Performance Analysis of 4-DOF RPRR Robot Manipulator Actuation Strategy for Pick and Place Application in Healthcare Environment

Hadha Afrisal [a,c,*], Ahmad Didik Setiyadi [a], Munawar Agus Riyadi [a,c], Rifky Ismail [b,c], Olimjon Toirov [d], Iwan Setiawan [a]

[a] *Department of Electrical Engineering, Faculty of Engineering, Universitas Diponegoro, Tembalang, Semarang, 50275, Indonesia*
[b] *Department of Mechanical Engineering, Faculty of Engineering, Universitas Diponegoro, Tembalang, Semarang, 50275, Indonesia*
[c] *PUI-PT CBIOM3S, Universitas Diponegoro, Tembalang, Semarang, 50275, Indonesia*
[d] *Department of Electrical Machine, Tashkent State Technical University, Tashkent, 100095, Uzbekistan*
Corresponding author: *[*]hadha.afrisal@elektro.undip.ac.id*

Abstract— **Direct and indirect physical contact of humans and objects become the main medium of transmissible diseases such as COVID-19. Some strategies have been proposed to mitigate the risks of infections by minimizing physical contact, such as using robotics technology. Tele-robotics is one of the sub-fields in robotics that aims to implement physical surrogates for monitoring and controlling robots from remote distances, either autonomous, semi-autonomous, or manually guided. This paper discusses experimental research for evaluating the performance of a 4-DOF robot manipulator for pick and place tasks on small medical objects, such as test tubes in table-top scenarios. The robot manipulator is designed as an RPRR manipulator and is equipped with a gripper attached to its end-effector. Inverse kinematics and trajectory planning methods have been successfully implemented in real-time. The inverse kinematic method utilizes a pseudo-inverse Jacobian solver, and the trajectory generation utilizes a sigmoid function. The performance analysis results show that pick and place missions have been demonstrated with minimum tolerable position error, which is not more than 3.5 mm. The robot manipulator can satisfy high precision during repetitive experiments and maintain its accuracy in picking and placing standard test tubes from one rack to another within its working space. The smooth trajectories of the end-effector are achieved by implementing the sigmoid function. Thus, it satisfies the requirement for handling objects with minimum vibrations even during the actuation process with maximum speed.**

Keywords— **Robot manipulator; actuation strategy; pick and place; healthcare assistive robot.**

I. INTRODUCTION

Human-to-human physical contact or physical contact with infectious objects is the primary means of transmission for infectious diseases such as COVID-19. Despite the utilization of complete Personal Protective Equipment (PPE), the infection rate of some communicable diseases remains soaring, particularly within intra-hospital environments and over-crowded healthcare facilities or laboratories [1]. The case of re-infections of specific communicable diseases is quite high for intra-hospital cases [2]. In order to mitigate the risks of contagion, telerobotics technologies can be potentially utilized in medical and healthcare facilities [3], [4].

There are various approaches to deploying unmanned systems, such as telerobotics in the healthcare environment, to minimize human-to-human and human-to-objects physical contact. Robotics, automation, and cyber-physical systems can be adopted to assist medical workers in risky tasks in contagious scenarios [5]. There are some approaches to robot deployment in healthcare, such as (1) disinfection and cleaning, (2) logistics and services, (3) telemedicine and telepresence, and (4) detection and control [6].

Autonomous robots and telerobotics applications can potentially be deployed in handling contagious materials in a healthcare environment to minimize the risks of transmissible disease [7]. Tele-robotics is a sub-field of robotics that aims to develop robots and autonomous systems that can interact with operators from a remote distance, either it is guided or unguided [5]. Tele-robotic systems in the medical field are widely applied in diagnostic procedures and treatments, such as medical interventions carried out remotely. This mechanism guarantees no physical contact between doctors/medical workers and patients [7].

Tele-robotics applications can be classified into four main categories to overcoming COVID-19 and similar cases of highly transmissible disease, such as (1) emergency or critical conditions, (2) primary prevention and healthcare support, (3) household and long-term homecare, and (4) medical education and training [8]. Referring to the categorization, this research focuses on evaluating robot manipulator performance for primary prevention and treatment support, which is in the application of infectious material handling, such as in pick and place tasks of utilized objects and tools, including medical samples placed in test tubes.

However, there are some challenges in deploying robotics technologies in the medical field. Some argue that the deployment of robotics technology in the medical field would not be significant as there is a reluctance to use it due to its high cost of procurement and its complex maintenance procedures [9]. Therefore, this paper demonstrates our custom-designed low-cost 4-DOF robot manipulator, which is relatively easy to maintain as it has a modular form. Other challenges come from the technical aspects of robot technology. To comply with the medical environment, the robot must satisfy some technical aspects, such as accuracy, precision, smoothness, and minimum vibrations and sounds during actuation. Hence the position control and trajectory planning method should be developed and evaluated [9], [10].

In this research, we perform an experimental analysis of the 4-DOF RPRR manipulator in terms of actuation strategy in the context of pick and place tasks in a table-top application. For position control, we utilize forward kinematic and inverse kinematic methods alternately. We also utilize the trajectory planning technique to regulate the end-effector's trajectory to move point-to-point. The forward kinematic model is represented as a Jacobian matrix, and the inverse kinematic calculation is performed using the Pseudo-Inverse Jacobian method. The trajectory planning utilizes the sigmoid function to smoothen the end-effector's trajectory both in Cartesian and joint space.

II. MATERIAL AND METHOD

A. System Design and Configuration of Robot Manipulator

This research uses a custom-designed 4-DOF robot manipulator with RPRR (Revolute, Prismatic, Revolute, Revolute) configuration. Revolute joints are configured for joint 1, joint 3, and joint 4, while the prismatic joint is configured for joint 2 as the translational height controller for the robot manipulator. The mechanical design of the robot manipulator is displayed in Fig. 1 (left), and the implementation of the robot is shown in Fig.1 (right). Table I shows the standard Denavit-Hartenberg (D-H) parameter [11] of our 4-DOF RPRR robot, which species the configuration dimension for joints 1, 2, 3, and 4, with a_2 is 7.02 cm, a_3 is 15.27 cm, and a_4 is 14.97 cm. Joint length is represented in centimeters, and angular position is represented in degrees. The θ_i represents the angular position of i-th joint, $\propto_i$ represents the angle of link twist of i-th joint, a_i represents the length of i-th joint link, and d_i represents i-th link joint's offset. Revolute joints are configured in joint 1, 3, and 4 by changing the value of theta (θ_i*) and prismatic joint is controlled by changing link joint's offset (d_i*) to move joint 2 up-and-down for linear translation. Stepper motors and optical rotary encoders are utilized in each joint, which are coupled directly via a link in each axis for sensing the angular position in each joint. In this robot manipulator, we utilize low-cost stepper motors with a resolution of 200 steps per revolution and optical rotary encoders with a resolution of 2400 PPR (Pulse per Rotation) in a full quadrature mode.

The main processor for this robot manipulator is Raspberry Pi 4 and is supported by 2 peripheral controllers of the ESP32 microcontroller in the function of handling the sensor readings and actuators control. Serial communications are used for communicating the data between the main controller and peripheral controllers. As visualized in Fig. 2., each joint utilizes a stepper motor which is controlled using a stepper motor driver. Each axis of the stepper motor is coupled to the axis of an optical rotary encoder for reading angular position. The end-effector of the robot manipulator is attached to a gripper which is actuated using a motor servo. With this design and configuration, our robot manipulator can pick and place medical-related tasks such as test tubes, syringes, and many other small objects in the healthcare environment.
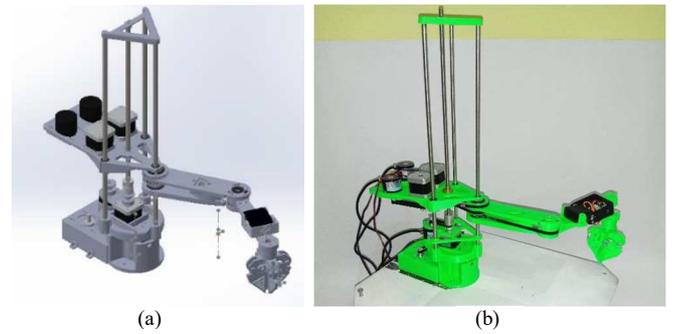
Fig. 1 Mechanical design (a) and implementation (b) of 4-DOF RPRR robot manipulator

TABLE I
STANDARD D-H PARAMETER OF ROBOT MANIPULATOR

Link	a_i	α_i	d_i	θ_i
1	0	0	0	θ_1*
2	a_1	0	d_2*	0
3	a_2	0	0	θ_3*
4	a_3	0	0	θ_4*
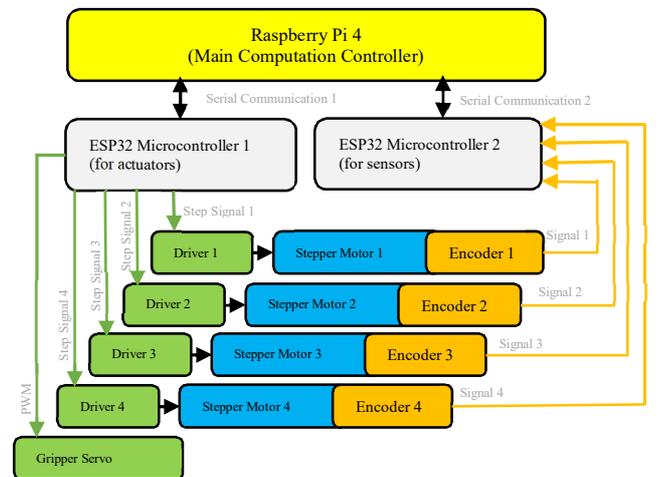
Fig. 2 Robot manipulator's diagram for control system

B. Actuation Strategy: Inverse Kinematics and Trajectory Planning

Actuation strategy applied in this robot manipulator for (1) position control using forward and inverse kinematics, and (2) trajectory planning in the joint space and cartesian space. The end-effector of robot manipulator is attached to a gripper which is able to grip a test tube, thus the minimum vibration, position error, and smooth actuation should be performed. In this section, methods of position control and trajectory planning are explained.

End-effector positioning and movement is manipulated using forward and inverse kinematics method with a height controller. The kinematic model is presented in Fig. 3. In this research, the 4-DOF RPRR robot manipulator is segmented into an RRR representation (joint 1, 3, and 4) with height control in joint 2 (prismatic), as presented in Fig. 4. This model segmentation can benefit in reducing computation time than using a whole direct 4-DOF inverse kinematic computation. Since we utilize a Raspberry Pi 4, this segmented model can significantly reduce the complexity of solving inverse kinematic computation. The positioning in Z-axis is solely manipulated using joint 2 which is in the form of prismatic joint, and the manipulation in X and Y axis is computed using inverse kinematic of RRR model such as presented in Fig. 4. The synchronization of 3-dimensional movement can be implemented using timewise matching of end-effector projected-displacement, which combine the result of RRR inverse kinematic computation with the height control using prismatic joint.
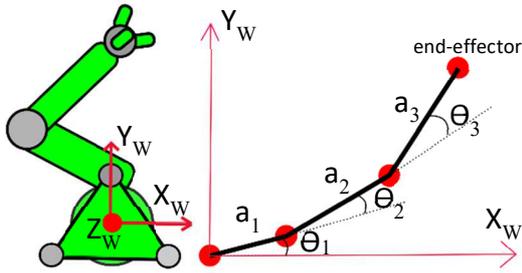
Fig. 3 Robot manipulator simplified kinematic model using RRR in plane XY for Joint 1, 3, and 4, while Joint 2 is used for controlling Z-axis position (height).
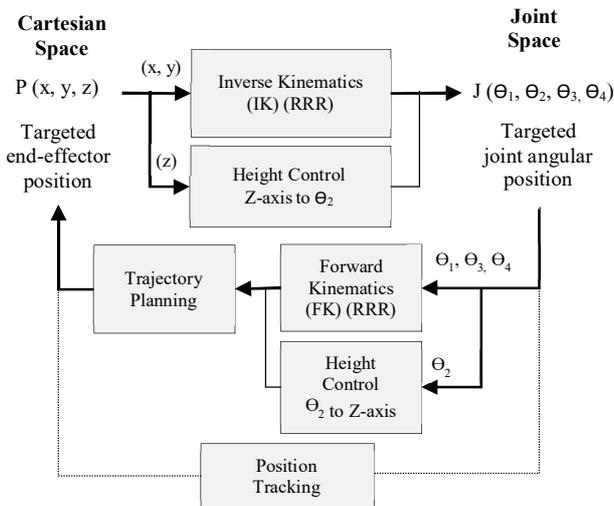
Fig. 4 Actuation strategy and method for robot manipulator control

The forward kinematic model of RRR configuration can be constructed from DH parameter in Table I. The model is then represented as Homogenous Transformation Matrix as presented in Equation (1) to Equation (5) which maps the joint space into 3D cartesian space, with A_1 represents the transformation matrix for joint 1, A_2 represents the transformation matrix for joint 2, and A_3 represents the transformation matrix for joint 3, which will be manipulated on X and Y axis. The cartesian position of end-effector can be represented using Equation 5, however the position of Z-axis is directly represented from height controller of prismatic joint.

$$A_1 = \begin{bmatrix} c_1 & -s_1 & 0 & a_1 c_1 \\ s_1 & c_1 & 0 & a_1 s_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}$$

$$A_2 = \begin{bmatrix} c_2 & -s_2 & 0 & a_2 c_2 \\ s_2 & c_2 & 0 & a_2 s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}$$

$$A_3 = \begin{bmatrix} c_3 & -s_3 & 0 & a_3 c_3 \\ s_3 & c_3 & 0 & a_3 s_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

$$T_3^0 = T_1^0 T_2^1 T_3^2 = A_1 A_2 A_3 \tag{4}$$

$$T_3^0 = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5}$$

Inverse kinematic computation is required to map the known targeted point of end-effector in Cartesian space to derive angular position of motor in joint space. There are several ways of solving inverse kinematics problems, namely using (1) algebraic approach, (2) geometrical approach, (3) iterative/numerical approach, and (4) soft computing-based approach [12]. Algebraic method utilizes algebraic equation to find joint angles from specified end-effector position in Cartesian space. However, this method is relatively slow and inadequate for a higher DOF robot and does not guarantee closed form solution [13], [14]. Geometrical approach uses theory of geometric algebra to represent rotation and orientation; this method is suitable for serial link manipulator. However, it is restricted with minimum number of required DOF for more than 3-DOF [15], [16]. Iterative/numerical methods proposed an iterative solution for solving inverse kinematic. This method mainly utilizes Inverse Jacobian matrix for deriving joint space from end-effector position in Cartesian space. However, it requires initial value selection and usually does not necessarily provide solution if the matrix is singular [17]. Soft computing method such as Artificial Neural Network [18], [19] and Adaptive Neuro-Fuzzy Inference Systems (ANFIS) [20] provides options of solving inverse kinematic using optimization approach. However, this method requires longer process for training the model in the beginning [12].

In the context of our specified configuration and systems design, we utilize an iterative method to derive the joint space from the known Cartesian space position of end-effector.

There are various approaches to implementing numerical methods for computing inverse kinematics solutions, such as of using Cyclic Coordinate Descent (CCD) [21] and Inverse Jacobian [22]. While CCD is more suitable for lower Degree of Freedom serial link manipulators, it does not guarantee solving problem of singularity and redundancy. The Inverse Jacobian method, in another hand, has a linear model of matrix representation with a more suitable formalized form for simultaneous actuation for all joints. There are number of approaches for solving Inverse Jacobian, such as using Jacobian Transpose, Pseudo-inverse, and Damped Least Square [23].

In this research, Jacobian Pseudo-Inverse method is used as a solver to find the solution to our inverse kinematic problem, as it provides a better solution without using direct matrix inverse. Jacobian pseudo-inverse is used to find the angular-position of each joint from the cartesian position of end-effector. The formulation and derivation of solution is provided in Equation (6) to (9).

$$\dot{X} = J(\Theta)\,\dot{\Theta} \tag{6}$$

$$J^T \dot{X} = J^T J(\Theta)\,\dot{\Theta} \tag{7}$$

$$(J^T J)^{-1} J^T \dot{X} = (J^T J)^{-1} J^T J(\Theta)\,\dot{\Theta} \tag{8}$$

$$\dot{\Theta} = J^+ \dot{X} \tag{9}$$

In order to generate a smooth trajectory for the end-effector's movement, we utilize the sigmoid trajectory method in the joint space. Sigmoid trajectory function [24] is applied in the angular velocity control for each joint space hence it guarantees the smooth actuation of end-effector. This trajectory planning is implemented to guarantee the production of smooth path movement of the end-effector from initial to targeted point [25].

III. RESULT AND DISCUSSION

This section discusses experimental works conducted to evaluate the performance of 4-DOF robot manipulator in a table-top scenario. The first experiment aims to observe the robot manipulator's working space in Cartesian space to map all possible end-effector positions in 3-dimensional axis. The second experiment is performed to analyze robot manipulator's capability of trajectory planning, which mainly evaluate the accuracy of actual positioning relative to the ideal generated trajectory. The last experiment is conducted to evaluate the integral performance of robot manipulator during pick and place of test tubes in a table-top implementation.

A. Robot Manipulator's Working Space

The robot manipulator's working space map is required to observe all possible and allowed end-effector positions in a 3-dimensional axis in Cartesian space. This working space is constrained by mechanical design of the rigid body of each joint link. Hence, we generated working space by attempting all possible position of end-effector in Cartesian space. The experiment is done by simulating and validating all possible end-effector positions in Cartesian space utilizing the URDF (Unified Robotic Description Format) model in ROS and

RVIZ. The URDF is generated from the CAD design our robot manipulator. Within ROS and RVIZ environment, all possible points of end-effector are simulated using IK, especially the outermost and the innermost of allowed positions. To validate the correct real 3-dimensional constraints, the outermost points of simulated working space are then checked by conducting FK to actuate the robot manipulator's end-effector. The result of mapping robot manipulator's working space is visualized in Fig. 5. As presented in Fig. 5., robot's working space shapes a tubular form with inner hollow. The maximum possible distal position is 382 mm from the center of the body, and the minimum possible proximal position is 179 mm from body center. The light green color is the working space of allowed possible end-effector's position.
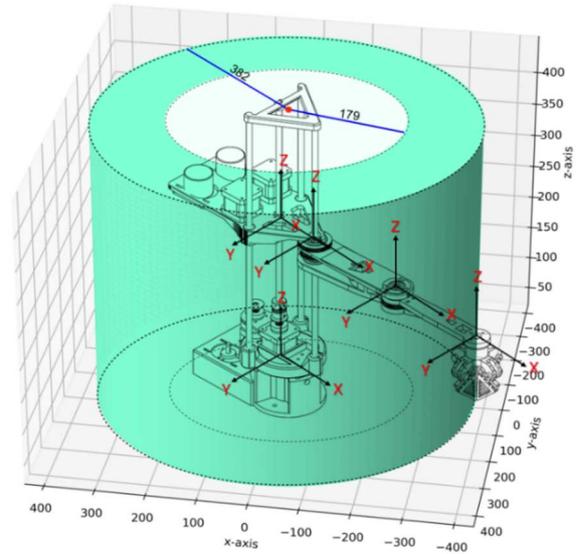
Fig. 5 Three-dimensional working space of robot manipulator [in mm unit]

B. Trajectory Planning Experiment

In this part, trajectory planning experiment was conducted to analyze the performance of robot manipulator in following the desired trajectory generated by our algorithm. We utilize sigmoid trajectory function in joint space to smoothen the movement. The trajectory tracking capability of robot manipulator is then observed in both joint and Cartesian space. In this section, experiments were done in two different types of trajectories of end-effector in Cartesian space, namely (1) 3D space diagonal (Fig. 6) and (2) planar circular trajectory (Fig. 11).
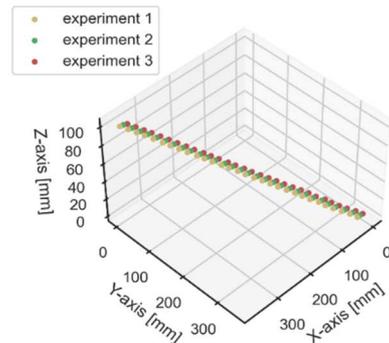
Fig. 6 Experiment on trajectory planning and tracking using space diagonal for low angular speed mode (10 deg/s)
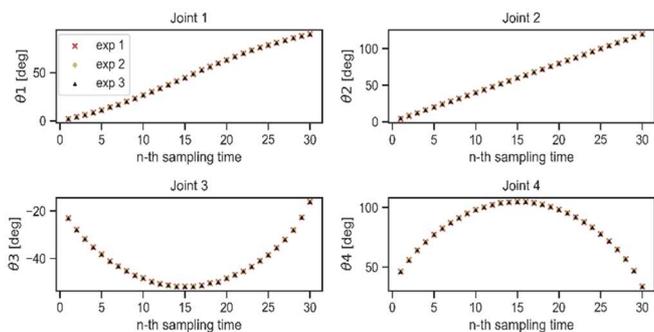
Fig. 7 Angular position tracking of joint 1, 2, 3, and 4 following 3D space diagonal trajectory movement of end effector in Cartesian space using 3 attempts with low angular speed mode (10 deg/s).
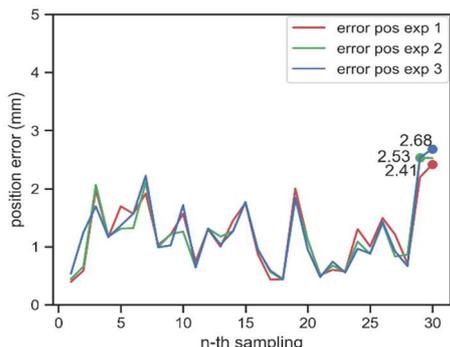
Fig. 8 Position error as 3-dimensional Euclidean distance between ideal position and actual position of end-effector during actuation in the 3D space diagonal trajectory with low angular speed mode (10 deg/s).
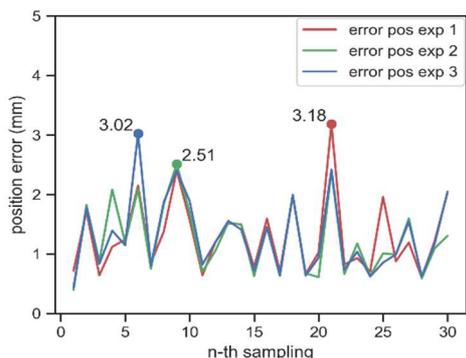
Fig. 9 Position error as 3-dimensional Euclidean distance between ideal position and actual position of end-effector during actuation in the 3D space diagonal trajectory with high angular speed mode (40 deg/s).
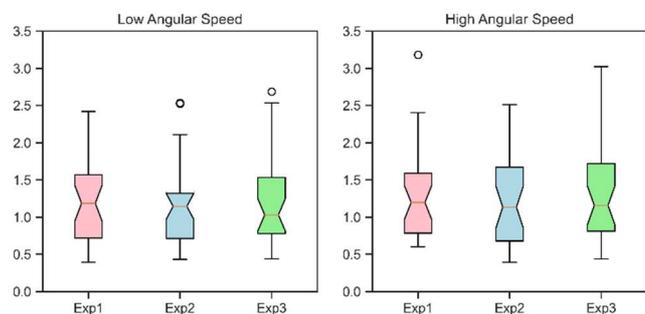
Fig. 10 Position error distribution as Euclidean distance between targeted position and actual position in the tracking of end-effector within space diagonal trajectory experiment, using 2 different angular speed modes.

The 3D space diagonal trajectory experiment observed the capability of trajectory tracking performance under variability of positions in X, Y, and Z axis, by performing actuation of all joints (4-DOF). The 3D space diagonal trajectory experiment has been done to evaluate linearity of robot manipulator during 3-dimensional actuation. This experiment has been conducted 3 times using the same targeted trajectory points which are started from initial point (0, 360, 0) to final point (360, 0, 100). The unit of point is represented in mm, 30 sampling points are taken every 16.7 ms, and the maximum angular speed is set to 10 deg/s. The end-effector actual tracking position in Cartesian space compared to the ideal trajectory is presented in Fig. 6.

We have observed the profile of all actual joints' trajectory in the joint space such as presented in Fig. 7. From this experiment, actual trajectory of robot manipulator, both in cartesian space and joint space, are of high accuracy and precision, this can be seen from the measured position error of 3 different attempts as presented in Fig. 8. The position error is calculated as 3-dimensional Euclidean distance between actual position and ideal position. The maximum error measured in the experiment with low angular speed is 2.68 mm.

Robot manipulator's experiments under different set of angular joint speeds are also performed. We tested our robot manipulator using 2 different maximum joint speeds, which are (1) low angular speed of 10 deg/s and (2) high angular speed of 40 deg/s. The comparison of position error rate during trajectory tracking can be observed in Fig. 8 and Fig. 9. In a higher angular speed, the measured maximum position error slightly increases from 2.68 mm to 3.18 mm. The distribution of error of low and high angular speed for 3 different attempts is compared in Fig. 10. From Fig. 10., the distribution of position error is not significantly distinct between the low and high angular speed, as it shows only small number of outliers in both experiments with maximum position error of not more than 3.18 mm.
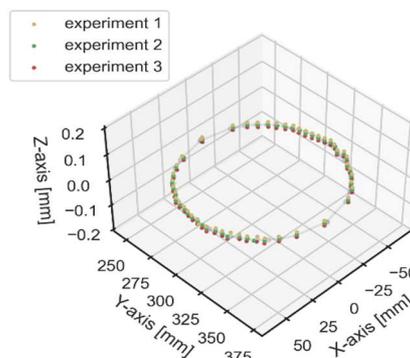
Fig. 11 Experiment on trajectory planning and tracking using circular trajectory in cartesian space with low angular speed mode (10 deg/s)
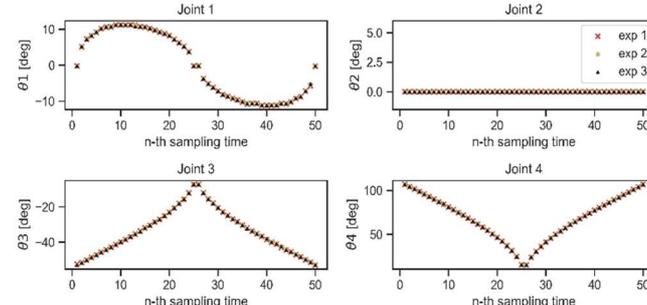
Fig. 12 Angular position for joint 1, 2, 3, and 4 during trajectory tracking with circular movement on XY axis.

2262
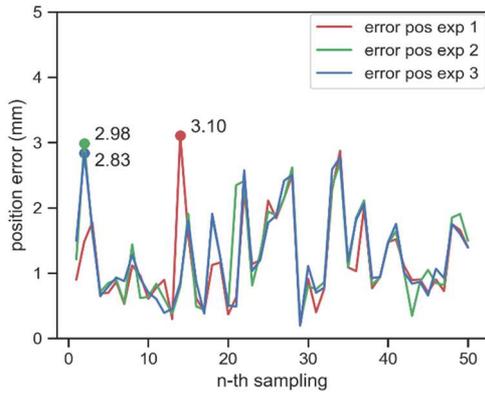
Fig. 13 Euclidean distance measurement between targeted trajectory and actual position of end-effector during actuation in circular movement with low angular speed (10 deg/s)
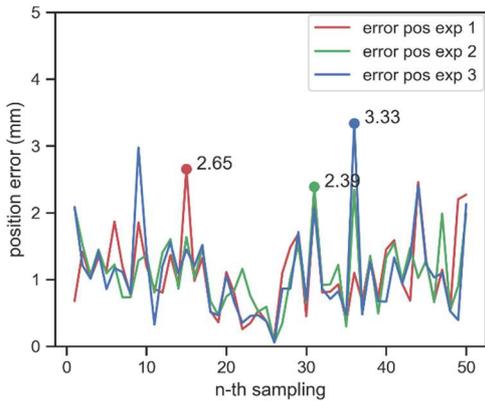
Fig. 14 Euclidean distance measurement between targeted trajectory and actual position of end-effector during actuation in circular movement with high angular speed (40 deg/s)
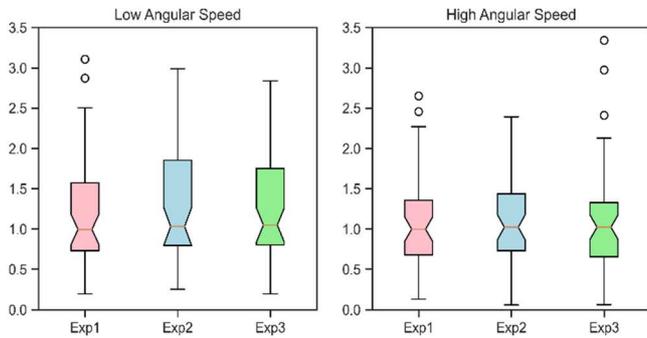
Fig. 15 Position error distribution as Euclidean distance between targeted position and actual position in trajectory tracking with circular trajectory, under 2 different angular speed modes.

The experiment of planar circular trajectory aims to analyze the capability of trajectory tracking performance under variability in X and Y axis, without variation in Z axis (height). This has been done using actuation of joint 1, 3, and 4 (RRR). In this experiment, the base of robot manipulator is situated in the zero axis of world coordinate frame (0, 0, 0). Experiments have been conducted 3 times by using the same targeted trajectory points with center point of circle located in (0, 310, 0) with radius of 60 mm. It utilized 30 sampling points which were taken every 16.7 ms.

We have observed the profile of all actual joints' trajectory in the joint space such as presented in Fig. 12. From this experiment, actual trajectory of robot manipulator, both in cartesian space and joint space, are of high accuracy and precision, this can be seen from the measured position error of 3 different attempts as presented in Fig. 13. The position error is in the form of 3-dimensional Euclidean distance between the tracked actual position and the projected ideal position. The maximum error measured in the experiment with low angular speed is 3.18 mm, such as presented in Fig. 13.

In this planar circular trajectory, robot manipulator's performance under different set of angular joint speed is also observed. We tested our robot manipulator using 2 different maximum joint speeds, which are (1) low angular speed of 10 deg/s and (2) high angular speed of 40 deg/s. The comparison of position error rate during trajectory tracking is presented in Fig. 13. and Fig. 14. In a higher angular speed, the measured maximum position error slightly increases from 3.10 mm to 3.33 mm. The distribution of position error of low and high angular speed for 3 different attempts is compared in Fig. 15. From Fig. 15, the distribution of position error is not significantly distinct between the low and high angular speed movement, as it shows only small number of outliers in both experiments with maximum position error of not more than 3.33 mm.

This trajectory planning and tracking experiment using both 3D space diagonal and planar circular trajectory with 2 different angular speeds shows good level of linearity and manipulability of our 4-DOF robot manipulator for table-top application. The measured position error is relatively low for both different angular speeds, hence it demonstrates that the trajectory planning method can be well-implemented for actuation strategy of low to high angular speed, which is in the speed range of 10 deg/s to 40 deg/s. With this result, pick and place strategy can be implemented and tested.

C. Pick and Place Experiment

The experiment of pick and place is conducted using experimental setup presented in Fig. 16. In this experiment, we utilized 2 test tube racks to demonstrate precision pick and place of test tubes from P1 to P2 which is separated in the distance of X_{pi}, which is 30 cm. Each hole on the rack is 2 cm in diameter, and the space between each hole (Y_{pi}) is 0.6 cm. We utilized standard test tubes with diameter of 1.5 cm with height of 15 cm. The robot manipulator's base is situated in zero axis of world coordinate frame (0, 0, 0), thus the picking zone is in negative X axis, and placing zone is in positive X axis. The height of the racks is 12.5 cm hence the lowest position of end-effector is set to 30 cm from the bottom of robot manipulator's base. In pick and place experiment, we utilized high angular speed mode (40 deg/s).
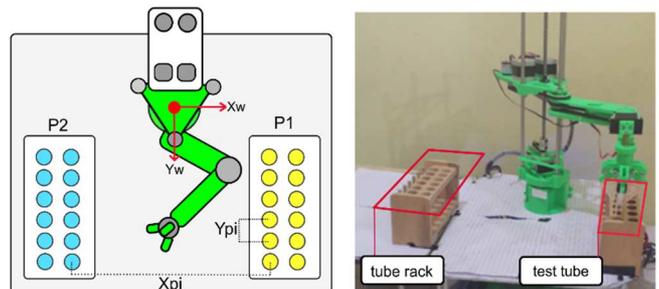
Fig. 16 Experimental setup for pick and place experiments
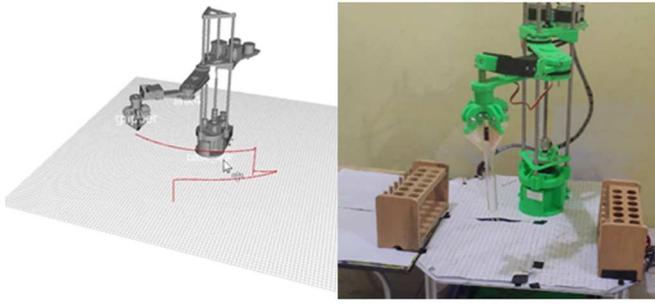
Fig. 17 Experiment for pick and place of test tube using robot manipulator
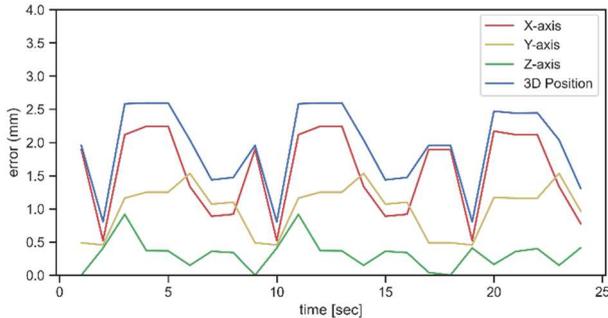
Fig. 18 Error rate on X-axis, Y-axis, Z-axis, and 3D position error (Euclidean distance) of end-effector during pick and place experiment conducted 3-times sequentially.
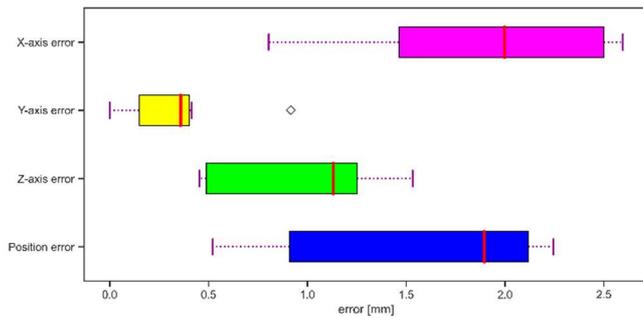
Fig. 19 Error distribution visualized as boxplot over X-axis, Y-axis, Z-axis, and 3-dimensional position error (Euclidean distance) of end-effector during pick and place experiment conducted 3-times sequentially.

The pick and place experiment is conducted 3-times sequentially to pick 3 test tubes from rack 1 (P1) and place them in rack 2 (P2). The demonstration of pick and place experiment is visualized in a real-time way using ROS and RVIZ as displayed in Fig. 17. In order to analyze the performance, we did data logging of the actual end-effector position in Cartesian space and compare it to the ideal trajectory and position. The error rate measured in X-axis, Y-axis, Z-axis, and 3-dimensional Euclidean distance is plotted in Fig. 18.

Error distribution and range during pick and place experiment conducted 3-times sequentially is presented in Fig. 19. From the overall experiments of pick and place, it can be observed that maximum position error in each axis is no more than 2.6 mm: (1) maximum X-axis error of 2.24 mm, (2) maximum Y-axis error of 1.53 mm, (3) Z-axis error of 0.92, and (4) 3-dimensional Euclidean distance as position error has maximum error of 2.59 mm. The measured maximum error can be tolerable during pick and place of test tubes. We have experimented that the maximum position error which is significant to make the pick error is above 5 mm and place error is above 4 mm. From all series of experiments of pick and place, all test tubes are successfully picked from rack 1 (P1) and placed into another rack (P2), hence it can be said that the accuracy of pick and place is 100%, with 0% of failed attempt.

IV. CONCLUSION

The position error distribution from all experiments shows no more than 3.5 mm in position error, thus it can be concluded that robot manipulator is able to maintain high accuracy of its end-effector position in Cartesian space. The implemented trajectory planning method using sigmoid function shows smooth actuation process of robot end-effector from initial point to targeted point with minimum vibrations. Our 4-DOF RPRR robot manipulator demonstrated effective mission of pick and place for test tube with 100% success rate during repetitive experiments, hence it can be noted that it has a relatively high precision rate. In the future, this 4-DOF robot manipulator control strategies can be improved, especially to demonstrate pick and place capability in dynamic environment with existence of obstacles, which is similarly situated in real medical environment or pharmacy laboratory.

REFERENCES

[1] W. H. Gan, J. W. Lim, and D. Koh, "Preventing Intra-hospital Infection and Transmission of Coronavirus Disease 2019 in Healthcare Workers," *Safety and Health at Work*, vol. 11, no. 2, 2020, doi: 10.1016/j.shaw.2020.03.001.

[2] M. Bongiovanni and F. Basile, "Re-infection by COVID-19: a real threat for the future management of pandemia?," *Infectious Diseases*, vol. 52, no. 8. 2020. doi: 10.1080/23744235.2020.1769177.

[3] T. Haidegger *et al.*, "Industrial and Medical Cyber-Physical Systems: Tackling User Requirements and Challenges in Robotics," 2020. doi: 10.1007/978-3-030-14350-3_13.

[4] G. Yang *et al.*, "Keep Healthcare Workers Safe: Application of Teleoperated Robot in Isolation Ward for COVID-19 Prevention and Control," *Chinese Journal of Mechanical Engineering (English Edition)*, vol. 33, no. 1, 2020, doi: 10.1186/s10033-020-00464-0.

[5] S. Mehrdad, F. Liu, M. T. Pham, A. Lelevé, and S. Farokh Atashzar, "Review of advanced medical telerobots," *Applied Sciences (Switzerland)*, vol. 11, no. 1. 2021. doi: 10.3390/app11010209.

[6] C. Tamantini, F. Scotto Di Luzio, F. Cordella, G. Pascarella, F. E. Agro, and L. Zollo, "A Robotic Health-Care Assistant for COVID-19 Emergency: A Proposed Solution for Logistics and Disinfection in a Hospital Environment," *IEEE Robotics and Automation Magazine*, vol. 28, no. 1, 2021, doi: 10.1109/MRA.2020.3044953.

[7] H. Zhang *et al.*, "Research on intelligent robot systems for emergency prevention and control of major pandemics," *Scientia Sinica Informationis*, vol. 50, no. 7. 2020. doi: 10.1360/SSI-2020-0107.

[8] M. Tavakoli, J. Carriere, and A. Torabi, "Robotics, Smart Wearable Technologies, and Autonomous Intelligent Systems for Healthcare During the COVID-19 Pandemic: An Analysis of the State of the Art and Future Vision," *Advanced Intelligent Systems*, vol. 2, no. 7, 2020, doi: 10.1002/aisy.202000071.

[9] S. D. Sierra Marín *et al.*, "Expectations and Perceptions of Healthcare Professionals for Robot Deployment in Hospital Environments During the COVID-19 Pandemic," *Frontiers in Robotics and AI*, vol. 8, 2021, doi: 10.3389/frobt.2021.612746.

[10] P. Khalaf and H. Richter, "Trajectory Optimization of Robots with Regenerative Drive Systems: Numerical and Experimental Results," *IEEE Transactions on Robotics*, vol. 36, no. 2, 2020, doi: 10.1109/TRO.2019.2923920.

[11] R. Campa and J. Bernal, "Analysis of the different conventions of Denavit-Hartenberg parameters," *International Review on Modelling and Simulations*, vol. 12, no. 1, 2019, doi: 10.15866/iremos.v12i1.13623.A. El-Sherbiny, M. A. Elhosseini, and A. Y. Haikal, "A comparative study of soft computing methods to solve inverse kinematics problem," *Ain Shams Engineering Journal*, vol. 9, no. 4, 2018, doi: 10.1016/j.asej.2017.08.001.

[12] M. Husty, I. Birlescu, P. Tucan, C. Vaida, and D. Pisla, "An algebraic parameterization approach for parallel robots analysis," *Mechanism and Machine Theory*, vol. 140, 2019, doi: 10.1016/j.mechmachtheory.2019.05.024.

[13] S. C. Zhen, Z. Zhao, X. Liu, F. Chen, H. Zhao, and Y. H. Chen, "A Novel Practical Robust Control Inheriting PID for SCARA Robot," *IEEE Access*, 2020, doi: 10.1109/ACCESS.2020.3045789.

[14] M. Crenganis, A. Barsan, M. Tera, and A. Chicea, "Dynamic analysis of a five degree of freedom robotic arm using MATLAB-Simulink Simscape," *MATEC Web of Conferences*, vol. 343, 2021, doi: 10.1051/matecconf/202134308004.

[15] Z. Liao, G. Jiang, F. Zhao, X. Mei, and Y. Yue, "A novel solution of inverse kinematic for 6R robot manipulator with offset joint based on screw theory," *International Journal of Advanced Robotic Systems*, vol. 17, no. 3, 2020, doi: 10.1177/1729881420925645.

[16] T. Liu *et al.*, "Iterative Jacobian-Based Inverse Kinematics and Open-Loop Control of an MRI-Guided Magnetically Actuated Steerable Catheter System," *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 4, 2017, doi: 10.1109/TMECH.2017.2704526.

[17] J. Demby'S, Y. Gao, and G. N. Desouza, "A Study on Solving the Inverse Kinematics of Serial Robots using Artificial Neural Network and Fuzzy Neural Network," in *IEEE International Conference on Fuzzy Systems*, 2019, vol. 2019-June. doi: 10.1109/FUZZ-IEEE.2019.8858872.

[18] E. Shahabi and C. H. Kuo, "Solving inverse kinematics of a planar dual-backbone continuum robot using neural network," in *Mechanisms and Machine Science*, 2019, vol. 59. doi: 10.1007/978-3-319-98020-1_42.

[19] D. Deshmukh, D. K. Pratihar, A. K. Deb, H. Ray, and A. Ghosh, "ANFIS-Based Inverse Kinematics and Forward Dynamics of 3 DOF Serial Manipulator," in *Advances in Intelligent Systems and Computing*, 2021, vol. 1375 AIST. doi: 10.1007/978-3-030-73050-5_15.

[20] Y. Chen, X. Luo, B. Han, Y. Jia, G. Liang, and X. Wang, "A general approach based on Newton's method and cyclic coordinate descent method for solving the inverse kinematics," *Applied Sciences (Switzerland)*, vol. 9, no. 24, 2019, doi: 10.3390/app9245461.

[21] M. K. Mishra, S. Ghosal, A. K. Samantaray, and G. Chakraborty, "Jacobian-Based Inverse Kinematics Analysis of a Pneumatic Actuated Continuum Manipulator," 2021. doi: 10.1007/978-981-16-1769-0_1.

[22] M. Dalvi, S. S. Chiddarwar, S. R. Sahoo, and M. R. Rahul, "Dual Quaternion-Based Kinematic Modelling of Serial Manipulators," 2021. doi: 10.1007/978-981-15-3639-7_1.

[23] Y. Fang, J. Hu, W. Liu, Q. Shao, J. Qi, and Y. Peng, "Smooth and time-optimal S-curve trajectory planning for automated robots and machines," *Mechanism and Machine Theory*, vol. 137, 2019, doi: 10.1016/j.mechmachtheory.2019.03.019.

[24] J. Kim, M. Jin, S. H. Park, S. Y. Chung, and M. J. Hwang, "Task space trajectory planning for robot manipulators to follow 3-d curved contours," *Electronics (Switzerland)*, vol. 9, no. 9, 2020, doi: 10.3390/electronics9091424.