Journal of Engineering and Development, Vol. 16, No.4, Dec 2012 ISSN 1813- 7822

Optimal Path Planning for Mobile Robot Based on Genetically Optimized Artificial Potential Field

Dr. Mohamed Jasim Mohamed
University of Technology
Control and Systems Eng. Dep.
dr.mohamed_jasim@uotechnology.edu.iq
moh62moh@yahoo.com

Mustaffa Waad Abbas
University of Technology
Control and Systems Eng. Dep.
Mustaffa.waad@yahoo.com

Abstract

This paper introduces a modified technique to find the shortest path between two points in known static environment for the mobile robot. The path planning in our proposal is based on the assumptions that; the robot is a small mass moving in two dimensions space with known static obstacles and subjected to an attractive force applied by the target as well as repulsive forces resultant from the obstacles. The combination of these forces moves the mass of robot directly toward the target in a manner that the mass of robot avoids all the obstacles on this way. The potential field is adapted (deformed) by manipulating potential field parameters according to static rules. The path of the mobile robot from start point to target point is optimized by choosing best values of the field parameters that give optimum form of potential field. The proposed genetic algorithm is used to search about these best values of field parameters. Simulation studies are carried out to verify and validate the effectiveness of the proposed method.

***Keywords*:** Genetic Algorithm (GA), Mobile Robot, Global Path Planning, Potential Field**.**

التخطيط الامثل لطريق الروبوت النقال المستند على مجال القوى الاصطناعي المحسن وراثيا

الخلاصة

هذا البحث يقدم تقنية محسنة لايجاد اقصر طريق بين نقطتين في بيئة ثابتة ومعروفة للروبوت النقال. ان تخطيط الطريق في مقترحنا هذا مستندا على الافتراضات التاليه:ان الروبوت النقال هو كتلة صغيرة تتحرك في فضاء ذو اتجاهيين مع وجود عوائق ثابتة معروفة ويخضع الى قوة جاذبة من قبل الهدف كذلك لقوى نبضية ناتجة من العوائق.ان محصلة هذه القوى تحرك كتلة الروبوت مباشرةً باتجاه الهدف باسلوب تجعل الكتلة تتجنب جميع العوائق الموجودة في ذلك الطريق . ان مجال

القوى يكييف (يشوه) بواسطة التلاعب بمعاملات مجال القوى تبعا لقواعد ثابتة. ان طريق الروبوت النقال من نقطة الانطلاق الى نقطة الهدف يحسن بواسطة اختيار افضل قيم لمعاملات المجال التي تعطي أمثل شكل لمجال القوى. الخوارزمية الوراثية المقترحة استخدمت للبحث عن هذة القيم المثلى لمعاملات المجال. لقد تمَ دراسةُ المحاكاةِ لغرضِ التحققِ والمصادقهِ على فعالية الخوارزمية المقترحةْ.

I. INTRODUCTION

Optimal path planning for a single mobile robot had considered the essential first step for realizing autonomous ground vehicles [1]. There are numerous methods and algorithms initially developed for a single mobile robot working in an environment containing static obstacles. Among these methods and algorithms like evolutionary algorithms; GAs [2] and particle swarm optimization (PSO) [3] and mathematical algorithms; probability recursive function [4] and other geometric algorithms like VORONOI diagram [5].

There are several problems associated with the optimal path planning algorithms: first, how to ensure that the path will not collide with any obstacle. Second, how force the path to reach the target. Third, how many computation processes needed to overcome the two above problems while finding the optimal path. In our attempt to solve these problems, we proposed a new hybrid optimal path-planning algorithm. This hybrid algorithm combines the ability of potential function to plan a free collision path (between a start point and a target point) and the strong optimization ability of GA that optimize the parameters of potential function to form the shortest path.

The reminder of this paper is organized as follows: section II presents a brief introduction to the conventional artificial potential field. Section III presents an overview of the traditional GAs. Section IV introduces the problem description. In section V, the proposed algorithm is introduced and in section VI, simulation results of two examples are presented. Finally, section VII introduces the conclusions and discussions.

II. CONVENTIONAL ARTIFICIAL POTENTIAL FIELD METHOD

The artificial potential field method is a fictitious one raised up by Khatib [6]. This approach uses repulsive potential fields around the obstacles (and forbidden regions) to force the robot away from obstacles and an attractive potential field around the goal to attract the robot. The robot experiences a combination of these forces, which equal to the negative of the total potential gradient of the two potential fields. This force drives the robot towards its goal until it reaches a minimum and then the robot stops [7]. The conventional attractive U_{att} and repulsive potential U_{rep} functions are shown in equation (1) and (2) respectively:

$$U_{att} = \frac{1}{2}\xi d^2(q, q_{goal}) \quad (1)$$

$$U_{rep} = \begin{cases} \frac{1}{2}\eta\left(\frac{1}{d(q,o)} - \frac{1}{\rho}\right)^2, & d(q,o) < \rho \\ 0, & d(q,o) \geq \rho \end{cases} \quad (2)$$

In the above formula, ξ and η are the gain coefficients of attraction and repulsion functions respectively, ρ is the effected distance of obstacle, $d(q, q_{goal})$ is the Euclidean distance between the robot location and the target, $d(q,o)$ is the minimum distance between the affected area of obstacle to the robot location.

The combination of the negative gradients of attractive and repulsive forces is shown in equation (3) which forms a vector field. This field is called potential field**.** A simple artificial potential field for single obstacle is shown in Figure (1). This field leads the robot to the target away from the obstacles.

$$F = -(\nabla U_{att} + \nabla U_{rep}) \quad (3)$$

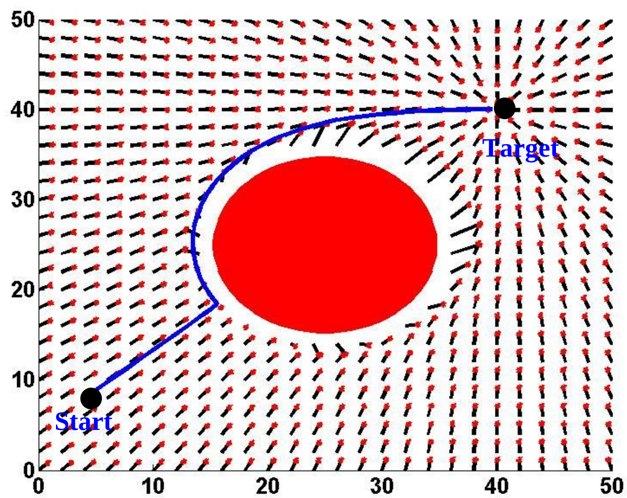

Figure (1) Simple artificial potential field.

Although, the artificial potential field path planning method is fast and efficient, it has some drawbacks and limitations [8]. These drawbacks are the local minima point where the robot reaches it and stop as illustrated in Figure (2-a) and the robot does not pass between the close obstacles as shown in Figure (2-b). According to these problems, this method will not ensure that the resultant path is an optimal path.

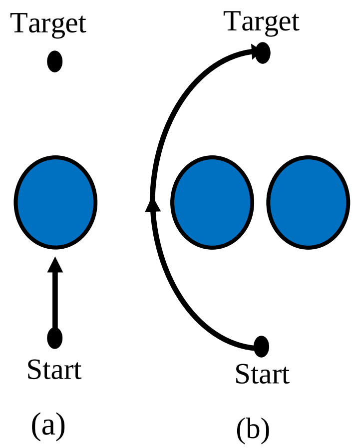

Figure (2) The artificial potential field path planning. (a) Local minimum (b) The robot does not pass between close obstacles.

III. GENETIC ALGORITHM

GAs are nondeterministic stochastic search/optimization methods that inspired by the theories of evolution and natural selection to solve a problem within a complex solution space. These algorithms encode a potential solution to a specific problem on a simple chromosome (e.g., strings of bits or letters). Each chromosome consists of genes (e.g., bits), each gene being an instance of a particular allele (e.g., 0 or 1). GA is a method for moving from one population of chromosomes to a new population by using a kind of natural selection together with the genetics inspired operators of crossover and mutation. Each chromosome in the population receives a measure of its fitness in the environment .The selection operator chooses those chromosomes in the population that are allowed to reproduce, and on average, the fitter chromosomes produce more offspring than the less fit ones. This evaluate-select-recombine sequence is repeated generation after generation until satisfactory (desired) solution is found.

GA is used to find the optimal path for mobile robot since it is less likely to be trapped at a local optimum. This ability of GA is due to several paths that are used as candidate solutions to the problem as well as its ability to search the entire search space rapidly [9]. Although it has rapid search and high search quality, there are two problems associated with this method. First, the initial population contains many infeasible paths, which have negative influence on the performance of the GA. Second, after generations, offspring may represent infeasible paths [10].

IV. PROBLEM DESCRIPTION

Our problem here is to find the shortest feasible (free colliding) path between two known points in two dimensions static environment for mobile robot. The environment is assumed well-known environment with many static obstacles. In addition, the spend computation effort to search about optimal feasible path is taken into account.

V. PROPOSED ALGORITHM

We introduce modified hybrid optimal path-planning algorithm. This hybrid algorithm is based on two principles. The first is the ability of potential field to plan a free collision path between a start point and a target point. The second is the strong optimization ability of GA to optimize parameters of a function. These two features motivate us to combine them in our proposed algorithm. Since, the characteristics of developed path determined according to the values of potential field parameters, therefore we proposed GA to search about the optimal values of these parameters to develop the optimal path.
In order to illustrate our proposed algorithm, we divided it into two stages. The first stage related to collection of data and settings of proposed GA, which is called preparation stage while the second stage describes the operations of proposed GA, which is called execution stage.

1. Preparation Stage

This stage consists of some operations, which are related to collection of data according to our special description of the problem.

a. Definition of Environment

Before starting to apply the proposed algorithm to find an optimal path, we need to define the underlying environment very well. The environment is described for our proposed algorithm as follow;
The environment space is classified into two parts; free space and occupied space. We assign one circle or more to surround each obstacle. The internal area of circle or circles represents the occupied space of the environment by the obstacle. Therefore, each obstacle is represented by the center/s and radius/s of one circle or circles that covers its space. The areas that present out of circles represent the free space. Figure (3) illustrates the occupied space and free space.

Furthermore, these obstacles are classified into three groups according to their distances from the straight line between the start point and the target point and we will refer to that line as direct line.

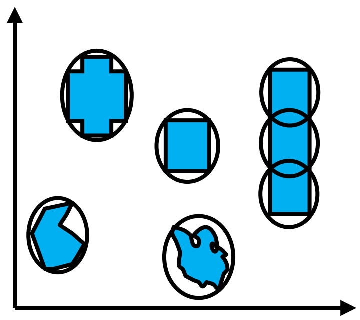

Figure (3) Obstacles representation.

The types of obstacles are listed below:

- **Frontal Obstacles (FO):** The obstacle is classified as a frontal obstacle, if it intersects the direct line or has a perpendicular distance from it equal to or less from the radius of largest obstacle that intersects the direct line. Frontal obstacles are ordered in ascending order according to their distances from the start point and they are marked like FO_1, FO_2,…...FO_n. These obstacles work like stations or control points that control the form of the parts of path that lie between the following two straight lines. The first line passes through the center of frontal obstacle and perpendiculars to the direct line (refers to that line as frontal line "FL"). The second line is the frontal line of the previous frontal obstacle or the start point in the case of FO1.
- **Adjacent Obstacles (AO):** These obstacles may be collided by the robot via its traveling to the target. Adjacent obstacles have a specific perpendicular distance from the direct line, this distance called the adjacent distance. The adjacent distance should be less from or equal to twice (or more) of the radius of largest obstacle in the whole environment. Thus, frontal obstacles are part of adjacent obstacles.
- **Unrelated Obstacles:** These obstacles lie outside the adjacent distance, where the path does not need to pass through their regions. Figure (4) shows all the three types of obstacle.

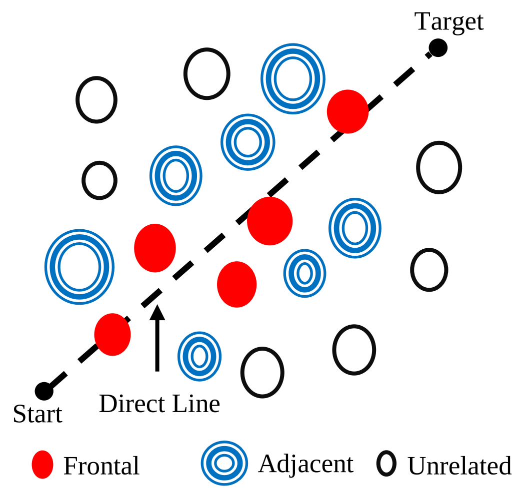

Figure (4) Obstacles classification.

The classification of obstacles plays a major role in our algorithm because it reduces the execution time and increases the reliability of algorithm.

b. Chromosome Representation

In this problem, each chromosome represents a path from a start point to a target point (or to any other point if it is an invalid path). The chromosome consists of specific number of genes. The number of genes equals to the number of frontal obstacles (FO) (i.e gene for each FO). The gene is a real number, which represents an amount of change (deformation) in the angle of the potential field of attractive force. The value of a gene affects only parts of the path that lie between a frontal line of the current frontal obstacle related to this gene and a previous one or the start point in case of FO1. The value of each gene represents radian angle, which is taken in range $-\pi/2$ to $\pi/2$. Figure (5) illustrates the role of chromosome in forming a path.

After the parts of path crosses a last frontal obstacle (FO3 in case of Figure-5), or in other word, when the parts of path lie between the last frontal obstacle and the target point, the potential field is not deformed and the path always go directly to the target by only the effect of attractive force.

2. Execution Stage

This stage describes the method of constructing the path as well as the proposed GA operations and fitness function.

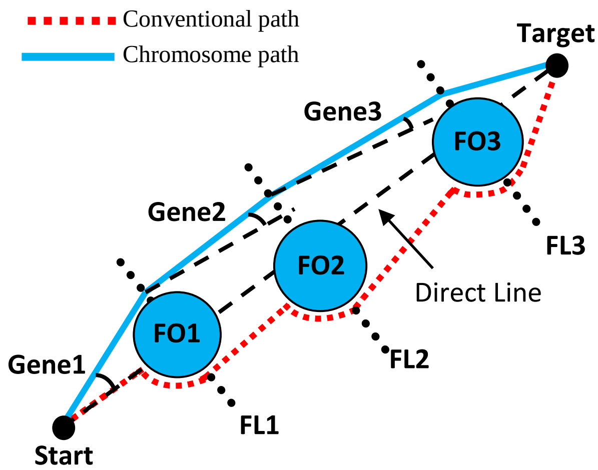

Figure (5) Conventional and chromosome paths

a. Path Construction

GA is an optimization technique used to find optimal solution for a certain problem according to a specific criterion. Thus, because we try to find the shortest path, our criterion is the length of path from start point to target point, so first we need to calculate the length of path. The deformed artificial potential field by the chromosome is used to form the path and calculates it fitness.

Each path is formed as an array of points. These points represent a Euclidean coordinates in the environment space. Each array begins from (P_0) point at the start position (P_S) and ends by (P_n) point which is either at the target position (P_T) (if the path is valid) or at any other position (if the path is not). The points of path are calculated one by one from (P_0) to (P_n) as illustrated below.

The next point (P_{i+1}) of the path is calculated according to the deformed artificial potential field as in following steps:

<u>First:</u>

Calculate the gradient of the effective attractive force at point (P_i).

The magnitude of the attractive force is taken as the Euclidean distance between (P_i) and (P_T) as shown in equation (4).

$$Uatt(P_i) = d(P_i, P_T) \qquad (4)$$

The gradient of $Uatt$ is calculated as a unit vector as in equation (5).

$$\nabla Uatt(P_i) = \frac{X_T - X_i}{d(P_i, P_T)} * i + \frac{Y_T - Y_i}{d(P_i, P_T)} * j \qquad (5)$$

Equation (5) can be written in another form as in equation (6).

$$\nabla Uatt(P_i) = cos\theta * i + sin\theta * j \qquad (6)$$

The angle θ is the angle between the vector $\overrightarrow{P_i P_T}$ and X-axis as illustrated in Figure (6).

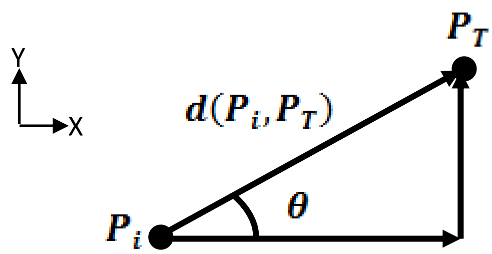

Figure (6) The Attractive force.

The gradient vector in equation (5) and (6) is not deformable gradient, which cannot be manipulated by GA.

The gradient vector of attractive force that could be deformed by GA is formed by finding the angle θ as in equation (7).

$$\theta = tan^{-1}\left(\frac{Y_T - Y_i}{d(P_i, P_T)} / \frac{X_T - X_i}{d(P_i, P_T)}\right) \qquad (7)$$

Therefore, the new gradient vector is as in equation (8).

$$\nabla Uatt(P_i) = cos(\theta + Ch_{i,j}) * i + sin(\theta + Ch_{i,j}) * j \qquad (8)$$

Where, $Ch_{i,j}$ is the gene j of the chromosome i.

The applied gene of a certain chromosome in equation (8) depends on which region the point (P_i) falls. For example, we apply gene 1 if point (P_i) in the region between FO1 and start point, then we apply gene 2 if point (P_i) in the region between FO1 and FO2 and so on.

Second:

Calculate the gradient of repulsive force of the obstacle that effects on point (P_i). The point (P_i) only affected by repulsive force of the nearest obstacle. The repulsive force is considered as the inverse of Euclidean distance from current point of the path to the nearest point of the nearest obstacle. Equations from (9) to (12) find the nearest point of the obstacle.

$$\emptyset = tan^{-1}(Y_O - Y_i / X_O - X_i) \tag{9}$$

Where, Y_O and X_O is the center of the nearest obstacle and $\emptyset$ is an angle between the vector $\overrightarrow{P_i P_o}$ and the X-axis.

$$d(P_i, P_n) = d(P_i, P_o) - R_o \tag{10}$$

Where, $d(P_i, P_n)$ is the distance between the point (P_i) and the nearest point of the obstacle(P_n), $d(P_i, P_o)$ is the distance between (P_i) and (P_o) the center of the nearest obstacle, and R_o is a radius of nearest obstacle.

$$X_n = X_{i+} d(P_i, P_n) * cos\emptyset \tag{11}$$

$$Y_n = Y_{i+} d(P_i, P_n) * sin\emptyset \tag{12}$$

In equation (11) and (12) , X_n and Y_n is the nearest point of obstacle to current point of path (P_i). Figure (7) illustrates equations (9) to (12). The magnitude of repulsive force is shown in equation (13).

$$Urep(Pc) = \varepsilon \frac{1}{d(P_i, P_n)} \tag{13}$$

Where, ε is a small factor between 0.001 and 0.1. ε is used to reduce the effect of the repulsive force, so that the path can pass near border of obstacle.

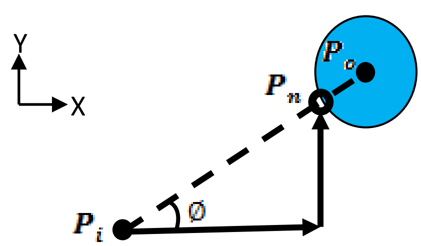

Figure (7) The nearest point of obstacle to current point of the path.

The gradient of the repulsive force is shown in equation (14).

$$\nabla Urep(P_i) = -\frac{\varepsilon}{d^3(P_i, P_i)}((X_n - X_i) * i + (Y_n - Y_i) * j \tag{14}$$

Third:

Calculate the next point (P_{i+1}). The total effective force on (P_i) is shown in equation (15).

$$\dot{P}_i = -(\nabla Urep + \nabla Uatt) \tag{15}$$

The next point (P_{i+1}) is found according to Euler's method with a variable step size as in equation (16).

$$P_{i+1} = P_i + H * -(\nabla Urep + \nabla Uatt) \tag{16}$$

Where, H is the step size. In order to reduce the computations time, we modified H to be as in equation (17).

$$H = \partial * d(P_i, P_n) \tag{17}$$

The first, second and third calculations begin at the start position and continue iterations until the whole path is formed. The iteration will stop if (P_{i+1}) is the target point (valid path) or the maximum number of iterations is executed (invalid path).

b. Fitness Function

The objective of our problem is to find a best feasible path with minimum length. Thus, we proposed a fitness function as an inverse of total length for the valid paths. However, for the invalid paths, we proposed the inverse of sum of the total length and the distance between last point of the path and the target point as a fitness as shown in equation (18).

$$F = \begin{cases} \dfrac{1}{\sum_{i=0}^{n} d(P_i, P_{i+1})} & P_n = P_T \\ \dfrac{1}{\sum_{i=0}^{n} d(P_i, P_{i+1}) + \mu * d(P_n, P_T)} & P_n \neq P_T \end{cases} \tag{18}$$

Where, μ is a constant value, $d(P_i, P_{i+1})$ is the Euclidean distance between (P_i) and (P_{i+1}), as shown in equation (19).

$$\boldsymbol{d(P_i, P_{i+1}) = \sqrt{(X_{i+1} - X_i)^2 + (Y_{i+1} - Y_i)^2}} \tag{19}$$

c. Sort Operation

At each generation, the proposed GA discards percent part of population (chromosomes) which has the lowest fitness. In order to facilitate this operation and separate easily the high fit chromosomes from others, the chromosomes are organized in descending order according to their fitness. Thus, the chromosome with the lowest order has the lowest fitness. The discarded operation of chromosomes through adaptation in each generation are started from down to up.

d. Initial Population

In our proposed GA, each chromosome in the initial population is generated randomly. Each chromosome consists of a number of genes, which is equal to number of frontal obstacles. The value of each gene represents radian angle, and initially filled by a value in range $-\pi/2$ to $\pi/2$.

e. Genetic Operations

During adaptation of the population via generations, a certain percentage of resultant population has very low fitness values. These chromosomes with lost fitness are discarded in each generation. The spaces of the discarded chromosomes are filled by new offspring developed by crossover and mutation operations.

The parents for the crossover operation are selected from the chromosomes of current population. In proposed GA, tournament selection method is used to select parents from the current population. Where, three chromosomes are selected randomly with uniform probability, and then chromosome with highest fitness is chosen as a parent. The other parents are selected in the same way.

The two parents are combined via crossover operation. Crossover operation swaps the two parents about a randomly selected crossover point to produce two new offspring chromosomes. The parent for mutation operation is selected in the same method. The gene of parental chromosome is mutated by replace the value of one gene by random value in range $-\pi/2$ to $\pi/2$.

The crossover and mutation operations are continuously executed until all the spaces of the discarded chromosomes are filled by new offspring chromosomes.

Finally, since the new population in each generation is a mix of new and old populations, the proposed GA belongs to a type of steady state GA.

VI. SIMULATION and RESULTS

The proposed algorithm is applied on computer of 2.1 GHz Core 2 Due CPU and the software written in C++ programming language. Parallel programming technique is used to calculate the fitness of the whole population in order to reduce the required computation time. Three different known static environments with many obstacles and different complexity are taken in our study as illustrated below. The first environment has 50 obstacles, the second has 150 obstacles, and the third environment has 5 obstacles. Some optimal paths with different start points and target points are searched for each environment. The setting parameters of the proposed GA are, population size is 50, the crossover rate is 0.8 and the mutation rate is 0.05. The results are shown in Table (1) and the graphs of these paths are shown in Figures (8-14).

The maximum error percentage in Table (1) is the percentage of the difference between longest and shortest paths over the shortest length of the path and it is calculated according to equation (20).

$$\textbf{Max Error \%}=\frac{\textit{Longest path}-\textit{Shortest path}}{\textit{Shortest path}}*\textbf{100\%} \qquad (20)$$

Max generation and min generation are the maximum and minimum numbers of generations of all runs for each path.

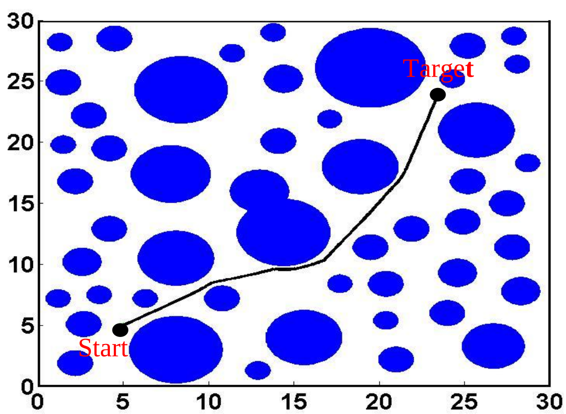

Figure (8) First Path of first environment.

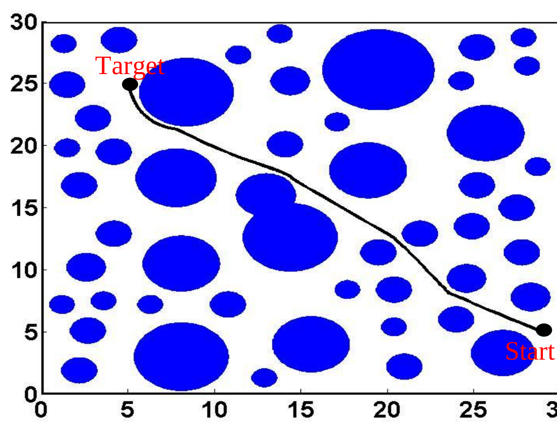

Figure (9) Second path of first environment.

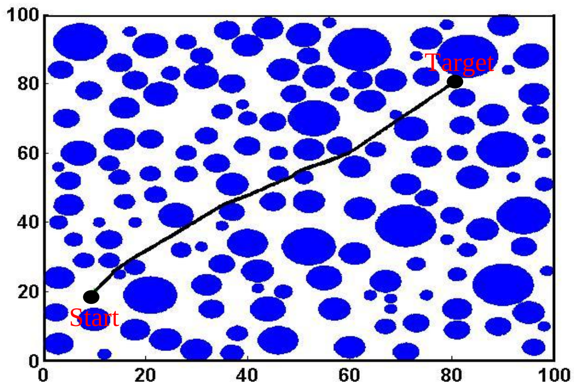

Figure (10) First path of second environment.

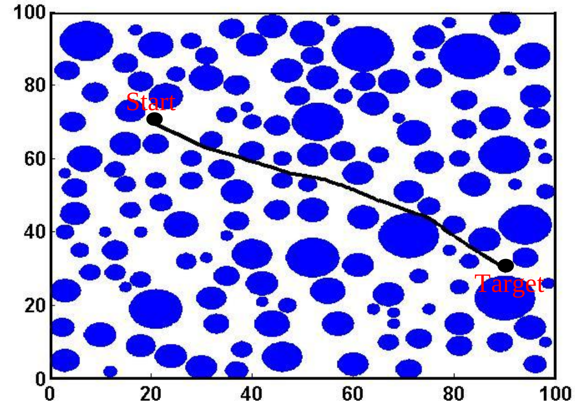

Figure (11) Second path of second environment.

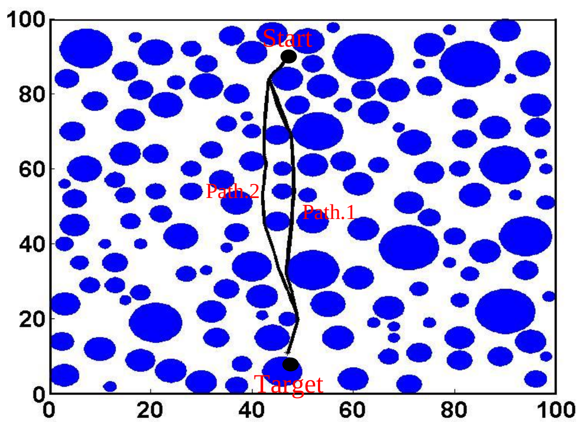

Figure (12) Third path of second environment.

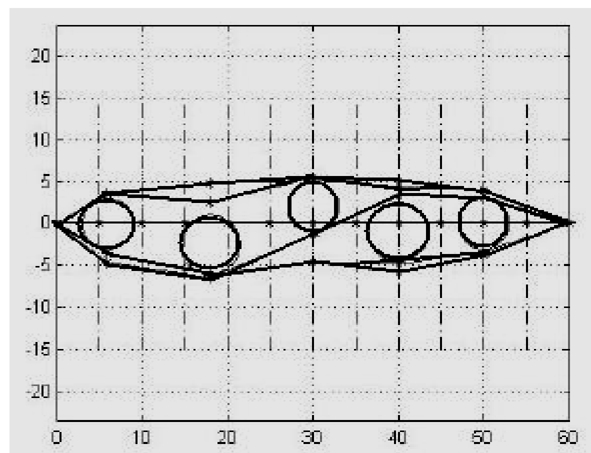

Figure (13) the results of improved PSO for 20×60 environment [11].

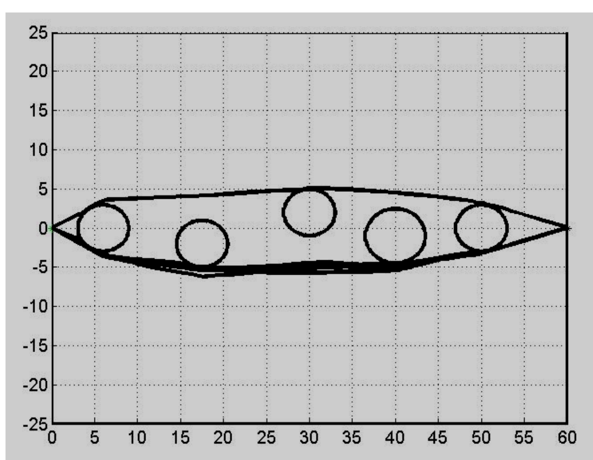

Figure (14) the result of the proposed algorithm in this paper for 20×60 environment [11].

Table (1) The results of the three Examples.

Map	Path Number	No. of Runs	Start Point	Target Point	Max Generations	Min Generations	Average Elapsed Time for runs in sec	Max Error Percentage	Shortest path length
	First Path	5	(5,5)	(26,25)	267	129	18.157	0.6%	30.462
	Second Path	5	(26,6)	(5,25)	187	77	9.722	0.006%	32.07
Second Map	First Path	5	(10,20)	(80,80)	343	126	78.334	0.233%	93.366557
	Second Path	5	(20,70)	(90,30)	341	180	67.2265	0.05%	81.537476
	Third Path	5	(47,90)	(47,11)	304	213	109.453	0.11%	81.2323
Third Map	Path	10	(0,0)	(60,0)	143	78	3.85	0.08%	61.64

Table (2) the comparison between PSO algorithm and our proposed algorithm.

Algorithm	Initial point	Target point	Straight line distance	Average optimal distance
Improved PSO	(0,0)	(60,0)	60	62.503
Proposed Algorithm	(0,0)	(60,0)	60	61.806

In order to prove the efficiency of our algorithm, a comparison is done between our algorithm and the improved particle swarm optimization (PSO) algorithm which is proposed in [11]. The improved PSO and our proposed algorithm are applied on same environment of 20×60. The results of both searching algorithms are shown in Figure (13) and Figure (14) respectively. The comparison between the results of the improved PSO and the proposed algorithm are illustrated in Table (2). The results in this table and the above figures show that the proposed algorithm finds shorter path than the PSO path for same start and target points. Therefore, we can say the proposed algorithm is superior than the improved PSO proposed in [11].

VII. CONCLUSIONS and DISCUSSIONS

In this paper, we proposed a modified hybrid algorithm to find the shortest path between two random points in known static environment. The ability of potential field method to plan a path and the ability of GA to search about optimum parameters of a function are merged in one algorithm to find an optimal path for a mobile robot. The results show that the proposed algorithm can find shortest paths for two crowded environments. In addition, this algorithm succeeded in all runs to reach the target as well as the percent error in the lengths of these paths is so small. Moreover, the computation effort is very low and the algorithm gave the result in short time. The results of the third path of second environment map prove that the proposed algorithm overcomes all drawbacks in the conventional potential field method. Since, the results prove that the proposed algorithm can develop a path which passes between two closely obstacles. In addition, the algorithm does not trap in local minimum points as well as can find redundant paths if they are existent in the environments. The proposed algorithm in this paper is compared with the improved PSO algorithm proposed in [11]. The results show that our algorithm is better because it found shorter path than PSO path. However, the differences between the two algorithms become more effective and obvious when they applied on more complex environment. For more emphasize, this algorithm has been applied to over fifteen environments within large numbers of obstacles and with the different start points and target points. In all those environments, the algorithm found the optimal paths or the paths that have a very small percentage difference in length.

Finally, the proposed hybrid algorithm proves that it is powerful, reliable, efficient, and fast to find optimal path for mobile robot.

References:

[1] Farbod Fahimi, "**Autonomous Robots Modeling, Path Planning, and Control**", Springer, July 2008.

[2] Liao WeiQiang, "**Genetic Algorithm Based Robot Path Planning**", International Conference on Intelligent Computation Technology and Automation, IEEE, 2008.

[3] Li Wei and Zhang Wei, "**Path Planning of UAVs Swarm using Ant Colony System**", Fifth International Conference on Natural Computation, IEEE, 2009.

[4] Wenqiang Guo, Zoe Zhu and Yongyan Hou, "**Graphical Model-Based Recursive Motion Prediction Planning Algorithm In Stochastic Dynamic Environment**", IEEE, 2010.

[5] Santiago Garrido, Luis Moreno, Mohamed Abderrahim and Fernando Martin, "**Path Planning for Mobile Robot Navigation using Voronoi Diagram and Fast Marching**", IEEE, 2006.

[6] O. Khatii, "**Real-time Obstacle Avoidance for Manipulators and Mobile Robots**", hitcmational Jomal of Robotics Research vol. 5, 110, 1986.

[7] M.G. Park and M.C. Lee, "**Experimental Evaluation of Robot Path Planning by Artificial Potential Field Approach with Simulated Annealing**", IEEE, Aug. 5-7-2002.

[8] Hossein Adeli, M.H.N. Tabrizi, Alborz Mazloomian, Ehsan Hajipour and Mehran Jahed," **Path Planning for Mobile Robots using Iterative Artificial Potential Field Method**", International Journal of Computer Science Issues, IEEE, Vol. 8, Issue 4, No 2, July 2011.

[9] Singiresu S. Rao,"**Engineering Optimization Theory and Practice**", Fourth Edition, WILEY, 2009.

[10] Soh Chin Yun, Veleppa Ganapathy and Lim Ooi Chong, "**Improved Genetic Algorithms based Optimum Path Planning for Mobile Robot**", IEEE, 7-10th December 2010.

[11] Yu Fei and Liu Yang-lei, "**The Application of an Improved PSO based on the Quantum Genetic Algorithm in the Submersible Path planning**", IEEE DOI, 2011.