IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁙ Peer-reviewed & Refereed journal ⁙ Vol. 15, Issue 7, July 2026

DOI: 10.17148/IJARCCE.2026.15740

Forward Kinematics of PUMA 560 Robot using DH Method

Gurjeet Singh

Director, School of Computing & Engineering, Rayat Bahra Professional university, Hoshiarpur, Punjab, India

Abstract. In this paper, solve the forward and inverse kinematics solution of Programmable Universal Manipulation Arm (PUMA) 560 Robot. PUMA 560 robot is the most popular robot in Research and Industry. It is 6 DOF robots having three revolute joints locate the end effector, and three joint gives the orientation to the end effector. One of the main concerns among robotic and control engineers is positioning the manipulator's End-effector to the most accurate place and transparent the effect of disturbance and errors which will effect on manipulator's final result. As a matter of fact, controlling manipulators are hard and expensive because they are multi-input, multi-output, time variant and non-linear, so it has been a topic for researchers to design the most efficient controller to help the manipulator to achieve the desired expectation under any circumstance.

INTRODUCTION

Robots have become an integral part of everyday life. Over the few preceding decades, the adoption of the industrial robot can be spotted worldwide and has substantially increased with a swift increasing trend [1]. Principally, these are being utilized for material handling, hazardous materials, undersea operations, etc. [4]. Robot manipulator ensnares an electromechanical device that necessitates human dexterity to carry out a variety of tasks [5]. Due to multiple applications, robots play an indispensable role in industrial productions. Robots are programmed with the help of different software and are used to perform complex operations that are not performed by the man. Six Degree of freedom (DOF) robots are much popular in the Industry and Academic and the robots which having more DOF have better performance and increase agility and dynamic performance. Programmable Universal Machine for Assemble (PUMA) 560 robot manipulator is a industrial robotic arm with 6 revolute joint and 6 links. The mathematical model of the forward kinematics of PUMA 560 robot manipulator has 6 degree of freedom and involves the rotation and displacement of its coordinate frames. The forward kinematic analysis of PUMA 560 robot manipulator becomes easier when we use Denavit–Hartenberg (DH) model and general homogeneous transformation matrix[6].

Denavit–Hartenberg (DH) model is a kinematic model that relates the rotation and displacement of a reference frame B0 to the last link or end-effector frame Bn. The 4 × 4 homogeneous transformation matrix holds the rotation and the displacement matrix. The 3 × 3 rotation matrix defines the orientation of a joint or the end-effector while the 3 × 1 displacement matrix has the position of a joint or end-effector position[11].

METHODS

Denavit–Hartenberg (DH) Model

The Denavit–Hartenberg (DH) model of the PUMA 560 robot manipulator. We start by setting the axes of its frames. The z-axis at each frame must be the rotation axis of the frame while the x-axis must be perpendicular to the z-axis of the frame Bn−1 and Bn. Also, the x-axis at frame Bn intersects with the z-axis at frame Bn−1. All axes at each frame must adhere to right-hand rule.

The DH parameters relates the rotation and displacement of frames Bn to Bn−1. The parameters are the joint angles θ, twist angle α, offset r, and link lengths d. The joint angle θ is the rotation at zn−1 to match the xn−1 to xn. The rotation at zn−1 includes all angular displacement at zn−1-axis. The joint angle at frame 1 is simply the angular displacement, θ1. This is true to all of the frames. In contrast, the twist angle α is the rotation at the xn to match the orientation of zn−1 to zn. For example, the twist angle α1 is −90◦ and it means that x1 rotates at −90◦ to match it with z0 to z1.

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 7, July 2026
DOI: 10.17148/IJARCCE.2026.15740

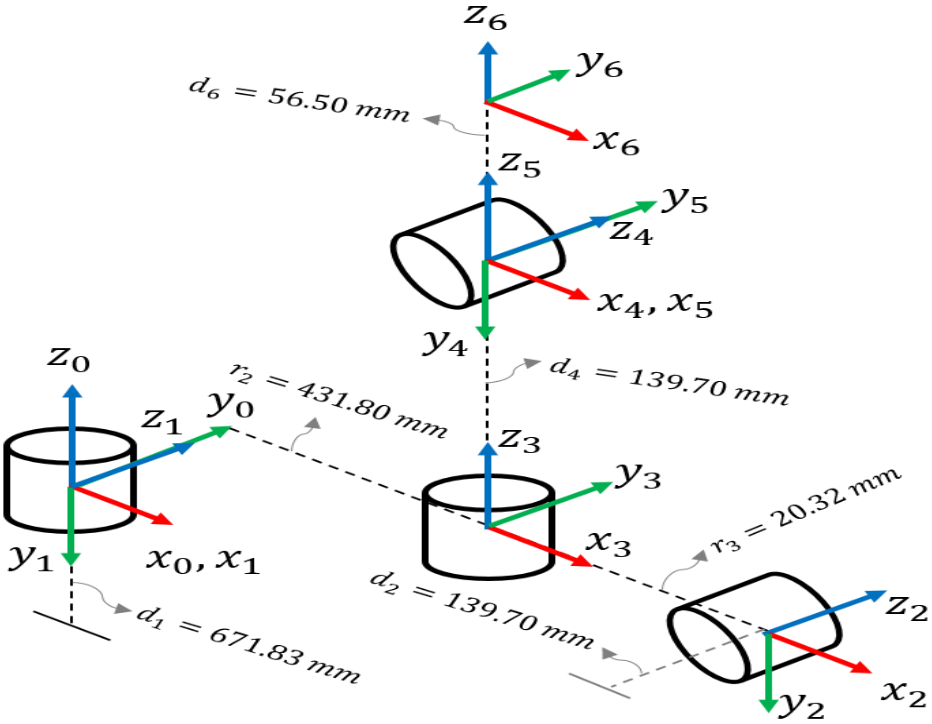

FIGURE 1. Joint representation of PUMA 560 Robot[4]

*Joint representation of PUMA 560 Robot

The offset r is the distance between the centers of n − 1 and n joints to the direction of xn. At frame 2, the offset r2 is equal to 431.80 mm. It denotes the distance at which the n-joint are displaced at xn as to n−1 joint. The link lengths d is the distance between the centers of n−1 and n joints to the direction of zn−1. In the figure, the link lengths d2 is equal to 139.70 mm hence it is the distance between n − 1 and n joint along z1.

Frame	θ	α	r	d
1	θ_1	-90°	0	671.83 mm
2	θ_2	0°	431.80 mm	139.70 mm
3	θ_3	90°	-20.32 mm	0
4	θ_4	-90°	0	431.80 mm
5	θ_5	90°	0	0
6	θ_6	0°	0	56.50 mm

Table 1: DH Parameters of PUMA 560 robot manipulator

The Denavit–Hartenberg (DH) model

$$T_n^0 = T_1^0 \cdot T_2^1 \cdots T_n^{n-1} \quad (1)$$

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 7, July 2026

DOI: 10.17148/IJARCCE.2026.15740

is a standard kinematic model expressing position and orientation of a fixed coordinated frame relative to the nth−link or end-effector. The general homogeneous matrix Tn−1n consist the rotation and displacement vectors for a given joint angles. We define the homogeneous forward transformation matrix at each frame as

$$T_n^{n-1} = \begin{bmatrix} \cos\theta_n & -\sin\theta_n \cdot \cos\alpha_n & \sin\theta_n \cdot \sin\alpha_n & r_n \cos\theta_n \\ \sin\theta_n & \cos\theta_n \cdot \cos\alpha_n & -\cos\theta_n \cdot \sin\alpha_n & r_n \sin\theta_n \\ 0 & \sin\alpha_n & \cos\alpha_n & d_n \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2)$$

where the rotation matrix is

$$R_n^{n-1} = \begin{bmatrix} \cos\theta_n & -\sin\theta_n \cdot \cos\alpha_n & \sin\theta_n \cdot \sin\alpha_n \\ \sin\theta_n & \cos\theta_n \cdot \cos\alpha_n & -\cos\theta_n \cdot \sin\alpha_n \\ 0 & \sin\alpha_n & \cos\alpha_n \end{bmatrix} \quad (3)$$

and displacement matrix is

$$d_n^{n-1} = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = \begin{bmatrix} r_n \cos\theta_n \\ r_n \sin\theta_n \\ d_n \end{bmatrix}. \quad (4)$$

Thus, the forward kinematic of PUMA 560 robot manipulator is given by the dot product of the forward transformation matrices at each frame. This is clearly establish in DH model. The position of the end-effector is in the displacement matrix d0n while its orientation is in the rotation matrix r0n.

2.2 MATLAB implementation:

A script that implements the numerical solution for the forward kinematics problem of a PUMA 560 robot manipulator. We start with initializing the DH parameters listed in Table 1.

Listing 1: Initialize DH parameters

```
function fPUMA(theta1, theta2, theta3, theta4, theta5, theta6)
  % DH parameters
  J = [0,0,0,0,0,0]; % Joint angles
  A = [-90, 0, 90, -90, 90, 0]; % Twist angle
  r = [0, 431.80, -20.32, 0, 0, 0]; % Offset as to xn
  d = [671.83, 139.70, 0, 431.80, 0, 56.50]; % Offset as to z(n-1)

  % Forward kinematics
  if all(J >= -160 & J <= 160) && all(J(2:6) >= -225 & J(2:6) <= 45) && all(J(1:3) >= -45 & J(1:3) <= 225) ...
      && all(J(1:4) >= -110 & J(1:4) <= 170) && all(J(1:5) >= -100 & J(1:5) <= 100) && all(J >= -266 & J <=
266)

    T = cell(1, 6); % Preallocate cell array for transformation matrices
    P = cell(1, 6); % Preallocate cell array for joint positions

    % Homogeneous Transformation
    for n = 1:6
      matT = [cosd(J(n)), -sind(J(n))*cosd(A(n)), sind(J(n))*sind(A(n)), r(n)*cosd(J(n));
          sind(J(n)), cosd(J(n))*cosd(A(n)), -cosd(J(n))*sind(A(n)), r(n)*sind(J(n));
          0, sind(A(n)), cosd(A(n)), d(n);
          0, 0, 0, 1];
      T{n} = matT;
    end
```

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁂ Peer-reviewed & Refereed journal ⁂ Vol. 15, Issue 7, July 2026
DOI: 10.17148/IJARCCE.2026.15740

```
        % Joint Positions
        for i = 1:6
            if i == 1
                P{i} = T{i};
            else
                P{i} = P{i - 1} * T{i};
            end
        end

        % Plotting the joint positions
        x = [0, P{1}(1, 4), P{1}(1, 4), P{2}(1, 4), P{3}(1, 4), P{4}(1, 4), P{5}(1, 4), P{6}(1, 4)];
        y = [0, P{1}(2, 4), P{2}(2, 4), P{2}(2, 4), P{3}(2, 4), P{4}(2, 4), P{5}(2, 4), P{6}(2, 4)];
        z = [0, P{1}(3, 4), P{1}(3, 4), P{2}(3, 4), P{3}(3, 4), P{4}(3, 4), P{5}(3, 4), P{6}(3, 4)];

        figure; hold on; grid on;
        plot3(x, y, z, 'k-o', 'LineWidth', 2);
        xlabel('x-axis');
        ylabel('y-axis');
        zlabel('z-axis');
        title('Forward Kinematics of PUMA 560 Manipulator (6DoF)');
        view(3);

        % Display end-effector position
        fprintf('The end-effector position is at Px = %.2f, Py = %.2f, and Pz = %.2f.\n', P{6}(1, 4), P{6}(2, 4), P{6}(3,
4));
        disp('The DH model is:');
        disp(P{6});
    else
        disp('Joint angle out of range');
    end
end
```

We create a PUMA function comprising all the numerical solution and iterations that leads to the forward kinematics of a 6R PUMA 560 robot manipulator. We input to the function a set of joint angles and it results in the end-effector point and a three-dimensional plot of the robot manipulator links and rotation. After which, we evaluate the derived solution by substituting the initial, boundary, and random conditions. This is to check if the analytical and computational solution yields to a correct end-effector and plot from the input joint angles. The full script for the fPUMA function is shown in listing 4.

Listing 4: PUMA function

```
function fPUMA(theta1,theta2,theta3,theta4,theta5,theta6)
%input: joint angles at each frames
%ouput: end-effector position
%author: Jayson c. Jueco
format compact
format short
%DH parameters
J = [theta1 theta2 theta3 theta4 theta5 theta6]; %joint angles
A =  [-90 0 90 -90 90 0] ; %twist angle
r = [0 431.80 -20.32 0 0 0]; %offset as to xn
d = [671.83 139.70 0 431.80 0 56.50]; %offset as to z(n-1)
%forward kinematics
if J(1,1) >= -160 && J(1,1) <= 160 && J(1,2)>= -225 ...
    && J(1,2) <= 45 && J(1,3) >= -45 && J(1,3) <= 225 ...
```

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⪻ Peer-reviewed & Refereed journal ⪻ Vol. 15, Issue 7, July 2026

DOI: 10.17148/IJARCCE.2026.15740

```
    && J(1,4) >= -110 && J(1,4) <= 170 && J(1,5) >= -100 ...
    && J(1,5) <= 100 && J(1,6) >= -266 && J(1,6) <= 266
T = [];
%Homogeneus Transformation
for n = 1:6
    matT = [cosd(J(n)) -sind(J(n))*cosd(A(n)) ...
        sind(J(n))*sind(A(n)) r(n)*cosd(J(n));
        sind(J(n)) cosd(J(n))*cosd(A(n)) ...
        -cosd(J(n))*sind(A(n)) r(n)*sind(J(n));
        0 sind(A(n)) cosd(A(n)) d(n);
        0 0 0 1];
    T = [T; {matT}];
end
P = [];
%Joint Positions
for i = 1:6
    if i == 1
        P = [P,{T{i}}];
    else
        matP = P{i-1}*T{i};
        P = [P, {matP}];
    end
end
%plotting the joint positions
x = [0 P{1}(1,4) P{1}(1,4) P{2}(1,4) ...
    P{3}(1,4) P{4}(1,4) P{5}(1,4) P{6}(1,4)];
y = [0 P{1}(2,4) P{2}(2,4) P{2}(2,4) ...
    P{3}(2,4) P{4}(2,4) P{5}(2,4) P{6}(2,4)];
z = [0 P{1}(3,4) P{1}(3,4) P{2}(3,4) ...
    P{3}(3,4) P{4}(3,4) P{5}(3,4) P{6}(3,4)];
hold on
grid on
rotate3d on
plot3([x(1) x(2)],[y(1) y(2)],[z(1) z(2)]...
    ,'Color','b','LineWidth',5)
plot3([x(2) x(3)],[y(2) y(3)],[z(2) z(3)]...
    ,'Color','r','LineWidth',5)
plot3([x(3) x(4)],[y(3) y(4)],[z(3) z(4)]...
    ,'Color','g','LineWidth',5)
plot3([x(4) x(5)],[y(4) y(5)],[z(4) z(5)]...
    ,'Color','c','LineWidth',5)
plot3([x(5) x(6)],[y(5) y(6)],[z(5) z(6)]...
    ,'Color','m','LineWidth',5)
plot3([x(6) x(7)],[y(6) y(7)],[z(6) z(7)]...
    ,'Color','y','LineWidth',5)
plot3([x(7) x(8)],[y(7) y(8)],[z(7) z(8)]...
    ,'Color','k','LineWidth',5)
plot3(x,y,z,'o','Color','k')
xlabel('x-axis')
ylabel('y-axis')
zlabel('z-axis')
title('Forward Kinematics of PUMA 560 Manipulator (6DoF)')
view(0,0)
disp('The end-effector position is ')
fprintf('at Px = %f, Py = %f, and Pz = %f'...
    ,P{6}(1,4),P{6}(2,4),P{6}(3,4));
disp('  ')
disp('The DH model is  ')
```

ISSN (O) 2278-1021, ISSN (P) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁙ Peer-reviewed & Refereed journal ⁙ Vol. 15, Issue 7, July 2026
DOI: 10.17148/IJARCCE.2026.15740

```
    disp(P{6})
  else
    disp('Joint angle out of range');
  end
  end
```

Result

In this section, we present the result of the MATLAB implementation and validate the forward kinematics in section 2.1. Initially, we set the joint angles to zero and verify the end-effector is at the initial position.

At initial condition, the end-effector point is located at Px = 411.480000, Py = 139.700000, and Pz = 1160.130000. the plot indicating the orientation and position of each frames of the robot manipulator. The orientation of the forearm of robot is the same to the basis of the DH model.

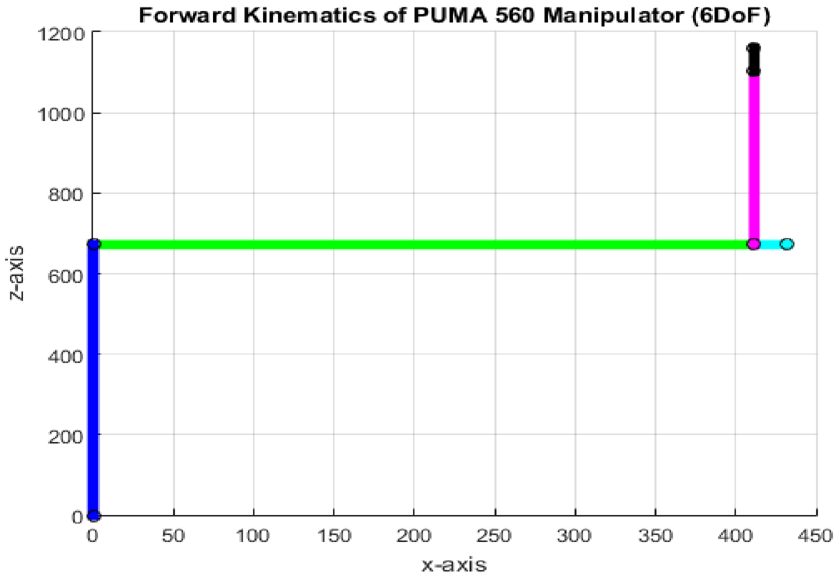

FIGURE 2. (a): 2D View (x to z plane)[8].

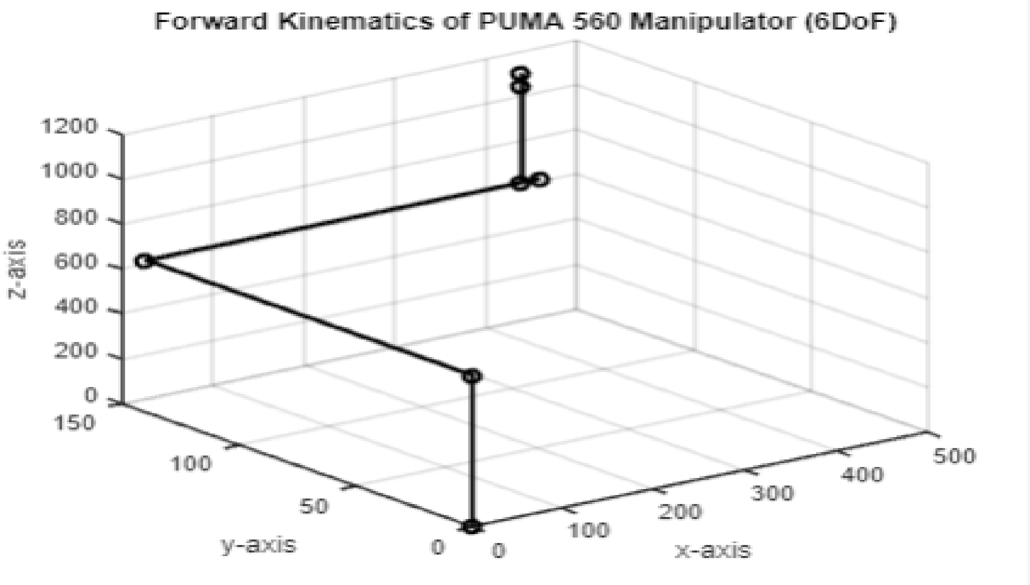

FIGURE 2. (b): 3D view

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⪤ Peer-reviewed & Refereed journal ⪤ Vol. 15, Issue 7, July 2026

DOI: 10.17148/IJARCCE.2026.15740

PUMA resulting plot for initial condition

Next, we checked the rotation of the elbow and the shoulder joints by setting its respective angles to 45◦. These joints must have the same plane rotation to have a correct response. When we set joint angle θ2 to 45◦, we observed that it rotates along the x-z plane, as shown in Figure 4a. The end-effector is located at Px = 636.240540, Py = 139.700000, and Pz = 726.149943 for joint angle θ2 at 45◦ while the other joints at zero. For joint angle θ3, we set it at 45◦ while all other joint to zero. This result to an end-effector point located at Px= 636.240540, Py = 139.700000, and Pz = 726.149943. we observed that the elbow joint rotate in the x-z plane[15].

As stated earlier, the shoulder and elbow joint have the same rotation plane to yield a correct end-effector. Figures 4a and 4b indicate that the shoulder and elbow joint rotate in x-z plane. The results verifies the correctness of the kinematic equation in section 2. Furthermore, when we set the shoulder and elbow joint to 45◦ and other joint angles to zero, we observed the same plane rotation.

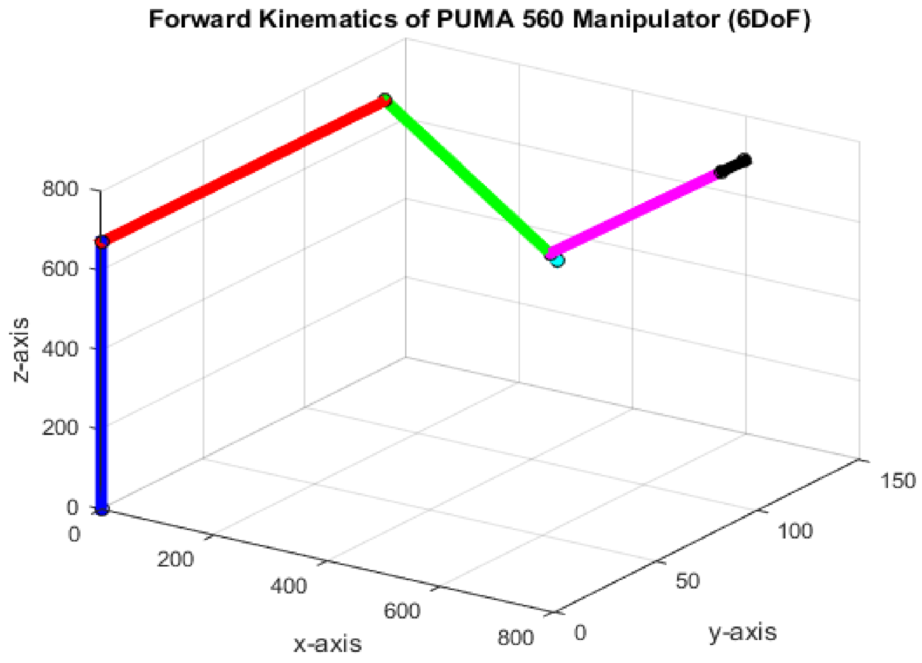

FIGURE 3. (a): Joint angle θ2 set to 45◦ (Shoulder)

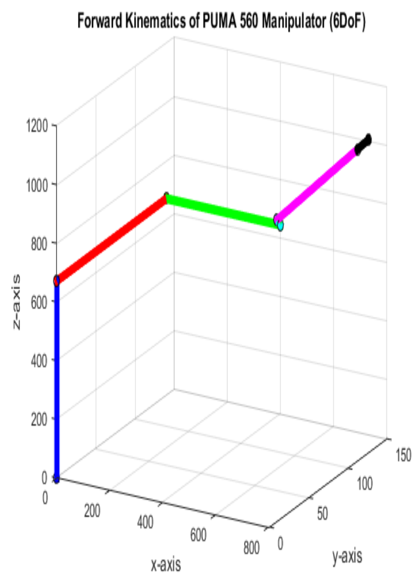

FIGURE 3. (b): Joint angle θ3 set to 45◦ (Elbow)

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 7, July 2026

DOI: 10.17148/IJARCCE.2026.15740

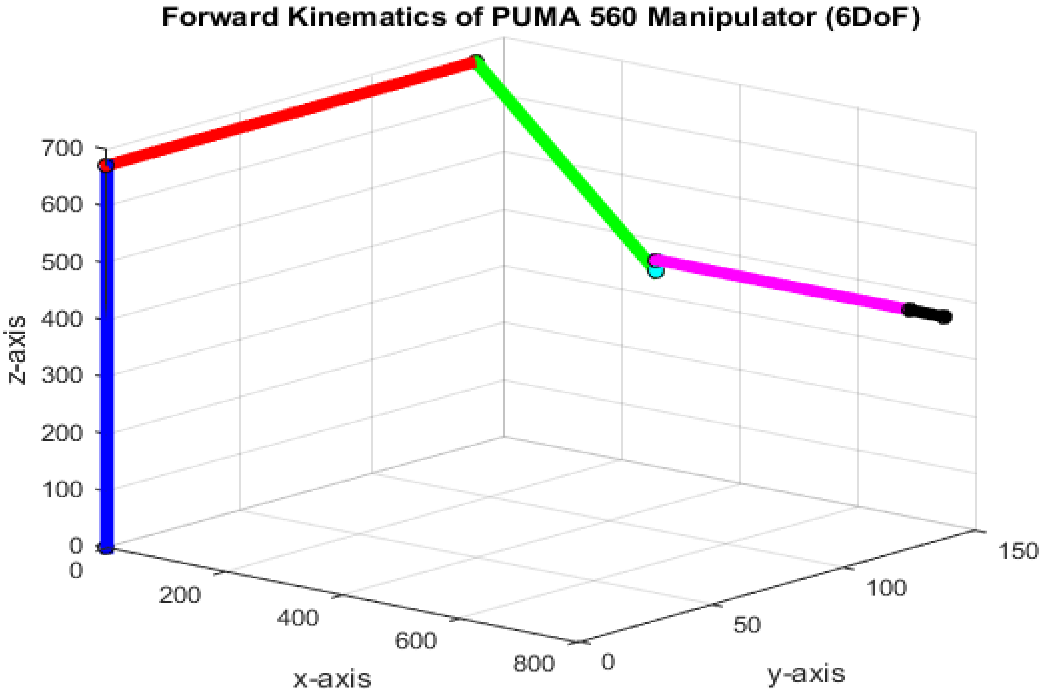

FIGURE 3. (c): Joint angle θ2 and θ3 set to 45◦

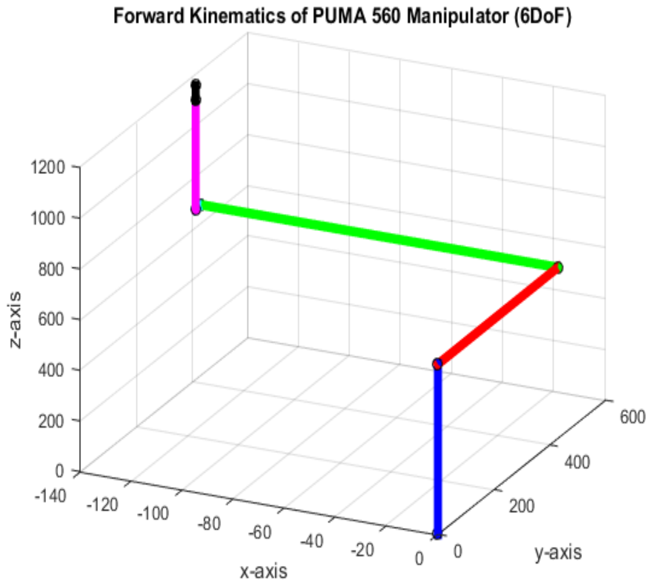

FIGURE 3. (d): Joint angle θ1 set to 90◦ (waist)

Then, we check the rotation of the waist joint. We set the joint angle (waist) θ1 to 90◦ and the other joint to zero. We observed that the waist rotate correctly, in reference to the initial condition. For a given set of joint rotation, we correctly locate the end-effector which was clear all throughout the plots.

APPLICATION

The PUMA 560 robot, developed by Unimation in the 1980s, was one of the first commercially successful robotic arms. Despite being an older model, it still finds applications in various industries and research fields. Here are some common applications of the PUMA 560 robot:

DOI: 10.17148/IJARCCE.2026.15740

1. Manufacturing: The PUMA 560 is commonly used in manufacturing processes for tasks such as welding, painting, assembly, and material handling. Its articulated arm design allows for precise and repeatable movements, making it ideal for tasks that require high accuracy.

2. Automotive Industry: In the automotive industry, PUMA 560 robots are used for tasks like spot welding, handling parts on assembly lines, and painting car bodies. They can work alongside human workers in collaborative manufacturing environments.

3. Material Handling: PUMA 560 robots are used for material handling tasks in warehouses and distribution centers. They can pick and place items with precision and efficiency, speeding up the logistics process.

4. Medical Applications: While newer robotic systems are more common in modern medical applications, the PUMA 560 has been used in research and development projects in the medical field. For example, it has been used for surgical training simulations and experimental procedures.

5. Research and Education: The PUMA 560 is still widely used in research and educational institutions for studying robotics, automation, and control systems. It serves as a platform for developing and testing algorithms for motion planning, path optimization, and robotic control.

6. Space Exploration: The PUMA 560 has also been used in space applications. Its precise movements make it suitable for tasks like manipulating objects or instruments in space environments. It has been used in various space-related research and development projects.

Overall, while newer and more advanced robotic systems have been developed since the introduction of the PUMA 560, it still has practical applications in a variety of industries and research fields due to its reliability, precision, and versatility.

CONCLUSION

The forward kinematics of PUMA 560 robot manipulator by using DH model. We established the DH parameters from the axes and joint representation for PUMA 560 robot manipulator. We formed the homogeneous transformation matrices for each frames using the equations. The forward kinematic equations was implemented in MATLAB and evaluate the rotation across the joints. In section 3, we observed that the shoulder joint (joint1) and elbow joint (joint 2) rotate in the same plane. The result in section 3 verifies that the joint has correct rotation and response. Aside from checking the joint rotation, the end-effector is also verified. At the initial condition (all joint angles θ equal to zero), the end-effector is at the correct location as observed from the basis configuration. Thus, the kinematic equation taken from the DH model correctly gives the end-effector position and orientation.

REFERENCES

[1] Frank Hoffmann, "An Overview on Soft Computing in Behavior Based Robotics" 2002.

[2] Dongbing Gu, Huosheng Hu, Jeff Reynolds, Edward Tsang, "GA-based Learning in Behavior Based Robotics" Proceedings of IEEE International Symposium on Computational Intelligence in Robotics and Automation, Kobe, Japan, 16-20 July 2003.

[3] O. Hachour, "The proposed Fuzzy Logic Navigation approach of Autonomous Mobile robots in unknown environments" International Journal Of Mathematical Models And Methods In Applied Sciences, 2009.

[4] Frank Hoffmann, "Soft Computing echniques for the Design of Mobile Robot Behaviors" INFORMATION SCIENCES xx, 1{xx (1994).

[5]Piero P. Bonissone, Yu-To Chen, Kai Goebel, And Pratap S. Khedkar,"Hybrid Soft Computing Systems: Industrial and Commercial Applications" Proceedings Of The Ieee, Vol. 87, No. 9, September 1999 1641

[6] K. Funahashi, On the approximate realization of continuous mappings by neural networks, Neural Networks 2 (1989) 183–192.

[7] K. Hornik, Multilayer feedforward networks are universal approximators, Neural Networks 2 (1989) 359–366.

[8] J.-S.R. Jang, C.-T. Sun, E. Mizutani, Neuro-Fuzzy and Soft Computing, PTR Prentice Hall, Upper Saddle River, NJ, 1997, pp. 335–368.

[9] K.S. Narendra, K. Parthasarathy, Identification and control of dynamical systems using neural networks, IEEE Transactions on Neural Networks 1 (1) (1990) 4–27.

ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁘ Peer-reviewed & Refereed journal ⁘ Vol. 15, Issue 7, July 2026

DOI: 10.17148/IJARCCE.2026.15740

[10] D.E. Rumelhart, G.E. Hinton, R.J. Williams, Learning internal representations by error propagation, in: D.E. Rumelhart, J.L. McClelland and the PDP Research Group (Eds.), Parallel Distributed Processing, Vol. 1, MIT Press, Cambridge, MA, 1986, pp. 318–362.

[11]M. Onder Efe A comparative study of soft-computing methodologies in identification of robotic manipulators ,Robotics and Autonomous System.2000 .pp.221=300

[12] Alessandro Saotti, "Fuzzy logic in Autonomous Robot Navigation a case study" November 1995.

[13] Dusko Katic, Miomir Vukobratovic,"Genetic Algorithms in Robotics",International Series on Microprocessor-Based and Intelligent Systems Engineering Volume 25, 2003.

[14] Gerardo Beni, "From Swarm Intelligence to Swarm Robotics", Swarm Robotics_Springer-Verlag Berlin Heidelberg 2005

[15] Aleksandar Jevti´C, Diego Andina, "Swarm Intelligence and Its Applications in Swarm Robotics" 6th WSEAS Int. Conference on Computational Intelligence, Man-Machine Systems and Cybernetics, Tenerife, Spain,December 14-16, 2007.

[16] Abdollah Homaifar, Daryl Battle, Edward Tunstel, "Soft Computing-based Design and Control for Mobile Robot Path Tracking", 2010.

[17].Gurjeet Singh and V .k. Banga "ANFIS Implementation with robotic arm manipulator ", International journal of engineering research and technology (IJERT), Vol. 1 Issue 4, June – 2012, pp.143-149.

[18] Zhiying Zen ; Optimization of Analytical Inverse Kinematic Solution for Redundant Manipulators Using GA-PSO Algorithm: IEEE 2018pp446-451

[19] Li Fen1 Path Planning of 6-DOF Humanoid Manipulator Based on Improved Ant Colony Algorithm, IEEE 2000 pp4158-4161

[20] B. Durmuş1 An Inverse Kinematics Solution using Particle Swarm Optimization 6th International Advanced Technologies Symposium (IATS'11), 16-18 May 2011, Elazığ, Turkey pp193=197

