Vol 4 (1) (2023)

Measurement, Control, and Automation

Website: https:// mca-journal.org

ISSN 1859-0551

Decentralized Collaboration Control and Obstacle Avoidance for Swarm Robot

Van-Phong Vu*, Viet-Hung Hoang, Hong-Thai Vu, Van-Tai Tran

Department of Automatic Control, Hochiminh City University of Technology and Education
**Corresponding author e-mail: phongvv@hcmute.edu.vn*

Abstract

Nowadays, swarm robot is widely used in practice. Controlling the swarm robot is always a challenge for many researchers. In this paper, the algorithm is proposed to control the swarm robot with three single mobile robots. The decentralized collaboration control is studied to control the swarm robot to move to the target. The robots of swarm mobile robots can communicate together via the wifi protocol. The camera and image processing are applied to localize the robots, determine the directions, and recognize the obstacles. The path-tracking algorithm is employed to control the leader robot which allows the swarm robot can track the reference path asymptotically. In addition, the Limit Cycle algorithm (LC) is proposed to control the swarm robot to avoid the obstacles on their way. Formation control based on leader-following is used to control the swarm robot moving in different formations. Both simulation and experiment results are provided to prove the effectiveness of the proposed method.

***Keywords**: Swarm Robot, Limit Circle, Formation Control, Obstacle Avoidance, Tracking Control.*

Tóm tắt

Ngày nay robot bầy đàn ngày càng được sử dụng rộng rãi trong thực tế. Việc điều khiển robot bầy đàn luôn là những thách thức cho các nhà nghiên cứu. Trong nghiên cứu này, các thuật toán điều khiển được đề xuất cho robot bầy đàn với ba robot di động. Phương pháp điều khiển hợp tác phân tán được nghiên cứu để điều khiển robot bầy đàn di chuyển tới vị trí mong muốn. Các robot của robot bầy đàn có thể giao tiếp thông qua giao thức wifi. Công nghệ xử lý ảnh được áp dụng để định vị, xác định hướng của các robot, và phát hiện vật cản. Thuật toán bám quỹ đạo được áp dụng cho robot dẫn đầu để điều khiển robot dẫn đầu bám theo đường đi xác định trước. Thuật toán "Limit Cycle" được sử dụng để điều khiển robot bầy đàn tránh vật cản trong quá trình di chuyển tới đích. Thuật toán điều khiển tạo hình cho robot bầy đàn đã được đề xuất để điều khiển robot bầy đàn có thể di chuyển với những hình dạng khác nhau. Kết quả mô phỏng và kết quả thực nghiệm đối với robot bầy đàn đã được trình bày để chứng minh sự thành công của phương pháp điều khiển.

1. Introduction

Recently, there is a wide range of applications of the swarm mobile robot in reality and it has also received great attention from researchers. Many previous studies paying attention to the controller design for swarm robots were found in the past few years [1]-[7]. For example, in paper [1], a new architecture for controlling the swarm robot to overcome the challenging of the environmental constraints and the dynamic changing of the mission is proposed. The adaptive fuzzy controller was studied to control the formation of the swarm robot, where the adaptive fuzzy was applied to approximate the error and the disturbance [2].

The algorithm based on Particle Swarm Optimization and the artificial potential field was proposed to control the swarm robot system to search the multi-goals [5]. In the paper [6], the moving distance of the swarm mobile robot was optimized by applying the PSO algorithm for minimizing energy consumption and moving time. The optimal algorithm for controlling the swarm mobile robot was analyzed in [7].

One of the most important problems for controlling the swarm robots is to keep the formation of the swarm mobile robots during movement. Recently, many papers focusing on formation control for swarm mobile robots have been published [8]-[16]. For instance, the modeling and controlling methods for the swarm robot were presented in [8]. In [9], the back-stepping control method was applied to control the formation of the swarm robot. The self-organization formation algorithm was investigated for swarm mobile robots in [10] and [11]. In addition, the method for controlling the swarm robot to keep the swarm mobile robot moving in a line formation was studied in [12]. Moreover, the positioning of every single mobile robot plays a very important role in formation control. *Morgan and Hereford* in [13] proposed the path formation method to control swarm mobile robots in case the information of the environment is limited. The method based on the vision technique was investigated in [14] and [15] for to obtain the position of the swarm mobile robot and keep the them moving in a specific formation. The IoT technology has been applied for the formation control of the swarm robot [16] in which the information of the trajectory of the swarm mobile robot is collected by the IoT platform.

However, these above papers merely pay attention to the formation control and other issues of swarm mobile robots such as path planning, and obstacle avoidance have not been addressed. It should be noted that during moving, encountering obstacle on the way is inevitable in practice.

Therefore, obstacle avoidance and path planning control for the swarm mobile robot is a pressing issues and great challenge for researcher. In the past few years, many papers have

Received: 26 November 2022; Accepted: 3 April 2023.

concentrated on the obstacle avoidance algorithm for mobile robots [17]-[23]. For instance, the Limit-Cycle algorithm was proposed for controlling the mobile robot to avoid the obstacles located on the way [17]. The orbit obstacle avoidance algorithm based on the limit-cycle method was proposed in [19] that allows the mobile robot can achieve the target without collision with obstacles. The navigation control algorithm and obstacle avoidance based on the distance sensor data were investigated for the mobile robots in the paper [20]. Additionally, the path planning method cooperating with obstacle avoidance based on the optimal methodologies such as SA, PSO, and FA were proposed for the mobile robots in [21]. Regarding the path planning and obstacle avoidance algorithm for swarm mobile robots, recently, there exist several studies have been published [24]-[26]. For example, *Meerza and Islam, et al.* in [24] have proposed a method based on the Q-learning for the path planning control of the swarm mobile robot. In [25], the problem of searching multiple targets of the swarm mobile robot and the hybrid path planning combined with obstacle avoidance issues were investigated in [25] and [26], respectively. However, these articles [24]-[26] only concentrated on obstacle avoidance and path planning and without mentioning the formation control of the swarm mobile robot.

With the aforementioned analysis, it is seen that some previous studies such as [8]-[16] focused on studying formation control and did not investigate the problems of obstacle avoidance and path planning. While several recent studies [24]-[26] merely researched path planning and obstacle avoidance. Due to this reason, the problems of formation control, positioning, obstacle avoidance, and path tracking will be investigated in this work. More importantly, not only the simulation results but also experimental results will be provided to demonstrate the effectiveness of the proposed method. The contributions of this work are summarized as follows:

1) The swarm robot with three single mobile robots is designed and implemented in this study.
2) The image processing technique is employed to determine the position of the swarm robot and detect the obstacles.
3) The leader-following control algorithm is applied to control the swarm robot. The robot leader is controlled for tracking the trajectories and then the two follower robots will follow the robot leader.
4) The formation control robot is used to keep the swarm robot in the specific formation.
5) The limit-cycle algorithm is employed for controlling the swarm robot to avoid obstacles and achieve the target accuracy.

The rest of this paper is organized as follows. The system description, communication protocol among the single robot, and the mathematical model of the robot are presented in Section 2.

The positioning and obstacle detection method is shown in Section 3. Section 4 presents the control algorithms that are applied to the swarm robot. The simulation and experiment results are founded in Section 5. Finally, the conclusions are drawn in Section 6.

2. System Description

2.1. System Description

In this paper, the swarm mobile robot structure is described in Fig. 1. The system includes a camera USB Logitech to collect the image and transmit it to the computer. The image will be processed in the computer to detect both the position and direction of the sub-robots as well as recognize the obstacles. The sub-robots are connected to the computer via wifi protocol by using Router Wi-Fi.

A single Robot is built with the structure in Fig. 2. Each sub-robot consists of Wi-Fi module AC8265, Jestson nano B01 board, Driver Expension Jetbot board, Memory card, and battery. Python language programming is used to make the program for the system. The system also has the GUI to monitor the operations of the swarm robot; and set up the formation and parameters of the system.

2.2. Communication Protocol

The swarm system is connected via wifi protocol that is presented in Fig. 3. The API server socket is built into the computer that contains the IP server, and three mobile robots are configured as the clients. The function of the socket server is to store and transmit the data among the sub-mobile robots. When the client sends the data to the server, other sub-mobile robots can access this data, however, the program in each sub-mobile robot will decide whether to receive this data or not.

In this paper, the server socket has to communicate with three sub-mobile robots, therefore, the multithreaded Process model is employed to help the system with multi-task operating simultaneously. Each task is executed independently, hence, if a task is an error, it will not affect others.

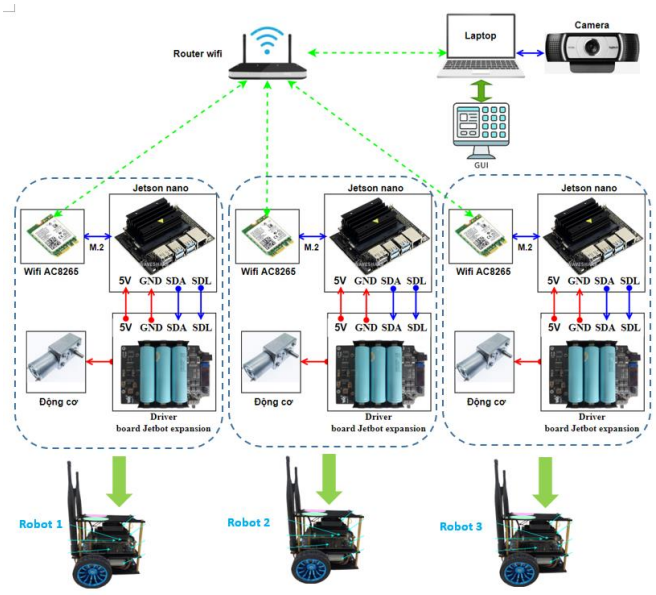


Figure 1: Structure of the Swarm Robot system

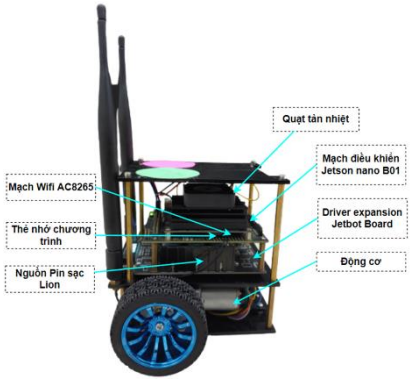


Figure 2: Structure of a single Robot

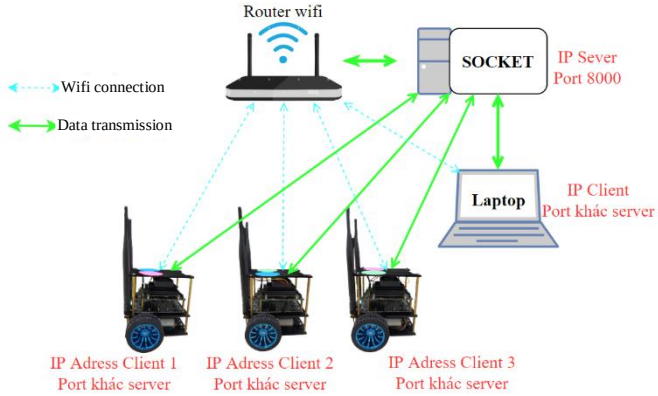


Figure 3: Data Transmission Protocol via API Socket

2.3. Mathematical Model of the sub-mobile robot

Let us consider the mobile robot in Fig. 4.

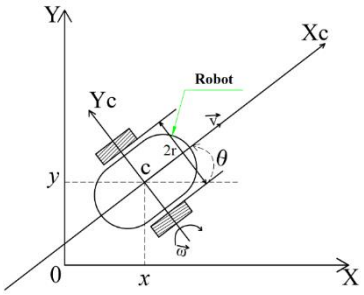


Figure 4: Model of the single mobile robot.

The mobile robot is represented under the following mathematical model

$$\dot{q} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} v\cos\theta \\ v\sin\theta \\ \omega \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (1)$$

where $q = [x \quad y \quad \theta]^T, x, y$ is the position of the robot, θ is the angle between the X-axis and line Xc (expressing the direction of the robot).

3. Positioning and obstacle detection

To control the warm robot, determining the position of each robot in the swarm plays a very important role. In this paper, the position of the mobile robot is obtained based on image processing. Each mobile robot is attached to two circle tags with different colors as Fig. 5.

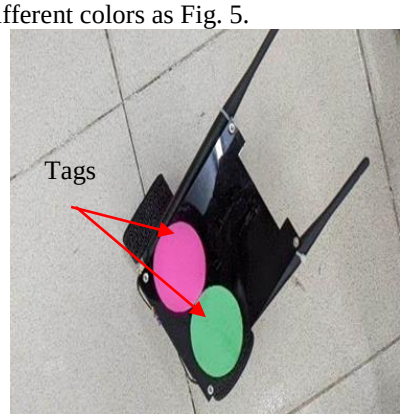


Figure 5. The mobile with color tags

The procure for determining the position of the swarm mobile robots and detecting obstacles is described in Fig. 6.

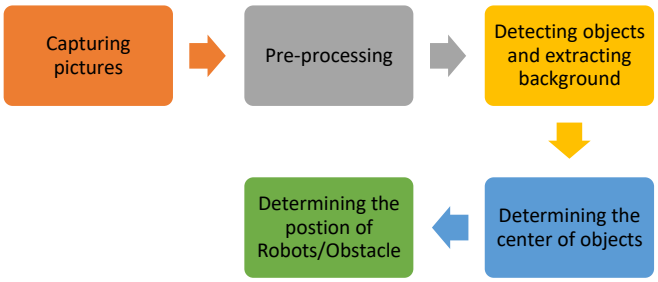


Figure 6. Procedure for obtaining the positions of the mobile robots and obstacles.

Pre-processing: Because the captured pictures from the camera are distorted that will affect the detecting results. Therefore, the images will be calibrated by applying the calibration tool of the OpenCV before using them for the next steps.

Detecting objects and extracting background: The objects will be detected based on the HSV color threshold. Using the Color Threshold will assist to extract the background and detect the objects.

Determining the center of the objects: The center of the objects is calculated in the following formula

$$C_x = \frac{M_{10}}{M_{00}} \text{ and } C_y = \frac{M_{01}}{M_{00}} \quad (2)$$

where (C_x, C_y) is the center of the objects. $M_{10}, M_{00}, and\ M_{01}$ are the image moment of each pixel.

Determining the position: The camera will detect the center of each tag by using the image processing algorithm as in Fig. 6. After obtaining the center of the two circle tags, the center of the mobile robot is computed as follows

$$x = \frac{x_1 + x_2}{2} \quad (3)$$

$$y = \frac{y_1 + y_2}{2} \quad (4)$$

and the direction of the mobile robot is calculated in the following formula

$$\theta = tan^{-1}\left(\frac{y_1 - y_2}{x_1 - x_2}\right) + 90^o \quad (5)$$

or

$$\theta = tan^{-1}\left(\frac{y_1 - y_2}{x_1 - x_2}\right) + 270^o \quad (6)$$

where (x_1, y_1), (x_2, y_2) are the centers of the tags, θ is the angle that expresses the direction of a mobile robot in Fig. 7.

After applying the image processing the position and the direction of the three mobile robots are determined in Fig. 8. The values of the positions of the leader mobile robot and two slave robots are shown in Fig.9

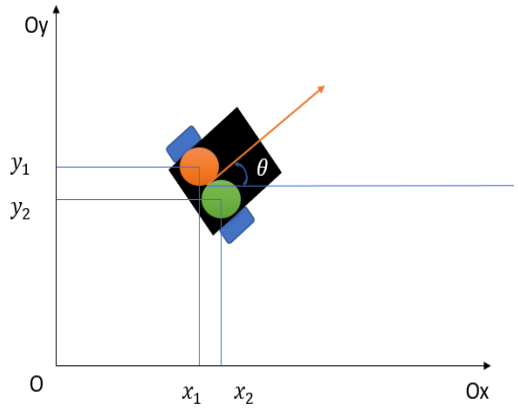


Figure 7: Positioning the mobile robot based on image processing.

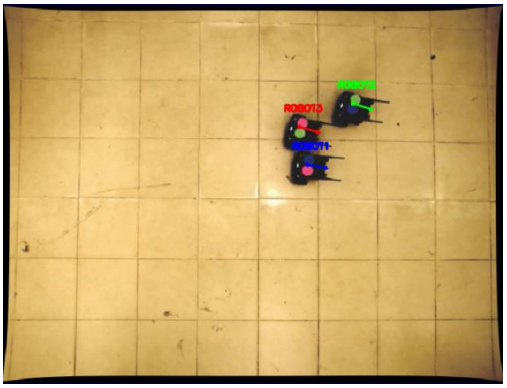


Figure 8: Position and direction of the mobile robot

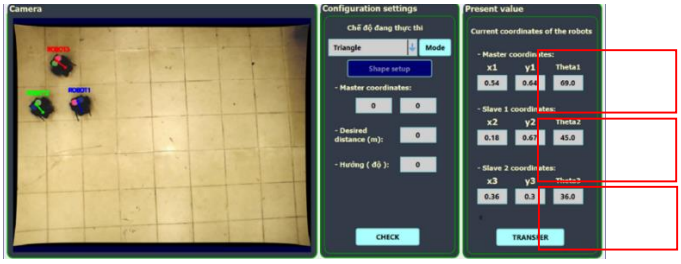


Figure 9: values of the position and direction of the mobile robot on the GUI

4. Control Algorithm for Swarm robot

In this paper, a swarm robot with three sub-robots is designed and implemented. Robot 1 is assigned as a leader robot; Robots 2 and 3 are follower robots. The motion of the Sub-Robot 1 will lead the motion of the swarm. The tracking path algorithm will be applied for the swarm robot to make the swarm robot to be able to track the pre-defined trajectories. The swarm also is controlled to avoid the obstacles that locate on their path by applying the Limit-Circle Algorithm.

Finally, the formation control method, called Leader-Follower, is employed to control the swarm robot moving in a specific form such as a triangle, vertical line, and horizontal line.

4.1. Tracking Path Algorithm

Because the movement of the swarm robot depends on the movement of the leader robot, in this article, the tracking control is applied for the leader robot to track the reference trajectories. The purpose of the tracking path algorithm is to control the center of the mobile robot *C* approach to the reference point and make the tracking error converge to zero when $t \to \infty$.

Suppose that the reference point is expressed in the following equation.

$$\dot{q}_r = \begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{\theta}_r \end{bmatrix} = \begin{bmatrix} v_r \cos\theta_r \\ v_r \sin\theta_r \\ \omega_r \end{bmatrix} = \begin{bmatrix} \cos\theta_r & 0 \\ \sin\theta_r & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_r \\ \omega_r \end{bmatrix} \quad (7)$$

where $q_r = [x_r \quad y_r \quad \theta_r]^T$, (x_r, y_r) is the reference position, θ_r is the reference angle. v_r and ω_r are the reference linear velocity and rotatory velocity (see Fig. 10).

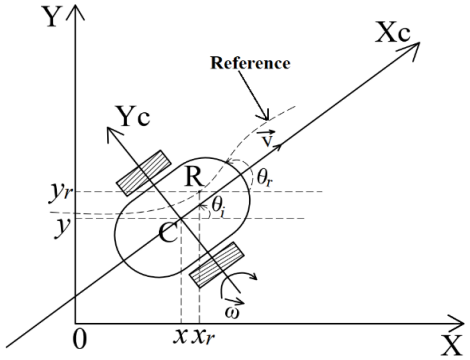


Figure 10. The mobile robot and tracking trajectory [22]

Let us define the tracking error

$$e_p = T_e(q_r - q) \quad (8)$$

where $T_e = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$, then (8) is written

$$\begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_r - x \\ y_r - y \\ \theta_r - \theta \end{bmatrix} \quad (9)$$

in which e_1, e_2 are the tracking error in the *x* and *y*-axis, respectively. e_3 is the angle error between θ_r and θ.

Based on the method in [22], the linear velocity and angle velocity for the leader robot are calculated as follows:

$$v_c = \begin{bmatrix} v_1 \\ \omega_1 \end{bmatrix} = \begin{bmatrix} v_r \cos e_3 + k_1 e_1 \\ \omega_r + k_3 v_r \sin e_3 + k_2 v_r e_2 \end{bmatrix} \quad (10)$$

where k_1, k_2, and k_3 are the positive scalars that are selected by using the trial-and-error method. Finally, based on the obtained v_1 and ω_1, the velocity of the left and right-wheel V_{R1}, V_{L1} of the leader robot are computed in the following formula

$$\begin{cases} V_{L1} = v_1 - \omega_1 r \\ V_{R1} = v_1 + \omega_1 r \end{cases} \quad (11)$$

where *2r* is the distance between the two wheels of the leader robot.

4.2. Obstacle avoidance algorithm

To avoid the obstacle located on the trajectory of the swarm robot, the Limit Cycle algorithm [5] is employed for the swarm robot. It should be noted that to avoid a collision between the robot and obstacles, the mobile robot and obstacle are surrounded by the safety circle with a radius *R*.

The limit cycle algorithm is operated based on the obstacle and mobile robot detection. Let us consider the system with three obstacles in Fig. 11 where O_d is the object that obstacle the moving of mobile robots and O_n is the object that is not impacted by the moving path of the robot.

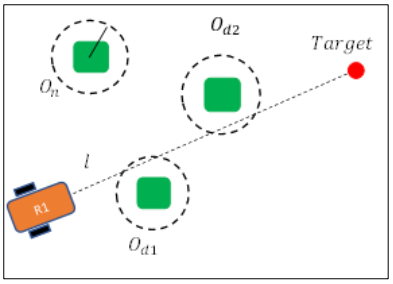


Figure 11: The mobile robot with three obstacles

To determine the path plan to achieve the target and avoid the obstacle, the Limit-Circle algorithm is carried out by the following procedure:

Step 1: Plot a line *l* connecting the center of the mobile robot and the center of the target that has an equation in Oxy coordinate: $ax + by + c = 0$

Step 2: If line *l* crosses the influence circle of the obstacle, this obstacle is considered a disturbing obstacle O_d, otherwise, it is taken into account non-disturbing obstacle O_n (see Fig. 10).

Step 3: The mobile robot will achieve to the target directly if there does not exist the obstacle O_d.

Step 4: If there exist O_d, the nearest distance *d* between the center of the obstacle O_d and line *l* is computed as follows:

$$d = \frac{aQ_x + bQ_y + c}{\sqrt{a^2 + b^2}} \quad (12)$$

where Q_x, Q_y are the center of the obstacle, *a*, *b*, and *c* are the coefficients of the line *l*.

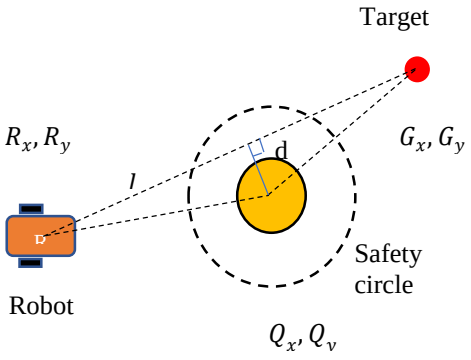


Figure 12: Determining the moving direction of the Mobile robot

Based on the value of *d*, the moving direction of the robot is determined. If *d* is positive, the robot will avoid the obstacle in the clockwise direction, otherwise, it will move in the counter-clockwise direction.

When there are disturbing obstacles, the avoiding obstacle trajectory is constructed by tracking the safety circle of the obstacle (see Fig. 13).

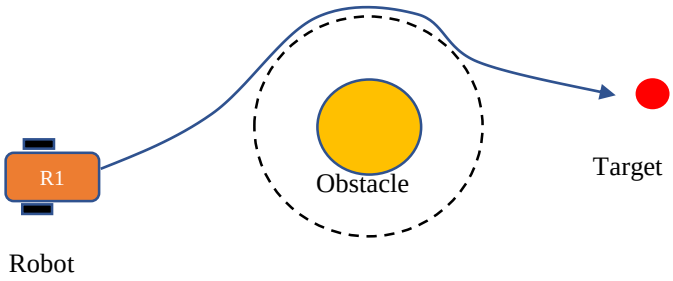


Figure 13: The avoidance obstacle operation

In order to build the avoidance obstacle trajectory, the coordinate needs to be re-established in the center of the obstacle as in Fig. 14.

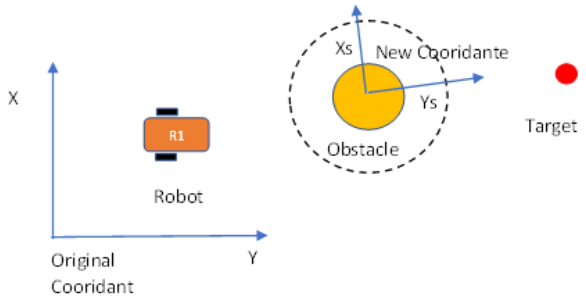


Figure 14: Re-establishing the coordinate

The new position (x_s, y_s) of the robot in the new coordinate is determined:

$$\begin{pmatrix} x_s \\ y_s \\ 0 \\ 1 \end{pmatrix}_O = \begin{bmatrix} cos\,\alpha & -sin\,\alpha & 0 & x_{obst} \\ sin\,\alpha & cos\,\alpha & 0 & y_{obst} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1} \begin{pmatrix} x \\ y \\ 0 \\ 1 \end{pmatrix}_A \quad (13)$$

After obtaining the incremental position of the robot in the new coordinate and based on the value of *d*, the robot will determine the moving direction (clockwise or counter-clockwise trajectory) to for avoiding the obstacle and achieve the shortest path. The two trajectories are described in the following equations.

Clockwise trajectory movement:

$$\begin{aligned} \dot{x}_s &= y_s + x_s(r_v^2 - x_s^2 - y_s^2) \\ \dot{y}_s &= -x_s + y_s(r_v^2 - x_s^2 - y_s^2) \end{aligned} \quad (14)$$

Counter-Clockwise trajectory movement:

$$\begin{aligned} \dot{x}_s &= -y_s + x_s(r_v^2 - x_s^2 - y_s^2) \\ \dot{y}_s &= x_s + y_s(r_v^2 - x_s^2 - y_s^2) \end{aligned} \quad (15)$$

r_v is calculated as follows

$$r_v = r_r + r_o + \delta \quad (16)$$

in which r_r is the radius of the robot, r_o is the radius of the obstacle and δ is the radius of the safety circle.

The angle for demining moving direction is obtained as follows:

$$\theta_r = arctan(\frac{\dot{y}_s}{\dot{x}_s}) \quad (17)$$

where

$$\dot{x}_s = \frac{d}{|d|} y_s + x_s(r_v^2 - x_s^2 - y_s^2) \quad (18)$$

$$\dot{y}_s = -\frac{d}{|d|} x_s + y_s(r_v^2 - x_s^2 - y_s^2) \quad (19)$$

From the obtained $\dot{x}_s$ and $\dot{y}_s$ in eq. (18) and eq. (19), the linear velocity for obstacle avoidance moving is calculated

$$v_r = \sqrt{\dot{x}_s^2 + \dot{y}_s^2} \quad (20)$$

The angular velocity is determined as follows

$$\omega_r = \frac{d\theta_r}{dt} \quad (21)$$

From (20) and (21), the reference trajectory for avoiding obstacles is

$$\begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{\theta}_r \end{bmatrix} = \begin{bmatrix} cos\,\theta & 0 \\ sin\,\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_r \\ \omega_r \end{bmatrix} \quad (22)$$

Applying the tracking reference path algorithm in Section 4.1, the linear velocity and angle velocity of the robot result in:

$$v_c = v_r\, cos\, e_\theta + k_x e_x \quad (23)$$

$$\omega_c = \omega_r + k_\theta v_r\, sin\, e_\theta + k_y v_r e_y \quad (24)$$

4.3. Formation Control Algorithm

To control the swarm mobile robots moving in a specific formation, the Leader-Following is applied in this paper. It means that the follower mobile robots will follow the action of the leader mobile robot when they are moving. There are three methods for formation control of the swarm mobile robot are illustrated in Fig. 15. In this paper, the swarm mobile robot with three mobile robots is considered as Fig. 16. The mobile robot R_1 is a leader and mobile robot R_2 and R_3 are the followers.

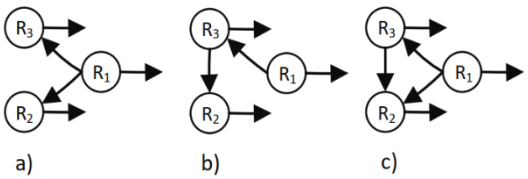


Figure 15: Formation control robot algorithm

In the algorithm in Fig. 15(a), the mobile robot R_2 and R_3 are controlled to keep the distance between mobile robots R_2 to R_1 and R_3 to R_1 to be a constant value. Fig. 15b expresses the formation moving of the swarm robots in which the swarm robot is controlled such that the distance between R_1 and R_3. R_3 to R_2 are constant. Unfortunately, the formation control methods in Fig. 15a and Fig. 15b is with low accuracy. Due to this reason, in this paper, the method in Fig. 15c is applied to control the swarm mobile robot.

From Fig. 16, it is seen that l_{12}, l_{13}, l_{23} are the distance between R_1 and R_2, R_1 and R_3, R_2 and R_3, respectively. In this paper, the control method will maintain the distances l_{12}, l_{13}, l_{23} (see Fig. 10) based on the formation of the swarm robot.

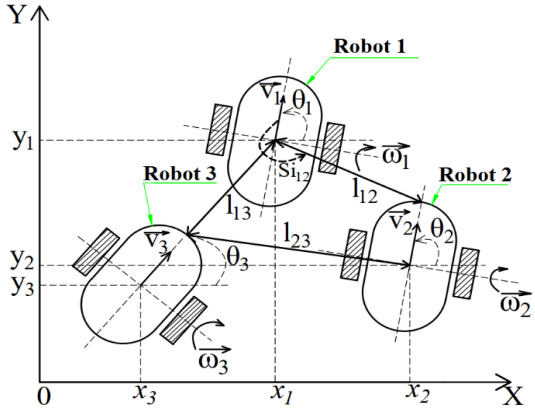


Figure 16: Swarm mobile robot system with three members.

From Fig. 16, we have

$$\begin{cases} l_{12}^x = x_1 - x_2 - d\,cos\,\theta_2 \\ l_{12}^y = y_1 - y_2 - d\,sin\,\theta_2 \\ l_{13}^x = x_1 - x_3 - d\,cos\,\theta_3 \\ l_{13}^y = y_1 - y_3 - d\,sin\,\theta_3 \\ l_{23}^x = x_2 - x_3 - d\,cos\,\theta_3 \\ l_{23}^y = y_2 - y_3 - d\,sin\,\theta_3 \end{cases} \quad (25)$$

Then the distances l_{12}, l_{13}, and l_{23} is computed as follows

$$\begin{cases} l_{12} = \sqrt{{l_{12}^x}^2 + {l_{12}^y}^2} \\ l_{13} = \sqrt{{l_{13}^x}^2 + {l_{13}^y}^2} \\ l_{23} = \sqrt{{l_{23}^x}^2 + {l_{23}^y}^2} \end{cases} \quad (26)$$

The angles $\psi_{12}, \psi_{13}, \psi_{23}$ (rad) are the angle between Robot 1 and robot 2; robot 1 and robot 3; and robot 2 and robot 3, respectively. These angles are computed in the following formula.

$$\begin{cases} \psi_{12} = arctan\left(\frac{l_{12}^y}{l_{12}^x}\right) - \theta_1 + \pi \\ \psi_{13} = arctan\left(\frac{l_{13}^y}{l_{13}^x}\right) - \theta_1 + \pi \\ \psi_{23} = arctan\left(\frac{l_{23}^y}{l_{23}^x}\right) - \theta_2 + \pi \end{cases} \quad (27)$$

in which θ_1, θ_2, and θ_3 (rad) are the angles between Robot 1, Robot 2, and Robot 3 with the Ox axis, respectively.

To design the formation controller for the swarm robot system, the dynamic equation of the swarm robot with three mobile robots is expressed in the following equations [22]:

$$\begin{cases} \dot{\theta}_2 = \omega_2 \\ \dot{\theta}_3 = \omega_3 \\ \dot{z} = G_2(z, \theta_1, \theta_2, \theta_3)u_{23} + F_2(z)u_1 \end{cases} \quad (28)$$

where $z = [l_{12} \quad \psi_{12} \quad l_{13} \quad \psi_{13}]^T$ is the output vector, $u_{23} = [v_2 \quad \omega_2 \quad v_3 \quad \omega_3]^T$ is the input vector. v_1, v_2, and v_3 are the linear velocities and $\omega_1, \omega_2, \omega_3$ are the angular velocities of Robot 1, Robot 2, and Robot 3, respectively.

Based on the method in [20], the parameters G_2, F_2, and p are determined

$$G_2 = \begin{bmatrix} cos\,\gamma_{12} & d\,sin\,\gamma_{12} & 0 & 0 \\ \frac{-sin\,\gamma_{12}}{l_{12}} & \frac{-d\,cos\,\gamma_{12}}{l_{12}} & 0 & 0 \\ 0 & 0 & cos\,\gamma_{13} & d\,sin\,\gamma_{13} \\ -cos\,\psi_{23} & 0 & cos\,\gamma_{23} & d\,sin\,\gamma_{23} \end{bmatrix} \quad (29)$$

$$F_2 = \begin{bmatrix} -cos\,\psi_{12} & 0 \\ \frac{sin\,\psi_{12}}{l_{12}} & -1 \\ -cos\,\psi_{13} & 0 \\ 0 & 0 \end{bmatrix} \quad (30)$$

$$p = \begin{bmatrix} k_1(l_{12}^d - l_{12}) \\ k_2(\psi_{12}^d - \psi_{12}) \\ k_3(l_{13}^d - l_{13}) \\ k_4(l_{23}^d - l_{23}) \end{bmatrix} \quad (31)$$

where k_1, k_2, k_3, and k_4 are the positive scalar controller gains that are selected arbitrarily by the user. $\gamma_{ij} = \beta_{ij} + \psi_{ij}$ with $\beta_{ij} = \theta_1 - \theta_2$. The control inputs for Robot 2 and Robot 3 are computed:

$$u_{23} = [v_2 \quad \omega_2 \quad v_3 \quad \omega_3]^T = G_2^{-1}(p - F_2 u_1) \quad (32)$$

It should be noted that the relations between the linear velocity and the rotatory velocity of Robot 2 and 3 and the velocities of the left and right wheels are represented

$$\begin{cases} v_2 = \frac{V_{L2}+V_{R2}}{2} \\ \omega_2 = \frac{V_{R2}-V_{L2}}{2r} \end{cases} \quad (33)$$

$$\begin{cases} v_3 = \frac{V_{L3}+V_{R3}}{2} \\ \omega_3 = \frac{V_{R3}-V_{L3}}{2r} \end{cases} \quad (34)$$

in which V_{L2} and V_{R2} are the velocities of the left and right wheels of Robot 2. V_{L3} and V_{R3} are the velocities of the left and right wheels of Robot 3.

From Eqs. (33) and (34), the velocities of the left and right wheels of Robot 2 and Robot 3 are determined:

$$\begin{cases} V_{L2} = v_2 - \omega_2 r \\ V_{R2} = v_2 + \omega_2 r \\ V_{L3} = v_3 - \omega_3 r \\ V_{R3} = v_3 + \omega_3 r \end{cases} \quad (35)$$

5. Simulation and Experiment Results

In this section, both simulation and experiments are executed for the swarm robot with three mobile robots the prove the effectiveness of the proposed method.

5.1 Simulation results

❖ **Tracking reference trajectory for a single mobile robot**

Firstly, to verify the effectiveness of the tracking controller, the single mobile robot is used for validation. The reference trajectory used for simulation in this paper is calculated by using eq. (7) in Section 4.1 with $(x_r, y_r) = (2,\ 1)\ (m)$ and $\theta_r = 45^o; v_r = 0.04(m/s)$. The initial position of the leader robot is $(x_o, y_o) = (2.18, 1)\,(m)$ and $\theta = 90^o$. The simulation results for tracking reference trajectory are shown in Figs. 17-20.

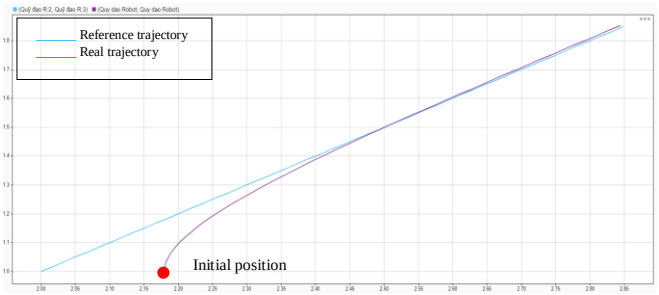


Figure 17: Tracking trajectory control for a robot

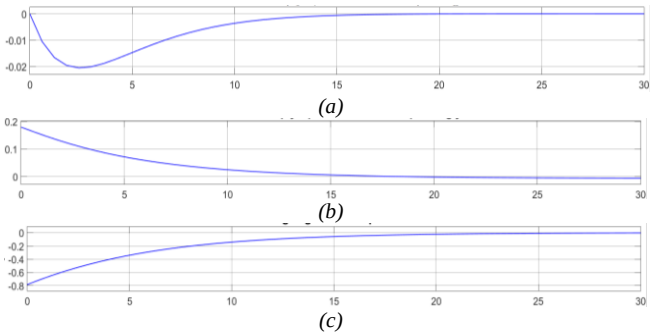


Figure 18: Tracking errors: (a) the tracking error in the x-axis, (b) the tracking error in the y-axis, (c) the direction angle error.

The simulation results are shown in Figs. 17 and 18. Fig. 17 shows that with a single mobile robot, the tracking controller can successfully control the robot to track the reference trajectory accurately. The tracking errors are illustrated in Fig. 18, it is obvious that the tracking errors in the x-axis, y-axis, and direction angle approach zero asymptotically.

❖ **Tracking reference path for a warm robot with a triangle formation**

In this section, Robot 1 is the leader; Robot 2 and Robot 3 are the followers. The leader robot is controlled to track the reference trajectory and two follower robots will follow the leader robot with a constant distance to achieve the target.

The initial position of the leader robot (Robot 1) is: $[x_1 \quad y_1 \quad \theta_1] = \left[1 \quad 1 \quad \frac{\pi}{3}\right]$and the linear speed and angular speed are $(v_o \quad \omega_o) = (0.04 \quad 0)$. The target is $(x_G \quad y_G) = (2.4 \quad 3.4)$. The initial positions of Mobile robot 2 and Mobile robot 3 are: $[x_2 \quad y_2 \quad \theta_2] = \left[0.8 \quad 0.8 \quad \frac{2\pi}{3}\right]$ and $[x_3 \quad y_3 \quad \theta_3] = \left[1.1 \quad 0.9 \quad \frac{\pi}{1.2}\right]$.

The expected distance between the mobile robot leader and two followers: mobile robot 2 and mobile robot 3, and the expected directions of these two robots are $[l_{12}^d \quad \psi_{12}^d \quad l_{13}^d \quad l_{23}^d] = [0.3 \quad 1.3\pi \quad 0.3 \quad 0.3]$.

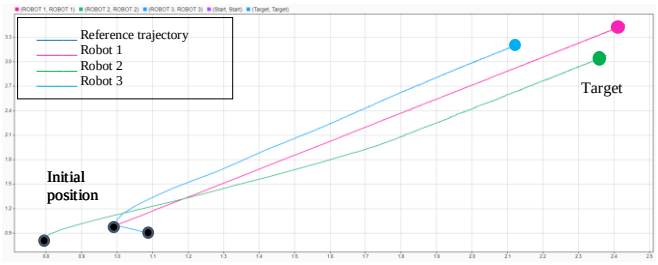


Figure 19: Moving the swarm robot with triangle formation

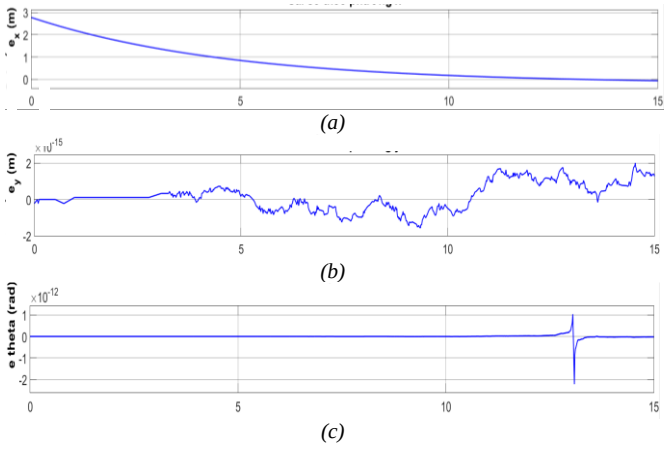


Figure 20: Tracking errors of the leading robot: (a) the tracking error in x-axis, (b) the tracking error in y-axis, (c) the direction angle error.

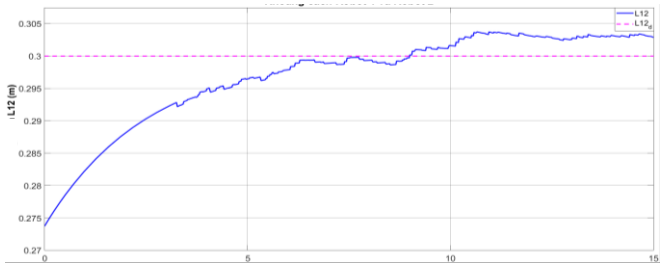


Figure 21: The distance between Robot 1 and Robot 2

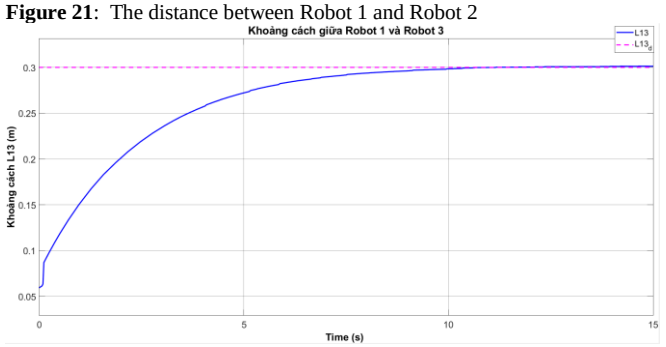


Figure 22: The distance between Robot 1 and Robot 3

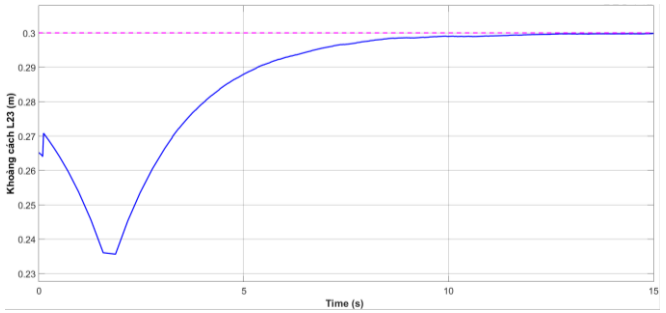


Figure 23: The distance between Robot 2 and Robot 3

Fig. 19 shows the trajectories of the reference, Robot 1, Robot 2, and Robot 3. The tracking errors $[e_x, e_y, e_{theta}]$ are demonstrated in Fig. 20. The distance between the leader robot and two follower robots is shown in Figs 19-21. From these simulation results, it is clear that the leader robot (Robot 1) can approach the reference trajectory asymptotically, the tracking errors $[e_x, e_y, e_{theta}]$ converge to zero. Figs 20-23 illustrated that the distances l_{12}, l_{13}, l_{23} approach the expected distances l_{12}^d, l_{13}^d, and l_{23}^d with very small errors.

Therefore, it concludes that the tracking trajectory and formation control algorithms operate well.

❖ Obstacle Avoidance for a leader robot

In this section, the obstacle avoidance algorithm Limit Circle is carried out for the single robot. The initial position of the leader robot is $(x_1, y_1, \theta_1) = \left(3{,}6, \frac{\pi}{4}\right)$. The position of the target G $(x_G \quad y_G) = (5.5 \quad 7.4)$. Center of the obstacle Q is $(x_Q \quad y_Q) = (4.4 \quad 7.2)$. The safety circle is established with a radius $\delta = 0.8$. The radius of the robot and obstacle are $r_r = 0.1$ and $r_o = 0.2$, respectively.

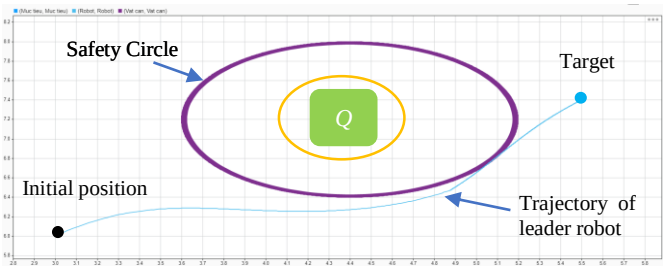


Figure 24: Obstacle avoidance of the leader robot

From Fig. 24, it is seen that the leader robot can move to the target and did not go inside the safety circle. Therefore, the leader robot can avoid obstacle Q.

❖ Obstacle Avoidance for the swarm mobile robot

For the swarm robot with three mobile robots, the Limit Circle algorithm is applied to control the swarm robot to avoid obstacles and go to the target. The Limit Circle algorithm is employed to control the leader robot (Robot 1) and two follower robots will follow with the constant distance $[l_{12}^d \quad \psi_{12}^d \quad l_{13}^d \quad l_{23}^d] = [0.15 \quad 2.365 \quad 0.15 \quad 0.296]$. The initial position is $(x_1, y_1, \theta_1) = \left(3{,}6, \frac{\pi}{4}\right)$. The position of target G is $(x_G \quad y_G) = (5.8 \quad 8)$. The Center of obstacle Q is $(x_Q \quad y_Q) = (4.4 \quad 7.2)$. The radius of the robot, obstacle, and safety circle are $[r_r \quad r_o \quad \delta] = [0.1 \quad 0.4 \quad 0.5]$.

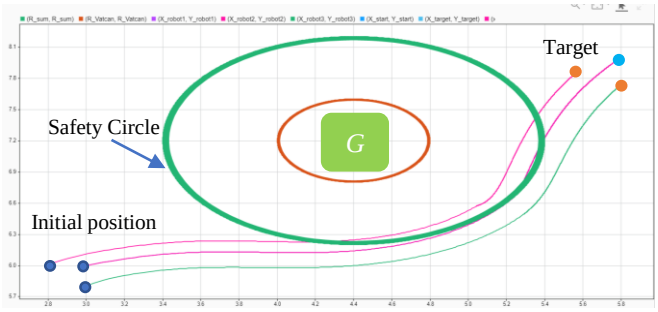


Figure 25: Obstacle avoidance of the swarm robot

Fig. 25 shows that the swarm robot with three robots maintains a triangle formation and avoids obstacles when they are moving to the target.

5.2. Experiment results

To prove the effectiveness of the proposed control methods, the swarm robot with three mobile robots is built and tested. The experiments are executed in three cases:

Case 1: Formation control
Case 2: Tracking reference trajectories with a triangle formation.
Case 3: Formation control and obstacle avoidance.

To monitor and set up parameters for the swarm robot, an Interface is built as in Fig. 26. Interface is designed for configuring formations, setting the target, monitoring the status and position of the swarm robot.

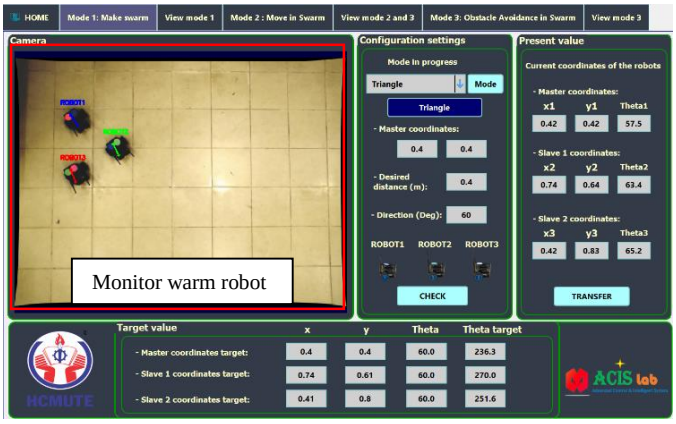


Figure 26: Interface for configuring and monitoring swarm robot.

❖ Formation Control of the swarm robot

There are three formations including vertical formation, horizontal formation, and triangle formation of the swarm robot to be tested in this paper.

Mode 1: Triangle formation

The swarm robot is set up at arbitrary positions: Leader robot $(x_1, y_1) = (1.9, 0.31)$, Robot 2 $(x_2, y_2) = (1.78, 1.31)$, and Robot 3 $(x_3, y_3) = (0.85, 1.4)$ (m) that are shown in Fig. 27. Applying the formation control and selecting the triangle formation in the Interface, three mobile robots move and arrange in the triangle formation that can be seen in Fig. 28 and the trajectories of the three mobile robots are shown in Fig. 29.

The results in Figs. 27-29 show that the formation controlling algorithm is successful to control the swarm robot arranged in a triangle formation from arbitrary positions.

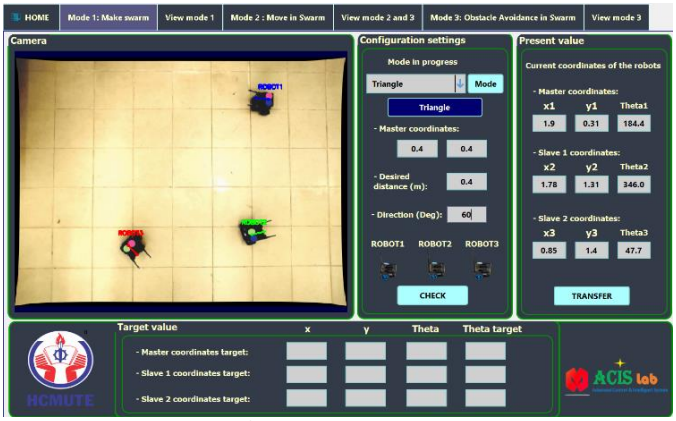


Figure 27: Initial position of the swarm robot

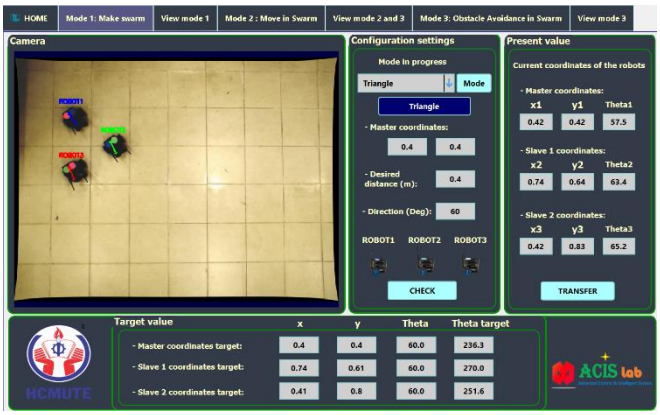


Figure 28: Triangle formation

Figure 29: Triangle formation and trajectory of swarm robot.

Mode 2: Horizontal Alignment

The initial positions of the swarm robot are $(x_1, y_1) = (0.99, 0.76)$, $(x_2, y_2) = (0.76, 1.18)$, $(x_3, y_3) = (2.08, 1.38)$ (m) that are shown in Fig. 30. The swarm robot is controlled to align horizontally in Fig. 31 with the expected distance of 0.4 (m), and the trajectories of the swarm robot are demonstrated in Fig. 32. It is seen that the experimental results in Fig. 30-32 shown that three robots of the swarm robot have the same position in Y-axis 0.4m, it means that the formation control is successful to arrange the swarm robot horizontally.

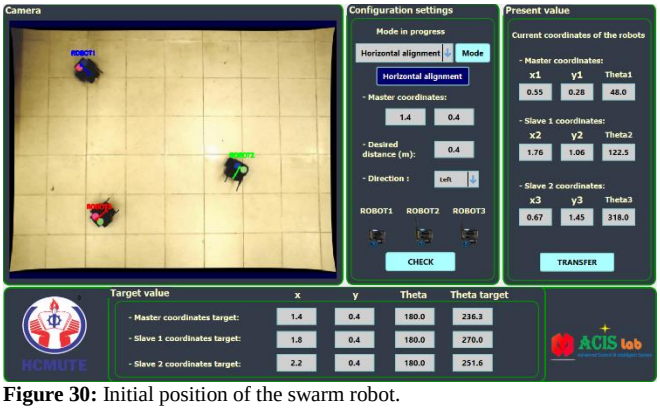


Figure 30: Initial position of the swarm robot.

Mode 3: Vertical Alignment

The initial positions of the swarm robot are illustrated in Fig. 33 with the position of Robot 1 $(x_1, y_1) = (0.99, 0.76)$, Robot 2 $(x_2, y_2) = (0.76, 1.18)$, and Robot 3 $(x_3, y_3) = (2.08, 1.38)$ (m). Formation control algorithm is employed to control the swarm robot moving in a vertical formation that is presented in Fig. 34 and their moving trajectories are shown in Fig. 35.

From the Figs. 33-35, it is easily seen that the swarm robot can arrange exactly in a vertical formation from an arbitrary position.

Figure 31: Horizontal Alignment

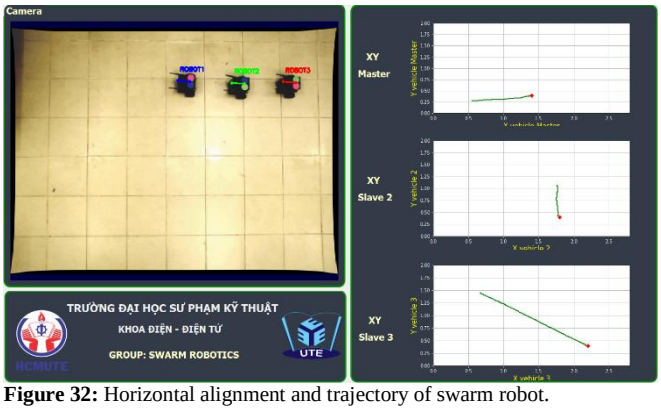


Figure 32: Horizontal alignment and trajectory of swarm robot.

Figure 33: Initial position of the swarm robot.

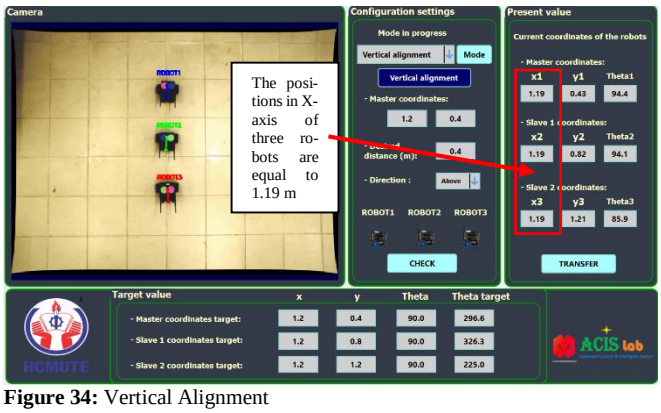


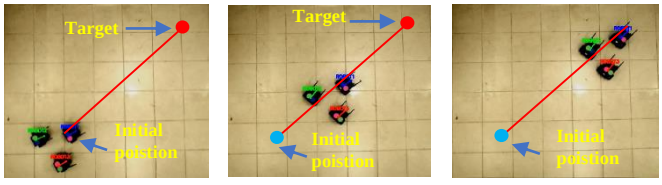
Figure 34: Vertical Alignment

Figure 35: Vertical alignment and trajectory of swarm robot.

❖ **Moving Swarm mobile robot with a triangle formation**

In this section, the swarm mobile robot is tested with moving in the triangle formation and achieved the target position. The initial position of the leader robot is $[x_1 \quad y_1 \quad \theta_1] = [1.96 \quad 0.43 \quad 309.3]$ and the position of the target is $(x_G \quad y_G) = (2 \quad 0.4)$.

The expected distances among the mobile robots of the swarm robot are $[l_{12}^d \quad \psi_{12}^d \quad l_{13}^d \quad l_{23}^d]^T = [0.24(m) \quad -147(deg) \quad 0.25(m) \quad 0.36(m)]$.



t = 0 (s) *t = 8 (s)* *t = 16(s)*

Figure 36: Swarm robot moving with the triangle formation

Figure 37: The results of the swarm robot moving in a triangle formation.

The experiments for the swarm robot moving to the goal with the triangle formation are illustrated in Figs. 36-38. From the results in Fig. 36, it is seen that the swarm robot can move in a triangle formation. Fig. 38 shows that the distance among the robots can approach the expected distances. The real distance l_{12} can track the reference distance l_{12}^d with small error $0.04(m)$. The distances l_{13},l_{23} also converge to the expected values. Therefore, we can conclude that the control algorithm can successfully control the swarm robot to move in a strangle formation.

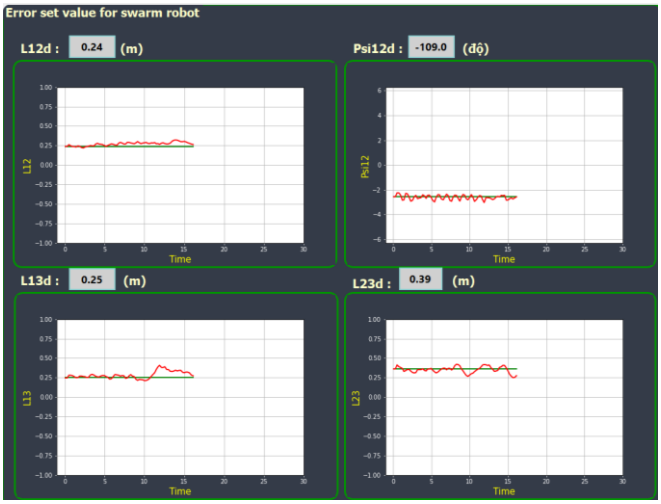


Figure 38: Expected distance among the robots.

❖ Avoidance obstacle

The results of the experiments for avoiding obstacles are demonstrated in Figs. 39-43. The initial positions of the swarm robot are $[x_1 \quad y_1 \quad \theta_1] = [0.8 \quad 0.52 \quad 38.7]$ that is shown in Fig. 39. The expected distances and angles that need to maintain our $[l_{12}^d \quad \psi_{12}^d \quad l_{13}^d \quad l_{23}^d] = [0.26(m) \quad 257.7(deg) \quad 0.29(m) \quad 0.45(m)]$. The position of the target G is $(x_G \quad y_G) = (2 \quad 1.4)\ (m)$. Based on the image processing technique, the center of the obstacle Q is determined $(x_Q \quad y_Q) = (1.5 \quad 0.7)\ (m)$. The radius of the robot, obstacle, and safety circle are $[r_r \quad r_o \quad \delta] = [0.15 \quad 0.15 \quad 0.2]\ (m)$.

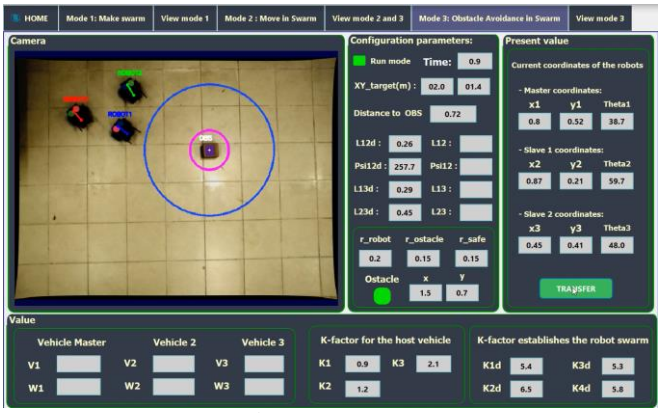


Figure 39: Initial positions of the swarm robot.

Figure 40: Results of the obstacle avoidance of the swarm robot.

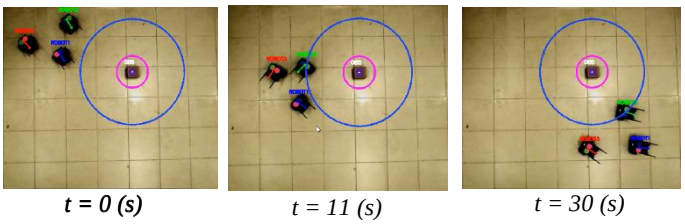


Figure 41: Avoidance Obstacle moving of the swarm robot.

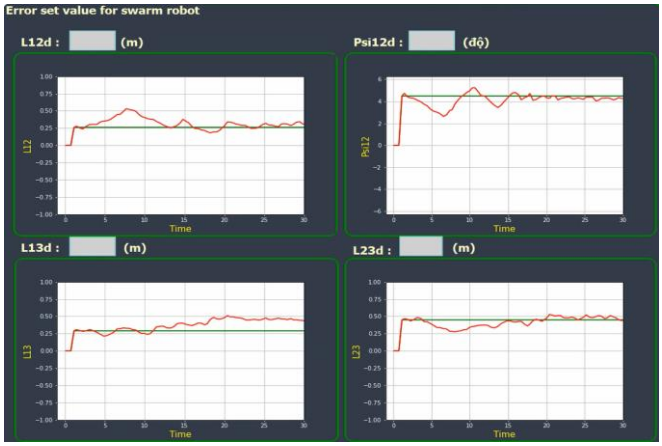


Figure 42: The real distances and the expected distances among robots.

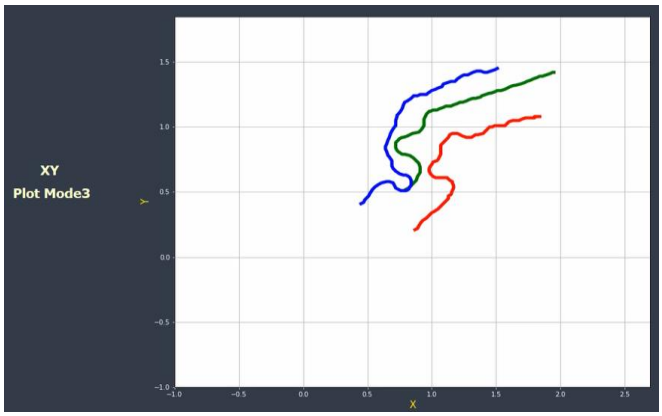


Figure 43: The trajectories of the three robots.

For avoidance obstacles, the experimental results are shown in Figs. 39-43. In Fig. 41, we consider the experiment at three points: start point (0 s), 11 s (avoiding obstacle action is taking place), and 30 s (swarm mobile robot successfully avoids the obstacle), it is seen that when the system detects the obstacles Q, it will create a safety circle with radius $\delta = 0.2\ (m)$ that is larger than the radius of the obstacle. On the basis of the Limit-Circle algorithm, the leader mobile robot (Blue color) will move outside of the safety circle and leads the swarm mobile robots to avoid colliding with obstacle Q. Moreover, it is also seen that the swarm mobile robot can avoid the obstacle during moving and keep the triangle formation to reach the target $(x_G \quad y_G) = (1.95 \quad 1.42)$; Fig. 42 illustrates that the swarm robot can maintain the distances among the robots during avoiding the obstacle. Hence, the algorithm control of this paper successfully controls the swarm robot both moving in a specific formation and avoiding obstacles.

6. Conclusion

The swarm robot with three mobile robots has been designed and implemented in this paper. Image processing technique is employed to determine the positions of each robot of swarm robot and detect the obstacle successfully. The controlling algorithm for tracking the trajectories, avoidance of obstacles, and formation control are studied and applied successfully for this swarm mobile robot. The simulation and experimental results proved that the swarm robot can track the reference trajectories, move in the specific formation, and avoid obstacles successfully.

Acknowledgment

This work is supported by the Ho Chi Minh City University of Technology and it is executed at Advanced Control and Intelligent System Lab of Ho Chi Minh City University of Technology and Education (ACIS lab).

References

[1] Marino, A., Parker, L.E., Antonelli, G. et al. "A Decentralized Architecture for Multi-Robot Systems Based on the Null-Space-Behavioral Control with Application to Multi-Robot Border Patrolling" *J. Intell Robot Syst* 71, pp.423–444, 2013.

[2] B. Shabaninia, A. Nemati, and S.-D. Stan, "A novel robust decentralized adaptive fuzzy control for swarm formation of multiagent systems," *IEEE Trans. Indust. Electr*., vol. 59, no. 8, pp. 3124–3134, 2012.

[3] D. D. Xu, X. N. Zhang, Z. Q. Zhu, C. L. Chen, and P. Yang "Behavior-Based Formation Control of Swarm Robots" *Mathematical Problems in Engineering*, vol. 2014, Article ID 205759, 13 pages, 2014.

[4] O. G. Miller and V. Gandhi, "A survey of modern exogenous fault detection and diagnosis methods for swarm robotics" *Journal of King Saud University – Engineering Sciences*, vol. 33, pp. 43–53, 2021.

[5] Q. Tang, F. Yu, Z. Xu, and P. Eberhard, "Swarm Robots Search for Multiple Targets," in *IEEE Access*, vol. 8, pp. 92814-92826, 2020.

[6] J. Zhang, Y. Lu, L. Che and M. Zhou, "Moving-Distance-Minimized PSO for Mobile Robot Swarm," in *IEEE Transactions on Cybernetics*, vol. 52, no. 9, pp. 9871-9881, Sept. 2022.

[7] A. B. Migranov and O. V. Darintsev, "Choosing a Swarm Algorithm to Synthesis an Optimal Mobile Robot Team Control Strategy," *2020 International Multi-Conference on Industrial Engineering and Modern Technologies (FarEastCon)*, Vladivostok, Russia, 2020, pp. 1-5.

[8] J. P. Desai, J. P. Ostrowski, and V. Kumar, "Modeling and control of formations of nonholonomic mobile robots", *IEEE Trans. Robot. Automat*, vol. 17, pp. 905–908, Dec. 2001.

[9] T. Dierks and S. Jagannathan, *"*Control of Nonholonomic Mobile Robot Formations: Backstepping Kinematics into Dynamics*", 16th IEEE International Conference on Control Applications, October 2007.*

[10] Q. Wang, J. Huang and X. Mao, "A Fast Self-Organizing Pattern Formation Method for Swarm Robots in Dynamic Multi-Region Environments," *2020 5th International Conference on Automation, Control and Robotics Engineering (CACRE)*, Dalian, China, 2020, pp. 124-129.

[11] Z. Qiao, J. Zhang, X. Qu and J. Xiong, "Dynamic Self-Organizing Leader-Follower Control in a Swarm Mobile Robots System Under Limited Communication," *in IEEE Access*, vol. 8, pp. 53850-53856, 2020.

[12] Z. Jiang, X. Wang and J. Yang, "Distributed Line Formation Control in Swarm Robots," *2018 IEEE International Conference on Information and Automation (ICIA)*, Wuyishan, China, 2018, pp. 636-641.

[13] S. Morgan and J. Hereford, "Path formation using a robot swarm with limited sensing capabilities," *2020 SoutheastCon*, Raleigh, NC, USA, 2020, pp. 1-6, 2020.

[14] A. K. Das, R. Fierro, V. Kumar, J. P. Ostrowski, J. Spletzer and C. J. Taylor, "A vision-based formation control framework," in *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 813-825, Oct. 2002.

[15] R. Lan, X. Wang, Z. Xiao, R. Wang, Y. Lin and R. Fang, "Design of Swarm Robots Formation Control System Based on Vision Guidance," *2019 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, Hong Kong, China, 2019, pp. 790-795.

[16] J. Guan, W. Zhou, S. Kang, Y. Sun and Z. Liu, "Robot Formation Control Based on Internet of Things Technology Platform," *in IEEE Access*, vol. 8, pp. 96767-96776, 2020.

[17] D. H. Kim, J.-H. Kim, *"*A real-time limit-cycle navigation method for fast mobile robots and its application to robot soccer*", Robotics and Autonomous Systems*, vol. 42, no. 1, pp. 17-30, January 2003.

[18] Mehdi Mouad, Lounis Adouane, *"*Mobile Robot Navigation and Obstacles Avoidance based on Planning and Re-Planning Algorithm'', in *10th IFAC Symposium on Robot Control International Federation of Automatic Control*, September 2012.

[19] Lounis Adouane, *"Orbital Obstacle Avoidance Algorithm for Reliable and On-Line Mobile Robot Navigation",* in 9th Conference on Autonomous Robot Systems and Competitions, May 2009.

[20] Li Y, Gao J, Su X, and Zhao J. "Cooperation control of multiple miniature robots in unknown obstacle environment". *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering, vol. 229, no. 3, pp. 202-214, 2015.*

[21] X. Peng, *"*Formation Control of Multiple Nonholonomic Wheeled Mobile Robots", *Centrale de Lille*, September, 2013.

[22] R. Fierro and F. L. Lewis, "Control of a Nonholonomic Mobile Robot: Backstepping Kinematics into Dynamics", *Journal of Robotic Systems*, vol. 14, no. 3, pp. 149-227, March, 1997.

[23] D. Agarwal and P. S. Bharti, "MATLAB Simulation of Path Planning and Obstacle Avoidance Problem in Mobile Robot using SA, PSO and FA," *2020 IEEE International Conference for Innovation in Technology (INOCON)*, Bangluru, India, 2020, pp. 1-6.

[24] S. I. A. Meerza, M. Islam and M. M. Uzzal, "Q-Learning Based Particle Swarm Optimization Algorithm for Optimal Path Planning of Swarm of Mobile Robots," *2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT)*, Dhaka, Bangladesh, 2019, pp. 1-5.

[25] Q. Tang, F. Yu, Z. Xu and P. Eberhard, "Swarm Robots Search for Multiple Targets," in *IEEE Access*, vol. 8, pp. 92814-92826, 2020.

[26] T. Zhang, J. Xu and B. Wu, "Hybrid Path Planning Model for Multiple Robots Considering Obstacle Avoidance," in *IEEE Access*, vol. 10, pp. 71914-71935, 2022.