

Semi-decentralized Lyapunov-based formation control of multiple omnidirectional mobile robots

Hendi Wicaksono Agung, Fransisco Jordan

Department of Electrical Engineering, Faculty of Engineering, University of Surabaya, Surabaya, Indonesia

Article Info

Article history:

Received Jan 27, 2024

Revised Mar 20, 2024

Accepted Mar 30, 2024

Keywords:

Cooperative mobile robots

Formation maintenance

Lyapunov-based control

Multi-robot system

Object transport

Semi-decentralized

ABSTRACT

This paper introduces an advanced formation control algorithm based on a Lyapunov approach for coordinating multiple omnidirectional mobile robots in collaborative object transport tasks. The semi-decentralized strategy ensures that the robots maintain a predefined geometric formation, crucial for stability during material transportation, and dynamically adapt to avoid collisions using onboard sensors. Experimental with a physical robot simulator demonstrates successful maintenance of line and triangle formations achieving an average side length maintenance of 1.00 meters with minimal deviation. Quantitative analysis across 30 experimental runs reveals consistent performance, with a maximum side length fluctuation of only 2 centimeters, validating the effectiveness of maintaining formation within a multi-robot system (MRS) framework. The Lyapunov-based approach proves to be an efficient method for cooperative object transport, achieving consistent performance with minimal deviation.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Hendi Wicaksono Agung

Department of Electrical Engineering, Faculty of Engineering, University of Surabaya

Surabaya, Indonesia

Email: hendi@staff.ubaya.ac.id

1. INTRODUCTION

Human beings, as social creatures, frequently encounter challenges that exceed their capabilities. When a person needs to move an object larger or heavier than they can manage alone, collaboration with others becomes essential. This often involves coordination and teamwork to successfully transport the large item to its intended location. Similar collaborative efforts are observed in animal groups, such as ants, where individuals work together with their fellow group members to transport food back to their nest [1], [2].

In the industrial sector [3], there is a growing shift towards using automated mobile robots rather than human-operated heavy-material vehicles such as forklifts [4]. Automated guided vehicles (AGV) are mobile robots that transport materials while following guidance systems like magnetic tape [4], whereas autonomous mobile robots (AMR) can carry materials without guidance [5], [6]. The size or weight of the material determines the type and size of AGV or AMR used for transportation [7]. For instance, smaller materials can be transported by small vehicles, while larger and heavier items require larger vehicles.

Using a single large AGV necessitates substantial space for maneuvering, intricate mechanics, and a significant supply of energy to operate the machinery. Complex mechanics also result in lengthy maintenance and repair processes [8]. Additionally, due to its sizable dimensions, ample storage space is essential when not in use. In contrast, employing multiple vehicles in the form of a multi-robot system (MRS) offers advantages such as uncomplicated mechanics, reduced energy consumption, cost-effectiveness, and minimal storage requirements [8]. However, successful implementation of the MRS strategy relies on the capability to navigate

without external guidance like magnetic tape [9]. As a result, an autonomous mobile robot was subsequently considered as a prototype model within this research endeavor.

The MRS utilizes different strategies for object transport, such as pushing, caging, and grasping [10]. However, pushing and caging strategies can be detrimental to industrial material transportation due to the risk of damage to the materials. On the other hand, the grasping strategy is not suitable for heavy items as it can be challenging to achieve a proper grasp. Instead, the MRS has adopted the on-top loading method, where materials are placed on top of the robots for efficient and safe transportation [11].

In order to transport materials by placing them on the body of a robot, it is necessary to ensure rigid coordination, with each robot moving in the same direction and at the same time to prevent the material from falling and being damaged. The challenge of material transportation is analogous to that of formation control, which is concerned with maintaining the behavior and shape of a group of robots. Formation control involves three main components: aggregation, keeping-formation, and geometry switching. The aggregation process involves the initial formation of the desired shape from random starting positions of the robots [12], [13]. Once the formation is established, it must maintain its shape during the journey to the target, even when encountering obstacles. The shape of the formation is heavily dependent on the number of robots involved and can take various forms such as lines, columns, triangles, squares, diamonds, quadrilaterals, V-shapes, and wedges [14]. If the formation encounters a narrow corridor that only allows a subset of the robots to pass through, then a geometry switch is required to adapt and continue to the target [15], [16]. However, it is important to note that in applications involving material transport, geometry switching should be avoided to prevent the risk of the transported material falling and becoming damaged [17].

The control strategy for maintaining formation in MRS can be approached through both centralized and decentralized control methods [18]. The choice between centralized and decentralized control depends on factors such as communication range, sensing capability, cost, resources, and system size. Using a centralized control approach, all robots in a MRS are controlled by a central unit [19], [20]. While in a decentralized control approach, each robot makes decisions based on its own local information and the information it receives from neighboring robots [18], [19]. The decentralized control approach offers advantages such as low operational costs, high adaptability, and robustness, while the centralized control approach allows for more coordinated and efficient control of the entire team. To achieve a balance between the two control approaches and benefit from their respective advantages, a semi-decentralized approach is proposed in this paper. The semi-decentralized approach aims to leverage the benefits of both the centralized and decentralized control methods, taking into consideration factors such as communication range, sensing capabilities, and system requirements.

The proposed method utilizes the Lagrangian swarm model [21], which was originally designed for non-holonomic systems and validated through computer simulations [22]. To address the current industry requirements for mobile robots capable of maneuvering freely and efficiently carrying materials, the Lagrangian swarm model has been adapted for holonomic models, specifically holonomic omniwheel AMRs. In this research, the holonomic model of an omnidirectional mobile robot is utilized. The holonomic model enables the robot to move in any direction with ease [23], making it suitable for various applications such as indoor navigation, and warehouse automation. By employing this model, the aim is to enhance the maneuverability and agility of the mobile robot, ultimately contributing to advancements in the field of robotics.

The effectiveness of the proposed algorithm was tested through computer simulations, but limitations were identified in terms of realism. To address this limitation, the proposed algorithm was further validated through experimental testing using a robot simulator with a physical plugin, called Webots simulator [24]. Webots is a free and open-source robotics simulator created by Ecole Polytechnique Fédérale de Lausanne (EPFL) in Switzerland. This platform is widely used in the industrial, educational, and research sectors [25]. It uses the open dynamics engine (ODE) to simulate the dynamics of rigid bodies and collisions [26]. The use of a physical plugin in the robot simulator reduces the disparity between the simulation and actual robot reality, enhancing the potential for successful implementation in real-world applications [24].

The paper presents the following main contributions: i) The development of a swarm model for holonomic omniwheel AMR that allows the robot to maintain formation and avoid obstacles in unknown environments; ii) The proposed algorithm is evaluated using a robot simulator with a physical plugin, which provides a more realistic evaluation of the algorithm's performance; and iii) The algorithm's stability is verified through the development of mathematical models and physical simulation experiments based on the motion of omniwheel mobile robots.

In the rest of this paper, the second section will provide a comprehensive description of the proposed

method, encompassing the design of the swarm control system for holonomic mobile robots, the models of omniwheels AMR, and the settings of the Webots environmental simulator. The third section will focus on the performance evaluation, providing detailed information on the experimental scenarios of the robots' formation in response to these scenarios, and evaluating the experimental results. Finally, in the fourth section, summarize the conclusions drawn from the proposed method and discuss potential prospects for further research in this field.

2. PROPOSED METHOD

2.1. System descriptions

In this research, a holonomic model featuring an omnidirectional mobile robot with three omniwheels was utilized. The three omniwheels mobile robot consists of three omniwheels arranged in a triangular configuration. Each wheel is capable of moving independently in a lateral direction, allowing the robot to move in any direction without changing its orientation. The design of the three-wheel drive (3WD) mobile robot is illustrated in Figure 1, presenting its construction in Figure 1(a) and driving instruction in Figure 1(b).

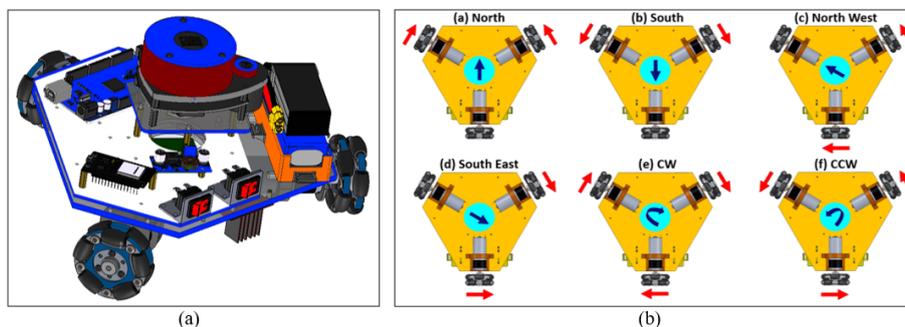


Figure 1. Robot 3WD using omniwheels (a) mechanical design and (b) driving instructions

Kinematic model. The initial step in the design process is to develop the model of the 3WD mobile robots. The kinematic model plays a crucial role in comprehending the robot's motion and agility, as it articulates the correlation between its wheel velocities and overall motion within a specific direction. This involves establishing relationships among the robot's linear and angular speeds, along with those of its individual wheels by taking into account factors such as wheel orientation, angles, and physical structure to formulate equations that govern the robot's movement.

The robot's coordinate vector, $q = [q_1 \ q_2 \ q_3]^T$, is used to calculate its motion, and its derivative, \dot{q} , is used to calculate the vector of velocity on the world's axis. The coordinate frame of the 3WD mobile robot is shown in Figure 2. The vector of velocity above can be converted into a velocity on the robot's frame using the in (2.1.).

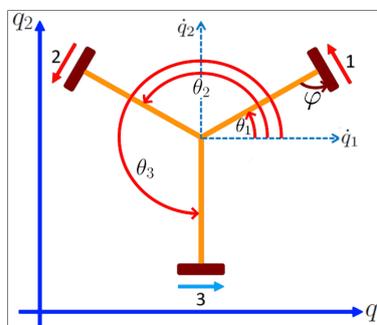


Figure 2. System coordinates of the 3WD omnidirectional mobile robot

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = \begin{bmatrix} c(\theta_1 + \varphi) & c(\theta_2 + \varphi) & c(\theta_3 + \varphi) \\ s(\theta_1 + \varphi) & s(\theta_2 + \varphi) & s(\theta_3 + \varphi) \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \quad (1)$$

$s(\bullet) = \sin(\bullet)$, $c(\bullet) = \cos(\bullet)$

Where v_1, v_2 , and v_3 are velocities of the first-, second-, and third wheel, respectively. The values of $\varphi, \theta_1, \theta_2$ and θ_3 are $90^\circ, 30^\circ, 150^\circ$, and 270° , respectively.

2.2. Lyapunov function components

Next, a Lagrangian swarm model is developed to obtain the motion of the 3WD mobile robot. The Lagrangian swarm model (position, velocity, and acceleration state) provides a comprehensive understanding of the motion and behavior of the 3WD mobile robots. The Lyapunov direct method, adopted from [22], is used to derive the instantaneous velocity in obstacle-filled environments, serving as the autonomous controller for each individual. Take into account a cooperative of $n \in \mathbb{N}$ mobile robots as point masses. For every $i \in \{1, 2, \dots, n\}$, the position of the mobile robot should be $q_{Ri} = (q_{1Ri}, q_{2Ri})$, with $q(0) = (q_{1Ri}(0), q_{2Ri}(0))$ as the starting position. Let's characterize the cooperative mobile robots as (2).

$$q_c = \left(\frac{1}{n} \sum_{k=1}^n q_{1k}, \frac{1}{n} \sum_{k=1}^n q_{2k} \right) \quad (2)$$

Then, $(v_{Ri}, w_{Ri}) = (\dot{q}_{1Ri}, \dot{q}_{2Ri})$ be the instantaneous velocity of the i th mobile robot. Let $q_{Ri} = (q_{1Ri}, q_{2Ri}) \in \mathbb{R}^2$, and $q = (q_1, q_2, \dots, q_n) \in \mathbb{R}^{2n}$ be the state vectors. If the velocity (v_{Ri}, w_{Ri}) has a state feedback law of the form $(v_{Ri}, w_{Ri}) = (-\mu_{Ri} f_{Ri}(q), -\varphi_{Ri} g_{Ri}(q))$, for $\mu_{Ri}, \varphi_{Ri} > 0$ and functions $f_{Ri}(q)$ and $g_{Ri}(q)$, which will be designed appropriately later. If $g_{Ri}(q) = (-\mu_{Ri} f_{Ri}(q), -\varphi_{Ri} g_{Ri}(q)) \in \mathbb{R}^2$ and $G(q) = (g_1(q), g_2(q), \dots, g_n(q)) \in \mathbb{R}^{2n}$, then the cooperative of n mobile robots can be seen in (3).

$$\dot{q} = G(q), q_{t=0} = q(0) \quad (3)$$

If the system has an equilibrium point and is denoted by $q_e = (q_{e1}, q_{e2}, \dots, q_{en}) \in \mathbb{R}^{2n}$. The stability of q_e will be analyzed by the direct method of Lyapunov.

Attraction to the formation's center. Based on the idea that the cooperative mobile robot must maintain formation for efficient material transport, the attraction function is part of the Lyapunov function of the system. The attraction function is the measurement of the distance between a mobile robot and the formation's center, as specified as (4).

$$R_{Ri}(q) = \frac{1}{2} [(q_{1Ri} - q_{1c})^2 + (q_{2Ri} - q_{2c})^2] \quad (4)$$

The goal of the formation of $n \in \mathbb{N}$ robots. The goal of formation is a disk with a radius of r_τ and a center of (a, b) . The components of the system's Lyapunov function for the target attraction function, which weight the distance between the system's center and the target, are as (5).

$$T(q) = \frac{1}{2} [(q_{1c} - a)^2 + (q_{2c} - b)^2] \quad (5)$$

Inter-robots collision avoidance. The i th and j th mobile robots repel one another over short distances, $j \neq i, i, j \in \{1, 2, \dots, n\}$, the function described as (6):

$$Q_{Rij}(q) = \frac{1}{2} [(q_{1Ri} - q_{1Rj})^2 + (q_{2Ri} - q_{2Rj})^2 - (2r_a)^2] \quad (6)$$

where the radius of the disk containing the mobile robot is represented by r_a .

Static obstacles avoidance. Let the system configuration space (2.2.) clustered with static obstacles $m \in \mathbb{N}$. The function of obstacle avoidance for the i th mobile robot was implemented to prevent a potential collision with the k th static obstacle, where $i \in \{1, 2, \dots, n\}$ and k is in the range of $\{1, 2, \dots, m\}$.

$$W_{Rik}(q) = \frac{1}{2} [(q_{1Ri} - o_{k1})^2 + (q_{2Ri} - o_{k2})^2 - (r_{ok} + r_a)^2] \quad (7)$$

Where the obstacle's radius is r_{ok} .

A tentative Lyapunov function. Given the variables $\alpha > 0, \gamma_{Ri} > 0, \beta_{Rij} > 0$ and $\lambda_{Rik} > 0$, describe a possible Lyapunov function for system (2.2.) for $i, j \in \{1, 2, \dots, n\}$,

$$L(q) = \alpha T(q) + \sum_{i=1}^n T(q) \left(\gamma_{Ri} R_{Ri}(q) + \sum_{j=1, j \neq i}^n \frac{\beta_{Rij}}{Q_{Rij}(q)} + \sum_{k=1}^m \frac{\lambda_{Rik}}{W_{Rik}(q)} \right) \quad (8)$$

the Lyapunov function's time derivative along the system's trajectories (2.2.) is (9).

$$\dot{L}(q) = \sum_{i=1}^n [f_i(q)v_{Ri} + g_i(q)w_{Ri}] \quad (9)$$

Due to page limitations, the specifics of $f_i(q)$ and $g_i(q)$ can be found in [22].

Velocity controllers. If $\mu_{Ri} > 0$ and $\varphi > 0$ are constants, then the instantaneous velocity components of the system (2.2.) are defined as $v_{Ri} = -\mu_{Ri}f_i(q), w_{Ri} = -\varphi_{Ri}g_i(q)$. Then,

$$\dot{L}(q) = - \sum_{i=1}^n \left[\mu_{Ri} (f_i(q))^2 + \varphi_{Ri} (g_i(q))^2 \right] = - \sum_{i=1}^n \left[\frac{v_{Ri}^2}{\mu_{Ri}} + \frac{w_{Ri}^2}{\varphi_{Ri}} \right] \leq 0 \quad (10)$$

Stability analysis. Velocities v_{Ri} and w_{Ri} are zero at the goal, where $(q_{1c}, q_{2c}) = (a, b)$, since $f_i = 0$ and $g_i = 0$. As a result, robots continue to align themselves consistently around the goal. They thus form part of an equilibrium q_e system (2.2.) because of their fixed placements. It is easy to see that $L(q_e) = 0, L(q) > 0 \forall q \neq q_e$, and $\dot{L}(q) \leq 0$. It follows that the Lyapunov function (2.2.) is the one that ensures the stability of system (2.2.).

2.3. Webots simulator

As mentioned earlier, the Webots simulator offers a distinct advantage compared to typical computer simulations. It includes a physics plugin that enables users to fine-tune the results to closely match real-world implementation. This capability delivers a more precise portrayal of the simulated system's behavior, making it an invaluable tool for testing and improving designs prior to physical implementation.

Webots provide sensors like proximity, light, touch, force, gyroscope, accelerometer, camera, compass, light detection and ranging (LiDAR), and global positioning system (GPS) sensors [26]. Using the information provided in the Lyapunov function in (9) and the velocity controller in (10), the Webots simulator requires inputs such as the target coordinate, the coordinates of all mobile robots, and obstacle coordinates. This research uses a LiDAR sensor to detect the distance from other robots and obstacles. Then, based on these measurements, it calculates the relative coordinate of surrounding objects, including neighboring robots and obstacles. Using these relative coordinates, collision avoidance algorithms and path planning strategies will be implemented in the Webots simulator based on the Lyapunov function. While the target coordinate is determined using GPS sensors. Also, Webots provides an interactive virtual world based on virtual reality modeling language (VRML) that supports programming in MATLAB, Java, C, C++, Urbi, or Python [26]. The choice for this research is to use the C language because it has been familiar for a long time.

Webots scene model. The Webots GUI, depicted in Figure 3, consists of four main windows. The scene Tree interface on the left provides an illustrative hierarchical structure of the current environment. In the middle, the 3D simulation window allows for interactive exploration. The text editor interface on the right enables users to update and submit code compilations, while at the bottom, console interfaces present output from each compilation and controller. A world file contains a hierarchical representation of a world's contents, including object locations, appearances, interactions, sky color, and gravity vectors' positions. Other details for the environment simulation setup can be found in the previous research [27].

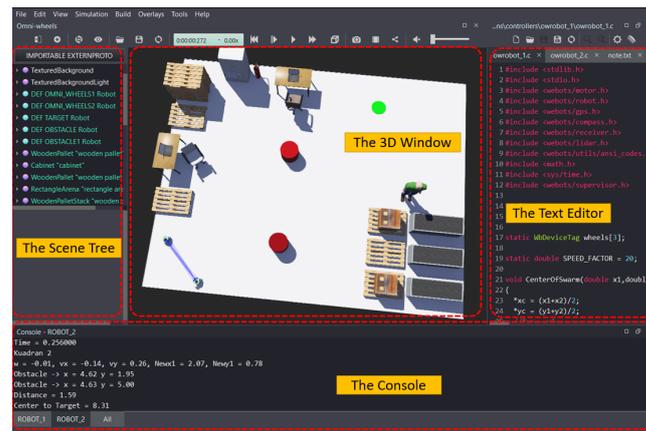


Figure 3. The Webots simulator's user interface.

Webots controller. Webots controller files are generated by using equations to describe the movements of an omnidirectional mobile robot. The “Controller” file in the application allows for defining the robot's behavior, including setting up nodes, motors, time control variables, and user-defined variables. Each robot is equipped with a C-programmed controller, facilitating two-way interaction without any supervisory control. The flowchart depicted in Figure 4 illustrates the semi-decentralized controller setup for robots 1 and 2 in line formation.

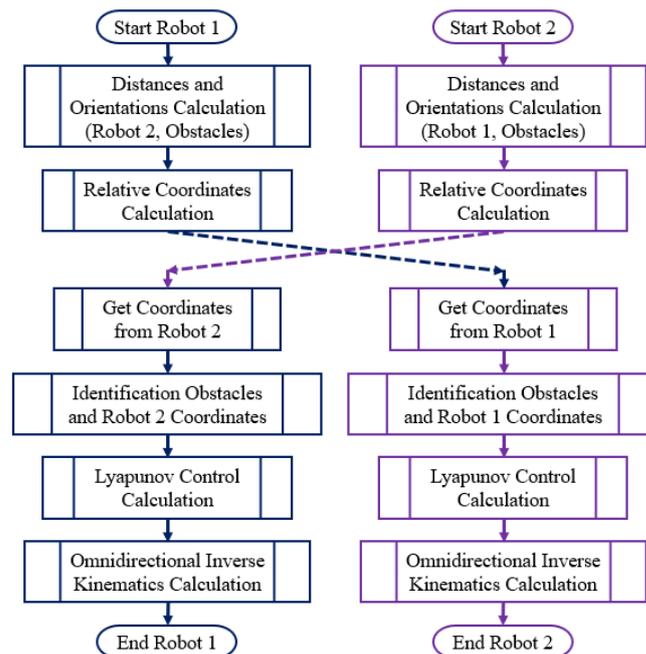


Figure 4. Robots 1 and 2's semi-decentralized controller's flowchart in line formation

2.4. Inverse kinematics of omniwheel robot

The Lyapunov function approach is used to calculate each robot's trajectory plan. The data required to run trajectory computations using the Lyapunov function approach are the present coordinates of each robot, target coordinates, and the coordinates of obstacles. The consequence of this procedure is a change in the robot coordinates \dot{q} . The inverse kinematics equation of the Omniwheel robot is utilized to achieve the stated velocities to reach the goal coordinates. From (2.1.), to make the calculation for inverse kinematics simpler,

let's assume that matrix A is as (11).

$$[A] = \begin{bmatrix} \cos(\theta_1 + \varphi) & \cos(\theta_2 + \varphi) & \cos(\theta_3 + \varphi) \\ \sin(\theta_1 + \varphi) & \sin(\theta_2 + \varphi) & \sin(\theta_3 + \varphi) \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} -\frac{1}{2} & -\frac{1}{2} & 1 \\ \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} & 0 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \quad (11)$$

The (12)-(14) is the inverse kinematics.

$$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}^{-1} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \quad (12)$$

$$[A]^{-1} = \frac{\begin{pmatrix} ei - fh & hc - ib & bf - ce \\ gf - di & ai - gc & dc - af \\ dh - ge & gb - ah & ae - db \end{pmatrix}}{aei + bfg + cdh - ceg - afh - bdi} \quad (13)$$

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} -0.33 & 0.58 & 0.33 \\ -0.33 & -0.58 & 0.33 \\ 0.67 & 0 & 0.33 \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} \quad (14)$$

The velocity of each motor is successfully specified using (2.4.) and (2.2.).

3. RESULTS AND DISCUSSION

This study investigates semi-decentralized Lyapunov-based formation control for multiple omnidirectional mobile robots in cooperative object transport applications. It focuses on maintaining formation geometry for the robots even when they encounter obstacles, using a formation maintenance solution based on the Lyapunov method. The performance of this control method is tested through computer simulations with physical plugins in configurations like line and triangle formations to ensure that robots can preserve their arrangement while moving toward a target and avoiding collisions. Earlier studies have investigated the use of a Lyapunov-based control strategy to maintain the formation of multiple non-holonomic mobile robots through computer simulation. However, they have not explicitly addressed the complexities of maintaining formation in real-world environments with required sensors.

The computer simulation begins with the aggregation of N robots from arbitrary points to the starting point, forming a preset geometric shape that matches the transport object. Once the material/object is placed on top of the robots, target location information and knowledge of the environment are transmitted to all robots to assist them in avoiding obstacles. The proposed method is tested using two geometric formations: a two-robot line formation and a three-robot triangle formation.

3.1. Line formation with two omniwheel robots

First of all, simulation involves two omniwheel robots, denoted as q_{R1} and q_{R2} , that form a straight line formation with an equilibrium distance reached by values of local gain. The Lyapunov-based controller of each robot is designed with gain $\alpha = 40$, $\gamma_{Ri} = 55$, $\beta_{Rij} = 1.52$, and $\lambda_{Rik} = 1.2$, which is the gain for each Lyapunov function. The initial positions of the robots are $q_{R1}(t_0) = [0.9, 1.75]^T$ and $q_{R2}(t_0) = [2.25, 0.5]^T$, respectively. Two static obstacles are located at $[4.7, 1.95; 4.6, 5]^T$. Formation maintenance is measured by the distances between two mobile robots during travel. The simulation environment is depicted in Figure 5.

Aggregation to line formation. Throughout the aggregating process, the two robots R1 and R2 travel from their initial points to the starting positions of the formation, forming a straight line with a relative side length of 1 m. The locations of the mobile robots after aggregation are $q_{R1}(t_f) = [1.2, 1.46]^T$, and $q_{R2}(t_f) = [2.04, 0.9]^T$. It is noted that larger values of attraction to the center gain result in faster aggregation time. Once the initial configuration is constructed, the group may start moving toward the target point, which is governed by attraction to the target gain and attraction to the center gain.

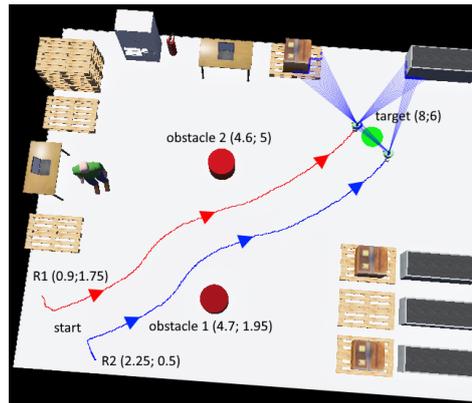


Figure 5. A simulation environment was used to test the algorithm

Formation maintenance with line formation. Formation control Lyapunov-based is implemented to simultaneously maintain formation while traveling to the target. The robot formation's trajectory as it goes from the starting position to the target position is presented in Figure 6. Initially, both mobile robots are able to maintain a relative side length of 1 m. However, as q_{R2} approaches an obstacle and begins to conduct collision avoidance steps, the relative side length between q_{R1} and q_{R2} affected. The analysis of line formation performance involved two measurements, as shown in Figure 7. Figure 7(a) shows the distances between robots, while Figure 7(b) shows the corresponding wheel speeds. To check the efficiency of the proposed method, the experiments were conducted 30 times, and the results are presented in Table 1.

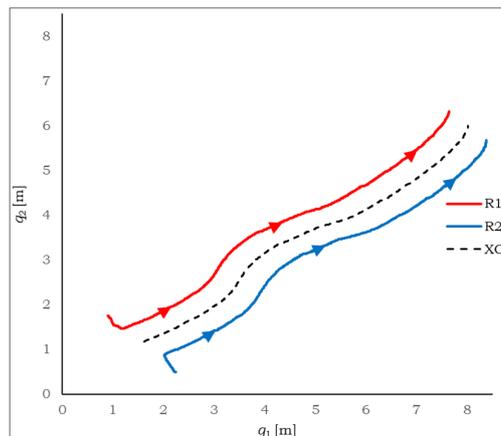


Figure 6. Formation-keeping two omnihwheel robots in line formation in Cartesian space

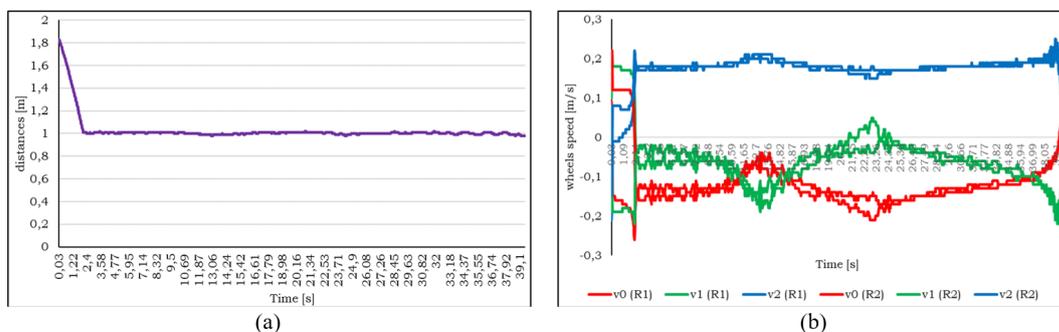


Figure 7. Line formation (a) robot's distances and (b) wheel speeds of each robot

Table 1. The travel time and distances of 30 experiments on a line formation

Parameters	Min	Average	Max
Side length [m]	0.98	1.00	1.02
Travel time [s]	39.58	39.58	39.58

The proposed controller in this study, along with the designed gain, can achieve optimum performance in side length. The average side length is 1.00 m, with minimum and maximum values of 0.98 and 1.02 m respectively. This indicates that the formation only stretches and shrinks by a mere 2 cm, leading to the conclusion that the formation was successfully maintained.

3.2. Triangle formation with three omniwheel robots

The proposed controller’s scalability is analyzed using a triangle arrangement of three robots, denoted as q_{R1} , q_{R2} , and q_{R3} . The initial locations of the mobile robots are $q_{R1}(t_0) = [1.5, 2]^T$, $q_{R2}(t_0) = [2.5, 1]^T$, and $q_{R3}(t_0) = [1.4, 0.9]^T$, respectively. Two static obstacles are assumed to be located at $[4.62, 1.95; 4.63, 4.8]^T$. The Lyapunov-based controller of each mobile robot is designed with gain $\alpha = 150$, $\gamma_{Ri} = 200$, $\beta_{Rij} = 1$, and $\lambda_{Rik} = 1.2$, which is the gain for each Lyapunov function.

Triangle formation aggregation: the robots move from their starting locations to the formation’s beginning point during the aggregation phase, constructing a triangle with a relative side length of 0.85 m. Following the completion of the aggregation phase, the group may begin to move in the direction of the target, utilizing a triangle formation under Lyapunov-based control.

Formation maintenance with triangle formation: the proposed Lyapunov-based controller is evaluated by considering the simultaneous formation-keeping performance as the mobile robots move toward the target location. The robot formation’s trajectory as it moves from the start to the target position is shown in Figure 8.

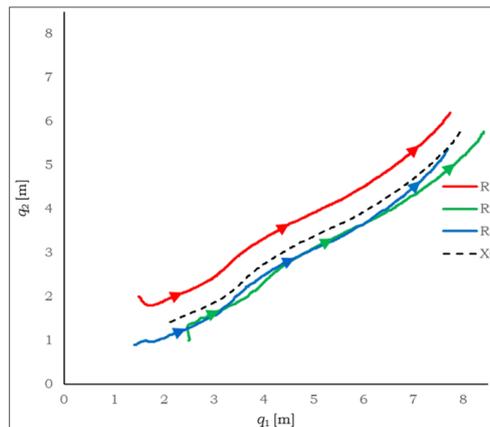


Figure 8. Formation-keeping of three omniwheel robots in the triangle formation in Cartesian space

Two measurements were taken to analyze the performance of triangle formation, as shown in Figure 9. Figure 9(a) shows the distances between three robots, while Figure 9(b) displays the corresponding wheel speeds of three mobile robots. The results indicate that formation is maintained. To check the efficiency of the proposed method, the experiments were conducted 30 times, and the results are presented in Table 2. The proposed controller in this study, along with the designed gain, can achieve optimal performance in side length. The average side length is 0.85 m, with minimum and maximum values of 0.82 and 0.94 m respectively. A larger size of the formation of cooperative mobile robots makes it more challenging for the system to converge smoothly. Therefore, careful consideration of the formation’s size and configuration of controller gains is important to achieve optimal performance and system convergence.

This study analyzes the effectiveness of a comprehensive Lyapunov-based approach for maintaining line and triangle formations in experiments. However, further studies are required to evaluate its scalability and robustness under different conditions. The algorithm’s performance has not yet been tested for a large number of robots, which could impact its suitability for scaling up in industrial applications. Subsequent research could

consider variables such as employing diverse formation geometries, adjusting robot speeds, and assessing how environmental disruptions affect the algorithm's efficiency.

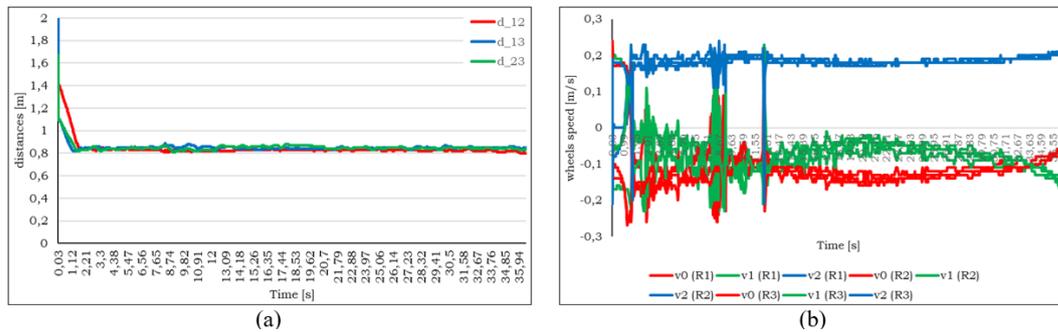


Figure 9. Triangle formation (a) robot's distances and (b) wheel speeds of each robot

Table 2. The travel time and distances of 30 experiments on a line formation

Parameters	Min	Average	Max
Side length [m]	0.82	0.85	0.94
Travel time [s]	35.81	36.1	36.5

4. CONCLUSIONS AND FUTURE WORKS

This research has demonstrated the effectiveness of semi-decentralized control strategies in maintaining the formation geometry of multiple omnidirectional mobile robots through rigorous simulation. The algorithm has successfully maintained line and triangle formations under experimental conditions, indicating that the balance between formation size and controller gains configuration is crucial for optimal performance and system convergence. The Lyapunov-based approach proved to be an efficient method for cooperative object transport, achieving an average side length maintenance of 1.00 meters with minimal deviation. Quantitative analysis across 30 experimental runs showed consistent performance, with a maximum side length fluctuation of only 2 centimeters, highlighting the accuracy and reliability of the formation control strategy. However, it should be noted that this study's experiments have limitations in the algorithm's performance with a larger number of robots or its robustness against various types of failures such as sensor malfunctions or unexpected robot behavior.

ACKNOWLEDGMENT

This research is funded and supported by the Research Center and Community Engagement at the University of Surabaya in Indonesia.

REFERENCES

- [1] H. F. McCreery, Z. A. Dix, M. D. Breed, and R. Nagpal, "Collective strategy for obstacle navigation during cooperative transport by ants," *Journal of Experimental Biology*, vol. 219, no. 21, pp. 3366–3375, Nov. 2016, doi: 10.1242/jeb.143818.
- [2] S. Na et al., "Bio-inspired artificial pheromone system for swarm robotics applications," *Adaptive Behavior*, vol. 29, no. 4, pp. 395–415, Aug. 2021, doi: 10.1177/1059712320918936.
- [3] A. Alhijaili, Z. M. Kilic, and A. N. P. Bartolo, "Teams of robots in additive manufacturing: a review," *Virtual and Physical Prototyping*, vol. 18, no. 1, Dec. 2023, doi: 10.1080/17452759.2022.2162929.
- [4] M. Javaid, A. Haleem, R. P. Singh, and R. Suman, "Substantial capabilities of robotics in enhancing industry 4.0 implementation," *Cognitive Robotics*, vol. 1, pp. 58–75, 2021, doi: 10.1016/j.cogr.2021.06.001.
- [5] A. F. Melo and L. M. Corneal, "Case study: evaluation of the automation of material handling with mobile robots," *International Journal of Quality Innovation*, vol. 6, no. 1, p. 3, Dec. 2020, doi: 10.1186/s40887-020-00037-y.
- [6] A. A. A. Rasheed, M. N. Abdullah, and A. S. Al-Araji, "A review of multi-agent mobile robot systems applications," *International Journal of Electrical and Computer Engineering*, vol. 12, no. 4, pp. 3517–3529, Aug. 2022, doi: 10.11591/ijece.v12i4.pp3517-3529.
- [7] A. A. Faieza et al., "Review on issues related to material handling using automated guided vehicle," *Advances in Robotics & Automation*, vol. 05, no. 01, 2016, doi: 10.4172/2168-9695.1000140.
- [8] Z. Yan, N. Jouandea, and A. A. Cherif, "A survey and analysis of multi-robot coordination," *International Journal of Advanced Robotic Systems*, vol. 10, no. 12, p. 399, Dec. 2013, doi: 10.5772/57313.

- [9] N. A. K. Zghair and A. S. Al-Araji, "A one decade survey of autonomous mobile robot systems," *International Journal of Electrical and Computer Engineering*, vol. 11, no. 6, pp. 4891–4906, Dec. 2021, doi: 10.11591/ijece.v11i6.pp4891-4906.
- [10] E. Tuci, M. H. M. Alkilabi, and O. Akanyeti, "Cooperative object transport in multi-robot systems: a review of the state-of-the-art," *Frontiers Robotics AI*, vol. 5, no. MAY, May 2018, doi: 10.3389/frobt.2018.00059.
- [11] A. Farinelli, N. Boscolo, E. Zanutto, and E. Pagello, "Advanced approaches for multi-robot coordination in logistic scenarios," *Robotics and Autonomous Systems*, vol. 90, pp. 34–44, Apr. 2017, doi: 10.1016/j.robot.2016.08.010.
- [12] H. M. Güzey, T. Dierks, S. Jagannathan, and L. Acar, "Hybrid consensus-based control of nonholonomic mobile robot formation," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 88, no. 1, pp. 181–200, Oct. 2017, doi: 10.1007/s10846-017-0541-6.
- [13] X. Chen *et al.*, "A novel virtual-structure formation control design for mobile robots with obstacle avoidance," *Applied Sciences (Switzerland)*, vol. 10, no. 17, p. 5807, Aug. 2020, doi: 10.3390/app10175807.
- [14] S. Campbell, W. Naeem, and G. W. Irwin, "A review on improving the autonomy of unmanned surface vehicles through intelligent collision avoidance manoeuvres," *Annual Reviews in Control*, vol. 36, no. 2, pp. 267–283, Dec. 2012, doi: 10.1016/j.arcontrol.2012.09.008.
- [15] K. Raghuvaiya, B. Sharma, and J. Vanualailai, "Leader-follower based locally rigid formation control," *Journal of Advanced Transportation*, vol. 2018, pp. 1–14, 2018, doi: 10.1155/2018/5278565.
- [16] A. D. Dang, H. M. La, T. Nguyen, and J. Horn, "Formation control for autonomous robots with collision and obstacle avoidance using a rotational and repulsive force-based approach," *International Journal of Advanced Robotic Systems*, vol. 16, no. 3, p. 172988141984789, May 2019, doi: 10.1177/1729881419847897.
- [17] H. W. Agung and I. Nilkhamhang, "Hierarchical decentralized LQR control for formation-keeping of cooperative mobile robots in material transport tasks," *Engineering Journal*, vol. 25, no. 12, pp. 37–50, Dec. 2021, doi: 10.4186/EJ.2021.25.12.37.
- [18] M. A. Kamel, X. Yu, and Y. Zhang, "Formation control and coordination of multiple unmanned ground vehicles in normal and faulty situations: A review," *Annual Reviews in Control*, vol. 49, pp. 128–144, 2020, doi: 10.1016/j.arcontrol.2020.02.001.
- [19] G. P. Liu and S. Zhang, "A survey on formation control of small satellites," *Proceedings of the IEEE*, vol. 106, no. 3, pp. 440–457, Mar. 2018, doi: 10.1109/JPROC.2018.2794879.
- [20] M. Dorigo, G. Theraulaz, and V. Trianni, "Swarm robotics: past, present, and future," *Proceedings of the IEEE*, vol. 109, no. 7, pp. 1152–1165, Jul. 2021, doi: 10.1109/JPROC.2021.3072740.
- [21] V. Gazi and K. M. Passino, "Stability analysis of social foraging swarms," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 34, no. 1, pp. 539–557, Feb. 2004, doi: 10.1109/TSMCB.2003.817077.
- [22] S. A. Kumar, J. Vanualailai, and B. Sharma, "Lyapunov-based control for a swarm of planar nonholonomic vehicles," *Mathematics in Computer Science*, vol. 9, no. 4, pp. 461–475, Dec. 2015, doi: 10.1007/s11786-015-0243-z.
- [23] A. R. Justus, "Degree of achievability of omnidirectional motion in various mobile robot designs: a review," *IAES International Journal of Robotics and Automation (IJRA)*, vol. 5, no. 1, p. 17, Mar. 2016, doi: 10.11591/ijra.v5i1.pp17-28.
- [24] M. Kirtas, K. Tsampazis, N. Passalis, and A. Tefas, "Deepbots: a webots-based deep reinforcement learning framework for robotics," *IFIP Advances in Information and Communication Technology*, vol. 584 IFIP, pp. 64–75, 2020, doi: 10.1007/978-3-030-49186-4_6.
- [25] A. Farley, J. Wang, and J. A. Marshall, "How to pick a mobile robot simulator: a quantitative comparison of CoppeliaSim, Gazebo, MORSE and Webots with a focus on accuracy of motion," *Simulation Modelling Practice and Theory*, vol. 120, p. 102629, Nov. 2022, doi: 10.1016/j.simpat.2022.102629.
- [26] W. R2022b, "http://www.cyberbotics.com," cyberbotics, 2019, http://www.cyberbotics.com.
- [27] F. Jordan, H. Wicaksono, and V. Indrawati, "Webots simulator for Lyapunov-based cooperative omnidirectional mobile robots evaluation," in *2022 International Conference of Science and Information Technology in Smart Administration, ICSINTESA 2022*, Nov. 2022, pp. 24–29, doi: 10.1109/ICSINTESA56431.2022.10041600.

BIOGRAPHIES OF AUTHORS



Hendi Wicaksono Agung    earned a B.Eng. graduated in Electrical Engineering from the University of Surabaya in Indonesia, in 2004, and received his M.Eng. degree in Electrical Engineering from the ITS University in Indonesia and Ph.D. in Electrical Engineering from SIIT, Thammasat University, Thailand in 2009 and 2022. He joined the University of Surabaya as a full-time faculty member in 2008. He is currently an assistant professor in the Electrical Engineering Dept. His research interests include robotics, mechatronics, multi-robots, autonomous vehicle, fuzzy logic systems, control systems, instrumentation, and industrial automation. He can be contacted at email: hendi@staff.ubaya.ac.id.



Fransisco Jordan    earned a Bachelor of Engineering in Electrical Engineering from the University of Surabaya in Indonesia in 2021. His research interests include robotics, control systems, instrumentation, programmable logic controllers (PLCs), and industrial automation technologies. Currently, he is employed by a private company that specializes in the production of livestock feed and operates within the automation system section. His experience in the automation system section of a livestock feed production company has provided him with valuable insights into industrial automation technologies, which will undoubtedly add depth to his research endeavors. He can be contacted at email: fransisco21jordan@gmail.com.