Contents lists available at *Dergipark*

Journal of Scientific Reports-A

journal homepage: *https://dergipark.org.tr/tr/pub/jsr-a*

Model-based nonlinear control applications with kinematic and dynamic analysis of 4 DoF SCARA robot manipulator

Halit Tirsi[a], Kamil Çetin[b,*]

[a]*Department of Electrical and Electronics Engineering, İzmir Katip Çelebi University, İzmir, Türkiye*, ORCID: 0009-0002-1749-1347
[b]*Akıllı Fabrika Sistemleri Uygulama ve Araştırma Merkezi, İzmir Katip Çelebi University, İzmir, Türkiye*, ORCID: 0000-0003-1029-5626

Abstract

The aim of this paper is to research and develop the model-based nonlinear control strategies (joint space position, Cartesian space position and hybrid position/force) of a 4 DoF SCARA robotic arm. To support these strategies, mathematical kinematics and dynamic equations are formulated using a dynamic model for prediction purposes. The joint space controller focuses on joint angles tracking while the end effector's position is controlled with the Cartesian space controller. The hybrid controller allows interaction with the surface and control of the interaction force while maintaining a given Cartesian position. The motion trajectories are planned considering the kinematic model of the robotic arm. The study is employed in MATLAB Simulink where kinematic and dynamics models, trajectory generation and control blocks are integrated into a simulation environment. Results indicate that the torque limits of the robotic arm are sufficient for effective trajectory tracking within the imposed constraints. This research is of particular significance as it aims to conduct appreciable kinematic and dynamics studies of one of the most common types of 4 DoF SCARA robotic arm and develops precise model-based nonlinear controllers for the industrial robots.

Keywords: Cartesian space; Dynamic model; Joint space; Kinematic model; Nonlinear control; Robotic arm.

1. Introduction

In recent years, robotic arms have been extensively used and this has been facilitated by the fact that they are accurate, efficient, and can work in a wide range of environments [1, 2]. Robotic arms have been increasingly used to extend productivity, safety and cost effectiveness from manufacturing assembly line to medical operations [3, 4]. With the increasing market for automation, there also seems to be an increasing market for sophisticated robotic arms that do not only perform their duties in the traditional manner but can also be more flexible and agile [5, 6]. One such robotic arm which drew considerable attention is the 4 Degree-of-Freedom (DoF) SCARA (Selective Compliance Assembly Robot Arm) robotic arm [7]. In the contemporary enterprises SCARA robotic arms are rather used in

* Corresponding author. Tel.: +90 232 2705500
E-mail address: kamil.cetin@ikcu.edu.tr

carrying out different work including assembler operations, pick and place operations, and material handling operations [8]. Unlike traditional ones that consist of several joints, the SCARA design has only 4 DoF. This simplifies the overall architecture, improved stiffness and speed without having a heavy profile [9].

In a 4 DoF SCARA arm, which in three revolute joints can perform motion in a horizontal plane (typically the x-y plane) and in one prismatic joint vertical movement, it is referred as either RRPR or RRRP (three revolute and one prismatic joints) arm provides great benefits. These various kinematic structures complete the arm with a wide range of possible functions, together with sufficient precision of movement and the ability to adapt to a wide variety of workpieces. This makes it ideal for applications that require both flexibility and accuracy in the performance of work [10, 11]. For digital twin studies, which are among the most important research topics of recent years, a digital twin successfully developed using a Scara robot in [12]. The study in [13] demonstrated the use of artificial neural networks in obtaining forward kinematics of the SCARA robot arm. In [14], the research investigated the motion behaviour and distance error of the SCARA robot along the X and Y axes at the end point via MATLAB. In the study [15], experimental tests were conducted to evaluate the tracking capabilities of a SCARA robot with different commanded trajectories and speeds via a PLC-based controller. In [16], a SCARA robot used for horizontal and vertical drilling in industries such as manufacturing and construction was modelled and simulated. As mentioned above, the SCARA robot arm is one of the most widely used robot arms in the industry today, as it has been for many years, and is still one of the most widely used and modelled and simulated robot arms in academic studies. There are very few studies in the literature that present simulation studies of nonlinear control methods together with both dynamic and kinematic analysis of such robot arms.

In this article, three different model-based nonlinear control methods are studied, namely position control in joint space, position control in Cartesian space and hybrid position/force control in Cartesian space of 4 DoF SCARA robotic arm. First, kinematic and dynamic motion equations of the robotic arm are defined mathematically. These mathematical equations are used to predict the motions of the robotic arm in the workspace for model-based control methods. In joint space, the aim is to reach the desired joint angles of the robotic arm with the position controller. In Cartesian space, the aim is to reach the desired point in the Cartesian coordinates of the robotic arm's end effector with the position controller. Finally, in Cartesian space, the hybrid position/force controller enables both the position of the robot arm's end effector in Cartesian coordinates and the interaction force with the moving surface to be followed. For model-based control methods designed according to the dynamic model of the SCARA robotic arm, position and force trajectory planning is made in both joint space and Cartesian space. Dynamic and kinematic model blocks, trajectory tracking model block and control blocks of the robotic arm were created in MATLAB Simulink simulation environment. According to the simulation results, these model-based control methods successfully followed the desired trajectories in joint space and Cartesian space according to the dynamic structure of the robotic arm. In this article, mathematical analysis of the kinematic and dynamic model, trajectory planning in joint space and Cartesian space, model-based nonlinear controller design and simulation studies for the 4 DoF SCARA robotic arm, which is one of the most widely used robot arms in industrial environments, are presented.

2. Mathematical model of 4 DoF SCARA

In this section, the kinematic and dynamic model equations of 4 DoF SCARA robotic arm are given in detail. As shown in Fig.1, the 4 DoF SCARA robotic arm's trajectory generation, forward kinematic, Jacobian, and dynamic model blocks are created in the MATLAB Simulink.

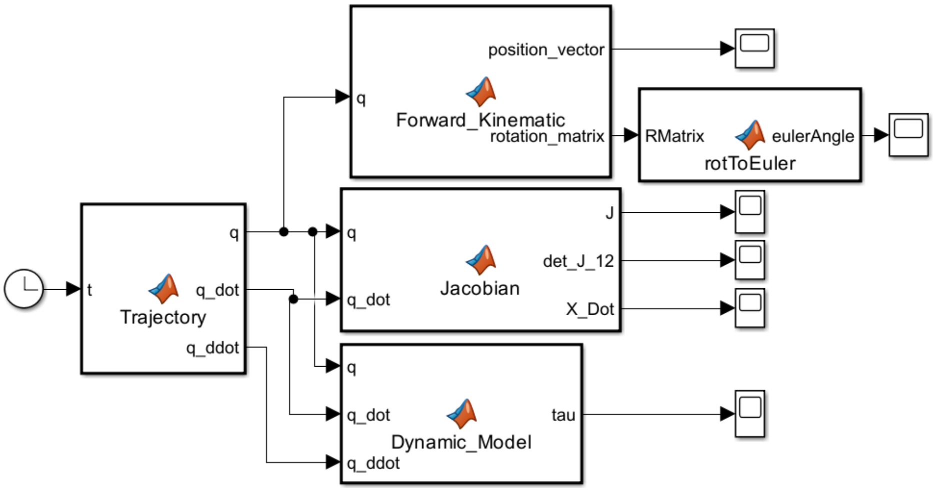

Fig.1. The 4 DoF SCARA robotic arm's model blocks in MATLAB Simulink.

2.1. Forward kinematics

A kinematic model is a mathematical equation that geometrically describes the mechanical movements of a robot. This model defines the fixed and variable parameters of the robotic arm's joints and the relationships between the joints. The kinematic model is used to calculate and control the necessary movements between joints for a robot to reach a desired trajectory. It plays an important role in determining how robotic arms will move to reach their target positions [17]. The DH parameters, also known as the Denavit-Hartenberg representation, define the geometric relationship between each robot joint. These parameters define the position of joints, the orientation of axes, and the motion between them [18]. DH parameters provide a fundamental structure for the mathematical representation of kinematic chains, accurately modelling the robot's motion. An Epson model SCARA robotic arm is shown in Fig.2.

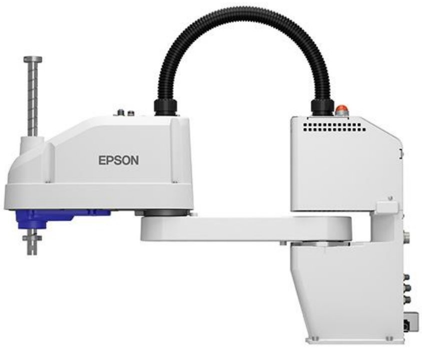

Fig.2. EPSON SCARA robotic arm.

Fig.3 shows the connections and variables of the SCARA arm. The joints of the SCARA arm are in RRPR configuration.

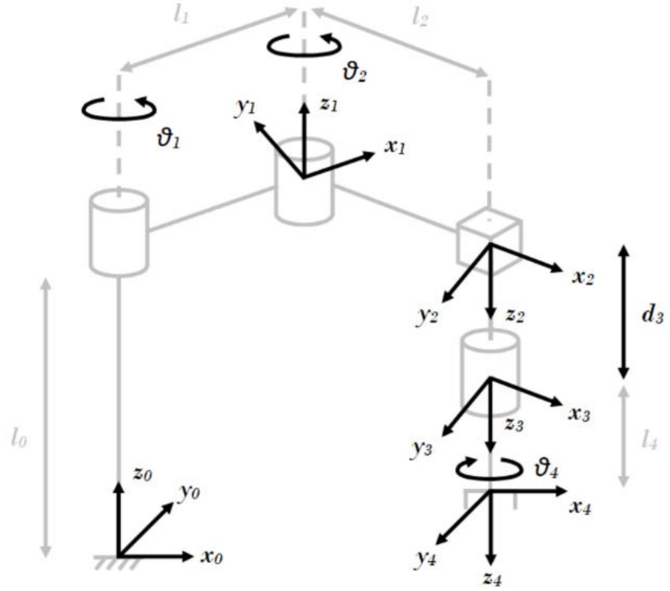

Fig.3. Schematic of SCARA arm with frame assignments [24], [25].

DH parameters of SCARA arm are given in Table 1.

Table 1. DH parameters.

Link i	ai	αi	di	θi
1	l1	0	l0	θ1
2	l2	π	0	θ2
3	0	0	d3	θ3
4	0	0	l4	θ4

Link-transformation matrices are calculated for each frame to define forward kinematics. In Table 1, a_i is link-length, α_i is link-twist, d_i is link-offset, and θ_i is joint-angle. The link-transformation matrices from base to the end-link are given in the equations (1-6).

$$A_1 = \begin{bmatrix} c_1 & -s_1 & 0 & l_1 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & l_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}$$

$$A_2 = \begin{bmatrix} c_2 & s_2 & 0 & l_2 \\ s_2 & -c_2 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}$$

$$A_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

$$A_4 = \begin{bmatrix} c_4 & -s_4 & 0 & 0 \\ s_4 & c_4 & 0 & 0 \\ 0 & 0 & 1 & l_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}$$

$$T_4^0 = A_1 A_2 A_3 A_4 \tag{5}$$

$$T_4^0 = \begin{bmatrix} c_{124} & s_{124} & 0 & l_1 c_1 + l_2 c_{12} \\ s_{124} & -c_{124} & 0 & l_1 s_1 + l_2 s_{12} \\ 0 & 0 & -1 & l_0 - l_4 - d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6}$$

where c_i represent cosines(θ_i), s_i represent sinus(θ_i), c_{12} is cosines(θ_1+θ_2) and s_{12} is sinus(θ_1+θ_2). T_4^0 is the product of link-transformation matrices A_1, A_2, A_3, A_4.

The first three columns and rows of the 4x4 transformation matrix represent a 3x3 rotation matrix. This matrix expresses the rotation in three-dimensional space by determining the amount and direction of rotation of the object. The fourth column with three rows represents a 3x1 position vector, indicating where the object is in three-dimensional space. The rotation matrix and the position vector are given in the equations (7-8).

$$R_4^0(\theta) = \begin{bmatrix} c_{124} & c_{124} & 0 \\ s_{124} & -c_{124} & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{7}$$

$$P_4^0(\theta) = \begin{bmatrix} l_1 c_1 + l_2 c_{12} \\ l_1 s_1 + l_2 s_{12} \\ l_0 - l_4 - d_3 \end{bmatrix} \tag{8}$$

2.2. Velocity kinematics

Velocity kinematics in the equation (9) relates the joint velocities $\dot{\theta}_1$ and end-effector velocities v of a robot with linear $\dot{p}$ and angular velocities ω. The Jacobian matrix $J(\theta)$ is used for velocity kinematics. This matrix is used to predict and control how an end-effector of the robotic arm will move at given joint velocities. The relationship is used to calculate the joint velocities needed for a robot's end-effector to move at an intended speed. Thus, it enables a robot to reach desired positions and velocities [19].

$$v_4^0 = \begin{bmatrix} \dot{P}_4^0 \\ \omega_4^0 \end{bmatrix} = \begin{bmatrix} -l_2(\dot{\theta}_1 + \dot{\theta}_2) sin(\theta_1 + \theta_2) - l_1 \dot{\theta}_1 sin(\theta_1) \\ l_2(\dot{\theta}_1 + \dot{\theta}_2) cos(\theta_1 + \theta_2) + l_1 \dot{\theta}_1 cos(\theta_1) \\ -\dot{d}_3 \\ 0 \\ 0 \\ \dot{\theta}_1 + \dot{\theta}_2 - \dot{\theta}_4 \end{bmatrix} \tag{9}$$

The Jacobian matrix of the robotic arm basically defines the mapping between the joint velocities $\dot{\theta}$ and the end effector velocities $\dot{X}$

$$\dot{X} = J(\theta)\dot{\theta} \tag{10}$$

As a result of the equations provided above, the Jacobian matrix for the 4 DoF SCARA arm has been determined to be of size 6x4 in the equation (11).

$$J(\theta) = \begin{bmatrix} -l_2 s_{12} - l_1 s_1 & -l_2 s_{12} & 0 & 0 \\ l_2 c_{12} + l_1 c_1 & l_2 c_{12} & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & -1 \end{bmatrix} \tag{11}$$

As given in the equations (12-14), the Jacobian matrix can be divided into two components: translational Jacobian (J_P) and rotational Jacobian(J_O), allowing for singularity analysis.

$$J = \begin{bmatrix} J_P \\ J_O \end{bmatrix} \tag{12}$$

$$J_P = \begin{bmatrix} -l_1 s_1 - l_2 s_{12} & -l_2 s_{12} & 0 & 0 \\ l_1 c_1 - l_2 c_{12} & l_2 c_{12} & 0 & 0 \\ 0 & 0 & -1 & 0 \end{bmatrix} \tag{13}$$

$$J_O = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & -1 \end{bmatrix} \tag{14}$$

In robotics, if J_{12} becomes rank-deficient, it signifies that the end effector has restricted maneuverability in the x-y Cartesian space. This leads to kinematic singularities were achieving finite speed in one direction requires infinite joint speed. Detecting these involves setting the determinant of J_{12} to zero.

$$J_{12} = \begin{bmatrix} -l_1 s_1 - l_2 s_{12} & -l_2 s_{12} \\ l_1 c_1 - l_2 c_{12} & l_2 c_{12} \end{bmatrix} \tag{15}$$

$$det(J_{12}) = 0 = sin(\theta_2) \tag{16}$$

When θ_2 assumes values of 0 or 180 degrees, it signifies the presence of a singularity issue.

2.3. Dynamic model

The dynamic model encapsulates the mathematical framework elucidating the mechanics of motion within a robot or robotic system, defining how forces and moments affecting its motion evolve through differential equations or matrices. This model not only facilitates understanding and controlling the dynamic behaviour of the robot, enabling predictions of its velocity, acceleration, position, and other motion characteristics crucial for design, control, and simulation but also signifies a critical transition between kinematic and dynamic models. The transition integrates data on the masses and moments of inertia of each link. This integrated data enables the analysis of the forces acting on the robot arm and the forces (or torques) required by the motor of each link. The analyzed force-torque values enable the effective regulation of the robotic arm's motion. [20].

The rigid-body dynamic model equation that describes the robotic arm's motion is given in (17)

$$\tau = M(\theta)\ddot{\theta} + V\left(\theta, \dot{\theta}\right) + G(\theta) \tag{17}$$

where $\ddot{\theta}$ is the joint acceleration, τ represents the outward forces or moments acting on the arm, $M(\theta)$ is the inertia-mass matrix, $V(\theta,\dot{\theta})$ captures the Coriolis and centrifugal effects, and $G(\theta)$ signifies the gravitational effects.

The calculations for the mass matrices to be used in the dynamic model are given in the equations (18-29) specifically for the SCARA robotic arm.

$$M(\theta)=\begin{bmatrix} M_{11} & M_{12} & 0 & M_{14} \\ M_{21} & M_{22} & 0 & M_{24} \\ M_{31} & 0 & M_{33} & 0 \\ M_{41} & M_{42} & 0 & M_{44} \end{bmatrix} \tag{18}$$

$$M_{11}=I_{zz1}+I_{zz2}+I_{zz3}+I_{zz4}+m_1\left((l_1+l_{c1x})^2+{l_{c1y}}^2\right)+m_2\left({l_1}^2+(l_2+l_{c2x})^2+{l_{c2y}}^2\right)+m_3\left({l_1}^2+(l_2+l_{c3x})^2+{l_{c3y}}^2\right)+m_4\left({l_1}^2+{l_2}^2+{l_{c4x}}^2+{l_{c4y}}^2\right)+(m_2+m_3+m_4)l_1l_2c_2+m_2l_1\left(l_{c2x}c_2+l_{c2y}s_2\right)+m_3l_1\left(l_{c3x}c_2+l_{c3y}s_2\right)+m_4\left(l_{c4x}(l_1c_{2\bar{4}}+2l_2c_4)+l_{c4y}(l_1s_{2\bar{4}}-2l_2s_4)\right) \tag{19}$$

$$M_{12}=M_{21}=I_{zz_2}+I_{zz_3}+I_{zz_4}+m_2\left((l_2+l_{c2x})^2+{l_{c2y}}^2\right)+m_3\left((l_2+l_{c3x})^2+{l_{c3y}}^2\right)+m_4\left({l_2}^2+{l_{c4x}}^2+{l_{c4y}}^2\right)+(m_2+m_3+m_4)l_1l_2c_2+m_2l_1\left(l_{c2x}c_2+l_{c2y}s_2\right)+m_3l_1\left(l_{c3x}c_2+l_{c3y}s_2\right)+m_4\left(l_{c4x}(l_1c_{2\bar{4}}+2l_2c_4)+l_{c4y}(l_1s_{2\bar{4}}-2l_2s_4)\right) \tag{20}$$

$$M_{22}=I_{zz_2}+I_{zz_3}+I_{zz_4}+m_2\left((l_2+l_{c2x})^2+{l_{c2y}}^2\right)+m_3\left((l_2+l_{c3x})^2+{l_{c3y}}^2\right)+m_4\left({l_2}^2+{l_{c4x}}^2+{l_{c4y}}^2+2l_2\left(l_{c4x}c_4+l_{c4y}s_4\right)\right) \tag{21}$$

$$M_{14}=M_{41}=-m_4\left(l_{c4x}(l_{c4x}+l_2c_4+l_1c_{2\bar{4}})+l_{c4y}\left(l_{c4y}-l_2s_4+l_1s_{2\bar{4}}\right)\right)-I_{zz4} \tag{22}$$

$$M_{24}=M_{42}=-m_4\left({l_{c4x}}^2+{l_{c4y}}^2+l_2\left(l_{c4x}c_4+l_{c4y}s_4\right)\right)-I_{zz_4} \tag{23}$$

The following equations illustrate the Coriolis matrix and the gravitational force matrix, both of which are essential components in the dynamic model.

$$V(\theta,\dot{\theta})=\begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix} \tag{24}$$

$$V_1=0 \tag{25}$$

$$V_2=l_1\dot{\theta}_1\left(\left(\dot{\theta}_1+\dot{\theta}_2\right)\left((m_2+m_3+m_4)l_1s_2+(m_2l_{c2x}+m_3l_{c3x})s_2\right)-l_1\dot{\theta}_1\left(\left(\dot{\theta}_1+\dot{\theta}_2\right)\left(m_2l_{c2y}+m_3l_{c3y}\right)c_2+\left(\dot{\theta}_1+\dot{\theta}_2-\dot{\theta}_4\right)\left(m_4l_{c2x}s_{2\bar{4}}-m_4l_{c4y}c_{2\bar{4}}\right)\right)\right) \tag{26}$$

$$V_3 = 0 \tag{27}$$

$$V_4 = m_4(\dot{\theta}_1 + \dot{\theta}_2 - \dot{\theta}_4)\left(l_2(\dot{\theta}_1 + \dot{\theta}_2)(l_{c4y}c_4 + l_{c4x}s_4) + m_4(\dot{\theta}_1 + \dot{\theta}_2 - \dot{\theta}_4)l_1\dot{\theta}_1(l_{c4y}c_{2\bar{4}} - l_{c4x}s_{2\bar{4}})\right) \tag{28}$$

$$G(\theta) = \begin{bmatrix} 0 \\ 0 \\ g(m_3 + m_4) \\ 0 \end{bmatrix} \tag{29}$$

All these dynamic model terms are employed to analyse the motion and force balance within the SCARA robotic arm. The table below presents all inertial parameters utilized in the simulation.

Table 2. Inertia parameters for four links.

Link	Link 1	Link 2	Link 3	Link 4
Length (m)	0.4	0.4	d_3	0.15
Mass (kg)	6.01	5.37	4.03	0.91
l_{cix} (m)	-0.185	-0224	0	0
l_{ciy} (m)	0	0	0	0
l_{ciz} (m)	0	0	-0.201	-0.122
$l_{xx_i}(kg.m^2)$	0.0132	0.0234	0.0802	0.0016
$l_{yy_i}(kg.m^2)$	0.1810	0.1261	0.0802	0.0016
$l_{zz_i}(kg.m^2)$	0.1807	0.1558	0.064	0.0025

3. Model-based nonlinear controllers

Model-based control is an approach used to design control strategies for a system by utilizing its mathematical model. This method employs the model to predict the system's behaviour and achieve the desired performance. To reach predefined objectives, it computes appropriate control signals using the system's model and current state. While this approach can be effective in controlling complex systems, it may encounter challenges such as the accuracy of the model and managing uncertainties within the system [21].

3.1. Joint space position controller

The joint space position controller aims that the joint positions θ of a robotic arm tracks the desired joint positions, denoted by θ_d. Fig.4 shows the model-based joint space position controller for the arm.

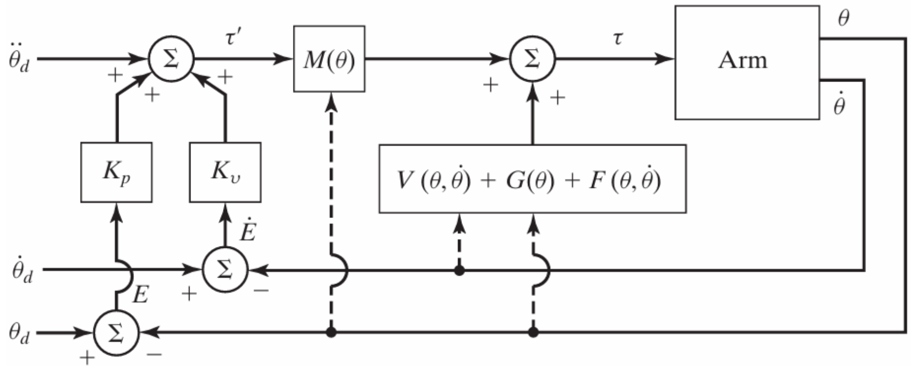

Fig.4. Model-based joint space control schematic [26].

Friction-added $F_r(\theta,\dot{\theta})$ rigid dynamic model of the arm is given in the equation (30)

$$\tau = M(\theta)\ddot{\theta} + V(\theta,\dot{\theta}) + G(\theta) + F_r(\theta,\dot{\theta}) \tag{30}$$

where the friction model consists of Coulomb friction f_c and viscous friction f_v formulas as given in the equations (31-32)

$$f_c = c\ sgn(\dot{\theta}) \tag{31}$$

$$f_v = v\ \dot{\theta} \tag{32}$$

where c and v are the Coulomb friction and viscous friction coefficients, and $\mathrm{sgn}(.)$ means that the signum function. The total friction model is given in the equation (33)

$$F_r = f_c + f_v \tag{33}$$

The position error term e gives the difference between the intended position and the actual position, whereas the velocity error term $\dot{e}$ is the difference between the intended velocity and actual velocity as defined in the equations (34-35)

$$e = \theta_d - \theta \tag{34}$$

$$\dot{e} = \dot{\theta_d} - \dot{\theta} \tag{35}$$

Applying the partitioned control approach, the model-based joint space position controller is defined in the equation (36)

$$\tau = M\left(\ddot{\theta}_d + k_v\dot{e} + k_p e\right) + V + G + F_r \tag{36}$$

where k_v and k_p are the control gains. Fig.5 shows the model-based joint space position controller in MATLAB Simulink.

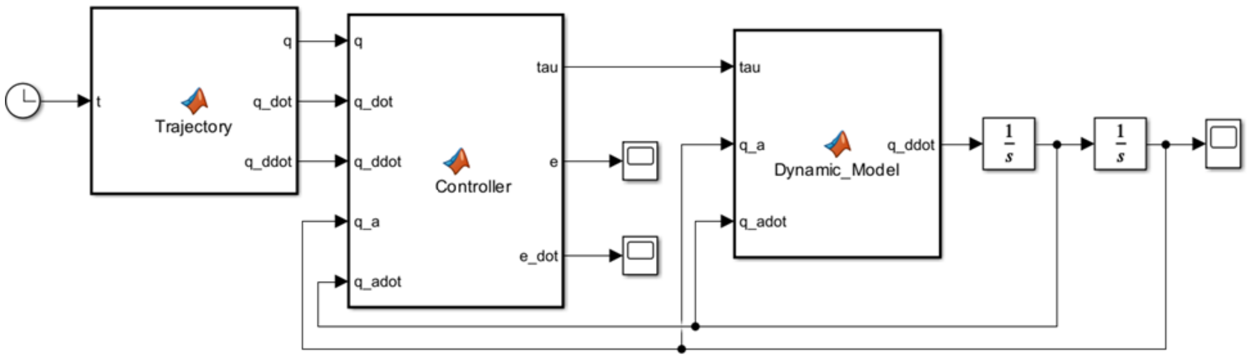

Fig.5. Model-based joint space control in Simulink.

3.2. Cartesian space position controller

Robot control in Cartesian space is an approach to control systems where the position, orientation, and velocity of the target point (usually considered the robot's end effector) at which the robot achieves its goal are defined in 6 DoF Cartesian coordinates. This approach defines the motion of the robot's end effector within a specific coordinate system (usually relative to the robot's fixed base) and is used to obtain a specific position and velocity along each axis of the robotic arm. The Cartesian coordinates represent the precise position of the robotic arm's base, joint frames, and end effector in a plane or space. This control approach allows the robotic arm's end effector to directly reach the intended position and facilitates the execution of complex joint movements [22]. In Cartesian space, the position and velocity error terms are defined in (37-38)

$$e = X_d - X \tag{37}$$

$$\dot{e} = \dot{X_d} - \dot{X} \tag{38}$$

where X and $\dot{X}$ are the actual end-effector position and velocity, respectively, and X_d and $\dot{X_d}$ are the intended end-effector position and velocity, respectively. Fig.6 shows the application of the model-based Cartesian position control scheme.

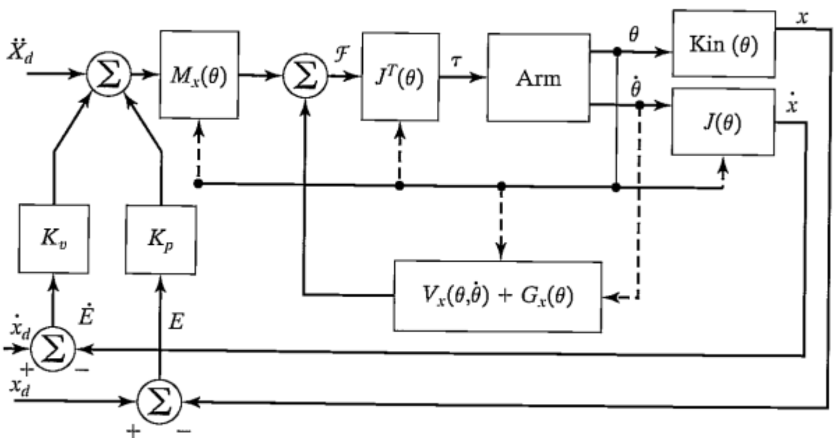

Fig.6. Model-based Cartesian space control schematic [26].

The Cartesian-based rigid-body dynamics can be written in (39)

$$F = M_x(\theta) + V_x(\theta, \dot{\theta}) + G_x(\theta) \tag{39}$$

where F is a hypothetical force-moment applied to the end-effector of the robotic arm. The dynamic model terms in joint space are formulated in Cartesian space by using the Jacobian of the arm as given in (40-42)

$$M_x(\theta) = J^{-T}(\theta) M(\theta) J^{-1}(\theta) \tag{40}$$

$$V_x(\theta, \dot{\theta}) = J^{-T}(\theta)\left(V(\theta, \dot{\theta}) - M(\theta) J^{-1}(\theta) \dot{J}(\theta) \dot{\theta}\right) \tag{41}$$

$$G_x(\theta) = J^{-T}(\theta) G(\theta) \tag{42}$$

Applying the partitioned control approach, the model-based Cartesian space position controller is defined in (43)

$$F = M_x \left(\ddot{X}_d + k_v \dot{e} + k_p e\right) + V_x + G_x \tag{43}$$

Then the torques that need to be applied to the joints are computed as (44)

$$\tau = J^T F \tag{44}$$

Fig.7 shows the blocks designed in the MATLAB Simulink for Cartesian space position control.

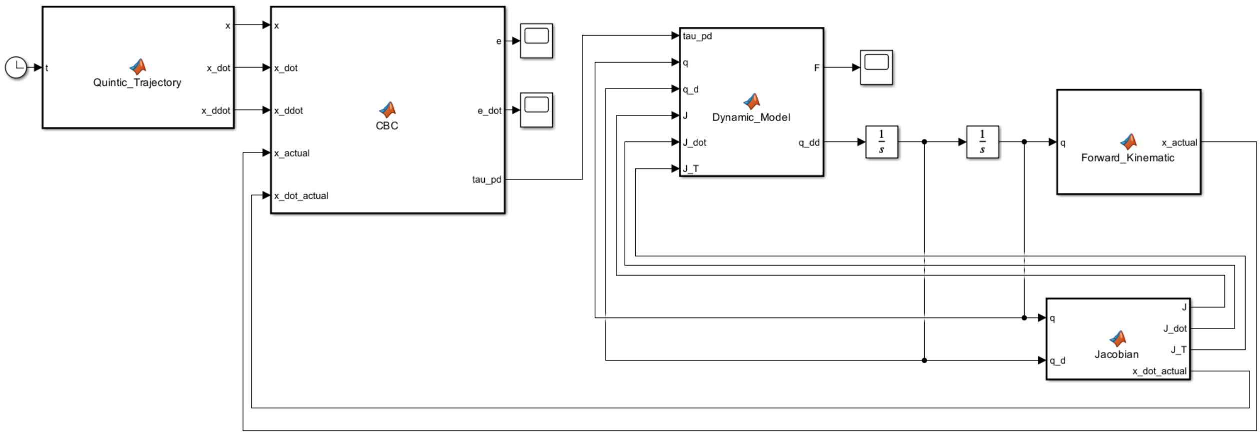

Fig.7. Model-based Cartesian space control in Simulink.

3.3. Cartesian space hybrid position/force control

Hybrid position/force controller in Cartesian space is a control system approach used to improve the physical interactions of the robotic arm with its environment, especially the end effector. Fig.8, in the control approach, the end effector of the robotic arm is moved to a certain position in Cartesian space and at the same time the robot end effector maintains its position with a certain force when interacting with objects or surfaces. In other words, thanks to the hybrid position/force controller that combines these two control modes, robotic arms are generally used for operations that require both position movement and a certain force movement, such as assembly, brushing, cutting, wiping [23].

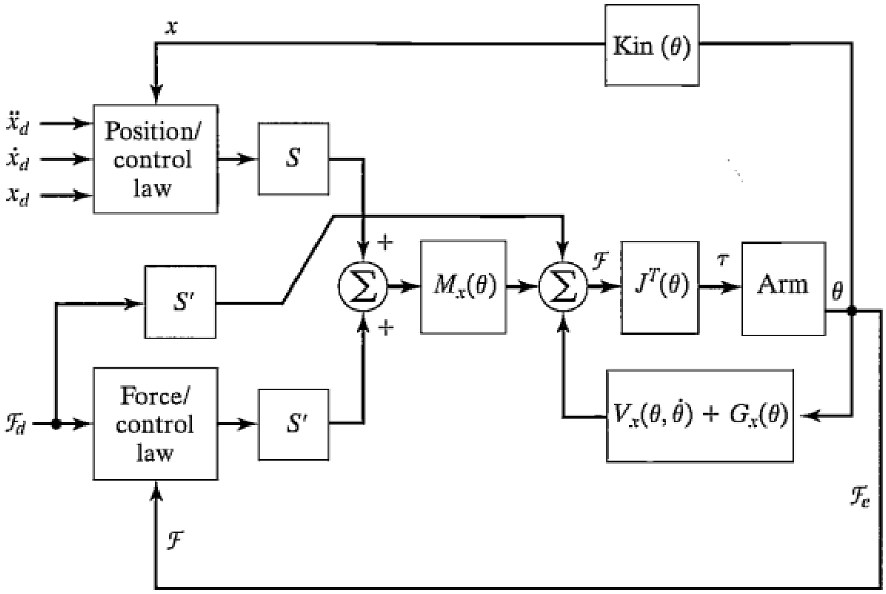

Fig.8. Model-based Cartesian space hybrid position/force control schematic [26].

Fig.9 shows the Hybrid Position/Force control block diagram in Simulink.

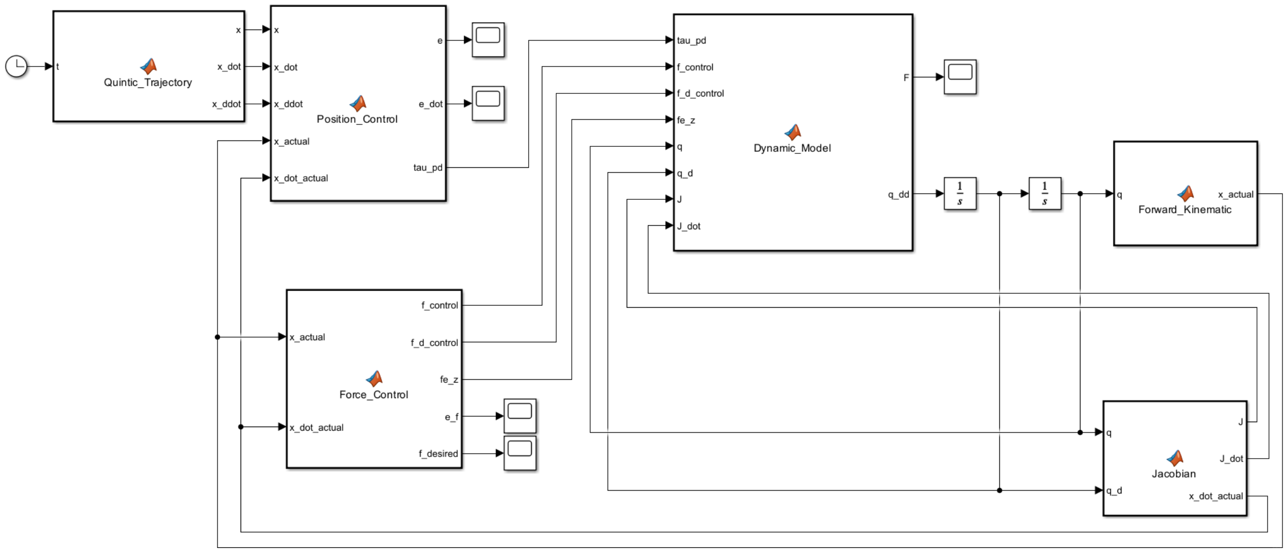

Fig.9. Model-based Cartesian space hybrid position/force control in Simulink.

The switch matrices S_p and S_f determine in which directions position and/or force control, respectively, will be made in the Cartesian space. In this hybrid control study, a Cartesian space position control is applied to the arm's end-effector in the x-y directions while a Cartesian space force control is applied to the arm's end-effector in the z direction with the following switch matrices in (45-46)

$$S_p = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{45}$$

$$S_f = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{46}$$

In this simulation study, an environmental contact force is generated by applying a spring model for the end-effector movement as the equation (47)

$$f_e = k_e x \tag{47}$$

where k_e is the stiffness of the environment model. The force error between the intended force, f_d, and the sensed force on the environment, f_e, as the equation (48)

$$e_f = f_d - f_e \tag{48}$$

The force control approach is defined by using the partitioned-controller concept for one-DoF spring-mass model as the equation (49)

$$F_f = m\left[k_{pf} k_e^{-1} e_f - k_{vf} \dot{x}\right] + f_d \tag{49}$$

where k_{pf} and k_{vf} are the force control gains. Cartesian space hybrid position/force controller is defined in the equation (50)

$$F = S_p\, F_p + S_f\, F_f \tag{50}$$

where F_p is Cartesian space position controller given in equation (43).

4. Simulation studies

This section presents the simulation results for all controllers. The simulation studies are performed for the kinematic and dynamic models of SCARA robotic arm given in previous sections using MATLAB Simulink. The simulations are run in 1kHz sampling frequency and for 10 seconds. The SCARA robot arm is aimed at performing a wide axis of the average movement capacity of the joints in 10 seconds according to the workspace and a movement of approximately 40 cm in diameter at the end effector. In addition, although control is provided by the nonlinear controller design based on the known equations of the model, the viscous and Coulomb friction effects in the joints and the situation where the robot carries a 1 kg load at the end effector are considered as disturbance effects for the controllers.

4.1. Results for joint space position controller

The arm's mathematical model was used by the model-based control method to forecast and accomplish desired performance. This method demonstrated precise control over the arm's movements, effectively managing the uncertainties in the system and the projected behaviour. The control precision was further improved by adding calculations for viscous and Coulomb friction. The quintic polynomial equation was used for the joints of the robot arm to reach the target angular positions [$\pi/4$; - $\pi/3$; 0.2; $3\pi/2$] from the initial positions. For the control gain values, k_p=diag([16 16 1000 64]) and k_v=diag([8 8 100 16]) were selected. The joint position error values are shown in Fig.10. The desired and actual joint positions are depicted in Fig.11. According to these results, joint position errors are less than 0.001rad for joints 1, 2 and 4, and 0.1mm for joint 3. However, the settling time is also less than 2 seconds at a very acceptable level.

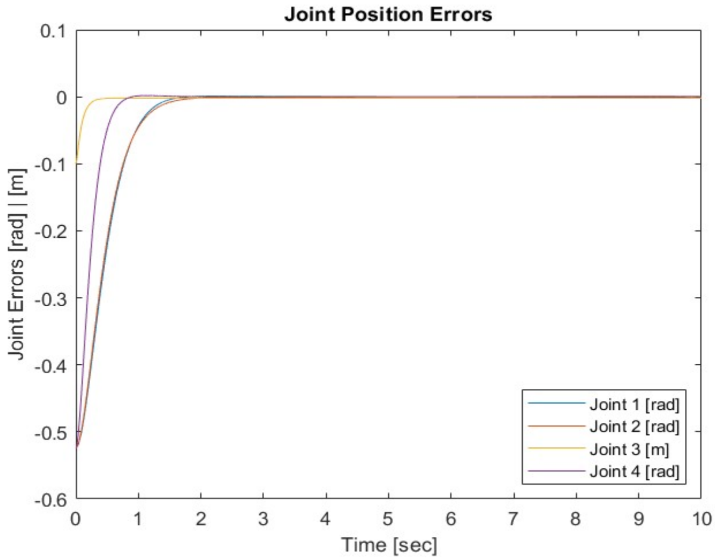

Fig.10. Joint position errors for model-based joint space position controller.

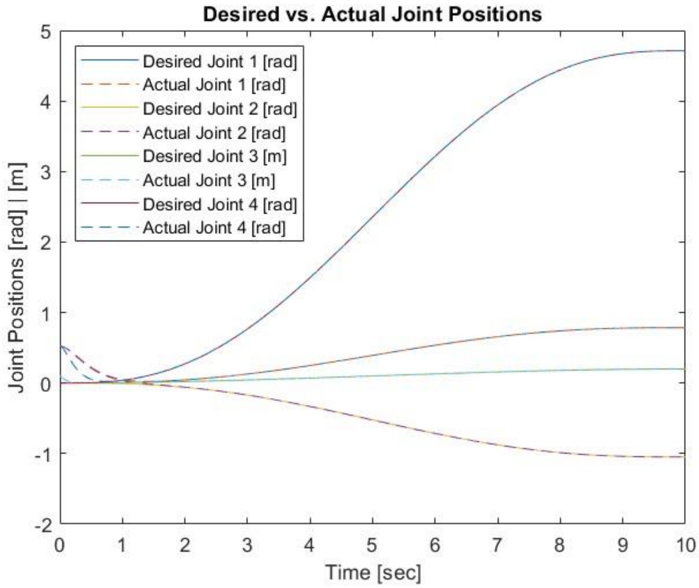

Fig.11. Desired versus actual joint positions for model-based joint space position controller.

4.2. Results for Cartesian space position controller

The positions of the robotic arm's end effector are calculated and controlled in x-y-z coordinates with the model-based Cartesian space controller. This method provides precise motion control of the robot's end effector in a coordinate system defined relative to the robot's fixed base, and therefore the model-based control method is advantageous for tasks that require precise positioning of the end effector. A simulation study was performed for the movement of the robot arm at the end point with 3D sinusoidal motion equations for a spiral laser cutting or welding robot movement. The desired motion equations in the XYZ axes at the end point were selected as x_d=[0.4+0.2sin(2π0.1t); 0.4+0.2cos(2π0.1t); 0.25+0.2sin(2π0.1t)]. For the control gain values, k_p=diag([400 400 400]) and k_v=diag([40 40 40]) were selected.

According to the simulation results showing the end effector position errors are less than 0.1mm and the actual end effector positions perfectly tracks the intended ones, respectively, in Fig.12 and Fig.13, it can be seen that the robotic arm achieves high accuracy in reaching the intended positions with this nonlinear model-based control method.

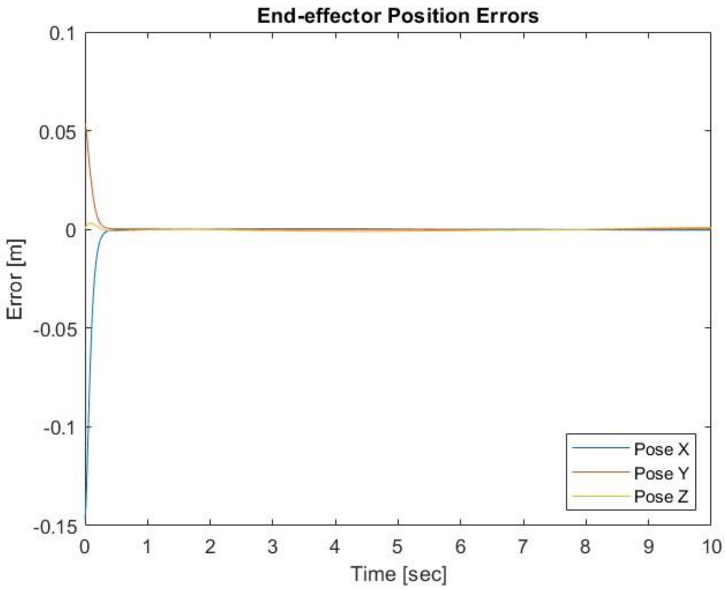

Fig.12. End-effector position errors for model based cartesian space position controller.

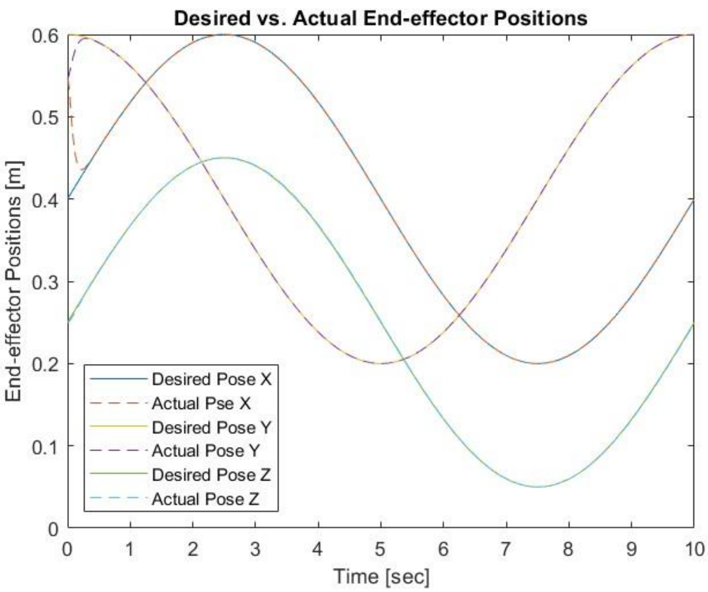

Fig.13. Desired versus actual end-effector positions for model based cartesian space position controller.

4.3. Results for hybrid position/force controller

The hybrid control method in Cartesian space combines both position and force control, allowing the robotic arm's end effector to interact with its environment precisely. A sinusoidal motion equation was planned in the XY axes at the end effector of the robot arm as follows; x_d=[0.4+0.2sin(2π0.1t); 0.4+0.2cos(2π0.1t)]. On the other hand, in the Z axis, the desired force equation was aimed at being in contact with the surface up to 2N force with a classical mass-spring-damper model. For control gain values, k_p=diag([400 400 300]) and k_v=diag([40 40 10]) were selected.

Fig.14 shows the end effector position errors in x-y directions (less than 0.1mm) and the force error in z direction (less than 0.1mN) in Cartesian space. Fig.15 shows the intended and actual end effector positions in x-y directions and the intended and actual force in z direction. These simulation results show that the SCARA robotic arm can maintain positional accuracy despite force levels at its end effector with the hybrid position/force control method.

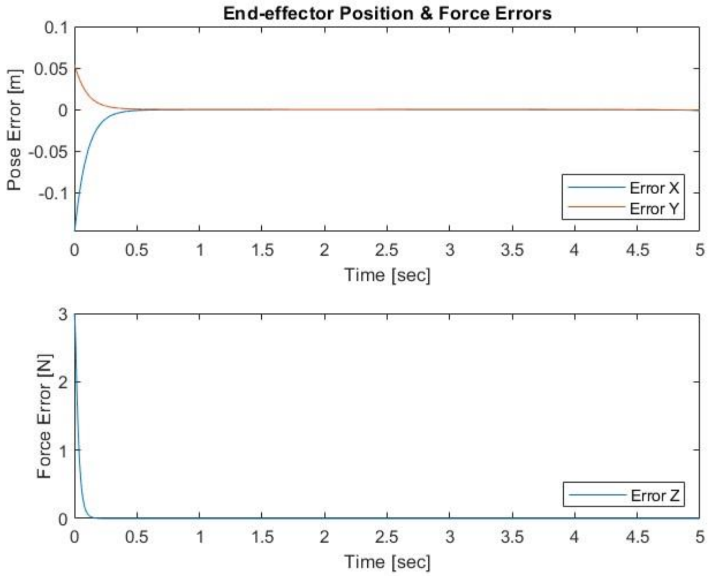

Fig.14. End-effector position errors on X and Y for model-based hybrid position/force controller.

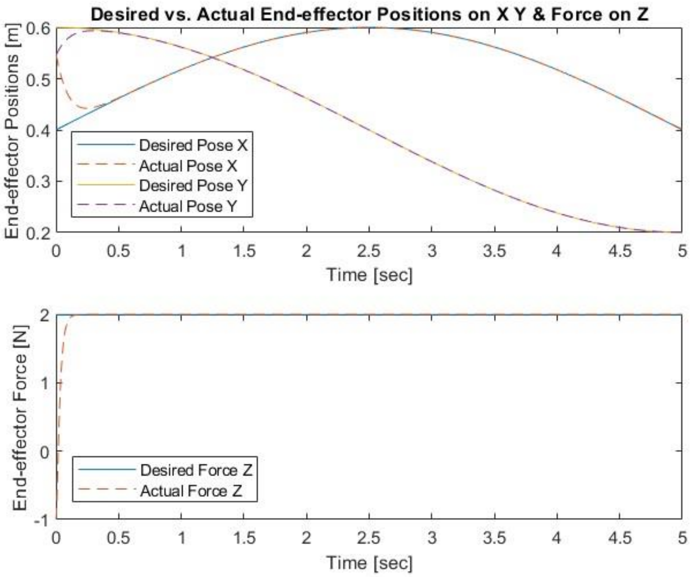

Fig.15. Desired versus actual end-effector positions on X and Y and force on Z for model-based hybrid position/force controller.

With these simulation results as shown in Fig.14 and Fig.15, this model-based control approach has proven effective in tasks requiring accurate motions and force management, such as manufacturing operations for a SCARA robotic arm.

5. Conclusion

This study presents a comprehensive analysis of different control strategies applied to a 4 DoF SCARA arm. Model-Based nonlinear controllers can provide precise control by predicting the behaviour of the robotic arm very well. In joint space, model-based controllers enabled the robotic arm to follow the trajectories of the joint positions very precisely. In Cartesian space, model-based controllers enabled the robotic arm to follow the end effector accurately along defined trajectories in the coordinate axis. In the Hybrid Position/Force controller, it has been shown that it can effectively manage both the position and the force at the end-effector of the robotic arm in Cartesian space, making it suitable for precise and accurate tasks. Simulation studies have shown that the 4 DoF SCARA robotic arm, one of the most used robots in industrial applications, can perform complex tasks precisely with model-based nonlinear controllers, and a significant contribution has been made to the high efficiency and adaptability of industrial operations.

Author contribution

H.T. and K.C. actively participated in conducting the simulation studies and writing the manuscript**.**

Acknowledgement

This research was carried out by the İzmir Katip Çelebi University Scientific Research Projects Coordination Unit with the project number 2022-GAP-MÜMF-0055.

References

[1] M. Javaid, A. Haleem, R. P. Singh, R. Suman, "Substantial capabilities of robotics in enhancing industry 4.0 implementation", *Cognitive Robotics*, vol. 1, pp. 58-75, 2021.
[2] R. Goel, P. Gupta, "Robotics and industry 4.0. A Roadmap to Industry 4.0: Smart Production", *Sharp Business and Sustainable Development*, vol. 1, pp. 157-169, 2020.
[3] G. Seeja, O. Reddy, K. V. R. Kumar, S. S. L. Mounika, "Internet of things and robotic applications in the industrial automation process", *In Innovations in the Industrial Internet of Things (IIoT) and Smart Factory*, pp. 50-64, 2021.
[4] T. Brogårdh, "Present and future robot control development — An industrial perspective", *Annual Reviews in Control*, vol. 31(1), pp. 69-79, 2007.
[5] R. Goel, P. Gupta, "Robotics and industry 4.0. A Roadmap to Industry 4.0: Smart Production", *Sharp Business and Sustainable Development*, vol. 1, pp. 157-169, 2020.
[6] G. Michalos, S. Makris, N. Papakostas, D. Mourtzis, G. Chryssolouris, "Automotive assembly technologies review: challenges and outlook for a flexible and adaptive approach", *CIRP Journal of Manufacturing Science and Technology*, vol. 2(2), pp. 81-91, 2010.
[7] B. S. K. Ibrahim, A. M. Zargoun, "Modelling and control of SCARA manipulator", *Procedia Computer Science*, vol. 42, pp. 106-113, 2014.
[8] M. A. González-Palacios, M. A. Garcia-Murillo, M. González-Dávila, "A novel tool to optimize the performance of SCARA robots used in pick and place operations", *Journal of Mechanical Science and Technology*, vol. 35, pp. 4715-4726, 2021.
[9] C. Gosselin, M. Isaksson, K. Marlow, T. Laliberté, "Workspace and sensitivity analysis of a novel nonredundant parallel SCARA robot featuring infinite tool rotation", *IEEE Robotics and Automation Letters*, vol. 1(2), pp. 776-783, 2016.
[10] L. Bruzzone, G. Bozzini, "A statically balanced SCARA-like industrial manipulator with high energetic efficiency", *Meccanica*, vol. 46, pp. 771-784, 2011.
[11] L. Bruzzone, S. E. Nodehi, G. Berselli, P. Fanghella, "Energy Efficiency of a SCARA-Like Manipulator with Elastic Balancing", *In International Workshop IFToMM for Sustainable Development Goals*, Cham: Springer Nature pp. 65-72, Switzerland, 2023.
[12] W.A.C. Lopes, et al. "Digital Twins (DT) Applied to the Customization of 3D Printed Scara Robots Using Intelligent Manufacturing." *IFIP International Conference on Advances in Production Management Systems*, Cham: Springer Nature Switzerland, 2024.
[13] R. Bouzid, J. Narayan, and H. Gritli, "Exploring artificial neural networks for the forward kinematics of a SCARA robotic manipulator using varied datasets and training optimizers", *Engineering Research Express*, vol. 4, 2024.
[14] W. Sirianusornsak, and C. Raksiri, "A Study of Scara Robot Vibration Behavior by Computer Aided Engineer Program", *The National Academic Conference*, Nakhon Pathom Rajabhat University, 2024.
[15] G. Boschetti, G. Rosati, and T. Sinico, "Retrofit and Interface of a SCARA Robot with a PLC for Real-Time Direct Joint Control", *European Conference on Mechanism Science*, Cham: Springer Nature Switzerland, 2024.
[16] P. V. S. Subhashini, et al. "Modelling, Simulation and Kinematic Analysis of a Modified SCARA Manipulator (RRRRP)." *International Conference on Cognitive Robotics and Intelligent Systems*, 2024.
[17] M. W. Spong, S. Hutchinson, M. Vidyasagar, "Robot modeling and control", John Wiley & Sons, 2020.
[18] P. I. Corke, "A simple and systematic approach to assigning Denavit–Hartenberg parameters", *IEEE transactions on robotics*, vol. 23(3), pp. 590-594, 2007.
[19] O. Altuzarra, O. Salgado, V. Petuya, A. Hernández, "Computational kinematics for robotic manipulators: Jacobian problems", *Engineering Computations*, vol. 25(1), pp. 4-27, 2008.
[20] P. K. Khosla, T. Kanade, "Parameter identification of robot dynamics", *IEEE conference on decision and control*, pp. 1754-1760, Fort Lauderdale, FL, USA, 1985.
[21] P. Tomei, "Adaptive PD controller for robot manipulators", *IEEE transactions on robotics and automation*, vol. 7(4), pp. 565-570, 1991.
[22] C. Brosilow, B. Joseph, "Techniques of model-based control", *Prentice Hall Professional*, 2002.
[23] W. J. Wilson, C. W. Hulls, G. S. Bell, "Relative end-effector control using cartesian position based visual servoing", *IEEE Transactions on Robotics and Automation*, vol. 12(5), pp. 684-696, 1996.
[24] M. H. Raibert, J. J. Craig, "Hybrid Position/Force Control of Manipulators", *ASME Journal of Dynamic Systems*, Measurements and Control, vol. 102, pp. 126-133, 1981.
[25] J. McDonald, "Hybrid Force-Position Control of a 4-DOF SCARA Manipulator", *Royal Millitary College of Canada*, Master Thesis, 2022.
[26] J. J. Craig, "Introduction to Robotics", Global Edition, *Pearson*, 2021.