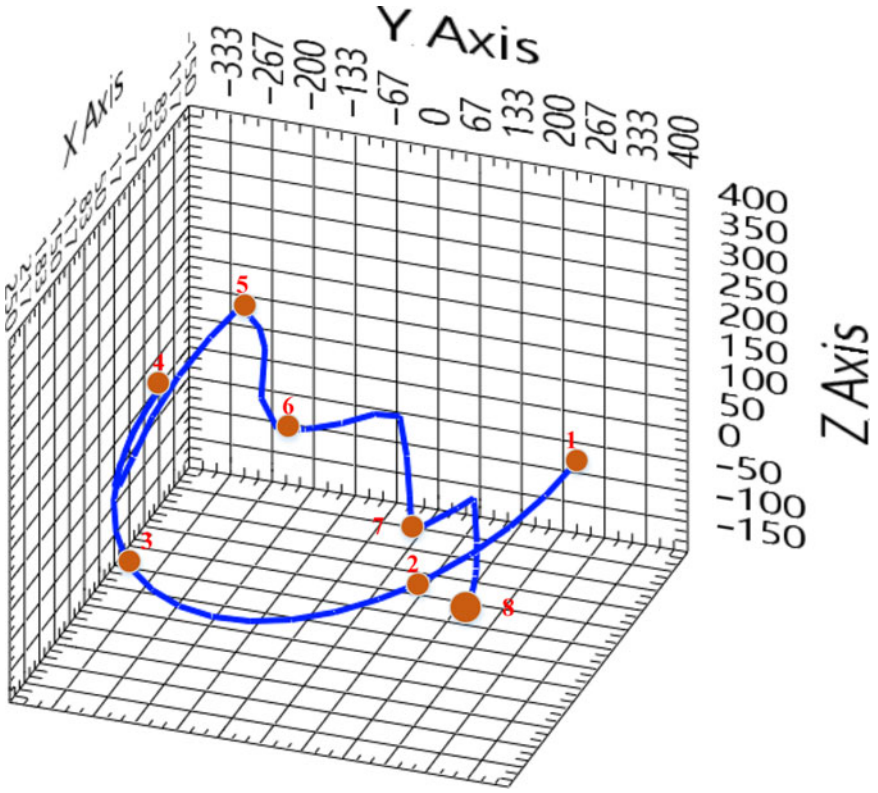


Fig. 8. Robot end-effector trajectory in National Instruments *SoftMotion Axis* for SolidWorks (II).

The connection between LabVIEW and SolidWorks software can be regarded as a master–slave connection. The motors that are specified in the *SolidWorks Assembly* toolbox are actuated using the National Instruments LabVIEW *SoftMotion Axis* toolbox, which is added to the project manager. If more than one motor is to move simultaneously, then a coordinate space that is formed from the axes of the respective motors must be added to the LabVIEW project (Fig. 5). A 3D virtual 1:1 model of a robot manipulator with the same functional features as the real robot manipulator is constructed using the LabVIEW *SoftMotion Axis* toolbox. This approach can be used to examine the motions that are involved in complex mechanisms using the software interface without making a real apparatus.

This section presents several simulation and experimental results that were obtained using the above methodologies. To evaluate the performance of the proposed approach, the robot manipulators were studied using the suggested simulation tuning technology in real-time experiments.

First, the initial joint angles (θ_1, θ_2, θ_3, θ_4) are set to (90°, -80°, 45°, 0°), and the final target joint angles are set to (-45°, -10°, 85°, 0°), respectively. The selection of the position of the end-effector is based on solutions for the forward kinematics. As a result, joint space coordinates can be computed using the *SoftMotion Axis* toolbox and used to verify the motion of the robot. The position of the robot end-effector is initialized as (0, 192.72, 336.69) in the three-dimensional space (X, Y, Z), as shown in Fig. 6. When the end-effector has moved to the final target point, the final target point of the end-effector is (122.74, −126.07, −86.63), as also shown in Fig. 6. Subsequently mapping this path to the joint space yields a time-scale trajectory under robot kinematic and dynamic constraints. As long as the coordinates of the end-effector of the moving robot arm are the target coordinates, the system can effectively realize the purpose of this study, as determined by analyzing the forward kinematics and simulation in the LabVIEW *SoftMotion Axis* software. Figures 6 and 7 display simulated and real-world images that are obtained using the display module, respectively.

In this investigation, the path of the robot end-effector motion was fed thoroughly in a real-life environment. The stepper motors of the real robot arms are manipulated using a pulse-width-modulation (PWM) signal. The joint space coordinates can be used to convert PWM signals into controller codes, which can be applied on a realistic robot to verify its motion (Fig. 7). Joint angle

https://doi.org/10.1017/S0263574720000417 Published online by Cambridge University Press

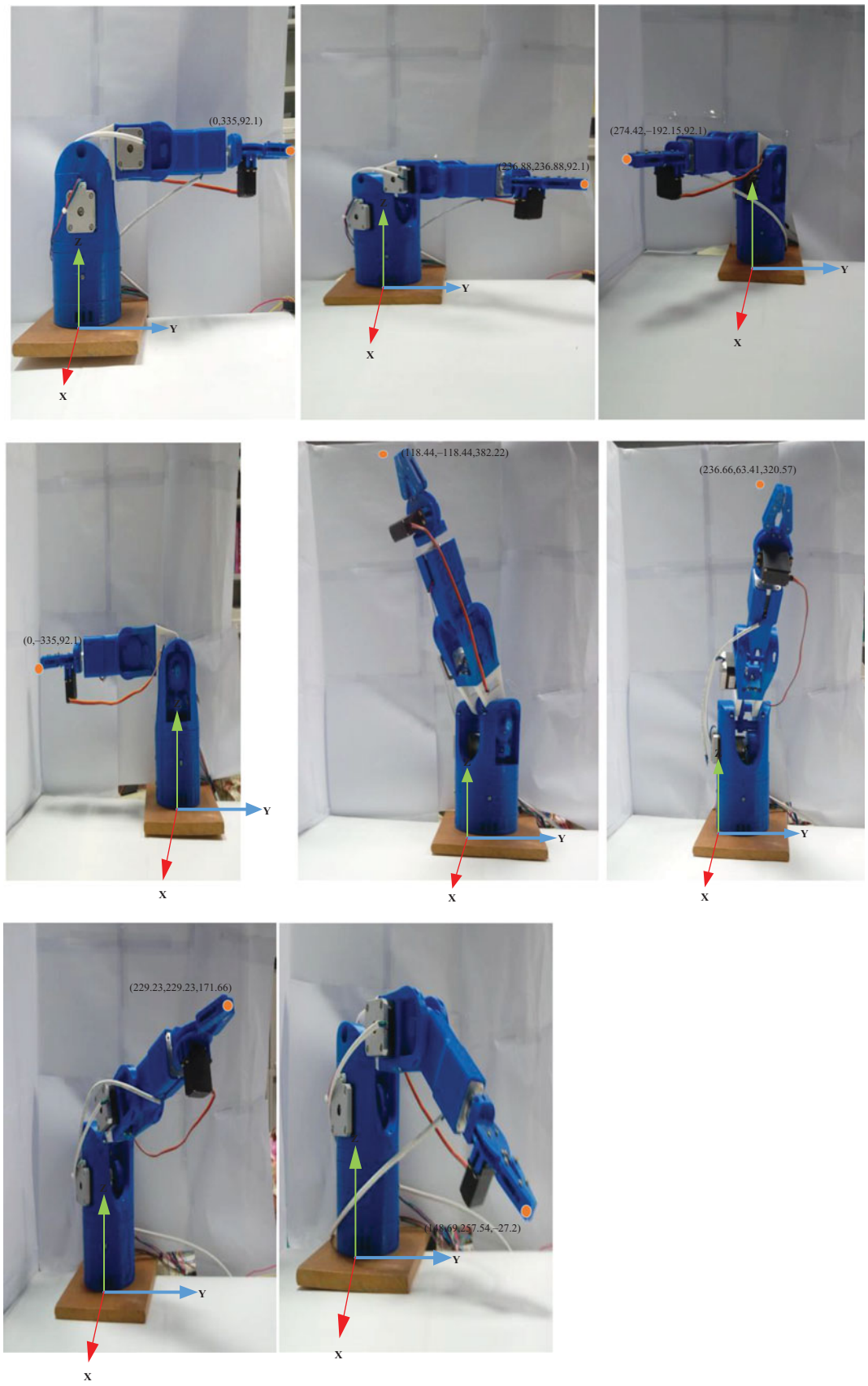


Fig. 9. Real-time robot end-effector trajectory (II).

https://doi.org/10.1017/S0263574720000417 Published online by Cambridge University Press

data for the real robot are used to animate its trajectory. They are transmitted to the virtual robot to determine the offset of the path. An evaluation of the animated trajectories of the simulated robot yields data that are fed back to the real robot to control its joint angles. The proposed methodology provides trajectory tracking in the simulation environment as well as for the real robot. Accordingly, the main advantage of the above method is the lack of any need to mount extra sensors on realistic robot manipulators to measure joint space coordinates, simplifying the hardware.

To verify the performance of the proposed scheme, another example of trajectory tracking is provided. The robot manipulator trajectory is given by the following points in three-dimensional space (X, Y, Z); (0, 335, 92.1), (236.88, 236.88, 92.1), (274.42, −192.15, 92.1), (0, −335, 92.1), (118.44, −118.44, 382.22), (236.66, 63.41, 320.57), (229.23, 229.23, 171.66), and target point (148.69, 257.54, −27.2). The joint angles are set based on solutions for the inverse kinematics. The inverse kinematics equation for the joint angles yields the position of the end-effector. Hence, the joint torques can be manipulated using stepping motors with a PWM signal to generate the desired joint angles. Similarly, the space coordinates of the robot manipulators can be computed using the *SoftMotion Axis* toolbox and used to confirm the motion of the robot. Figures 8 and 9 display the virtual and real-world images, respectively, that are generated by the display module. Users of the various simulation capabilities of the *SoftMotion Axis* toolbox can preview the simulated motion and perceive and resolve discrepancies between the planned and simulated paths prior to execution of a task.

5. Conclusions

The main objective of this paper is to design a robot manipulator that is based on 3D printing to meet the need for a cheap and light structure. A 3D virtual 1:1 model of a robot manipulator with the same functional features as a real robot manipulator is constructed using the LabVIEW *SoftMotion Axis* toolbox. The animated trajectories of the simulated robot are compared, and the relevant data are feedback to the real robot to adjust the joint angle torque input. The system is demonstrated effectively to realize the purpose of this investigation by analyzing the inverse kinematics and running a simulation in the LabVIEW *SoftMotion Axis* software. Accordingly, a simulated system was developed to assist users in programming a robot manipulator and improve its positioning. Experimental results verify that the system is reliable and has low cost of implementation. The major benefit of the system that is presented here is the lack of any need to mount extra sensors on realistic robot manipulators to measure joint space coordinates, simplifying the hardware. Most robot manufacturers provide software for the offline and online programming of robots, but this software is typically sold separately from the robots. Few robot manufacturers provide dedicated software for simulating their robots. This study presents a methodology for integrating CAD modeling, 3D printing, and virtual simulation software into a mechatronic kit. This generic method can be used with other types of robotic manipulator. Therefore, these outcomes can be used to demonstrate the application of the proposed scheme as part of a primary robotics course.

Declaration of Conflicting Interests

The authors declare that there is no conflict of interest.

References

1. R. R. Ma, L. U. Odhner and A. M. Dollar, "A Modular, Open-Source 3D Printed Underactuated Hand," *Proceedings of the 2013 IEEE International Conference on Robotics and Automation (ICRA)*, Karlsruhe, Germany (2013) pp. 2737–2743.
2. Z. Kappassov, Y. Khassanov, A. Saudabayev, A. Shintemirov and H. Varol, "Semi-Anthropomorphic 3D Printed Multigrasp Hand for Industrial and Service Robots," *Proceedings of the 2013 IEEE International Conference on Mechatronics and Automation*, Takamatsu, Japan (2013) pp. 1697–1702.
3. C.-H. Chen, "Mechatronics Design of Multi-Finger Robot Hand," *Proceedings of the 12th International Conference on Control, Automation and Systems*, Jeju Island, South Korea (2012) pp. 1491–1496.
4. G. Carbone, *Grasping in Robotics* (Springer, Dordrecht, Holland, 2013).
5. D. Cafolla, M. Ceccarelli, M. F. Wang and G. Carbone, "3D printing for feasibility check of mechanism design," *Int. J. Mech. Cont.* **17**(1), 3–12 (2016).
6. T. Liu, F. Guo and Y. Yu, "Design of Low-Cost Desktop Robot Based on 3D Printing Technology and Open-Source Control System," *2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference*, Chengdu, China (2019) pp. 739–742.

https://doi.org/10.1017/S0263574720000417 Published online by Cambridge University Press

7. M. Pollák, J. Zajac, J. Török and M. Koèiško, "The Structural Design of 3D Print Head and Execution of Printing via the Robotic Arm ABB IRB 140," *2018 5th International Conference on Industrial Engineering and Applications*, Singapore (2018) pp. 194–198.
8. J. P. Desai, J. Sheng, S. S. Cheng, X. Wang, N. J. Deaton and N. Rahman, "Towards patient-specific 3D-printed robotic systems for surgical interventions," *IEEE Trans. Med. Robot. Bionics.* **1**(2), 77–87 (2019).
9. P. Urhal, A. Weightman, C. Diver and P. Bartolo, "Robot assisted additive manufacturing: A review," *Robot. Comput. Integr. Manuf.* **59**, 335–345 (2019) .
10. K. Khetan, M. Dhaka and V. Sharma, "Control of mechanisms and robots using labview and solidworks and arduino," *Int. J. Eng. Res. Tech. (IJERT).* **5**(2), 286–289 (2016).
11. G. A. Rathy and A. Balaji, "Arduino based 6dof robot using labview," *Int. J. Adv. Res. Ideas Innov. Tech.* **4**(1), 354–358 (2018).
12. A. Patwardhan, A. Prakash and R. G. Chittawadigi, "Kinematic analysis and development of simulation software for nex dexter robotic manipulator," *Procedia Comput. Sci.* **133**, 660–667 (2018).
13. P. Rückert, L. Wohlfromm and K. Tracht, "Implementation of virtual reality systems for simulation of human-robot collaboration," *Procedia Manuf.* **19**, 164–170 (2018).
14. J. Lin, Z. M. Li and J. Chang, "Gait motion stabilization tuning approach of biped robot based on augmented reality," *Robotica* **32**(3), 325–339 (2014).
15. F. L. Lewis, D. M. Dawson and C. T. Abdallah, *Robot Manipulator Control-Theory and Practice* (Marcel Dekker Inc., New York, USA, 2004).

https://doi.org/10.1017/S0263574720000417 Published online by Cambridge University Press