2009 World Congress on Computer Science and Information Engineering

Shape and Size Computation of Planar Robot Workspace

Yi Cao[1, 2], Suiping Qi[1], Ke Lu[2], Yi Zang[2], Guanying Yang[2]
[1] *Henan Research Center of Automatic Engineering Technology, Henan Academy of Sciences*
[2] *School of Electrical Engineering, Henan University of Technology, Zhengzhou, 450007, P.R.China*

Abstract

The workspace of a robot is an important criterion in evaluating manipulator geometry and kinematic performance. A systematical approach was presented in the paper to computing the shape and size of the workspace of planar robot manipulator. Firstly, a novel algorism based on the Monte Carlo method was proposed to depict the boundary curves of robot workspace. Then, two methods for calculating area of the workspace were presented. One method based on numerical integration algorithm on closed curves. And the second method, the area can be estimated by probability and statistics. A planar robot manipulator consisting of 3 revolute joints was used as an example to illustrate the systematical approach.

1. Introduction

In performing a task, a robot manipulator has to reach a number of work pieces or fixtures. Workspace is the set of all the points in space which can be reached by the end-effort. Workspace is also named as work volume or work envelope. The size and shape of the workspace depends not only on the coordinate geometry of the robot manipulator and the end-effector, but also on the number of degrees of freedom.

The workspace of a robot is an important criterion in evaluating manipulator geometry and kinematic performance. The first efforts to compute the manipulator workspace, based on its kinematic geometry, started in the mid 1970's [1]. The workspace of robot manipulator has been studied for more than two decades and many methods have been proposed, among which are geometric analysis [2][3], random search by the Monte Carlo method [4] and polynomial discriminant [5]. In this paper, we develop two numerical algorithms – the Monte Carlo method , the probability and statistics method– to obtain the shape and size of the reachable spaces of two robot manipulators.

In the paper, a planar robot manipulator consisting of 3 revolute (3R) joints was illustrated as an example. For a three dimensional (3D) robot manipulator, if the first joint is fixed, a single slice or profile of workspace would be obtained. So although planar robot is not commonly used in practical engineering, the shape and size of its workspace could provide some useful information for the design of 3D robot manipulator.

Firstly, a novel algorithm based on Monte Carlo method was proposed to depict the boundary curves of the robot manipulator workspace, according to the kinematics mapping relationship from the joints space to the workspace. Then, two methods for calculating area of the workspace were presented. One method based on numerical integration algorism on closed curves. And the second method, the area can be estimated as the ratio of number of random points that are contained within the workspace to the total number of random points generated, multiplied by the volume of the box. Finally, the disadvantages of the systematical approach were discussed and an improvement idea was proposed in the conclusion.

2. Mechanism of one planar robot

A planar robot manipulator with three links and three revolute joints is shown in Figure 1. The kinematics of the robot manipulator can be obtained in the coordinates *OXY* by Denavit–Hartenberg[6].

$$\begin{cases} x = l_1 c_1 + l_2 c_{12} + l_3 c_{123} \\ y = l_1 s_1 + l_2 s_{12} + l_3 s_{123} \end{cases} \quad (1)$$

where c and s denote cosine and sine, respectively, c_{12} is $\cos(\theta_1 + \theta_2)$ in short.

978-0-7695-3507-4/08 $25.00 © 2008 IEEE
DOI 10.1109/CSIE.2009.266

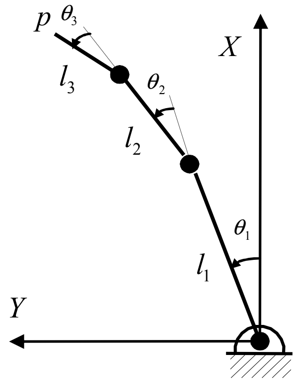

Figure 1. An example of 3R planar robot

Equation (1) can be thought as a mapping from joint variable to task workspace. Since the lengths of robotic links may vary among zero and infinity, and can be measured by different unit system, such as meter and foot, it is convenient to eliminate physical sizes of the mechanism by adopting the ratio of all the link lengths. Therefore, l_1 =4, l_2 =2, l_3 =1.

Most robot manipulators have limits on the range of rotational angle. For this example, the joint limits are: $-2\pi/3 \le \theta_1 \le 2\pi/3$, $-\pi/3 \le \theta_1 \le \pi/3$, respectively.

3. Generating approximate workspace

The Monte Carlo method is a numerical method for solving mathematical problems by means of random sampling. Since the method- a kind of numerical method indeed, involving no inverse Jacobian calculation, is relatively simple to apply [4]. In this paper, the Monte Carlo method, which further developed the method proposed in [4] is used to determine the workspace of robot manipulators.

3.1. Generating workspace of Monte Carlo points

The Monte Carlo method of random sampling is applied to the joint space of the robot manipulator to approximate the workspace. For each uniformly random set of variables in the joint space, there is a unique position in the workspace of robot's end-effector, according to its forward kinematics. For the robot example, according to (1), the computer yields a quantity of sets of the three joint variables (θ_1, θ_2, θ_3) randomly in their rotational range and generates corresponding samples in the workspace. The values of three variables are used to calculate the position of the end-effector P with forward kinematics and a dot is plotted at that position for each sample. Figure 2 shows the workspace of the example with points cloud. A large number of points are required for getting good results in subsequent steps of the method.

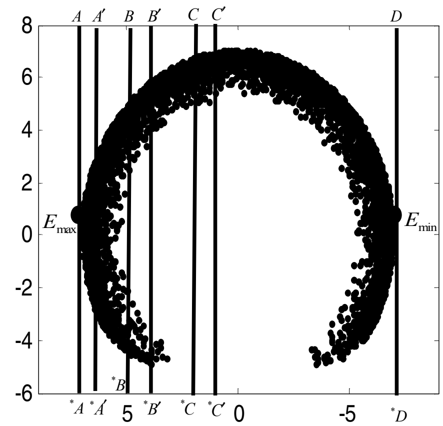

Figure 2. Distribution of Monte Carlo points

3.2. Depicting boundary curve

To acquire the boundary curve, the workspace is partitioned into a series of equally spaced rows as illustrated in Figure 2. Through this process, the 2D problem is reduced into searching boundary points in 1D, i.e. a line. For each element (row) having at least one Monte Carlo point, the boundary points can be obtained by the following steps:

(1) Search for the extreme values y_{max} , y_{min} along Y-axis and corresponding points E_{max}, E_{min}. Supposing $y_L = |y_{max} - y_{min}|$, the width of each row is $\xi = y_L / m$, where m represents the number of rows determined by the required precision to display.

(2) According to the y-coordinate values, distribute Monte Carlo points into corresponding rows.

(3) Search the boundary points in each row along X-axis. The method to search the boundary points may be different from one row to another. For example in Figure 3, it is only need to find the maximum and minimum points (i.e. A_1, A_2) in the row between the lines A^*A and A'^*A' . However, there are some difficulties in searching the boundary points in the row between the lines B^*B and B'^*B' . It is readily to find the extreme points B_1 and B_4 as above, and then sorting the points by their x-coordinate values. Calculate the difference of x-coordinate value between two adjacent points one by one from B_1. If the difference is much larger than the Row Width ξ (such as 10 times), it indicates there is a *holl* or *void* in this row, and the two

points (B_2 and B_3) are on boundary. The rest may be deduced by analogy, all the boundary points of the planar workspace can be obtained.

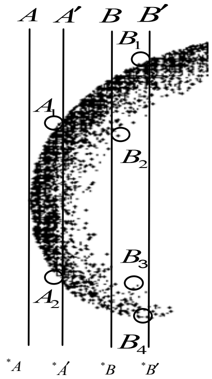

Figure 3. Depiction of Boundary points

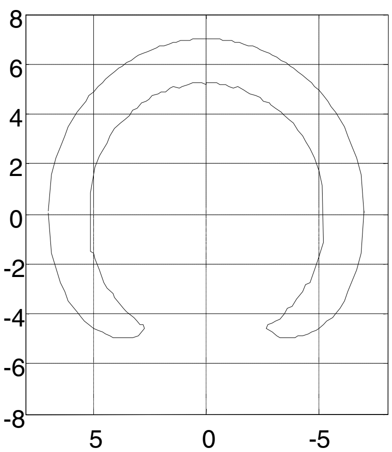

Figure 4. Boundary curve of the workspace

(4) Construct the boundary curve by connecting the boundary points. The nearest points should be searched one by one from $E_{\max}$, and connected by straight line segments, until all the points composing a closed curve (polygon indeed).

By the four steps, the boundary curve in the main working plane can be obtained and presented graphically in Figure 4.

4. Calculating the Area of the Workspace

4.1Numerical integration method

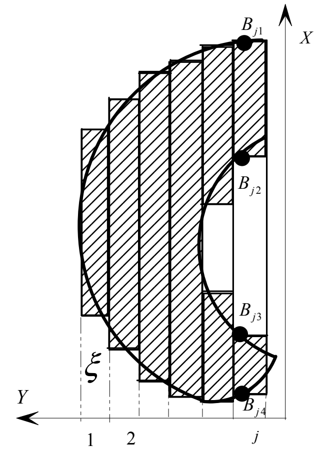

Figure 5. Area approximation with rectangles

As shown in Figure 5, the workspace has been partitioned into a number of equally spaced rows, which could be viewed as a series of narrow rectangles. Figure 6, which depict the rectangle of row B^*B, consisting of two sub-rectangles, is the local magnified figure of Figure 5. The area of the workspace can be calculated by these rectangles. Firstly, we should get the area of each row. For the example of row B^*B, the width of the rectangles is ξ, the height of each sub-rectangle can be obtained by calculating the absolute value of the difference between B_1, B_2, and B_3, B_4, along X-axis.

The area of each row, number j for example, can be represented as

$$A_j = \xi \cdot h_j = \xi \cdot [(x_{j1} - x_{j2}) + (x_{j3} - x_{j4}) + \cdots + (x_{j,i-1} - x_{j,i})] = \xi \sum_{k=1}^{i/2} (x_{j,2k-1} - x_{j,2k}) \quad (2)$$

Thus, because the workspace consist of rows, the area of the workspace A_w, can be calculated by the following equation:

$$A_w = A_1 + A_2 + \cdots + A_n = \sum_{j=1}^{m} A_j \quad (3)$$

Substituting (2) into (3) yields:

$$A_w = \sum_{j=1}^{n} A_j = \xi \cdot \sum_{j=1}^{m} \sum_{k=1}^{i/2} (x_{j,2k-1} - x_{j,2k}) \quad (4)$$

For the 3R robot example, the averaging size of the workspace area is $A_w = 56.236$ after 10 times running with (4), under the condition that random points are 50,000, the partitioned rows are 40.

4.2. Probability and statistics method

4.2.1. Preliminary

In the paper, we proposed another algorithm to evaluate the area by probability and statistics based on the Monte Carlo method again.

Let *S* is an area that may be geometrically complex (workspace), *B* be a simple area (e.g., a rectangle) that encloses *S*. To estimate the area size we apply the Monte Carlo method and obtain

$$I_s = C_B \cdot \Sigma' f(Q_i) / N \quad (5)$$

where $Q_i = (x_i, y_i)$ are random points independently of Euclidean 2-space (E^2)and uniformly distributed over S, C_B is the area of B , N is the number of Monte Carlo points and the prime indicates that the sum ranges only over the Q_i that belong to *S*.

Observe that all of the quantities on the right side of (5) either are known or easy to compute if one can distinguish computationally between the points Q_i which are in (or on) *S* from those which are outside *S*, i.e., if one can evaluate the characteristic function of *S*. The technique which to determine whether or not a point lies inside or outside a closed 2D workspace is ray method. An early description of the problem in computer graphics shows two common approaches (ray casting and angle summation) in use as early as 1974 [7].

Good (i.e., tight) bounds on the errors associated with (5) can be derived by standard statistical analysis [8]. If round-off errors are ignored, *Is* converges to the correct value *I* when *N* increases, but the convergence is slow; the standard deviation of the errors is inversely proportional to the square root of *N*. The existence of good error bounds and the general applicability of Monte Carlo methods are the methods' advantages.

4.2.2. Steps of area calculation

There are 3 important steps to compute the area.

① Find the envelope rectangle containing the shape of the workspace.

Search for the extreme values Y_{max}, Y_{min} and X_{max}, X_{min} along *Y*-axis and along *X*-axis in the boundary points of the workspace, which had depicted in the section 3.2. There are four points corresponding to the four values, respectively. We could draw four line segments on the fours points along X-axis or Y-axis, which construct a rectangle, shown in Figure 6. This is an envelope rectangle containing the workspaces.

② Calculate the rectangle area C_B

$$C_B = (X_{max} - X_{min})(Y_{max} - Y_{min})$$

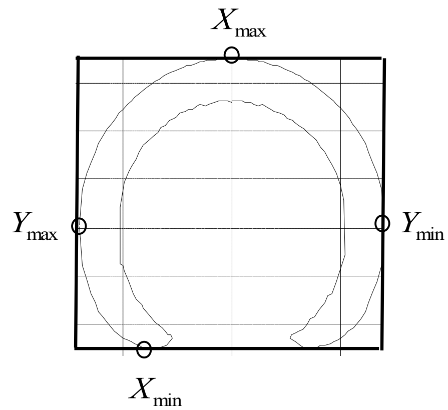

Figure 6. Envelope rectangle of the workspace

③ Estimate the area of workspace.

Once again, the computer yields a number of *N* points uniform randomly in their ranges denoted in (6), for N=1000 points as example.

$$\begin{gathered} Q_i(x_r, y_r), \\ x_r \in [X_{min}, X_{max}], y_r \in [Y_{min}, Y_{max}] \end{gathered} \quad (6)$$

Then we should distinguish how many of the N points in the cooperative workspace. The method to obtain the information is ray method which had explained in the formal section. Suppose that there are M ($0 \le M \le N$) points in the cooperative workspace after computing. From (5), the area of the workspace is

$$C = C_B \frac{M}{N} \quad (7)$$

As we can see, the method is very simple to apply. Although the accuracy of the method is not very high, i.e. the standard deviation of the errors is inversely proportional to the square root of *N*, it is suitable for the engineering design and practice. The averaging area size of the workspace is about 55.83 after 10 times running with 1000 random points.

5. Conclusion

A systematical approach was proposed to generate the shape of the workspace of the planar robot manipulator and calculate the corresponding area.

There is a main problem in the method which causes errors in depicting the boundary curve and calculating the area. It is related to applying the Monte Carlo method. One disadvantage in using the Monte Carlo method is that most of the points it yields are within the workspace boundary instead of on the boundary. Therefore, the participation process can not depict exact boundary points but only approximate boundary. Because the second method to compute the workspace area is based on the Monte Carole and statistics, the result is stochastic and depends on the number of uniformly distribute points. One solution for finding the exact boundary point locations is to combine the Monte Carlo method with an optimization method.

The approach works well for most manipulators, especially for the manipulator workspace with cave or hole. The method provides a powerful and simple tool for the design and evaluation of robot manipulators from their shape and size. The work which can exactly depict the boundary points of workspace based on combining the Monte Carlo method with an optimization method would be published in future.

Acknowledgments

This work was supported in part by National Key Technology R&D Program under Grant No.2008BADA8B04, by Basic and Frontier Program of Henan Province under Grant No.072300440130, and by Natural Science Foundation of the Education Department of Henan Province under Grant No.2007460003.

References

[1] B. Roth, “Performance Evaluation of Manipulators from a Kinematics Viewpoint”, *NBS Special Publication*, 1975. pp. 39-61.

[2] K. Gupta, “On the nature of robot workspace”, *International Journal of Robotics Research*, 1986,5(2), pp. 112–121.

[3] Z. Lai, C. Meng, “The dexterous workspace of simple manipulators”, *IEEE Transactions on Robotics and Automation*, 1988, 4(1), pp.99–103.

[4] G.A David, C.D Chung. “Determining manipulator workspace boundaries using the Monte Carlo method and least squares segmentation.” ASME, Design Engineering Division, 1994, 72, pp:141-146.

[5] D. Kohli, J. Spanos, “Workspace analysis of mechanical manipulators using polynomial discriminants.” *ASME Journal of Mechanisms, Transmissions, and Automation in Design*, 1985, 107, pp. 209–215.

[6] R.. P. Paul, *Robot manipulators: mathematics, programming, and control : the computer control of robot manipulators*. Cambridge, Mass. : MIT Press, c1981.

[7] I. E. Sutherland, R. F. Sproull, R. A. Schumacker, “A Characterization of Ten Hidden-Surface Algorithms”, *Computing Surveys*, 1974,6(1), pp.1-55

[8] P. J. Davis, P. Rabinowitz, *Methods of Numerical Integration*. Academic, New York, 1975.