Journal of Engineering Science and Technology
Vol. 16, No. 1 (2021) 398 - 414

STATIC POLYGON FORMATION IN LEADER-FOLLOWER ROBOTIC SYSTEM BY UTILIZING RP LIDAR SENSOR IN UNKNOWN ENVIRONMENT

BAYADIR A. ISSA[1,*], ABDULMUTTALIB T. RASHID[2]

[1]Collage of Technical Management, Information Technology Department,
Southern Technical University, 61001, Basrah, Iraq
[2]Collage of Engineering, Electrical Engineering Department,
University of Basrah, 61001, Basrah, Iraq
*Corresponding Author: bayader.abdulrazaq@stu.edu.iq

Abstract

The leader-formation problem is considered as one of the most important problems in the multi-robot systems. In this paper, two algorithms are proposed as a solution for controlling the leader-follower formation problems. In leader-follower formation, one robot acts as a leader with the characterized movement and other robots travel as followers in which their positions have been set randomly in an unknown environment. Firstly, the leader robot characterizes the followers by using the acquired data from the RP LIDAR sensors which are equipped in every robot instead of using a camera and image processing. The leader restricted the followers in the right positions by utilizing the Heap permutation algorithm. After that, it guided them to form the polygon static shape by utilizing two algorithms: least power origination and minimum time algorithm. Simulation results show that the proposed algorithms have a good performance and efficiency 95.8% for least power algorithm and 95.3% for minimum time algorithm in getting a satisfying solution for the formation control problem as compared with the cluster matching algorithm.

Keywords: Formation control, Heap permutation, Leader-follower formation, Multi-mobile robot, RP LIDAR sensor, Static Formation.

1.Introduction

Mobile robot's formation control has many applications such as transportation, observation and search tasks. Therefore, it has been considered by many researchers [1, 2]. Formation control means the problem of maintaining the relative position and orientation of robots in a cluster while at the same time enabling the cluster to travel in general. The utilization of formation control allows one to accomplish complex missions as every specific task can be achieved in one robot navigation [3, 4]. Fundamental formation control issue includes maintaining required geometric movements of adjusting shapes and sizes such as triangular, square, polygon or section. Numerous types of control methods have been proposed to control the formation such as support issues [5, 6] including a virtual-structure approach, behaviour-based approaches, and leader-follower approaches [7].

In the virtual structure approach, the controller has three stages. In the first stage, the desired dynamics of the virtual structure are characterized. In the second stage, the ideal movement of the virtual structure is converted into the required movement for every robot. At the final stage, singular follows controllers for every robot are obtained from robot tracking [8]. The advantage of this approach is that an individual scientific guideline interprets the whole tangible information space into the actuator yield space without the requirement for numerous standards or practices. Furthermore, the obtained behaviour can be incorporated by utilizing vector activities. A few properties, for example, robustness and stability can be demonstrated by utilizing the theoretical tools from physics, control theory and graph theory. Additionally, the virtual structure approach strategy is frequently used to structure aggregate practices that require multi-robot formation [9, 10].

While in the behaviour-based approach, a group of behaviour or missions which comprise various raw decentralized activities (subtasks) that are coordinated to accomplish a global task, while the control activity is acquired as a blend of these primitives [11]. The main advantage of this strategy is the planned response bonded together by matching loads of the activities that rely upon the complementary partners of the connecting robots. However, this approach suffers from a drawback that is the mathematical analysis is complex and accordingly, the simultaneousness of the robotic formation of the desiderated structure cannot be established [12].

The third technique is the leader-follower that depends on the nearness of a leader, which might be virtual or real. In this approach, robots in the formation simply follow the leader and they maintain the relative distance between each other to preserve the ideal shape formation. Every robot direction has been acknowledged by the leader. This strategy depends on data that are collected from nearby sensors and every robot which in turn helps to select another leader once the previous one fails to maintain the formation shape [13]. Leader-follower formation for a group of mobile robots has attained a huge consideration from many researchers due to its convenience in numerous applications, for example, military applications, transportation, stockroom robotization [14-16]. Dai et al. [17] suggested a leader-follower formation that is implemented based on the relative movement states to shape and preserve the formation of multi-robots. The main idea of this strategy is to locate an objective and accurate speed to change the robot's present state. In [18] a virtual leader-follower technique and potential capacities for the formation control and hindrance shirking issue for multi-robot systems.

The proposed algorithm draws revelation from Rashid et al. [10] which is used to localize multi-robots through a static polygon formation. Formatting the configuration allows to localize the mobile robot with three static robots that act as source nodes. This mobile robot will be fixed once the step is completed and it will be used as one of the source nodes. One fixed robot is selected and switch to the next mobile robot. By repeating this process, the static formation is achieved without an external source signal. In the configuration format, the mobile robot moves through the shifting and switching sequences. In the shift sequence, each robot moves to the desired path. In the switching sequence, the multi-robot is started and follows the desired path. As each robot moves, it must remain within the range of the infrared sensor. a formation coordination is used as a strategy to create a regular polygon shape. Every robot in a multi-robot system gets its data from its own sensor and by communicating with other robots. First, all robots use a block-matching algorithm to estimate the initial position and direction. The strategy then begins to choose the appropriate robots to move depending on the shape and location of these robots. This process is accomplished by a block-matching algorithm that selects robots that are far from the polygon formation site. In triangulation, robot motion estimation is performed by at least three fixed robots, but the proposed strategy requires only two robots. This reduces the number of transitional events when building a polygon formation. Each robot uses the algorithm of the diagram of the binary tree to determine its path. These robots stop at the boundaries of the communication range of the neighbouring fixed robot. Then another distant robot begins moving to target formation and repeats the same procedure. This process continues until all robots reach their targets. This paper proposes a new localization and formation control algorithm that is called a leader-follower formation control algorithm which is applied to the kinematic model of the differential-drive mobile robots. In this algorithm, a leader robot controls the limitation and direction of the multi-robot system to obtain a static formation that forms a polygon shape. This approach utilizes the acquired data from the RP LIDAR sensor to connect to every robot to recognize the other robots by the leader.

The Heap permutation algorithm is used to localize and orientate every robot in the formation control cluster. Heap's algorithm is utilized to generate all permutations of n objects [19]. The main idea is to generate every permutation from the last stage by selecting a pair of elements to exchange, without irritating the other *(n-2)* elements. The Heap algorithm is very efficient and straightforward as compared to other algorithms since it swaps just 2 elements for every one created. The weakness might be that there is no apparent order in the generated permutations. Finally, the leader robot coordinated the followers to perform a static formation that is shaped as a polygon form. This occurs by using two algorithms, least power origination, and a short timeframe. In these two algorithms, the least distances or least time has been selected by robots to reach the new areas and shape the static form. Then, the best methodology has been recommended according to the obtained results. This paper has been organized as follows: The mobile robot kinematic used shows in Section 2. Section 3 depicts the reproductive consequences of leader-follower formation algorithm. While the main results about the control algorithms are defined and analysed in Section 4. Finally, Section 5 shows the conclusion.

2. Mobile Robot Kinematic

The robot kinematic aims to find the robot velocity in the inertial casing as an element of the wheels' speed and the geometric parameters of the robot [20]. In this work, the differential mobile robot has been utilized to simulate the proposed navigation approach. Every robot is equipped with RP LIDAR sensors that used for distance estimations, two DC motors and two encoders to provide a genuine position of the robot in the environment. The name "differential" originates from the way that the robot can alter its direction by shifting the general pace of its wheels that do not require an extra guiding movement [21]. Here, the posture vector of the mobile robot is displayed as, $p = [x \quad y \quad \theta]$, where x and y present the position of the mobile robot and θ is characterized as the angle between the X-coordinate and the heading direction as shown in Fig. 1.

$$\dot{x} = v \cos\theta \tag{1}$$

$$\dot{y} = v \sin\theta \tag{2}$$

$$\dot{\theta} = w \tag{3}$$

$$\dot{x} = \frac{R}{2}(v_r + v_l)\cos\theta \tag{4}$$

$$\dot{y} = \frac{R}{2}(v_r + v_l)\sin\theta \tag{5}$$

$$\dot{\theta} = \frac{R}{L}(v_r - v_l) \tag{6}$$

where,

$$v = \frac{R}{2}(v_r + v_l) \tag{7}$$

$$w = \frac{R}{L}(v_r - v_l) \tag{8}$$

and v_r and v_l represent the right and left velocities of the two-wheeled mobile robot respectively, and L is the distance between the two wheels of the robot [20]. R is the distance from p to the Instantaneous Centre of Curvature (ICC) at which the robot rotates about.

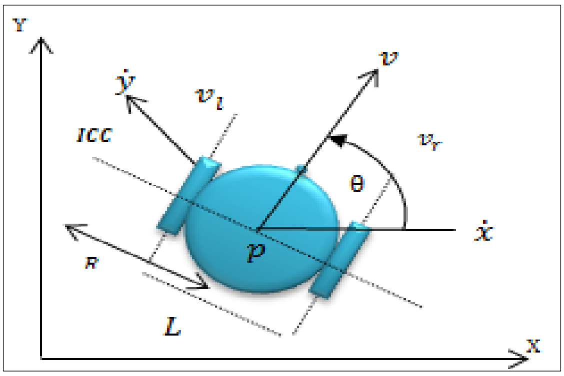


Fig. 1. The kinematic model of differential two wheeled mobile robot.

3. Proposed Formation Algorithms

In this section, the leader-follower static formation is proposed as new approaches for multi-robot static formation in polygon shape as shown in Fig. 2. The static formation means that the goal position is fixed with time. On the other hand, the dynamic formation occurs when the goals change their positions with time. In this approach, a static polygon formation is achieved by assigning a specific goal to each follower by a leader robot. This system consists of five follower robots that are distributed randomly to scan the environment and one leader to identify and lead the follower in order to modify the position and perform the static formation. Each follower robot is equipped with RP LIDAR sensors to scan the environment and send the scanning information to the leader via the Wi-Fi connection. The leader robot chooses the objective area for every follower in an unknown environment. Here, two cases for selecting the objective. The first one is the minimum power by making every robot travelling to the closest objective from its current location.

This strategy prevents the situation when at least two robots decide driving to a similar objective area during their movement. Finally, the leader robot-guided these robots to the rectified areas. The second one is to decrease the time of formation by rearranging the goals of the robots in a manner that makes the longest time of arrival as small as possible. In the two cases, the selection of managing the objectives depends on the structure of an exhibit that contains data about the distances between every robot and all objectives. This procedure is achieved by scanning through this exhibit and utilizing the Heap permutation algorithm to locate the best plan among objectives and robots.

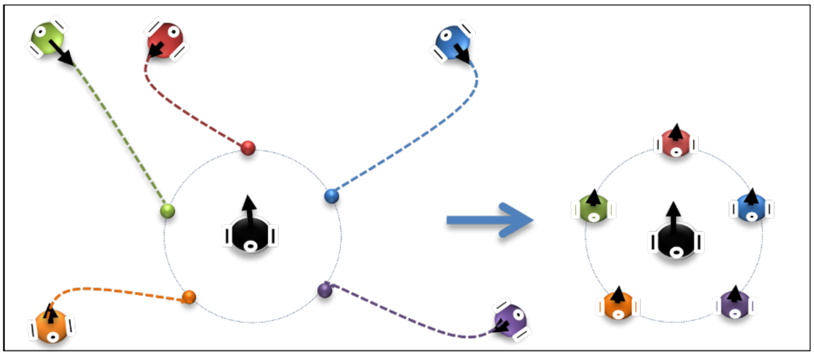


Fig. 2. The static polygon formation algorithm.

3.1 Formation with minimum power algorithm

This method is employed in an unknown environment where follower robots are localized randomly and directed by the leader robot. The Heap permutation algorithm is utilized to disseminate robots on objectives that can attempt to determine the best path of formation. The using of Heap permutation algorithm is connected to data that have been included in the two-dimensional cluster containing the distances among robots and objectives. The best shape formation occurs in the situation when the distances between robots and goals are the smallest. The execution procedure of this algorithm is explained as follows:

Step 1. Estimation the initial position and orientation of the follower robots: As initial state, all members of a multi-robot team are scattered randomly in an unknown environment. The leader robot estimates the initial position and orientation for the follower i (x_i, y_i, θ_i) by using the received information from the follower RP LIDAR sensor through the Wi-Fi network. The leader-follower classification and localization algorithm are used to determine the position and direction of every robot in the environment. This has been estimated by utilizing the distances and the orientation among these robots as shown in Figs. 3 to 6. The estimation of the initial position and orientation for the follower robots are calculated as follows:

1. Every follower robot in the system sends its checked data acquired from the RP LIDAR sensor. The RP LIDAR sensor rotates 360 degrees to measure the distances and the angle according to other robots. The measured information is stored in the one-dimensional array as follows:

$$[\, L_i\,, \theta_i] = [d_{i,j}, \theta_{i,j}] \tag{9}$$

where $d_{i,j}$ represents the distance between the robots (i, j), $\theta_{i,j}$ is the orientation angle between the robots (i, j).

2. Then, the obtained data will be sent to the leader. The checking procedure begins from the current direction and turns that well-ordered in an anti-clockwise direction until complete the 360 degrees.

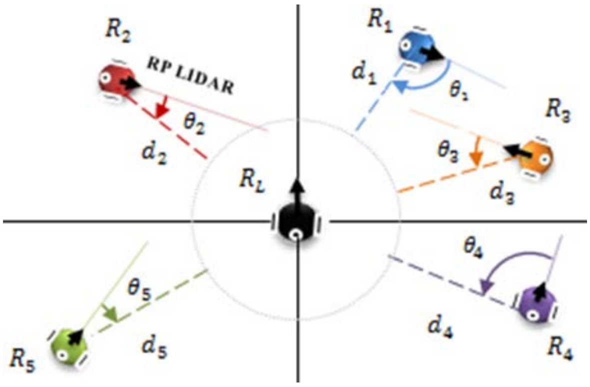


Fig. 3. Scan the environment by the followers RP LIDAR sensors.

3. The leader robot identifies every follower in the environment by viewing and coordinating the collected data from those followers through the Wi-Fi network. The leader robot stored data about the followers in the two-dimensional array as shown below:

$$\boldsymbol{A} = \begin{bmatrix} L_{10} & L_{12} & L_{13} & L_{14} & L_{15} \\ L_{21} & L_{20} & L_{23} & L_{24} & L_{25} \\ L_{31} & L_{32} & L_{30} & L_{34} & L_{35} \\ L_{41} & L_{42} & L_{43} & L_{40} & L_{45} \\ L_{51} & L_{52} & L_{53} & L_{54} & L_{50} \end{bmatrix} \tag{10}$$

$$\boldsymbol{B} = \begin{bmatrix} \theta_{10} & \theta_{12} & \theta_{13} & \theta_{14} & \theta_{15} \\ \theta_{21} & \theta_{20} & \theta_{23} & \theta_{24} & \theta_{25} \\ \theta_{31} & \theta_{32} & \theta_{30} & \theta_{34} & \theta_{35} \\ \theta_{41} & \theta_{42} & \theta_{43} & \theta_{40} & \theta_{45} \\ \theta_{51} & \theta_{52} & \theta_{53} & \theta_{54} & \theta_{50} \end{bmatrix} \tag{11}$$

where $\boldsymbol{L_{i,j}}$ represents the distance between the robot *i* and robot *j* and $\theta_{i,j}$ represents the orientation of robot i about robot *j*.

4. In this step, the localization procedure between any two robots relies on the perusing of the laser shaft. Every laser bar converges any robot in more than one point. The localization procedure is performed by choosing the appropriate direction and distance between two robots. Based on data from Eqs. (9) and (10) the coordinate axis of the robot I regarding the robot j can be calculated as:

$$x_i = L_{ij} * cos\,\theta_{ij} \tag{12}$$

$$y_i = L_{ij} * sin\,\theta_{ij} \tag{13}$$

where (x_i, y_i) represents coordinates that show the location of the follower robot i to the follower robot j, θ_{ij} is the angle that shows the orientation of the follower robot *i* to the follower robot *j*. The distances *Lij* between robots *i* and *j* can be calculated as follows:

$$L_{ij} = \sqrt{\left(\left(y_i - y_j\right)^2 + \left(x_i - x_j\right)^2\right)} \tag{14}$$

$$\theta_{ij} = tan^{-1}\frac{y_i - y_j}{x_i - x_j} \tag{15}$$

Each laser beam reaches any robot in more than one point. The laser beam with the most limited distance provides exact guidance of the robot position due to the robot has a circle shape. To avoid the error of calculating in the correct position, the shortest distance between two robots has been estimated as shown in Fig. 4. The shortest distance can be obtained by determining the radius of the robot that represents the distance between the laser beam and the centre of the robot. The radius is estimated as follows:

a- The laser beam line equation formulation

$$m = (x_2 - x_1)/(y_2 - y_1) \tag{16}$$

$$b = -1 \tag{17}$$

$$y = m * x + n \,, n = y - m * x \tag{18}$$

b- Determine the distance between the centre of the circle $(x_0, y_0)$and the laser beam ((x_m, y_m) which represented by *r*

$$r = \frac{|m*x_0 + b*y_0 + n|}{\sqrt{m^2+1}} \tag{19}$$

c- The coordinate axis of the leader robot at centre point is (x_0, y_0) can be determined as follows:

$$x_0 = x_1 + (d + r) * cos(\theta) \tag{20}$$

$$y_0 = y_1 + (d + r) * sin(\theta) \tag{21}$$

where r is the radius of the robot and d is the distance between the RP LIDAR of the first robot and the centre of the second robot.

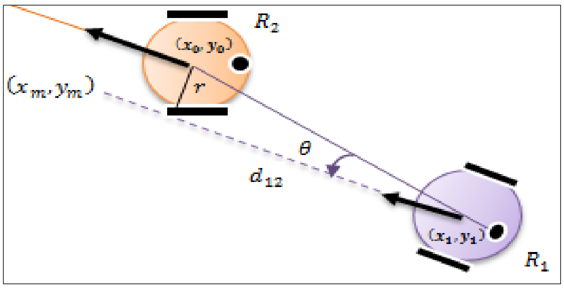


Fig. 4. Minimum distance calculation between two robots.

Step 2. Apply the minimum power calculation: The next position of each follower robot can be determined by using the Heap permutation algorithm that selects the nearest location for the follower robot to perform the static formation. The new position of each follower can be determined as follows:

1. Compute the distance between the position of each follower (x_0, y_0) and the virtual position (x_L, y_L) that selected by the leader robot to perform the static formation as:

$$d_{L0} = \sqrt{((y_L - y_0)^2 + (x_L - x_0)^2)} \tag{22}$$

2. Scan all values with the Heap permutation algorithm, where each value ($\boldsymbol{P}$) represents a distance between each follower robot to one goal position that selected by the leader as shown in Fig. 5 and explained by the following equations:

$$\begin{aligned} S_1 &= [p_4, p_1, p_2, p_3, p_5] \\ S_2 &= [p_5, p_1, p_2, p_3, p_4] \\ S_3 &= [p_3, p_5, p_1, p_2, p_4] \\ S_4 &= [p_2, p_4, p_5, p_1, p_3] \\ S_5 &= [p_4, p_5, p_1, p_2, p_3] \end{aligned} \tag{23}$$

3. At each scan value, the sum of all distance's values between robots and their goals is estimated.

$$L_{sum} = \sum_{i=1}^{n} S(P_i) \tag{24}$$

where l_{sum} is the summation of all distances between the follower robots and their goals in one group of permutation S, where n represents the number of follower robots.

4. The lowest value obtained in Heap permutation scanning corresponds to the best scenario in terms of minimum power that can be obtained as:

$$L_{i=\min L_{sum}}(S) \tag{25}$$

5. At each interval, the leader robot should check positions and orientations of all robots in environment to avoid the collision with other robots.

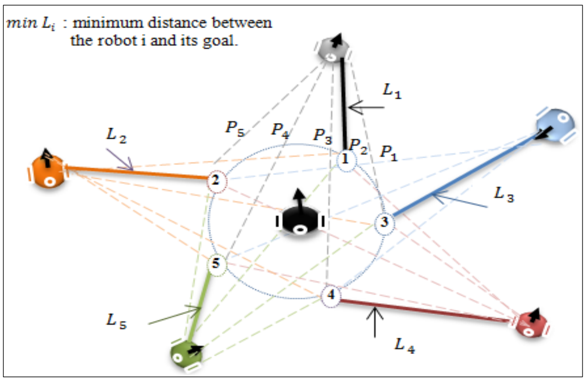


Fig. 5. Scanning the locations by Heap permutation.

Step 3. Path planning, every robot uses visible tangent diagram calculation to select the shortest path to reach the goal position [22]. The visible tangent diagram is built by drawing a tangent line from a circle at the robot source position to the external point at the goal position as shown in Fig. 6. The distance length from the source point to the tangent point on the virtual location of the target must be processed. Therefore, the diagram of a visible tangent comprises two paths from source to target with their distances that were acquired. The shortest path from source to target is selected over these paths to be a preserved path for the robot to arrive at the target with the least distance. The proposed algorithm can be described by the following steps.

1- Estimate the position, radius, the speed of robot $(P_s(x_s, y_s), r_s, \theta_s, v_s)$, and the target location $P_t(x_t, y_t)$.

2- Calculate an immediate robot path from source to target. This path has two sub-paths: a curve path that is important to coordinate the differential drive robot toward the target, and a straight line which is the tangent line between the circular path of the robot and the target position.

3- Then, the shortest path among the two acquired paths is selected by utilizing the shortest path algorithm that helps to reduce the distance among robot and target when altering the robot direction in order to avoid the collision.

4- By utilizing the shortest path algorithm, each immediate path in the tangent visible graph is adjusted to a straight line and a circular path direction that can be calculated as follows:

a. Straight line calculation: The advanced differential analyser calculation is utilized to calculate the straight line and the circular path directions from source to target as indicated by the following steps:

i. Input points $A(x_s, y_s)$and $B(x_t, y_t)$ that represent the end line points. Compute

$$\Delta x = x_s - x_t \tag{26}$$

$$\Delta y = y_s - y_t \tag{27}$$

where Δx and Δy represents the differences in x-axis and y-axis between the end points.

ii. Then, calculate the number of steps J

$$J = Max(\Delta x, \Delta y) \quad (28)$$

$$Inc\ x = \Delta x / J \quad (29)$$

$$Inc\ y = \Delta y / J \quad (30)$$

where Inc x and Inc y represent the amount of increment between two points lie on the line.

$$x = x + Inc\ x\ ,\ \ y = y + Inc\ y \quad (31)$$

iii. Rounding the result by a constant c

Round (x, y, c) (32)

b. The circular path drawing algorithm: The computerized differential analyser calculation can be additionally used to actualize the circular path of mobile robots. This calculation expects that the circular path is a part of a circle with the centre at the origin. The calculation of the mobile robot direction is achieved by the following steps:

i. Select the angle θ, the radius R and the centre of the circular arc (x_0, y_0).

ii. Let the addition angle equal to dθ and initial location (x, y).

Round (x, y, c) (33)

iii. For angle = 0 to θ step by $d\theta$

$$x = x + (x_0 - x) * d\theta \quad (34)$$

$$y = y + (y_0 - y) * d\theta \quad (35)$$

Round (x, y, c) (36)

iv. Finally, the shortest path is selected from these two paths.

Step 4. At every interval, every robot utilizes the leader-follower algorithm to check the positions and directions of all robots to maintain a strategic distance from the collision with them.

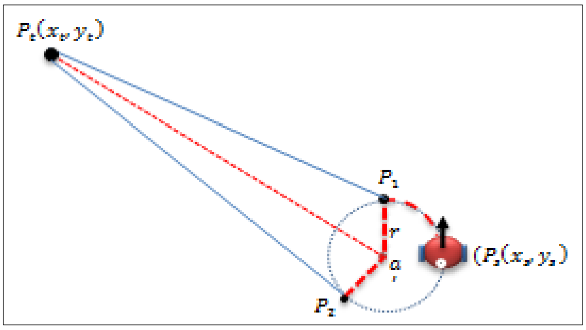


Fig. 6. The shortest path algorithm.

3.2. Formation with minimum time algorithm

This method is designed to minimize the time that is required to achieve the multi-robot formation in an unknown environment as shown in Fig. 7. Additionally, by utilizing the Heap permutation algorithm, every one of the probabilities of the circulations of the robots on targets can be examined to select the best formation. In this case, the best formation happens when the greatest distance in this formation is the minimum as compared to other formations. This procedure has been done as follows:

Step 1. Perform step one of the minimum power algorithm.

Step 2. Apply minimum time algorithm: to determine the next position of each follower robot also:

1- Scan all estimations of the Heap permutation algorithm, where each value assigned one robot to one goal as in Eq. (18).

2- At each scan value, compute the maximum of all distances between every follower robot and its goals.

$$L_{Max} = Max_{i=1}^{n} S(P_i) \tag{37}$$

where l_{max} is the maximum distance between the follower robot and its goal, *n* is the number of follower robots in every group of *S*.

3- Choose the minimum estimation of all the most extreme estimations of all the possible formation to represent the instance of the minimum time of formation which can be obtained as:

$$L_{i=min\,L_{max}} \tag{38}$$

Step 3. Path planning, the goal position of each follower robot with short distance can be obtained also by utilizing a visible tangent diagram calculation.

Step 4. At each time, the algorithm repeated by the leader robot after utilizing the leader-follower localization algorithm to check positions and directions of all robots to maintain a strategic distance from the collision with them. The complete algorithms of static formation in polygon shape can be described with the block diagram shown in Fig. 8 with two methods: least power and minimum time.

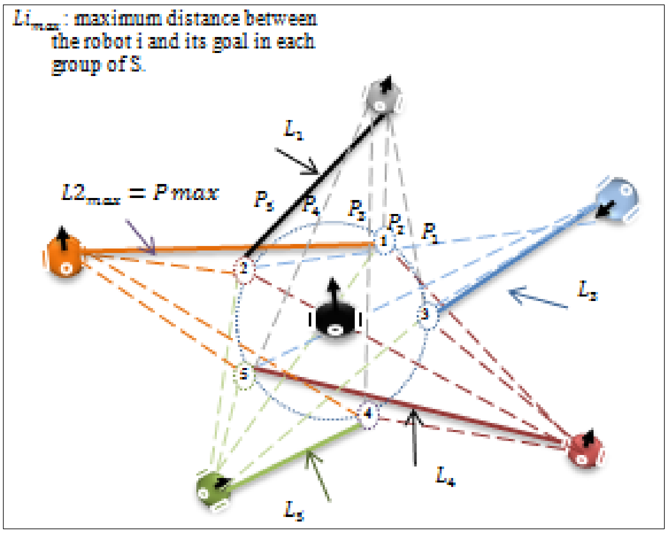


Fig. 7. Maximum distance with minimum time formation.

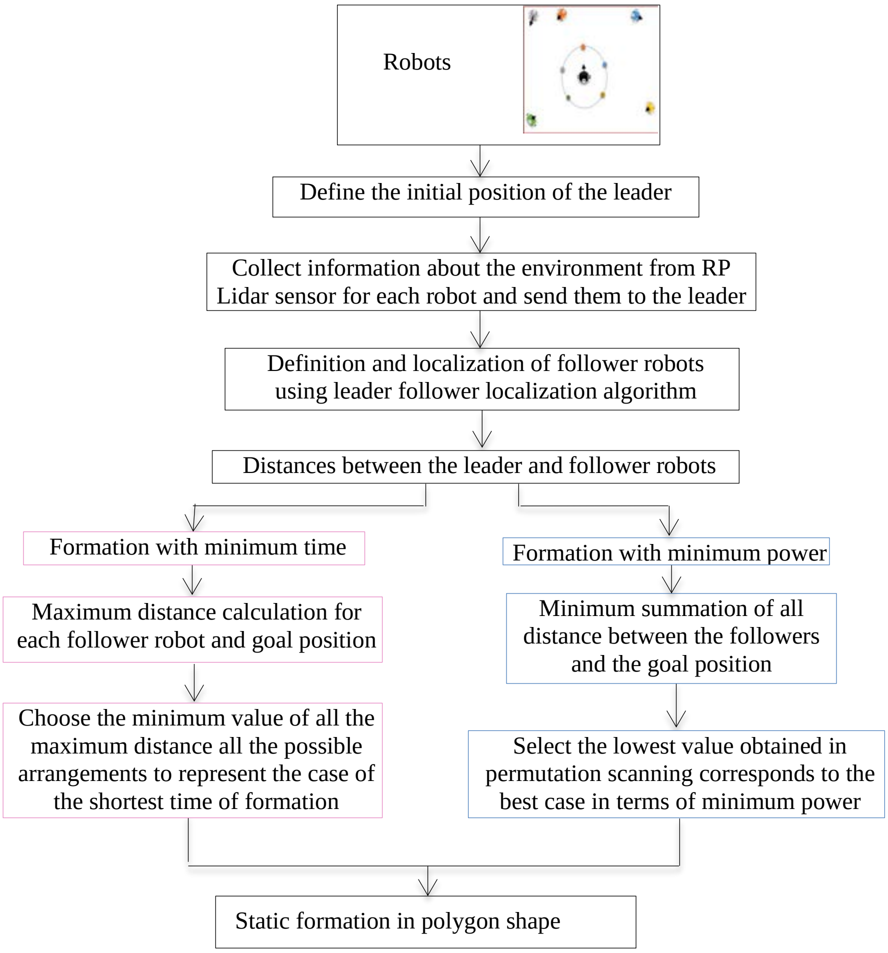


Fig. 8. Block diagram of leader-follower formation algorithms.

4. Simulation Results

In this paper, the proposed formation control algorithms of multi mobile robots have been simulated and the performance results have been recorded. The various methodologies of formation, generation are simulated by considering the impact of the leader control, the formation of follower robots in unknown environments and the type of formation to explore static formation. There were actualized utilizing visual basic 2010. The algorithm suggested in this paper is compared with the cluster matching algorithm in reference [10]. All simulations were performed over various topologies addressing various system sizes (n) extending from 3 to 7 robots. The robots were randomly set in a 500×500-pixel region. There are three parameters utilized in this simulation:

1- The size of the environment: the robots distributed in (100, 150, 200) pixels

2- The scope of RP LIDAR sensors means the furthest distance can be covered between two robots.

3- The number of robots: the number of robots in the network with a square area of side length1. The implementation is done with (3-7) robots that are distributed randomly in the environments.

Simulation parameters are chosen to evaluate the performance of these polygon formation strategies. Different measurements have been analysed in this simulation estimation:

- The percentage of achievement: that means the maximum time taken by multi-robots from their current position to the final destination.
- The time of achievement can be utilized to complete the formation by utilizing different networking topologies.

Fig.9(a) presents the localization algorithm of the follower robots in which the position and orientation of follower robots are determined concerning the leader. Figure 9(b) represents the formation control with least power algorithm of five follower robots. Each robot has 360 degrees of sensing of other robots and goals at different time steps. Figure 9(c) represents the formation with minimum time approach of five robots. Increasing the number of robots influences the formation and increases the time of shape completion.

Fig.10(a) and (b) show the comparison among the least power algorithm, the minimum time algorithms and the cluster matching algorithm to compute the percentage of the formation completing time in a static environment with five and seven distributed robots, respectively. the comparisons show that as the number of robots increases in the environment, the accomplishment percentage increases, too. the least power and minimum time algorithms to perform static polygon formation algorithm have the optimal performance comparing to cluster matching algorithm. Moreover, the degree of rotation of RP LIDAR sensor for each follower will affect the localization and formation procedure as shown in Fig. 10(c). This shows that the time required to complete the formation with one degree is less than the two-degree or three because its effects the localization procedure and does not obtain the corrected position and orientation of the follower robots. This is because it requires more time to complete the formation. Also, for the three algorithms. Its shows that the least power also take less time and the minimum time algorithm than the cluster matching one.

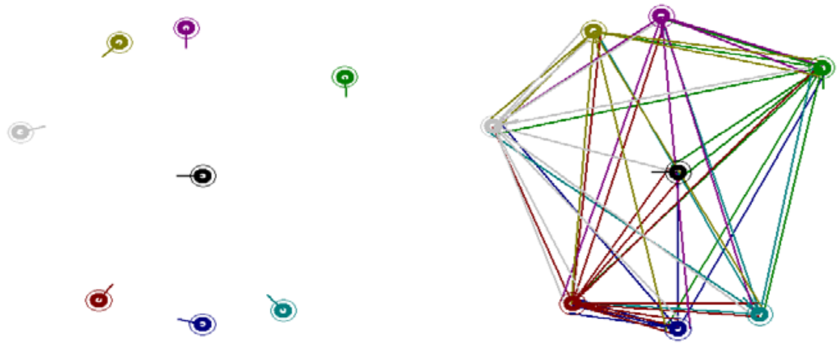


Fig. 9(a). Leader follower localization for seven follower robots.

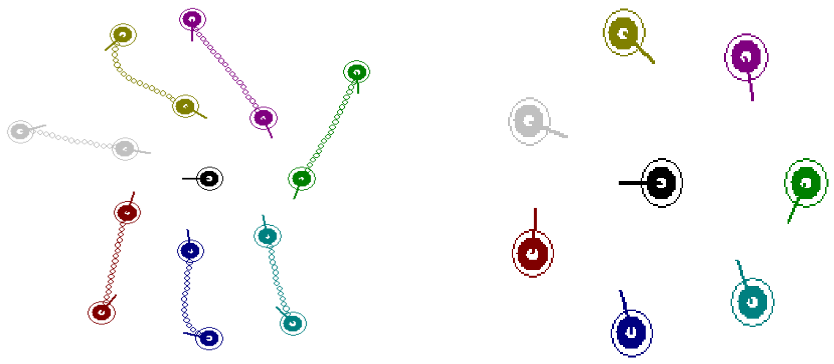


Fig.9(b). The leader-follower formation with least power algorithm for seven follower robots.

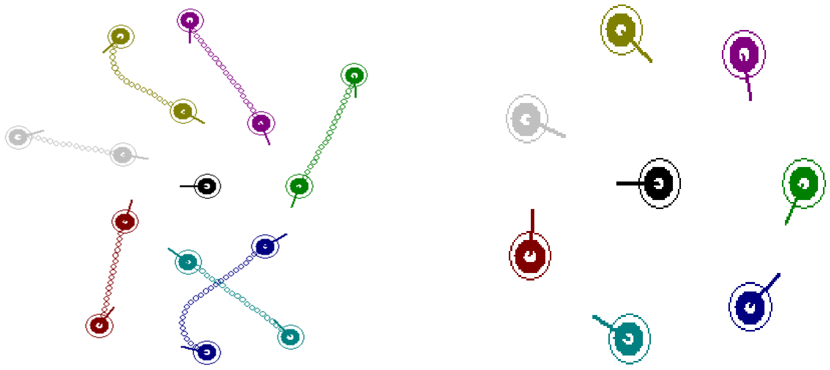
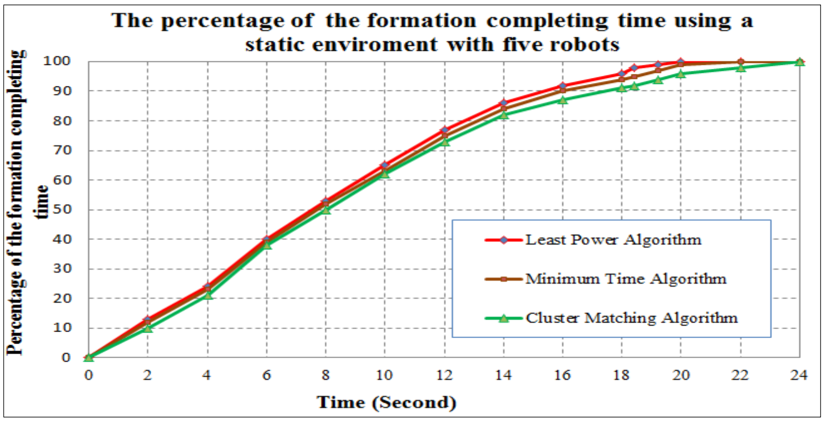
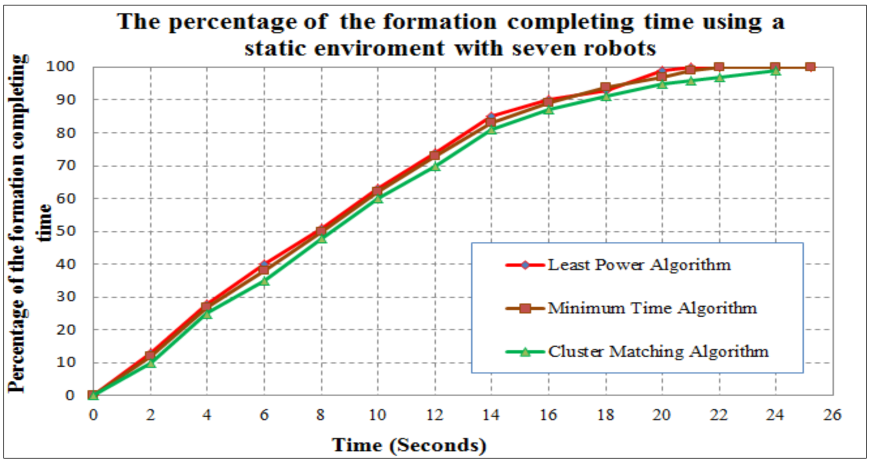


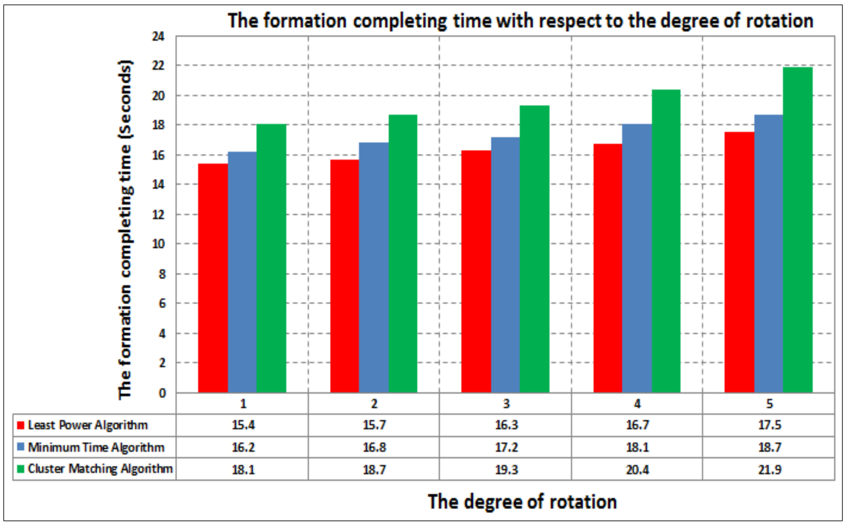
Fig.9(c). The leader-follower formation with minimum time algorithm for seven follower robots.



(a).Simulation result for five robots



(b). Simulation result for seven robots



	1	2	3	4	5
Least Power Algorithm	15.4	15.7	16.3	16.7	17.5
Minimum Time Algorithm	16.2	16.8	17.2	18.1	18.7
Cluster Matching Algorithm	18.1	18.7	19.3	20.4	21.9

(c).Simulation result with different degree of rotation

Fig.10. Simulation results of static polygon formation with different number of robots.

5. Conclusions

In this paper, a novel organizing remote adjacent robot's formation strategy for the static polygon shape formation is proposed in an unknown environment by using several numbers of mobile robots localized and distributed randomly. Simulation results are implemented in an environment with a different number (3 to 7) of robots using the least power algorithm, the minimum time algorithm and the cluster matching algorithms. the results show that the least power algorithm and the minimum time algorithm have a better efficiency with 95.8% for least power algorithm and 95.3% for minimum time algorithm in getting a satisfying solution for the formation control problem as compared with the cluster matching algorithm to complete the formation since it uses the Heap to determine the short distance. Also, determine the execution time and the accomplishment percentage. The other algorithms depend on several algorithms to perform the same task which needs more computation time. From the results, it is found that the accomplishment percentage and the execution time are increasing as the number of robots and the degree of rotation of sensor of each robot are increase for all algorithms. But still the least power and the minimum time algorithms have a better efficiency than the cluster matching algorithm.

References

1. Guo, J.; Lin, Z.; Cao, M.; and Yan, G. (2010). Adaptive leader-follower formation control for autonomous mobile robots. *Proceedings of the 2010 American Control Conference*. Baltimore, USA, 6822-6827.
2. Tews, A.; Matari, M.J.; and Sukhatme, G.S. (2004). Avoiding detection in a dynamic environment. *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Sendai, Japan, 1-6.
3. Yang, H.; Bao, X.; Zhang, S.; and Wang, X. (2019). A multi-robot formation platform based on an indoor global positioning system. *Applied Sciences*, 9(6), 1-16.
4. Gamage, G.W.; Mann, G.K.I.; and Gosine, R.G. (2010). Leader follower based formation control strategies for non holonomic mobile robots: design, implementation and experimental validation. *Proceedings of the 2010 American Control Conference*. Baltimore, USA, 224-229.
5. Nascimento, T.P.; Fontes, F.A.; and Moreira, A.P. (2011). Leader following formation control for omnidirectional mobile robots. *Proceedings of the 8th International Conference on Informatics in Control, Automation and Robotics*. Noordwij kerhout, Netherlands, 135-144.
6. Hachour, O. (2009). The proposed path finding strategy in static unknown environments. *International Journal of Systems Applications, Engineering and Development*, 4(3), 127-138.
7. Wang, Y. (2019). *Multi-robot formation control in obstacle environments*. Ph.D. Thesis, Nanyang Technology University, Singapore.
8. do Nascimento, T.P. (2012). *Coordinated multi-robot formation* control. Doctoral Program in Electrical and Computers Engineering, University of Porto, Portugal.
9. Issa, B.A.; and Rashid, A.T. (2019). A survey of multi-mobile robot formation control. *International Journal of Computer Applications*, 181(48), 12-16.
10. Rashid, A.T.; Ali, A.A.; and Frasca, M. (2019). Polygon shape formation for multi-mobile robots in a global knowledge environment. Iraqi *Journal of Electrical and Electronic Engineering*, 15(1), 76-88.
11. Ćosić, A.; Šušić, M.; Graovac, S.; and Katić, D. (2013). An algorithm for formation control of mobile robots. *Serbian Journal of Electrical Engineering*, 10(1), 59-72.

12. Sugihara, K.; and Suzuki, I. (1996). Distributed algorithm for formation of geometric patterns with many mobile robots. *Journal of Robotic Systems*, 13(3), 127-139.
13. Chen, Y.Q.; Wang, Z. (2005). Formation control: a review and a new consideration. *Proceedings of the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Edmonton, Alta, 3181-3186.
14. Widyotriatmo, A.; Joelianto, E.; Prasdianto, A.; Bahtiar, H.; and Nazaruddin, Y.Y. (2017). Implementation of leader-follower formation control of a team of nonholonomic mobile robots. *International Journal of Computers Communication and Control*, 12(6), 871-885.
15. Zhao, Y.; Zhang, Y.; and Lee, J. (2019). Lyapunov and sliding mode based leader-follower formation control of multiple mobile robots with an augmented distance-angle strategy. *International Journal of Control, Automation and Systems*, 17(5), 1314-1321.
16. Consolini, L.; Morbidi, F.; Prattichizzo, D.; and Tosques, M. (2007). A geometric characterization of leader-follower formation control. *Proceedings of the 2007 IEEE International Conference on Robotics and Automation*. Roma, 2397-2402.
17. Dai, Y.; Tran, V.; Xu, Z.; and Lee, S. (2010). Leader-follower formation control of multi-robots by using a stable tracking control method. *Advances in Swarm Intelligence: Lecture Notes in Computer Science*, 6146, 291–298.
18. Yan, J.; Guan, X.; Luo, X.; and Chen, C. (2017). Formation control and obstacle avoidance for multi-agent system-based on virtual leader-follower strategy. *International Journal of Information Technology and Decision Making*. 16(03), 865-880.
19. Bacher, A.; Bodini, O.; and Kuei, H. (2016). Generating random permutations by coin-tossing: classical algorithms. *ACM Transactions on Algorithms*, 13(2), Article 24.
20. Petrović, E.; Nikolić, V.; Ćirić, I.; Simonović, M.; Pavlović, S.; Mančić, M.; and Rajković, B. (2016). Kinematic model and control of mobile robot for trajectory tracking. *Annals of Faculty Engineering Hunedoara-International Journal of Engineering*, 161-164.
21. Aouf , A.; Boussaid, L.; and Sakly, A. (2019). Same fuzzy logic controller for two-wheeled mobile robot navigation in strange environments. *Journal of Robotics*, 2019, 1-11.
22. Ibrahim, Z.Y.; Rashid, A.T.; and Marhoon, A.F. (2016). An algorithm for path planning with polygon obstacles avoidance based on the virtual circle tangents. *Iraqi Journal of Electrical and Electronic Engineering*, 12(2), 221-234.