INVERSE KINEMATICS SOLUTION FOR 6 DEGREE OF FREEDOM OF FANUC LR MATE 200IB ROBOT

Mohd Khafizuddin Bin Khairuddin

Bachelor of Mechatronic Engineering
May 2010

“I hereby declare that I have read through this report entitle “Inverse Kinematics Solution For 6 Degree of Freedom of Fanuc LR Mate 200iB Robot” and found that it has comply the partial fulfillment for awarding the degree of Bachelor of Mechatronics Engineering”

Signature : ……………………………….

Supervisor’s Name : ……………………………….

Date : ……………………………….

INVERSE KINEMATICS SOLUTION FOR 6 DEGREE OF FREEDOM OF FANUC LR MATE 200IB ROBOT

MOHD KHAFIZUDDIN BIN KHAIRUDDIN

A report submitted in partial fulfillment of the requirements for the degree of
Bachelor of Mechatronics Engineering

Faculty of Electrical Engineering

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2010

I declare that this report entitle “Inverse Kinematics Solution For 6 Degree of Freedom of Fanuc LR Mate 200iB Robot” is the result of my own research except as cited in the references. The report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature : ....

Name : ....

Date : ………………………………..........

ACKNOWLEDGEMENT

Alhamdulillah, finally the Final Year Project 2 (FYP 2) is complete. There are several people I would like to acknowledge for their assistance and support over the FYP 2 which is compulsory to all UTeM students to pass before awarded the degree.

First of all, I would like to sincerely thank to my project supervisor, Pn Irma Wani bt Jamaludin for undivided support morally and physically, assistance, guidance, tolerance, which proved to be invaluable as to the completion of my FYP 2. I would like to also express my gratitude to the panels, Muhammad Herman bin Jamaluddin and Mohd Razali bin Mohamad Sapiee, who give me a good comment during my presentation.

Thanks are also extended to my family and friends for their patients, understanding and also for their undivided support that they had gave me throughout the completion of my project. Last but not least, I also would like to thank all those helping and supporting me directly and indirectly during my Final Year Project 2.

ABSTRACT

This project develops an algorithm for solving the inverse kinematics equation. The inverse kinematics equation is solving to determine angles (θ) of rotation linkages of the three main axes of the robot by using programming software for 6 DOF of Fanuc LR Mate 200iB robot called ivFanuc. The process involves the control of parameter known as the value of the final position and orientation of the robot. This parameter is determined by user before solve the inverse kinematics equation. Its inverse kinematics problem will be solved by using conventional method and simulation. Both results will be compared to ensure the algorithm that was developed is correct and can be used for solving the inverse kinematics equation. The GUI was designed which ask user to insert the position and then the software will show the theta (θ). The correct output of interface prove that this command can be used for solves the inverse kinematics equation. Overall, this project will describe the technique, calculation, method and algorithm on how the inverse kinematics is solved.

ABSTRAK

Projek ini membangunkan algoritma untuk menyelesaikan permasalahan kinematik songsang. Permasalahan kinematik ini diselesaikan untuk menentukan sudut putaran bagi 3 paksi dengan menggunakan perisian untuk robot Fanuc LR Mate 200iB yang dipanggil ivFanuc. Proses yang terlibat ialah kawalan terhadap parameter iaitu nilai pada kedudukan dan pergerakan akhir robot itu. Parameter ini akan ditentukan oleh pengguna bagi menyelesaikan permasalahan kinematik tersebut. Permasalahan kinematik akan diselesaikan menggunakan kaedah konvensional dan simulasi. Kedua-dua hasil akan dibandingkan untuk memastikan bahawa algoritma yang dibangunkan berjaya dan boleh digunakan untuk menyelesaikan permasalahan kinematik. Seterusnya, pengantara muka telah direka di mana pengguna akan diminta untuk memasukkan nilai kedudukan dan kemudian memberikan sudut yang dimahukan. Hasil yang tepat dari pengantara muka ini membuktikan bahawa pengantara muka ini boleh digunakan untuk menyelesaikan permasalahan kinematik. Secara keseluruhannya, projek ini menerangkan tentang teknik, pengiraan, kaedah dan algoritma bagaimana permasalahan kinematik diselesaikan.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

DH - Denavit-Hartenberg
DOF - Degree of Freedom
GUI - Graphical User Interface
KAP - Kinematics Analysis Program
EOAT - End-of-arm-tooling

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

The “Inverse Kinematics Solution for 6 Degree of Freedom of Fanuc LR Mate 200iB Robot” project is to identify and develop software and design graphical user interface (GUI) to determine the angle of each joints after the final position of the Fanuc robot is inserted by user. Figure 1.1 shows the Fanuc LR Mate 200iB robot. The GUI will be used and help the user to calculate the inverse kinematics problem for 6 DOF of Fanuc LR Mate 200iB robot. This chapter will discuss about the background of the project, the problem statement, objective and scope of the project.

Figure 1.1 Fanuc LR Mate 200iB

1.1 Background of the project

Today robots are used in every walk of human life. All over the world, robots are on the move. As the robots grow tougher, nimbler, and smarter, today's robots are doing more and more things human cannot or do not want to do. In order to co-operate with a human, a robot should have a humanlike behavior when moving. Then, the robot can act in a world made for humans and does not perform movements which cannot be anticipated by the human partner. To achieve this, it is necessary to give the robot human like configuration and human like kinematics.

Robotics is a multidisciplinary, highly mathematical topic usually taught at the graduate level. A typical course will start with a review or introduction of mathematical concepts, then on to kinematics of robot manipulators, followed by elements of dynamics and control that still rely heavily on the kinematics portion of the course. [1]

Kinematics is one of the most important aspects of robots movement and it is essential to understand the concept to analyze the complex movements of a robot and the operations it can perform. The human like shape and configuration of the robot enables the robot to co-operate with the human partner the same way humans co-operate with each other. Therefore, the robots arm moves like a human arm and interacts with a human by physical means.

In order to solve the kinematics problem, a conventional method which has a lot of complexity equation and calculation must be applied. However, it is hard to be implemented in practice even many industrial robots are built with simple geometries to simplify the associated kinematics computations such as Fanuc LR Mate 200iB robot.

It is now well accepted that the use a package of software is the best options to solve the inverse kinematics problem. In this project, a GUI will be designed for solving the inverse kinematics problems which focus only for Fanuc LR Mate 200iB robot. The advantage of this software is to minimize time for solving the inverse kinematics.

1.2 Problem statement

Many industrial robots are built with simple geometries such as intersecting or parallel joint axes to simplify the associated kinematics computations. The mathematical complexity of solving robots of general architecture detracts instructors and students from using robots with arbitrary structures in illustrative examples and assignments. This is also one of the major problems for the 6 DOF of an industrial robot which is to solve the inverse kinematics.

The inverse kinematics problem is much more interesting and its solution is more useful. It does the reverse of forward kinematics. Given the end point of the structure, what angles do the joints need to achieve that end point? At the position level, the problem is stated as, "Given the desired position of the robot's hand, what must be the angles at all of the robots joints?" Humans solve this problem all the time without even thinking about it. When you are eating your cereal in the morning you just reach out and grab your spoon. You don't think, "My shoulder needs to do this, my elbow needs to do that, etc." So, we will look how most robots have to solve the problem. [2]

Solving the inverse kinematics problem need a lot of calculation when using conventional method. The complexity of formulation and computational burden make this method hard to be implemented in practice which also will waste a lot of time. Therefore, a new method of solution must be developed to solve the inverse kinematics equation such as using software programming.

A software package that eases the kinematics study of robots also provides computer assistance and time savings in all associated areas of robotics. While the forward kinematics is fairly easy and always leads to a unique solution, the inverse kinematics is far more mathematically involved and usually leads to several solutions. The inverse kinematics of high degree of freedom manipulators is one of the hardest problems in robotics.

1.3 Objective

The objectives of this project are to solve an inverse kinematics for 6 DOF of Fanuc LR Mate 200iB robot. In order to solve the problem, the algorithm will be develop and then the programming software will be identify and develop which is to design the graphical user interface. Specifics objectives of this project are:

1. To calculate inverse kinematics solution.
2. To develop an algorithm of inverse kinematics solution.
3. To design a graphical user interface (GUI) for inverse kinematics for 6 DOF of Fanuc LR Mate 200ib robot.

1.4 Scope

In general, this project is to identify and develop the programming software for solving the inverse kinematics problem which focuses only for 6 DOF of Fanuc LR Mate 200iB robot. Thus, it cannot be apply to other type of robot unless both program and interface is modified.

To solve the inverse kinematics problem for 6 DOF of Fanuc LR Mate 200iB robot, the programming software will be identify whether it is suitable for solve the inverse kinematics. Then, an algorithm will be developed. At the end the GUI will be developed to solve the inverse kinematics problem.

Type of programming software for develops an algorithm and GUI is already identified. MATLAB is chosen for design and develop an algorithm and GUI.

CHAPTER 2

LITERATURE REVIEW

In this chapter, a review of previous research project that are related of this project will be discussed. The information about forward and inverse kinematics and Fanuc LR Mate 200iB robot are also described in this chapter.

2.1 Forward Kinematics

The forward kinematics problem is concerned with the relationship between the individual joints of the robot manipulator and the position and orientation of the tool or end-effector. Stated more formally, the forward kinematics problem is to determine the position and orientation of the end-effector, given the values for the joint variables of the robot. The joint variables are the angles between the links in the case of revolute or rotational joints, and the link extension in the case of prismatic or sliding joints.

In order to describe the location of each link relatives to its neighbors we define a frame attached to each link. The link frames are named by number according to the link to which they are attached. That is, frame $\{i\}$ is attached rigidly to link i. To locate frames on the links, this convention is used: The Z-axis of frame $\{i\}$, called Z_i is coincident with the joint axis i. The origin of frame $\{i\}$ is located where a_i perpendicular intersects the joint i axis. X_i points along a_i in the direction from joint i to joint $i+1$ [3].

In the usual case of revolute joint, θ_i is called the joint variable, and the other three quantities would be fixed link parameters. For prismatic joints, d_i is the joint variable and the other three quantities are fixed link parameters. The definition of mechanism by means of these quantities is a convention usually called Denavit-Hartenberg (DH) notation [3].

2.1.1 Denavit Hartenberg Representation

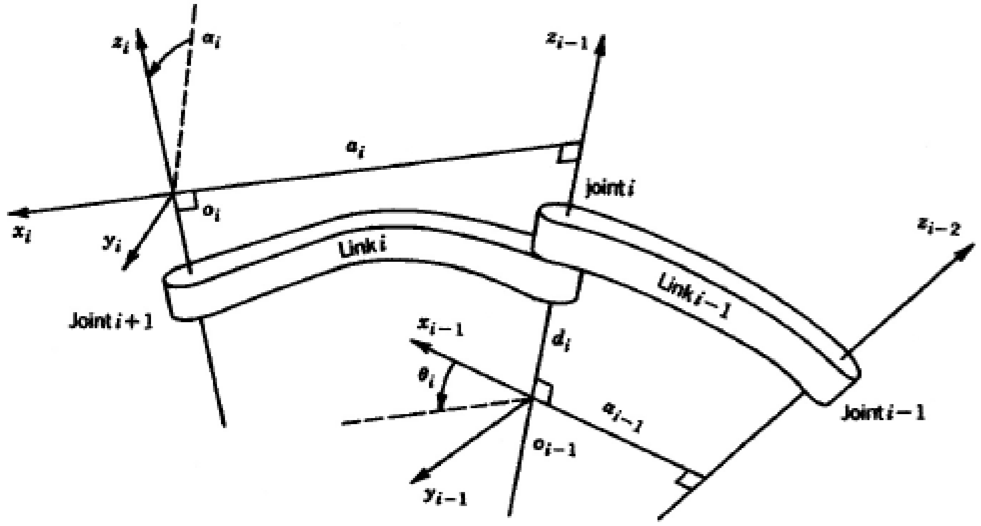

Figure 2.1: Link Parameter.

A commonly used convention for selecting frames of reference in robotic applications is the Denavit-Hartenberg, or DH convention. In this convention, each homogeneous transformation A_i is represented as a product of four basic transformations. The procedure based on the DH convention in the following algorithm for deriving the forward kinematics for any manipulator are summarize as below [4]:

Step 1 : Locate and label the joint axes $z_0, \ldots., z_{n-1}$.

Step 2 : Establish the base frame. Set the origin anywhere on the z_0-axis. The x_0 and y_0 axes are chosen conveniently to form a right-hand frame. For $i = 1, \ldots., n-1$, perform Steps 3 to 5.

Step 3 : Locate the origin O_i where the common normal to z_i and z_{i-1} intersects z_i. If z_i intersects z_{i-1} locate O_i at this intersection. If z_i and z_{i-1} are parallel, locate O_i in any convenient position along z_i.

Step 4: Establish xi along the common normal between z_{i-1} and z_i through O_i i, or in the direction normal to the $z_{i-1} - z_i$ plane if z_{i-1} and z_i intersect.

Step 5 : Establish y_i to complete a right-hand frame.

Step 6 : Establish the end-effector frame $o_n x_n y_n z_n$. Assuming the *n*-th joint is revolute set $z_n = a$ along the direction z_{n-1}. Establish the origin o_n conveniently along z_n, preferably at the center of the gripper or at the tip of any tool that the manipulator may be carrying. Set $y_n = s$ in the direction of the gripper closure and set $x_n = n$ as $s \times a$. If the tool is not a simple gripper set x_n and y_n conveniently to form a right-hand frame.

Step 7 : Create a table of link parameters $a_i, d_i, \alpha_i, \theta_i$.

$a_i =$ Distance along xi from o_i to the intersection of the xi and z_{i-1} axes.

$d_i =$ Distance along z_{i-1} from O_{i-1} to the intersection of the x_i and z_{i-1} axes. d_i is variable if joint i is prismatic.

$\alpha_i =$ The angle between z_{i-1} and z_i measured about x_i (see Figure 2.1).

$\theta_i =$ The angle between x_{i-1} and x_i measured about z_{i-1} (see Figure 2.1). θ_i is variable if joint i is revolute.

Step 8 : Form the homogeneous transformation matrices A_i by substituting the above parameters into equation 2.1.

Step 9 : Form $T_n^0 = A_1 \ldots. A_n$. This then gives the position and orientation of the tool frame expressed in base coordinates.

The overall transformation is obtained by post multiplication of individual transformations:

$$T_i^{i+1} = Rot(z,\theta_i)Trans(0,0,d_i)Trans(0,0,a_i)Rot(x,\alpha_i) \tag{2.1}$$

$$T_i^{i+1} = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & 0 \\ s\theta_i & c\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c\alpha_i & -s\alpha_i & 0 \\ 0 & -s\alpha_i & c\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.2}$$

$$T_i^{i+1} = \begin{pmatrix} \cos\theta_i & -\sin\theta_i\cos\alpha i & \sin\theta_i\sin\alpha_i & a_i\cos\theta_i \\ \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i & a_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{2.3}$$

2.2 Inverse Kinematics

The inverse kinematics problem is concerned with the inverse problem of finding the joint variables in terms of the end-effector position and orientation. This is the problem of inverse kinematics, and it is, in general, more difficult than the forward kinematics problem. It is begin by formulating the general inverse kinematics problem. Following this, it will describe the principle of kinematic decoupling and how it can be used to simplify the inverse kinematics of most modern manipulators. Using kinematic decoupling, we can consider the position and orientation problems independently. We describe a geometric approach for solving the positioning problem, while we exploit the Euler angle parameterization to solve the orientation problem.

In the general problem of inverse kinematics is quite difficult, it turns out that for manipulators having six joints, with the last three joints intersecting at a point. It is possible to decouple the inverse kinematics problem into two simpler problems, known respectively, as inverse position kinematics, and inverse orientation kinematics. To put it another way, for a 6 DOF manipulator with a spherical wrist, the inverse kinematics problem may be separated into two simpler problems, namely first finding the position of the intersection of the wrist axes, hereafter called the wrist center, and then finding the orientation of the wrist.

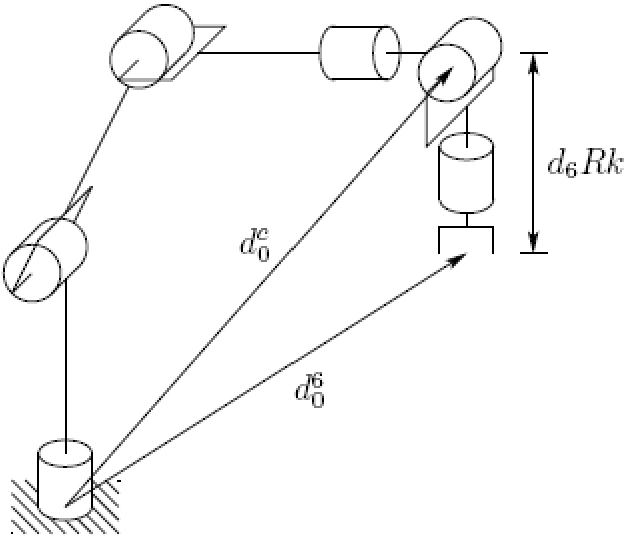

Figure 2.2: Kinematic Decoupling

For this class of manipulators the determination of the inverse kinematics can be summarized by the following algorithm.

Step 1: Find q1, q2, q3 such that the wrist center o_c has coordinates given by

$$o_c^0 = o - d_6 R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Step 2: Using the joint variables determined in Step 1, evaluate R_3^0.

Step 3: Find a set of Euler angles corresponding to the rotation matrix

$$R_6^3 = \left(R_3^0\right)^{-1} R = \left(R_3^0\right)^T R$$

2.3 Kinematics Analysis Program (KAP)

Kinematics Analysis Program is a program written by Rachid Manseur [1]. KAP is currently written to compute the forward and inverse kinematics of robot manipulators with 5 or 6 revolute degrees of freedom. The functional block diagram of KAP is shown on Figure 2.3. There are two types of KAP:

i. KAP5 – to compute the inverse kinematics problem for 5 DOF of Rhino XR-3 robot.
ii. KAP6 – to compute the forward kinematics of Fanuc LR Mate 200iB robot.

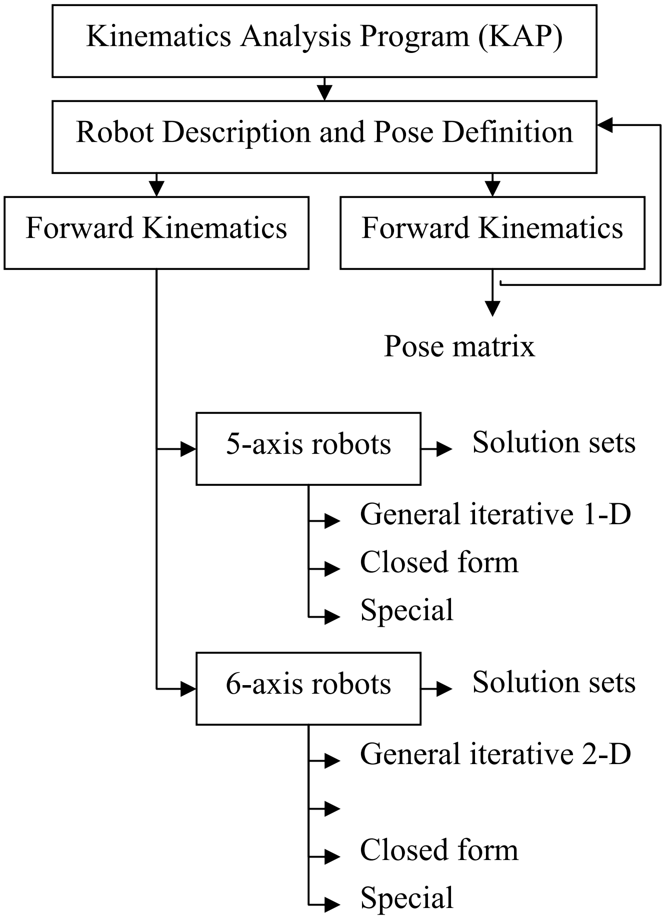

Figure 2.3: Functional Block Diagram of KAP

In this program, the user is asked to enter the Denavit-Hartenberg (DH) parameter table for the robot. The DH-parameters can be entered from the keyboard or by specifying a pre-existing file name containing the robot DH parameters in the format shown on Table 2.1. The end effectors pose can also be entered through the keyboard, read from a pre-existing file named POSE.DAT, or computed from joint angles that the user enters through the forward kinematics block. The program determines the number of DOF of the robot and applies the proper algorithm [1]. Even this program is to compute the inverse kinematics of Fanuc LR Mate 200iB robot, it still can be use as a sample for design a GUI for this project.