Vietnam Journal of Science and Technology ***59*** *(3) (2021) 345-356*

doi:10.15625/2525-2518/59/3/15583

DEVELOPMENT OF DIRECTIONAL ALGORITHM FOR THREE-WHEEL OMNIDIRECTIONAL AUTONOMOUS MOBILE ROBOT

Le Phuong Truong[1, *], Huan Liang Tsai[2], Huynh Cao Tuan[1]

[1]*Faculty of Mechatronics and Electronics, Lac Hong University, Bien Hoa, Viet Nam*
[2]*Department of Electrical Engineering, Da-Yeh University, Chang-Hua 51591, Taiwan*

[*]Emails: *lephuongtruong@lhu.edu.vn or letruonglhu@gmail.com*

Received: 12 October 2020; Accepted for publication: 11 February 2021

Abstract. An omnidirectional algorithm has been developed to control autonomous mobile robots with three wheels. The implementation system consists of three planet DC motors with a rated power of 80 W for three wheels, three encoders for speed feedback, one encoder for distance feedback, and one digital compass sensor for angle feedback. The main system with an STM32F407 microcontroller is designed for directional control of wheels based on the signal received from compass sensor and encoder, and then controls three subsystems to adjust the steering speed of each wheel. The subsystem is built to control only one DC motor for each wheel with the built-in proportional integral derivative controller (PID) algorithm by an STM32F103 microcontroller. Furthermore, the directional control algorithm is developed for three omnidirectional wheels and a PID algorithm is designed to control the speed of DC motor for each wheel. It can be seen from the results obtained that the proposed system has several advantages: (1) automatically adjusting the angle and position; (2) erasing the sensor for tracking line of the automobile robot; (3) being self-developed; (4) ensuring reliable accuracy.

Keywords: directional algorithm, three wheels, PID control, mobile robot.

Classification numbers: 4.1.2; 4.10.2.

1. INTRODUCTION

The international ABU Asia-Pacific Robot Contest (ABU Robocon) is a scientific challenge to apply the advanced research for both development and implementation of mobile robots with the preset task in the fields of automation, electronics, computer vision and image processing, as well as mechatronics system. The motion of autonomous mobile robot with high accuracy is very important. Furthermore, the wheel of autonomous robot is either Mecanum wheel or Omni wheel because it is designed for movement with omni-direction and flexibility. The structure of the two wheels is shown in Fig. 1.

In this work, the Omni directional wheels are applied to create the motion of mobile robot system. The advanced DC motors are utilized to control wheel torques independently and thus are capable of simultaneous and independent control over their rotational and translational motion. In recent years, several researches were focused on the Mecanum and Omni wheels as listed in Table 1.

(a) (b)

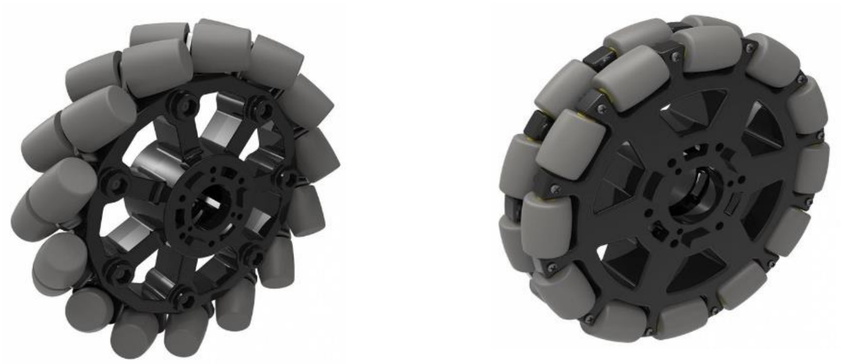

Figure 1. Wheel types: (a) Mecanum wheel, (b) Omni wheel.

Table 1. The comparison of Mecanum and Omni wheels.

Wheel type	Algorithms	Hardware	Direction control	Wheel number	References
Mecanum	Closed-loop control and dead-reckoning for navigation	Mitsubishi M16C/62 microcontroller	Directional	4	Cooney *et al*. 2004 [1]
	Kinematic equation	None	Directional	4	Guo *et al*. 2016 [2]
	None	Nucleo-F411 RE	Directional	4	Yamada *et al*. 2017 [3]
	PSO based Fuzzy-PI	On-board computer	Directional	3	Feng *et al*. 2010 [4]
	Fuzzy-PI	MRL Robo Cup SSL	Directional	4	Hashemi *et al*. 2011 [5]
	Side-slip angle estimation method	None	Directional	4	Li et al. 2016 [6]
	Model predictive control/filtered Smith predictor	PIC32 family of Microcontroller	Directional	3	Jessivaldo Santos *et al*. 2017 [7]
	Barrel-shaped friction-driven	Arduino Mega2560	two-wheel driven Omni-directional	3	TatsuroTerakawa *et al*. 2019 [8]
Omni	**PID, directional algorithm**	**STM32F-family MCUs**	**Directional, position**	**3**	**Proposed system**

In addition, the Mecanum and Omni wheels have been controlled based on the motion equation. Several authors have demonstrated the accuracy of motion equation for the Omni-directional control. Table 2 shows the motion equation with 3 and 4 Omni wheels.

Table 2. The motion equation with three- and four-wheel Omni platform.

Type	Motion equation	Descriptions	References
Three-wheel	$\dot{x}(t) = Ax(t) + B[u(t) + K\,\mathrm{sgn}(x(t))]$ $y(t) = Cx(t)$ $u(t) = \begin{bmatrix} u_1(t) & u_2(t) & u_3(t) \end{bmatrix}^T$ $A = -\begin{bmatrix} \frac{3j^2K_t^2}{2MR_ar^2} + \frac{B_v}{M} & 0 & 0 \\ 0 & \frac{3j^2K_t^2}{2MR_ar^2} + \frac{B_{vn}}{M} & 0 \\ 0 & 0 & \frac{3b^2j^2K}{JR_ar^2} + \frac{B_w}{I_n} \end{bmatrix}$ $B = \frac{iK_t}{R_a}\begin{bmatrix} 0 & \frac{\cos(\delta)}{M} & -\frac{\cos(\delta)}{M} \\ \frac{-1}{M} & \frac{\sin(\delta)}{M} & \frac{\sin(\delta)}{M} \\ \frac{b}{I_n} & \frac{b}{I_n} & \frac{b}{I_n} \end{bmatrix}$ $K = \begin{bmatrix} -\frac{C_v}{M} & 0 & 0 \\ o & -\frac{C_n}{M} & 0 \\ o & 0 & -\frac{C_w}{I_n} \end{bmatrix}, C = I$	$u(t)$ is the continuous control input applied with the zero order. B_v , B_{vn} are viscous friction relative to v and v_n . B_ω, B_{vn} viscous friction relative to ω and v_n . C_v, C_{vn}, C_ω Coulomb friction relative to v, v_n and ω I_n is robot's inertial momentum; δ is angle; M is robot's mass; b is robot's radius . r is wheels radius; l is reduction; R_a is armature resistances; k_t . torque constants.	Santos *et al.* 2017 [7]
Four-wheel	$\dot{q} = k(q)v$ with $k(q) = \left[j^T(q)j(q)\right]^{-1} j(q)$ $j(q) = \begin{bmatrix} -s_1' & -c_1' & d \\ -s_2' & -c_2' & d \\ -s_3' & -c_3' & d \\ -s_4' & -c_4' & d \end{bmatrix}$ $s_i' = \sin[q_3 + (i-1)\pi/2]$ $c_i' = \cos[q_3 + (i-1)\pi/2]$ $q = [q_1, q_2, q_3]^T \in R^3$ $c_i^{'} = \cos(q_3 + (i-1)\pi/2;$ $s_i^{'} = \sin(q_3 + (i-1)\pi/2.$ $q = (q_1, q_2, q_3)^T \in R^3$	(q_1, q_2) is stands for the location of the center of the vehicle; q_3 is the angle made between the line joining the robot center with the first wheel center and the horizontal line; d is distance from the center of the robot to each of the wheel center.	Galicki. 2019 [9]

In this study, the proportional integral derivative controller (PID) algorithm is applied for the speed control of each wheel, angle and distance algorithms are applied to control directional position using both STM32F407 and STM32F103 MCUs, respectively. The main part of this paper includes a description of the system configuration in section 2, followed by section 3 revealing the development of directional algorithm and section 4 showing the diagram of algorithm and hardware development. Discussions and brief conclusions are represented in section 5 and section 6, respectively.

2. SYSTEM DESCRIPTION

Figure 2 shows the schematic diagram of the three-wheel Omni directional system which consists of a mechanical system, the main control interface circuitry with a STM32F407 MCU, three sub control systems with a STM32F103 MCU-based interface circuitry, and a C## programming platform. The three encoder subsystems can give feedback on the speed of each wheel under practical operation conditions for the PID control through an interface circuitry built-in with a STM32F107 MCU. In addition, a compass sensor is built for measuring the angular motion of system and an encoder transfers the distance data through an interface circuitry of STM32F407 MCU-based main system which is configured in the C## program.

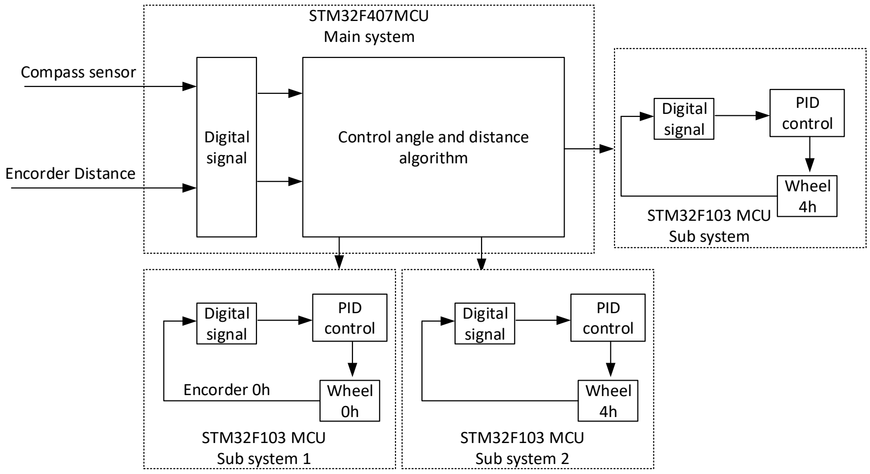

Figure 2. The schematic diagram of the three-wheel Omni directional system.

The main interface circuitry is designed with the self-developed algorithm for both angle and distance control to calculate the speed and angle of each wheel based on the feedback data of the compass sensor and encoder system. Furthermore, the interface circuitry of the control sub-system is the built-in PID algorithm through an STM32F103 MCU for the speed control of each wheel.

3. DEVELOPMENT OF OMNIDIRECTIONAL WHEELS AND ALGORITHM

3.1. Omnidirectional wheels

The omnidirectional wheels are designed on the basis of alternating passive rollers around the circumference which are perpendicular to the turning direction (see Fig. 3.). The result is that the wheel can be driven with full force, but will also slide laterally with great ease.

Figure 3. Prototype of Omnidirectional wheels.

3.2. Equation of motion

Omni-directional drives have been realized in several innovative ways such as the Newton's laws of motion (Kalmár *et al*. [10]) and the equation of motion (Tamás Kalmár [11]) as follows:

$$A\ddot{z}(t) + B\dot{z}(t) = C(\theta)v(t) \tag{1}$$

where $z(t) = \begin{bmatrix} x & y & \theta \end{bmatrix}^T$ is the state vector and $v(t) = \begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix}^T$ is the control input vector, and A, B, C are respectively given by

$$A = \begin{bmatrix} \frac{m}{\alpha} & 0 & 0 \\ 0 & \frac{m}{\alpha} & 0 \\ 0 & 0 & \frac{h}{\alpha L} \end{bmatrix} \tag{2}$$

$$B = \begin{bmatrix} \frac{3\gamma}{2\alpha} & 0 & 0 \\ 0 & \frac{3\gamma}{2\alpha} & 0 \\ 0 & 0 & \frac{3\gamma L}{\alpha} \end{bmatrix} \tag{3}$$

and

$$C = \begin{bmatrix} -\sin\theta & -\sin\left(\frac{\pi}{3} - \theta\right) & \sin\left(\frac{\pi}{3} + \theta\right) \\ \cos\theta & -\cos\left(\frac{\pi}{3} - \theta\right) & -\cos\left(\frac{\pi}{3} + \theta\right) \\ 1 & 1 & 1 \end{bmatrix} \tag{4}$$

The state variables x, y, θ are the linear and angular displacements, m is the mass of the vehicle, v_i is the input voltage of motor to drive the *i*th wheel. The characteristic constants of the motors are denoted by α and γ. The variable L is the distance.

3.3. Development of the equation of voltage input for each wheel

One of the important parameters to develop directional algorithm for Omnidirectional three-wheel is the voltage input of DC motor for each wheel, which can be given by [8 - 11]:

$$V_1 = \omega_d \sin\left(\theta_d + \frac{\pi}{3}\right) + \omega_\theta \tag{5}$$

$$V_2 = \omega_d \cos\left(\theta_d + \frac{\pi}{3}\right) - \omega_\theta \tag{6}$$

and

$$V_3 = \omega_d \cos\left(\theta_d + \frac{\pi}{3}\right) + \omega_\theta \tag{7}$$

where V_1, V_2, V_3 are the voltage input to DC motor of the 1st, 2nd and 3rd wheels, ω_d is the value of rotation of each wheel, ω_θ is the value of rotation of the robot head, θ_d is the angular of robot. In this case, the values of ω_d and ω_θ are in the range of $[-1, 1]$.

3.4. Development of the equation of rotation angle of autonomous robot

It is assumed that at the beginning the autonomous robot has the angle (α), angular direction (δ), and angle responded from compass sensor (α_{cs}). When the autonomous robot is moving, the angular motion of autonomous robot could have an angular error (β),which can be written as:

$$\beta = \alpha_{cs} - \alpha \tag{8}$$

Case study 1: The value of β=0, the autonomous robot keeps moving.

Case study 2: The value of $\beta \neq 0$, the system needs to control the rotation angle by the equation below:

$$\delta_a = \delta - \beta \tag{9}$$

where δ_a is the angular adjustment.

3.5. PID equation

The PID controller in a feedback control loop continuously calculates an error signal $e(t)$, which is the difference between a desired set point and a measured output variable, and applies a correction based on proportional, integral and derivative terms. The function of PID controller can be written as [12]:

$$u(t) = k_p \left[e(t) + T_d \frac{d(e)}{d(t)} + \frac{1}{T} \int_0^t e(t) d(t) \right] \tag{10}$$

$$u(t) = \left[k_p e(t) + k_d \frac{d(e)}{d(t)} + k_i \int_0^t e(t) d(t) \right] \tag{11}$$

where k_p , k_i , k_d are the coefficients corresponding to the proportional, integral and derivative terms.

4. THE DIAGRAM OF ALGORITHM AND HARDWARE DEVELOPMENT

4.1. The diagram of algorithm

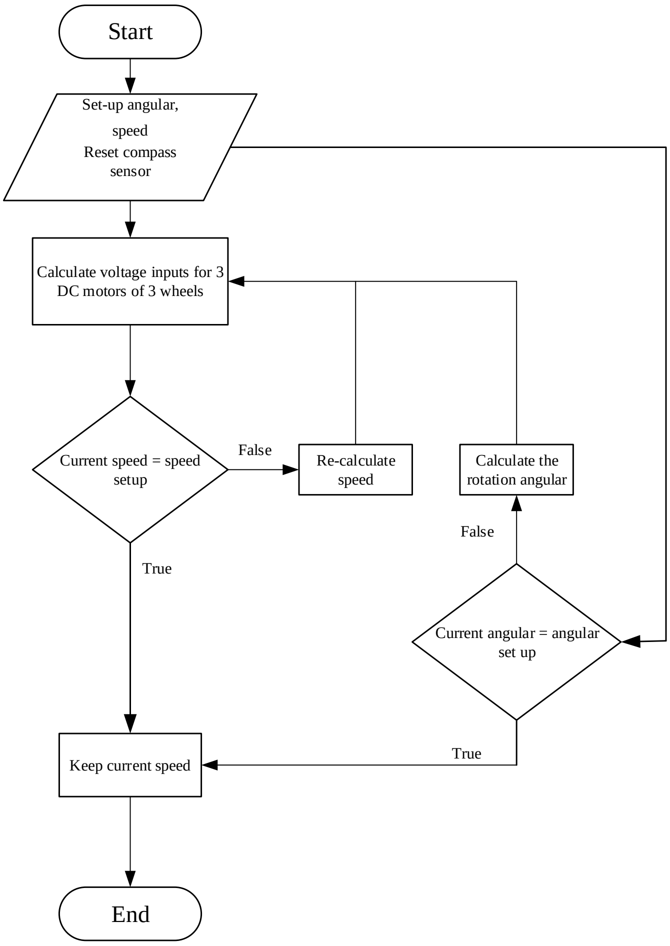

Figure 4. The algorithm flowchart of omnidirectional three-wheel.

The algorithm of the omnidirectional three-wheel is developed as shown in Fig. 4. First, the angle and speed of the autonomous robot are initialized. The system will calculate voltage inputs for each wheel based on the Eqs. (5-7) and then check the current speed. If false, the voltage for each wheel will be increased or decreased until current speed equals to the desired speed.

Furthermore, the importance of omnidirectional control is the direction of an autonomous robot. In this case, if the current angle of the autonomous robot is not equal to the desired one, the system calculates the rotation angle.

4.2. Hardware implementation

The hardware implementation of the proposed system includes a STM32F407 MCU-based main control interface circuitry, a STM32F103 MCU-based control subsystem, four kinds of hardware devices for encoder system, and a compass sensor as shown in Fig. 5.

Firstly, three Planet DC motors with the rated 24 V, 60 W, 9000 rpm are built as omnidirectional wheels in the mobile robot. The speed of wheel motor is measured using an encoder with the rated 13 ppr. Secondly, the encoder for feedback on the distance was HEDS 5540 500 implse 3 Kamal **(**http://www.alldatasheet.com/datasheet-pdf/pdf/88208/HP/HEDS5540.html)

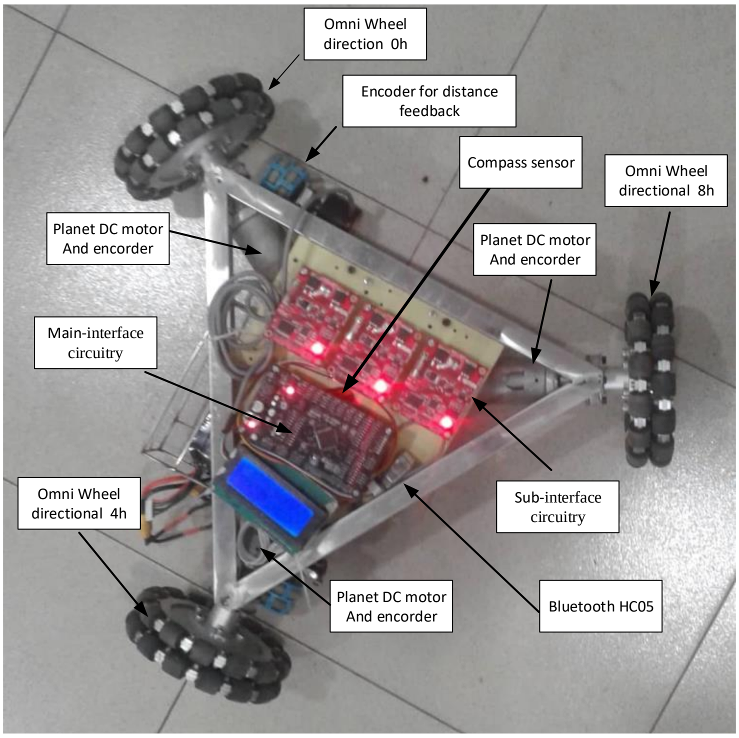

Figure 5. Experimental set-up.

Finally, the compass sensor was used to determine the angle of mobile robot using the MPU 6050 GY-521 (https://www.addicore.com/GY-521-MPU6050-p/170.htm). It contains a Tri-Axis angular rate sensor with a sensitivity up to 131 LSBs/dps in a full-scale range of ± 250, ± 500, ± 1000, and ± 2000 dps. Furthermore, the main interface circuitry is a self-developed platform with an embedded STM32F407VGT6 microcontroller (MCU) which consists of the ARM 32-bit Cortex-M4 CPU with a float point unit single precision, embedded memories of 1 MB flash memory and 192 kB SRAM, and three 12-bit ADCs. With the embedded universal synchronous/asynchronous receiver transmitters (USART) and ST-LINK/V2 interfaces and the sub, the interface circuitry is developed by STM32F103VG using the Cortex-M3 core, at a maximum CPU speed of 72 MHz. The portfolio covers from 16 kbytes to 1 Mbyte of Flash with motor control peripherals, USB full-speed interface and CAN.

5. RESULTS AND DISCUSSION

The auto mobile robot is desired to operate at an angle of 30° and a motor rate ranging from 2000 to 0 rpm depending on encoder feedback for directionally forwarding and backwarding.

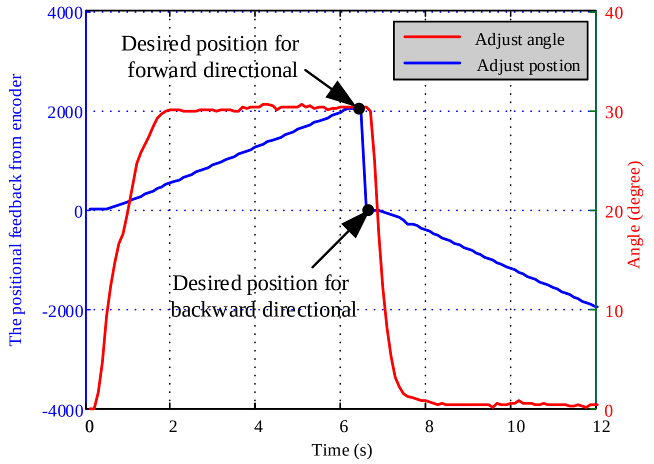

Figure 6. Adjustment of steering angle and position of mobile robot.

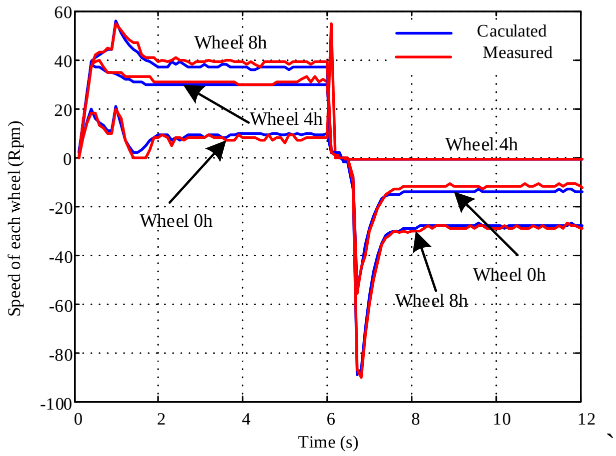

Figure 7. Adjustment of steering speed of each wheel.

From the results, the auto mobile robot needs 2 seconds to adjust the angle and 6 seconds to increase from 0 to 2000 rpm when moving forward (feedback from the encoder). It takes the same duration for the robot to decrease from 2000 to 0 rpm when moving backward. Figure 7 and Fig. 8 show that the 0 h directional wheel has a mission of adjusting the moving direction of the mobile robot, so the speed of this wheel is lower than that of 4 h and 8 h directional wheels and Fig. 9 show the time respond at difference angles from 30 to 60^0. Apart from that, as can be seen from

Fig 6, the duration for the auto mobile robot to reduce the speed from maximum value to 0 for each wheel is 0.5 s.

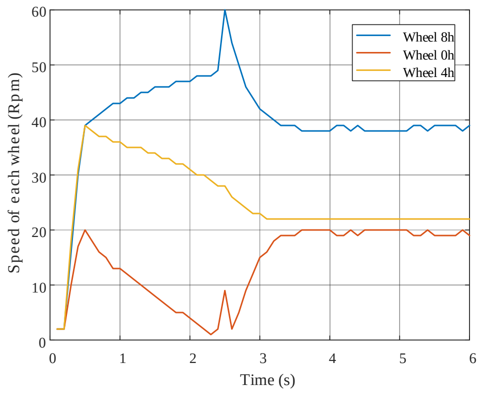

Figure 8. Speed of each wheel at an angle of 60 degree.

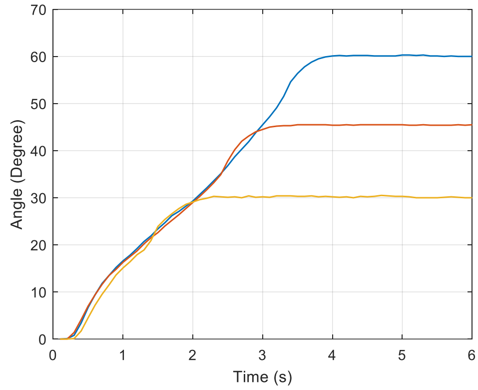

Figure 9. Time respond at difference angles.

In order to easily assess the precision of the proposed system, the difference between the results of measurement and calculation for omnidirectional autonomous robot output speed and angle is defined as

$$e = x_i - \hat{x}_i \tag{13}$$

where x_i and $\hat{x}_i$ are the i^{th} values of measurement and calculation, n is the total number of observations. The results of difference analysis for the omnidirectional output speed for each wheel are depicted in Fig. 10.

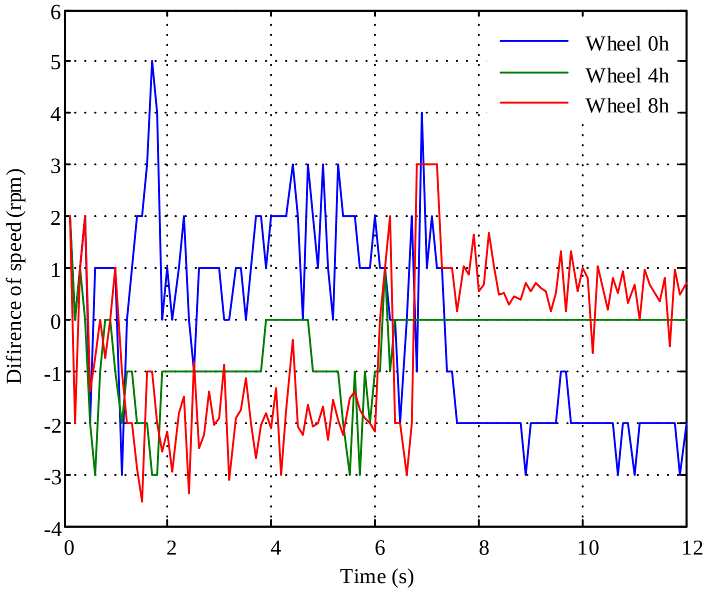

Figure 10. Difference analyses between measurement and calculation results.

The difference ranges for the omnidirectional output speed of each wheel are listed in Table 3. Furthermore, the corresponding root mean square errors (RMSE) with reference to the simulation are given by [13].

$$\sigma_{\mathrm{RMSE}} = \sqrt{\frac{\sum_{i=1}^{n}\left(x_i - \hat{x}_i\right)^2}{n}} \tag{14}$$

Table 3. Difference analyses of proposed system.

items	$e(rpm)$	$\sigma_{RMSE}(rpm)$
Wheel 0 h	-3 □ 5	1.66
Wheel 4 h	-3 □ 1	0.59
Wheel 8 h	-3.52 □ 3	1.40

Table 3 reveals the RMSE results of omnidirectional output speed of 1.66 rpm, 0.59 rpm and 1.40 rpm for 0 h, 4 h and 8 h wheels, respectively. Based on the confidence in algorithm for omnidirectional devices, the proposed system is validated to have a sufficient degree of precision and confidence in the control ability.

6. CONCLUSIONS

This paper presented the self-development of a directional algorithm to control three omnidirectional wheels of autonomous mobile robot system including the encoder system and compass sensor based STM32F407 and STMF103MCU with interface circuitry. From a practical viewpoint, the directional algorithm to control three omnidirectional wheels of autonomous mobile robots has many practical advantages including self-adjust the angle and position; erase the sensor for tracking line of the automobile robot; self-developed; ensure reliable accuracy. In the other way, another important practical implication is that the effectiveness of the low-level controller results in simplification of the high-level one.

CRediT authorship contribution statement. Phan Le Phuong Truong: methodology and experiments, Huan Laing Tsai: Reviewing and Editing; Huynh Cao Tuan: Supervision.

Declaration of competing interest. The authors declare that, they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

REFERENCE

1. Cooney J. A., Xu W. L., Bright G. - Visual dead-reckoning for motion control of Mecanum-wheeled mobile robot, Mechatronics **14** (6) (2004) 623-637.

   https://doi.org/10.1016/j.mechatronics.2003.09.002.

2. Guo S., Jin Y., Bao S., Xi F. F. - Accuracy analysis of omnidirectional mobile manipulator with mecanum wheels, Advances in Manufacturing **4** (4) (2016) 363-370.
3. Yamada N., Komura H., Endo G., Nabae H., Suzumori K. - Spiral mecanum wheel achieving omnidirectional locomotion in step-climbing, In the Proceedings of the IEEE

International Conference on Advanced Intelligent Mechatronics 2017, pp. 1285-1290. DOI: 10.1109/AIM.2017.8014195.

4. Feng H. M., Chen C. Y., Horng J. H. - Intelligent omni-directional vision-based mobile robot fuzzy systems design and implementation, Expert Systems with Applications **37** (2010) 4009-4019.
5. Hashemia E. J., Adidi M. G., Jadidi N. G. - Model-based PI–fuzzy control of four-wheeled omni-directional mobile robots, Robotics and Autonomous Systems **59** (2011) 930-942. https://doi.org/10.1016/j.eswa.2009.11.030.
6. Li B., Du H., Li W., Zhang Y. - Side-slip angle estimation based lateral dynamics control for omni-directional vehicles with optimal steering angle and traction/brake torque distribution, Mechatronics **30** (2019) 348-362.

   https://doi.org/10.1016/j.mechatronics.2014.12.001.
7. Jessivaldo Santos, Andr´e G. S. Concei¸c˜ao, Tito L. M. Santos - Trajectory tracking of Omni-directional Mobile Robots via Predictive Control plus a Filtered Smith Predictor, IFAC Papers OnLine **50** (1) (2017) 10250-10255.

   https://doi.org/10.1016/j.ifacol.2017.08.1286.
8. Terakawa T., Komori M., Yamaguchi Y., and Nishida Y. - Active omni wheel possessing seamless periphery and omnidirectional vehicle using it, Precision Engineering **56** (2019) 466-475. https://doi.org/10.1016/j.precisioneng.2019.02.003
9. Galicki M. - Collision-free control of an omni-directional vehicle, Robotics and Autonomous Systems **57** (2009) 889-900. https://doi.org/10.1016/j.robot.2009.06.005
10. Kalmár T. Nagy, D'Andrea R., Ganguly P. - Near-optimal dynamic trajectory generation and control of an omnidirectional vehicle, RobotAutonSyst **46** (2004) 47-64.

    https://doi.org/10.1016/j.robot.2003.10.003.
11. TamásKalmá Nagy T. - Real-time trajectory generation for omni-directional vehicles by constrained dynamic inversion, Mechatronics **35** (2016) 44-53.

    https://doi.org/10.1016/j.mechatronics.2015.12.004.
12. Zulfatman Z., Rahmat M. F. - Application of self-tuning fuzzy PID controller on industrial hydraulic actuator using system identification approach, The International Journal on Smart Sensing and Intelligent Systems **2** (2) (2009) 246-261.
13. Chouder A., Silvestre S., Sadaoui N., Rahmani L. - Modeling and simulation of a grid connected PV system based on the evaluation of main PV module parameters, Simulation Modelling Practice and Theory **20** (1) (2012) 46-58.

    https://doi.org/10.1016/j.simpat.2011.08.011.