Dynamics Of Non Linear Robotic Systems

SUBMITTED TO:

Alexander Klimchik

SUBMITTED BY:

Muhammad Ilyas Raza
m.raza@innopolis.university

Institute of Robotics
Innopolis University, Russia

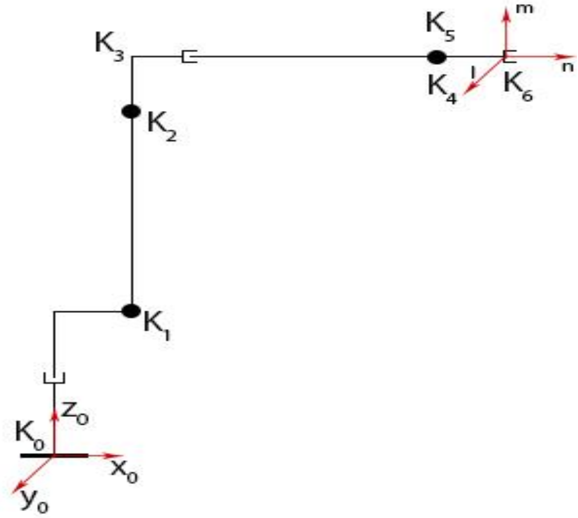
KUKA KR 10 R1100-2.

Robot description

KUKA KR 10 R1100-2.robot is a 6 dof robot capable of handling 20kg load capacity and reach has horizontical range 2150 mm and has 1972mm vertical range. Equipped with a thin wrist and capable of delivering excellent cycle times. KUKA KR 10 R1100-2.also consists of a fixed base, a serial chain of flexible links, a number of flexible actuated joints and an “End-effector”. So all links are elastic and all joints are actuated. All joints are rotational.

Forward kinematics

Kinematic model of the KUKA KR 10 R1100-2 robot can be drawn as shown below.



Any robot manipulator can be described kinematically by giving the values of four quantities for each link of robot joint: link twist – α, joint angle – θ, link length – a, link offset – d. Two describe the link itself, and two describe the link's connection to a neighboring link. The definition of mechanisms by means of these quantities is a convention usually called the Denavit—Hartenberg notation ..

D-H parameters for KUKA KR Agilus is shown in Table 1. All the a_i and d_i values can be found from the robot specification .

Link	θ_i	α_i	a_i	d_i
1	Q1	90	25	400
2	Q2	0	560	0
3	Q3	90	25	0
4	Q4	-90	0	515
5	Q5	90	0	0
6	Q6	0	0	90

.

For forward kinematics we will follow the following step;

I. We will generate the matrices for forward kinematics using DH-parameter given following

$$A_i = \begin{bmatrix} c_{\theta_i} & -s_{\theta_i}c_{\alpha_i} & s_{\theta_i}s_{\alpha_i} & a_i c_{\theta_i} \\ s_{\theta_i} & c_{\theta_i}c_{\alpha_i} & -c_{\theta_i}s_{\alpha_i} & a_i s_{\theta_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

II. By entering the parameter for each link we will drive the six matrices A1,A2,A3,A4,A5,A6

III. After that we will multiply all six matrices and final result will be give the forward kinematics

FK= A1*A2*A3*A4*A5*A6

The whole result is shown in matlab file which is names as **assigment.mat**

.

Inverse Kinematics

The task of inverse kinematics is to find out the joint variables when you are given end effector position and orientation. For the case of Fanuc Robot, all six joint variables are angles. The task of inverse kinematics for these kinds of serial manipulators is solved by **Pieper solution** which suggests the robot as two decoupled manipulators in which the first three joints are responsible for positioning of the end effector whereas the second three joints are responsible for orientation of the end effector. The first three joints are treated as 3R Orthoparallel basis structure and the second three as spherical wrist. The robot can be modeled as shown in fig. 1

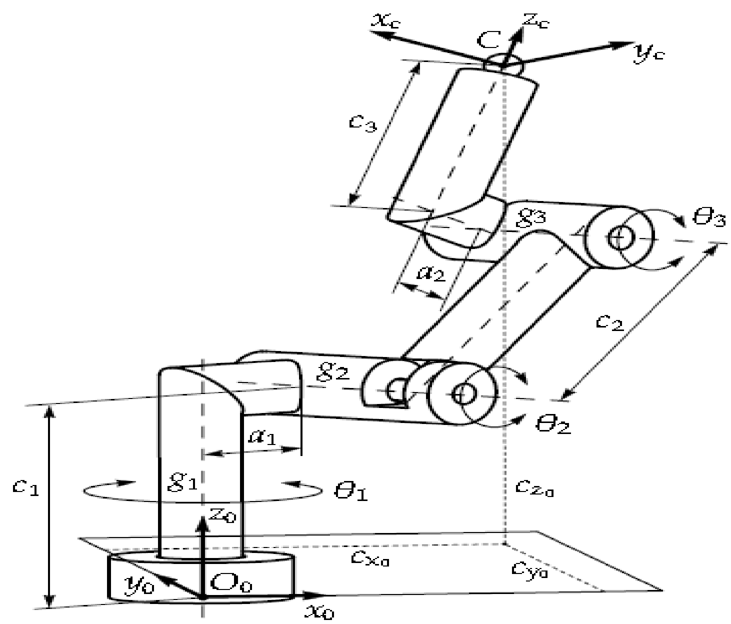


Fig. 2. Scheme with parameters of a serial ortho-parallel 3R manipulator.

Robot, the parameters are written as follows:

	a1	a2	b	c1	c2	c3	c4
Link Lengths (mm)	25	25	0	400	560	515	90

So in this framework, the diagram can be shortened as follows by replacing last three joints with a spherical joint.

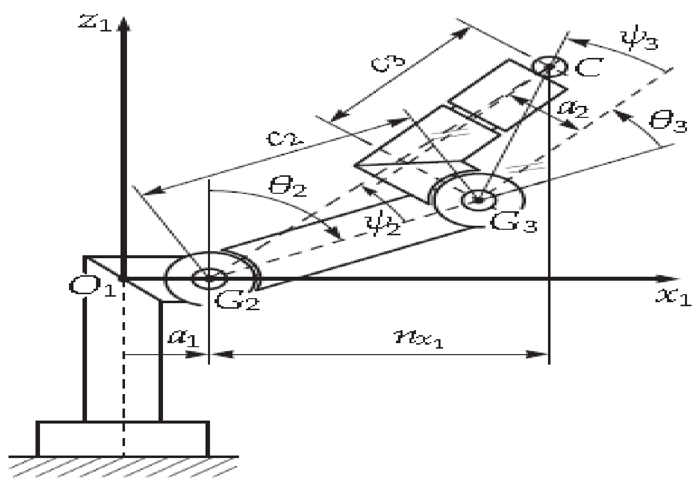


Fig. 3. Planar 2dof resemblance for first 3 joints

Positioning part formulas:

$$n_{x_1} = \sqrt{c_{x_0}^2 + c_{y_0}^2 - b^2} - a_1$$
$$s_1^2 = n_{x_1}^2 + (c_{z0} - c_1)^2$$
$$s_2^2 = (n_{x_1} + 2\,a_1)^2 + (c_{z0} - c_1)^2$$
$$k^2 = a_2^2 + c_3^2$$

Positioning part formulas:

θ_1 = atan2 (C_{y0}, C_{x0}) – atan2 (b, n_{x1} + a1)
θ_2 = acos (($s_1^2 + c_2^2 - k^2$)/(2*s_1*c_2)) + atan2 (n_{x1}, c_{z0} – c1)
θ_3 = acos (($s_1^2 - c_2^2 - k^2$)/(2*c_2*k)) – atan2(a_2, c_3)

Orientation part formulas:

m_p = e(1,3)*sin($\theta_2 + \theta_3$)*cosθ_1 + e(2,3)*sin($\theta_2 + \theta_3$)*sinθ_1 + e(3,3)*cos($\theta_2 + \theta_3$)

θ_4 = atan2 (e(2,3)*cosθ_1 – e(1,3)*sinθ_1, e(1,3)*cos(θ_2+θ_3)*cosθ_1 + e(2,3)*cos(θ_2+θ_3)*sinθ_1 – e(3,3)*sin(θ_2+θ_3)

θ_5 = atan2 (sqrt(1-m_p^2), m_p)

θ_6 = atan2 (e(1,2)*sin(θ_2+θ_3)*cos θ_1 + e(2,2)* sin(θ_2+θ_3)*sinθ_1, -e(1,1)*sin(θ_2+θ_3)*cosθ_1 - e(2,1)*sin(θ_2+θ_3)*sinθ_1 - e(3,1)*cos(θ_2+θ_3)

The values of final joint variables are saved in **‘assigment.mat’** file.