Modelling and control of SCARA manipulator

Ibrahim, B. S. K. K. & Zargoun, A. M. A.

Modelling and Control of SCARA manipulator

B. S. K. K. Ibrahim* and Ahmed M. A. Zargoun

ᵃDepartment of Mechatronic and Robotic Engineering, Faculty of Electrical & Electronic Engineering, University Tun Hussien Onn Malaysia 86400, Batu Pahat, Malaysia

Abstract

Nowadays manipulators are getting more complex due to the development of the motor drives and the nonlinearities of the manipulator dynamics. Due to complexity of this system, the modeling process will become more complicated especially if modeled by using mathematical representation or white box approach. Therefore Computed Aided Design (CAD) modeling approach is more suitable to be applied in this system. In this paper presents the development of the CAD model of robot arm by using SolidWorks software. Then a controller based on Proportional-integral-derivative (PID) has been designed in simulation environment by using Matlab/Simulink platform. This paper shows the advantages of the combination of MATLAB and SolidWorks. SolidWorks is able to ease the modeling process. The performance of PID controller for 4 Degree of Freedom (DOF) of SCARA (Selective Compliance Articulated Robot Arm) manipulator has been assessed for first 2 DOF and shown good results.

1. Introduction

The SCARA (Selective Compliance Articulated Robot Arm) robot is used in many applications such as pick and place, assembly, and packaging. The SCARA robot is a nonlinear dynamic system which possesses some uncertainties as friction. Since the SCARA is highly nonlinear it is difficult to identify parameters of the SCARA model using traditional methods. It is very difficult to calculate for the real SCARA robot inertia moments, masses, etc.

This robot manipulator needs to follow the trajectory for the any application. Therefore, the trajectory tracking of robot manipulators needs a correct dynamic model and accurate factors of the robot. However, every dynamic model has some degree of error. The incorrectness and errors eventually result placement and/or trajectory tracking errors. There are many research activities have been made on how to simplify the robot manipulator dynamics modeling. However, all these researchers have been used a complex method to overcome the traditional dynamics modeling errors [1-4].

Mathworks was introduced SimMechanics Link which is a third party Simulink Software product. The SimMechanics Link utility generates Physical Modeling XML files from external applications such as computer-aided design (CAD) platforms. These Physical Modeling XML files can be used with SimMechanics software to generate SimMechanics models representing mechanical systems.

There are a very few research about the benefits of the combination between SolidWorks and Matlab, for example [5] who has been used it to model and simulate a multi-fingered robot hand in order to avoid/simplify the complexity of the robot hand dynamics. In [6] the simulations a CAD model that mimics the real robotic arm was created into SolidWorks

* Corresponding author. Tel.: +607-4537355; fax: +607-4536060.
Email addresses: babul@uthm.edu.my

environment, followed by the conversion of the CAD parts into Simulink SimMechanics model to describe the inverse kinematics solutions of an anthropomorphic redundant robotic with seven degrees of freedom and human like workspace. In this paper SimMechanics was used as the main simulation environment which could import the manipulator from the XML file created by SolidWorks. The XML file contains all the dynamic parameters of the robot manipulator.

2. SCARA manipulator design

2.1. Kinematics description

The kinematics was described using Denavit-Hartenberg algorithm, the coordinate transformation from the link i frame to the link i-1 frame will now be derived with the aid of Figure 1.

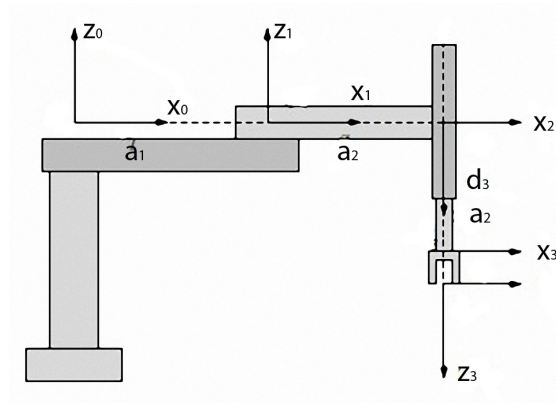Fig. 1.   SCARA robot arm parameters.

Assign z_0, z_1…, z_{n-1} axes along the motion axes of joint 1, joint 2…, joint n, respectively. Therefore, the z_i axis coincides with the motion axis of the joint i+1[1]. In this paper the first two revolute joints have been used to identify the performance of the PID controller, because they have a non-linear dynamics.

Table 1. The SCARA robot manipulator parameters

i	θ_i	d_i	a_i	α_i
1	θ_1	0	a_1	0
2	θ_2	0	a_2	180^0
3	0	d_3	0	0
4	θ_4	0	0	0

The forward kinematics parameter is as in table I. then the inverse kinematics can be determined by projecting manipulator onto the x_0 - y_0 plane as shown in figure 2, so θ_2 is determined as in Eq. 1.
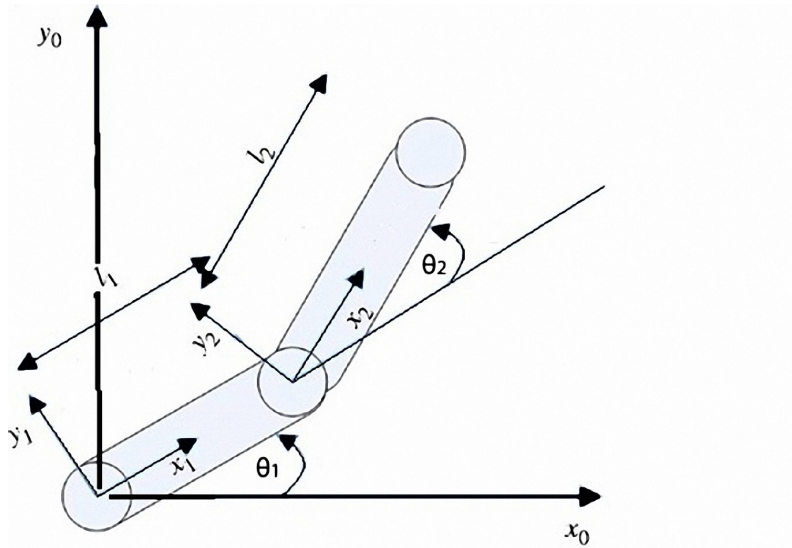
Fig. 2. x_0 - y_0 projection of the robot arm

$$\theta_2 = A\tan 2\left(c_2, \pm\sqrt{1-c_2}\right) \tag{1}$$

Where

$$c_2 = \frac{O_x^2 + O_y^2 - l_1^2 - l_2^2}{2l_1l_2} \tag{2}$$

Since θ_2 been determine therefore θ_1 as shown in Eq.2.

$$\theta_1 = A\tan 2(O_x, O_y) - A\tan 2\left(l_1 + l_2\cos\theta_2, l_2\sin\theta_2\right) \tag{3}$$

2.2. CAD design of SCARA robot

The dimension of each part was defined and designed such that all parts can be matched together, since the simulation needs only the first two joints, therefore the other joints been wielded. All of SCARA robot arm parts were named base, link1, link2 and link3.

After part modeling, the individual components were assembled into a single assembly file. Friction applied on the joints of the arm to make the dynamics of the arm nonlinear.

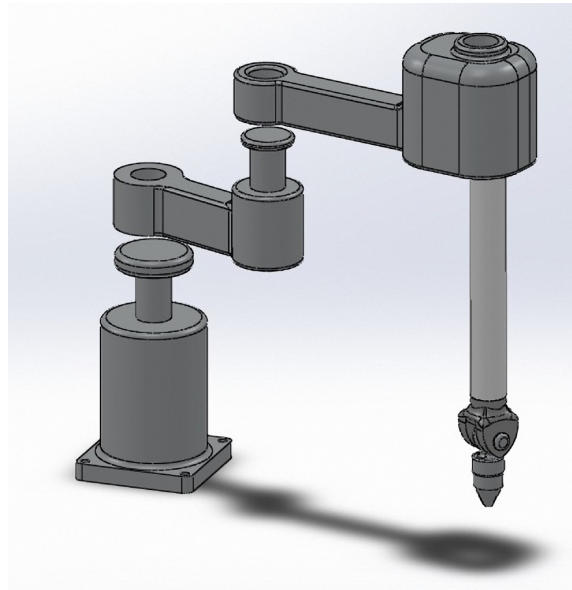Figure 3 shows the components that were used to assemble the SCARA arm.

Fig. 3.　Component of SCARA manipulator design

2.3. Controller design

MATLAB Simulink has been used to design the PID controller of the SCARA manipulator. Each joint is controlled by individual PID controller. The overall of the system is shown in Figure 4. The gain values of P, I and D have been set by using heuristic method.
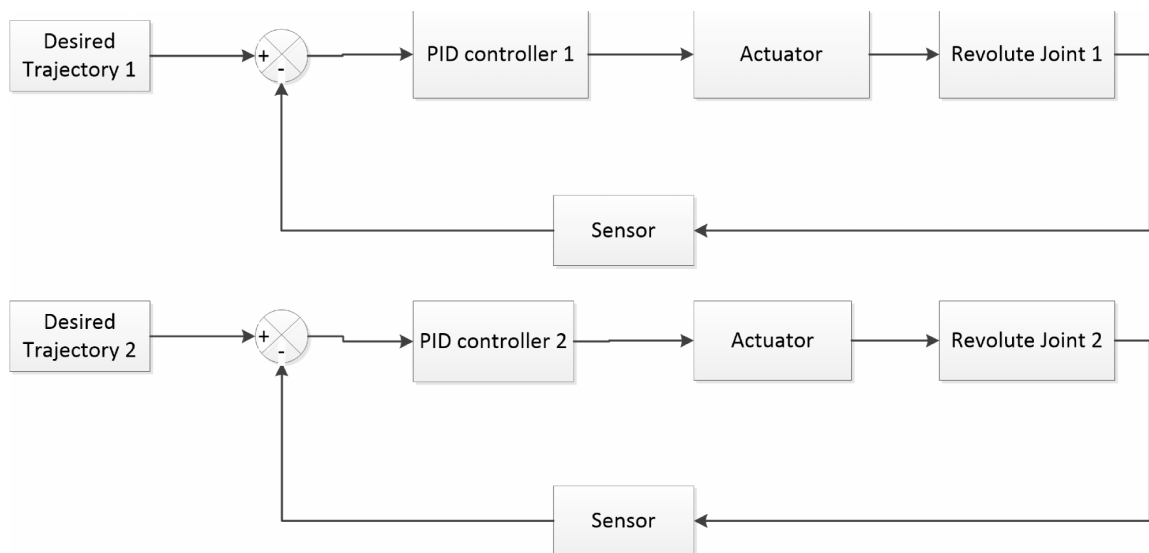

Fig. 4.　Overall block diagram

Figure 5 shows the PID controller block which applied to both joints with different values of gain due to the difference of each link dynamics.
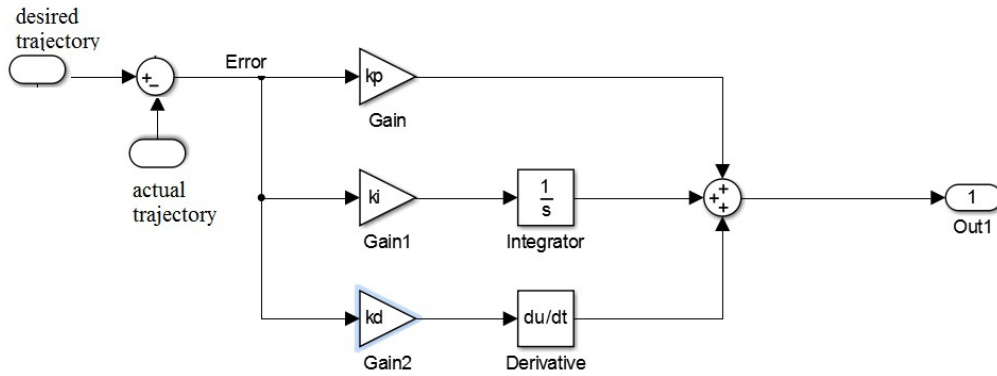
Fig. 5.   PID controller of the joint

2.4. Equations and formulae Trajectory planning

In most of the manipulators, it is desirable for a robot manipulator to move its trajectory at a constant velocity. Since the initial velocity and the final velocity are zero, the joint movement of the link will accelerate at the beginning, in order to reach the blended velocity, then the movement of the link will decelerate so that it can stop at the goal position exactly. In such plan the trajectory should be divided into three segments.

First segment is the acceleration segment ($t < t_b$) the equations of position θ, velocity and acceleration are as in equation Eq.4, Eq.5 and Eq.6.

$$\theta(t) = \theta_0 + \frac{a_1 t^2}{2} \tag{4}$$

$$\dot{\theta}(t) = a_1 t \tag{5}$$

$$\ddot{\theta}(t) = a_1 \tag{6}$$

Second segment is the constant velocity segment ($t < (t_f - t_b)$) also called as blended segment, the equations of position θ, velocity and acceleration are as in equation Eq.7, Eq.8 and Eq.9.

$$\theta(t) = \theta_1 + v(t - t_b) \tag{7}$$

$$\dot{\theta}(t) = v \tag{8}$$

$$\ddot{\theta}(t) = 0 \tag{9}$$

where

$$\theta_1 = \theta_0 + \frac{a_1 t_b^2}{2} \tag{10}$$

$$v = a_1 t_b \tag{11}$$

Third segment is deceleration segment ($t \le t_f$) the equations of position θ, velocity and acceleration are as in equation Eq.10, Eq.11 and Eq.12.

$$\theta(t) = \theta_2 + v\left(t - \left(t_f - t_b\right)\right) - \frac{a_1\left(t - \left(t_f - t_b\right)^2\right)}{2} \tag{12}$$

$$\dot{\theta}(t) = v - a_1\left(t - \left(t_f - t_b\right)\right) \tag{13}$$

$$\ddot{\theta}(t) = -a_1 \tag{14}$$

where

$$\theta_2 = \theta_1 + v\left(t_f - 2t_b\right) \tag{15}$$

Figure 6 shows the overall procedure developed in order to satisfy the objectives of this project.
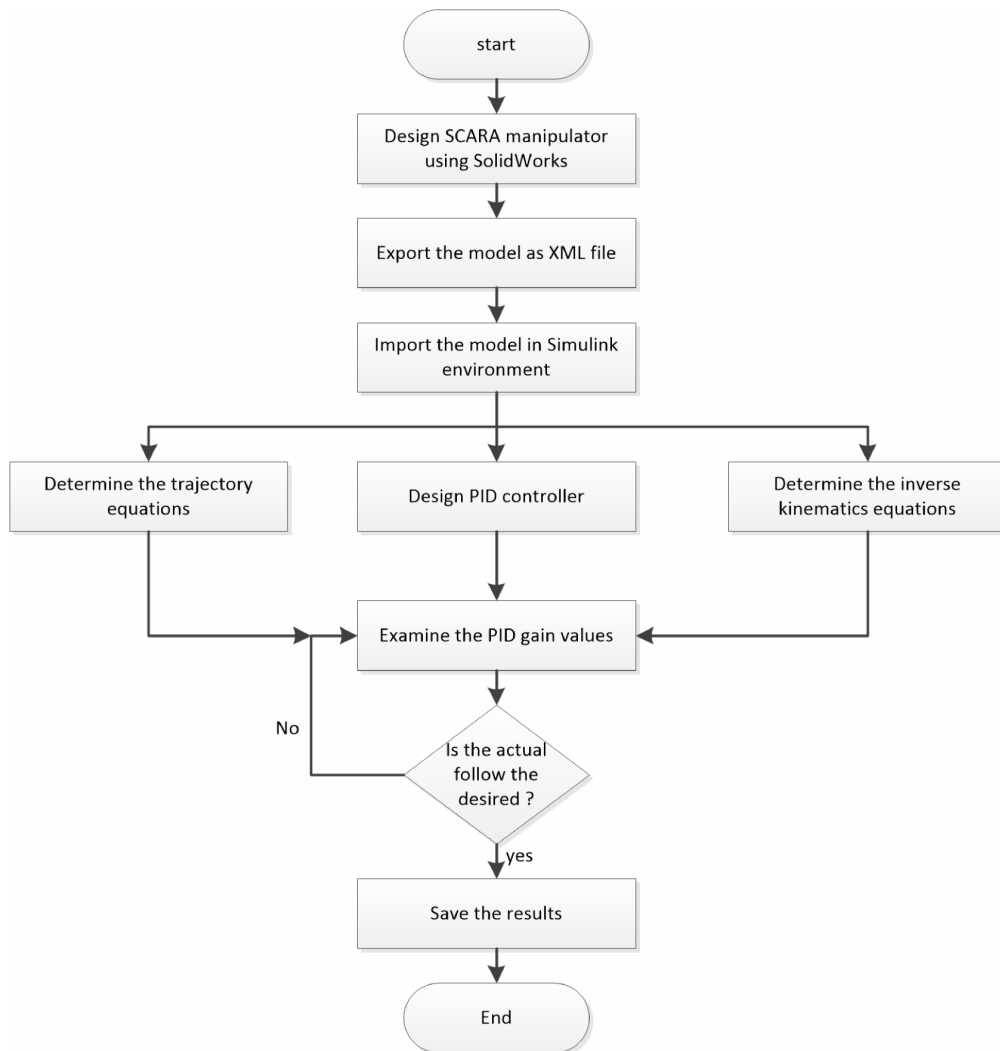

Fig. 6.   The Overall procedure

3. Result

Based on the inverse kinematics equations the final position of the joint 1 is 43.54° and joint 2 position was 115°. The best gain values were determined as follows $K_P=1$, $K_I=7$ and $K_D=16$ for joint 1 and $K_P=1$, $K_I=6$ and $K_D=12$. The simulation
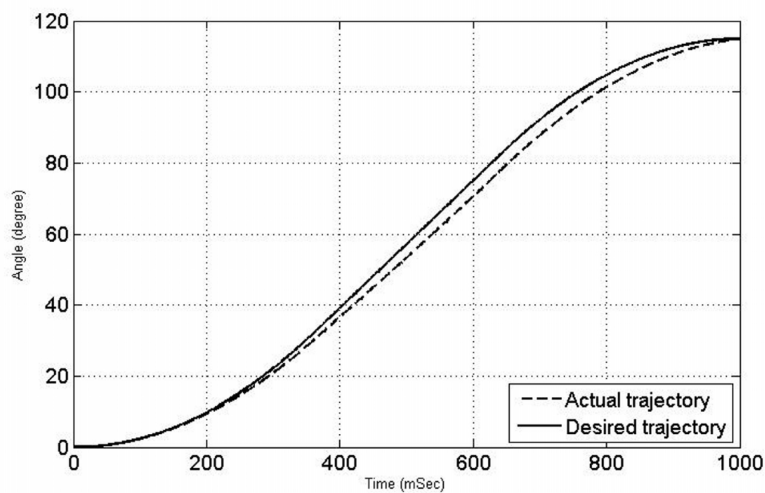
results for joint 1 and joint 2 are shown in Figure 7(a) and 7(b) respectively. It can be observed that the actual outputs for both joints are able to track the desired trajectories.

Table 2. The Gain values of joint1 and joint2

	Joint 1	Joint 2
K_P Value	1	1
K_I Value	7	6
K_D Value	16	12
Desired position	43.54°	115°
Actual position	43.27°	114.9°
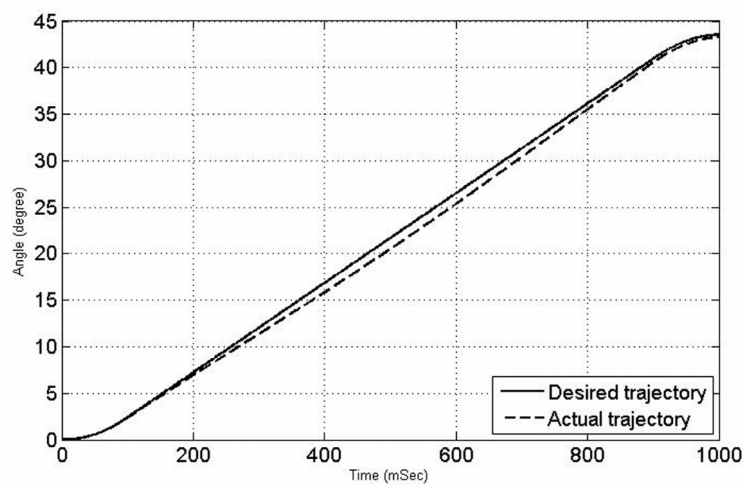
(a)

(b)

Fig. 7.   PID controllers response (a) joint 1 (b) joint 2

4. Conclusion

Modelling of robot manipulator is significantly challenging for researchers due to the complexity of the system. The objective of this study is to develop a model by using CAD modeling approach by using SolidWorks for the computer simulation study purpose. This modelling approach is comparatively less burdened with complex mathematical model. The CAD model of robot arm then imported to Matlab/Simulink environment. Then the well known conventional controller; PID controllers have been used to control the movement of these robot joints so that follow the desired trajectory. Simulation of the SCARA manipulator shows the tracking performance having less than 1% error which is acceptable pick and place applications. This accuracy can be improved so that can be applied for medical application such as surgeon robot.The study has identified some of the potential benefits of using Matlab/Simulink and SolidWorks. CAD is simpler to model a complex system as it eliminates the complicated mathematical modelling process.

References

1. Kikuuwe, R. Semiglobally strictly passive two-port network representation of rigid link manipulator under PID trajectory-tracking control. in Decision and Control (CDC), 2010 49th IEEE Conference on. 2010.
2. Muro-Maldonado, D., A. Rodriguez-Angeles, and C.A. Cruz-Villar. Sliding PID Control for Trajectory Tracking of a 2 DOE Robot Manipulator: Simulations and Experiments. in Electrical and Electronics Engineering, 2007. ICEEE 2007. 4th International Conference on. 2007.
3. Yiu, Y.K. and Z.X. Li. PID and adaptive robust control of a 2-DOF over-actuated parallel manipulator for tracking different trajectory. in Computational Intelligence in Robotics and Automation, 2003. Proceedings. 2003 IEEE International Symposium on. 2003.
4. Zhihong, M., H.R. Wu, and M. Palaniswami, An adaptive tracking controller using neural networks for a class of nonlinear systems. IEEE Trans Neural Netw, 1998. 9(5): p. 947-55.
5. Tarmizi, W.F.B.W., et al. Modeling and simulation of a multi-fingered robot hand. in Intelligent and Advanced Systems (ICIAS), 2010 International Conference on. 2010.
6. Crenganis, M., et al. Inverse kinematics of a 7 DOF manipulator using Adaptive Neuro-Fuzzy Inference Systems. in Control Automation Robotics & Vision (ICARCV), 2012 12th International Conference on. 2012.