

Design of Circular Trajectory Forming Algorithm for a Manipulator

Zhao Rongrong^{1,a}, Zhao Shuai^{2,b} and Li Yufeng¹

¹ School of Mechanical Engineering, Shandong Huayu University of Technology, Dezhou 253034, China

² Department of Wheel Loader, Degong Machinery CO.,LTD, Dezhou 253000, China

^a qihezhaorongrong@163.com, ^b shuairesume2015@163.com

Keywords: circular trajectory, solver, algorithm, manipulator.

Abstract: With the rapid development of science and technology, manipulators are becoming more and more widely used in various industries. When using the manipulator to achieve various trajectories, it is necessary to calculate the coordinates of the key track points, especially the circular trajectory. For many manipulators, the process of achieving a circular trajectory is going to approximating the circle by a point-to-point line. Therefore, the design of the circular trajectory solver own significant application value. Based on circular trajectory forming algorithm, the circular trajectory solver is designed using an easy language. The application of solver will greatly improve the efficiency of circular trajectories such as manipulators.

1. Introduction

The progress of science and technology has provided strong technical support for the development of enterprises, especially in the industrial field, and the development of advanced equipment such as manipulator have greatly improved the production efficiency of enterprises[1]. However, when the movement of manipulator is programmed, some supporting software defects restrict the efficiency of programming, such as KUKA robot developed by KUKA, Germany. It uses the KUKA's off-line programming software, and the function of software is to automatically compile the path trajectory designed by KUKA robot simulation software[2]. But when using the KUKA's software for actual machining path planning, the software can not automatically realize the circular trajectory, can only achieve a straight path from point-to-point, so a circular trajectory is generated in the offline state, it must through the point-to-point straight line to approach the circular trajectory, and automatically translated into motion of the manipulator.

When using point-to-point straight line to approach circular trajectory, it is necessary to calculate the coordinates of the trajectory points on the trajectory path to ensure that the actual running route of the manipulator meets the user's requirement. But when calculating the trajectory point, simple manual calculation consumes both time and brain power. If you can follow the trajectory requirement, a few key coordinates of the trajectory path are input into a solver and the coordinates of trajectory points can be calculated automatically by the solver, which greatly saves programming time. This paper is based on this purpose to study the forming algorithm of circular trajectory and the design of the solver based on the algorithm.

2. Analysis of Circular Trajectory Forming Algorithm

Now there are many algorithms for generating circular trajectories, which are frequently seen with midpoint generation algorithm and the generation algorithm based on the Bresenham circle[2]. There is a disadvantage of these algorithms that the selection of the path points of the circle and the uneven distribution of the lines between the points can cause the generated circles to be very irregular in the process of circular formation. The algorithm that using the inner polygon to approximate the circular can ameliorate the disadvantages of these algorithms[3]. This method is to approach the circle through the positive polygon of the rule, as long as the increasing number of regular polygon edge can be infinitely close to the circular to make the circular beauty symmetry.

The principle of the algorithm shown in Figure 1.

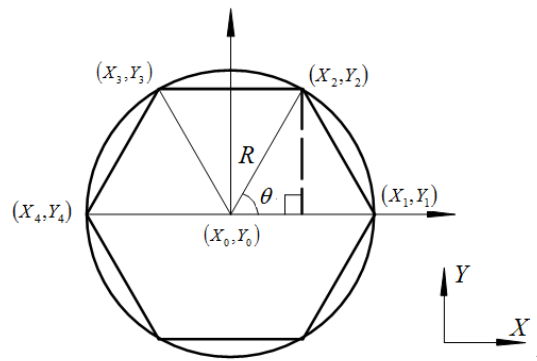


Figure 1 The principle diagram of the algorithm

Assumes that the center coordinate of circular is X_0, Y_0 and the circle is divided into m equal parts by the positive polygon, so the corresponding angle θ is $2\pi/m$. The starting point X_1, Y_1 of the regular polygon is the intersection of the ray and the circular that starting from the center of the circular and paralleling to the X -axis. Counterclockwise is the positive direction of the circular and the intersection of the positive and the circle is X_1, Y_1 ; X_2, Y_2 ; X_3, Y_3 ; X_m, Y_m . Seen from Figure 1, the vertex coordinates of the positive polygon[4] are:

$$X_1 = X_0 + R \qquad Y_1 = Y_0 \qquad (1)$$

$$X_2 = X_0 + R * \cos\left[\frac{2\pi}{m}\right] \qquad Y_2 = Y_0 + R * \sin\left[\frac{2\pi}{m}\right] \qquad (2)$$

$$X_3 = X_0 + R * \cos\left[\frac{4\pi}{m}\right] \qquad Y_3 = Y_0 + R * \sin\left[\frac{4\pi}{m}\right] \qquad (3)$$

$$X_4 = X_0 + R * \cos\left[\frac{6\pi}{m}\right] \qquad Y_4 = Y_0 + R * \sin\left[\frac{6\pi}{m}\right] \qquad (4)$$

By observing all of the above and according to the recursive method, the formula of the vertex calculation of the normal polygon is:

$$X_n = X_0 + R * \cos\left[\frac{2(n-1)\pi}{m}\right] \qquad n=1, 2, 3, \dots, m \qquad (5)$$

$$Y_n = Y_0 + R * \sin\left[\frac{2(n-1)\pi}{m}\right] \qquad n=1, 2, 3, \dots, m \qquad (6)$$

3. Design of Circular Trajectory Solver

Based on the above formula (5)-(6), it can write the corresponding algorithm program in programming language and build the corresponding visual solver. In many programming languages, Easy Programming Language is the Chinese programming software, and it is a visualized object-oriented programming language[5]. According to formula (5) and formula (6), it can write the language program that automatically solves the vertex coordinate of the positive polygon. The specific requirement is to enter the initial value of X_0, Y_0, R, m . According to formula (1) and formula (2), the coordinates of $X_1, Y_1, X_2, Y_2, X_3, Y_3, \dots$ are solved. By compiling the above solution into an Easy Language Program, the solver of the vertex coordinate of the positive polygon

can be generated.

This article sets the solver icon to a robot image and double-click the icon to enter the solution interface. In the solution interface, first click the clear button to remove the initial data inside the solver, then enter the initial values of X_0, Y_0, R, m in the blank line and finally click the count button to calculate the X_n and Y_n ($n=1,2,3,\dots,m$). The specific solution process is shown in Figure 2.

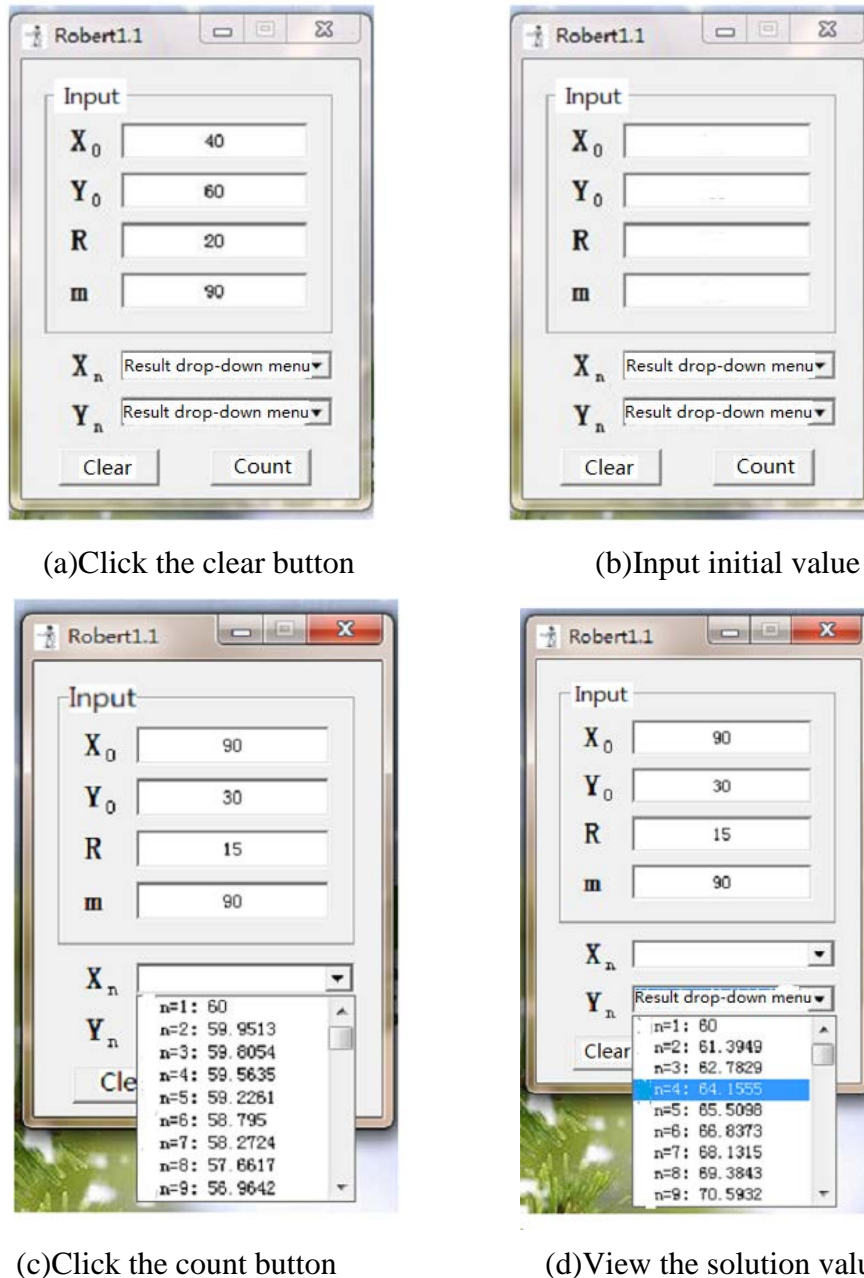


Figure 2 Flow diagram of solving coordinates

4. Conclusions

In this paper, solving the manipulator circular trajectory point coordinates is designed by analyzing the circular trajectory, and the design of circular trajectory solver is carried out. Maniplator is used more and more widely in industry and it can improve production efficiency while reducing manpower. The solver is applied to the offline programming of the manipulator circular trajectory, which greatly improves the programming efficiency and own a significant application value.

References

- [1] Wei H. Z, Xue D, Bai W. F. (2012) Based on the Error Analysis And Simulation of KUKA 6 Freedom Robot. *Journal of Changchun University of Technology*, 3, 328.
- [2] Sun J. G. (1998) *Computer Graphics*. Tsinghua university press.
- [3] Zhang B. (2011) Generate a New Algorithm for the Positive Polygon and the Circle. *Journal of Engineering Drawing*, 2, 1.
- [4] Zhao S. (2014) Study on Waterway Structural Improvement of 200L Material Barrel's Bottom Mould and the Forming Technology of the Waterway by Laser Cladding [D]. Liaoning University of Technology.
- [5] Zhang J Y, Dai L. H, Wang X. L. (2012) Study on the Database of Remanufacturing Overburden Machining Based on Easy Programming Language. *Computer Science*, 3, 195-196.