Design and Implementation of Embedded Direct Drive SCARA Robot Controller with Resolved Motion Rate Control Method

Shahram Dashti[1]
Faculty of Engineering,
Majlesi Branch, Islamic Azad University, Majlesi, Iran
E-mail: shahram.dashti@gmail.com

Mohsen Ashourian[2, *]
Faculty of Engineering,
Majlesi Branch, Islamic Azad University, Majlesi, Iran
E-mail: ashourian@iaumajlesi.ac.ir
*Corresponding author

Afshin Soheili[3]
Faculty of Engineering,
Majlesi Branch, Islamic Azad University, Majlesi, Iran
E-mail: f.soheili@gmail.com

Received: 24 February 2020, Revised: 8 June 2020, Accepted: 15 June 2020

Abstract: Most of SCARA (Selective Compliance Articulated Robot Arm) direct drive robots today are equipped with a circular feedback system. The Resolved Motion Rate Control (RMRC) method increases the accuracy and compensates the lack of movement transmission system in accurate pick and place actions. In this study, a pick-and-place SCARA robot is developed by using a developed robot manipulator arm and controlling with its designed control systems. To make the end-effector of the SCARA robot arm following desired positions with specified joint velocities, the inverse kinematics technique, known as the RMRC generates motion trajectories automatically. In this research, the kinematics method has been applied with the Jacobian pseudo-inverse or Jacobian singularity-robust inverse to generate and record the pick-and-place motion of the SCARA robot. These records are then compared with the records after using RMRC methods. Several system features like the variation of samples during 50 seconds for the first and second robot joint, and mean deviation for the detailed analysis by the controller after using RMRC motion control algorithm demonstrates the preference of RMRC method in SCARA direct drive robots.

Keywords: Direct Drive, Resolved Motion Rate Control, Robotic, SCARA

Reference: Shahram Dashti, Mohsen Ashourian, and Afshin Soheili, "Design and Implementation of Embedded Direct Drive SCARA Robot Controller with Resolved Motion Rate Control Method", Int J of Advanced Design and Manufacturing Technology, Vol. 13/No. 3, 2020, pp. 83-90.
DOI: 10.30495/admt.2020.1894177.1172.

Biographical notes: Shahram Dashti received his master in Mechatronics from Majlesi Branch, IAU. He is currently senior engineer at Aurora Flight Sciences in the USA. **Mohsen Ashourian** received his PhD in Electrical Engineering from University Technology Malaysia in 2001. He is currently Associate Professor at the Faculty of Engineering, Majlesi Branch, IAU, Isfahan, Iran. His current research interest includes intelligent signal processing and computer vision and automation. **Afshin Soheili** received his PhD in biomedical engineering from UKM, Malaysia. He is currently Assistant Professor at the Faculty of Engineering, Majlesi Branch, IAU, Isfahan, Iran. His current research interest includes biomedical and biotechnological systems and industrial automation.

1 INTRODUCTION

Most of the SCARA robots are used in industry with a fixed speed and structure. While in the new generation of robots along with speed, accuracy and workspace, the operation type of the system is also important. A horizontal revolute robot has four degrees of freedom in which two or three horizontal servo-controlled joints connect base, shoulder, elbow and wrist. SCARA direct drive robots are one of the newest types of these robots. In the traditional motors, the rotating of actuator is geared down to generate the speed or torque required by the manipulator.

Manufacturers are offering direct-drive manipulators, which remove these issues. The direct drive motors drive the arm directly, without the need for reducer gears [1], [5].

The prototype of a direct drive arm was developed by scientists at Carnegie-Mellon University in 1981 [2]. But due to the lack of power transmission, they have some difficulties in their motion patterns. To make an end-effector of the SCARA robot arm following desired positions with specified joint velocities, the inverse kinematics technique, known as the resolved motion rate control (RMRC), can help generating motion trajectories automatically.

Using resolved motion rate control has been started from 1969 as one of the known methods to control manipulators [3]. Whitney first presented this method to make the control problem linear, regardless of the arm configuration, provided that no joints variable encounter limits [3]. His power full algorithm is based on resolving the commanded Cartesian rates to the required joints rates [3-4]. The RMRC use either Jacobian pseudo-inverse or Jacobian singularity-robust inverse as a technique for solving the inverse kinematics of robot arm [4-5].

The innovation in this study is using RMRC method in a direct drive SCARA robots and evaluation of its performance. For this purpose, we make control motion of the predesigned SCARA robot arm from the origin to the destination, and we recorded data before and after using RMRC method. By sampling the data, we demonstrate that using RMRC method increases the accuracy of the arm and increases the reaction time to correct the desired angle of each arm.

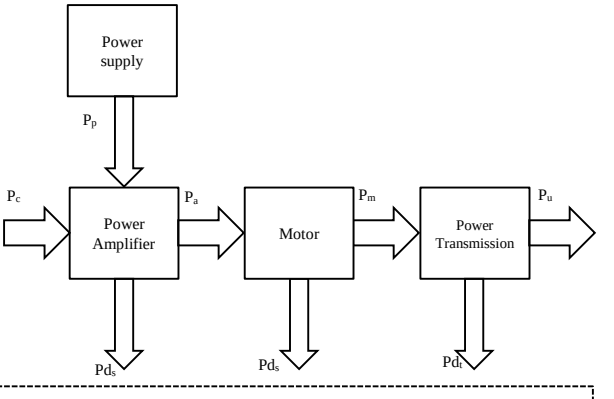
In the following sections, first in Section 2, we explain the architecture of direct drive SCARA robots along with brief overview of the various actuators and their function in the robot transmission system. In Section 3, we explain Forward and inverse kinematic and Jacobian equations of a direct drive SCARA robot, and later in Section 4, we will explain our developed scara robot and its test rig with their associated kinematic analysis. Finally, Section 5 concludes the paper.

2 THE ARCHITECTURE OF DIRECT DRIVE SCARA ROBOT

A typical robot structure with a direct drive robot follows the same pattern. The only difference is the transmission system and actuators. Actuators are robot muscles that are classified into the following parts:

1. Power Supply
2. Power Amplifier
3. Servo Motor
4. Power transmission system

Figure 1 shows the connections between actuators of a typical robot. To choose a good direct drive robot actuator, it is better that the amount of power required by power unit which define joint motion, be determined [6].



Motor Input power: Pa Input control Power: Pc Primary source of Power: Pp

Pu: Mechanical Power: Motor output power: Pm

Loss power of Amplifier and motor transmission and transmission system: Pd_t, Pd_s

Fig. 1 A typical robot driving system.

Robots that use electric direct drive actuators require less space. The traditional robots with geared drive have problems like backlash, friction, compliance, and wear. These issues cause reduced accuracy, poor real time response, periodic maintenance, reduced torque control, and inadequate speed on longer moves. The basic structure of a direct-drive motor is shown in "Fig. 2" [5]. On the other hand, eliminating power transmission have these advantages:

- Removing nonlinear effects due to centrifugal, Coriolis, and inertial forces.
- The use of more accurate closed loop system for intelligent control.
- The ability to move objects from one point to another point. This ability exists even if the control is a long path to follow without transmission system errors propagation.

- They can be used for robots which should remove the barriers in the planning process.

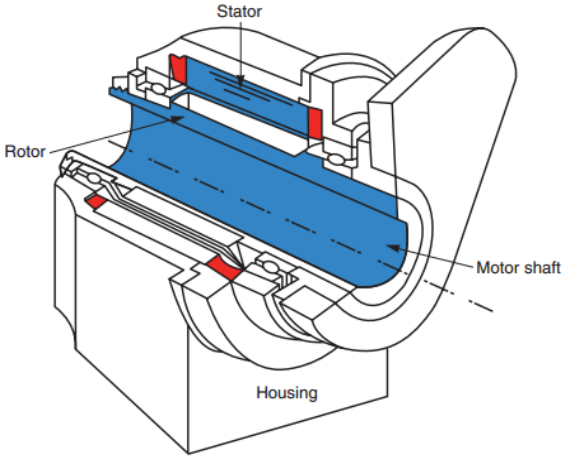


Fig. 2 Driving a robot with direct drive motor derived from Theory and Industrial Applications book.

3 FORWARD AND INVERS KINEMATICS AND JACOBIAN OF SCARA ROBOT

Figures 3 and 4 illustrate algebraic methods for the graph analysis which the coordinates of the end effector of the robot are obtained. We then use these formulas in our controller.

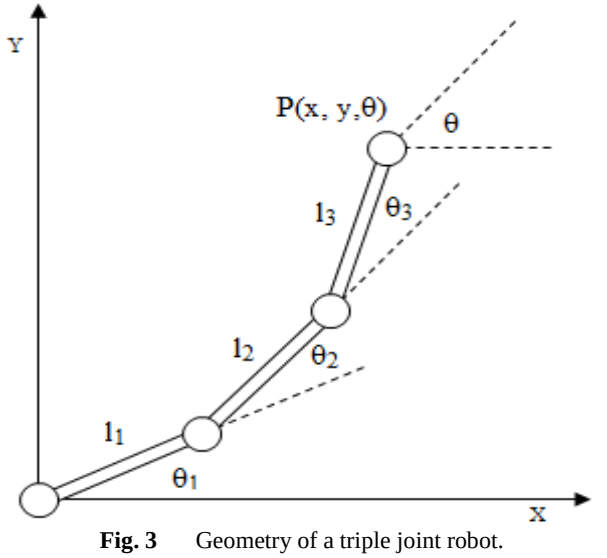


Fig. 3 Geometry of a triple joint robot.

Forward kinematic equations

The variables x and y with respect to the coordinate system of these equations is obtained as below.

$$x = l_1\cos(\theta_1) + l_2\cos(\theta_1+\theta_2) + l_3\cos(\theta_1+\theta_2+\theta_3) \quad (1)$$

$$y = l_1\sin(\theta_1) + l_2\sin(\theta_1+\theta_2) + l_3\sin(\theta_1+\theta_2+\theta_3) \quad (2)$$

$$\theta = \theta_1+\theta_2+\theta_3 \quad (3)$$

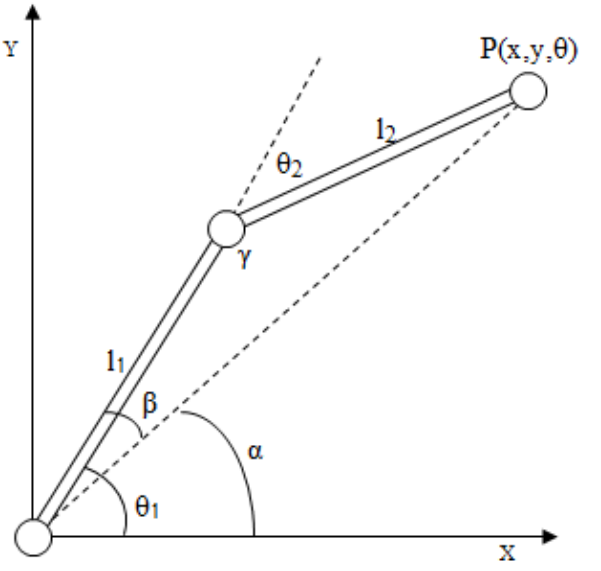


Fig. 4 Geometry of a double joint robot.

Inverse Kinematic Equations

Desired angles with respect to the coordinate system and the formulas are obtained this way.

$$x^2+y^2 = l_1^2+l_2^2-2l_1l_2\cos(180-\theta_2) \quad (4)$$

$$\theta_2 = \cos^{-1}\left[\frac{x^2+y^2-l_1^2-l_2^2}{2l_1l_2}\right] \quad (5)$$

$$\frac{\sin\beta}{l_2} = \frac{\sin\gamma}{\sqrt{x^2+y^2}} \quad (6)$$

$$\alpha = \tan^{-1}\left[\frac{y}{x}\right] \quad (7)$$

$$\sin\gamma = \sin(180-\theta_\gamma) = \sin(\theta_\gamma) \quad (8)$$

$$\beta = \sin^{-1}\left[\frac{l_2\sin\theta_2}{\sqrt{x^2+y^2}}\right] \quad (9)$$

$$\theta_1 = \sin^{-1}\left[\frac{l_2\sin\theta_2}{\sqrt{x^2+y^2}}\right] + \tan^{-1}\left[\frac{y}{x}\right] \quad (10)$$

Jacobian equations

In "Eq. (11)", the time derivative of the kinematics equations yields the Jacobian of the robot, which relates the joint rates to the linear and angular velocity of the end-effector. The Jacobian model delivers a connection across joint torques and the force and torque applied to the end-effector which is shown in "Eq. (12)" [7].

$$\dot{X} = J_{(q)}.\dot{\theta} \quad (11)$$

$$J(q) = \begin{bmatrix} -l_1 S_1 - l_2 S_{1+2} & -l_2 S_{1+2} & 0 \\ l_1 C_1 + l_2 C_{1+2} & l_2 C_{1+2} & 0 \\ 0 & 0 & \frac{e^{(\theta_4/\pi)}}{\pi} \end{bmatrix} \quad (12)$$

4 IMPLEMENTATION OF DIRECT DRIVE CONTROL WITH RESOLVED MOTION RATE CONTROL

As illustrated in "Fig. 5", the RMRC technique can calculate joint rotation speed (θ_{actual}) for every time step, given the end effector desired position and velocity.

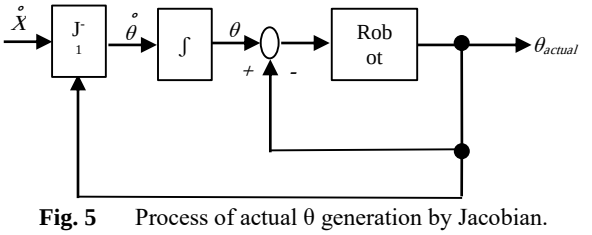


Fig. 5 Process of actual θ generation by Jacobian.

Joint angular velocity can be written as a linear combination of desired end-effector velocity and end-effector position error multiplying with the Jacobian inverse, as shown in "Eq. (13)".

$$\dot{\theta} = J^{-1}\left(\dot{P}_x + K(P_x - P)\right) \quad (13)$$

K is a gain coefficient for adjusting a convergence speed of the robot arm position. Using the forward kinematics, the end effectors position Px at different time steps can be calculated from joints rotational angle by integrating $\dot{\theta}$ using the Euler integration and the formula [8].
The Jacobian pseudo-inverse ($J^{\#}$) can be implemented easily from Equation (14).

$$J^{\#} = J^T(JJ^T)^{-1} \quad (14)$$

The motion routes of the robot effector can be produced after solving the inverse kinematics problem. In next sections we show how to solve these formulas in our controller.

5 ARM-IAUM SCARA ROBOT DIRECT

The main problem of SCARA robot users is the large robot size which limits its application in many pick and place applications. We tried to reduce this problem in our developed robot which is named ARM-IAUM.
As "Fig. 6" shows, ARM-IAUM robot with a robust base using a rigid surface has eliminated the transportation problem of heavy parts. Also, workspace of robot can be increased or decreased in a few minutes by changing the linkage located in the middle of the robot. Due to the external cover of the insulation and the use of anti-acid rubbers, robot can work in wet and dirty environments and it has been designed through the IP51 and IP52 standards [11].

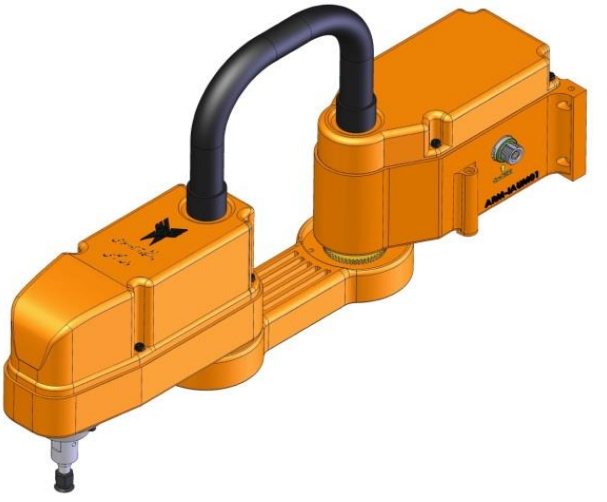


Fig. 6 The Developed ARM-IAUM SCARA Robot.

Kinematic analyse of ARM-IAUM
Based on "Fig. 7", with Combining the geometric and the Denavid Hartenberg method, the kinematic and dynamic equations of the robot can be calculated [9-10].

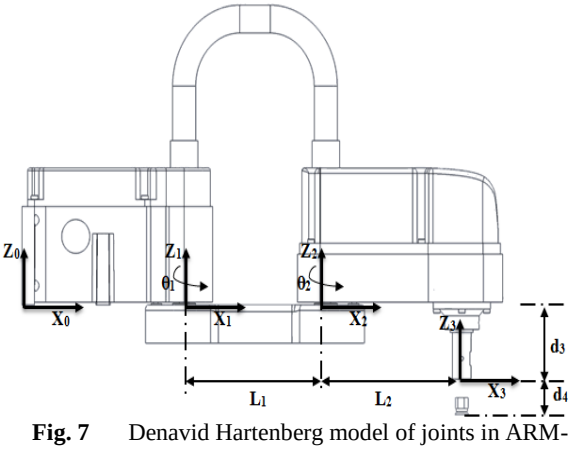


Fig. 7 Denavid Hartenberg model of joints in ARM-IAUM.

$$T = \begin{bmatrix} S_4S_{1+2} + C_4C_{1+2} & S_4C_{1+2} + C_4S_{1+2} & 0 & l_2C_{1+2} + l_1C_1 \\ S_4C_{1+2} + C_4S_{1+2} & -S_4S_{1+2} + C_4C_{1+2} & 0 & l_2S_{1+2} + l_1S_1 \\ 0 & 0 & -1 & -d_4 - d_3 + d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (15)$$

$$P_{x=} l_2C_{1+2} + l_1C_1 \quad (16)$$

$$P_{y=} l_2S_{1+2} + l_1S_1 \quad (17)$$

$$P_{z=} - d_4 - d_3 + d_1 \quad (18)$$

$$\theta_2 = \tan^{-1}\frac{S_2}{C_2} \quad (19)$$

$$\theta_1 = \tan^{-1}\frac{S_1}{C_1} = \tan^{-1}\frac{(L_1+L_2C_2)p_y+(L_2S_2)p_x}{(L_1+L_2C_2)p_x+(L_2S_2)p_y} \quad (20)$$

1.1.a Inverse Velocity solution

From equation (16) and (17) we obtain:

$$\dot{P_x} = -L_1 s_1 \dot{\theta_1} + L_2 s_{12}\left(\dot{\theta_1} + \dot{\theta_2}\right) \quad (21)$$

$$\dot{P_y} = -L_1 \dot{\theta_1} + L_2 c_{12}\left(\dot{\theta_1} + \dot{\theta_2}\right) \quad (22)$$

$$\dot{P_x} = (-L_1 s_1 + L_2 s_{12})\dot{\theta_1} - L_2 s_{12}\dot{\theta_2} \quad (23)$$

$$\dot{P_y} = (L_1 c_1 + L_2 c_{12})\dot{\theta_1} - L_2 c_{12}\dot{\theta_2} \quad (24)$$

$$\dot{\theta_1} = \frac{\dot{p_x} c_{12} + \dot{p_y} s_{12}}{L_1 s_2} \quad (25)$$

$$\dot{\theta_2} = \frac{\dot{p_y}(L_1 s_1 + L_2 s_{12}) + \dot{p_x}(L_1 c_1 + L_2 c_{12})}{L_1 L_2 s_2} \quad (26)$$

$$\dot{d_3} = -\dot{p_z} \quad (27)$$

$$\dot{\theta_4} = \frac{d\theta_4}{d_t} = \frac{c_{12}\dot{n_y} + s_{12}\dot{n_x} - (n_x c_{12} + n_y s_{12})\dot{\theta_{12}}}{c_4} \quad (28)$$

Dynamics analyse of ARM-IAUM

The momentum of the ARM-IAUM robot joints is also calculated as follows:

$$T_1 = b_{11}\ddot{\theta_1} - b_{12}\ddot{\theta_2} - b_{13}\theta\ddot{d}_3 - b_{14}\dot{\theta_1}\dot{\theta_2} - b_{15}\dot{\theta_1^2} \quad (29)$$

$$T_2 = -b_{21}\ddot{\theta_1} - b_{22}\ddot{\theta_2} - b_{23}\ddot{d_3} - b_{24}\dot{\theta_1^2} \quad (30)$$

$$T_3 = -b_{31}\ddot{\theta_1} - b_{32}\ddot{\theta_2} - b_{33}\ddot{d_3} - b_{34} \quad (31)$$

So far, the model and the corresponding motion equations for mechanical arms of Direct Drive SCARA robots are obtained. In practice, we need to add a model of the friction in our software functions [10]. The assumed friction model is a negligible function of joint locations. Therefore, we ignore them, and we study mathematical model and simulations which are critical in the theoretical calculations.

6 PROGRAMMING OF THE ROBOT

Creating mathematical models of the robot are unique and complex because we need to run the underlying mathematical equations through a software program. Implementing software of robot controller is more difficult and complex than hardware implementation. To implement the software, the simulation functions should be analysed in an application. With using excel, implementations of these functions are performed. We implemented three examples of functions to calculate the torque and power, direct and inverse kinematics.

Dynamic Calculation of Robot (power and torque)

As shown in "Table 1 and 2", by using any of the following variables, torque of the first and the second joint was calculated. The mass is not being yet calculated because we only want to check our formulas.

Table 1 Force and torque calculation

	Mass (Kg)	Length (Cm)	Centre of gravity
Base	0.25	14.3	7.15
Joint 1	0.25	12.5	6.25
Joint 2	0.25	12	6
Joint 3	0.25		
Load	0.25		

Table 2 Other System Performance Parameters

Joint 1 Acceleration	0.5
Joint 2 Acceleration	0.5
Joint 1 Torque	18.254
Joint 2 Torque	5.176

Kinematic Calculation of Robot (Direct)

In the "Table 3", by entering any of the following variables, positions of the first and second joints are calculated.

Table 3 Forward Kinematics calculations

Joint	Angle (degree)	X coordinate Position (cm)	Y coordinate Position (cm)
Joint 0		0	0
Joint 1	30	10.83	6.25
Joint 2	25	17.71	16.08

Kinematic Calculation of Robot (Inverse)

Inversely, in the "Table 4", by entering the desired position, angle variables of end-effector are calculated. In "Fig. 8" the position of robot joints is illustrated.

Table 4 Inverse Kinematics Calculations

Desired X (cm)	17.71	Angle (degree):
Desired Y (cm)	16.08	30.0125498
c2	0.906535	
s2	0.42213066	
theta	0.523817811	
psi	0.435794371	
Joints	X coordinate Position (cm)	Y coordinate Position (cm)
Base	0	0
Joint 1	10.82	6.25
Joint 2	17.71	16.08

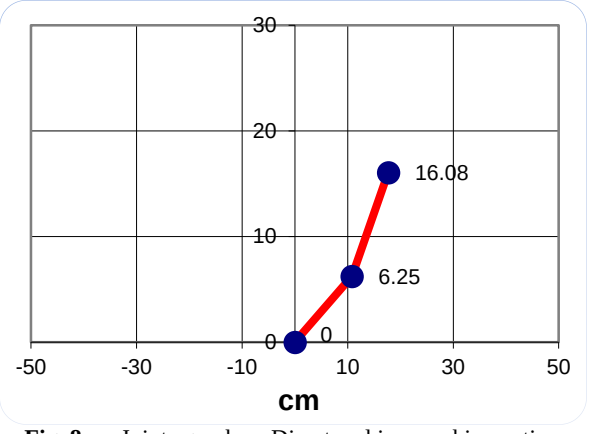


Fig. 8 Joints graphs – Direct and inverse kinematics.

7 RESULT AND ANALYSIS

We examined our result with other reported practical results in [12-13]. In [12-13], the sampled data of writing the text "ohio" is provided and depicted in "Fig. 9". Figures 10 and 11 show the angular location of first and second joints during 50 second. The creation of mathematical models of the robot are unique and complex because we needed to perform the underlying mathematical equations through a software program. Implementing the software of the robot controller is more complex than hardware integration. To create the desired procedures, the main body of the software performs a significant role and needs more extra considerations. For this purpose, simulation functions should be analysed in an application. Excel software is used to implement three examples of functions calculating the torque, power, direct and inverse kinematics.

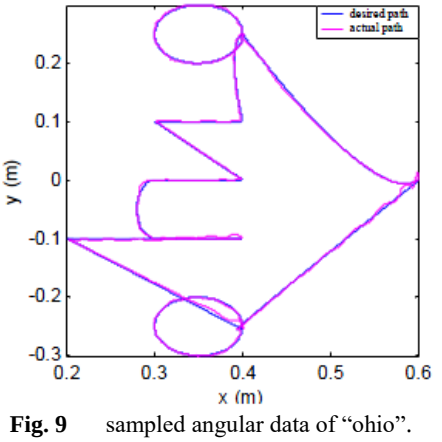


Fig. 9 sampled angular data of "ohio".

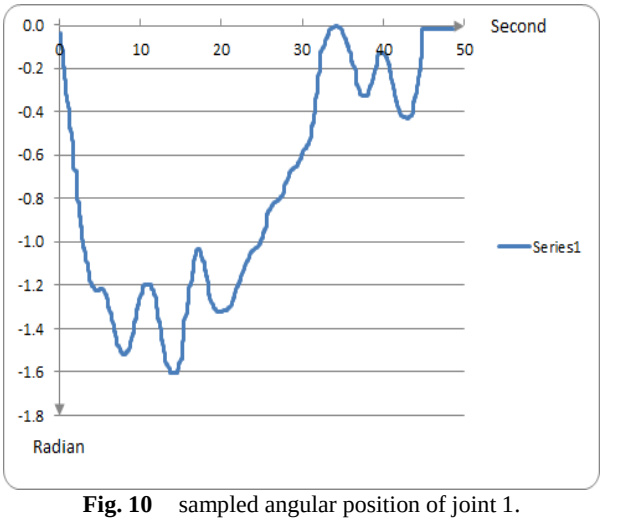


Fig. 10 sampled angular position of joint 1.

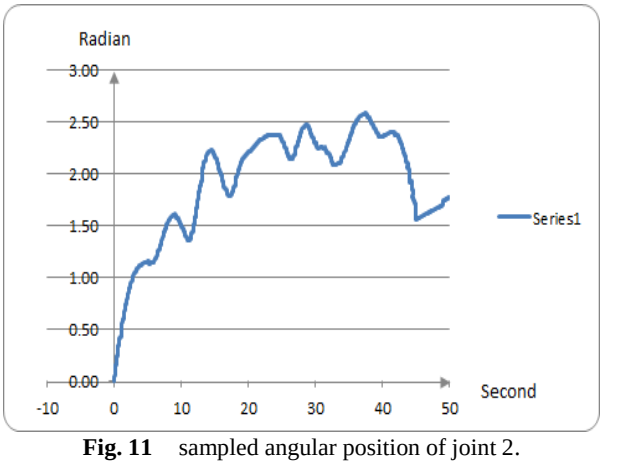


Fig. 11 sampled angular position of joint 2.

We obtained the base records of reference robot, and the parameters were registered in a file as raw numerical values. The contents of this file were saved in MMC card and placed in the designed test rig illustrated in "Fig. 12". The program reads the data in FAT32 format from MMC. Controller analyses the data and puts them in the controller EEPROM. When user runs the associated function, data will be read from EEPROM and the function of kinematic calculation will start. These data drive the test rig motors illustrated in "Fig. 12". The test rig output of ARM-IAUM is depicted in "Figs. 13 and 14". The records of each driver will be then read by an interrupt in 20ms intervals. These functions have been given during UART output port to the device display. Finally, after 50 seconds the data will be transferred into an editor. Variation of the samples extracted from the controller is shown on the "Figs. 15 and 16".

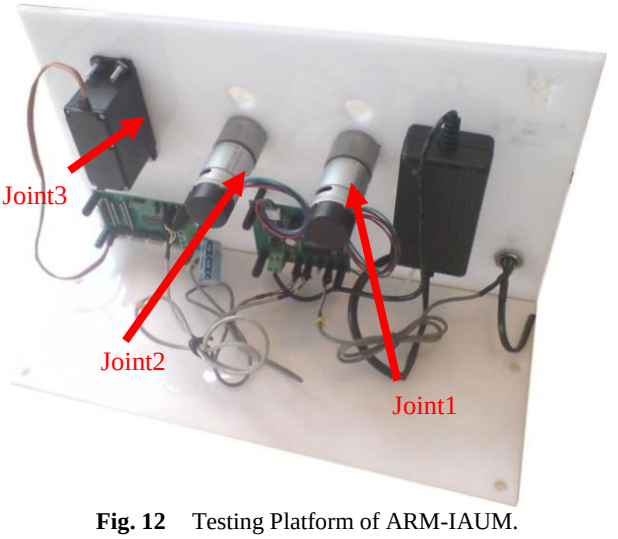


Fig. 12 Testing Platform of ARM-IAUM.

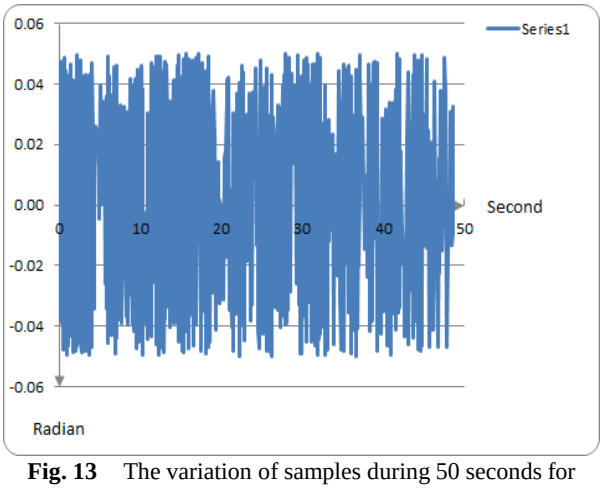


Fig. 13 The variation of samples during 50 seconds for First Joint of ARM-IAUM (RAD/S).

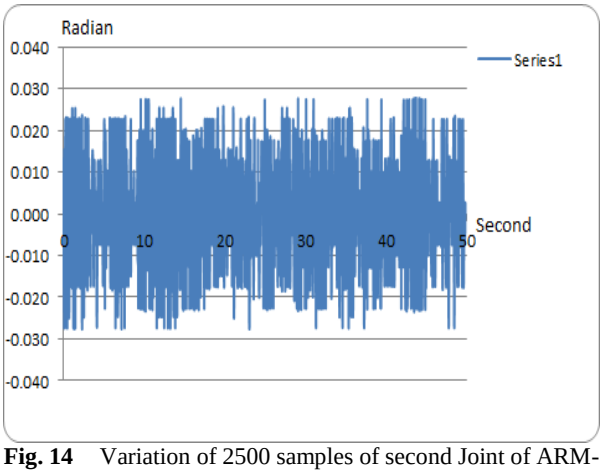


Fig. 14 Variation of 2500 samples of second Joint of ARM-IAUM.

We applied the analysed direct drive motion rate control method in the controller. The results show that the joint angles had a significant lower angular variation. The peak to peak angular variation at first joint is reduced from 0.1 in "Fig. 13" to almost 0.02 Radian in "Fig. 15". For the second joint, the peak to peak angular variation is reduced from 0.04 in "Fig. 14" to 0.009 in "Fig. 16". It means that implementing a direct drive robot with RMRC function can improve the pick and place accuracy up to 4 times. This effect not only for the first but also for all joints including the end-effector and second joint is true. Figure 17 and 18 show the deviation from the mean before and after using RMRC. The variance of deviation for each case is calculated:

$$\text{Varaince}_{\theta} = \sqrt{\sum_{1}^{2500}(\theta - \text{mean})^2/2500} = 0.255$$

$$\text{Varaince}_{\theta\ \text{RMRCC}} = \sqrt{\sum_{1}^{2500}(\theta - \text{mean})^2/2500} = 0.000029$$

The variance of deviation is reduced from 0.255 to 0.000029.

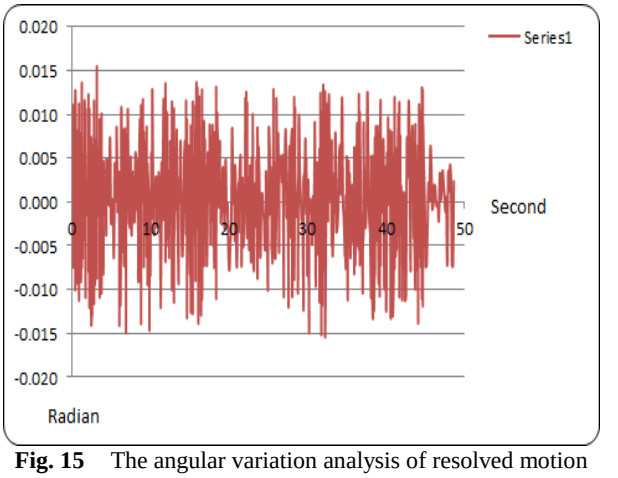


Fig. 15 The angular variation analysis of resolved motion rate control algorithm used by first joint in controller of ARM-IAUM.

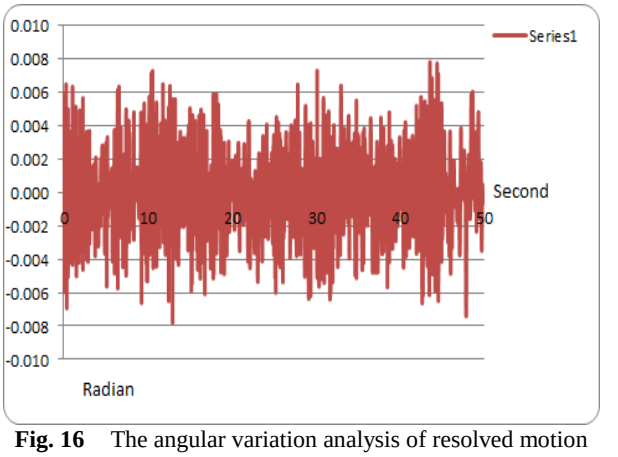


Fig. 16 The angular variation analysis of resolved motion rate control algorithm used by second joint in controller of ARM-IAUM.

From "Figs. 17 and 18", the deviation of the first joint has been increased by 0.2 radians. However, after

applying RMRC, the diversion of controlled rate approach is not greater than ±0.1. Figure 18 also shows how the resolved motion method has been controlling the deviation across the zero level. However, the controller in a none RMRC method as "Fig. 17", has not been able to resolve the variation of the joint angle.

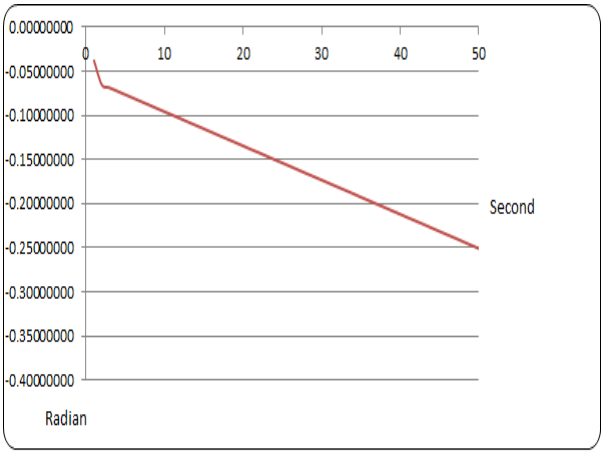


Fig. 17 Deviation of the mean for controller without using RMRC motion control algorithm.

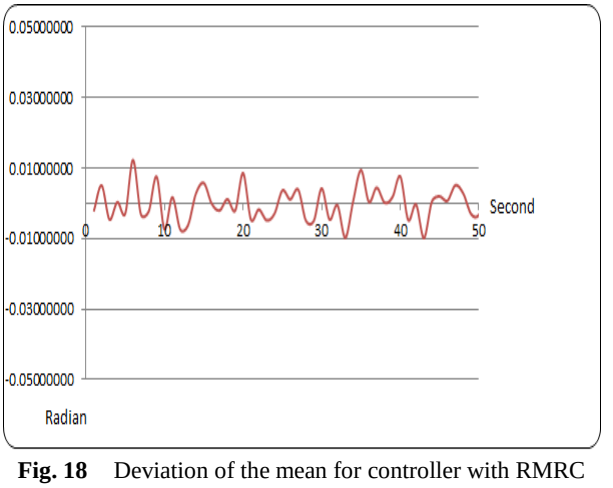


Fig. 18 Deviation of the mean for controller with RMRC motion control algorithm.

8 CONCLUSION

In this paper, we described the derivation of the RMRC method to control the new type of robots named direct-drive robots which are being vastly used in the industries. In this research, the concept of a robot controller with an RMRC method and associated formula is introduced and the results are compared with a none RMRC system. The performance of the method has been compared based on the variance of deviation of the angle of the joints from the reference in an RMRC and none RMRC platform. The results show that using the RMRC method improves the accuracy of the arm and increases the response time to correct the desired angle of each joint.

REFERENCES

[1] Saha, S. K., Introduction to Robots, McGraw-Hill, 3th edition. Canada, 2014.

[2] Coronel-Escamilla, A., et al. On the Trajectory Tracking Control for A Scara Robot Manipulator in A Fractional Model Driven by Induction Motors with PSO tuning, Multibody System Dynamics, Vol. 43, No. 3, 2018, pp. 257-277.

[3] Whitney, D., Resolved Motion Rate Control of Manipulators and Human Prostheses IEEE Transactions on Man Machine Systems, Vol. 10, No. 2, pp. 47-53, 1969.

[4] Dessaint, L. A., Saad, M., Hebert, B., and Al-Haddad, K., An Adaptive Controller for a Direct-Drive SCARA Robot, IEEE Transactions on Industrial Electronics, Vol. 39, No. 2, 1992, pp. 105-111.

[5] Larry, T. R., Stephen, W. F., James, W. M., and Robert, L. T., Robotics Theory and Industrial Applications, Goodheart-Willcox, 2th Edition. USA, 2011, pp. 23-27.

[6] Alshamasin, M. S., Ionescu, F., and Al-Kasasbeh, R. T., Kinematic Modeling and Simulation of a SCARA Robot by Using Solid Dynamics and Verification by MATLAB/Simulink, European Journal of Scientific Research, Vol. 37, No. 3, 2009, pp. 388-405.

[7] Craig, J., Introduction to Robotics Mechanics and Control, Addison-Wesley, 3th Edition. USA, 2009.

[8] Tang, Zh., Ying, P., and Zhi Ch., Vibration Analysis and Passive Control of SCARA Robot Arm. Proceedings of the 2019 International Conference on Robotics, Intelligent Control and Artificial Intelligence, 2019.

[9] Koyuncu, B., Güzel, M., Software Development for the Kinematic Analysis of a Lynx 6 Robot Arm, International Journal of Engineering and Technology, Vol. 4, 2008.

[10] Spong, W., Hutchinson, S., and Vidyasaqar, M., Robot Modelling and Control, Wiley, 1th Edition's, 2005.

[11] Ruibo, H., Yingjun, Zh., Shunian, Y., Shuzi, Y., 1.1.1. Kinematic-Parameter Identification for Serial-Robot Calibration Based on POE Formula, IEEE Transactions on Robotics, Vol. 26, No. 3, 2010, pp. 411-423

[12] He, Yunbo, et al. Dynamic Modeling, Simulation, and Experimental Verification of a Wafer Handling SCARA Robot with Decoupling Servo Control. IEEE Access, Vol. 7, 2019, pp. 47143-47153.

[13] Ernur, K., Teleoperation of an Industrial Robot Using Resolved Motion Rate Control with Visual Servoing. Diss. Ohio University, 2005.