Path Planning of Intelligent Mobile Robot Using Modified Genetic Algorithm

Nadia Adnan Shiltagh, Lana Dalawr Jalal

Abstract - One aspect of interest in robotics is planning the optimal path for a mobile robot. The objective of path planning is to determine the shortest feasible path with the minimum time required for mobile robots to move from a starting position to a target position. In this study, Modified Genetic Algorithm (MGA) is developed for a global path planning, and the application of MGA to the problem of mobile robot navigation is investigated under an assumption that an environment model has been established already. The proposed algorithm read the map of the working environment which expressed by grid model and then creates an optimal or near optimal collision free path. The MGA algorithm was simulated using MATLAB R2012a. Adaptive population size without selection and mutation operators are used in the proposed algorithm. The simulation results demonstrate that this algorithm has a great potential to solve the path planning with satisfactory results in terms of minimizing distance and execution time.

Index Terms - global path planning, intelligent mobile robot, modified genetic algorithm, optimal path.

I. INTRODUCTION

The most fundamental intelligent task for a mobile robot is the ability to plan a valid path from its initial to terminal position while avoiding all obstacles located on its way [1], this is known as robot path planning [2]. The efficiency of the mobile robot path planning is considered as an important matter since one of the main concerns is to find the destination in a short time. Accordingly, a desirable path should result from not letting the robot waste time taking unnecessary steps or becoming stuck in local minimum positions. Furthermore, a desirable path should avoid all known obstacles in the area [3].

The path planning problem for mobile robots has been an active research area for many years which is started from mid-1970. There are many methods have been proposed to address the path planning problem for mobile robots. Each method differs in their effectiveness depending on the type of application environment and each one of them has its own strengths and weaknesses. The choice of a good method is necessary in order to achieve both quality and efficiency of a search, and some optimization criteria with respect to time and distance must be satisfied.

[4] describes the various methods applied for navigation of an intelligent mobile robot. They found that the heuristic approaches (Fuzzy logic (*FL*), Artificial Neural Networks (*ANN*), Neuro-Fuzzy, Genetic Algorithm (*GA*), Particle Swam Optimization (*PSO*), Ant Colony Optimization (*ACO*) and Artificial Immune System) gave suitable and effective results for mobile robot navigation. Using the heuristic approach, the mobile robot can navigate safely among the obstacles without hitting them and reach the predefined target point. These approaches are also helpful for the solution of the local minima problem.

Many researchers have been worked in the field of path planning using *GA* to generate the optimal path by taking the advantage of its strong optimization capability. The idea of using *GA* approach to solve the mobile robot path planning problem in static environment is presented in [5]. Their experiments results showed that the proposed approach is effective and efficient in handling different types of tasks in static environments. [6] used *GA* to find a feasible path for the mobile robot in an environment with obstacles. They implement grid-based environment model, which is frequently used in indoor applications, as the motion area of mobile robot. [7] proposed a method of mobile robot path planning based on modified genetic algorithm to find a feasible path for the mobile robot in the dynamic environments. Their simulation results demonstrate that the proposed method achieved considerable improvements, with respect to the basic *GA*, in convergence speed.

The objective of this study is performing modification in the *GA* to increase the capability of this algorithm to generate an optimal path for mobile robot navigation. Moreover, the *MGA* is implementing for a global path planning to determine the shortest/optimal path for a mobile robot from start point to target point without colliding any obstacles in the known working environment in minimum running time.

II. NAVIGATIONAL PLANNING FOR MOBILE ROBOTS

The navigational planning problem persists in both static and dynamic environments. In a static environment, the position of obstacles is fixed, while in a dynamic environment the obstacles may move at arbitrary directions with varying speeds, lower than the maximum speed of the robot [8]. Navigation consists of two essential components known as localization and planning. Localization in robotics refers to the ability of determining accurate positions in the search space according to the environmental perceptions gathered by sensors. Planning is considered as the computation of a path through a map which represents the environment.

Navigation and obstacle avoidance are very important issues for the successful use of an autonomous mobile robot [10]. All obstacle avoidance approaches find a path from an actual position of the controlled robot to a desired goal position, while all these parameters stand as the inputs of the algorithm; the output is the optimal path from start to goal [11]. Optimal path planning is an important issue in navigation of autonomous mobile robots, which is to find an optimal path according to some criteria such as distance, time or energy while distance or time being the most commonly adopted criterion [12].

Manuscript received on May, 2013.

Nadia Adnan Shiltagh, Computer Engineering, University of Baghdad, Baghdad, Iraq.

Lana Dalawr Jalal, Electrical Department, Faculty of Engineering, University of Sulaimani, Sulaimani, Kurdistan Region, Iraq.

III. PATH PLANNING

Path planning research of autonomous mobile robot has attracted attention since the 1970. Research in this area has increased due to the reason that mobile robots are now applied in various applications [13]. The problem of path planning consists of finding a sequence of moves for rearranging the robot in a certain environment, the robot occupy certain position in the environment initially, the task is to move the robot to the given goal positions, the robot must avoid obstacles in the environment. The main difficulties for robot path-planning problems are computational complexity, local optimum and adaptability [14].

There are two categories of path planning algorithms: namely the global path planning (off-line) and the local path planning (on-line), based on the availability of information about the environment. Global path planning of robots in environments where complete information about stationary obstacles and trajectory of moving obstacles are known in advance, so that the robot only needs to compute the path once at the beginning and then to follow the planned path up to the target point. When complete information about environment is not available in advance, mobile robot gets information through sensors as it moves through the environment, this is known as on-line or local path planning [15].

IV. MODIFIED GENETIC ALGORITHMS (*MGA*)

Genetic algorithm has rapid search and high search quality, in the existing algorithm, there are three problems associated with this method. First, the initial population contains many infeasible paths. Second, there are not sufficient heuristic knowledge based genetic operators. Third, after a generation, offsprings may contain infeasible paths [14]. These problems can be overcome by introducing of an improved mechanism into the *GA*.

In this study, a modified genetic algorithm is proposed based on the traditional genetic algorithm. Simulations have been done to illustrate the significant and effective impact of these algorithms; the proposed algorithms are simulated using MATLAB R2012a.

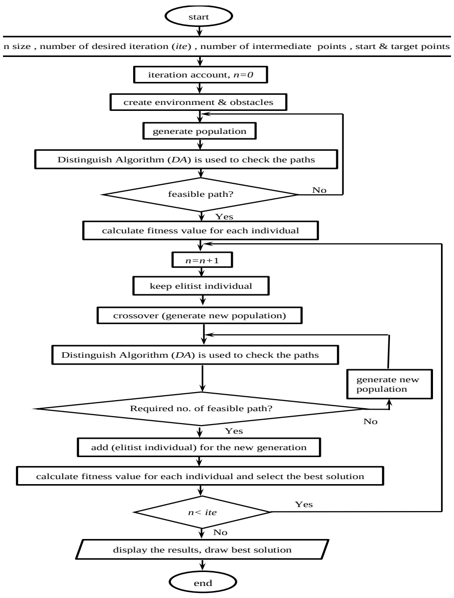

Fig 1: Flowchart of Modified Genetic

The *MGA* does not require any encoding scheme because it uses the real representation, which is reduce calculation time because no time is required for encoding and decoding of the population. The *MGA* starts with generation of initial population; which may contain feasible and infeasible path. To generate the initial population and verify that the generated path in the initial population is feasible or not, Distinguish Algorithm (*DA*) is used, and also, the *MGA* leads to quick generation of a path solution because no selection and mutation operations are used which leads to improve the execution time. The goal of *MGA* is to minimize the total distance from the starting position to the desired position without colliding with any of the obstacles in the environment with satisfied time. The flowchart of the modified genetic algorithm is illustrated in Fig 1.

AlgorithmIn this study, mobile robot environment which occupied by a number of static obstacles is represented by a grid-based model, consider a two-dimensional (*2D*) square map overlaid with a uniform pattern of grid points. Grid-based model makes the calculation of distance and representation of obstacle easier. To verify the effectiveness of *GA*, the simulation has been applied for the working environment which is presented in Fig 2.

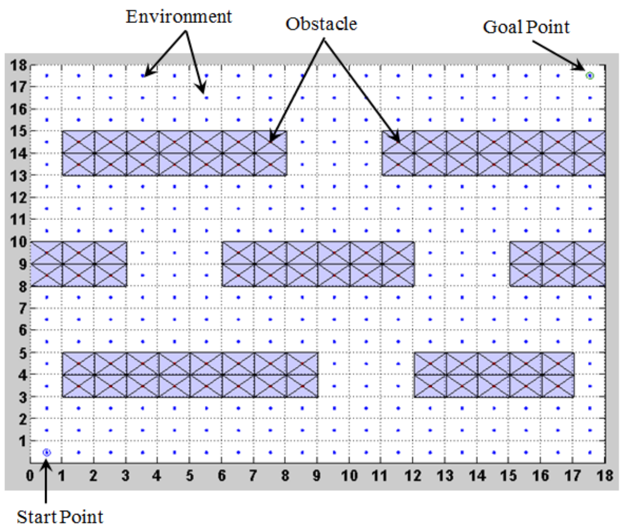

Fig 2: Working environment

As displayed in the figure, the blue grids represent obstacle free areas where mobile robot can move freely. There is no unit used to measure the path length because each cell in the environments can represent any unit. The circle sign in the environment is the robot's starting and goal location. The grid size, the number of obstacles, the coordinate of the start and target point are shown in Table 1.

Table 1: Grid size, the start and target point, and number of obstacles

grid size [unit]	start point [unit]	target point [unit]	number of obstacles
18 × 18	x=0.5 , y=0.5	x=17.5 , y=17.5	76

Obstacle areas in the working environment are represented by Blue Square and it has an average area 1 × 1 unit. Boundaries for obstacles area are formed by their actual boundaries plus a safety distance that is defined with consideration to the size of the mobile robot.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

The obstacles are putting randomly but carefully placed such that they keep some distance from the starting point and the target point to make sure that the robot has some space to move in the begging.

A potential path is formed by line segment which is connecting the points falling on the grids of the working environment .The points are represented by their respective x and y coordinates. In *GAs*, a possible solution to a problem is referred to as an individual, which is represented by a computational data structure called a chromosome. Each chromosome consists of a string of cells called genes. The value of each gene is called allele. In the *MGA*, an allele is a real number, and the length of a chromosome is constant. Each path represented by two chromosomes, the first chromosome is for *x*-coordinate and the second one is for *y*-coordinate.

All individuals of the initial population are assumed to be generated randomly. This is lead to generate large number of infeasible paths which intersect an obstacle, and infeasible paths should be avoided. If there is/are obstacle(s) either in the vertical direction or horizontal direction, the mobile robot has to keep itself away from the obstacles.

Initial population stored in a single matrix. Each row corresponds to a particular solution. After generating the initial population, and to allow the robot to move between its current and final configurations without any collision within the surrounding environment, each individual must be checked whether it intersects an obstacle or not.

A. Distinguish Algorithm (DA)

Because there are both feasible path and infeasible path in the population, therefore the Distinguish Algorithm (*DA*) is used to check the paths, whether the path is feasible or not, in order to come out with all feasible individuals in the population. After applying this algorithm, feasible path will be used for next generation and infeasible path will be deleted.

B. Fitness Function

In *GA* the fitness function of each chromosome is evaluated in terms of its path distance. Thus, the better chromosome has the smaller distance. The length of the feasible path is compute as:

$$d_{(i)} = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \quad (1)$$

The objective function of the overall path can be expressed as:

$$dist_{(j)} = \sum_{i=1}^{h} d_{(i)} \quad (2)$$

where h is the number of links that consist the path, $d_{(i)}$ is the distance between two points, x_i and y_i are robot's current horizontal and vertical positions, x_{i+1} and y_{i+1} are robot's next horizontal and vertical positions, and $dist_{(j)}$ is the distance of the j^{th} path in the working environment.

The fitness function is the inverse of the total distance which is Euclidean distance. The Euclidean distance between starting and target point is the length of the line segment connecting them. The fitness function used in this study is [14]:

$$F_{(j)} = 1/dist_{(j)} \quad (3)$$

where *F* is the fitness function and *j* represents the j^{th} path.

It is obvious that the best individual will have the maximum fitness value. At each generation (iteration), all the chromosomes will be updated by their fitness. A chromosome with good fitness has a much higher probability than other inferior chromosomes to appear in the next generation.

C. Elitism

In order to keep the best chromosome from each generation, the elitism method is employed. In Elitism It is obvious that the best individual will have maximum fitness value. The main goal of the elitism rule is to keep the best chromosome from the current generation. Thus, under this rule, the best chromosome from each generation will not undergo any mutation or crossover event and will safely move to the next generation. Since the best or elite member between generations is never lost, the performance of *GA* can significantly be improved. The remaining chromosomes are then sorted according to their fitness. Since small population sizes lose diversity very fast, therefore in the proposed algorithm no selection operator is used, and all the remaining chromosomes will be selected to undergo the crossover operator. Using this approach will increase the expectation of maintaining diversity in the population.

D. Crossover Operator

In crossover a group of chromosome undergoes crossover at each generation. All the crossover events are controlled by a certain crossover probability (Pc). The algorithm creates a random number in range [0 1] for each chromosome. If the generated number is less than Pc, the chromosome is a candidate for the crossover event, otherwise the chromosome proceed without crossover. The left most genes and the right most genes will avoid the crossover event since these two points are the start and target points and cannot be eliminated. For the purpose of diversity, the crossover point bit is randomly selected in each generation. Single-point crossover operator is used in this algorithm. The genes of two parents' individuals before or after the crossover point are swapped. Then the individual of the father replaced by individual of the offspring after crossover, a new population would be produced. Finally, the *DA* is used to check the paths of the newly generated off springs. There is no mutation operator used in the proposed modified genetic algorithm.

E. Generation Algorithm

Commonly in the complex environment along with high density of obstructions the number of generated path is inefficient. Generation algorithm is used to increase the population and help to prevent the stagnating at local optima, then the new generated population must be checked for infeasibility, if the number of generated path is still insufficient a new population is generated, this process is repeated over again until the desired number of feasible path is reached. So the proposed method does not use fixed population size but adaptive population size. Generated population formed in this way increase the efficiency of the proposed algorithm, and will not lose the overall genetic algorithm searching capabilities. In the next iteration, generation algorithm continues generate path instead of infeasible path.

Genetic algorithm is terminated when the maximum number of iteration exceeds a certain limit, and also terminated when it does not find a path from the start point to the target point. Simulation Results and Dissuasion

In this section, the simulation results of path planning using Modified Genetic Algorithm (*MGA*) is presented to find the optimal path along the obstacle-free directions. To illustrate its wide applicability and their effectiveness, the proposed algorithm is implemented to solve the path planning problem through the computer simulation for working environment. The programs are written in MATLAB 2012 and run on a computer with 2.5 *GHz* Intel Core i5 and 6 *GB RAM*.

V. THE IMPLEMENTATION OF *MGA* IN PATH PLANNING

In this section, the implementation of *MGA* to solve path planning problem is demonstrated. The working environment has been proposed to test the performance of this algorithm to find the optimal or near optimal path with satisfied time. For the proposed algorithm the simulation parameters are set as: population size =100, P_c (crossover probability) =0.5. In all cases, an optimal path is formed by line segment which is connecting the points (5 points) falling on the grids of the working environment.

The performance of the *MGA* has been tested by applying *MGA* to the working environment presented Fig 2. Since *MGA* is stochastic algorithm, every time they are executed they may lead to different trajectory convergence. Therefore, multiple test groups were considered. The best results obtained after implementing the *MGA* are shown in Table 2, and the best discovered path using *MGA* are shown in Fig 3 to Fig 9.

Table 2: The simulation results for the working environment using *MGA*

Simulation Results			
Initial Population	Iterations	Distance	Elapsed time [s]
100	20	30.86	374.47
100	10	33.77	238.84
100	5	34.79	148.99
100	4	36.95	111.37
100	3	37.46	108.81
100	2	38.09	83.45
100	1	39.65	42.31

The simulation results of the *MGA* revealed that the elapsed time is 374.47 seconds to find shortest generated path (distance) with the length of 30.86. From these results, one can concluded that, the required time to find optimal path is small.

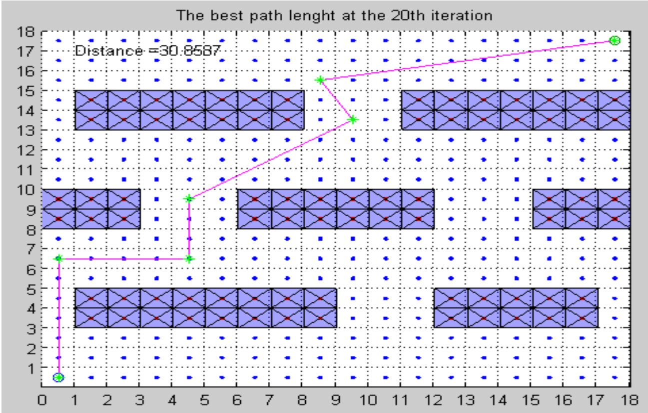

Fig 3: Path generated for the working environment using *MGA* (20 iterations)

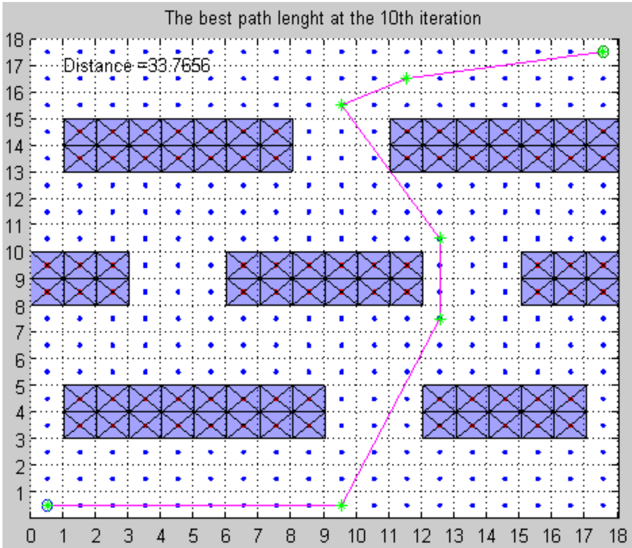

Fig 4: Path generated for the working environment using *MGA* (10 iterations)

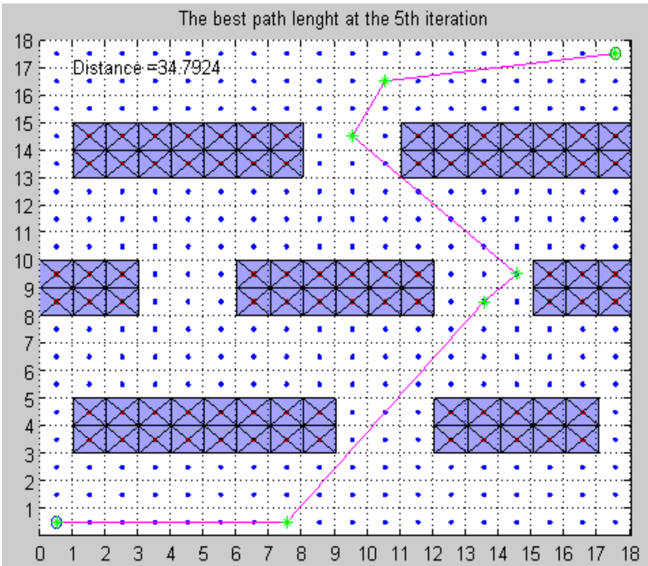

Fig 5: Path generated for the working environment using *MGA* (5 iterations)

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

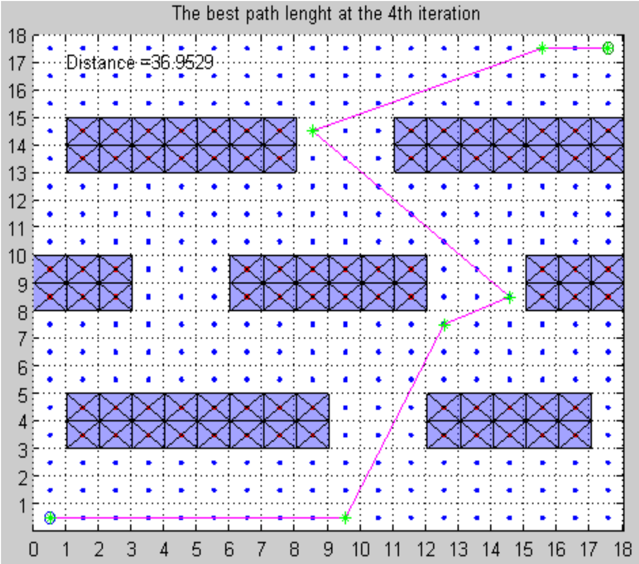

Fig 6: Path generated for the working environment using *MGA* (4 iterations)

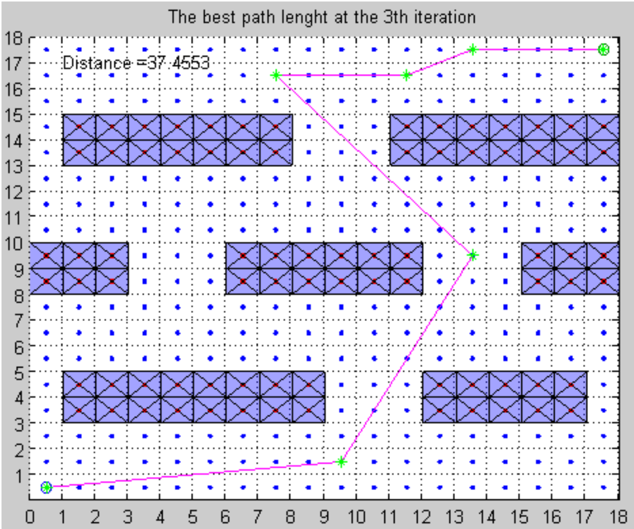

Fig 7: Path generated for the working environment using *MGA* (3 iterations)

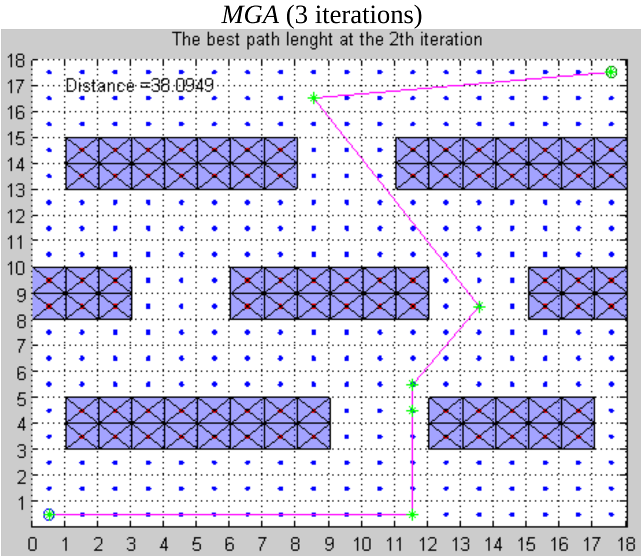

Fig 8: Path generated for the working environment using *MGA* (2 iterations)

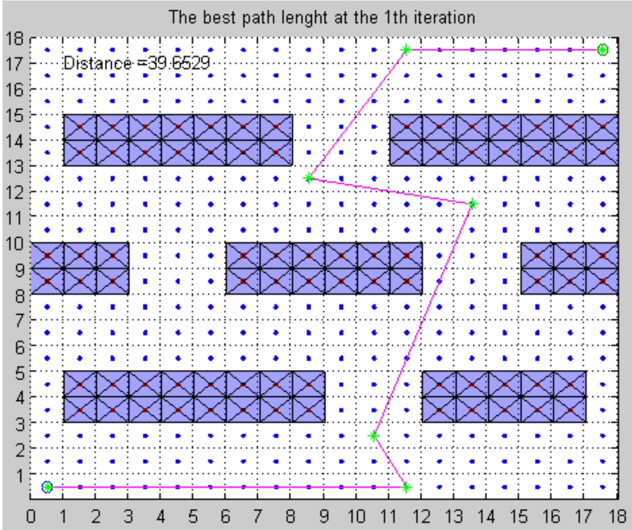

Fig 9: Path generated for the working environment using *MGA* (1 iteration)

In order to show that the algorithm does not converge when there is no path to target; the algorithm has been implemented to the working environment which turned to closed environments. The simulation results (see Fig 10) show that if the *MGA* could not find a path between the start point and the target point for the working environment, an error message is returned, so that the mobile robot cannot find the target.

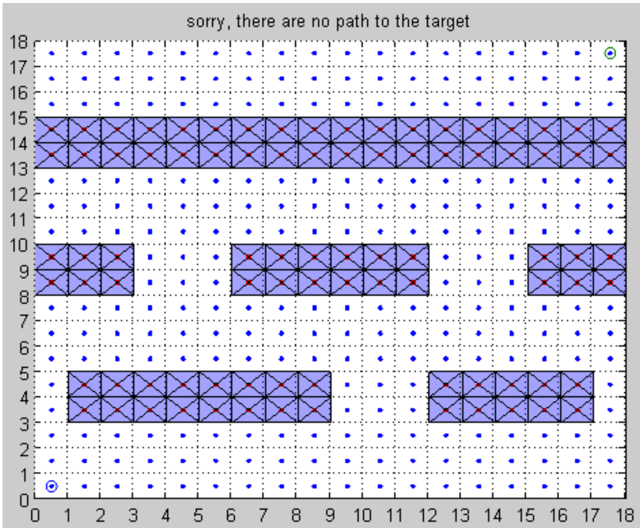

Fig 10: Closed working environment

VI. CONCLUSIONS

This study investigated path planning for a mobile robot by application of an efficient optimization algorithm known as Modified Genetic Algorithm (*MGA*). A *MGA* algorithm is used to find optimal path for the mobile robot in working environment with obstacles. This algorithm is programmed using MATLAB 2012. From the simulation results the following conclusions are drawn:

The proposed algorithm (*MGA*) are capable of effectively guiding a robot moving from start position to the goal position and find optimum/shortest path without colliding any obstacles in complex environments.

Using adaptive population size without selection and mutation operators in the proposed *MGA* led to improve the execution time and the computational cost. Adaptive population size grows depending on the size, structure and the number of obstacles in the environment which is led to improve the execution time.

The *MGA* algorithm has the capability to find path planning of the closed environments which there is no path between the start point and the target.

Future research can investigate the performance of *MGA* algorithm in dynamic environment. Another future direction is to examine the effectiveness of *MGA* algorithm with physical robots in a real-world application. Also future research can investigate the application and examine the performance of *MGA* algorithm to solve obstacle avoidance to real objects limited to three dimensions (*3D*) environment.

REFERENCES

[1] S. Kolski, “Mobile Robots Perception and Navigation”, Advanced Robotic Systems International and pro literaturVerlag, Germany, 2007.

[2] G.Yogita, G. Kusum, Artificial Intelligence in Robot Path Planning, International Journal of Soft Computing and Engineering (IJSCE), , 2012, 2(2): 471-474.

[3] K. M. Han, “Collision free path planning algorithms for robot navigation problem”, Master Thesis, University of Missouri-Columbia, 2007.

[4] P. K. Mohanty, D. R. Parhi, “Controlling the Motion of an Autonomous Mobile Robot Using Various Techniques: a Review”, Journal of Advance Mechanical Engineering, 2013, 1: 24-39

[5] I. Al-Taharwa, A. Sheta, M. Al-Weshah, “A Mobile Robot Path Planning Using Genetic Algorithm in Static Environment”, Journal of Computer Science, 2008, 4 (4): 341-344.

[6] A. Tuncer, M. Yıldırım, K. Erkan, “A Motion Planning System for Mobile Robots’, Advances in Electrical and Computer Engineering, 2012, 12(1): 57-62.

[7] W. Yan-Ping, W. Bing, “Robot Path Planning Based on Modified Genetic Algorithm”, 2nd International Conference on Future Computer and Communication, 2010 3:725-728.

[8] A. Konar, “Artificial Intelligence and Soft Computing Behavioral and Cognitive Modeling of the Human Brain”, CRC Press LLC, 2000.

[9] S. Dutta, “Obstacle Avoidance of Mobile Robot using PSO-based Neuro Fuzzy Technique”, International Journal of Computer Science and Engineering, 2010, 2(2): 301-304.

[10] P. Li, X., Huang, M. Wang, “A New Hybrid Method for Mobile Robot Dynamic Local Path Planning in Unknown Environment”, Journal of Computers, North America, 2010, 5(5): 773-781.

[11] M. Saska, M., Macas, L., Preucil, L. Lhotska, “Robot Path Planning using Particle Swarm Optimization of Ferguson Splines”, In Proceedings 11th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2006), Prague, Czech Republic, 2006, pp. 833-839.

[12] P. Raja, S., Pugazhenthi, “Path Planning for Mobile Robots in Dynamic Environments using Particle Swarm Optimization”, International Conference on Advances in Recent Technologies in Communication and Computing, 2009, pp. 401-405.

[13] N. Sariff, N. Buniyamin, “An overview of autonomous mobile robot path planning algorithms”, 4th Student Conference on Research and Development, Scored 2006, pp. 183-188.

[14] S. C. Yun, V. Ganapathy, L. O. Chong, “Improved genetic algorithms based optimum path planning for mobile robot”, Proceedings of ICARCV, 2010, pp. 1565-1570.

[15] P. Raja, S. Pugazhenthi, “Optimal path planning of mobile robots: A review”, International Journal of Physical Sciences, 2012, 7(9): pp. 1314 - 1320.

[16] P. K. Mohanty, D. R. Parhi, “Controlling the Motion of an Autonomous Mobile Robot Using Various Techniques: a Review”, Journal of Advance Mechanical Engineering, 2013,1: 24-39.

[17] W. Yan-Ping, W. Bing, “Robot Path Planning Based on Modified Genetic Algorithm”, 2nd International Conference on Future Computer and Communication, 2010, 3: 725-728.