

Modified bug-1 algorithm based strategy for obstacle avoidance in multi robot system

Jom J. Kandathil¹, Robins Mathew¹ and Somashekhar S. Hiremath^{1*}

¹*Department of Mechanical Engineering, Indian Institute of Technology Madras, Chennai*

Abstract. One of the primary ability of an intelligent mobile robot system is obstacle avoidance. BUG algorithms are classic examples of the algorithms used for achieving obstacle avoidance. Unlike many other planning algorithms based on global knowledge, BUG algorithms assume only local knowledge of the environment and a global goal. Among the variations of the BUG algorithms that prevail, BUG-0, BUG-1 and BUG-2 are the more prominent versions. The exhaustive search algorithm present in BUG-1 makes it more reliable and safer for practical applications. Overall, this provides a more predictable and dependable performance. Hence, the essential focus in this paper is on implementing the BUG-1 algorithm across a group of robots to move them from a start location to a target location. The results are compared with the results from BUG-1 algorithm implemented on a single robot. The strategy developed in this work reduces the time involved in moving the robots from starting location to the target location. Further, the paper shows that the total distance covered by each robot in a multi robot-system is always lesser than or equal to that travelled by a single robot executing the same problem.

Keywords: BUG Algorithm, Obstacle Avoidance, Mobile Robot, Multi Robot System, Simulation.

1 Introduction

In recent years, mobile robots have become a more fundamental part of the industry. In general, obstacle avoidance has turned out to be pre-requisite to mobile robots and this has resulted in the introduction of a large variety of algorithms to support obstacle avoidance. Obstacle avoidance in unknown terrain is one of the key challenges in this regard. BUG algorithms are class of algorithms that deals with obstacle avoidance in unknown terrains. BUG algorithms were introduced as a methodology for non-heuristic motion planning in an uncertain environment where information about the environment is not fully pre-known [1]. The sensors attached to the robot are used to attain dynamic information about obstacles and the BUG algorithm uses this dynamic information for navigating the robot avoiding obstacles. Among the various preliminary BUG algorithms, BUG-1 is more reliable and safer for practical applications due to the exhaustive search algorithm present in it. Figure 1 show the trajectory of the robot when BUG-1 was used for obstacle avoidance. The robot

* Corresponding author: somashekhar@iitm.ac.in

initially moves towards target in a straight line. Upon detection of obstacle, the robots start circumnavigating the obstacle in either clockwise or anticlockwise direction. The point of shortest distance to the target is measured and stored during this circumnavigation. Once the robot completes the circumnavigation it will move towards the point of shortest distance through the shortest path. When it reaches the point of shortest distance, it will continue to move towards target in straight line.

There are many existing strategies, such as IBA [2] and K-Bug [3] that aims at improving the travel time of the BUG algorithm. More effective varieties of these strategies have been introduced to higher order problems such as military surveillance and video game designs. The working of various strategies that form the class of BUG algorithms and a comparison of their performances were provided by James et.al [4]. The potential for this branch of robotics is vast and hence this paper tries to address several issues of the basic variations of BUG algorithm with a new strategy using multiple robots in leader-follower formation. The concept of using a multi robot system to navigate around obstacles [5] is the basis of this new strategy.

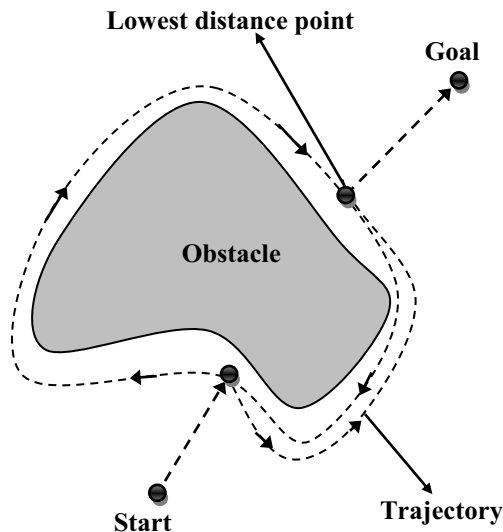


Fig .1. Bug-1 algorithm

Multi robot system in leader follower formation is a highly effective system in swarm robotics. The leadership exchange is the backbone of the leader-follower system [6], [7]. This was utilized in the present work to design an efficient multi robot obstacle avoidance strategy. This new strategy assumes that the robots in the system possess only basic sensing and movement capabilities thus allowing other robots to take over and complete the task in case of breakdown of the leader robot. The energy levels of individual robots were monitored in real time and the lower energy robots were made to conserve their energy by reassigning the task to other higher energy robots. These two features of the algorithm improve the success probability of the provided task. The concept of information exchange across multiple robots [8], [9] was also implemented to achieve better path navigation. Thus, this new strategy is a modified BUG-1 strategy which is implemented across multiple robots for improving travel time and energy expenditure of robots and providing a better success probability of task at hand.

The better travel time and energy efficiency of the proposed algorithm is described by comparing the algorithm with conventional algorithms. In principle, it has been shown that the strategy has better travel time than conventional BUG algorithms and provides more efficient and flexible energy distribution. Reminder of the paper is organized as follows. Section II explores the obstacle avoidance strategy in detail. Section III presents a comparison of this new strategy and BUG-1 algorithm with respect to travel time and energy efficiency. Preliminary experimental results are presented in section IV.

2 Multi robot obstacle avoidance

The multi robot bug algorithm, similar to conventional bug algorithm, navigates multiple robots across a planar surface with unknown obstacles of arbitrary shape provided the start and goal location are known. A conventional BUG-1 algorithm (hence will be referred to as BUG-1 algorithm) has memory element which enables it to store information about the coordinates tracked by the robot and hence can opt for the shortest path. The algorithm introduced in this paper is based on the exchange of this stored information across multiple robots. Since the robots can access information about the coordinates travelled by other robots, it enables them to make an optimal decision at a faster rate. The algorithm has three major parts: first two are concerning the motion of the robots while the third is regarding the formation. A pictorial representation of this strategy is shown in Fig.1.

Initially, two robots are assigned, a leader robot (at S_1) and a follower robot (at S_2). They move together towards the common goal (G). When these robots encounter an obstacle (O), they circumnavigate the obstacle in opposite direction and continue to move till they meet each other at a point (M). As the robots move around the obstacles, both the robots check the distance from its current position (P) to goal G and the point of minimum distance (D) is stored in the memory. From M , the robots take the shortest route to D and await the arrival of rest of the robots to D . At M , a new leader and a follower are assigned based on energy consideration. The direction of circumnavigation is chosen by taking the midpoint (C) of the line connecting start points of the leader and follower. An imaginary line is drawn from C to G as shown in Fig 2 and the robot that is on the left side of this line will be a right sensing robot (moves around the obstacle in the clockwise direction, sensing the obstacles on the right). The robot on the right side acts as follower robot and goes around in the other direction to that of the leader robot. This ensures that there is no collision between the robots before the completion of circumnavigation, which otherwise leads to overlapping and reduces efficiency of the system.

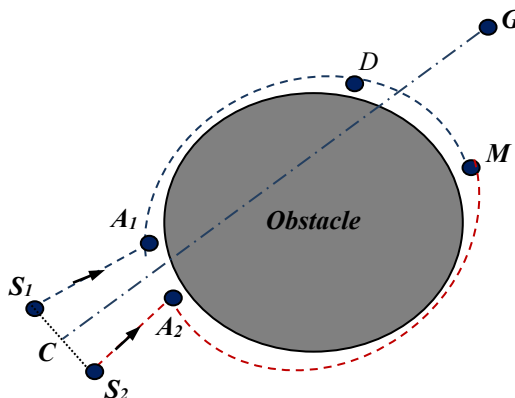


Fig. 2. Choice of direction of circumnavigation in multi robot system

The flow chart of the algorithm is shown in Fig. 3:

- 1) Leader and follower move towards target till one of the following events occur
 - a) Target is attained- stop. Rest of the robots move to meet the leader and follower robots by taking the shortest path to reach the path tracked by leader and follower the path to reach target.
 - b) Obstacle is encountered. Then move to step 2.
- 2) Circumnavigate the obstacle in opposite directions while recording the distance to target till the two robots meet. After meeting move to the point of minimum distance to target. Then move to step 3.
- 3) Rest of the robots move to meet the leader and follower robots by tracking the shortest distance path to reach the point of minimum distance. This ensures effective use of energy in reaching the goal. Then move to step 4.
- 4) The robot with highest leftover energy is assigned the role of leader and the robot with second highest leftover energy is assigned the role of follower robot. Then move to step 1.

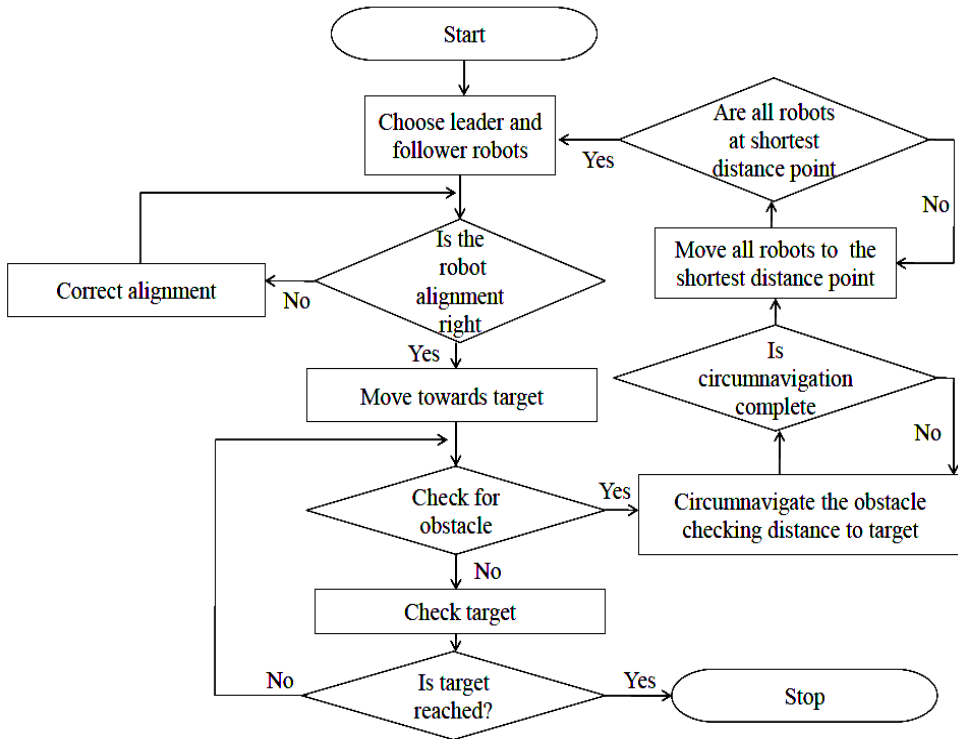


Fig. 3. Flow chart

There are many advantages for this approach in comparison to conventional algorithms especially when it comes to systems involving more than one robot. One of the most basic merits of the system is that it improves performance with increasing number of robots. Further, due to increased number of robots involved, the probability of success is also

boosted considerably. Effective distribution of energy ensures that maximum number of robots reach the goal.

3 Algorithm analysis

3.1 Travel time

Without loss of generality, this paper considers an obstacle shape as shown in Fig.4. Tangents are drawn from S in Fig. 4 (i) and S_1, S_2 in Fig. 4 (ii) to the obstacle. These tangents provide the possible locations of goal since the obstacle does not obstruct a goal if these tangents do not enclose the goal. The worst case scenario is goal is along the tangent, or more generally, as the goal comes closer to the point of initial contact of robot and obstacle. The performance of the algorithm decrease in this scenario since the meeting point of the two robots will be away from the point of minimum distance.

Consider the arrangement as shown in Fig.4 (i) and (ii), S, S_1 and S_2 are the starting positions of robots while A, A_1 and A_2 are the initial points of contact of robots with obstacle and t is the time taken by the robots to reach A (or A_1 and A_2) from their respective starting positions (to avoid the effect of starting position on this time we have assumed a common time t for all cases). Let the time taken by a single robot to go around the obstacle and return to A be T_1 and the time taken for the leader and follower to reach M , the meeting point be T_2 . If D is the point of minimum distance to goal, then time taken by the single robot to reach D from A will be T_3 and time taken by the leader-follower robots to reach D from M be T_4 . Finally, let the time taken for the robots to move from M to goal be T_5 .

The total time taken by a single robot to reach goal, $T_{total-I}$ is given by Eq.1 and time $T_{total-M}$ taken by a multi robot system to accomplish the same is given by Eq.2. Equation 1 and Eq. 2 are compared in the following section to prove that $T_{total-I} \geq T_{total-M}$. For this comparison it is assumed that all the robots move with the same velocity. Further, to include the worst possible scenario, T_3 is taken as zero.

$$T_{total-I} = t + T_1 + T_3 + T_5 \tag{1}$$

$$T_{total-M} = t + T_2 + T_4 + T_5 \tag{2}$$

But,

$$T_3 = 0$$

$$T_4 \leq T_2$$

$$T_1 \geq 2T_2$$

Hence,

$$(t + T_1 + T_3 + T_5) \geq (t + T_2 + T_4 + T_5) \tag{3}$$

$$T_{total-I} \geq T_{total-M}$$

From above mathematical analysis, Eq.(3) states that the travel time associated with the multi robot system is always lesser than that of a single robot executing Bug-1 algorithm.

3.2 Energy distribution

Efficient distribution of energy in robots ensures task completion. The energy efficiency of the robots is analyzed using the concept of average energy of the robots and by calculating energy of individual robot. In the first section average energy of the robots involved in the multi robot system is analyzed. In the second section it is shown that after executing the

task, energy of each robot in a multi-robot system will be higher than that of a single robot system.

The energy expenditure of a robot depends on several aspects such as motor, sensor elements, Transmission and reception of data and processing. Here it shall be assumed that all forms of energy expenditure are in direct relation to distance covered. This is a good enough approximation since we assume the robots to be idle when not in motion and almost all forms of energy expenditure mentioned above happen simultaneously when robot is in motion

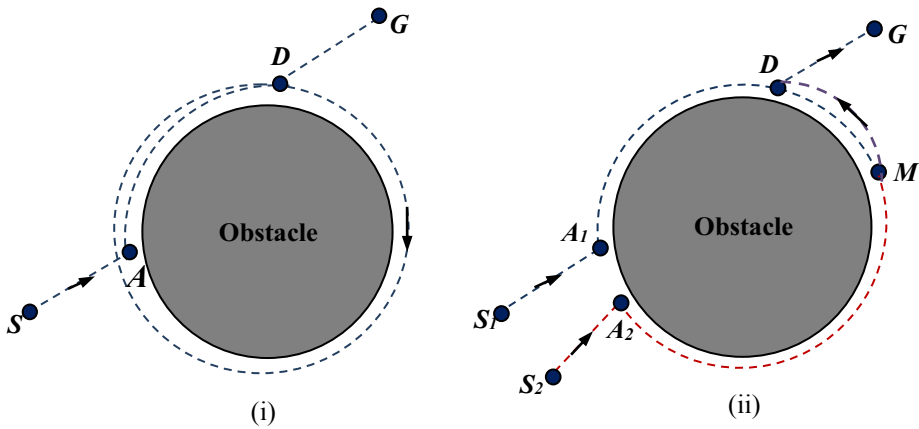


Fig. 4. Comparison of algorithms (i) Conventional Bug 1 algorithm (ii) Multi-robot Bug algorithm

3.2.1 Average energy

Total energy spend by a single robot, $E_{total-I}$ is compared with $E_{total-M}$ by using the same assumptions used to calculate travel time. To calculate the average energy from distance travelled, a constant k is used as the conversion factor as given in Eq.4 and Eq.5.

$$E_{total-I} = k(SA + curve(AM + MA + AD)) \tag{4}$$

$$E_{total-M} = k((SA_1 + SA_2) + curve(A_1M + A_2M + 2MD)) \tag{5}$$

From the analysis made in previous section, average distance covered by the leader-follower system is always lesser than the distance covered by a single robot. This implies that the average energy spent by a leader-follower system is lesser than that of a single robot executing the same task. i.e $E_{total-I} > E_{total-M}$. It also shows that the average energy expenditure shall always be lower in a multi-robot system irrespective of the number of robots involved.

3.2.2 Energy of an individual robot of multi-robot system

Consider one of the robots in the multi-robot system; if there is more than one obstacle and there are more than two robots, the initial leader-follower set of robots turn idle and another set of robots of higher energy levels take up the roles of leader and follower. Thus, assuming similar starting positions, the energy levels of individual robots of a system with more than two robots shall always remain higher compared with a two robot system. This inference arises from the fact that there is no other robot to take up the role of leader robot. So the worst case scenario in a multi robot system arises with only two robots in the multi-robot system. As in the case of average energy, distance covered multiplied by a conversion

factor shall be considered as a measure of the energy expenditure, E_{total} . Equation 6 provides the total energy expenditure of a single robot. This is compared with $E_{total-M1}$ and $E_{total-M2}$ (Eq.(7) and Eq.(8)), the individual energy expenditure of the leader and follower robots in the two robot system.

$$E_{total-I} = k(SA + curve(AM + MA + AD)) \tag{6}$$

$$E_{total-M1} = k(S_1A_1 + curve(A_1M + MD)) \tag{7}$$

$$E_{total-M2} = k(S_2A_2 + curve(A_2M + MD)) \tag{8}$$

It can be deduced from these equations and from Fig.4 that even if $curve(AD) = 0$ in Eq.6, the distance covered by a single robot executing the task is more than the distance covered by both the individual robots in a multi-robot system.

The most important inference from the above two observations is that the final energy of individual robots in a multi-robot system is always more than that of a single robot executing the task with the help of bug-1 algorithm. This shows that the energy distribution in a multi-robot system provides better success rate and it ensures all the robots reaching the goal with sufficient energy.

4 Simulation results

The theoretical results in the previous section were examined with the help of a simulation study. The energy expenditure based on the distance covered by the robots and the total time taken for task completion is considered in this study. In this simulation study the distance is considered in meters and the time taken to complete the task is considered with clock cycle as the base. By counting the number of loops involved, an estimate of the total running time is obtained. Since the same base is used to measure the performance of all cases, the values provide are reliable source for comparison.

Path traveled by the robots during simulation for different number of robots is shown in Fig. 5.(i) to Fig.5.(vi). The distance covered by each robot, total cycles in simulation and the average distance involved are also calculated. In each of the simulation, the goal is assumed to be at coordinates (4,6) and the obstacles are placed at coordinates (2,2) and (3,4) with a radius of 0.5 units. The coordinates of the starting location of the robots are assigned as (1,1) for robot 1, (1.5,1) for robot 2, (1,1.5) for robot 3, (1.2,0.7) for robot 4, (1.25,1.35) for robot 5 and (1.3,0.9) for robot 6.

The results obtained from the simulations were plotted to draw inference regarding the travel time and average distance travelled by the multi robot system in comparison to a single robot executing the conventional Bug-1 algorithm. Figure 6 (i) displays the variation of average distance and Fig. 6 (ii) displays the variation in total travel time as the number of robot increases. The slight deviation from the decreasing trend occurs in the 5-robot and 6-robot system, which can be justified by the starting position that is further away from the goal which results in extra distance to cover thus taking more time to reach goal.

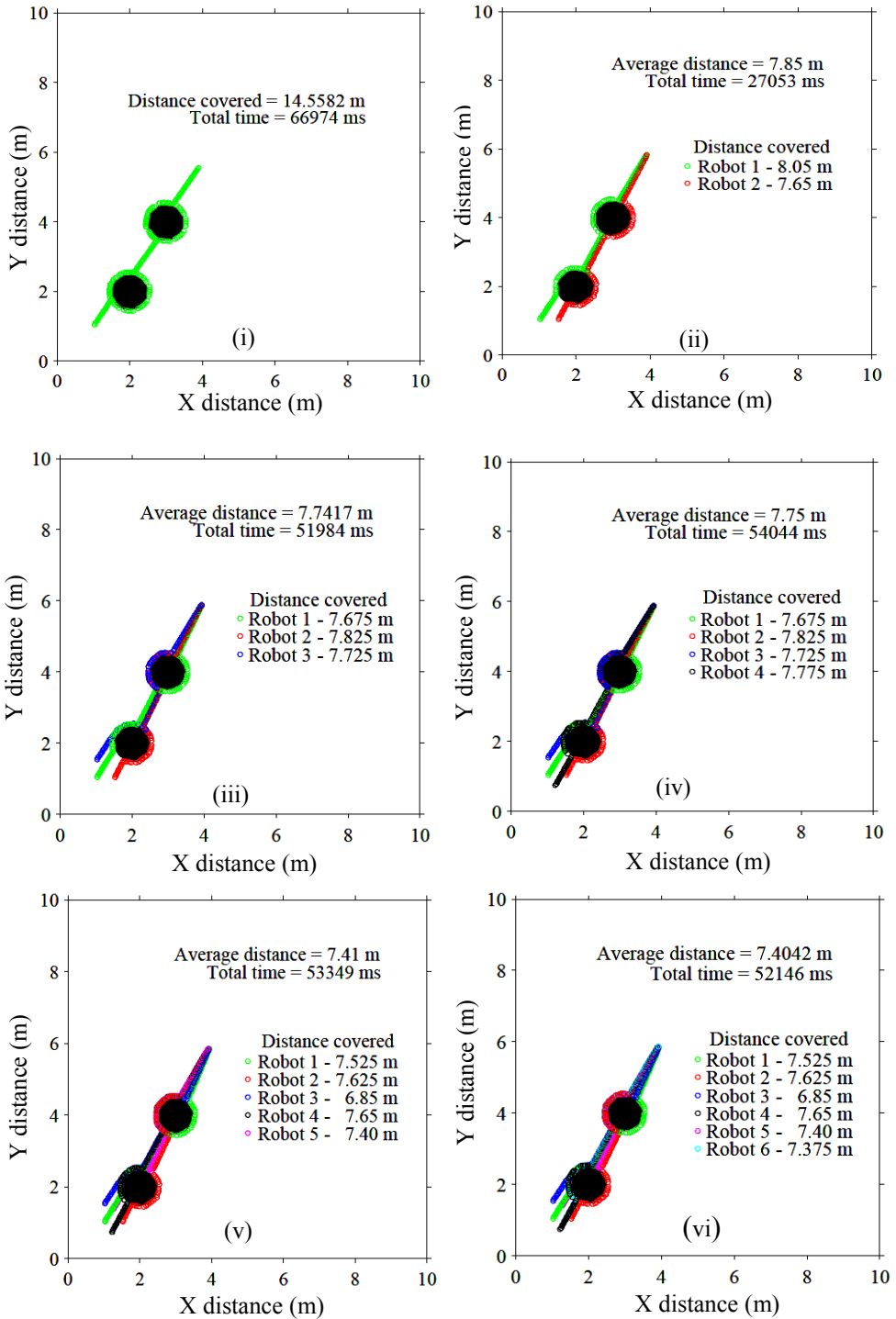


Fig. 5. Path tracked by robots (i) Single robot (ii) Dual robot system (iii) Three robot system (iv) Four robot system (v) Five robot system (vi) Six robot system

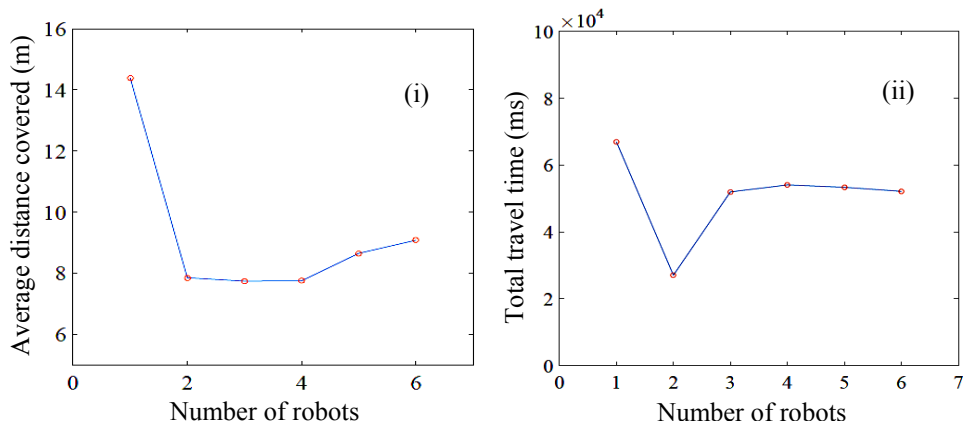


Fig. 6. Simulation results (i) Average distance plotted against number of robots in the system (ii) Total travel time plotted against number of robots in the system

The plots validates that the average distance travelled by the robots decreases as the number of robots involved increases. Another inference drawn is that the distance travelled by each robot is lower than that of a single robot executing Bug algorithm. The total time required for task completion reduces considerably when multiple robots are used instead of single robot. It was also observed that as the number of robots increases in multi robot system the total time to complete the task increases

5 Conclusions

This paper presented a more innovative form of Bug-1 algorithm for multi-robot system. The strategy makes use of exchange of information across a leader-follower robot system. The algorithm provides better performance in comparison to conventional Bug-1 algorithms. Theory behind this better performance was discussed in support. The travel time and energy distribution of the robots have been compared with that of a Bug-1 algorithm and it shows that the algorithm provided in the paper have a shorter travel time and better energy conservation strategy. A further argument for this strategy is that it also works better than conventional algorithms in many mazes where the conventional algorithm fails to deliver proper results. The simulation results suggests that the improved nature of this algorithm makes it a very effective in cases involving more than one robot dedicated to execution of a common task such as in swarm robotics

References

1. V. Lumelsky, A. Stepanov, IEEE Trans. Automat. Contr. **31**, 1058–1063(1986).
2. M. Zohaib, S.M Pasha, N. Javaid, J. Iqbal, *International multi topic conference*, 291-299 (2013)
3. R.A. Langer, L.S. Coelho, G.H Oliveira, *IEEE Inernational. Conference on control application*, 403-408 (2007)
4. J. Ng, T. Braunl, J. of Intelligent and Robotic Systems , **50,1**(2007)
5. T. Facchinetti, t-bots: a coordinated team of mobile units for searching and occupying a target area at unknown location

6. L. Consolini, F. Morbidi, D. Prattichizzo, M. Tosques, *Automatica*, **44**, 5, 1343-1349, (2008)
7. R.G. Lord, D.J Brown, S.J Freiberg, *Organizational behavior and human decision processes*, **78**, 3, 167-203, (1999)
8. J.Y.S. Luh, Y.F Zheng, *The Int. J. of robotics res.*, **6**, 3, 60-70, (1987)
9. Y.F Zheng, J.Y.S Luh, *IEEE conference on decision and control* , 1761-1766, (1985)