A software package for computer-aided robotics education

Rachid Manseur
Electrical Engineering
University of West Florida
Pensacola, Florida 32514

Abstract

The kinematics of robot manipulators is a corner stone in the study of robotics in general. The computational complexity of the kinematics quite often prevents robotics instructors from using robots of general structure in their illustrative examples and assignments. The software discussed in this article, developed from recent research to support an undergraduate course in robotics, offers computational relief to educators and students in the study of robot manipulators with revolute joints and renders their use as classroom examples possible . Five or six revolute joints robot manipulators of general architecture can be solved easily with the help of this software

Introduction

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. 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. Many industrial robots are built with simple geometries such as intersecting or parallel joint axes to simplify the associated kinematic 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. Several well known texts in robotics use the popular PUMA 560 robot in their discussions of kinematics [1] [2] [3] [4] but shy away from a discussion of general structure 6 DOF robots. The PUMA 560 is attractive because of its simple geometry which includes two parallel joint axes and three axes intersecting at the wrist. The kinematics of this robot is solvable in closed form [5]. Its work space is well defined and the number of kinematics solutions at every point inside its work envelope is always eight. Such robots however are in no way descriptive of general robot architectures or general internal workspace topology. A general architecture robot may have only two solutions for a given pose and sixteen solutions at another pose [6].

Computer assisted robotics education has recently found a powerful tool with the recent introduction of a robotics toolbox for Matlab [7] made available, free of charge, on the Internet. The inverse kinematics portion of the toolbox consists of an iterative solution that computes one inverse kinematics solution. The toolbox requires the commercial software Matlab. The software package discussed here, KAP (Kinematics Analysis Program), requires only a computer running DOS and no other mathematical packages such as Matlab or Mathcad.

KAP originated from research in the past few years and was developed to assist in an undergraduate robotics class offered at the University of West Florida. Some of the problems in workspace topology and in path planning, associated with general structure robots, can be tackled by undergraduate students using KAP. For example, planning a path from an end-effector pose achievable by twelve robot configurations to one that has only two. Do the 12 paths originating from the starting pose merge at certain points in space into two paths ending at the destination pose?

Kinematics of Robot Manipulators

A robot manipulator is mathematically modeled by a set of DH-parameters as shown in Table 1. From the four parameters di, ai, αi, θi, associated with joint i, a homogeneous matrix is obtained as

$$A_i = \begin{bmatrix} C_i & -\ {}_iS_i & {}_iC_i & a_iS_i \\ S_i & {}_iC_i & -\ {}_iS_i & a_iC_i \\ 0 & {}_i & {}_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

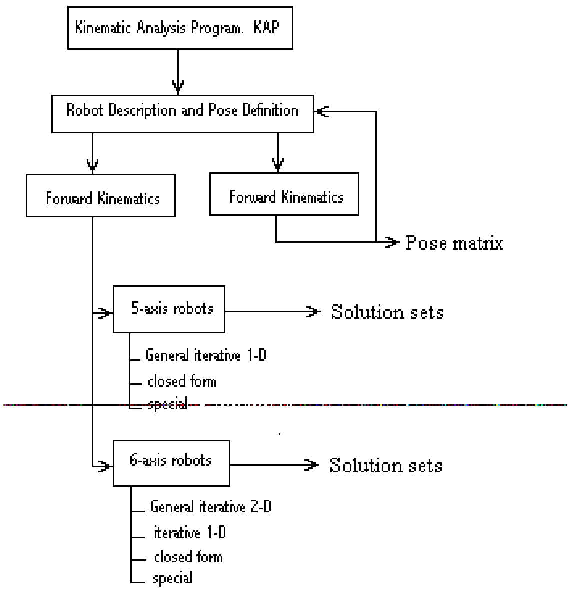

Figure 1. Functional Block diagram of KAP

with $C_i = \cos(\ _i)$, $S_i = \sin(\ _i)$, $\ _i = \cos(\ _i)$, $\ _i = \sin(\ _i)$. $\theta = (\ _1, \ _2, \ _3, \ _4, \ _5, \ _6)$ is the joint variable vector. The position and orientation of the robot end-effector is likewise described by a 4X4 matrix P:

$$P = \begin{bmatrix} n_x & b_x & t_x & p_x \\ n_y & b_y & t_y & p_y \\ n_z & b_z & t_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} n & b & t & p \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

.

The kinematics equation for a 6 DOF robot is then given by

$$A_1 A_2 A_3 A_4 A_5 A_6 = P. \tag{1}$$

The forward kinematics problem consists of computing P from **θ**, while the inverse kinematics consists of solving for θ from eq. (1). A more compact representation for equation 1 is

$$P = K(\ ) \tag{2}$$

for the direct or forward kinematics and

$$= K^{-1}(P) \tag{3}$$

for the inverse kinematics.

No closed form solution for Eq. (3) exists for the general 5- or 6-DOF robot arm, although several numerical techniques have been proposed [8]-[16]. The software described here provides assistance to educators and students in implementing Eq. (2) and three for general geometry 5- or 6-DOF robots. The inverse kinematics part of the software is based on algorithms described in [14]-[16].

Kinematic Analysis Program (KAP)

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 1. A brief description of each section follows.

Robot and pose definition

In this section, the user is asked to enter the Denavit-Hartenberg [7] 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 1. The end effector 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 DOFs of the robot and applies the proper algorithm.

Table 1- Robot DH-parameter table

6 Joints	d	a	α	θ
1	d1	a1	α1	θ1
2	d2	a2	α2	θ2
3	d3	a3	α3	θ3
4	d4	a4	α4	θ4
5	d5	a5	α5	θ5
6	d6	a6	α6	θ6

Five-DOF robots. The most complex 5-DOF robots are those that have no intersecting or parallel joint axes. Yet, they can be solved by a one dimensional Newton-Raphson technique described in [14] and implemented in KAP. 5R robots which have at least one intersecting or parallel pair are solvable in closed from [14]. KAP applies the proper algorithm to solve the robot.

Six DOF robots. Solving 6R robot manipulators of general architecture has been the subject of much research and no closed form solution exists for those robots with no intersecting or parallel pairs. Most inverse kinematic techniques described in the litterature use numerical methods that rely on the robot inverse

Jacobian to solve a system of six non linear equations in the six joint angles [9], [10]. The inverse kinematics function provided in the robotics toolbox for example is an inverse Jacobian method. Because of its mathematical complexity, solving 6-DOF robots is where KAP is most useful. The technique used in KAP is fully described in [16]. The inverse kinematics problem is reduced to a system of two non linear equations in only two of the robot joint variables. Once two joint values are known, the remaining four joint values are determined in closed form.

For 6R robots with a parallel or intersecting pair: The kinematics problem reduces to a single non-linear equation in one joint variable. The remaining joint variables can then be determined in closed form [16].

The closed-form robots class includes all robots with at least three parallel or intersecting joint axes. Equation (1) is then solvable in closed form [5]. The inverse kinematics problem reduces to solving a quartic polynomial equation in one of the joint variable.

The special robots category includes robots that have architectures similar to industry specific robots such as the PUMA560, the GP66, the Stanford manipulator, or the T3 robot.

Inverse kinematics solution search

In real time robot control, the inverse kinematics algorithm must be able to compute a solution as fast as possible to allow for fast robot control. In education however, speed of computation is not essential. Kap is based on numerical algorithms that can iterate to a single solution in a few milliseconds with a proper initial estimate. To search for solutions, KAP uses a sequence of equally spaced estimates. Each time a solutions is found it is checked to determine if it is indeed a solution through forward kinematics and checked against a list of previously found solutions. The program stops if it reaches 16 solutions which is the maximum configuration that a 6R manipulator can have [6]. While there is no guarantee that KAP will find all solutions, it has done so in all experimental cases so far. A solution search will typically yield several solutions in a few minutes.

Illustrative Example

Kap is most useful in the kinematic study of 6R robots of general architecture. In this example, students are asked to determine at least four paths from an initial pose P_i to a final pose P_f shown below for the general geometry robot modeled by Table 2 and drawn approximately on Figure 2. A minimum of 9 intermediate poses had to be determined with as many robot configuration for each pose as can be found by using KAP. The robot is given by the DH-parameters in Table 2.

Table 2

Joint	d	a	alfa	theta
1	0.25	1	65	x
2	0.5	1.5	90	x
3	0.25	1	130	x
4	0.5	2	85	x
5	0.5	0.5	90	x
6	0	0	0	x

$$P_i = \begin{bmatrix} 0.4398 & -0.4484 & 0.7781 & 5.72 \\ 0.1338 & -0.824 & -0.5505 & 0.7844 \\ 0.888 & 0.3463 & -0.3024 & 1.6511 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$P_f = \begin{bmatrix} 1 & 0 & 0 & -3 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

For the sake of brevity only solutions found by KAP for the initial and final poses are shown.

For P_i:

Angle:	θ1	θ2	θ3	θ4	θ5	θ6
1	9.97	20.02	29.95	40.01	49.99	60.02
2	18.84	15.06	43.00	37.76	53.94	54.29

For P_f:

	θ1	θ2	θ3	θ4	θ5	θ6
1	69.65	124.14	-89.69	2.04	-24.94	-157.60
2	178.99	-81.36	46.45	-109.59	50.55	165.70
3	-150.44	3.45	161.62	111.41	-154.87	20.85
4	-18.22	-151.75	49.52	100.99	12.31	-115.06

Conclusion

A software program that offers computer assistance in teaching robot manipulator kinematics, KAP, was described and discussed. KAP will be available soon on the world wide web at the University of West Florida, Electrical engineering department site. The software is able to search for inverse kinematics solutions of 5 or 6 axis robot manipulators of general geometry. It allows robotics instructors to present examples and provide assignments on robot kinematics for such robots. While, KAP has been used effectively in an undergraduate robotics class at the University of West Florida, it can also be useful to graduate students in advanced studies of workspace topology, path planning and trajectory singularities.

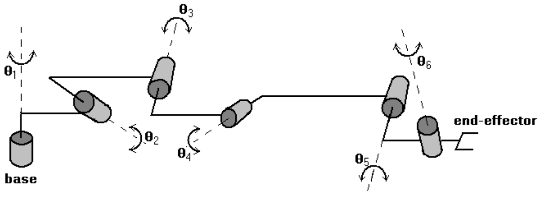

Figure 2: General architecture 6R robot manipulator

References

1. Paul, R. P., "Robot Manipulators," M.I.T. Press, 1992.
2. .Craig, J. J., "Introduction to Robotics - Mechanics and Control, 2nd Edition," Eddison-Wesley, 1989.
3. Spong, M. W & Vidyasagar, M., "Robot Dynamics & Control," John Wiley & Sons, 1989.
4. McKerrow, P. J., "Introduction to Robotics," Addison-Wesley, 1991.
5. Pieper, D. L., “The kinematics of manipulators under computer control”. Ph.D. Diss., Stanford University, 1969.
6. Manseur, R. & Doty, K. L., "A Robot Manipulator with 16 Real Inverse Kinematic Solution Sets," Int. J. of Robotics Research. Vol. 8, Number 5, pages 75-79. 1989.
7. Corke, P. I., "A robotics toolbox for MATLAB," IEEE robotics & Automation Magazine, March 1996.
8. Denavit, J. & Hartenberg, R. S., Trans. ASME Jl. Appl. Mech. Vol. 77, pp. 215-221, 1955.
9. Angeles, “On the numerical solution to the inverse kinematics problem,” Int. J. of Robotics Research, Vol. 4, No. 2, pp. 21-37, 1985.
10. Goldenberg, A. & Lawrence, D. L., “A generalized solution to the inverse kinematics of robotic manipulators,” IEEE Trans. on Dynamic Systems Measurement and Control, Vol. 107, pp. 103-106, 1985.
11. Grudic, Z. & Lawrence, P. D., “Iterative inverse kinematics with manipulator configuration control.” IEEE Trans. on Robotics and Automation, vol. 9, No. 4, pp. 476-483, 1994.
12. Tsai, W. & Morgan, A. P., “Solving the kinematics of the most general six-and five-degree of freedom manipulators by continuation methods.” ASME Paper 84-DET-20. ASME Design Engineering Technical Conference, 1984.
13. Manocha & Canny, J. F., “Efficient inverse kinematics for general 6R manipulators,” IEEE Trans. on Robotics and Automation, vol. 10, No. 5, pp. 648-657, 1994.
14. Manseur, R. & Doty, K. L., "Fast Inverse Kinematics of Five-Revolute-axis Robot Manipulators." Mechanism and Machine Theory. Vol. 27, No. 5, pp. 587-598. 1992.
15. Manseur, R. & Doty, K. L., "A fast algorithm for inverse kinematic analysis of robot manipulators." Int. J. of Robotics Research, Vol. 7, No. 3, pp. 52-63, 1988.
16. Manseur, R. & Doty, K. L., "Structural kinematics of 6-revolute axis robot manipulators," Mechanism and Machine Theory. In press.