Research on Development-oriented IEEE MicroMouse Simulation System

Jingming Cao, Shenghua Dai, Lan Na*, Zhengjiao Li, Xing Zhou and Zhaoyi Li

School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, China

*Corresponding author e-mail: 20120233@bjtu.edu.cn

Abstract. With the popularity of MicroMouse project growing up, more and more people have participated in this activity. However, it is not efficient to develop and debug a MicroMouse in a real environment, and it is difficult to observe the details of the robot's motion state. This paper proposes a development-oriented IEEE MicroMouse simulation system that can accurately simulate the decision-making, control algorithm, and sensor information collection of real MicroMouse on embedded device, so that the development of mobile robots can free from the restricted of real environment.

1. Introduction

The MicroMouse is a wheeled mobile robot with embedded system that can explore in an unknown maze environment and make a sprint movement from the starting point to the ending point. Since the first competition held in 1979, the IEEE MicroMouse Competition has become a well-known intelligent robot competition until today [1]. Competitions are held in multi countries and regions yearly. The competition is able to test MicroMouse comprehensively which including the ability of maze searching, the control performance in sprint process as well as the stability of operation control.

The MicroMouse control system, as a product of multi-disciplinary intersection which is certain complexity, involves machine design, circuit design, control algorithms design, sensor perception application and some other technology. And with the labyrinth and narrow maze environment, the control of MicroMouse requires high accuracy. Therefore, there are many difficulties in developing a flexible movement mobile robot like MicroMouse in real environment directly. Specifically, the running of real robot makes single step debugging control algorithm impossible, circuit and mechanical equipment may damage since the unreasonable control parameters, and the real manual operation is inefficient.

Computer simulation technology is widely used in MicroMouse researches to simulate and test the performance of the control algorithm to optimize the robot control algorithm [2-4]. However, the simulation method for algorithm research only considers the system function and ignores the development characteristics for robots with embedded system. It is difficult to guarantee the algorithm program’s real-time performance and effectiveness on embedded devices with limited computing resources if the algorithm program is developed by this kind of simulation system. So, this kind of program cannot be directly applied to real robots. In order to facilitate the development and debugging of MicroMouse control programs without the need for real maze environment, it is necessary to study the simulation system for MicroMouse development.
Recently, Luis Piardi and others have proposed a MicroMouse hardware-in-the-loop simulation system which improved the application of computer simulation in MicroMouse development significantly [5]. However, this system just directly uses computer communication to obtain the simulation environment information and lacks the simulation of the physical sensor of MicroMouse, which cannot replace the real environment yet.

Therefore, this paper proposes a development-oriented IEEE MicroMouse simulation system, which can accurately simulate the decision-making process, control algorithm performance and sensor information collection of the MicroMouse on the embedded device. It can keep the development of mobile robots free from real environment. The structure of this paper is as follows. Chapter 2 introduces the MicroMouse control system, Chapter 3 gives the method of development-oriented MicroMouse simulation, Chapter 4 explains the experimental results of the simulation system and Chapter 5 is the conclusion of the research.

2. MicroMouse Control System
As a racing robot, the comprehensive performance of MicroMouse is affected by many aspects such as mechanical structure, control system performance, and intelligent decision-making processing. Fig. 1 is the MicroMouse device for education designed and manufactured by our team. The following will take this device as an example to introduce wheeled mobile robots with embedded system like MicroMouse.

**Figure 1. The appearance of the MicroMouse.**

**Figure 2. The control system structure of MicroMouse.**
The MicroMouse device is composed of mechanical structure, circuit system, monitoring module and etc. As shown in Fig. 2, the typical structure of MicroMouse control system has four layers which including sensor data processing, motion control, behaviour decision and state monitoring.

2.1. Sensor Data Processing
The MicroMouse obtains motion state information and maze environment information through a variety of sensors. The information collected by sensor is the basis for motion control and decision-making. The MicroMouse collects the distance information from the front, left and right sides to the walls of the maze through infrared sensors, the angular velocity in formation is detected through the gyroscope, and encoders get the rotational speed information of the left and right wheels. Sometimes, researchers will use multi-sensor fusion to improve accuracy and stability of sensors measurement [6, 7].

2.2. Motion Control
Motion control is the basis for the MicroMouse to search and perform tasks in the maze [8]. Motion control aims at the motion state expectations provided by the layer of behavior decision-making. According to the current motion state information provided by the layer of sensor data processing, the feedback control method is used to manipulate the running speed of the left and right motors of the robot to let robot complete go forward, turn left, turn right, and turn around and other basic actions. At present, the motor control methods commonly used in MicroMouse include PID [9], ADRC [10], adaptive control [11] and etc.

2.3. Behaviour Decision
The running process of MicroMouse in maze can be regarded as a combination sequence of a large number of basic actions [12]. The action sequence is planned by decision-making and executed by motion control finally. In maze searching stage, the MicroMouse aims to find the shortest path from the starting point to the end point as soon as possible, which need to selects the next search action based on the maze environment information is acquired.

2.4. State Monitoring
Since the MicroMouse has a fast speed of motion and requires high control accuracy, it is difficult to observe the details of motions visually. When debugging in real environment, key data of the system operation always be recorded by state monitoring in order to assists in analyzing the effect of the algorithm and improves the debugging efficiency [13]. In each control cycle, the interactive information between the above layers is recorded, including original data from sensor, position, posture and motor speed of MicroMouse, maze map information, maze area which already been searched and etc. These monitoring data can be used for online and offline data visualization analysis. It should be noted that in most of the MicroMouse competitions, it is not allowed to have wireless communication functions on robots in order to ensure that the robots are running autonomously and not be manipulated. Therefore, the condition monitoring function is only used in the development and pre-match debugging phase of the robot.

3. Method of Development-oriented MicroMouse Simulation
The key functions of a development-oriented robot simulation system are simulating the robot running environment sufficient and accurate, and have a complete visualization of the simulation running process. Therefore, the MicroMouse simulation system consists of a hardware simulation layer and a visualization layer, whose specific structure is shown in Fig. 3. In the hardware simulation layer, we simulate all kind of the running process of MicroMouse through build sensor model, dynamics model, and kinematics model. At the same time, the key data in the hardware simulation layer is sent to the visualization layer for display. The key model and technology in MicroMouse simulation will be introduced below in detail.
3.1. Sensor Simulation Model

The MicroMouse is equipped with a variety of sensors, which convert the real environmental information into electrical signals and send them to the microcontroller for analysis. The goal of the sensor simulation model is to convert the simulation state information of MicroMouse in the virtual maze into the original output signal of the corresponding sensor. After that, in the hardware simulation layer of the simulation system, the electrical signals reflecting the virtual environment are output to the corresponding pins of the MicroMouse microcontroller. Therefore, the sensor simulation model is the basis of hardware-in-the-loop simulation [14] which can replace real sensors and provide virtual environment information for the microcontroller of the MicroMouse.

To ensure the accuracy of the original signal simulation of the sensor, three kinds of sensor simulation methods commonly used in MicroMouse are introduced below, which contains infrared sensor, gyroscope and encoder.

3.1.1. Infrared Sensor. The infrared sensors of MicroMouse are composed of infrared transmitter SFH4550 and infrared receiver TPS601A. The two components are stacked on top of each other and kept in the same orientation. The switches of each infrared transmitters are controlled by MicroMouse processor. When switch is on, the infrared transmitter radiates infrared rays with a constant intensity $I_e$.

Suppose the distance between the infrared sensors and the maze wall is $x$, and the reflectivity of the maze wall is $\alpha$, then the radiant incidence $E$ detected by infrared receiver can be expressed as Formula 1:

$$E = \frac{\alpha I_e}{4x^2}$$

According to the electrical characteristics of infrared receiver, its output current $I_o$ and the radiant incidence $E$ have a dual logarithmic linear relationship within the effective range, as shown in Formula 2. The parameters $k$ and $b$ in Formula 2 can be found in the data sheet of TPS601A.

$$\ln I_o = k \ln E + b$$

According to Ohm's law, the output current $I_o$ of the infrared receiver is converted into voltage signal $V_o$ through the resistance $R_o$, and the voltage signal is connected to the analog-to-digital conversion (ADC) channel of the MicroMouse microcontroller. The voltage $V_o$ can be expressed as Formula 3:

$$V_o = I_o \cdot R_o$$

Finally, the relationship between the output voltage $V_o$ of the infrared receiver and the distance $x$ from the maze wall to it can be derived from Formula 1 to 3, as shown in Formula 4. Therefore, through the position and posture of the MicroMouse in maze, we can determine the distance of any infrared sensor to the wall of the maze and calculate its theoretical output voltage $V_o$. 

![Figure 3. Structure of MicroMouse simulation system.](image-url)
\[ V_0 = R_o \left( \frac{a t_e}{A x^2} \right)^k e^b \] (4)

3.1.2. Gyroscope. The gyroscope is an inertial navigation element, and its output a voltage related to the angular acceleration of the device which can be used to detect the rotation state of the MicroMouse. In the gyroscope sensor LY3200ALH data sheet, the sensitivity \( S_0 \), electrical zero \( V_{off} \), and measurement range of angular acceleration \((-a_m, a_m)\) can be queried.

According to Formula 5, we can solve the theoretical output voltage \( V_g \) of gyroscope according to the rotational angular acceleration \( \omega \) of MicroMouse.

\[
V_g = \begin{cases} 
-a_m \cdot S_0 + V_{off} & a \in (-\infty, -a_m) \\
 a \cdot S_0 + V_{off} & a \in (-a_m, a_m) \\
 a_m \cdot S_0 + V_{off} & a \in (a_m, +\infty)
\end{cases} \tag{5}
\]

3.1.3. Encoder. In the mechanical structure of MicroMouse, two encoders are respectively connected with the left and right wheels through gears to detect the running speed and direction of the wheels.

It is easy to obtain the wheel radius \( r \), tooth number of the wheel gear \( n_1 \), tooth number of the encoder gear \( n_2 \) and the encoder resolution \( N_0 \). According to the connection relationship between wheel gear and encoder gear, the relationship between wheel running distance \( s \) and the number of encoder output pulse \( N \) can be solved, as shown in Formula 6.

\[
N = \frac{s}{2\pi r} \cdot \frac{n_1}{n_2} \cdot N_0 \tag{6}
\]

The output of encoder can be simulated by pulse width modulation signal, and the instantaneous frequency \( f \) of the signal can be expressed as the differential of the encoder output pulse number \( N \) with respect to time. Combined with Formula 6, the relationship between the frequency \( f \) of the PWM signal output by the encoder and the speed \( v \) of MicroMouse wheel can be solved, as shown in Formula 7.

\[
f = \frac{v}{2\pi r} \cdot \frac{n_1}{n_2} \cdot N_0 \tag{7}
\]

3.2. Kinetic Model

The two PWM speed control signals output by MicroMouse processor are amplified by the motor driver and then connected to the DC motor. The kinetic model is used to simulate the operation of the left and right motors under the control signal which can simulate the motor driver and DC motor in the real system.

3.2.1. Motor Driver. The essence of motor drive is to amplify the power of the control signal so that it has the ability to drive the motor. For a PWM signal with a duty cycle of \( \eta \) and a peak value of \( U_0 \), the average input voltage \( U_i \) of the motor driver can be expressed as Formula 8:

\[
U_i = \eta U_0 \tag{8}
\]

The average output voltage \( U_m \) of the motor driver will carry out the DC motor. The \( U_m \) is related to the amplification factor \( K_s \) and delay time \( T_\text{delay} \) of the motor driver, which can be expressed as Formula 9:

\[
U_m(s) = K_s \cdot U_i(s) \cdot e^{-T_\text{delay}} \tag{9}
\]
3.2.2. DC Motor. The DC motor can be equivalent to a series structure of a single resistance $R_m$, $L_m$, and the electromotive force $e_m$ created by excitation, as shown in Fig. 4. The input voltage $U_m$ on motor is the output of the motor driver, and the current through the motor is denoted as $I_m$.

\[ U_m = I_m R_m + L_m \frac{dI_m}{dt} + e_m \]  

(10)

According to the equivalent circuit of DC motor, its electrical characteristics can be expressed by Formula 10:

In Formula 10, the back electromotive force $e_m$ is related to the motor rotation angular velocity $\omega_m$. Through the electromotive force coefficient $C_e$, this relationship can be expressed as Formula 11:

\[ e_m = C_e \omega_m \]  

(11)

According to the mechanical characteristics of DC motor, the electromagnetic torque $M$ is related to the current $I_m$. Denoted the torque coefficient is $C_m$. Then we can have Formula 12.

\[ M = C_m I_m \]  

(12)

The torque balance of the DC motor during operation can be expressed by Formula 13. In the formula, $M_L$ represents the load torque, $J$ represents the moment of inertia, and $\mu$ represents the damping coefficient.

\[ M = M_L + J \omega_m + \mu \omega_m \]  

(13)
According to Formula 10 to 13, the transfer function of the DC motor model can be calculated. The system structure block diagram also can be drawn, as shown in Fig. 5. This model describes the basic dynamic properties of the DC motor, which is able to realize the task of calculating the motor speed by the input driving voltage.

3.3. Kinematics Model

The motion state of MicroMouse is mainly composed of position coordinates \((x, y)\), angle \(\theta\), linear velocity \(v\), angular velocity \(\omega\) and etc. The kinematic model needs to calculate all the operating state information of MicroMouse according to the rotational angular velocity \(\omega_m\) of the robot motor provided by the kinetic model. The rotational angular velocity of the left motor is recorded as \(\omega_{mL}\), and the rotational angular velocity of the right motor is recorded as \(\omega_{mR}\). After that, the information used for the simulation process display is sent to the visualization layer. Only the information that can be observed by MicroMouse is provided to the sensor simulation model.

3.3.1. Sensor Observable Information. The left and right encoders can observe the running speed of the wheels. Assume the wheel radius of MicroMouse is \(r_w\), the running speed of the left wheel is \(v_L\) and the running speed of the right wheel is \(v_R\). Then we can get Formula 14 and 15:

\[
\begin{align*}
v_L &= \omega_{mL} r_w \\
v_R &= \omega_{mR} r_w
\end{align*}
\]  

The gyroscope can detect the rotation angular acceleration \(\alpha\) of the robot which can be calculated by the speed of left and right wheels of MicroMouse according to the relationship of kinematic, as shown in Formula 16. Assume the width of MicroMouse is \(d\).

\[
\alpha = \frac{r_w}{d} \left( \frac{d\omega_{mL}}{dt} - \frac{d\omega_{mR}}{dt} \right)
\]

3.3.2. Information for Visualization. In order to display the simulation running process of MicroMouse in real time, we need to calculate the position coordinates \((x, y)\) and angle \(\theta\) of MicroMouse.

According to the information provided by the kinetic model, the operating angle \(\theta\) of MicroMouse at time \(t\) can be expressed as Formula 17:

\[
\theta(t) = \frac{r_w}{d} \int_0^t (\omega_{mL} - \omega_{mR}) dt
\]

Mark the starting position of MicroMouse as \((x_0, y_0)\). According to kinematic law, the abscissa \(x\) and ordinate \(y\) of MicroMouse at time \(t\) can be expressed by Formula 18 and 19:

\[
\begin{align*}
x(t) &= x_0 + \frac{r_w}{2} \int_0^t (\omega_{mL} + \omega_{mR}) \sin \theta \, dt \\
y(t) &= y_0 + \frac{r_w}{2} \int_0^t (\omega_{mL} + \omega_{mR}) \cos \theta \, dt
\end{align*}
\]  

4. Simulation System Experiment of MicroMouse

According to the structure and method introduced in Chapter 3 of this article, a development-oriented MicroMouse hardware-in-the-loop simulation system was manufactured by our team. The implementation method will be described below and the simulation effect will be shown.

The circuit hardware platform of the MicroMouse simulation system is shown in Fig. 6. The microcontroller STM32F4 is a MicroMouse processor, and the program running on it has not undergone any special processing, which is consistent with the program running under the real environment. The
microcontroller STM32H7 realizes the task of the hardware simulation layer, obtains the motor control signal of the MicroMouse from the processor STM32F4, and outputs virtual sensor information through the DAC and PWM signals. The USB interface on the circuit hardware platform is used to connect the master computer PC, and the microcontroller that performs the simulation task provides the information needed for visualization.

The software for visualization layer of the simulation system is programming by C#, and the display effect is shown in Fig. 7. The software displays the operating status of the MicroMouse in real time according to its position and angle in the maze. The master computer for simulation can also be used to timing the simulate competition, monitor and analysing the running status of virtual MicroMouse.

![Circuit hardware platform of MicroMouse simulation.](image1)

**Figure 6.** Circuit hardware platform of MicroMouse simulation.

![Visualization software of MicroMouse simulation.](image2)

**Figure 7.** Visualization software of MicroMouse simulation.
5. Conclusion

With the increasing popularity of MicroMouse competitions, more and more researchers have begun to get involved in this field. In order to improve the efficiency of MicroMouse development and debugging, this paper proposes a development-oriented MicroMouse simulation system. Its main feature is that it uses sensor simulation model, kinetics model, and kinematics model to completely replace the real environment.

Using the MicroMouse simulation system described in this article for development has those following advantages:

- **Economize resources**: No IEEE standard MicroMouse maze environment and real MicroMouse device is required, only need the processor of MicroMouse and simple circuit hardware;
- **Improve debugging efficiency**: No need to place the maze manually and worry about wear and tear of the hardware of MicroMouse.
- **Convenient for observe the details of operating effects**: Without using high-speed cameras, the software of visualization can monitor key indicators in the operating process in real time;
- **Easy to transplant testing code to real device**: Because of the use of the same embedded processor, the code that the developer runs in the simulation environment can be easily migrated to the real MicroMouse device.

At the same time, the system has a certain approximation in the environmental modelling process, and it is difficult to fully simulate the objective environment, which needs to be further improved and optimized.

References

[1] Dai S, Zhao B, Li Z, et al. Design and practice from the micromouse competition to the undergraduate curriculum [C]. In Proceeding of 2015 IEEE Frontiers in Education Conference, 2015.

[2] Huo M, Wu J, Cai J, et al. Micromouse competition training method based on 3d simulation platform [C]. In Proceeding of 10th IEEE International Conference on Computer and Information Technology, 2010.

[3] Zhang H, Li H, and Wang Y. Principle and realization of micromouse fast dashing turning based on stm32 [J]. Open Automation and Control Systems Journal, 2015, 7 (1): 212-220.

[4] Cai J, Wu J, Huo M, et al. A micromouse maze solving simulator [C]. In Proceeding of the 2nd International Conference on Future Computer and Communication, 2010.

[5] Piardi L, Eckert L, Lima J, et al. 3d simulator with hardware-in-the-loop capability for the micromouse competition[C]. In Proceeding of 19th IEEE International Conference on Autonomous Robot Systems and Competitions, 2019.

[6] Su JH, Lee CS, and Chen CW. Sensor fusion algorithms for encoder resolution enhancement in educational mobile robots[C]. In Proceeding of 2016 International Conference on Advanced Robotics and Intelligent Systems, 2016.

[7] Sun SL and Deng ZL. Multi-sensor optimal information fusion kalman filter [J]. Automatica, 2004, 40 (6): 1017-1023.

[8] Zhang HM, Wang YH, Wang YH, et al. Design and realization of two-wheel micro-mouse diagonal dashing [J]. Journal of Intelligent & Fuzzy Systems, 2016, 31 (4): 2299-2306.

[9] Zhang GG and Furusho J. Speed control of two-inertia system by pi/pid control [J]. IEEE Transactions on Industrial Electronics, 2000, 47 (3): 603-609.

[10] Zheng Z, Dai S, Cao J, et al. IEEE micromouse control based on adrc algorithm [C]. In Proceeding of 4th IEEE Advanced Information Technology, Electronic and Automation Control Conference, 2019.

[11] Jung JW, Leu VQ, Do TD, et al. Adaptive pid speed control design for permanent magnet synchronous motor drives [J]. IEEE Transactions on Power Electronics, 2015, 30 (2): 900-908.

[12] Yuan CH, Liu Q, Liu XM, et al. An oblique sprint algorithm with eight directions for micromouse contests [C]. In Proceeding of 2018 International Conference on Electronics and Electrical...
[13] Ye Q, Li ZJ, and Cao JM. Study on the moving posture monitoring system of two-wheeled mobile robot [C]. In Proceeding of 6th International Symposium on Test Automation & Instrumentation, 2016.

[14] Ahmad N, Meng A, and Sultan M. Applications of hardware-in-the-loop simulation in automotive embedded systems [C]. In Proceeding of SAE 2020 World Congress Experience, 2020.