JETIR

ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue JETIR.ORG JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)

An International Scholarly Open Access, Peer-reviewed, Refereed Journal

FORWARD AND INVERSE KINEMATIC **ANALYSIS OF 4-DOF TRRR ROBOTIC ARM USING MATLAB**

¹Arumalla Johnson, ²M. Venkatesh,

¹Senior Assistant Professor, ² Assistant Professor

^{1&2}Department of Mechanical Engineering

¹Kamala Institute of Technology and Science, Singapur, Huzurabad, Telangana, India

Abstract: This paper aims to perform the kinematic analysis of Four degrees of freedom robotic arm. The kinematic analysis is the relationship between the positions, velocities, and accelerations of the links of a manipulator. The kinematics separate into two types, direct kinematics, and inverse kinematics. In forward kinematics, the length of each link and the angle of each joint is given and we have to calculate the position of any point in the work volume of the robot. In inverse kinematics, the length of each link and position of the point in work volume is given and we have to calculate the angle of each joint.

The joint arrangement of every robot can be described by the Denavit-Hartenberg parameters. These parameters are enough to obtain a working of the robot described and presented in a MATLAB program. The main aim of this work is to develop forward and inverse kinematic models of three degrees of freedom articulated arm using MATLAB GUI to optimize the manipulative task execution. Forward kinematics analysis is done for the flexible twist angle, link lengths, and link offsets of each joint by varying joint angles to specify the position and orientation of the end effectors. Forward analysis can be used to provide the position of some point on the end effectors together with the orientation of the end effectors measured relative to a coordinate system fixed to the ground for a specified set of joint variables. This simulation allows the user to get forward kinematics and inverse kinematics of articulated robots in various link length parameters and joint angles and corresponding end effector's position and orientation.

Index Terms – Kinematic Analysis, Forward Kinematics, Inverse Kinematics, MATLAB.

I. INTRODUCTION

A robot is a machine that collects information about the environment using some sensors and makes a decision automatically. People prefer it to use different fields, such as industry, some dangerous jobs including radioactive effects. In this point, robots are regarded as a server. They can be managed easily and provide many advantages. Robot kinematics is the study of the motion(kinematics) of robots. In a kinematic analysis, the position, velocity, and acceleration of all the links are calculated without considering the forces that cause this motion. The relationship between motion and the associated forces and torques is studied in robot dynamics [1]. Robot kinematics deals with aspects of redundancy, collision avoidance, and singularity avoidance. While dealing with the kinematics used in the robots, we deal with each part of the robot by assigning a frame of reference to it, and hence a robot with many parts may have many individual frames assigned to each movable part. For simplicity, we deal with the single manipulator arm of the robot. Each frame is named systematically with numbers, for example, the immovable base part of the manipulator is numbered 0, and the first link joined to the base is numbered 1, and the next link 2 and similarly till n for the last nth link [1]. In the kinematic analysis of manipulator position, there are two separate problems to solve: direct kinematics, and inverse kinematics. Direct kinematics involves solving the forward transformation equation to find the location of the hand in terms of the angles and displacements between the links. Inverse kinematics involves solving the inverse transformation equation to find the relationships between the links of the manipulator from the location of the hand in space. In the next chapters, inverse and forward kinematic will be represented

in detail [2]. A robot arm is a known manipulator. It is composed of a set of joints separated in space by the arm links. The joints are where the motion in the arm occurs. In basic, a robot arm consists of the parts: base, joints, links, and a gripper. The base is the basic part over the arm, it may be fixed or active. The joint is flexible and joins two separated links. The link is fixed and supports the gripper. The last part is a gripper. The gripper is used to hold [2].

In the paper, the Homogeneous transformation is used to solve kinematic problems. This transformation specifies the location (position and orientation) of the hand in space with respect to the base of the robot, but it does not tell us which configuration of the arm is required to achieve this location. It is often possible to achieve the same hand position with many arm configurations.

The Denavit-Hartenberg illustration of forward kinematic equations of robots has grown to be the standard procedure for model robots and their motions. The technique summarizes the relationship between two joints in a concise set of four parameters. Any robot can be modeled using the D-H representation. A computer code has been formed in MATLAB to implement the modeling of any robot with only the DH parameters as input. The purpose of the simulator is to create an accurate Forward kinematic and inverse kinematics representation of any type of robot and its motions. The simulator also allows for the independent manipulation of each joint of the modeled robot. Presented in this paper are the details of this TRRR Robot as well as background on the D-H representation and some analysis on how effectively the TRRR model is in MATLAB.

II.KINEMATICS OF TRRR ROBOT

2.1 Forward Kinematics

Four frames were assigned to the robotic arm as shown in Figure 2.1.1. The frame assignment procedure followed can be summarized in three steps

Assign the z_i axis pointing along the i-th joint axis (either one of two choices). It is better to assign $z_0 z_{n-1}$ and parallel to

 z_1 z_n and, respectively.

Assign the x_i axis pointing along the common perpendicular from z_i to z_{i-1}

Assign the y_i axis according to the right-hand rule.





The non-rotating reference Frame $\{0\}$ was attached to a fixed point at the center of the rotary base. Frame $\{1\}$ corresponding to Link (1) was attached at its upper end. Note that the origin of this frame could have been appointed to any location along with Link (1) since Z0 and Z1 are parallel to each other. Frame $\{2\}$ rotates with Link (2) and is located at the intersection between Link (1) and Link (2), thus having X2 along the common perpendicular pointing from Z2 to Z3. Frame $\{3\}$ rotates with Link (3) and is attached at its intersection with Link (2). Finally, Frame $\{5\}$ corresponding to the end-

effector is located midway between the two grippers. Based on the defined frames, the Denavit-Hartenberg (D-H) parameters were determined as shown in Table 2.1.

Link (i)	$ heta_i$	d_{i}	$lpha_{i}$	a_i	q_{i}	$C heta_i$	$S\theta_i$	$C \alpha_i$	$S\alpha_i$
1	$ heta_{ m l}$	$d_{_1}$	90	0	$ heta_{ m l}$	$C heta_1$	$S heta_1$	0	1
2	$ heta_2$	0	0	L_2	$ heta_2$	$C heta_2$	$S\theta_2$	1	0
3	θ_{3}	0	0	L_3	$ heta_3$	$C heta_3$	$S\theta_3$	1	0
4	$ heta_4$	0	0	L_4	$ heta_{_4}$	$C heta_4$	$S\theta_4$	1	0

Table 2.1.D-H parameters of a 4-DOF TRRR Robotic Arm

The definition of each of the D-H parameters is as follows:

- α_i : Link twist is the angle about x_i between (z_i, z_{i+1})
- a_i : Link length is the distance along X_i between (z_i, z_{i+1})
- d_i : Link offset is the distance along z_i between (x_{i-1} , x_i)
 - θ_i : Joint angle is the angle about z_i between (x_{i-1}, x_i)

The D-H parameters were then used to compute the homogenous transform matrices $(T_1^0, T_2^1 \ a T_3^2 \ and \ T_4^3)$ by substituting in the general formula T_i^{i-1} .

$$T_{i}^{i-1} = \begin{bmatrix} C\theta_{i} & -S\theta_{i}C\alpha_{i} & S\theta_{i}S\alpha_{i} & C\theta_{i}a_{i} \\ S\theta_{i} & C\theta_{i}C\alpha_{i} & -C\theta_{i}S\alpha_{i} & S\theta_{i}a_{i} \\ 0 & S\alpha_{i} & C\alpha_{i} & d_{i} \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad T_{1}^{0} = \begin{bmatrix} C_{1} & 0 & S_{1} & 0 \\ S_{1} & 0 & -C_{1} & 0 \\ 0 & 1 & 0 & d_{1} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

	$\left\lceil C_2 \right\rceil$	$-S_2$	0	C_2L_2		C_3	$-S_3$	0	C_3L_3		$\left\lceil C_4 \right\rceil$	$-S_4$	0	C_4L_4
T^1 –	S_2	C_2	0	S_2L_2	$T^{2} -$	S_3	C_3	0	S_3L_3	$T^3 -$	S_4	C_4	0	S_4L_4
$I_2 -$	0	0	1	0	$I_{3} -$	0	0	1	0	$I_4 -$	0	0	1	0
	0	0	0	1		0	0	0	1		0	0	0	1

Where C_i is the cosine of the angle θ_i , and S_i represents the sine.

By multiplying the homogeneous transformation matrices in the right order, T_4^0 describing the end-effector frame with the reference frame may be obtained.

$$T_{4}^{0} = T_{1}^{0} T_{2}^{1} T_{3}^{2} T_{4}^{3}$$
$$T_{4}^{0} = \begin{bmatrix} R(\theta) & P(\theta) \\ 0 & 1 \end{bmatrix}$$

$$R(\theta) = \begin{bmatrix} C_1 C_{234} & -C_1 S_{234} & S_1 \\ S_1 C_{234} & -S_1 S_{234} & -C_1 \\ S_{234} & C_{234} & 0 \end{bmatrix}$$
eq (1)

$$P(\theta) = \begin{vmatrix} dx \\ dy \\ dz \end{vmatrix} = \begin{bmatrix} C_1(L_3C_{23} + L_2C_2 + L_4C_{234}) \\ S_1(L_3C_{23} + L_2C_2 + L_4C_{234}) \\ d_1 + L_3S_{23} + L_2S_2 + L_4S_{234} \end{bmatrix}$$
Eq (2)

Notice that C_{234} is the cosine of the sum of $\theta_2 \ \theta_3 \ \theta_4 \ C_{23}$ and is the cosine of the sum of θ_2 and θ_3 . Similarly sine angles.

Here eq(1) represents the orientation of the end effector and eq(2) represents the position of an end effector.

The end effector orientation is:

The DH parameters are akin to the configuration of the robot. For different manipulator structures, the kinematics equations are not unique. Moreover, kinematics equations of the manipulator based on the DH convention provide some singularity making the equations difficult to solve or unsolvable in some cases. In addition, in the DH convention, the common normal is not defined properly when axes of two joints are parallel. In this case, the DH method has a singularity, where a little change in the spatial coordinates of the parallel joint axes can create a huge misconfiguration in the representation of the DH coordinates of their relative position.

2.2 Inverse Kinematics

Here we got a set of nonlinear equations with four unknowns. Solving these equations algebraically, known as inverse kinematics, requires that we need to know the joint variables θ_1 , θ_2 , θ_3 , θ_4 and for a given end-effector position and orientation. We get from equation (2), by dividing, squaring, adding, and using some trigonometric formulas.

Eq (4)

Eq (5)

$$\theta_1 = \tan^{-1}(\frac{d}{d})$$

$$\theta_2 = \tan^{-1}(c \pm \sqrt{r^2 - c^2}) - \tan^{-1}(a, b)$$

 $\theta_3 = \cos^{-1}\left(\frac{A^2 + B^2 + C^2 - l_2^2 - l_3^2}{2l_2 l_3}\right)$ Eq (6)

Where

$$a = l_3 \sin \theta_3$$
, $b = l_2 + l_3 \cos \theta_3$, $c = dz - d_1 - l_4 \sin \phi$, and $r = \sqrt{a^2 + b^2}$

 $A = (dx - l_4C_1C\phi), B = (dy - l_4S_1C\phi),$ and $C = (dz - d_1 - l_4S\phi),$

Having determined θ_1 , θ_2 and θ_3 , we can then find θ_4 from the end effector orientation ϕ as follows:

$$\theta_4 = \phi - \theta_2 - \theta_3$$

III.SIMULATION OF THE ROBOTIC ARM

Since robotics is a multidisciplinary subject, its study requires skills from different fields of knowledge, not easily available. Thus, simulation has been recognized as a suitable tool that combines all these features enabling the user of direct visualization of different kinds of motion that a robot may perform, making the role of simulation very important in robotics. Using the robotics toolbox together with the MATLAB software, the kinematics of a robotic arm can be simulated and analyzed based on the DH convention described before. The toolbox takes a conventional approach to represent the kinematics and dynamics of serial-link robotic arms. Besides, it provides several functions and routines, which are handy for the simulation and scrutiny of robotic manipulators, like kinematics, dynamics, and trajectory creation.

MATLAB is a software package for high-performance numerical computation and visualization. It provides an interactive environment with hundreds of built-in functions for technical computation, graphics, and animation. Best of all, it also provides easy extensibility with its high-level programming language. The name MATLAB stands for Matrix Laboratory.

The diagram in Fig 3.1 below shows the main features and capabilities of MATLAB. MATLAB's built-in functions provide excellent tools for linear algebra computations, data analysis, signal processing, optimization, numerical solution of ordinary differential equations (ODEs), quadrature, and many other types of scientific computations. Most of these functions use state-of-the-art algorithms. There are numerous functions for 2-D and 3-D graphics as well as for animation. Also, for those who cannot do without their Fortran or C codes, MATLAB even provides an external interface to run those programs from within MATLAB. The user, however, is not limited to the built-in functions; he can write his functions in the MATLAB language. Once written, these functions behave just like built-in functions. MATLAB's language is very easy to learn and to use.



. Fig.3.1 A schematic diagram of MATLAB features.

3.1 MATLAB Robot Creation Flow Chart

In this section, a detailed discussion on the implementation, creation, and forward and inverse kinematics analysis of the robotic arm using the MATLAB tool is provided. The flow chart below explains the process starting with creating the robot, controlling the robot with input joint angles, forward kinematics, and inverse kinematics functions used for the implementation.

JETIR2110290 Journal of Emerging Technologies and Innovative Research (JETIR) <u>www.jetir.org</u> c711

www.jetir.org (ISSN-2349-5162)

In this MATLAB R2019a was used for robot creation and simulation. In the MATLAB tool, the Robotic toolbox with the rvc feature which consists of the robotic 3D capability is initialized. With the RVC feature, the robot arm can be developed, controlled, and manipulated. we have a file, startup_rvm.m which calls several robot functions by calling the respective .m files of each function.





A MATLAB file, LINK.m performs the process of link creation. This file holds all information related to a robot link such as the kinematics parameters, rigid body inertial parameters, motor, and transmission parameters. Also, there are classes and functions to receive the four parameters of the DH convention and construct the link. The fifth parameter of a link determines its type, revolute (0), or prismatic (1). The robot is created by serially connecting all links. The MATLAB file, SerialLink.m does connect the vector of link objects and forms a serial-link robot object. The output of SerialLink.m gives all the links information and robot dimensions. Figure 1.4 shows the output of SerialLink.m (Serial-Link robot).

In this, a robot with four links in the revolute configuration is considered. The D-H parameters of these four DOF robots are obtained from the result as represented in the following figures

Sec. 1

1		
 MODULUTIN 		
state and and and		A CONTRACTOR AND A SALE
🖾 🖄 😅 🖾 Charman 🕉	a calender Calender Cale Breterie	
the day has the come of	and the Deschade's Lands - Deschade - Addition Hand	
Rose latitude + + N	A Mature Constitution	
CONTRACT AND A MARY & Burn & Aven	A Design A Martine A Martine and A Design A Trade of A Martine A M	
Control South	Extension Descent	Weiner .
C. Netw-	st fail	A success and the
0 b ettal	a a 1 2 0 1 2 12	and the second se
E = Aust	4.414	
# 8 statu	1 1 1 1 1 1	Tell 1. Art Barrell Ltd.
# 2 denor	8 0 0.8 W	and the second se
a = statem		
a content	n -	
C I Car	h 3.2000 V 3.5196	
# 3 receiv	1 1 1.0000 1	
6 7 private	n n e.7543. e	
T - minimum.	0 0 0.0,3930 0	
0 2 011		
the second	ap beginning (m)	
T I and Am		
# I und		
California California	BUILDERIN 4 4823, BUIL, TUSER, TUSER, TUSER	
T Antonin		
AdultAC-#	a generation of the state of th	
-latar00.e		
a horem		
and a strength of the strength		
F) (600- 70.0		
Acriation		
Contes		
Same	may reference to the second seco	
El contactor on		
Tumperat.		
C. Martin a	A 411 -	
Contract Sector		
		1.000/2 -
D. Tops have to march		TOTAL TOTAL
and the second s		A A A A A A A A A A A A A A A A A A A

Fig3.1.2 Four-link revolute robot (TRRR Arm)

www.jetir.org (ISSN-2349-5162)

Here · It Will (1) April 0 1.000 0 1.5700 April 0 1.000 0 1.5700 0 0 1.1000 0		Contract We	556 * MAJLAS A	dd Ors • Toolb	cais • Schools	a Torothick for MA	musees ×	and more	2010	
data # 1.2003 # 1.2708 # 1.003 # 1.2708 Mendu manya # 0 1.003 0 1.003 0 Manya # 0 0.1003 0 <th>ame - 15) 91</th> <th>- ai -</th> <th></th> <th></th> <th></th> <th></th> <th></th> <th>" have</th> <th> Weise Initial And shadow </th> <th>Mini D</th>	ame - 15) 91	- ai -						" have	 Weise Initial And shadow 	Mini D
and marking minipulation 0 0 1.0003 0 0 0 0.1003 0 0 0 0.1003 0 0 0 0.1003 0 0 0 0.1003 0 0 0 0.1003 0 0 0 0.1003 0 0 0 0.1003 0 0 0 0.1003 0 0 0 0.1003 0 0 0 0.1004 0 0 0 1.1 0.1 0.101 0 1.1 0.1 1.2.1 0.1 0.1 0 1.1 0.1 1.2.1 0.1 0.1 0 1.1 0.1 1.2.1 0.1 0.1 0.1 0 1.1 0.1 0.1 0.1 0.1 0.1 0 1.2.1 0.1 0.1 0.1 0.1 0.1	te		0 1.2003		1.8708			144, r	107 Seculari	
0 0 0 0 0 0 0 0 0 <td>tracia terración</td> <td></td> <td>2</td> <td>1,8008</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td>	tracia terración		2	1,8008						
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 1 1 1 0 0 0 1 1 1 0 0 0 1 1 1 0 0 0 1 1 0 0 0 0 1 1 0 0 0 0 1 1 0 0 0 0 1 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 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 <td>an garde</td> <td></td> <td>8 8</td> <td>0.7508</td> <td>1.6</td> <td></td> <td></td> <td></td> <td></td> <td></td>	an garde		8 8	0.7508	1.6					
M Xx ==0eriallikidiii Settion F = I F = I Annaerii 4 dalo, maar, rultri, slowno I I				0.5008	1.1					
9 >> 1=000111110010000 0 0 0 0 1 0 1 0 1 1										
with marries C = interme formame/i 4 wils, mmm, multit, slowmm wave iiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiii		30 today	infilition and the							
minu 0 matter 0 matter 0 matter 1 matter	odatu									
Impute Impute wave in the fill of the fill	water .	1.0								
Normality Downame ii 4 Akis, HTMPR, FULUHI, SlowHTMP Stati 13 thetta 0 14 0 12 0 <td>in film</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td>	in film									
Name Name Name a (3) theta (0,0) a (3) theta (0,0) a (3) theta (0,0) b (1) (1) (1) a (3) (1) (1) a (1) (1) (1) b (1) (1)		Colorane 1	a dariat men	a sectoral at	Longerter.					
Mathe I I I I I I I 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	1078	Strongerer	4 441237 1800	ex roaten p	Loanse.					
0 13 Carlos 13 Carlos 14 Carlos Carlos 11 14 1.21 0 1.25700 0 0 15 121 121 0 1.25700 0 0 10000 121 121 0 1.1 0 0 11 14 121 0 1.1 0 0 10000 13 0 8.751 0 0 10000 14 0 0.27 0 0 10000 14 0 0.27 0 0 10000 14 0 0.27 0 0 10000 14 12 10 0.27 0 10000 14 12 140 14 10000 14 12 140 10000 14 12 140 10000 14 12 140 10000 14 12 140 10000 14 12 140 10000 14 12 140 10000 14 12 140 10000 14 12 140 10000 14<	rdale.		Phates		0.00	a fatta d	uffort 1			
0.1903 0.1913 1.21 0.1 0.1 97054w 1.21 421 0.1 0.1 97054w 1.21 421 0.1 0.1 98054w 1.41 421 0.1 0.31 98054w 1.41 421 0.1 0.31 98054w 1.41 421 0.1 0.31 98054w 1.41 421 1.41 98054w 1.41 1.41 1.41 98054w 0.25362 -0.7648 -0.254 98054 0.23362 -0.7648 -0.254 98054 0.23362 -0.4646 0.2545 98054 0.23262 -0.4005 2.2304	0	1.4.1	COPLA			without 1	LILIEL I			
m 1 J q.1 1 J J <thj< th=""> <thj< th=""> <thj< th=""> <th< td=""><td>1,951</td><td>1.11</td><td>1.000</td><td>1.00</td><td></td><td>a marine</td><td></td><td></td><td></td><td></td></th<></thj<></thj<></thj<>	1,951	1.11	1.000	1.00		a marine				
Sector Sector<	The second s	- 문화	44.1	- H - H - H - H - H - H - H - H - H - H	100	- 1.1.1.01				
anno 1,21, 421, 0, 2,72, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,	9703.0W		44.1	50 B		U.				
Newcork 1 41 441 01 0.31 01 01 Opexam ************************************	Latory in the second se	1 31	411	85	0,753	01				
Approximation Total formation approximation See E.fkine (III) 14 12 14(1) approximation See E.fkine (IIII) 14 12 14(1)	terre and the second seco	1.41	-441	01	0.17	01	E1.			
ag2m cerk.wEisereg.co Cerk.wEisereg.co Cerk.wEisereg.co Cerk.wEisereg.co Kinologia Mich M	N. A. T.	+1-4-1+1-4-1				*********	······································			
OperAux[instration See E. Exclusion [14] 14 (12) 14(12) Obmy/data Obmy/data Altroin Altroin Million Obmy/data <	al.m	100003444		0.000						
CAW / Man Tahtew NKR88 00.2236 -0.7488 -0.254 NKR89 0.2176 -0.4806 0.1254 0.2176 -0.4806 0.1255 UK/NKR05 -0.8165 +0.4008 -0.1008 2.304	overlanding ro	2011.16	Dettin 14-12	1413						
CATCRN 403 % 900938 403 % 100-0 02336 -07643 -0214 00-0 0176 -04606 06295 -02285 DEMONDUDS -04105 -04005 -02304	2bw,rtbm									
XH034 4634 0.53236 -0.32336 -0.25463 -0.2546 -	TATEIN.	10.0000								
App 0.2316 -0.346 -0.248 optimiz 0.2176 -0.4906 -0.2328 DNADRODS -0.4016 0.1295 -0.2328	WORR .	404 +								
0a20m 0.3176 -0.400 0.1393 -0.2285 DRINENDIS -0.4001 -0.1001 2.304	8.41	0,35	416 -0,3680	-0.1440	-0.254					
-0.0145 +0.4000 -0.1000 2.100	Ražituti	0.21	76 -0.4908	0,0395	-0.2295					
	PENDENCIES	-0.81	45 -0.4000	-0,0000	2.104					
	Hersone	*	Q 2	1.1	1					

Fig3.1.3 Forward kinematics with joint angles (10,14,12,16)

Forward kinematics with joint angles (10,14,12,16) are obtained in a 4x4 matrix form from which the values of the position vectors are obtained.



Fig3.1.4 Robot position with joint angles (10,14,12,16)

www.jetir.org (ISSN-2349-5162)



Fig.3.1.5 Forward kinematics with joint angles (1,4,6,10)

Forward kinematics with joint angles (1,4,6,10) obtained in 4x4 matrix form from which the values of the position vectors are obtained



Fig 3.1.6 Robot position with joint angles (1,4,6,10)

www.jetir.org (ISSN-2349-5162)

	THE NEW PROFESSION	
C g2 * + C + IMen + Ram + AppOrts + Rom	Anny + MuthWork + Minilak Add Chi + Touttenet + Rokoly Toothor for Minilak (s. +	2 Webure
Linux -	1 31 (441 33 State 1 41	P hand in the later
152	1 41 g41 11 0_51 #1 #1	TEPTO - VALUE DOIN
10	++	107 MD
ta III		TILES SPECIAL CONTRACT
P804	yo s.fitime(10) 14 12 14()	and the second sec
anales	Sector Physics against a sector secto	
arrigante .	and the second	
14-17-11	A52	
	3.3356 -0.7696 -0.5448 -0.354	
B	3.7176	
pdets.	-3 9145 -6 4800 -0.0008 -2.204	
ester and a second		
1	ss r. procettan re ar reli	
nutrik (>> r.fmine([0 14 3 12])	
related		
H		
UT. 1995	ATT THE DESIGN AND A DESIGN AND	
df 10-1-1-1	-0.1331 -0.0500 0.9094 -0.1393	
trobility	8,9050 8,3997 0,1455 0,9488	
tar.m	-0.404D 0.9147 -0.0008 -0.5452	
tarMOO M		
tarPD.m	55 r. plot(18, 13, 2, 133)	
geisin.	vs - Permedits 18 73 2818	
igit m		
odehartstrating m		
disciption .		
TADION -	AT3 =	
24748	0,7594 -9,0202 0,6503 0,4406	
ajire	-0.8501 0.0173 0.7597 -0.3772	
Haulann	-0.0266 -0.9996 -0.0000 D.2168	
EPENDENCES	0 0 1	
Thinking .		
		100

Fig.3.1.7 Forward kinematics with joint angles (15,18,23,25)

Forward kinematics with joint angles (15,18,23,25) are obtained in a 4x4 matrix form from which the values of the position vectors are obtained.



To view the 3D representation of the robot created, the Plot.m function is executed on the robot links to display the graphical animation based on the kinematic model. The plot function in the file joins the origins of the link co-ordinate frames and displays the 3D-View. The robot's movement can be controlled by changing the joint angles.

The file, fkine.m has a function that outputs the robot-end effector position as a homogenous transformation for the joint configuration. The result is provided in a matrix format as described in the forward kinematic chapter. The fkine.m file uses the transformation matrix method to provide the result. Similarly, the file ikine.m outputs the joint coordinates corresponding to the robot end-effector position.

IV.RESULTS AND SUMMARY

The following table shows the parameters of each link of the four-link robotic arm. The parameters are joint offset, link length, twist angle, and joint angle. Also, it shows the configuration of each link (prismatic or revolute). All these parameters together form the DH parameters for the four-link robotic arm used in this project.

LINK	Joint angle	Joint offset (d_i)	Link length (a_i)	Twist angle (α_i)	Prismatic or Revolute
Link 1	q_1	1.2	0	$\frac{\pi}{2}$	Revolute (twist)
Link 2	q_2	0	1	0	Revolute
Link 3	q_3	0	0.75	0	Revolute
Link 4	q_4	0	0.5	0	Revolute

Table	4.1:	Robot link	output
I uoio		ROOOt min	output

Table 4.1 shows the four joint-link Parameters for each input (example). The robot position for each input (example) is shown in the previous section. It is also proven that changing the angle of any joint would result in a different end-effector position.

. Table 4.2: Input Joint Angles

Joint Variable	Joint angle 1	Joint angle 2	Joint angles 3
θ_1	10		15
$ heta_2$	14	4	18
$ heta_3$	12	6	23
$ heta_4$	16	10	25

Table 4.2 shows the input joint angles at three joints excluding the base joint in terms of four variables

Position vector	Joint angle 1	Joint angle 2	Joint angle 3
r_1	0.3356	0.2205	0.7594
r_2	-0.7690	-0.4933	-0.0202
<i>r</i> ₃	-0.5440	0.8415	0.6503
r_4	0.2176	0.3434	-0.6501
<i>r</i> ₅	-0.4986	-0.7682	0.0173
<i>r</i> ₆	0.8391	-0.5403	0.7597
<i>r</i> ₇	-0.9165	0.9129	-0.0266
r ₈	-0.4000	0.4081	-0.9996
r_9	-0.0000	-0.0000	-0.0000
p_x	-0.354	-0.5829	0.4406
p_y	-0.2295	-0.9079	-0.3772
p_z	2.304	0.4917	0.3168

Table 4.3 shows the forward kinematics result for the input combinations (with three different joint angles) mentioned in table 3. $r_1 - r_9$ Vectors represent the orientation of the end-effector of the robot. Vector P represents the position of the end-effector of the robot.

Joint Variable	Joint angle 1	Joint angle 2	Joint angle 3
$ heta_1$	10	T	15
$ heta_2$	14	4	18
$ heta_{3}$	12	6	23
$ heta_{_4}$	16	10	25

Table 4.4: Inverse Kinematics Result

Table 4.4 shows the inverse kinematics result of different input combinations of the position vectors obtained in the forward kinematics result mentioned in table 4. With the orientation and position vectors as input, the joint angles are obtained as output. It proves that forward and inverse kinematics is an inverse function of each other.

V. CONCLUSION

In this Paper, Forward and Inverse Kinematic Analysis of a 4- DOF TRRR articulated arm was studied. Then simulation of the robotic arm was done using the MATLAB simulation tool. The simulation has detected the movement of each joint of the robot arm the motion of the robot was controlled in different directions using various joint angle combinations. The concept of forward and inverse kinematics was used to determine the end-effector position for fixed joint angles and the joint angles for a fixed end-effecter. Through research and implementation of the kinematics concept in the work, it became clear that inverse kinematics is widely used in practical applications. The result of the forward kinematics was also discussed for each joint angle combination emphasizing the difference in the configuration of the position vector. The results confirmed that the forward kinematics is the inverse function of the inverse kinematics, i.e. the forward kinematics problem uses the kinematics for a desired position of the end effector.

REFERENCES

[1] Robot Kinematics, Wikipedia Web Site. http://www.wikipedia.com

[2] Kay, J. Introduction to Homogeneous Transformations & Robot Kinematics, Rowan University Computer Science Department

[3] Mittal, RK; Nagrath, J; Robotics and Control, Tata McGraw-Hill. 2005

[4] Denavit, J. and Hartenberg, R.S. 1955. A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices. Journal of Applied Mechanics 22 (June 1955), 215 – 221

[5] Shah, J., Rattan, S. S., and Nakra, B.C. 2013. EndEffector Position Analysis Using Forward Kinematics for 5 DOF Pravak Robot Arm. International Journal of Robotics and Automation 2(3) (Sept. 2013), 112 - 116.

[6] Desh ande VA, and P.M. George, "Analytical solution for inverse kinematics of Scorbot ER Vu Plus, International Journal of Emerging Technology Advanced. Engineering, Vol. 2, pp. 478-481, 2012.

[7] Niku, SB; Introduction to Robotics: Analysis, Systems, Applications. Prentice-Hall. 2001