Control of Balancing Mobile Robot on a Ball with Fuzzy Self-adjusting PID

Xu Su[1], Cun Wang[1], Weiwei Su[1,2], Yan Ding[1]

1. College of Electronic Information Engineering, Inner Mongolia University, Hohhot 010021, China
E-mail: ballbot@163.com

2. Test Institute of Aerospace Dynamic Machine in Inner Mongolia, Hohhot 010076, China
E-mail: 15947035105@163.com

Abstract: The on-ball balancing mobile robot is a robot which achieves a full range of movement by rolling the ball, it can work flexibility and freedom in narrow areas. In this paper, a dynamic model is established by using Lagrange method to the on-ball balancing mobile robot .On the basis of PID control, a fuzzy self-adjusting PID control method is proposed to control the robot's balance. The simulation results show that the fuzzy self-adjusting PID controller can achieve the balance control of the ball effectively and has the characteristics of fast response and small overshoot.

Key Words: On-ball balancing mobile robot, Dynamic model, Fuzzy self-adjusting PID control

1 INTRODUCTION

The on-ball balancing mobile robot is a robot which achieves a full range of movement by rolling the ball, it can work flexibility and freedom in narrow areas. From the structure, it is a typical self-balancing robot, because there is only a contact point with the ground, cannot maintain static stability, can only be in a dynamic balance.

The dynamic equation is a multivariable, strong coupling, time-varying nonlinear high-order differential equations with complex characteristics. This kind of robot system has the characteristics of multi degree of freedom, multi driver and multi sensors at the same time. It is a very complex nonlinear system with multi input and multi output.

Studying on the on-ball balancing mobile robot starting from abroad, the research is still in its infancy at home now, and it is a relatively new research object in the field of control.

The first on-ball balancing mobile robot was developed by Lauwers et. al [1] at Carnegie Mellon University (CMU) in 2006 which utilizes the mechanism of an inverse mouse-ball drive. By incorporating active balancing control, the robot was able to navigate quickly in any directions despite its high center of gravity. Active balancing control allows the attitude of the robot to be continuously corrected to maintain stability. In 2008, Kumagai and Ochiai [2] introduce several versions of balancing robot which are able to balance itself on a pipe and ball. The literature presents the details of the mechanical design of the robot and experimental result, however information related to balancing and control strategies are not clearly reported.Literature [5] is based on the linear quadratic optimal control, realize the balance control, position control and trajectory tracking control. The literature [6] design the proportional integral controller and linear quadratic optimal controllers to realize the stability and trajectory tracking control of the robot. In this paper, the designed structure of on-ball balancing mobile robot is simple, the dynamics model of robot is established by using the Lagrange method. The fuzzy self-adjusting PID controller is designed according to the model, and verified the effectiveness of the robot's balance control through the simulation.

2 SYSTEM DESCRIPTION

The structure of the on-ball balancing mobile robot is shown in Figure 1, which is composed of the driving body and the driven ball. The driving body is composed of the driving mechanism, the attitude acquisition, the wireless transmission, the battery and the motor drive, etc. the driving mechanism is composed of electric motors and omni-directional wheels.

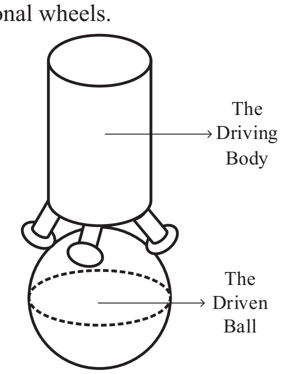

Fig 1. Schematic diagram of the on-ball balancing mobile robot structure

The driving body attitude data acquiring is based on MPU6050 high precision inertial measurement unit (IMU) which produced by InvenSense Company. It integrated the 3-axis MEMS Gyro and 3-axis accelerometer. It can deduce the robot's attitude by detecting the direction angle and angular velocity information of the driving body. Three

This work is supported by the Inner Mongolia University to enhance the comprehensive strength of the project.

driving mechanisms project onto the horizontal plane in 120 degrees, the axial and vertical direction in 45 degrees. The microcontroller sends motion commands to the robot after getting the robot's attitude by MPU6050, and makes the corresponding action by the motor driving the omni-directional wheel, which makes the robot keep the dynamic balance and track the command movement. And the communication between the host computer and the robot can be realized through the wireless module, the status of the robot and the transmission of the motion commands are displayed by the host computer.

3 SYSTEM MODELING

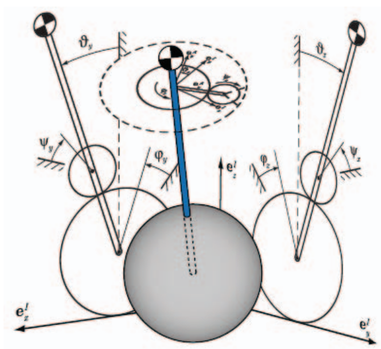

Fig 2. The planar model of on-ball balancing mobile robot

To establish a three-dimensional coordinate system, project the three-dimensional model in Figure 1 onto YOZ, XOZ and XOY plane, get the three plane models as shown in figure 2. The XOZ and YOZ plane models are the same, the YOZ plane model equivalent to the robot translating / rotating along the X axis, the XOZ plane model equivalent to robot translating / rotating in the horizontal plane along the Y axis, and the XOY plane model equivalent to the robot rotating around the body midline. The three plane models are independent of each other, which can be analyzed separately, without considering the coupling between them.

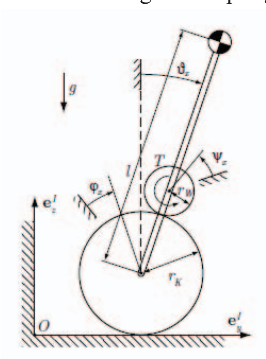

Fig 3. The model of on-ball balancing mobile robot inverted pendulum

Due to the same principle between XOZ and YOZ plane in Figure 2, so the analysis of robot model can only use YOZ, XOY plane, however we do not involve in the rotation of robot in this paper, which do not need pay attention to the XOY plane model, so that it can only needs to analyse YOZ plane model.

As can be seen from Figure 2 that the XOZ/YOZ plane model can actually be equivalent to two identical single inverted pendulum, the inverted pendulum model is shown in Figure 3, achieving the balance of the body by moving left and right at the bottom of the ball. And thus the research of the robot model is transformed into the balance of the single stage inverted pendulum.

The planar model is composed of the ball, the virtual wheel and the body, and each plane has rotation degrees of freedom of the bottom ball and the body. Figure 2 shows the minimum coordinate system and dynamic modeling of the YOZ plane, assume that the dip angle of the body is ϑ_x and driven ball is φ_x, then:

$$x_K = \varphi_x r_K \quad (1)$$

$$x_W = x_K + \sin\vartheta_x \cdot (r_K + r_W) \quad (2)$$

$$x_A = x_K + \sin\vartheta_x \cdot l \quad (3)$$

In the formula, r_K represent the radius of the driven ball ; r_W represent the radius of the omni-directional wheels; l represent the gravity height of the body; andx_K, x_W and x_A are the displacement of the ball wheel, virtual drive wheel and the body in the X axis.

Due to the same speed between the ball and the virtual drive wheels at the contact point, the relationship between the rotation angle of the virtual drive wheel and the minimum coordinate system can be obtained.

$$\dot{\psi}_x = \frac{r_K}{r_W}\left(\dot{\varphi}_x - \dot{\vartheta}_x\right) - \dot{\vartheta}_x \quad (4)$$

$$\dot{\psi}_y = \frac{r_K}{r_W}\left(\dot{\varphi}_y - \dot{\vartheta}_y\right) - \dot{\vartheta}_y \quad (5)$$

$$\dot{\psi}_z = \frac{r_K}{r_W} \cdot \sin\alpha \cdot \left(\dot{\varphi}_z - \dot{\vartheta}_z\right) \quad (6)$$

In the formula, α is the intersection angle between axial and vertical axis of the omni-directional wheel which is zenith angle; and the rotation angles of the intermediate virtual driving wheels are ψ_x, ψ_y and ψ_z.

The kinetic energy and potential energy of each part of the system under each plane are obtained.

For the driven ball:

$$T_K = \frac{1}{2}\mathrm{m}_K(r_K\dot{\varphi}_x)^2 + \frac{1}{2}\mathrm{J}_K\dot{\varphi}_x^2 \quad (7)$$

$$V_K = 0 \quad (8)$$

For the driving wheel:

$$T_W = \frac{1}{2}\mathrm{m}_W(\dot{x}_W)^2 + \frac{1}{2}\mathrm{J}_W\dot{\psi}_x^2 \quad (9)$$

$$V_W = \mathrm{m}_W \cdot g \cdot (r_K + r_W) \cdot \cos\vartheta_x \quad (10)$$

For the body:

$$T_A = \frac{1}{2}\mathrm{m}_A(\dot{x}_A)^2 + \frac{1}{2}\mathrm{J}_A\dot{\vartheta}_x^2 \quad (11)$$

$$V_A = \mathrm{m}_A \cdot g \cdot l \cdot \cos\vartheta_x \quad (12)$$

In the formula, m_K, m_W and m_A are expressed as the quality of the driven ball, the omni-directional wheel and the body respectively; the J_K , the J_W and the J_A are expressed as the moment of inertia of the driven ball, the omni-directional wheel and the main body respectively. Lagrange equation can be expressed as

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}}\right)-\left(\frac{\partial T}{\partial q}\right)+\left(\frac{\partial V}{\partial q}\right)-F_{NP}=0 \qquad (13)$$

In the formula, T and V are represent for the total kinetic energy and total potential energy of the system respectively, and the generalized force is expressed by F_{NP}.

$$\mathrm{T}=T_K+T_W+T_A \qquad (14)$$

$$\mathrm{V}=V_K+V_W+V_A \qquad (15)$$

The dynamic equation of the on-ball balancing mobile robot in the YOZ plane is obtained:

$$M_x(q,\dot{q})\ddot{q}+C_x(q,\dot{q})+G_x(q)=F_{NP} \quad (16)$$

In the formula, $q=[\varphi_x,\vartheta_x]^T$

$$M_x=\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \qquad (17)$$

$$M_{11}=m_{tot}r_K^2+J_K+\left(\frac{r_K}{r_W}\right)^2 J_W \qquad (18)$$

$$M_{12}=-\frac{r_K}{r_W^2}(r_K+r_W)J_W+\gamma r_K\cos\vartheta_x \qquad (19)$$

$$M_{21}=-\frac{r_K}{r_W^2}(r_K+r_W)J_W+\gamma r_K\cos\vartheta_x \qquad (20)$$

$$M_{22}=-\frac{(r_K+r_W)^2}{r_W^2}J_W+J_A$$

$$+m_A l^2+m_W(r_K+r_W)^2 \qquad (21)$$

$$C_x=\begin{bmatrix} -r_K\gamma\sin\vartheta_x\dot{\vartheta}_x^2 \\ 0 \end{bmatrix} \qquad (22)$$

$$G_x=\begin{bmatrix} 0 \\ -g\sin\vartheta_x\gamma \end{bmatrix} \qquad (23)$$

$$m_{tot}=m_K+m_A+m_W \qquad (24)$$

$$\gamma=lm_A+(r_K+r_W)m_W \qquad (25)$$

Table 1.Pparameters of the On-ball Balancing Mobile Robot

Description	Symbol / Unit	Value
Mass Of The Driven Ball	m_K/Kg	0.25
Mass Of The Omni-Directional Wheel	m_W/Kg	1.5
Mass Of The Body	m_A/Kg	10
Radius Of The Driven Ball	r_K/m	0.123
Radius Of The Omni-directional Wheel	r_W/m	0.05
Radius Of The Body	r_A/m	0.15
Gravity Center Height of the Body	l/m	0.3
Inertia Of The Ball	$\mathrm{J}_K/(\mathrm{Kg}\cdot m^2)$	0.0025
Inertia of the omni-directional wheel	$\mathrm{J}_W/(\mathrm{Kg}\cdot m^2)$	0.0014
Rotational Inertia Of The Body	$\mathrm{J}_A/(\mathrm{Kg}\cdot m^2)$	2.2
Zenith Angle	α/rad	π / 4
Height of the Body	h/m	0.5

4 CONTROLLER DESIGN

Many scholars have done a lot of work in the research of robot control algorithms at home and abroad, such as used the LQR and PI double loop control in the literature [1], used calculated torque method and sliding mode variable structure control in literature [3], used the hierarchical sliding mode variable structure control in literature [4], and in literature [5, 6] is LQR. Based on the previous research in on-ball balancing mobile robot, we use the fuzzy self-adjusting PID control in this paper.

4.1 Introduction To Fuzzy Self-adjusting

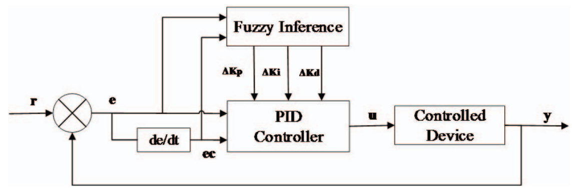

Fig 4. Schematic of the fuzzy self-adjusting PID control

Fuzzy self-adapting PID control is added the fuzzy inference method to achieve automatic adjusting of the three PID control parameters which based on the traditional PID control, in order to obtain satisfactory control effect, we use the error and error change rate as inputs.

When the control algorithm of fuzzy self-adapting PID is running, it detects the system value of d error E and error change rate EC constantly, then according to the fuzzy rules to judge, at last, the three PID control parameters value are obtained by defuzzication. Then carries on the PID control, the control quantity is output. Therefore, the PID control parameters achieve the adaptive adjustment, so that the system achieves steady-state is more faster, reduces the overshoot greatly, dynamic performances are excellent. Obviously, the key of fuzzy self-adapting PID control is to establish a suitable fuzzy rule table for the three parameters of Kp, Ki and Kd.

(1) Determine fuzzy subsets of inputs and input variables.

The input variables of error e , error change rate ec, proportion coefficient increment ΔKp, integral coefficient increment ΔKi and differential coefficient increment ΔKd are divided into seven fuzzy sets: NB (negative big), NM (negative medium) and NS (negative small), ZO (zero), PS (positive small), PM (positive medium), PB (positive big).

$\mathrm{e, ec, \Delta Kp, \Delta Ki\ and \Delta Kd} =$ {NB、NM、NS、ZO、PS、PM、PB}.

(2) Determine the actual domain of the input and output variables.

According to the actual situation of the controlled system, to determine the actual domain of each variables, assuming that the input and output variables of the actual domain is: { -6, -4, -2, 0 ,2 ,4 ,6}

Then at the k-th sampling time, the PID control parameters are:

$$\mathrm{Kp(k)} = \mathrm{Kp0} + \Delta\mathrm{Kp(k)}$$
$$\mathrm{Ki(k)} = \mathrm{Ki0} + \Delta\mathrm{Ki(k)}$$
$$\mathrm{Kd(k)} = \mathrm{Kd0} + \Delta\mathrm{Kd(k)}$$

Among them, Kp0, Ki0 and Kd0 are the initial values of the PID controller parameters.

(3) Define membership functions for each input and output variables.

Membership function is a characteristic function that represents the degree of the element in a fuzzy set belongs to the set, and its value varies between 0 and 1 continuously, expressed as "membership degree". Generally use the trapezoidal, Gauss-type, S-type, Z-type and other types of membership functions.

(4) Adding fuzzy inference rules item by item, then sets up the fuzzy control rule table.

Kp fuzzy rules as shown in Table 2, Ki fuzzy rules as shown in Table 3, theKd fuzzy rules as shown in Table 4.

Table 2. Fuzzy Rules Table Of Kp

ec \ e	NB	NM	NS	ZO	PS	PM	PB
NB	PB	PB	PM	PM	PS	ZO	ZO
NM	PB	PB	PM	PS	PS	ZO	ZO
NS	PM	PM	PM	PS	ZO	NS	NS
ZO	PM	PM	PS	ZO	NS	NM	NM
PS	PS	PS	ZO	NS	NS	NM	NM
PM	PS	ZO	NS	NM	NM	NM	NB
PB	ZO	ZO	NM	NM	NM	NB	NB

Table 3. Fuzzy Rules Table Of Ki

ec \ e	NB	NM	NS	ZO	PS	PM	PB
NB	NB	NB	NM	NM	NS	ZO	ZO
NM	NB	NB	NM	NS	NS	ZO	ZO
NS	NM	NM	NS	NS	ZO	PS	PS
ZO	NM	NM	ZO	PS	PS	PM	PM
PS	NM	NS	ZO	PS	PS	PM	PM
PM	ZO	ZO	PS	PS	PM	PB	PB
PB	ZO	ZO	PS	PM	PM	PB	PB

Table 4. Fuzzy Rules Table Of Kd

ec \ e	NB	NM	NS	ZO	PS	PM	PB
NB	PS	NS	NB	NB	NB	NM	PS
NM	PS	NS	NB	NM	NM	NS	ZO
NS	ZO	NS	NM	NM	NS	NS	ZO
ZO	ZO	NS	NS	ZO	NS	NS	ZO
PS	ZO	ZO	ZO	ZO	ZO	ZO	ZO
PM	PB	NS	PS	PS	PS	PS	PB
PB	PB	PM	PM	PM	PS	PS	PS

4.2 Controller Design

The inputs of the system are the torque of three motors T_1 , T_2 and T_3 , due to the generation of T_x , T_y and T_z equivalent as the torque in two dimensional plane, there is a big difference between them and the torque in actual system, so the T_x , T_y and T_z are transformed into torque outputs T_1 , T_2 and T_3 of the robot's driving wheel by three motors, so that can control the balance and movement of the robot.

That is：$u = [T_1 \quad T_2 \quad T_3]$。

The outputs are deflection angle φ_x and angular velocity $\dot{\varphi}_x$ of the robot's body and deflection angle ϑ_x and angular velocity $\dot{\vartheta}_x$ of the driven ball.

That is：$x = [\varphi_x \quad \vartheta_x \quad \dot{\varphi}_x \quad \dot{\vartheta}_x]$。

Linearized the model and get the state space equation of the system is:

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases}$$

Bring the data in table 1 into the state space equation, then get the model in XOZ, YOZ plane:

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -31.4603 & 0 & 0 \\ 0 & 16.3333 & 0 & 0 \end{bmatrix} \mathrm{B} = \begin{bmatrix} 0 \\ 0 \\ 13.4993 \\ -2.8567 \end{bmatrix}$$

$$\mathrm{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathrm{D} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Figure 5 is the fuzzy inference subsystem, and Figure 6 is the PID subsystem.

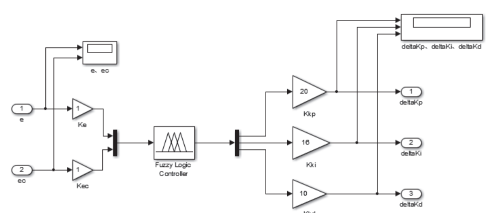

Fig 5. The fuzzy inference subsystem

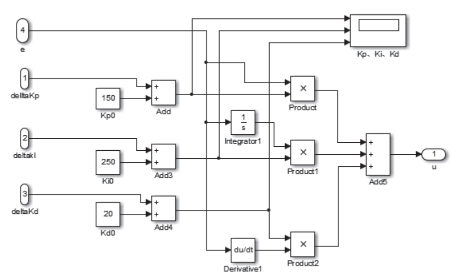

Fig 6. The PID subsystem

5 SIMULATION ANALYSIS

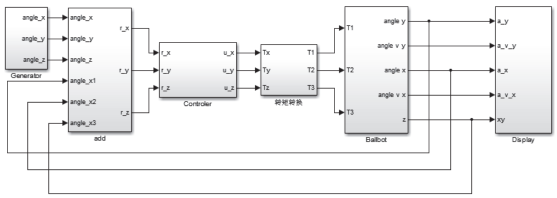

Fig 7. Schematic of the simulation in Simulink

Figure 7 shows the general schematic of simulation. Set the initial state of XOZ and YOZ are $[0 \quad 0.5 \quad 0 \quad 0]$ and $[0 \quad -0.5 \quad 0 \quad 0]$.

Simulation results are shown in Figure 8 and Figure 9.

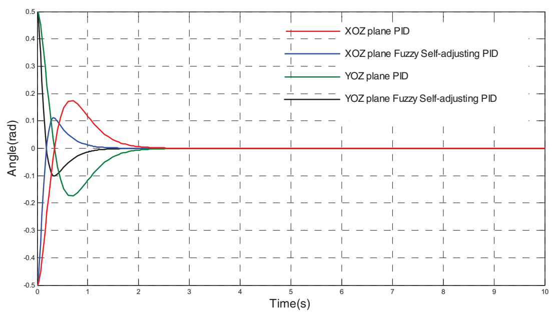

Fig 8. Simulation of angle

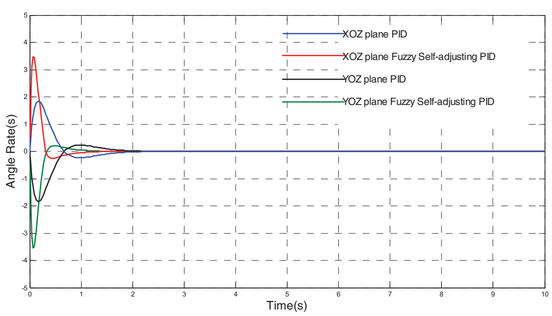

Fig 9. Simulation of angular velocity

Fig 10. Simulation changes of Kp, Ki and Kd at real-time

Figure 10 shows the changes of fuzzy self-adjusting PID control Kp, Ki and Kd at real-time, fuzzy self-adjusting PID control parameters for the initial values are $Kp0 = 150$, $Ki0 = 250$ and $Kd0 = 20$; PID control parameters are $Kp = 150$, $Ki = 250$ and $Kd = 20$.

In the simulation, the on-ball balancing mobile robot can response quickly in the initial state, reach the stable in less than 2 seconds and shows that the robot has strong robustness, and the fuzzy self-adjusting PID control has the better effect than PID control.

6 CONCLUSION

In this paper, we introduced the structure of the on-ball balancing mobile robot, and the dynamic model is established by using the Lagrange method. The state space equations of each plane is obtained by linearizing for the dynamic model, and the fuzzy self-adjusting PID control is proposed to control the robot's equilibrium. It is proved that the control strategy can achieve the balancing control of the robot by using simulation under Simulink, and has a good effect.

REFERENCES

[1] T. B. Lauwers, G. A. Kantor, R. L. Hollis, "A dynamical stable single wheeled mobile robot with inverse mouse-ball drive," Proceedings of IEEE International Conference on Robotics and Automation, pp. 2884-2889, May 2006.

[2] M. Kumagai and T. Ochiai, "Development of a robot balancing on a ball," Proceedings of International Conference on Control, Automation and Systems, pp. 433-438, Oct 2008.

[3] Tsai Ching-Chih, Juang Ming-Han,Chan Cheng-Kai , et al . Self-Balancing and Position Control Using Multi-Loop Approach for Ball Robots. 2010(08):251-256.

[4] Masaaki Kumagai, Takaya Ochiai. Development of a Robot Balancing on a Ball. International Conference on Control, Automation and Systems 2008(10):433—438.

[5] Z.H. Jia, S.K. Zhang, Linear quadratic optimal control of on-ball balancing mobile robot, Journal of Beijing Information Science and Technology University, 2013(12):91-96.

[6] A.G Zhou, J. Hong, Modeling and Control Design for a Ballbot, Mechatronics, 2012(11):13-19.