Three-wheel full-wheel robot speed algorithm

Qi Wang1,*, DaZhang You1,b, Jing Yang1,e
1College of Mechanical Engineering, Hubei University of Technology, Wuhan 430068, China
*Corresponding author e-mail:yxwq6318@163.com,
yodazhag@163.com, 857679682@qq.com

Abstract. This paper introduces the design of the chassis speed algorithm for the Robocon2018 wheeled robot, and designs the motion algorithm of the wheel chassis for the Robocon2018 competition. Establish the coordinate system, through to the robot Chassis movement modelling analysis and based on the actual debugging situation, according to the decomposition and synthesis of vector velocity, the velocity projection value of three omnidirectional wheels is calculated by means of vector dot product, and then the velocity projection value is converted to STM32 PWM output to control the motor. It also proves the effectiveness and feasibility of the velocity algorithm to make the robot move according to the predetermined speed.

1. Introduction

In Robocon competition, the speed algorithm of robot is of great significance to the overall design. The flexibility and high efficiency of the motion algorithm of robot are of great importance to the improvement of the overall performance of robot speed and stability. In the actual design, the motion algorithm of the mobile robot is unstable, which makes the speed and direction control of the robot inconvenient in the actual movement process. It becomes very difficult to achieve the ideal motion effect.

In the control of the motion algorithm designed in this paper, the velocity of the moving direction is projected to the direction of three omnidirectional wheels according to the given velocity and angle according to the position relation of the three-wheeled omni-directional wheel robot, which is based on the kinematic modeling. The velocity projection value is converted to the output of the STM32 PWM on the basis of the driving requirement to control the motor, thus stabilizing the motion effect of the wheeled omnidirectional robot.

2. Block Diagram of Three-Wheel Omnidirectional Wheel Motion Algorithm

In the process of speed algorithm implementation, the robot's three omni-directional wheel need install radial symmetry, omni-directional wheel and steering gear axis vertical, each wheel speed the Angle between the direction of 120°, the wheel is the same distance machine center of gravity.
The model configuration of each drive should be exactly the same, and the quality shape of each wheel should be exactly the same.

Given the size and angle of the target velocity, the velocity projection value in the direction of each omnidirectional wheel is obtained by means of vector dot product between the omnidirectional wheel coordinates set in the algorithm and the velocity vector coordinates of the target. It is stored in an array.

Figure 1. STM32 hardware design

3. Three-wheel full-wheel robot motion modeling
3.1 Modeling and analysis of velocity vector graph

The vector coordinates of the three omnibearing wheels are shown in figure 1, which are respectively (100,0);(-100,173);(-100, 173) between each vector angle is 120 °.

The coordinates of the three velocity vectors are stored in a static array. As shown in figure 2, the velocity vector with a given coordinate value of (X,Y) moves along the target speed and direction in the process of motion.

In figure 1, (X,Y) represents the projection value of the velocity vector onto the X and Y axes of the coordinate system, and represents the angle between the robot and the X axis of the coordinate system.

Figure 2. velocity vector modeling diagram
3.2 Speed direction analysis diagram of single wheel

3.2.1 Projection value of target velocity in the direction of (100,0)

As shown in figure 3, the projection value of the target velocity in the direction of (100,0) is actually equivalent to the projection value in the direction of the square up of the X-axis. The projection value of the calculated velocity in the direction of (100,0) should be the same as the projection on the X-axis and can be used for verification.

3.2.2 Projection value of target velocity in the direction of (-100,173)

As shown in figure 4, L1 is the projection of the velocity vector on the default vector (-100,173).
3.2.3 projection value of target velocity in the direction of (-100,-173)

As shown in figure 5, L1 is the projection of the velocity vector on the default vector (-100,-173).

\[ \text{Figure 6. Projection of target velocity in the direction of (-100,-173)} \]

4. The motion realization of vector dot product motion algorithm in STM32.

The core of the motion algorithm of vector dot product is to calculate the velocity value of three omnidirectional wheels by input velocity vector (including two variables of velocity value and velocity direction).

4.1 Program flow chart

\[ \text{Figure 7. Procedure flow chart} \]

4.2 Store the creation of an array

First, as shown in figure 6, set up a one-dimensional array float motor_tem[3] with a capacity of 3 in advance. After storing vector dot product algorithm, the velocity values of L0, L1, L2 on three omnidirectional wheels and an array with the same capacity of 3 float motor_load[3] are obtained to store the filling value of the driver's duty ratio.;Float dis_motor[3] is used to supplement the rotation effect.
4.3 Velocity projection calculation and drive numerical filling

As shown in figure 7, by calculating and projecting the velocity vector to X and Y rectangular coordinate systems respectively, two variables X and Y are set to store the projection values. Because the different motor drives we use have different configuration modes, we need to check whether X and Y are within the range of normal operation of the drive before calculation, so as to avoid motion distortion.

4.3.1 The projection of the velocity vector onto the default vector (100,0)

As shown in figure 3, L0 is the projection amount of the velocity vector on the default vector (100,0), which was previously stored in the Motor_R structure, so motor_dir[0]. X = 100. Motor_dir [0]. Y = 0, so the motion vector is calculated as follows:

\[
\text{motor}\_\text{tem}[0]=(x*\text{motor}\_\text{dir}[0], x+y*\text{motor}\_\text{dir}[0], y)/(\sqrt{x^2+y^2});
\]

By calculating the value of the obtained L0, the velocity projection value L0 is transformed into an updated array of motor_tem[I] by the formula of the speed value N0. According to the driving requirements, the PWM duty ratio of the target speed is converted into the PWM required by the speed.

4.3.2 The projection of the velocity vector onto the default vector (-100,-173)
Also shown in figure 3, L1 is the projection of the velocity vector on the default vector (-100,-173), motor_dir[1]. Y = -173, calculated by motion vector:

\[ motor_{\text{tem}[1]} = (x \times \text{motor_dir}[1].x + y \times \text{motor_dir}[1].y) / (sqrt(x^2 + y^2)); \]

By calculating the value of L1 obtained, the velocity projection value of L1 is converted into the array motor_tem[I] of the rotating speed value N0 by the formula. According to the driving requirements, the PWM vacancy ratio is converted to the target speed required by the target speed.

4.3.3 The projection of the velocity vector onto the default vector (-100,173)

As shown in figure 4, L2 is the projection amount of velocity vector on the default vector (-100,173), motor_dir[2]. Y = 173, calculated by motion vector:

\[ motor_{\text{tem}[2]} = (x \times \text{motor_dir}[2].x + y \times \text{motor_dir}[2].y) / (sqrt(x^2 + y^2)); \]

By calculating the value of the obtained L2, the velocity projection value of L2 is transformed into the array motor_tem[I] of the rotating speed value N0 through the formula. According to the driving requirements, the PWM duty ratio of the target speed is converted into the PWM required by the driving speed.

As shown in FIG. 7, the whole algorithm is packaged into a function, and the value and direction of speed are exposed as parameters. In the process of using the speed algorithm, the value and direction of speed are constantly modified, so as to output PWM wave forms with different duty ratios.

5. Actual motion diagram

Parameter values as shown in figure 8, the given speed 0.5 m/s, the direction of 90 °, the velocity projection value is in (100, 0); In the (100173) to 0.43 m/s, in (100, 173) to 0.43 m/s, two vector Angle is 120 °. The forward velocity is 0.497 m/s.

![Figure 10. Omnidirectional wheel cart physical map](image-url)
6. Conclusion

Based on the background of Robocon2018 competition, this paper studies and analyzes the realization of the speed of the Omni-wheeled robot, and describes in detail the algorithm principle, velocity vector projection modeling and the realization process on the STM32 MCU. Summed up, mainly for the following three points:

1. The motion model of the round wheel is established, and the angle between the rotational speed vectors of the whole wheel is 120°.
2. The velocity vector is decomposed in three velocity directions, and the projection value in three projection direction is obtained according to the algorithm of vector dot product.
3. According to the requirements of driving parameters, the velocity projection value is converted to the PWM parameter value of the target speed.

References

[1] Han Jianhai. Industrial Robot [M]. Wuhan: Huazhong University of Science and Technology Press, 2014.

[2] X. Gong, L. Wang, X. Cui, et al. Dome A site testing and future plans[C]. An Astronomical Observatory, Frascati, Italy, 2010, 65-72

[3] T. B. Eric. Motion planning technologies for planetary rovers and manipulators[C]. Proceedings of International workshop on motion planning in virtual environments, toulouse, France, 2005, 56-78

[4] Tan Min, Wang Shuo's research progress [J]. Acta Automatica Sinica, 2013

[5] C. K. Ronald. Task Modeling in Collective Robotics[J]. Autonomous Robots, 1997, 4 (1): 53-72

[6] Xia Tian. Study on Structural Design and Operation Control Algorithm of SCARA Robot[D]. Wuhan: Hubei University of Technology, 2016.