# Polygon Shape Formation for Multi-Mobile Robots in a Local Knowledge Environment

Abdulmuttalib T. Rashid
*Electrical Engineering Depart.*
*Basrah University*
*Basrah, Iraq*
abdturky@gmail.com

Abduladhem A. Ali
*Computer Engineering Depart.*
*Basrah University*
*Basrah, Iraq*
abduladem1@yahoo.com

Abstract **The formation is an important task in multi-mobile robots coordination in a defined environment. In a local knowledge environment the multi-mobile robot formations are realized using small robots with minor hardware requirements. The localization, path planning and obstacle avoidance processes are required to perform formation. In this paper a static strategy for polygon shape formation is implemented using a several number of mobile robots. This strategy has a better efficiency, since it use the cluster matching algorithm instead of the triangulation algorithm in completing the formation. Also, the visibility binary tree algorithm and the reciprocal orientation algorithm are used in this paper. This strategy has better performance in the multi-robot formation, since it use the cluster matching algorithm instead of the triangulation algorithm.**

*Index Terms*—Local knowledge environment, mobile robot, Polygon shape formation.

## I. Introduction

Recently, it has been pointed out how replacing a single high performance robot by simple mobile robots may be advantageous since these simple cooperative robots may enhance the efficiency of work and data transmission between robots. Therefore, multi-mobile robots are expected to be used in a variety of applications including surveillance [1], object manipulation [2], intelligent transportation systems [3], and exploration [4]. Thus, this paper presents the formation algorithms needed to coordinate multi-mobile robots arrangement within a group. It is required to realize geometric patterns adapting to environmental changes to enable multi-mobile robots to perform the assigned tasks. The formation investigation require several task to be implemented: The localization algorithm is used to calculate the initial location and orientation of each robot, which is assumed to be known by other robots [5, 6]. The formation is achieved by driving every robot to its own goal through a straight line trajectory using path planning algorithm [7, 8]. However, in this strategy, these trajectories may cause collision among robots and then break the whole system. The robots use the collision avoidance algorithm [9, 10].

For multi-mobile robots, geometric constraints are constraints due to obstacles in the environment and kinematic constraints are essentially non-holonomic constraints. Since, multi-mobile robots have very good velocity controllers and are therefore able to follow a given trajectory, it is useful to find a kinematic motion plan for such a system by considering the velocity of mobile robot as the input variable.

Formation can be considered as a special type of swarm, in which multi robots must show a fixed pattern while moving along a path. In swarms, the robots should only be in the neighborhood of one another, and the motion between them is less structured. Additionally, formations can be classified as rigid or flexible. In rigid formations robots must enforce a fixed shape [11, 12]. That is a robot must not move from its given position in the formation, at all times. On the other hand, robots in flexible formations can break the formation by change their positions, and returning to their previous positions, at any possible time [13]. This gives advantage to perform obstacle avoidance behavior for each individual robot, instead of the need to implement it at the group level. There are many applications for formations. Transportation tasks are important field of application for formations, where a group of robots, moving in a rigid shape can be used to carry any object from one place to another [3]. Another example of formations is cooperative surveillance in military applications, where a group of individuals should attack an enemy by using group of robots to observe an area and distinguish probable targets [1]. Distributed mobile sensor networks is another application area, where group of robots arranged in a rigid formation can be equipped with sensors with a small detection range to simulate a wider range sensor. Distributed communications networks are another application for formations, where multiple robots provide temporary communications in a disaster zone, where each robot represents an agent in a specific location [12]. In order to fulfill the goal of formation, several problems have to be solved. These problems are: the problem of knowing the initial position of these robots [14], and the path planning from initial to the final locations in formation [15]. This process also needs to deal with obstacle avoidance and collision avoidance, when robots move to their final location [9].

## II. POLYGON SHAPES FORMATION STRATEGY

The static formation with local knowledge environment is achieved by self-mobile localization to enable multi-robot systems to spread along the desired trajectory without any external reference signal [16]. The idea underlying this strategy is to locate multi-robots via formation coordination. In formation coordination, a mobile robot can be localized by using three fixed robots that act as reference nodes. This mobile robot is fixed when it has completed its movement and then it is used as one of the reference nodes, while one of the fixed robots is selected and switched to be the next mobile robot. By repeating this process, the static formation is achieved without any external reference signal. In formation coordination mobile robots are moving through shifting and switching sequences. In shifting sequence each robot moves to the desired path, where in switching sequence multi-robots are driven to follow the desired trajectory. When each robot moves, it should stay within a communication range of the infrared sensors.

In this section the formation coordination is used as a strategy to form a static regular polygon, as shown in Fig. 1. Each robot in a multi-robot obtains information from its own sensors and by communicating with other robots. At first all robots estimate their initial positions and orientations by using the cluster matching algorithm [14]. After that, the strategy starts to select suitable robots to move according to the shape of formation and location of these robots. This process is done by clusters matching algorithm which selects the far robots from the place of polygon formation. In triangulation estimation of the robot movement is done at least by three fixed robots, while the proposed strategy needs only two robots. This leads to reduce the number of switching events when the polygon formation is built. Each robot decides its trajectory by using the binary tree tangent graph algorithm.

These robots stop at the boundaries of the communication range of the neighbor fixed robots. At that time another far fixed robots start to move to the goal formation and repeat the same procedure. This process continues until all robots reach their goals. The steps of implementation of this strategy on a polygon formation are as follows:

Step 1: Compute the initial position $(x_i, y_i)$ of each robot by using the cluster matching algorithm [14]. Fig.1 shows the pseudo-code for the cluster matching algorithm.

Step 2: Select the far robots from goal formation as mobile robots. The selection of these robots is done by using the cluster matching algorithm.

Step 3: Each robot decides its trajectory by using the visibility binary tree algorithm. The shifting sequences are done through this trajectory [15]. Fig.2 shows the pseudo-code for the visibility binary tree algorithm.

Step 4: The robots use the reciprocal orientation algorithm to avoid collision with each other [9]. The pseudo-code for this algorithm is shown in Fig. 3.

**Input** $P(x_s, y_s)$ : Distance IR sensor position,
$n$: number of robots
$R$: Maximum detection range of IR sensors
**For** each robot $i$ **do**
Measure robot distance $(d_i)$ and angle $(\theta_i)$
Compute robot position $(x_i, y_i)$
Store $(d_i)$ and $(x_i, y_i)$ in matrix $A_1$ as nodes
**Next** robot i
Rearranged matrix $A_1$ descending according to $(d_i)$ values
**For** each node $i$ **do**
**For** each node $j$ **do**
Compute distance $L_{ij}$ between node $i$ and $j$
**If** $L_{ij} > R$ **then** $L_{ij} = \infty$
**Else** $nb_{ij} = nb_{ij} + 1$ {neighbor nodes}
**End if**
Store $(L_{ij})$ in matrix $A_2$
**Next** node $j$
Store $nb_{ij}$ in matrix $A_2$
**Next** node $i$
**For** each row $i$ in matrix $A_2$ **do**
Max = 0
**For** each $L_{ij} \neq \infty$ in row $i$ **do**
**If** $nb_{jj} >$ Max **then** node $j$ is cluster head
**Else** $nb_{jj} = 0$
**End if**
**Next** $j$
**Next** row $i$
**For** each cluster $i$ **do**
**For** each $L_{ij} \neq \infty$ in cluster $i$ **do**
Compute the orientation $\theta_i$
**Next** $j$, $i$
**For** each cluster $i$ **do**
**For** each *successive* $\theta_i$ in cluster $i$ **do**
Compute the angles $\varnothing_{ij}$ {anticlockwise angle from node j to node i}
**Next** $j$
Store $nb_{ij}$ and $\varnothing_{ij}$ in matrix $A_3$
**Next** $i$

Fig. 1. Pseudo-code of the cluster matching algorithm.

Step 5: The robots stop when they reach the boundaries of communication range of neighbor robots as shown in Fig. 4 .a, b.

Step 6: Repeat steps 2, 3, 4, and 5 by switching to the next far mobile robots as shown in Fig. 4 .c.

Step 7: The shifting and switching sequences are repeated until all mobile robots reach their formation goals, as shown in Fig. 4.(d, e, and f).

The overall approach is summarized by the pseudo-code shown in Fig. 5.

**Input** $(P_s(x_s, y_s), r_s, \theta_s, v_s)$ Robot parameters
$(P_1, P_2, \ldots, P_n, r_1, r_2, \ldots, r_n)$ Obstacles parameters
$(P_g(x_g, y_g))$ Target location
Compute the future trajectory of robot
**Repeat**
Check if trajectory intersects with obstacle
**If** intersection is occur **then**
Compute the end points of outer tangent
Compute the end points of inner tangent
**End if**
**Until** reach to target
Construct the modified binary tree paths
Adjust the modified binary tree paths
Compute the shortest path using the searching algorithm

Fig. 2. Pseudo-code of the visibility binary tree algorithm.

**Input** $P(x, y), v, r, \theta$ : The robots parameters
t: time interval
n:number of robots
**For** each interval t **do**
**For** each robot i **do**
Sense $q_i = (x, y)$ and $v_i$
**For** each robot j **do**
Sense $q_j = (x, y)$ and $v_j$
Calculate $\alpha$
Use transformation algorithm
**Repeat** for each interval e
Calculate $P_j$
Calculate $d_j$
**Until** $d_j < (r_i + r_j)$ or e > intervals
**If** $d_j < (r_i + r_j)$ **then**
Rotate robot i and robot j
**End if**
**Next** robot j
Combined Reciprocal trajectory algorithm
Calculate $P_j$.
**For** each robot j **do**
Calculate $d_j$
**If** $d_j <$ deadlock zone **then**
Use deadlock algorithm
**End if**
**Next** robot j
**Next** robot i
**Next** interval t

Fig. 3. Pseudo-code of the reciprocal orientation algorithm.

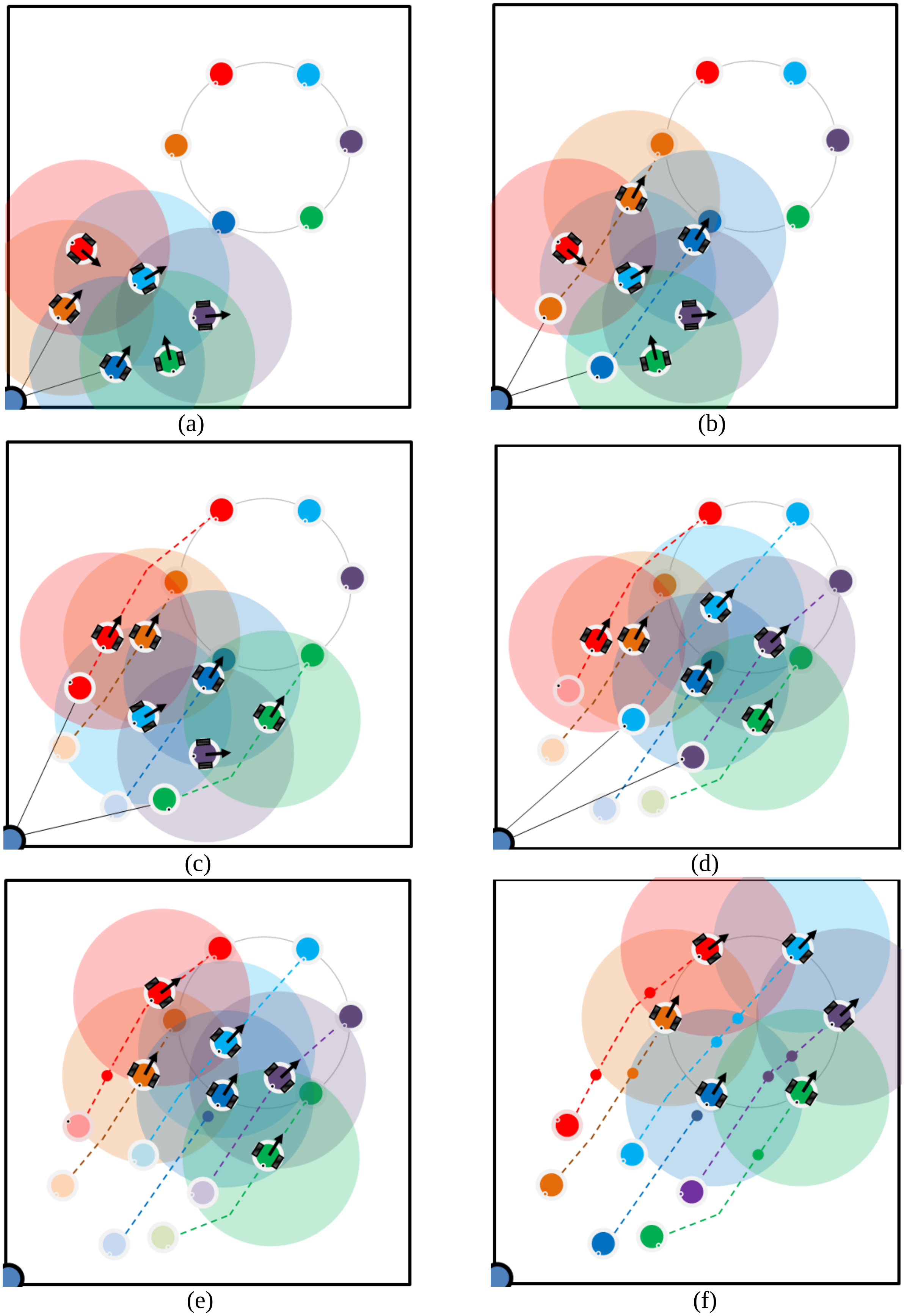

Fig. 4. Static polygon formation with local knowledge environment

**Input** $n$: number of robots
$R$: Maximum detection range of IR sensors
$P$ ($x_s$, $y_s$): Laser sensor position
g1($x_1$, $y_1$)…, gn($x_n$, $y_n$): Goals positions.
**For** each robot $i$ **do**
Use cluster matching algorithm to estimate robot position ($x_i$, $y_i$) and orientation ($\theta i$).
Compute distance between robot $i$ and laser sensor position
**Next** robot $i$
**Do while** formation not complete
Select robot $i$ which has minimum distance to laser sensor position.
**Do while** robot $i$ has $R$ distances from at least two neighbor Robots.
Compute robot $i$ trajectory to its goal using binary tree tangent graph algorithm.
Estimate the future position of robot $i$ using reciprocal Orientation algorithm to avoid collision with other robots.
**Loop**
**Loop**

Fig. 5. Pseudo-code of the static polygon formation within a local knowledge environment

## III. SIMULATION RESULTS

In this paper the algorithm which proposed to investigate the formation of multi mobile robots is simulated using visual basic 2010. The strategy of formation is simulated in local knowledge environment, and investigates as a statically type formation. The simulation is performed over different topologies representing different network sizes (n) ranging from 4 to 8 robots. The robots were randomly placed on a 500x500 pixels area. Two parameters are used in this simulation:

- Network size (n): the number of robots in the simulation environment.
- Maximum detection range of infrared sensors (R): The maximum graph distance between any two neighbor robots.

The purpose of this simulation is to evaluate the following performance metrics:

1. Percentage of accomplishment: Indicates the time spent by multi-robots from the initial to the final positions.
2. System Efficiency: The system efficiency [17] is defined by the following equation:

$$\text{System efficiency} = \frac{1}{n}\sum_{i=1}^{n}\frac{|x_{R_i}(t) - x_{R_i}(0)|}{l_{R_i}(t)} \qquad (1)$$

Where $|X_{Ri}(t) - X_{ri}(0)|$ is the distance from the initial to the final positions at time t, and $l_{Ri}(t)$ is the total travelling distance of each robot at time t.

Fig. 6 shows the simulation. Fig. 6(a)–(f) represent the Screenshots of simulations at different time steps to investigate the static polygon formation in local knowledge environment. This process is achieved by using the clusters matching algorithm to localize each robot in the environment and the moving of these robots from the start position to the goal are achieved by using the visibility binary tree algorithm. Through the movement of these robots the collision may be occurs among these robots. The reciprocal orientation algorithm is used to solve this problem.

Fig. 7 shows the comparison of the accomplishment time between triangulation algorithm and cluster matching algorithm. This comparison is done on four robots. The maximum detection range of infrared sensors is assumed to be equal to 80 pixels. The accomplishment time of cluster matching algorithm is better than triangulation algorithm, because the triangulation algorithm needs at least three fixed robots to achieve the static polygon formation, while this process in cluster matching algorithm needs only two fixed robots.

Fig. 8 repeats the comparison in Fig. 7 with robots having a maximum detection range of infrared sensors equal to 100 pixels. The system efficiency of triangulation algorithm is equal to 96%, while it is equal to 98% for cluster matching algorithm. These graphs show that the accomplishment time is improved as the maximum detection range of infrared sensor is increased and by the use of the cluster matching algorithm.

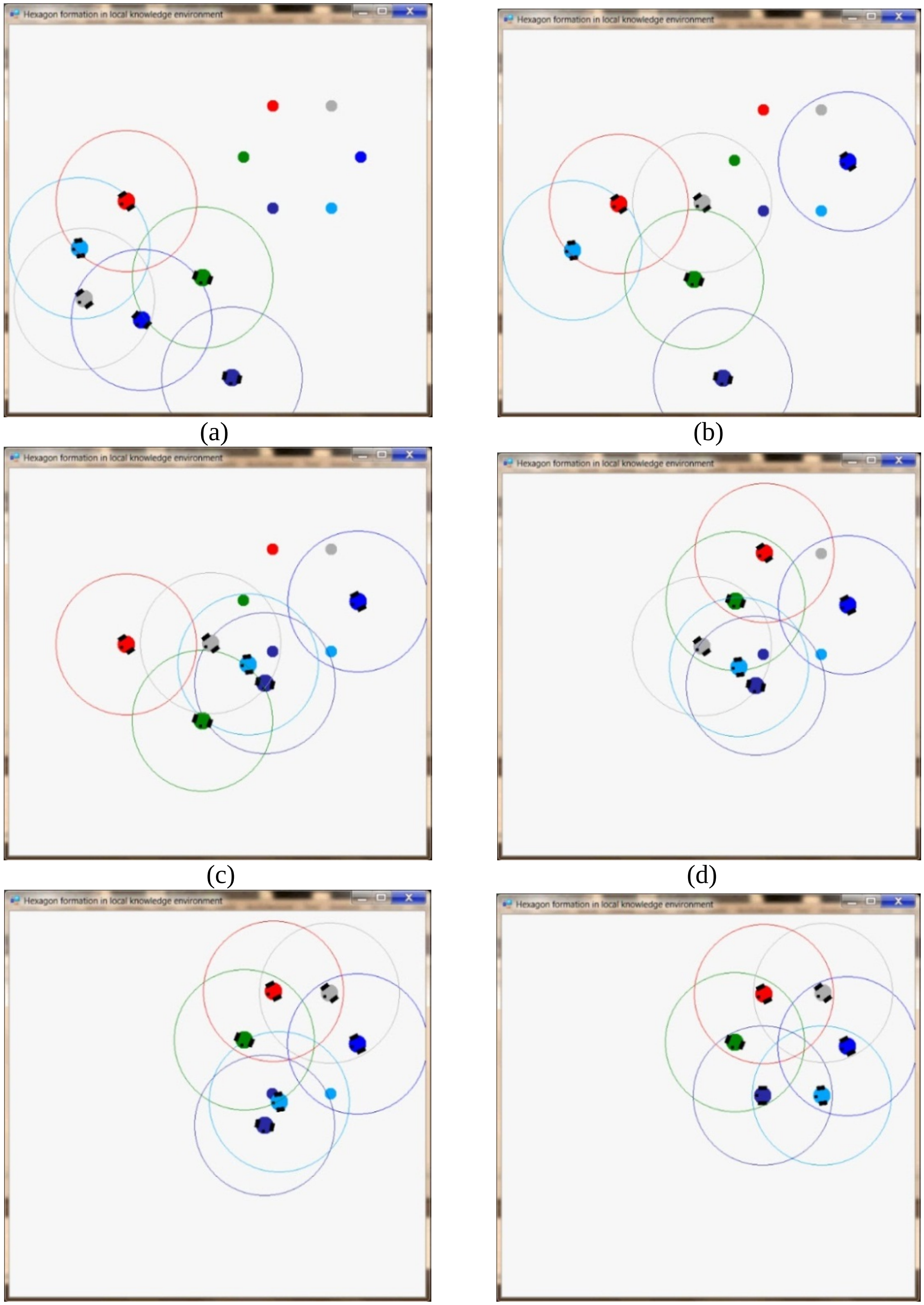


Fig. 6. Static polygon formation with local knowledge environment.
(a)-(f) Screenshots of simulations in different time steps.

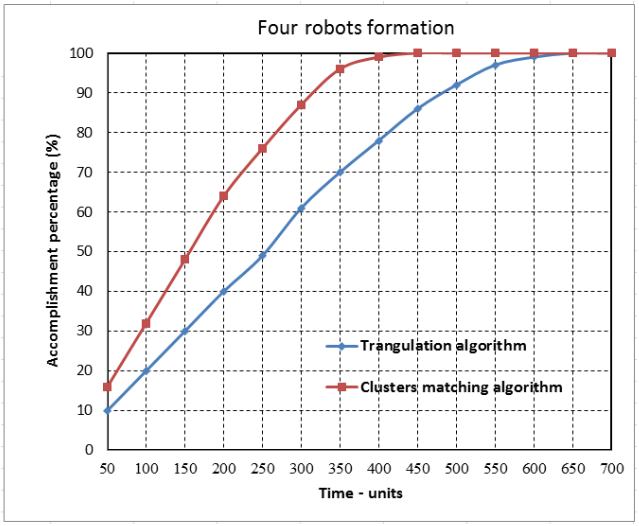


Fig. 7. Comparison between triangulation and cluster matching algorithm with 80 pixels maximum detection range of infrared sensors.

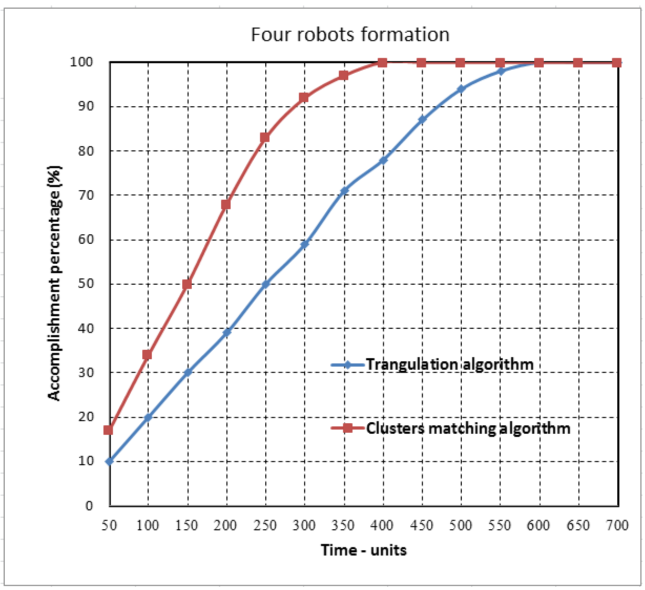


Fig. 8. Comparison between triangulation and cluster matching algorithm with 100 pixels maximum detection range of infrared sensors.

## IV. Conclusions

In this paper a static strategy for polygon shape formation is simulated on local knowledge environment using a several number of mobile robots. This simulation is implemented on environment with four robots using the cluster matching algorithm and the triangulation algorithm. The simulation is repeated with 80 and 100 pixels for the maximum detection range of infrared sensors. The results show that the modified self-localization strategy has a better efficiency to complete the formation, since it uses the cluster matching algorithm instead of the triangulation algorithm. The accomplishment time of cluster matching algorithm is better than triangulation algorithm in both detection ranges of the infrared sensors, because the triangulation algorithm needs at least three fixed robots to achieve the static polygon formation, while this process in cluster matching algorithm needs only two fixed robots.

References

[1] J. T. Feddema, C. Lewis and D. A. Schoenwald, “Decentralized control of cooperative robotic vehicles: theory and application”, IEEE Transactions on Robotics and Automation, vol. 18, no.5, pp. 852–64, 2002.

[2] F. Yusuke, T. Chomchana, J. Ota, H. Yuasa, T. Arai, H. Asama and K. Kawabata, “Realizing the exploration and rearrangement of multiple unknown objects by an actual mobile robot”, Advanced Robot, vol. 19, no.1, pp. 1–20, 2005.

[3] J. Baber, J. Kolodko, T. Noel, M. Parent and L. Vlacic, “Cooperative autonomous driving”, IEEE Robotics & Automation Magazine, vol. 12, pp. 44–49, 2005.

[4] D. F. Hougen, M. D. Erickson, P. E. Rybski, S. A. Stoeter, M. Gini and N. Papanikolopoulos “Autonomous mobile robots and distributed exploratory missions”, Distributed autonomous robotics systems, vol. 4, Springer-Verlag, pp. 21–30, 2000.

[5] O. A. Hasan, A. T. Rashid, R. S. Ali, “Hybrid approach for multi-node localization and Identification”, Basrah Journal for Engineering Sciences, vol.16, No 2, pp. 11-20, 2016.

[6] O. A. Hasan, R. S. Ali, A. T. Rashid, “Centralized approach for multi-node localization and identification”, Iraq J. Electrical and Electronic Engineering, vol.12, No 2, pp. 178-187, 2016.

[7] Z. Y. Ibrahim , A. T. Rashid, and A. F. Marhoon, " An algorithm for Path planning with polygon obstacles avoidance based on the virtual circle tangents", Iraq J. Electrical and Electronic Engineering, Vol. 12, No. 2, pp. 221-234 , 2016.

[8] Z. Y. Ibrahim , A. T. Rashid, and A. F. Marhoon, " Prediction-Based Path Planning with Obstacle Avoidance in Dynamic Target Environment ", Basrah Journal for Engineering Sciences, Vol. 16, No. 2, pp. 48 – 60, 2017.

[9] A.T. Rashid, A.A. Ali, M. Frasca, L. Fortuna, Multi-robot collision-free navigation based on reciprocal orientation, Robotics and Autonomous Systems, vol. 60, no. 10, pp. 1221–1230, 2012.

[10] A.T. Rashid, A.A. Ali, M. Frasca, L. Fortuna, An algorithm for multi-robot collision-free navigation based on shortest distance , Robotics and Autonomous Systems, vol. 75, pp. 119–128, 2016.

[11] T. Eren, W. Whiteley, B. D.O. Anderson, A. S. Morse and P. N. Belhumeur, “Information Structures to Secure Control of Rigid Formations with Leader-Follower Architecture”, 2005 American Control Conference June, USA, 2005.

[12] L. Asimow and B. Roth, “The Rigidity of Graphs”, Journal of Mathematical Analysis and Applications, vol. 68, pp. 171-190, 1979.

[13] B. Roth, “ Rigid and Flexible Frameworks”, The American Mathematical Monthly, Vol. 88, No. 1, pp. 6-21, 1981.

[14] A. T. Rashid, M. Frasca, A. A. Ali, A. Rizzo, L. Fortuna, “Multi-robot localization and orientation estimation using robotic cluster matching algorithm”, Elsevier / Robotics and Autonomous Systems, vol.63, Part 1, pp. 108-121, 2015.

[15] A. T. Rashid, A. A. Ali, M. Frasca, L. Fortuna, Path planning with obstacle avoidance based on visibility binary tree algorithm, Robotics and Autonomous Systems, vol. 61, pp. 1440–1449, 2013.

[16] H. S. Ahn, “Formation Coordination for Self-mobile Localization: Framework”, IEEE International Symposium on Computational intelligence in Robotics and Automation (CIRA), Daejeon, pp. 340-348, 2009.

[17] J. Lee, S. Venkatesh and M. Kumar, “Formation of a geometric pattern with a mobile wireless sensor network”, Journal of Robotic System, Vol. 21, No. 10, pp. 517–530, 2004.