Improved Artificial Potential Field Method for Robotic Path Planning

by

Soham Bhattacharyya

Submitted in partial fulfilment of the requirements
for the degree of Master of Applied Science

at

Dalhousie University
Halifax, Nova Scotia

November 2020

This thesis is dedicated to Ma, Baba & Didibhai for being my everything.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Determination of a collision free path for a robot between start and goal positions through obstacles cluttered in a workspace is central to the design of an autonomous robot path planning. This thesis presents an improved artificial potential field-based navigation algorithm for mobile robots that produce an optimal path for a robot to navigate in an environment. To complete the navigation task, the algorithm will read the map of the environment or workspace and subsequently create a potential map for the robot to traverse in the workspace without colliding with objects and obstacles. This method overcomes the issue of deadlock which was a major bottleneck in the case of artificial potential field method. The simulation results infer the ability of the proposed method to overcome the deadlock issue and navigate successfully from an initial position to the goal without colliding into obstacles in both static or dynamic environment.

LIST OF ABBREVIATIONS AND SYMBOLS USED

1D	One dimensional
2D	Two dimensional
3D	Three dimensional
$a(\boldsymbol{x})$	Vehicle acceleration at the current vehicle position $\boldsymbol{x}$
ACD	Approximate Cell Decomposition
ACO	Ant Colony Optimization
ANN	Artificial Neural Network
APF	Artificial Potential Field
BPF	Bacterial Potential Field
C_{free}	Free space
$C_{obstacle}$	Obstacle space
C_{space}	Configuration space
CD	Cell Decomposition
EA	Evolutionary Algorithm
EAPF	Evolutionary Artificial Potential Field
ECD	Exact Cell Decomposition
$\boldsymbol{F}(\boldsymbol{x})$	Sum of the forces from the various potential fields computed at the current vehicle position $\boldsymbol{x}$
GA	Genetic Algorithm
GPS	Global Positioning System
m	Vehicle mass
PCD	Probabilistic Cell Decomposition
PF	Potential Field

PRM	Probabilistic Road Map
PSO	Particle Swarm Optimization
RRT	Rapidly-exploring Random Trees
RRT*	Rapidly-exploring Random Trees-star
SA	Simulated Annealing
SN	Subgoal Network
VD	Voronoi Diagram
VFF	Virtual Force Field
VFH	Virtual Force Histogram
VG	Visibility Graph
Ω	The workspace
Γ	Boundary
$U_{att}(X)$	The attractive potential at point X
$U_{reps}(X_i)$	Repulsive potential model of the i-th static obstacle
X	The position $[x, y]^T$ of robot's central point in movement space
X_S	The start point
X_g	The target point position $[x_g, y_g]^T$
$\rho(X, X_g)$	The distance between the current location of the central point of mobile vehicle's body and target point
$\rho(X, X_i)$	The shortest distance of between current location of the center of mobile vehicle's body and the i-th obstacle
k	Proportional gain coefficient
ρ_0	The effective effect distance of obstacle
ρ_a	Judgment distance of whether the mobile body reaches to the target point

n	The summation of static obstacles
η	Proportional position gain coefficient
$F(q)$	Force at position q
$U_{add}(X)$	Added potential at position X
s	Proportional coefficient
σ	Potential constant

ACKNOWLEDGEMENTS

I am extremely thankful to my supervisor Professor Dr. Jason J. Gu for providing me with this wonderful opportunity to conduct my graduate studies with his research team. I am grateful for his thorough guidance and feedback, inspiration, and patience all through the dissertation work. I would also like to thank my parents and my sister for their blessings, patience, and support.

Chapter 1

INTRODUCTION

1.1 General

Robotic is now gaining a lot of space in our daily life and in several areas in modern industry automation and cyber-physical applications. This requires embedding intelligence into these robots for ensuring (near)-optimal solutions to task execution. Thus, a lot of research problems that pertain to robotic applications have arisen such as planning of path, motion, and mission, task allocation problems, navigation and tracking. In the present work, a research is carried out on path planning.

Moving from one place to another is a trivial task, for humans. One decides how to move in a split second. For a robot, such an elementary and basic task is a major challenge. In autonomous robotics, path planning is a central problem in robotics. The typical problem is to find a path for a robot, whether it is a vacuum cleaning robot, a robotic arm, or a magically flying object, from a starting position to a goal position safely. The problem consists in finding a path from a start position to a target position. This problem was addressed in multiple ways in the literature depending on the environment model, the type of robots, the nature of the application, etc. Safe and effective mobile robot navigation needs an efficient path planning algorithm since the quality of the generated path affects enormously the robotic application. Typically, the minimization of the travelled distance is the principal objective of the navigation process as it influences the other metrics such as the processing time and the energy consumption. This chapter presents a comprehensive overview on mobile robot global path planning and provides the necessary background on this topic. It describes the various global path planning categories and presents a taxonomy of global path planning problem.

Nowadays, we are at the cusp of a revolution in robotics. A variety of robotic systems have been developed, and they have shown their effectiveness in performing different kinds of tasks including smart home environments, airports, shopping malls, manufacturing laboratories. An intelligence must be embedded into robot to ensure (near)-optimal execution of the task under consideration and efficiently fulfil the mission.

However, embedding intelligence into robotic system imposes the resolution of a huge number of research problems such as navigation which is one of the fundamental problems of mobile robotics systems. To successfully finish the navigation task, a robot must know its position relatively to the position of its goal. Moreover, it has to take into consideration the dangers of the surrounding environment and adjust its actions to maximize the chance to reach the destination. Putting it simply, to solve the robot navigation problem, we need to find answers to the three following questions: Where am I? Where am I going? How do I get there? These three questions are answered by the three fundamental navigation functions localization, mapping, and motion planning, respectively.

• **Localization**: It helps the robot to determine its location in the environment. Numerous methods are used for localization such as cameras, GPS in outdoor environments, ultrasound sensors, laser rangefinder. The location can be specified as symbolic reference relative to a local environment (e.g., centre of a room), expressed as topological coordinate (e.g., in Room 23) or expressed in absolute coordinate (e.g., latitude, longitude, altitude).

• **Mapping**: The robot requires a map of its environment in order to identify where he has been moving around so far. The map helps the robot to know the directions and locations. The map can be placed manually into the robot memory (i.e., graph representation, matrix representation) or can be gradually built while the robot discovers the new environment. Mapping is an overlooked topic in robotic navigation.

• **Motion planning or path planning**: To find a path for the mobile robot, the goal position must be known in advance by the robot, which requires an appropriate addressing scheme that the robot can follow. The addressing scheme serves to indicate to the robot where it will go starting from its starting position. For example, a robot may be requested to go to a certain room in an office environment with simply giving the room number as address. In other scenarios, addresses can be given in absolute or relative coordinates.

Planning is one obvious aspect of navigation that answers the question: What is the best way to go there? Indeed, for mobile robotic applications, a robot must be able to reach the goal position while avoiding the scattered obstacles in the environment and reducing the path length. There are various issues need to be considered in the path planning of mobile robots due to various purposes and functions of the virtual robot itself. Most of the proposed approaches are focusing on finding the shortest path from the initial position to goal position. Recently, research is focusing on reducing the computational time and enhancing smooth trajectory of the virtual robot. Other ongoing issues include navigating the autonomous robots in complex environments. Some researchers consider movable obstacles and navigation of the multi-agent robot. Whatever the issue considered in the path planning problem, three major concerns should be considered: efficiency, accuracy, and safety. Any robot should find its path in a short amount of time and while consuming the minimum amount of energy. Besides that, a robot should avoid safely the obstacles that exist in the environment. It must also follow its optimal and obstacle-free route accurately. Planning a path in large-scale environments is more challenging as the problem becomes more complex and time-consuming which is not convenient for robotic applications in which real-time aspect is needed. In this research work, the focus is on finding the best approach to solve the path planning for finding optimal path in a minimum amount of time. It is also considered that the robot operates in a complex large environment containing several obstacles having arbitrary shape and positions.

Chapter 2

LITERATURE REVIEW

2.1 General

The mobile robot path planning is the task to find a collision-free path, through an environment with obstacles, from a specified start location to a desired goal location. This chapter classifies the various path planning approaches in different ways and gives some general information about traditional path planning methods in different environments such as the Visibility Graph method, Cell Decomposition method, and Artificial Potential Field method by literature survey. Further, the drawbacks related to the potential field method are also discussed.

There are three useful terms that are commonly used in graph based path planning techniques; configuration space (C_{space}), obstacle space ($C_{obstacle}$), free space (C_{free}) and free path.

Configuration space **(C_{space})** concept is basically used in robot path planning in an environment including stationary known obstacles. Configuration space is a transformation of physical space where the robot and obstacle have the real size into another space where the robot is treated as a particle. This is achieved by shrinking the size of the robot to a point while expanding the obstacles by the size of the robot.

Free space **(C_{free})** is defined simply as the consist of areas which are not occupied by the obstacles of the configuration space.

Obstacle space **($C_{obstacle}$)** and free space are the two sub spaces in the C_{space}. $C_{obstacle}$ which are defined as a set of infeasible configurations that represents the obstacles existing in the C_{space}.

Free path is the path between the starting point and the goal point which lies completely in the free space and does not come into contact with any obstacle in the environment. [29]

2.2 Path Planning Approaches

The origin of robot motion planning can be tracked to the middle of the 1960s [12]. It has been dominated by classical approaches such as the Roadmap, Cell Decomposition, Mathematical Programming and Artificial Potential Field. Representative proposals of Roadmaps approaches are the Visibility graph which is a collection of lines in the free space that connects the trait of an object to another. The Voronoi diagram of a collection of geometric objects is a partition of space into cells, each of which consists of the points closer to one particular object than any other [13]. The Silhouette approach consists of generating the silhouette of the work cell and developing the Roadmap by connecting these silhouettes curves to each other [14]. The idea of Cell Decomposition algorithms is to decompose the C-space into a set of simple cells, and then compute the adjacency among cells. In the Mathematical Programming approach, the requirement of obstacle avoidance is represented by a set of inequalities on the configuration parameters; the idea is to minimize certain scalar quantities to find the optimal curve between the start and goal position [15].

In [16], 1381 papers dating from 1973 to 2007 were surveyed, which covered a sufficient depth of works in the robot motion planning field. In this work, a broad classification of heuristic techniques is presented, which facilitates its analysis and method's expectations. Broadly, the given classification is as follows: Probabilistic, heuristic and meta-heuristic approaches [17]. In the former are the Probabilistic Roadmaps, Rapidly-exploring Random Trees, Level set and Linguistic Geometry. In the heuristic and meta-heuristic approaches are the Neural Networks, Genetic Algorithms (GAs) [18;19;20], Simulated Annealing [21], Ant Colony Optimization [22], Particle Swarm Optimization, Stigmergy, Wavelets, Tabu Search and Fuzzy Logic. All the mentioned methods have their own strengths and drawbacks; they are deeply connected to one another, and in many applications, some of them were combined together to derive the desired robotic controller in the most effective and efficient manner.

These path planning approaches can be categorized into two categories based on the aspect of completeness. From the completeness point of view the approaches can be categorized as classical and heuristic methods. Classical methods aim to find an optimal path if exists or proves that there is no solution. Heuristic methods try to find a better solution (path) in a short time but do not guarantee to find a solution always. Some example approaches for the above classification can be given as bellow.

1. Classical approaches
   - Roadmap
   - Cell Decomposition (CD)
   - Artificial Potential Field (APF)
   - Mathematical Programming
   - Virtual Force Histogram (VFH)
   - Virtual Force Field (VFF)
   - Subgoal Network (SN)
2. Heuristic-based approaches
   - Neural Network (NN)
   - Genetic Algorithm (GA)
   - Particle Swarm Optimization (PSO)
   - Ant Colony Optimization (ACO)
   - Simulated Annealing (SA)

2.2.1 Roadmap Methods

The roadmap approach, also known as the *Retraction, Skeleton, Highway* or the *Freeway* approach, is one of the earliest path planning methods that have been widely employed to solve the shortest path problem. This approach is dependent upon the concept of configuration space (C_{space}) and continuous path. In this approach, the Cspace is used and the key feature of this approach is the construction of a roadmap or a freeway.

The roadmap method is based on capturing the connectivity if the robot's free space in the form of a network of 1D curves (straight lines). This set line straight lines which connect two nodes of different polygonal obstacles lie in the free space C_{free} represent the roadmap. All the segments that connect a vertex of one obstacle to a vertex of another without entering the interior of any of the polygonal obstacles are drawn.

If a continuous path can be found in the free space of the roadmap, the initial and goal points are then connected to this path to arrive at the final solution, a free path. If more than one continuous path is found and the number of nodes in the graph is relatively small, Dijkstra's shortest path algorithm is often used to find the best path.

Various types of roadmap approaches including the visibility graph, voronoi diagram, the freeway net and silhouette have become more popular in robot path planning in known environments than the others.

One of the oldest path planning methods is the visibility graph method which was introduced by N.J. Nilsson early in 1969. The visibility graph is the collection of lines in the free space that connects a feature of an object to that of another. In its principal form, these features are vertices of polygonal obstacles. A basic example construction of visibility graph is given in Fig. 2.1(a). One of the disadvantages of visibility graph is that the resultant shortest paths touch the obstacles at the vertices or even edges and thus it is not safe. Even though such short comings are existing in VG method, it is still useful in the environments in which the obstacles can be represented as polygonal shapes.

The Voronoi diagram is defined as the set of points that are equidistant from two or more object features. It is a collection of regions that divides the plane. When the edges of the convex obstacles are taken as features, the VD of the C_{free} consists of a finite collection of straight line segments and parabolic curve segments. It is necessary to mention that the use of VD is highly dependent on the sensory range and its accuracy, because this method maximizes the distance between the obstacles and the robot. This has a capability of addressing the drawbacks of the VG. A simple VD example is given in Fig. 2.1(b).

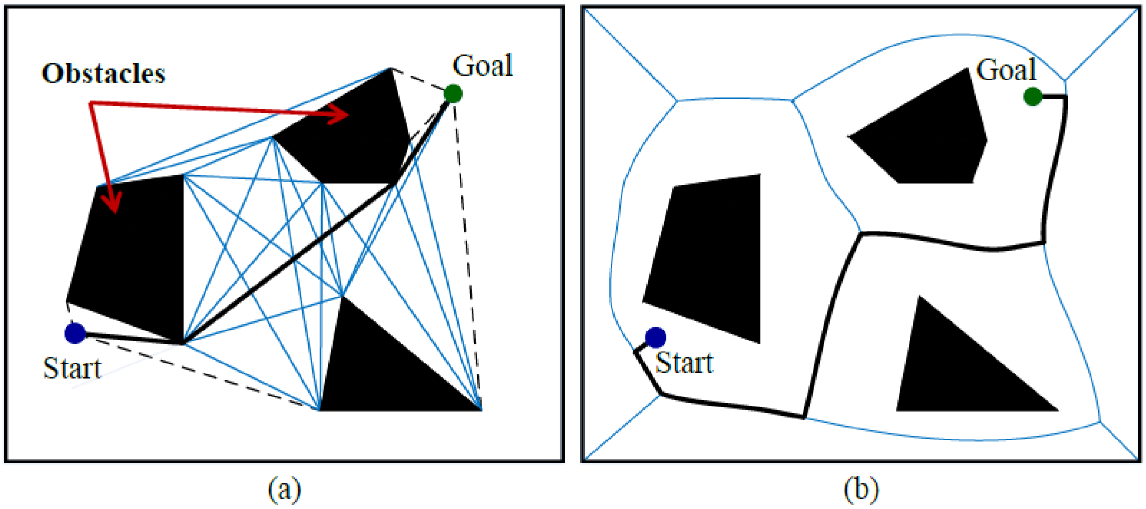


Figure 2.1 Roadmap based path planning: (a) Visibility Graph, (b) Voronoi Diagram [29]

The roadmap is classified as a complete approach, (i.e. it finds a free path, if one exists.) however, other non-complete (probabilistic) variations exist for constructing and searching the roadmap. Probabilistic roadmaps in general, improve the speed of the algorithm. However, the principle disadvantages of the roadmap approaches are:

(i) The roadmap goal is to find a free path (not an optimal path or near-optimal)

(ii) It is complex and not suitable for dynamic environments due to the need for reconstructing the roadmap whenever a change occurs. [29]

2.2.2 Cell Decomposition (CD) Methods

Cell decomposition method is highly used in literature in path planning problems. The basic idea behind this approach is to find a path between the initial point and the goal point that can be determined by subdividing the free space of the robot's configuration into smaller regions called cells. In this representation of the environment it reduces the search space or in other words, the C_{free} is decomposed into cells. After this decomposition process, a search operator is used to find a sequence of collision-free cells from starting point to the goal point. This connectivity graph is generated according to the adjacency relationships between the cells, where the nodes represent the cells in the free space, and the links between the nodes show that the corresponding cells are adjacent to each other. From this connectivity graph, a continuous path or channel can be determined by simply connecting adjacent free cells from the initial point to the goal point.

In the case of a cell is corrupted by containing a part of an obstacle, the corresponding cell is divided into two new cells and then the obstacle-free cell would be added to the collision-free path. Steps to be followed in CD based path planning approach for mobile robot can be described as bellow.

- Divide the search space to connect regions called cells.
- Construct a graph through adjacent cells. In such a graph vertices denote cells and edges connect cells that have common boundary.
- Determine goal and start cells and also provide a sequence of collision-free cells from start to goal cells.
- Provide a path from the obtained cell sequence.

Different CD techniques have been introduced.

- Exact Cell Decomposition
- Approximate Cell Decomposition
- Probabilistic Cell Decomposition

Exact Cell Decomposition

The principle of exact CD approach is to first decompose the free space C_{free} which is bounded both externally and internally by polygons, into a collection of non-overlapping trapezoidal and triangles, called cells. The generated cells are complicated due to their irregular boundaries. This is performed by simply drawing parallel line segments from each vertex of each interior polygon in the configuration space to the exterior boundary as shown in Fig. 2.2. These individual cells are numbered and represented as the nodes in the connectivity graph. The connectivity graph is constructed by searching the adjacency relation among the nodes and linking the configuration space. A path in this graph corresponds to a channel in free space, which is illustrated by the sequence of stripe cells. Hence this channel is then translated into a path in this graph by connecting the centering points of cell boundaries together from the initial configuration to the goal configuration. Such configuration results in providing unnecessary turning points in the point in the path, makes the motion unnatural.

The exact cell decomposition is considered complete, but this accuracy is a more difficult mathematical process for which the computational time is high, especially in crowded environments.

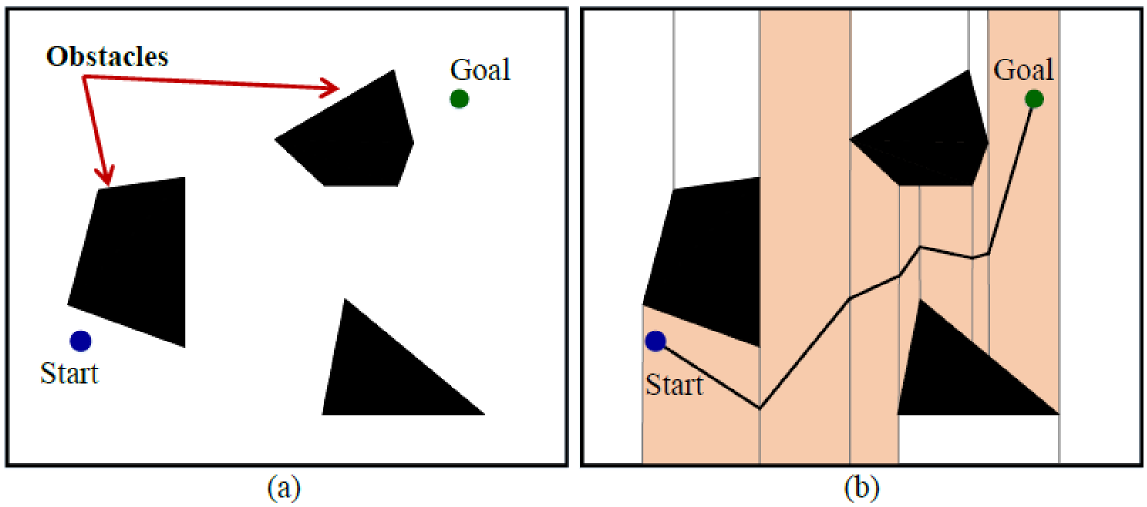


Figure 2.2 Exact Cell Decomposition (a) Configuration space (C_{space}), (b) Path generated by connecting the connectivity graph in the free space (C_{free}) [29]

Approximate Cell Decomposition

This approach of cell decomposition is different from the exact CD because it uses a recursive method to continuous subdividing of the cells until one of the following scenarios occurs.

- Each cell lies either completely in C_{free} space or completely in the $C_{obstacle}$ region
- An arbitrary limit resolution is reached

Approximate cell decomposition method is also referred as "Quadtree" decomposition and it effectively reduces the computational complexity. This method is recursively decomposing the C_{space} into smaller cells by dividing a cell into four smaller identical new cells each time in the decomposition process (see Fig. 2.3). This decomposition continuously subdivides the cells until it fulfills one of the above criteria with an arbitrary resolution limit. After the decomposition process, the free path can be found easily through the initial point to the goal point by following the adjacent, decomposed cells in the C_{free}.

Both the exact and approximate cell decomposition methods have advantages and disadvantages. The decomposition should be guaranteed to be complete, meaning that if a path exists, exact cell decomposition will find the path, however, it is a more difficult mathematical process to get high accuracy. Approximate cell decomposition is not as expensive as exact cell decomposition, and can yield similar or if not exactly as the same results as those of the exact cell decomposition. However, cell decomposition approaches are not suitable for dynamic environments due to the fact that when a priori unknown object appears, a new decomposition must be performed.

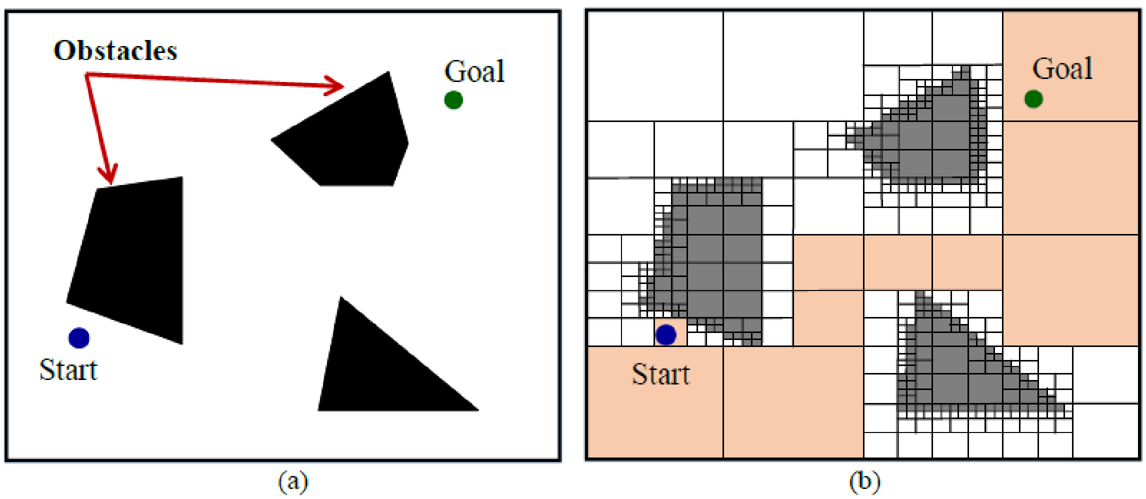


Figure 2.3 Approximate Cell Decomposition (a) Configuration space (C_{space}), (b) Obstacle free path after approximate cell decomposition [29]

Probabilistic Cell Decomposition

Probabilistic cell decomposition (PCD) is similar to approximate CD method except the cell boundaries which do not have any physical meaning. PCD is a probabilistic path planning approach which combines two concepts of approximate cell decomposition and probabilistic sampling methods. PCD resembles an approximate cell decomposition method where the cells have a simple predefined shape. As in approximate cell decomposition methods, PCD divides the configuration space, C_{space} into closed rectangular cells. PCD does not require an explicit representation of the obstacle configuration space, $C_{obstacle}$ but collision avoidance algorithm is able to check the collision free configuration space. Therefore, it does not know whether a cell is entirely free or entirely occupied by the obstacles. A cell is called “possibly free” as long as only collision free sample has been found in the cell. Accordingly, it is called “possibly occupied” if all the samples that have been checked are colliding. If both collision free and colliding samples have been found in the same cell, it is called “mixed” and it has to split up into possibly free and possibly occupied cells. Even though the approximate CD and PCD methods have the advantages of fast implementation, they are not reliable in the environments in which the free space, C_{free} has a small fraction of the environment. [29]

2.2.3 Artificial Potential Field Methods

The potential field method is based on a grid representation by discretizing the space into fine regular grid of the configuration. This method involves modeling the robot as a particle moving under the influence of the artificial potential field that is generated by the obstacles and the goal of the configuration space. The potential field method is based on the idea of attraction/repulsion forces; the attraction force tends to pull the robot towards the goal configuration, whereas the repulsion force pushes the robot away from the obstacles. At each step, the total potential force, generated by the potential function at the robot’s current location, changes the direction and moves the robot incrementally to the next configuration. Thus, the computed information is directly used in the robot’s path planning and no computational power is wasted.

The artificial potential field concept was first introduced by Khatib as a local collision avoidance approach, which is applicable when the robot has no a priori knowledge about the environment, but the robot can sense the surrounding environment during the motion execution. The only drawback of this method is the local minimum problem; since this approach is local rather than a global (i.e. the immediate best course of action is considered). The robot can get stuck at a local minimum of the potential field rather than its global minimum, which is the target destination. This is generally referred as “Deadlock” in robot path planning.

Escaping the local minimum is enabled by constructing potential field function that contains no local minimum or by coupling this method with some other heuristic techniques that can escape the local minimum. The artificial potential field approach can be turned into a systematic motion planning approach by combining it with graph search techniques [29]. We have discussed more about the artificial potential field approach in Chapter 3.

2.2.4 Mathematical Programming

Mathematical programming is another conventional path planning approach. This approach represents requirements by a set of inequalities for obstacle avoidance in the configuration space C_{space}. Path planning problem is formulated as a mathematical optimization problem that finds a solution which defines a curve between the starting point and the goal point by minimizing certain parameter quantities. Since this type of optimization problems are nonlinear and many inequality constraints, numerical methods are used to find a solution. [29]

2.2.5 Genetic Algorithm

Genetic algorithm was introduced by John Holland in 1960s and it mimics the process of biological evolution in order to solve the problems. This technique is successfully applied in the optimization problems such as classical traveling salesman problem, etc. Various studies on GA have been done in path planning problems. GA is one of the widely used algorithms in path planning because of its global optimization ability. Path planning using GA shows good obstacle avoidance capability and path planning in unknown environments but it increases the length of individuals by adopting binary encoding and that causes low efficiency of the occupied memory. [29]

2.2.6 Particle Swarm Optimization (PSO)

Particle swam optimization is a population based algorithm inspired from animals' behaviors that is use to find the global minimum by using particles to get influence from the social and cognitive behaviors of swarm. In the PSO, basic particles are defined based on their position and their velocity in the search space. Particles get attracted towards positions in the search space that represent their best personal finding and the swarm's best finding (local-best and global-best positions). However, the PSO has its own weaknesses in terms of i) controlling parameters ii) premature convergence, and iii) lack of dynamic adjustment which results in the inability to hill-climb solution. In order to overcome these drawbacks associated with the PSO, some modified versions of PSO have been introduced for path planning and mobile robot navigation. But in many studies, PSO has shown the performances better than GA. [29]

2.2.7 Ant Colony Optimization (ACO)

Ant colony optimization method is inspired from ants' social behaviors and imitates the collective behavior of ants foraging from a nest towards a food source in order to find an optimum in the search space. Ants use a chemical substance called pheromone to mark the taken path and this helps them to track the path again. The quality of the path is assessed based on the amount of the pheromones left by the ants that passed from that route using factors such as Concentration and Proportion. Proportion and Concentration of the pheromones indicate the length of the route and the number of ants travelled through that route respectively. Ants chose the routes for travelling with the highest probability of proportion to the concentration of the pheromone. ACO uses a population of randomly initialized ants in the search space that forage towards the goal location to find the optimal path. The optimization of the path is achieved through evaluation of the amount of pheromones deposited by ants on the paths. [29]

2.2.8 Artificial Neural Network (ANN)

Neural Network (NN) is the study of understanding the internal functionality of the brain. NN has been widely used in optimization problems, learning, and pattern recognition problems due to its ability to provide simple and optimal solution. NN is defined as the study of adaptable nodes that would be adjusted to repeatedly solve problems based on stored experimental knowledge gained from process of learning. The use of simple processing which mimics the brains neurons is the fundamental aspect of ANN. Later, these elements connect to each other shaping a network. The overall operational characteristics of the network would be defined based on the potentials and the nature of neurons' interconnection. The NN-based methods can be categorized based on various factors:

- The configuration of their layers: Single Layer, Multi Layers, Competitive Layer
- Their training methodology: Supervised training, Unsupervised training, Fixed

weights (No training), and Self Supervised training

NN-based methods can also be categorized based on their pattern of neurons' interconnection, the methodology they used to determine neurons' connection weights, and also the neurons' activation function. NN in robot navigation has been categorized in the three types i) Interpreting the sensory data, ii) Obstacle avoidance, and iii) Path planning. Hybrid approaches of NN in combination with other artificial intelligent based methods such as Fuzzy logic, knowledge based systems, evolutionary approaches are more appropriate for addressing the robot navigation problem in real-world applications [29].

2.3 Summary on Literature Review

The detailed review on various path planning methods, presented in this chapter, reveals that all the methods, such as Visibility Graph, Voronoi Diagram, Exact Cell Decomposition, Approximate Cell Decomposition, Probabilistic Cell Decomposition, Artificial Potential Field, Mathematical Programming, Genetic Algorithm, Particle Swarm Optimization, Ant Colony Optimization and Artificial Neural Network, have their own advantages and disadvantages. For example, Visibility Graph method [32],[33] is complete and produces optimal length path in both two and three dimensional configuration space for a point robot subject in a static environment. However, this non-optimal method, with heavy computation and high time complexity, produces a path closer to the obstacles. Again, Voronoi Diagram [32],[33] is complete and produces safer path, as they are furthest from the obstacles, in two dimensional or arbitrary configuration space for a point robot subject. However, it is non-optimal, and it requires long range sensor for local path planning. In general, the goal of the roadmap approaches is to find a free path not an optimal or near-optimal path, and it is complex and not suitable for dynamic environments due to the need for reconstructing the roadmap whenever a change occurs. On the other hand, the Exact Cell Decomposition [34] method is complete in a two dimensional configuration space for a point robot. Although, it is non-optimal and requires heavy computation and time. The Approximate Cell Decomposition [34] requires low computation in a two dimensional configuration space for a point robot. However, it is non-optimal and not complete. The heuristic methods [29] take less time, allow parallel search for a point robot. However, they are not complete and not sound. Table 2.1 gives a comparative study on the advantages and disadvantages of the conventional methods.

Table 2.1 Advantage and disadvantages of various methods[29]

Algorithm	Advantage	Disadvantage
APF	Real-time, 2D or 3D, point or rigid robot	Not-complete, non-optimal, local minima
CD	Complete, sound, 2D and 3D, point or rigid robot	Non-optimal, heavy computation, time
ECD	Complete, 2D, point robot	Non-optimal, heavy computation, time
ACD	Low computation, 2D, point robot	Non-optimal, not-complete
VG	Complete, optimal length path, 2D or 3D, point robot, static environment	Non-optimal, heavy computation, time complexity, path closer to obstacles
VD	Complete, safer path, 2D or arbitrary, point robot	Non-optimal, long range sensor for local path planning
Bug	Complete, 2D, point robot	Non-optimal, long path, time complexity
Heuristic	Less time, parallel search, point robot	Not-complete, not sound

Heuristic methods such as A* algorithm has a wide variety of scientific applications. However, it is computationally very expensive in a high dimensional grid, which is a major setback. Genetic Algorithm is one of the widely used algorithms in path planning because of its global optimization ability, good obstacle avoidance capability and good path planning in unknown environment. However, it opts for binary encoding which causes low efficiency of the occupied memory and slowdown. Furthermore, when an unforeseen obstacle blocks a planned path, re-planning is required, and it results a computationally taxing specially in unknown or dynamic environments. Again, the complexity of the environment leads the increase of computational time of global path planning algorithms. Fuzzy Logic method can overcome this pitfall; however, it has the drawback of being not complete. It is evident from the existing literature that there is a lack of a path planning algorithm there that has low computational requirements, applicable in a wide variety of tasks, in two or three dimensional configuration space with no extra requirements or drawbacks.

2.4 Objectives and scope of present study

In this work, an improved artificial potential field method is proposed for path planning for mobile robotics that ensures a feasible and safe path for the robot navigation. This proposal uses concepts from the APF to solve efficiently a robot path planning problem, ensuring a reachable configuration set and controllability if it exists, outperforming current APF approaches. The APF is a reactive motion planning method with inherent well-known difficulties to find global optimal paths, because it cannot solve all local minima problems [23]. Hence, modern methods that overcome these challenges have been developed [24]. In one article, the APF is blended with Evolutionary Algorithms (EA) obtaining a different potential field methodology named Evolutionary Artificial Potential Field (EAPF) [25]. Here, the APF method is combined with GAs to derive optimal potential field functions [24]. Further, the variational planning approach uses the potential as a cost function, and it attempts to find a path to reach the goal point that minimizes this cost [26]. There are also many approaches based on bacterial genetic algorithm as well which produce a range of outcomes [27],[28],[6].

Since the artificial potential field is one of the best methods that can be used for path planning in known or unknown environments as well as in static or dynamic environments, it has been chosen as the basic technique for present investigation. Even though it is simple in analysis and implementation, it is suffering from local minima problem which causes the deadlock of the robot. Number of research papers have addressed this problem and provided some solutions but only for special situations.

The objective of this research is to develop an artificial potential field based path planning algorithm for solving deadlock problems in structured unknown environments. This algorithm should consider most of the situations where the deadlock can happen. To materialize this approach, a new potential component is introduced which forces the robot to move away from the deadlock positions.

2.5 Organization of thesis

The thesis consists of five chapters. The first chapter of this work thoroughly deal with the autonomous robotics related definitions and applications, and mobile robot path planning and localization problem. In Chapter 2, we provide the necessary background to define the path planning problem, path planning algorithms and classification. Existing problems of the conventional approaches have been deeply reviewed in this chapter. Chapter 3 contains mathematical overview of the conventional Artificial Potential Field method. A detailed discussion of the proposed Artificial Potential Field based algorithm is also given. Chapter 4 contains the simulation study along with the performance analysis of the proposed method and results for the static environments is presented. The robustness of the proposed method is also examined. Chapter 5 contains a comparative study between the proposed improved Artificial Potential Field algorithm and a set of conventional approaches. Finally, chapter 6 provides conclusions and suggestions for future work.

Chapter 3

IMPROVED ARTIFICIAL POTENTIAL FIELD

3.1 General

The potential field approach [11] is especially popular in mobile robotics as it seems to emulate the reflex action of a living organism. A fictitious attractive potential field is considered to be centred at the goal position [Fig. 3.1(a)]. Repulsive fields are selected

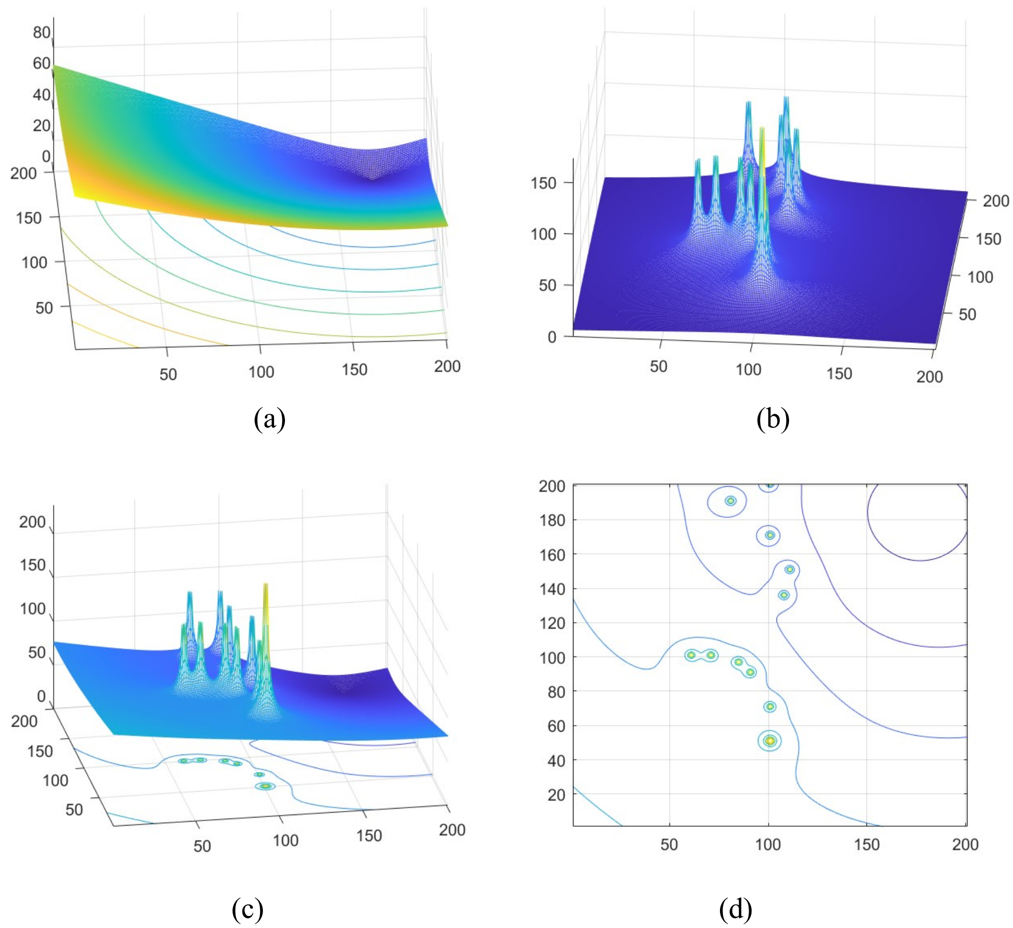


Figure 3.1 Potential field approach to navigation: (a) attractive field for goal at lower right corner, (b) repulsive fields for obstacles, (c) sum of the attractive and repulsivepotential fields, (d) contour plot showing motion trajectory

to surround the obstacles [Fig. 3.1(b)]. The sum of the potential fields [Fig. 3.1(c)] produces the robot motion. Using $\boldsymbol{F}(\boldsymbol{x}) = ma$, with m the vehicle mass and $\boldsymbol{F}(\boldsymbol{x})$ equal to the sum of the forces from the various potential fields computed at the current vehicle position $\boldsymbol{x}$, the required vehicle acceleration $a(\boldsymbol{x})$ is computed. The resulting motion avoids obstacles and converges to the goal position. This approach does not produce a global path planned a priori. Instead, it is a real-time on-line motion control technique that can deal with moving obstacles, particularly if combined with manoeuvring board techniques. Various methods have been proposed for selecting the potential fields; they should be limited to finite influence distances, or else the computation of the total force $\boldsymbol{F}(\boldsymbol{x})$ requires knowledge of all obstacle relative positions.

The potential field approach is particularly convenient as the force $\boldsymbol{F}$ may be computed knowing only the *relative positions* of the goal and obstacles from the vehicle; this information is directly provided by onboard sonar and laser readings. The complete potential field does not need to be computed, only the force vector of each field acting on the vehicle. A problem with the potential field approach is that the vehicle may become trapped in *local minima* (e.g., an obstacle is directly between the vehicle and the goal); this thesis proposes a novel way to get the vehicle out of these false minima. Potential fields can be selected to achieve specialized behaviours such as *docking* (i.e., attaining a goal position with a prescribed angle of approach) and remaining in the centre of a corridor (simply define repulsive fields from each wall). The sum of all the potential fields yields an emergent behaviour that has not been preprogramed (e.g., seek goal while avoiding obstacle and remaining in the centre of the hallway). This makes the robot exhibit behaviours that could be called intelligent or self-determined.

3.2 Artificial Potential Field

In this chapter the work that has been done before on artificial potential field path planning in the obstacle avoiding scenario is studied. In the artificial potential approach, the obstacle to be avoided are presented by a repulsive artificial potential and the goal is represented by an attractive potential, so that a robot reaches the goal without colliding with obstacles. This approach is computationally much less expensive than the global approach and is therefore suited for real-time implementation. The artificial potential approach, however, has been limited due to the existence of local minima, which can be overcome by using harmonic potential field [1]. A harmonic function should satisfy Laplace's equation, it should not have local extrema in a space free from singularities, it should have second order derivatives. A harmonic function should also satisfy principle of superposition and principle of maxima and minima. These principles indicate that the harmonic function has its extremes only on the boundary, so it does not have local maxima/minima inside the boundary. Hence, it is convenient for us to define boundary conditions for boundary of all obstacles and boundary of goal. Panati *et al.* showed that the Dirichlet boundary condition states that the boundary of all obstacles will be assigned with the maximum value in the region and the boundary of goal position has the minimum value in the region. By defining the boundary conditions in this format the potential field is harmonic field with only global minimum [3].

$$\nabla^2 V(X) = 0\ X \tag{3.1}$$

subject to: $V(X_S) = 1$, $V(X_T) = 0$, and $\frac{\partial V}{\partial n} = 0$ at $X = \Gamma$,

where Ω is the workspace, Γ is its boundary, n is a unit vector normal to Γ, X_S is the start point, and X_T is the target point [4].

According to above improved measures, the improved artificial potential field model is proposed. The attractive force model of the target to a full range of vehicle's body is:

$$U_{att}(X) = 0.5k\rho^2(X, X_g) \tag{3.2}$$

where $\rho(X, X_g)$ is the distance between the current location of the central point of mobile vehicle's body and target point; k is a proportional gain coefficient; X is the position $[x, y]^T$ of robot's central point in movement space; and X_g is the target point position $[x_g, y_g]^T$.

Repulsive potential model of the i-th static obstacles on the full range of movement of the body is:

$$U_{reps}(X_i) = \begin{cases} 0.5\eta \left(\frac{1}{\rho(X,X_i)} - \frac{1}{\rho_0}\right)^2 \|X - X_g\|_2, if\ \rho(X,X_i) \le \rho_0 \\ 0, if\ \rho(X,X_i) > \rho_0 \end{cases} \tag{3.3}$$

where $i \in (1, 2^m, n)$, n is the summation of static obstacles; $\rho(X, X_i)$ is the shortest distance of between current location of the center of mobile vehicle's body and the i-th obstacle; ρ_0 the effective effect distance of obstacle; and η is proportional position gain coefficient. Therefore, the whole potential field becomes,

$$U = U_{att}(X) + \sum_{i=1}^{n} U_{reps}(X_i) \tag{3.4}$$

$$F(q) = -\nabla U(q) \tag{3.5}$$

3.3 Added Potential

The problem that plagues all gradient descent algorithms is the possible existence of local minima in the potential field. Gradient descent algorithm is generically guaranteed to converge to a minimum in the field, but there is no guarantee that this minimum will be the global minimum. This means that there is no guarantee that gradient descent will find a path to the goal X_g. This problem is overcome by adding some potential to the local minima, so that it can differentiate itself from the global minima.

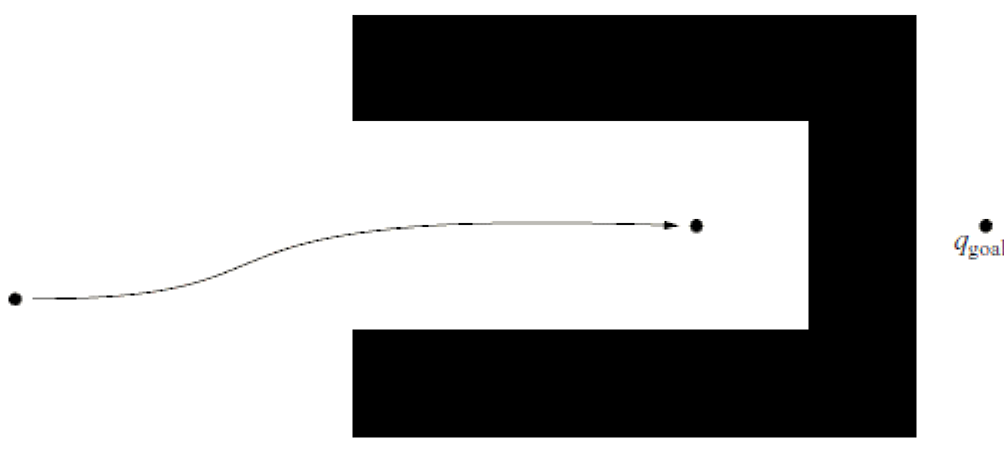


Figure 3.2 Local minimum inside the concavity. The robot moves into the concavity until the repulsive gradient balances out the attractive gradient [1]

When the robot is in local minimum point, the "added potential" is brought to solve the problem of local minimum, the added potential model is:

$$U_{add}(X) = \begin{cases} s\rho^2(X, X_g) + \sigma, \rho(X, X_g) > \rho_a \\ 0, \rho(X, X_g) \leq \rho_a \end{cases} \tag{3.6}$$

where ρ_a is the judgement distance of whether the mobile body reaches to a target point; s is a proportional coefficient and σ is potential constant. Therefore, the whole potential field becomes,

$$\mathrm{U} = U_{att}(\mathrm{X}) + \sum_{i=1}^{n} U_{reps}(\mathrm{X}_i) + U_{add}(\mathrm{X}) \tag{3.7}$$

which is void of any local minima.

3.4 Proposed Improved Artificial Potential Field Algorithm

The proposed potential field method skeleton for robot trajectory planning is described as follows:

1. *Design the attractive PF U^{goal} according to global state with parameter α.*
2. *Design the set of repulsive PF U_i^{obs} according to each obstacle with its parameter β.*
3. *Determine the total PF of the space $U^q = U^{goal} + \sum_{i=1}^{n} U_i^{obs}$.*
4. *Assign the initial state Q_i to the path vector.*
5. *Calculate the distance r_{goal} between Q_i and Q_{goal}.*
6. *while $(r_{goal} > \epsilon\ \&\&\ step < MAXSTEP)$ do*

6.1. *Call the potential gradient descent algorithm to determine the next state.*
6.2. *Add the current state to the path vector.*
6.3. *Add the additional potential to the current coordinate.*
6.4. $step = step + 1.$

end while.

Chapter 4

SIMULATION STUDY

4.1 General

As we discussed in chapter 3, there are some drawbacks associated with the traditional potential field-based path planning algorithm. The traditional APF often suffers from which causes trapping or dead-lock due to local minima and goal non reachability issues. Even though this algorithm has become popular, these fatal problems make some limitations of using it in path planning. Aiming these shortcomings of the traditional potential field based path planning methods, an improved algorithm has been proposed. We propose an APF based path planning method which helps the robot perform a dead-lock free motion. The proposed method is developed and proved for 2D path planning which may be extended straight forward for 3D problems too. In the proposed method, we have introduced an additional field component, which includes the location information to prevent from local minima related issues of the traditional method. As a result, the proposed method will create an improved potential field, which will help the robot to move towards the goal without hitting the obstacles. The proposed method fills the void of incompleteness of APF theoretically. It improves the applicability of the APF and excels in scenarios unconquerable by the APF. The main focus of this chapter is the experimental user study that was conducted to study the performance of the proposed method on several test environments.

4.2 Experimental Setup

The experimental setup of the project includes a working computer that can test the algorithm. The hardware used for simulation is a computer with the processor Intel Core i7-7700HQ, 24 GBs of RAM, NVIDIA GeForce GTX 1050 Ti graphics, and Crucial 1TB SSD. The software used for simulation is MATLAB 2020a.

4.3 Task

To perform the tests our first step is to create a map or environment with obstacles placed in places. Following that, we place the robot in starting coordinate and set its destination to the target coordinate. Finally, we let the robot find its path through the obstacles, with the algorithm.

4.4 Map 1: The map with random obstacles

An environment is generated by placing obstacles in two zones. First, a few obstacles are placed in the direct diagonal path in front of the source point of the robot. As the path planning is simulated, the traditional APF falls in the deadlock. The proposed APF overcome the obstacles and finds a way to the goal. Now to make the environment a bit more complex, a few more obstacles are placed in the path taken by the robot. The proposed algorithm successfully finds a path to the goal. The coordinates of all the obstacles placed in the environment are shown in Table 4.1.

Table 4.1 Obstacle configuration of the map

No. of obstacle	x	y
1	90	90
2	100	70
3	70	100
4	60	100
5	100	50
6	84	96
7	100	170
8	110	150
9	107	135
10	100	200
11	100	50
12	80	190

It can be observed from the Table 4.1 that the obstacles are strategically placed. The obstacles clearly create a challenge to the algorithm, and the algorithm aces it. The traditional APF algorithm fails to find a path as shown in Fig. 4.1. Fig.4.2 shows the path traced by the proposed APF algorithm. Fig. 4.3 shows the potential map after the path is traced by the robot. The algorithm takes 0.66 second to complete.

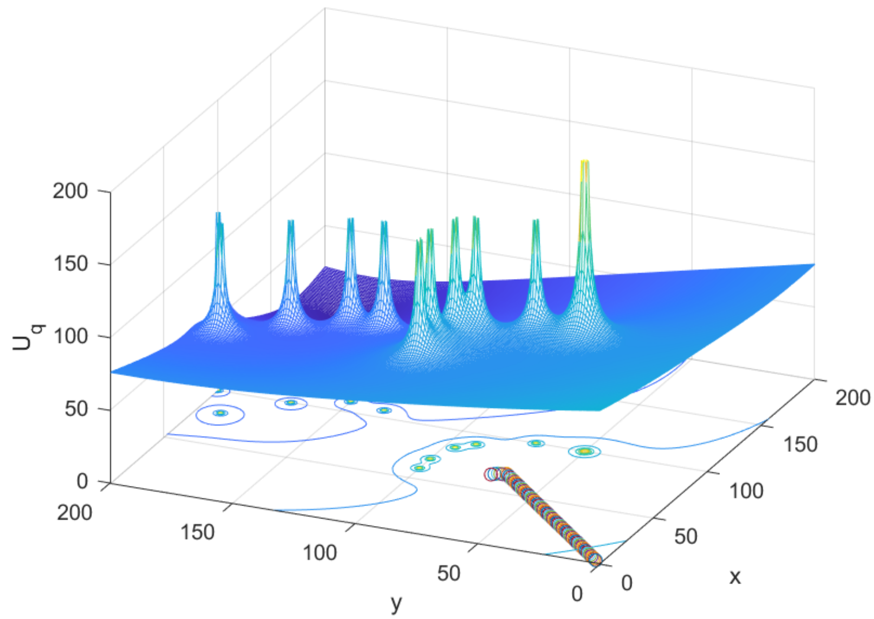


Figure 4.1 The traditional APF algorithm fails to reach the goal

The traditional APF fails to cross the first group of obstacles, and gets stuck in a deadlock. It is clearly a limitation of the traditional APF algorithm.

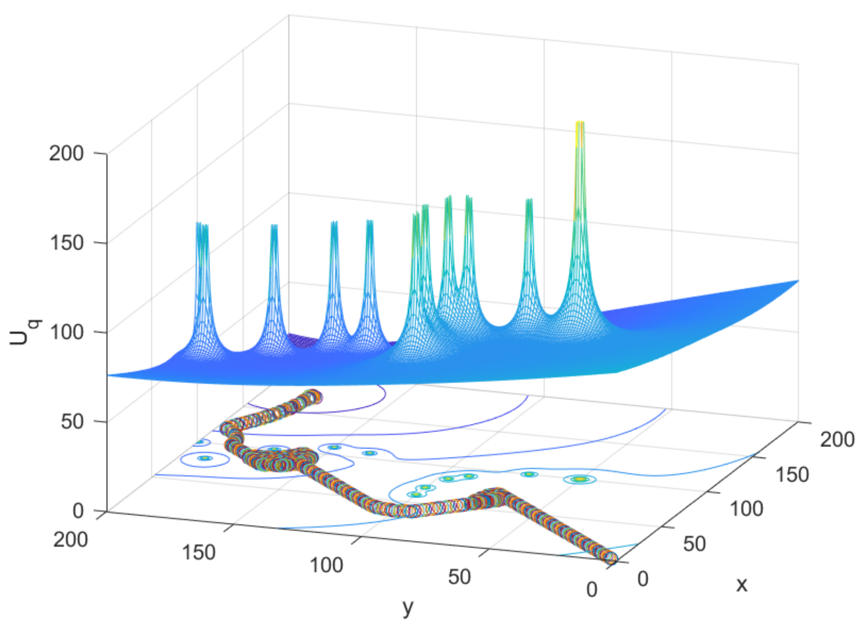


Figure 4.2 The improved APF algorithm reaches the goal

The proposed APF algorithm successfully navigates through both the set of obstacles to reach the goal. Recognizing the hurdles respectively the robot overcomes the set of obstacles in both the cases. It clearly concludes the excellency of the proposed improved APF algorithm.

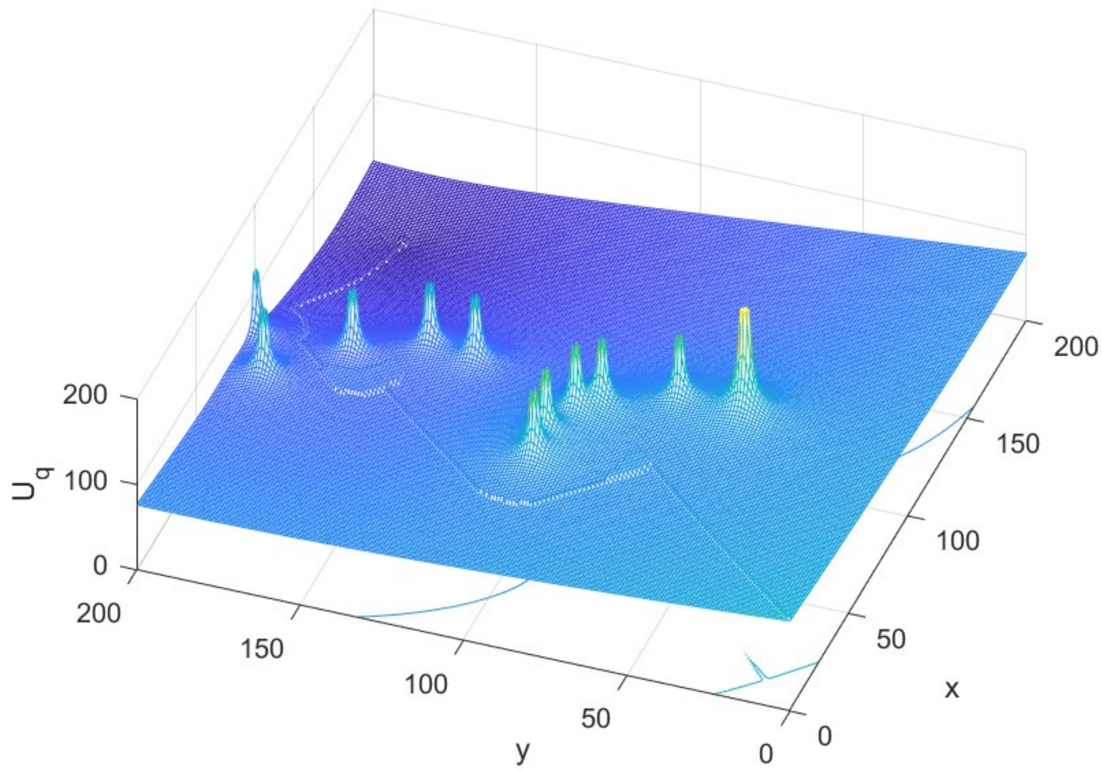


Figure 4.3 The added potentials by the improved APF algorithm

It is observable from Fig. 4.3 that the proposed APF algorithm can overcome obstacles by adjusting the potential field. In both the hurdles, the robot increases the potential and successfully adapts to the environment. This concludes the utility of the proposed algorithm.

4.5 Map 2: The cave

The proposed improved APF algorithm excels in the environment with two set of obstacles. However, it is of much interest to see how it performs in a cave like obstacles in its path. An environment is generated to test the algorithm against a cave like structured obstacle. The coordinates of all the obstacles placed in the environment are shown in Table 4.2.

Table 4.2 Obstacle configuration of the cave map

No. of obstacle	x	y
1	130	150
2	150	130
3	145	145
4	116	144
5	144	116
6	90	126
7	126	90
8	137	105
9	107	135
10	150	140
11	140	150
12	112	75
13	76	112
14	147	123
15	123	147

It can be observed from the Table 4.2 that the obstacles are placed in a cave like structure. The obstacles clearly create a challenge to the algorithm. The traditional APF algorithm falls in a deadlock as shown in Fig. 4.4.

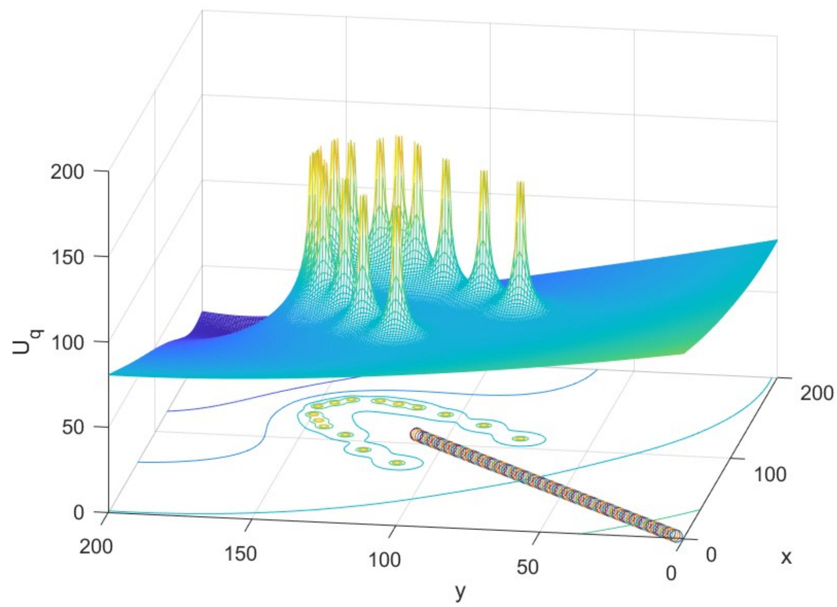


Figure 4.4 The traditional APF algorithm fails to reach the goal

The traditional APF fails to navigate out of the cave shaped obstacles, and gets stuck in a deadlock. It is clearly a limitation of the traditional APF algorithm. Fig. 4.5 shows that the proposed improved APF algorithm can successfully reach the goal. Fig. 4.6 shows the potential field map after the robot completes its journey.

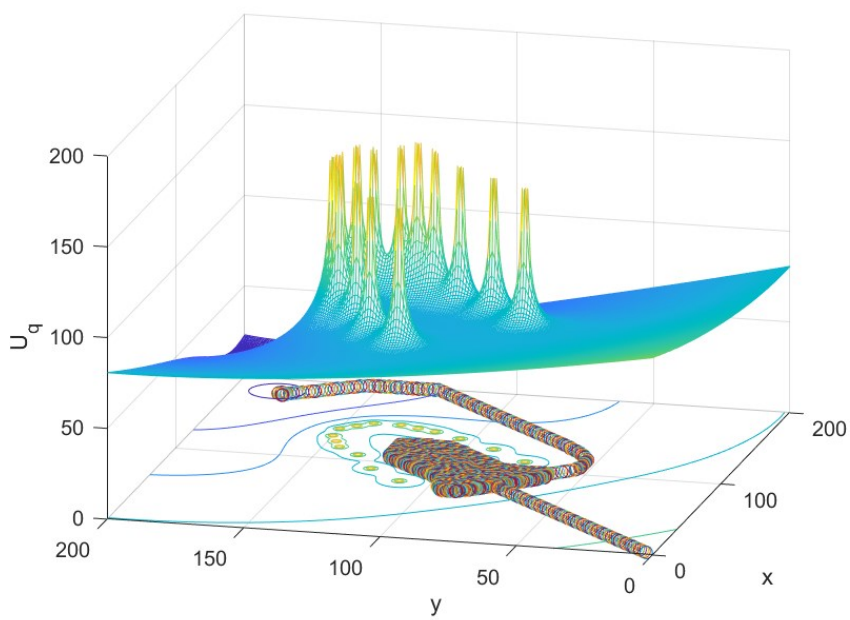


Figure 4.5 The proposed improved APF algorithm reaches the goal

The proposed improved APF algorithm successfully navigates through the cave shaped structure of obstacles to reach the goal. The superiority of the proposed improved APF algorithm hence can be concluded. The algorithm takes 2.11 seconds to run.

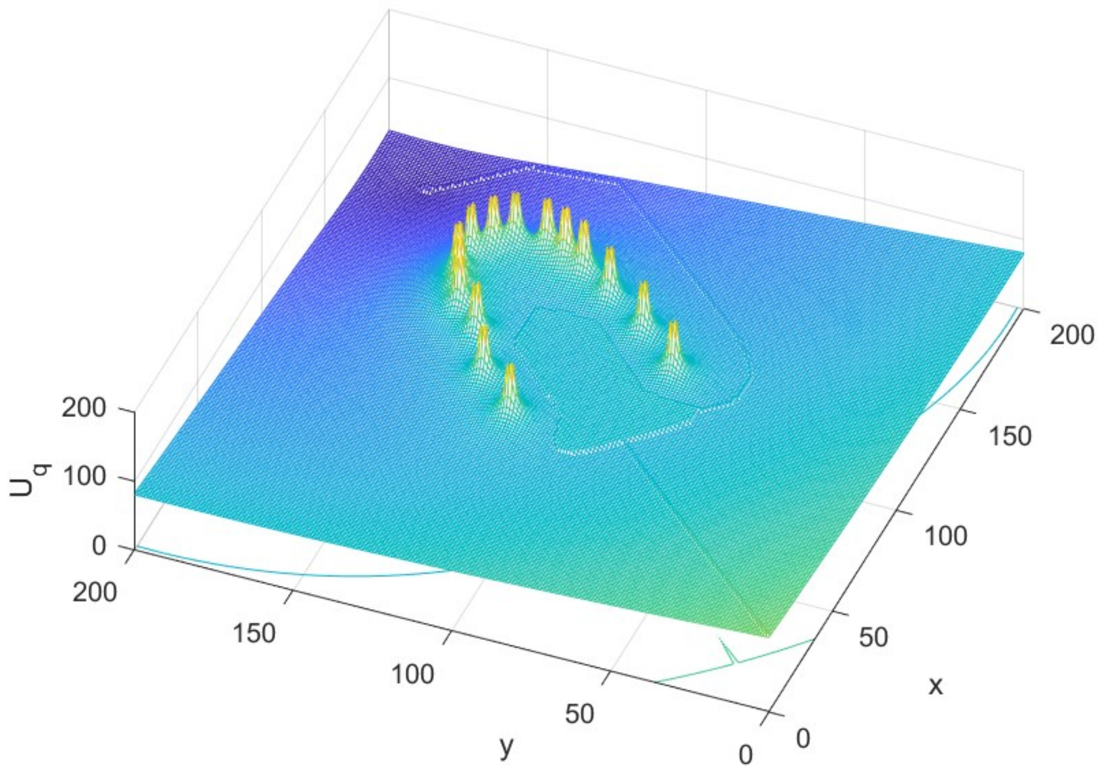


Figure 4.6 The added potentials by the proposed improved APF algorithm

It is observable from the Fig. 4.6 that the proposed improved APF algorithm can modify the potential in such a way that leads to overcoming local minima in an eclectic way. This concludes the superiority of the proposed algorithm.

4.6 Map 3: The wall

The proposed algorithm excels in the environment with random obstacles and the cave shaped obstacles. However, it is of much interest to see how it performs when the goal is separated by a wall. An environment is generated to test the effectiveness of the algorithm in a wall shaped obstacle separated goal structure. The coordinates of all the obstacles placed in the environment are shown in Table 4.3.

Table 4.3 Obstacle configuration of the wall map

No. of obstacle	x	y
1	150	160
2	150	170
3	150	180
4	150	190
5	150	200
6	150	195
7	150	185
8	150	175
9	150	165
10	150	100
11	150	110
12	150	120
13	150	130
14	150	140
15	150	150

It can be observed from the Table 4.3 that the obstacles are placed in a wall shape parallel to the y-axis. The obstacles clearly set a challenge for the algorithm because primarily the robot will try to reach the goal from the other side of the wall. How the traditional APF performs in this scenario is shown in Fig. 4.7.

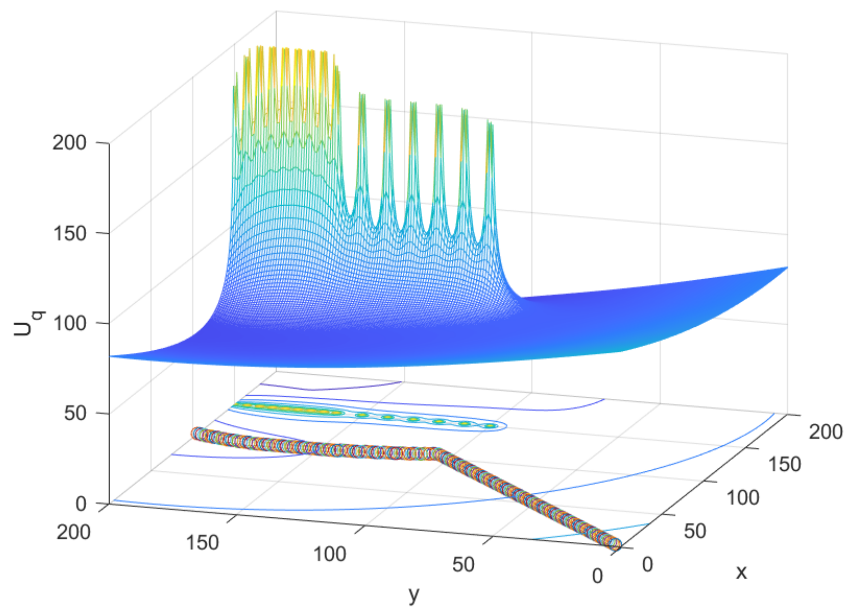


Figure 4.7 The traditional APF algorithm fails to reach the goal

The traditional APF fails to succesfully navigate to the goal, which is on the other side of the wall. The robot gets stuck in the other side of the wall. It is clearly a limitation of the traditional APF. Fig. 4.8 shows how the proposed improved APF algorithm performs in the environment.

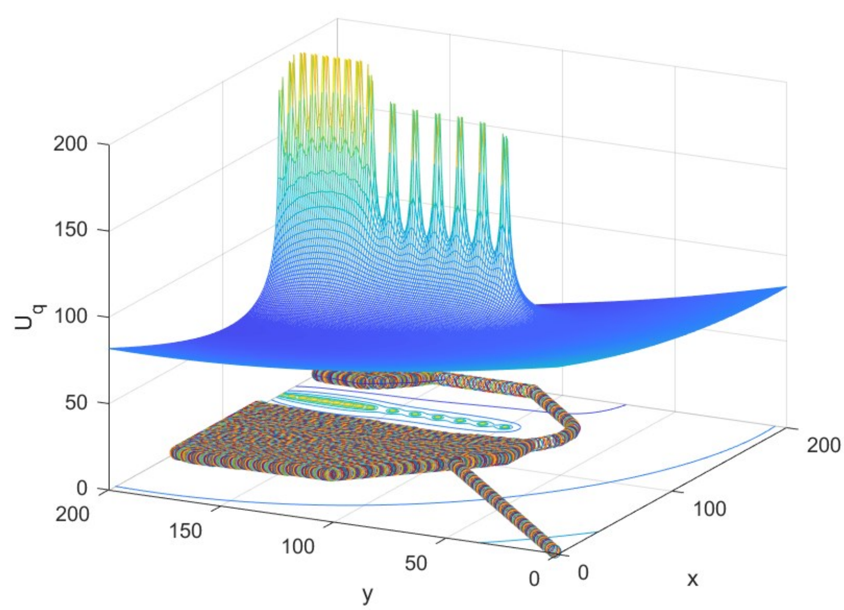


Figure 4.8 The proposed improved APF algorithm reaches the goal

The proposed improved APF algorithm successfully navigate in the environment to reach the goal on the other side of the wall. The algorithm can overcome any amount of hollowness caused by the local minima. The superiority of the proposed algorithm over the traditional algorithm hence can be concluded. The algorithm takes 15.13 seconds to complete. Fig. 4.9 shows the potential field map after the algorithm is run in the environment.

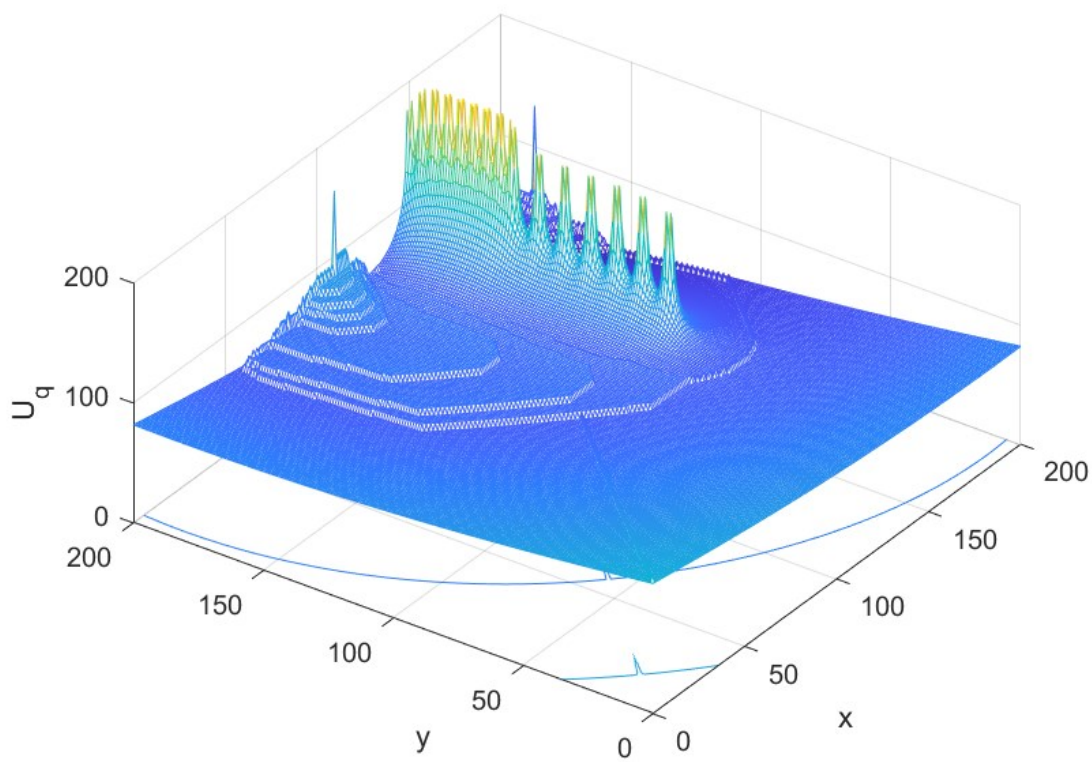


Figure 4.9 The added potentials by the proposed improved APF algorithm

It is observed from the Fig. 4.9 that the proposed improved APF algorithm can modify the potential in an eclectic way such that obstacles of any shape can be overcome. This concludes the superiority of the proposed algorithm.

4.7 Map 4: The bug trap

The proposed algorithm excels in the environments with random obstacles, cave shaped obstacles and wall shaped obstacles. However, it is of much interest to see how it performs when the environment consists of a bug trap. A bug trap is a typical shaped structure that the bug algorithm gets caught in and has to complete the whole outline before it can come out again and complete the path. An environment is generated to test the effectiveness of the algorithm in a bug trap shaped obstacles situation. The coordinates of all the obstacles placed in the environment are shown in Table 4.4.

Table 4.4 Obstacle configuration of the bug trap map

No. of obstacle	**x**	**y**
1	61	108
2	108	61
3	115	55
4	55	115
5	122	57
6	57	122
7	65	128
8	128	65
9	129	72
10	72	129
11	131	81
12	81	131
13	127	88
14	88	127
15	96	127
16	127	96
17	102	126
18	126	102
19	107	124
20	124	107
21	112	122
22	122	112
23	117	117

It can be observed from the Table 4.4 that the obstacles are placed in the shape of a bug trap. The obstacles clearly set a challenge for the algorithm because the robot may tend to go inside the hollow and get stuck. How the traditional APF performs in this scenario is shown in Fig. 4.10.

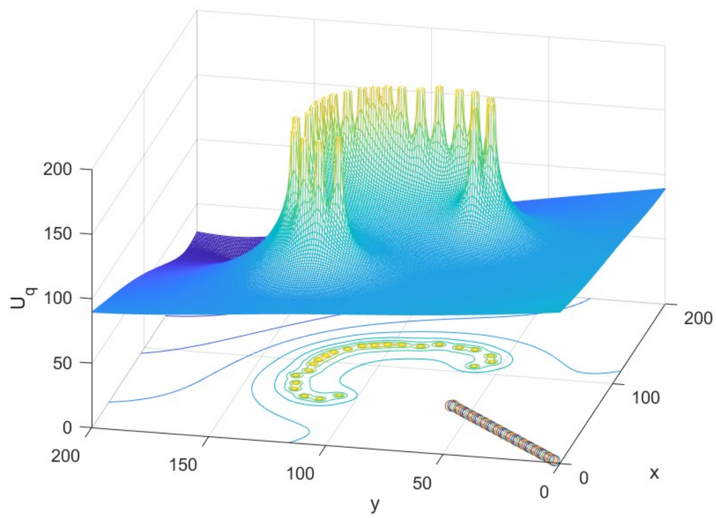


Figure 4.10 The traditional APF algorithm fails to reach the goal

The traditional APF fails to navigate out of the bug trap, and gets stuck in a deadlock. It is clearly a limitation of the traditional APF algorithm. Fig. 4.11 shows how the proposed improved APF algorithm performs in the environment.

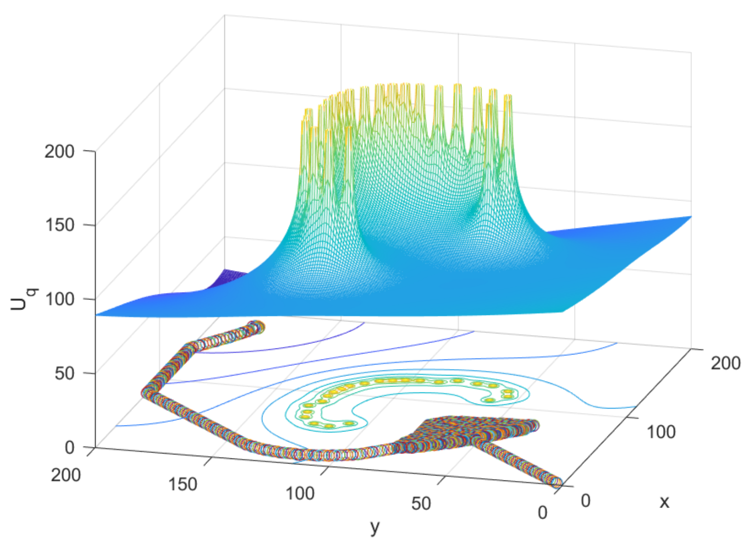


Figure 4.11 The proposed improved APF algorithm reaches the goal

The proposed improved APF algorithm successfully navigates through the bug trap to reach the goal. The algorithm can overcome any type of bug trap problem due to its adaptibility. Hence the superiority of the proposed algorithm over the traditional algorithm can be concluded. The algorithm takes 1.43 seconds to complete. Fig. 4.12 shows the potential field map after the algorithm is run in the environment.

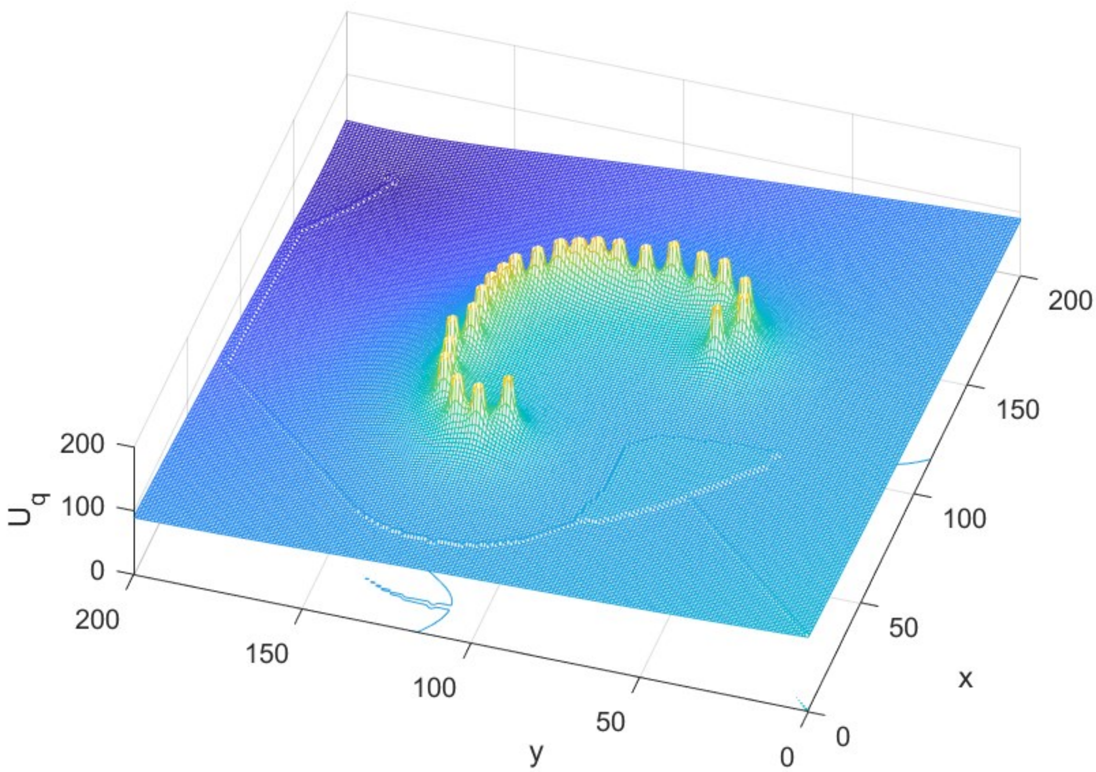


Figure 4.12 The added potential by the proposed improved APF algorithm

It is observed from the Fig. 4.12 that the proposed improved APF algorithm can modify the potential in an eclectic way such that obstacles like a bug trap can be overcome. This concludes the superiority of the proposed algorithm.

4.8 Map 5: The maze

The proposed algorithm excels in several environments namely with random obstacles, cave shaped obstacles, wall shaped obstacles, and bug trap shaped obstacles. However, it is of much interest to see how it performs when the environment consists of a maze. An environment is generated to test the effectiveness of the algorithm in a maze situation.

The coordinates of all the obstacles placed in the environment are shown in Table 4.5.

Table 4.5 Obstacle configuration of the maze map

No. of obstacle	x	y
1	140	200
2	140	190
3	140	180
4	140	170
5	140	160
6	140	150
7	140	140
8	140	130
9	50	200
10	50	190
11	50	180
12	50	170
13	50	160
14	50	150
15	50	140
16	50	130
17	100	80
18	100	90
19	100	100
20	100	110
21	100	120
22	100	60
23	100	70

It can be observed from the Table 4.5 that the obstacles are placed in the shape of a maze with three parallel walls. The obstacles clearly set a challenge for the algorithm as this maze is very tricky for the robot to solve. How the traditional APF performs in this scenario is shown in Fig. 4.13.

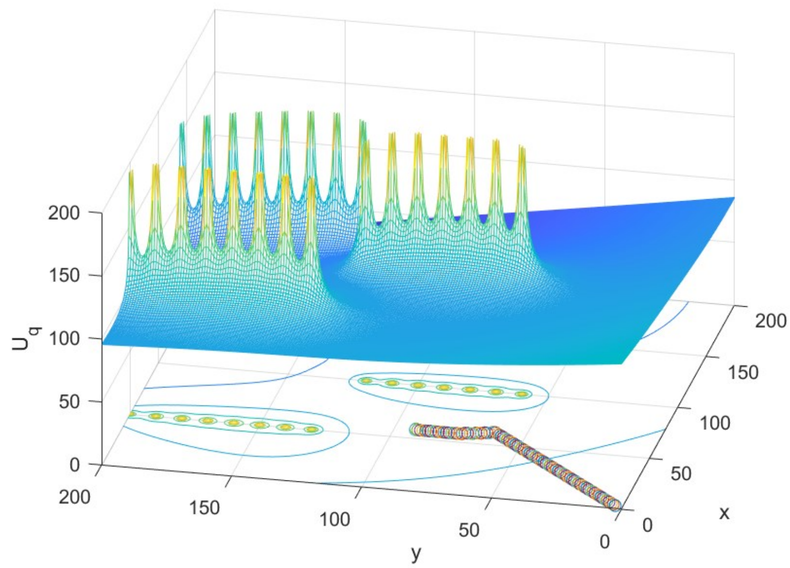


Figure 4.13 The traditional APF algorithm fails to reach the goal

The traditional APF fails to navigate through the maze to reach the goal, and gets stuck in a deadlock. It is clearly a limitation of the traditional APF algorithm. Fig. 4.14 shows how the proposed improved APF algorithm performs in the environment.

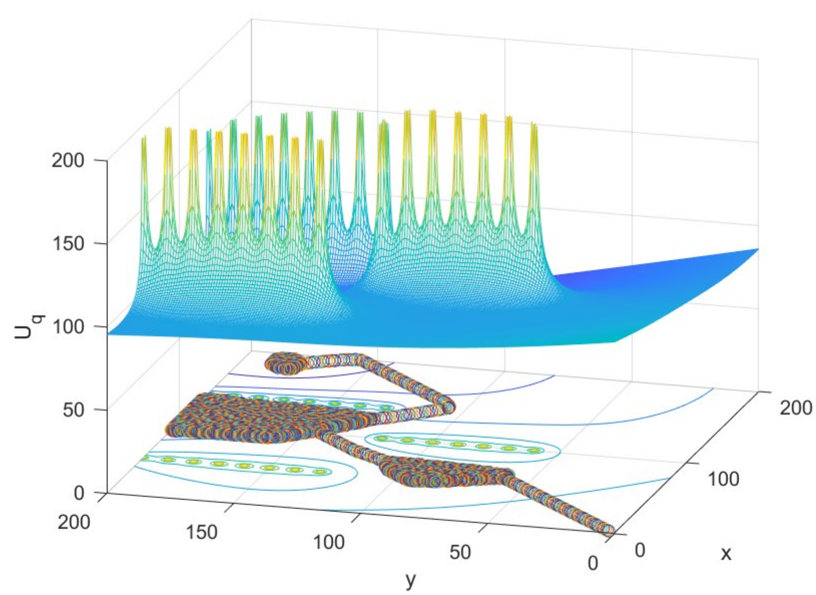


Figure 4.14 The proposed improved APF algorithm reaches the goal

The proposed improved APF algorithm successfully navigates through the maze to reach the goal. The algorithm can solve a large range of maze problems due to its adaptibility. Hence the superiority of the proposed algorithm over the traditional algorithm can be concluded. The algorithm takes 4.93 seconds to complete. Fig. 4.15 shows the potential field map after the algorithm is run in the environment.

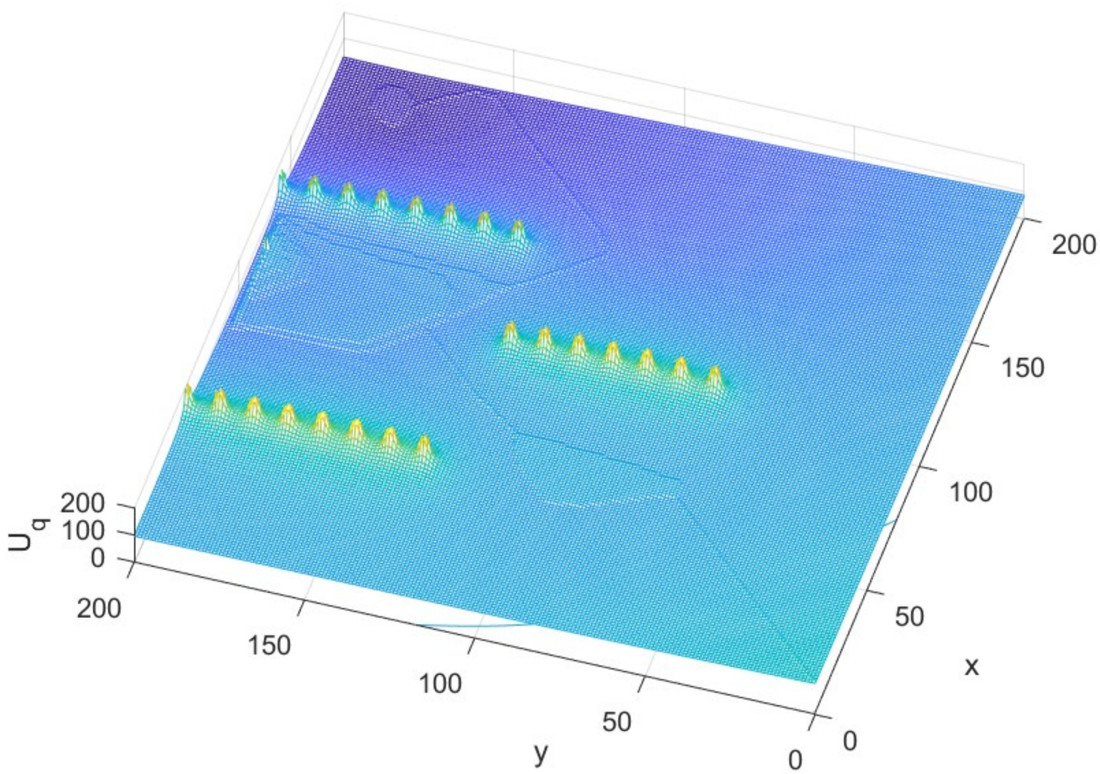


Figure 4.15 The added potentials by the proposed improved APF algorithm

It is observed from Fig. 4.15 that the proposed improved APF algorithm can modify the potential in an eclectic way such that any maze of obstacles can be solved. This concludes the superiority of the proposed algorithm. Table 4.6 shows the time taken by the algorithm to solve the aforementioned environments.

Table 4.6 Time taken by the algorithm in various environments

Configuration Space	**Mean Time** (sec)
Map 1: The map with random obstacles	0.66
Map 2: The cave	2.11
Map 3: The wall	15.13
Map 4: The bug trap	1.43
Map 5: The maze	4.93

It can be noticed that in all the taken complex cases the traditional APF algorithm fails to find a path, and the proposed improved APF finds a path successfully within limited amount of time.

Chapter 5

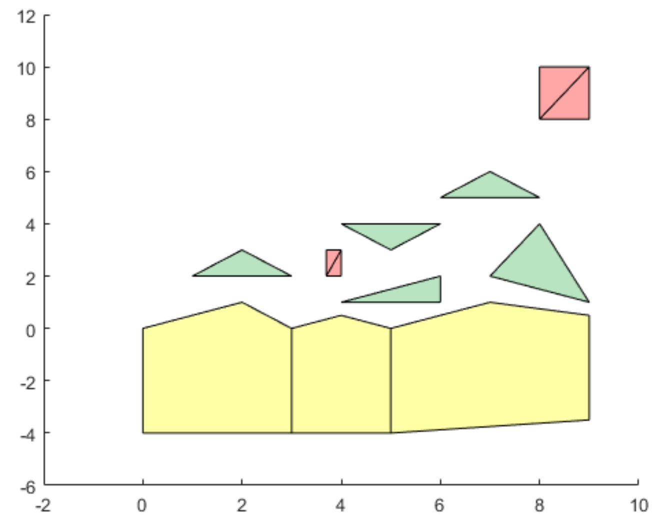
COMPARISON OF THE PROPOSED ALGORITHM WITH OTHER ALGORITHMS

5.1 Introduction

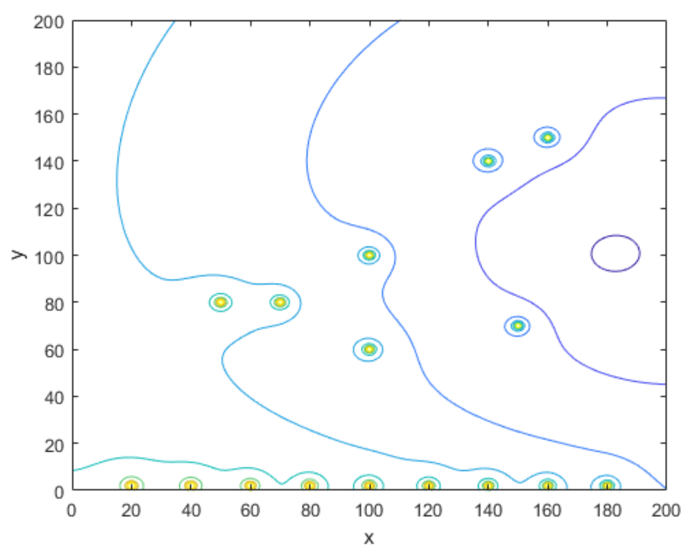
In the previous chapter, the simulations of the proposed improved Artificial Potential Field algorithm were performed in various configuration spaces and the results were produced, which showed that it is very efficient in a broad range of configuration spaces. This chapter is dedicated to study and compare the performance of the proposed improved Artificial Potential Field algorithm with other relevant algorithms for robotic path planning such as Visibility Graph, Rapidly-exploring Random Trees (RRT), A*, Genetic Algorithm, and Fuzzy Logic. The MATLAB implementations of the methods are collected from MATLAB Central or other code repositories such as R. Kala code repository [30], [31], then the images are regenerated, and the results are reproduced in the same configuration of computer. In each case, a similar configuration space is created by creating the same obstacle map. Then at each step snapshots of both the algorithms is reproduced in the comparisons. At the end, a comparative time study is performed to show how the proposed improved Artificial Potential Field algorithm excels over other algorithms.

5.2 Comparison between the proposed algorithm and the Visibility Graph method

To compare the proposed improved Artificial Potential Field method with the Visibility Graph algorithm, a similar configuration space is taken. The algorithm used with Visibility Graph method to find the path from the graph is Dijkstra’s algorithm. At first step, the Visibility Graph method finds all the possible paths, then Dijkstra’s algorithm finds the shortest path among the possible paths. Fig. 5.1 shows the map of both the algorithms.



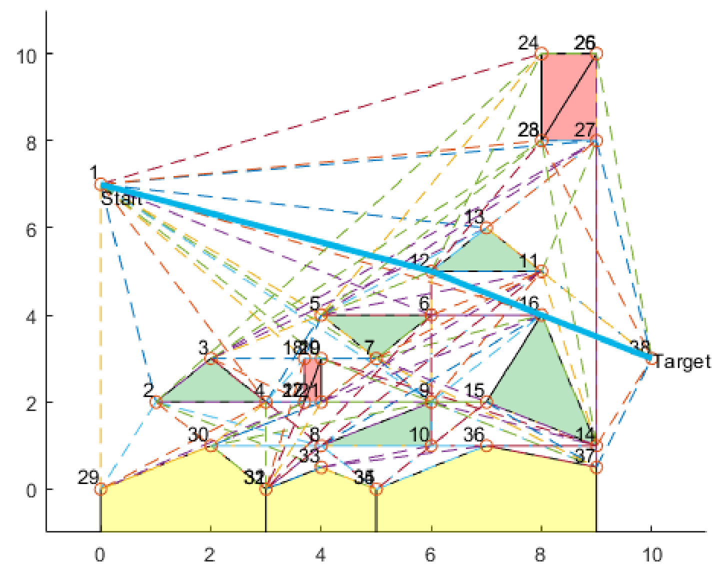
(a)



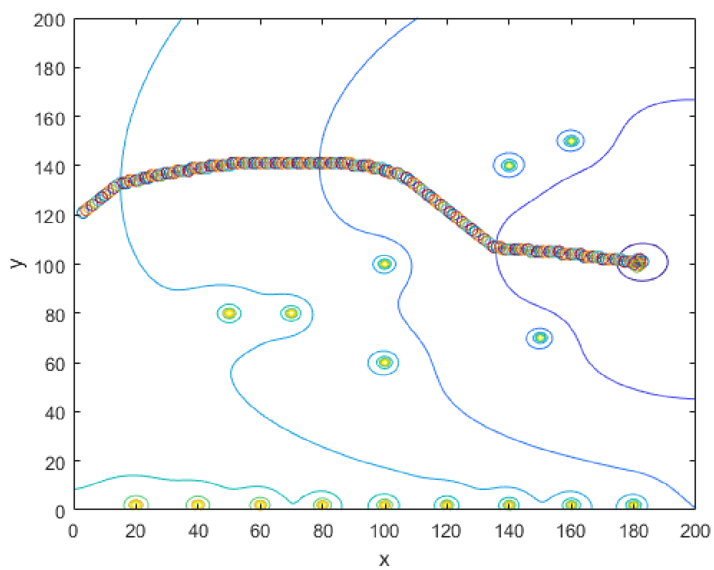
(b)

Figure 5.1 Configuration space of the two algorithms: (a) Visibility Graph, (b) improved Artificial Potential Field

It is observable from the configuration space that both maps are same and hence a comparison can be performed. Fig. 5.2 shows the path of the robot taken in two algorithms.



(a)

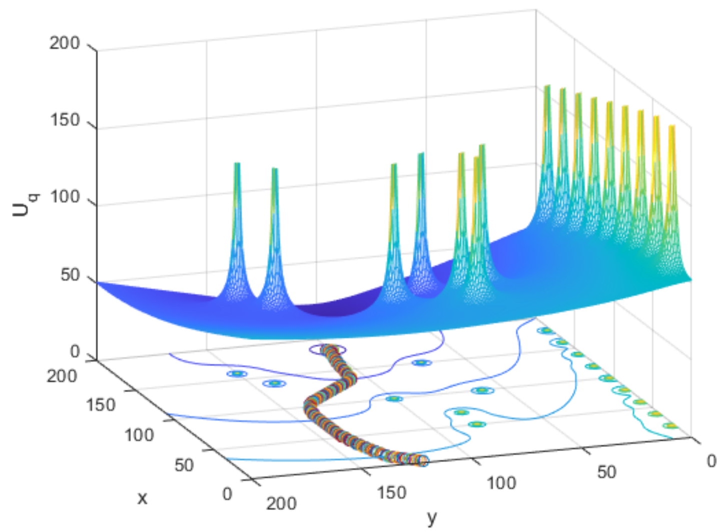


(b)

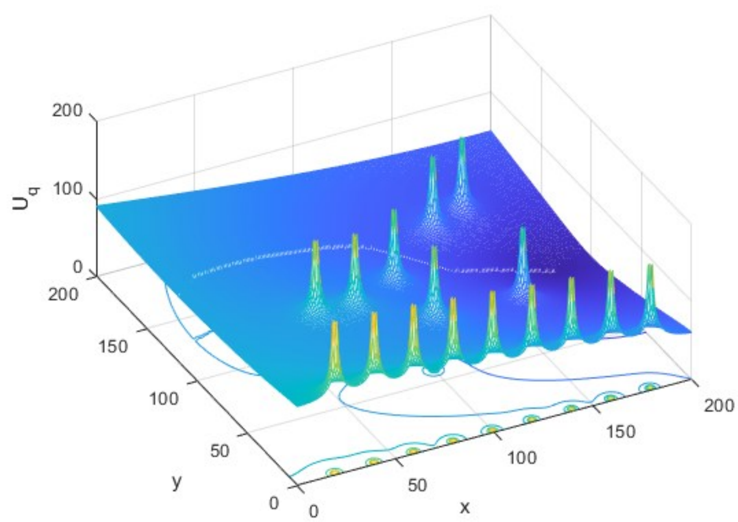
Figure 5.2 Path taken by the robot in: (a) Visibility Graph algorithm, (b) improved Artificial Potential Field algorithm

5.2.1 Advantages of proposed algorithm over Visibility Graph method

From Fig. 5.2 it is observable that the path taken in the proposed improved Artificial Potential Field is smoother and more optimal as it avoids touching the corner of the obstacles, which is a common pitfall of the Visibility Graph algorithm. Further, the path taken by the robot in improved Artificial Potential Field can be analyzed from the potential field of the configuration space and the modification performed by the robot to the potential field. Fig. 5.3 shows both the potential field maps.



(a)



(b)

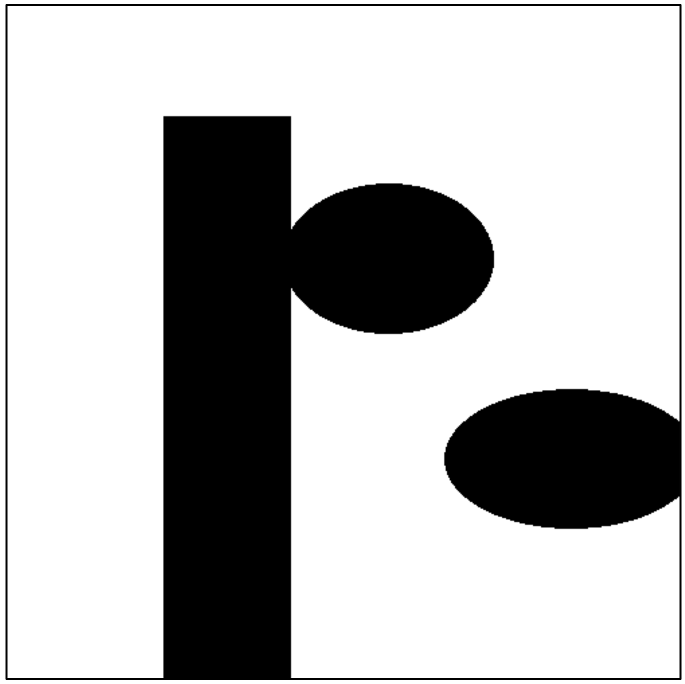
Figure 5.3 Potential field of the configuration space: (a) before the robot traversed, (b) after the robot traversed

From Fig. 5.3 it is observed that the proposed improved Artificial Potential Field algorithm can modify the artificial potential field of a configuration space in such an eclectic way that a wide variety of configuration space can be navigated using the algorithm. Moreover, the mean time taken by the Visibility Graph Dijkstra's algorithm is 1.12 seconds, and the mean time taken by the proposed improved Artificial Potential Field algorithm for the same configuration space is 0.39 seconds. It concludes the superiority of the proposed algorithm over the Visibility Graph method. Hence, the proposed improved Artificial Potential Field algorithm is more desirable.

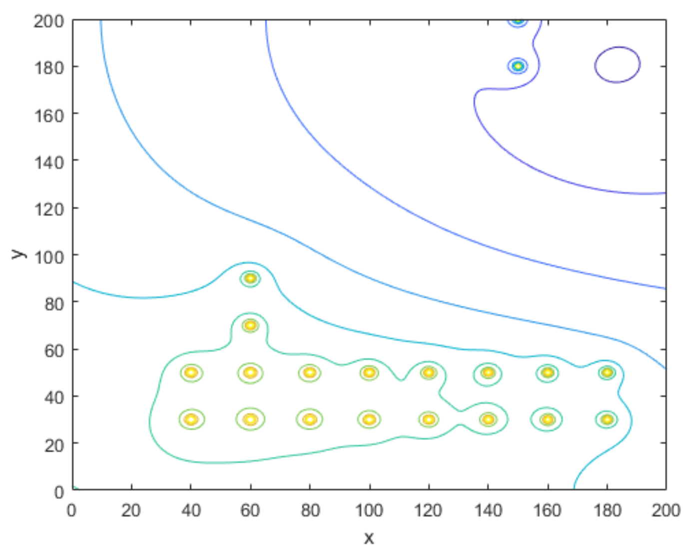
5.3 Comparison between the proposed algorithm and the Rapidly-exploring Random Trees (RRT) method

To compare the proposed improved Artificial Potential Field algorithm with the Rapidly-exploring Random Trees algorithm, the same configuration space is taken for both of the algorithms. Fig. 5.4 shows the map of both the algorithms.

It is observable from the configuration space that both maps are same and hence a comparison can be performed. Fig. 5.5 shows the tree formed and the artificial potential field generated in the respective algorithms.

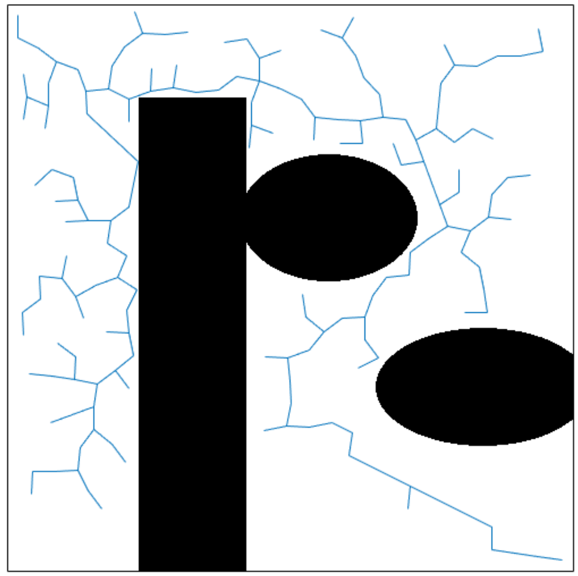


(a)

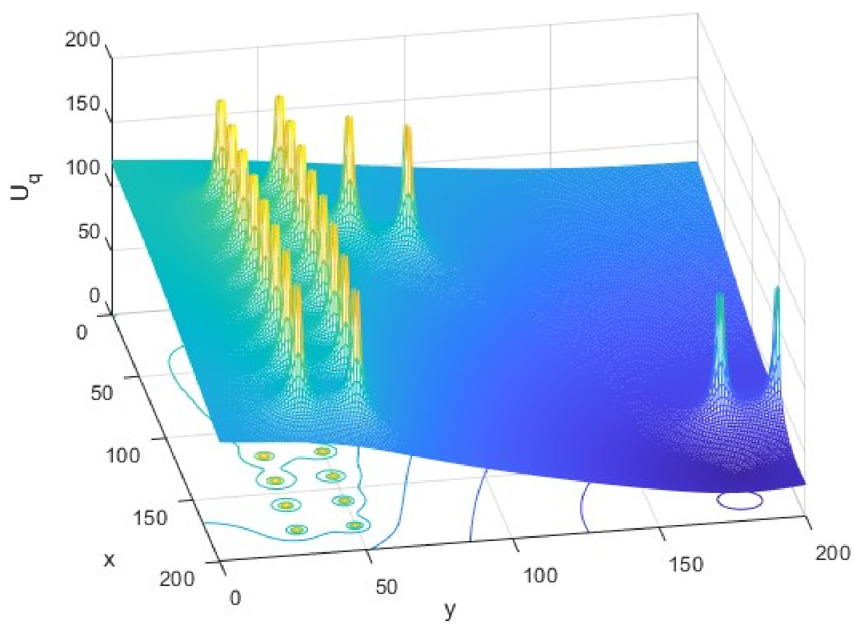


(b)

Figure 5.4 Configuration space of the two algorithms: (a) Rapidly-exploring Random Trees, (b) improved Artificial Potential Field



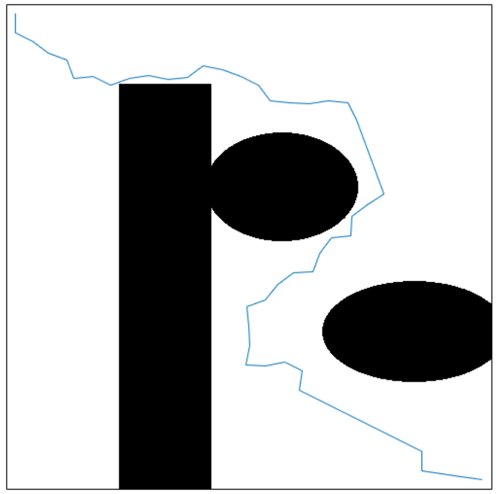
(a)



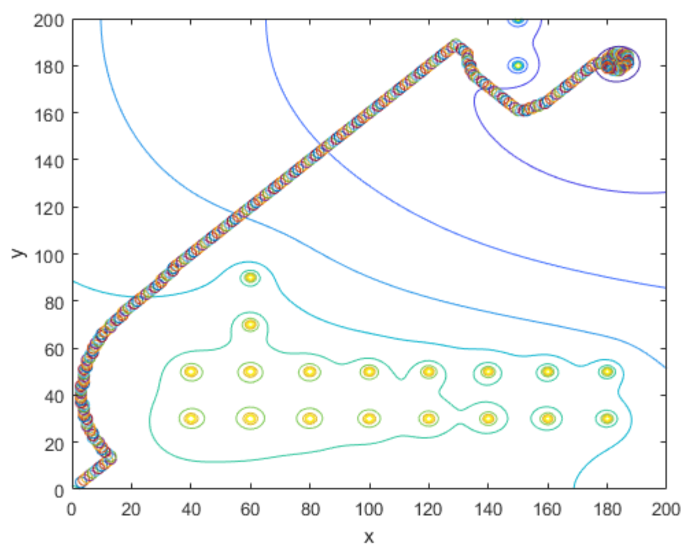
(b)

Figure 5.5 (a) The tree formed by the RRT algorithm, (b) the artificial potential field generated by the improved Artificial Potential Field algorithm

From Fig. 5.5 it is observable that in the RRT method, the tree has found the goal, and in the improved Artificial Potential Field method the field has been generated and the goal is at the global minima. It can also be seen that there are several local minima that the algorithm has to overcome. Fig. 5.6 shows the paths taken by both the methods.



(a)



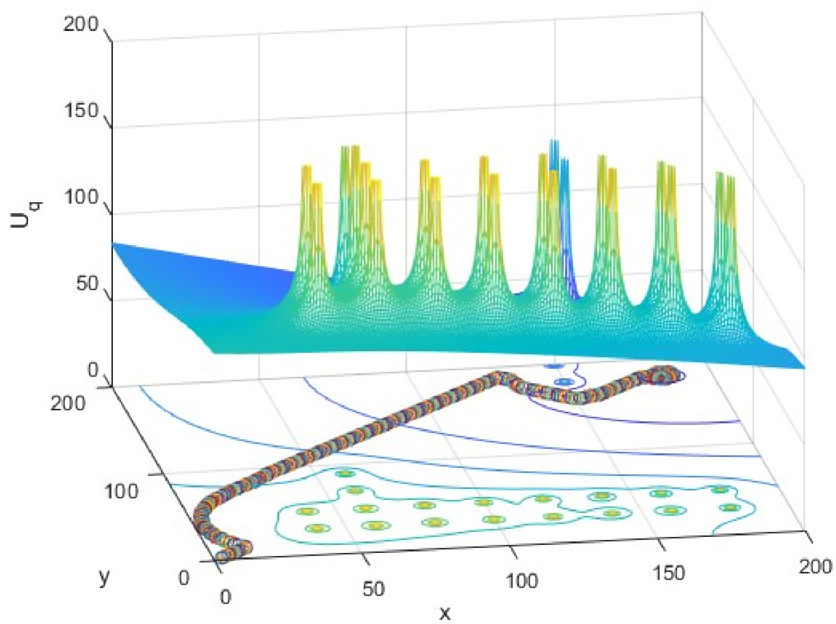
(b)

Figure 5.6 Path taken by the robot in: (a) Rapidly-exploring Random Trees method, (b) improved Artificial Potential Field method

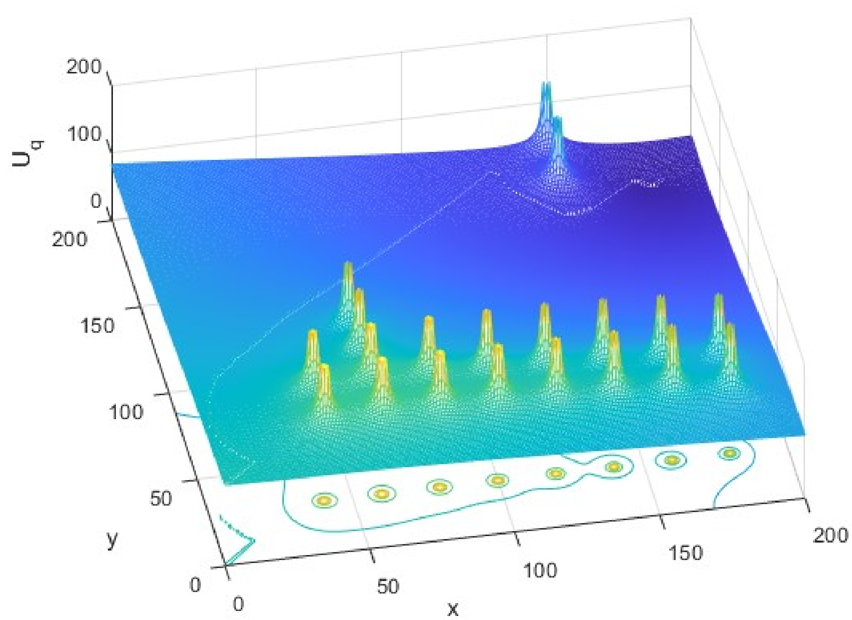
5.3.1 Advantages of the proposed algorithm over Rapidly-exploring Random Trees (RRT) method

From Fig. 5.6 it is observable that in case of RRT, the tree must grow in all direction first, to find the goal from the start. On the other hand, the improved Artificial Potential Field method has to develop the artificial potential field of the configuration space, which is much more robust. Further, the path taken by the robot in improved Artificial Potential Field can be analyzed from the potential field of the configuration space and the modification performed by the robot to the potential field. Fig. 5.7 shows both the potential field maps.

From Fig. 5.7 it is observed that the proposed improved Artificial Potential Field algorithm can modify the artificial potential field of a configuration space in such an eclectic way that a wide variety of configuration space can be navigated using the algorithm. Moreover, the mean time taken by the Rapidly-exploring Random Trees algorithm is 5.36 seconds, and the mean time taken by the proposed improved Artificial Potential Field algorithm for the same configuration space is 0.39 seconds. It concludes the superiority of the proposed algorithm over the RRT method. Hence, the proposed improved Artificial Potential Field algorithm is more desirable.



(a)

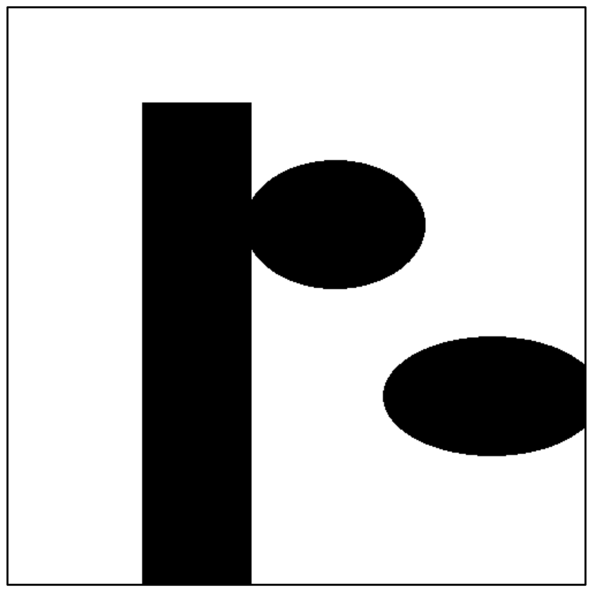


(b)

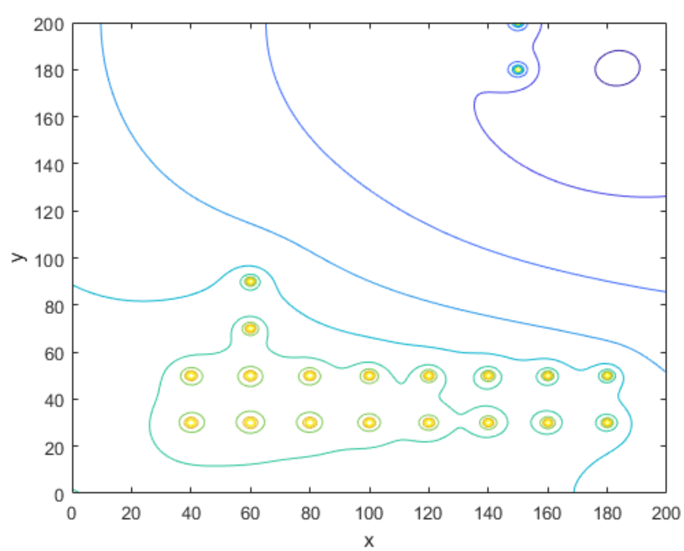
Figure 5.7 Potential field of the configuration space: (a) before the robot traversed, (b) after the robot traversed

5.4 Comparison between the proposed algorithm and the A* algorithm

To compare the proposed improved Artificial Potential Field algorithm with the A* algorithm, the same configuration space is taken for both of the algorithms. Fig 5.8 shows the map of both the algorithms.



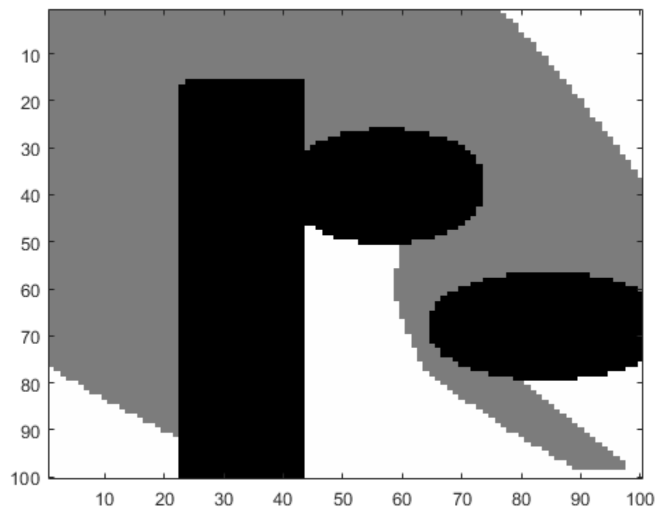
(a)



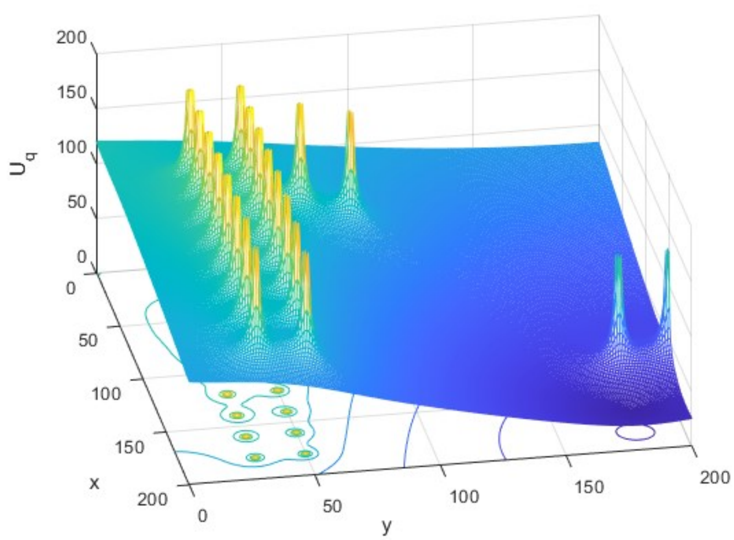
(b)

Figure 5.8 Configuration space of the two algorithms: (a) A* algorithm, (b) improved Artificial Potential Field

It is observable from the configuration spaces that both maps are same and hence a comparative study can be performed. Fig. 5.9 shows the heuristics performed and the artificial potential field generated by the respective algorithms.



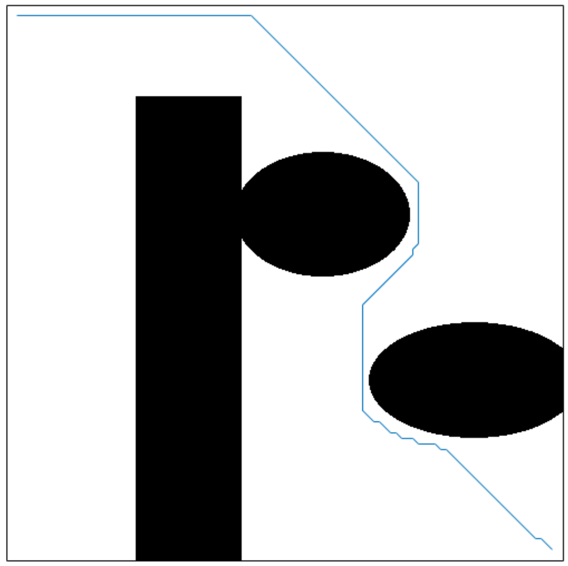
(a)



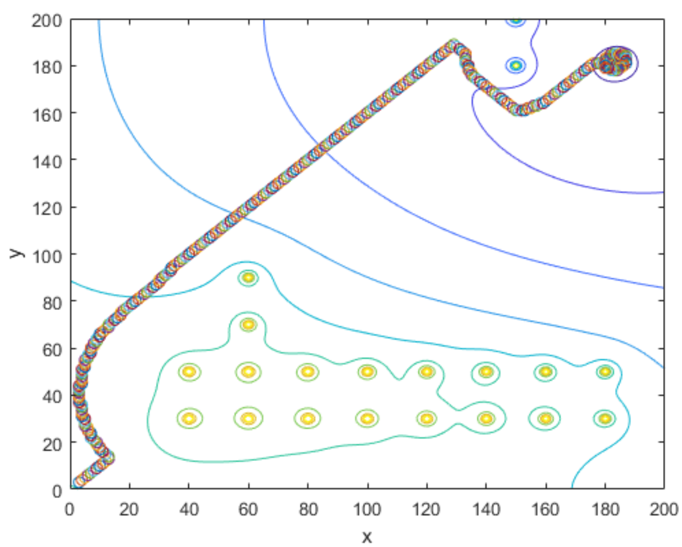
(b)

Figure 5.9 (a) The heuristics performed by the A* algorithm, (b) the artificial potential field generated by the improved Artificial Potential Field algorithm

From Fig. 5.9 it is observable that in the A* method, the heuristics have found the goal, and in the improved Artificial Potential Field method the field has been generated and the goal is at the global minima. It can be also seen that there are several local minima that the algorithm has to overcome. Fig. 5.10 shows the paths taken by both the methods.



(a)



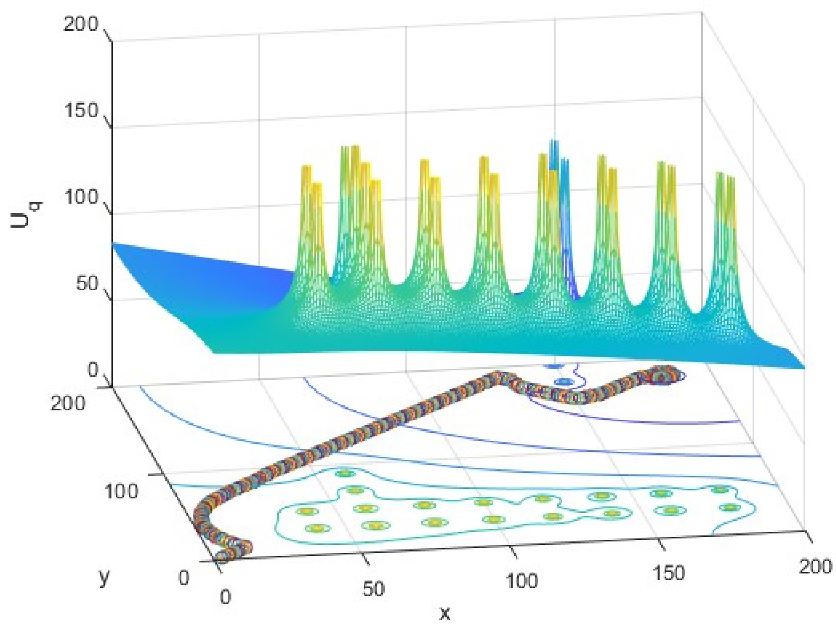
(b)

Figure 5.10 Path taken by the robot in: (a) A* method, (b) improved Artificial Potential Field method

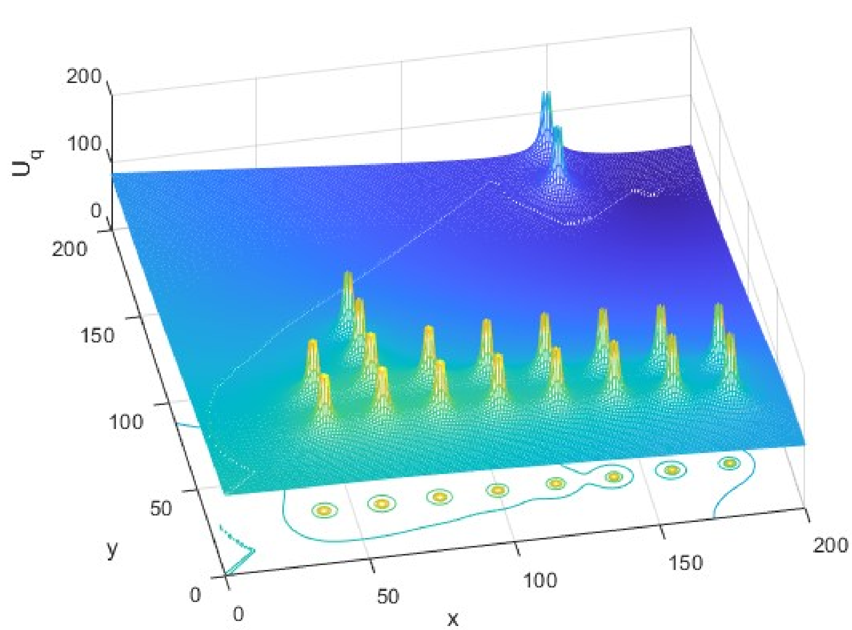
5.4.1 Advantages of the proposed algorithm over the A* algorithm

From Fig. 5.10 it is observable that in case of A* the parameters are calculated, which is computationally very much expensive. On the other hand, the improved Artificial Potential Field method has to develop the artificial potential field of the configuration space, which is computationally light and much more robust. It can be said that the proposed improved Artificial Potential Field algorithm is more desirable. Further, the path taken by the robot in improved Artificial Potential Field can be analyzed from the potential field of the configuration space and the modifications performed by the robot to the potential field. Fig. 5.11 shows both the potential field maps.

From Fig. 5.11 it is observed that the proposed improved Artificial Potential Field algorithm can modify the artificial potential field of a configuration space in such an eclectic way that a wide variety of configuration space can be navigated using the algorithm. Moreover, the mean time taken by the A* algorithm is 128.64 seconds, and the mean time taken by the proposed improved Artificial Potential Field for the same configuration space is 0.39 seconds. It concludes the supremacy of the proposed algorithm over the A* method. Hence, the proposed improved Artificial Potential Field algorithm is more desirable.



(a)

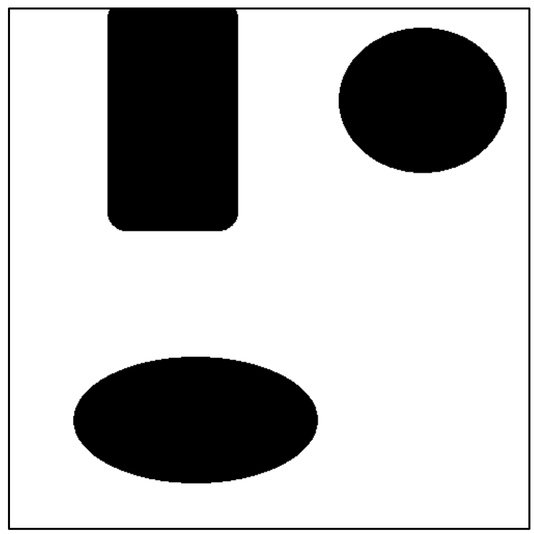


(b)

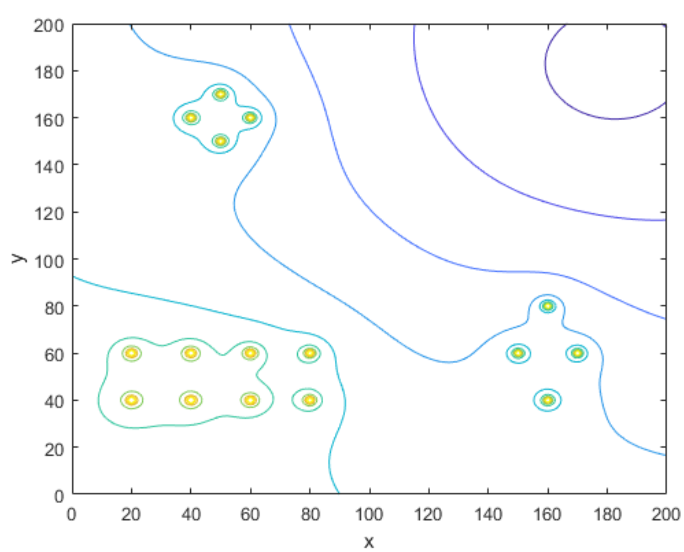
Figure 5.11 Potential field of the configuration space: (a) before the robot traversed, (b) after the robot traversed

5.5 Comparison between the proposed algorithm and the Genetic Algorithm (GA) method

To compare the proposed improved Artificial Potential Field algorithm with the Genetic Algorithm, the same configuration space is considered for both of the algorithms. Fig. 5.12 shows the map of both the algorithms.

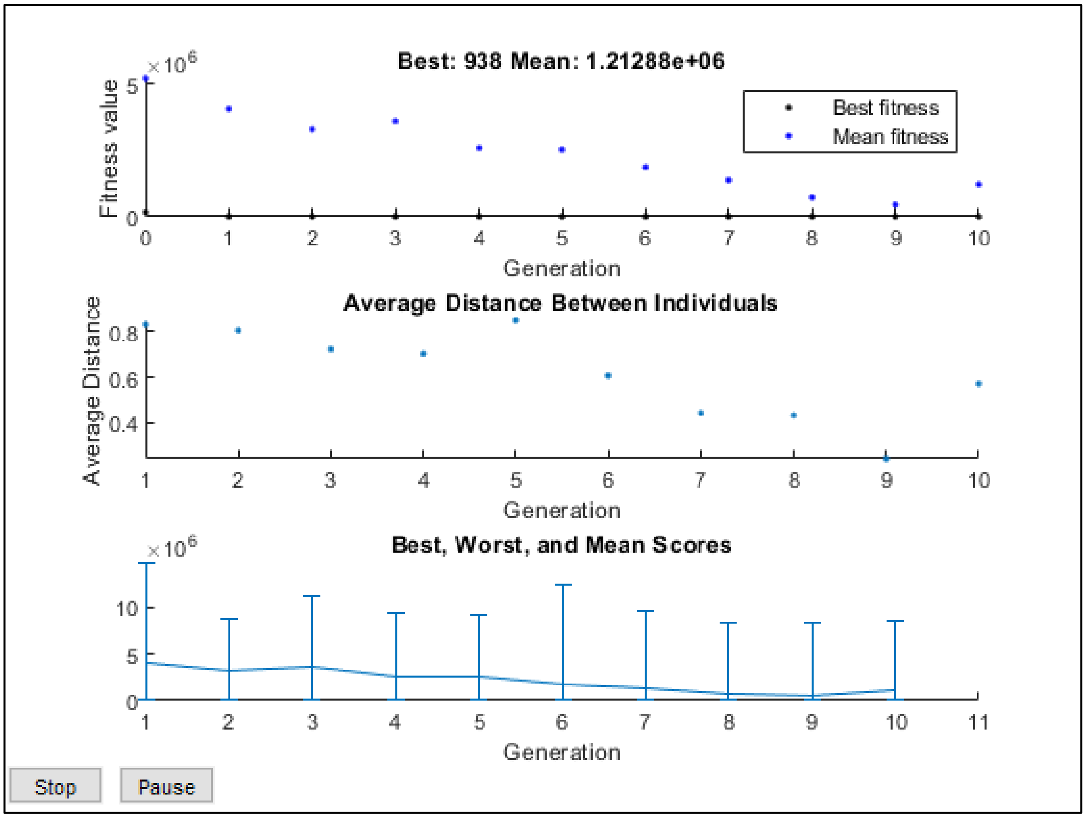


(a)

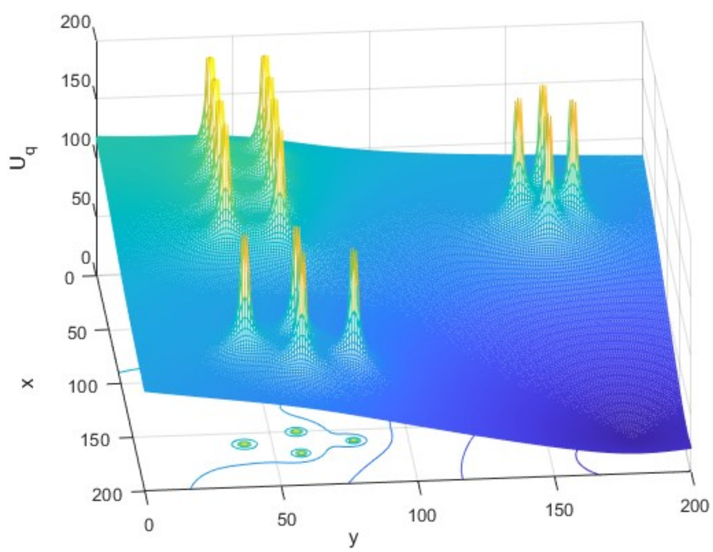


(b)

Figure 5.12 Configuration space of the two algorithms: (a) Genetic Algorithm, (b) improved Artificial Potential Field



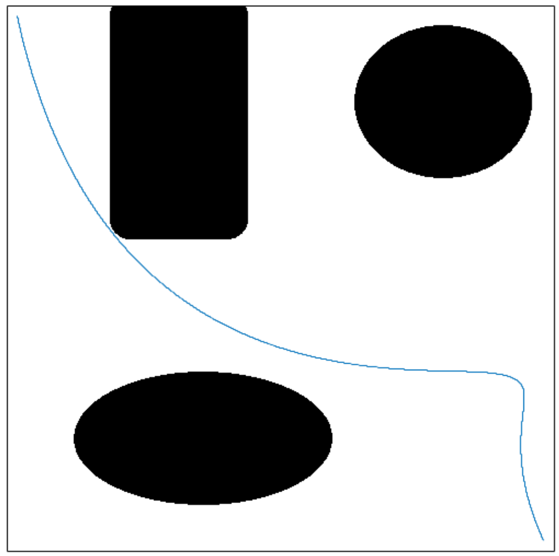
(a)



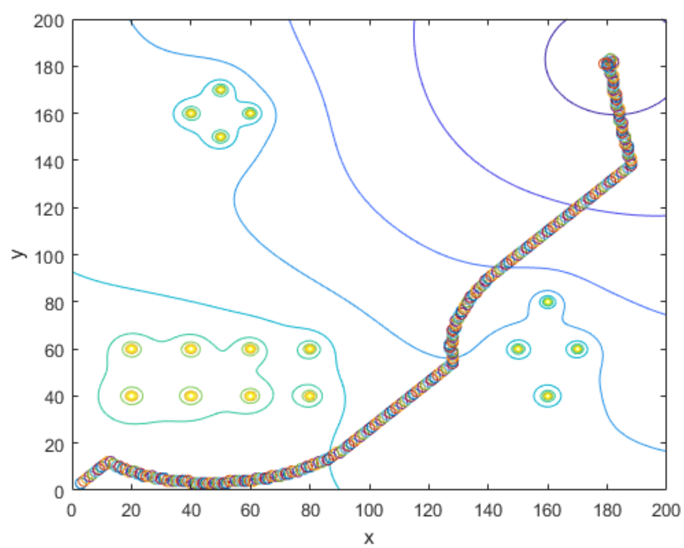
(b)

Figure 5.13 (a) The computations performed by the Genetic Algorithm, (b) the artificial potential field generated by the improved Artificial Potential Field algorithm

It is observable from the configuration spaces that both maps are same and hence a comparative study can be performed. Fig. 5.13 shows the generation computations performed and the artificial potential field generated by the respective algorithms.



(a)



(b)

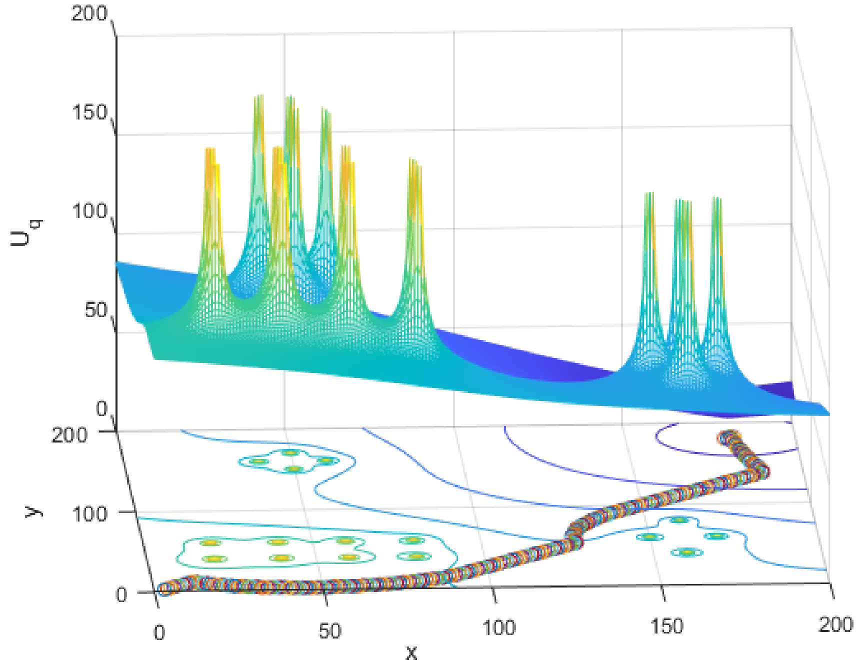
Figure 5.14 Path taken by the robot in: (a) Genetic Algorithm method, (b) improved Artificial Potential Field method

From Fig. 5.13 it is observable that in the Genetic Algorithm method, the heuristics have found the goal, and in the improved Artificial Potential Field method the field has been generated and the goal is at the global minima. It can also be seen that there are several local minima that the algorithm has to overcome. Fig. 5.14 shows the paths taken by both the methods.

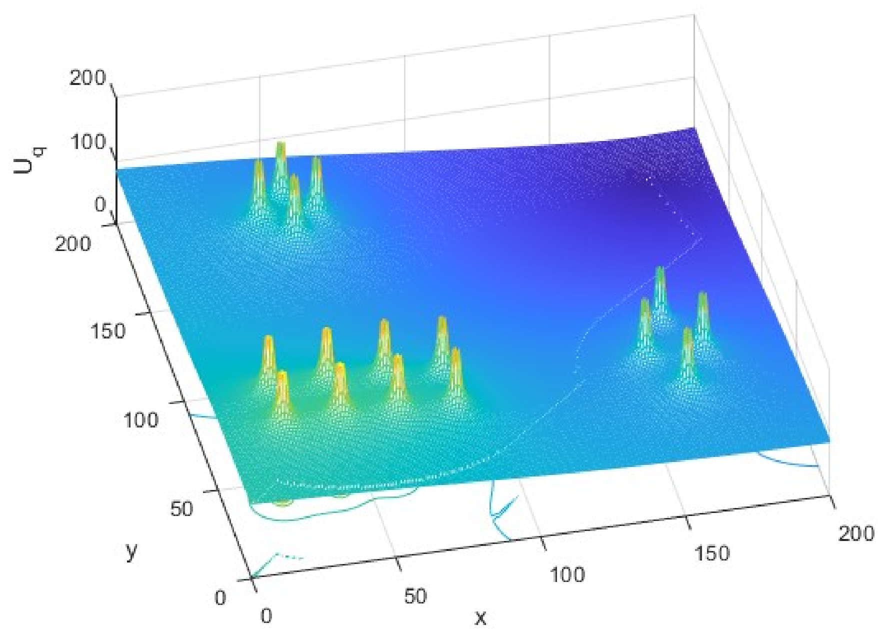
5.5.1 Advantages of the proposed algorithm over the Genetic Algorithm (GA) method

From Fig. 5.14 it is observable that in case of the Genetic Algorithm, several generations are calculated, which is computationally very much expensive. On the other hand, the improved Artificial Potential Field method has to develop the artificial potential field of the configuration space, which is computationally light and much more robust. It can be said that the proposed improved Artificial Potential Field algorithm is more desirable. Further, the path taken by the robot in improved Artificial Potential Field can be analyzed from the potential field of the configuration space and modifications performed by the robot to the potential field. Fig. 5.15 shows both the potential field maps.

From Fig. 5.15 it is observed that the proposed improved Artificial Potential Field algorithm can modify the artificial potential field of a configuration space in such an eclectic way that a wide variety of configuration space can be navigated using the algorithm. Moreover, the mean time taken by the Genetic Algorithm is 8.27 seconds, and the mean time taken by the proposed improved Artificial Potential Field for the same configuration space is 0.36 seconds. It concludes the supremacy of the proposed algorithm over the Genetic Algorithm method. Hence, the proposed improved Artificial Potential Field algorithm is more desirable.



(a)

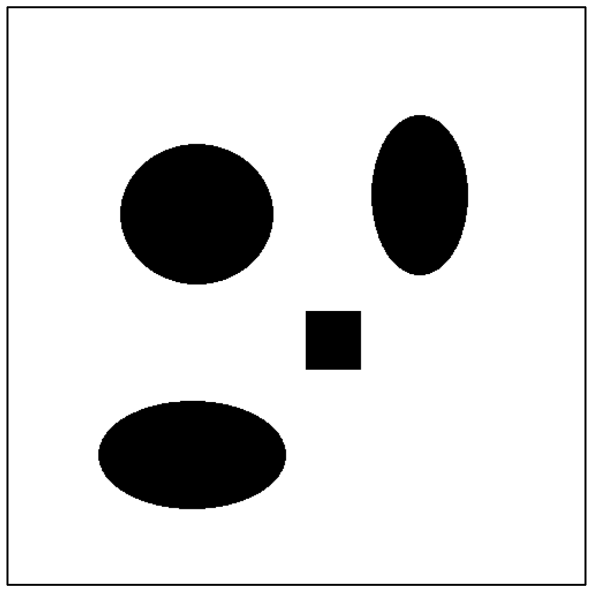


(b)

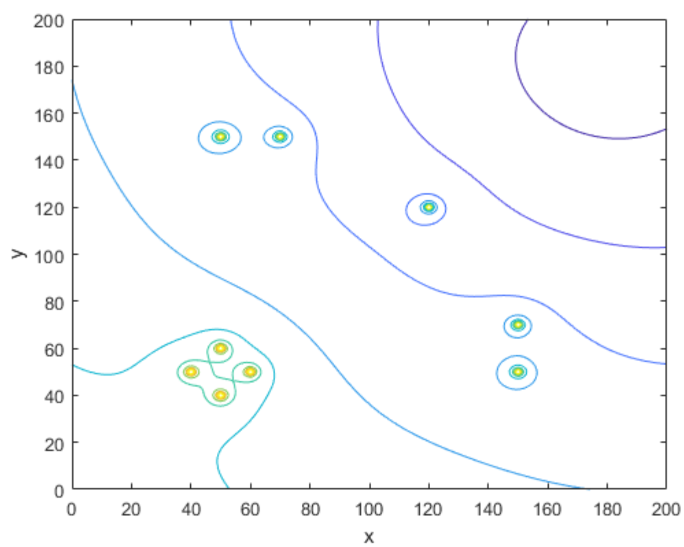
Figure 5.15 Potential field of the configuration space: (a) before the robot traversed, (b) after the robot traversed

5.6 Comparison between the proposed algorithm and the Fuzzy Logic method

To compare the proposed improved Artificial Potential Field algorithm with the Fuzzy Logic algorithm, the same configuration space is taken for both of the algorithms. Fig. 5.16 shows the map of both the algorithms.

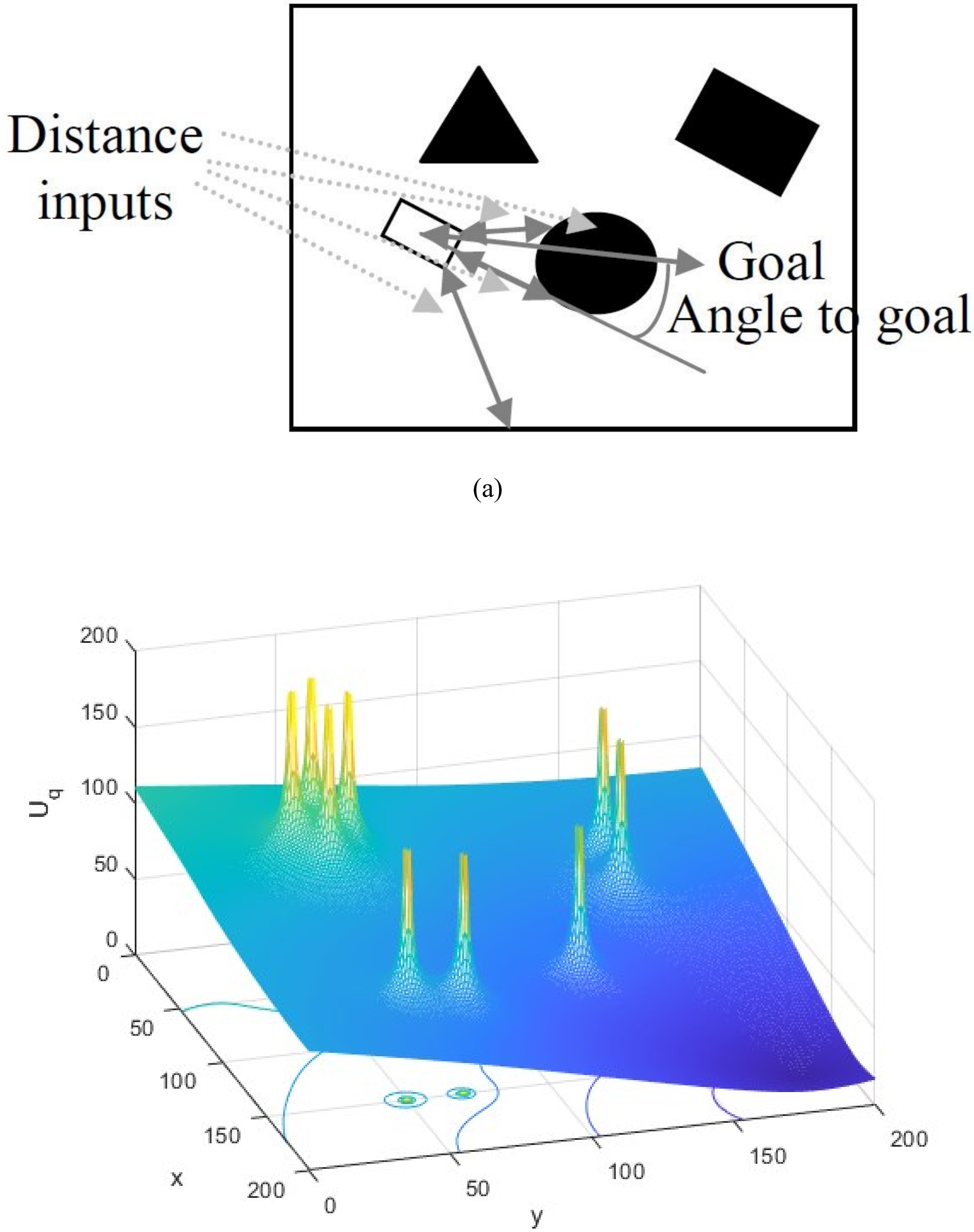


(a)



(b)

Figure 5.16 Configuration space of the two algorithms: (a) Fuzzy Logic, (b) improved Artificial Potential Field



(a)

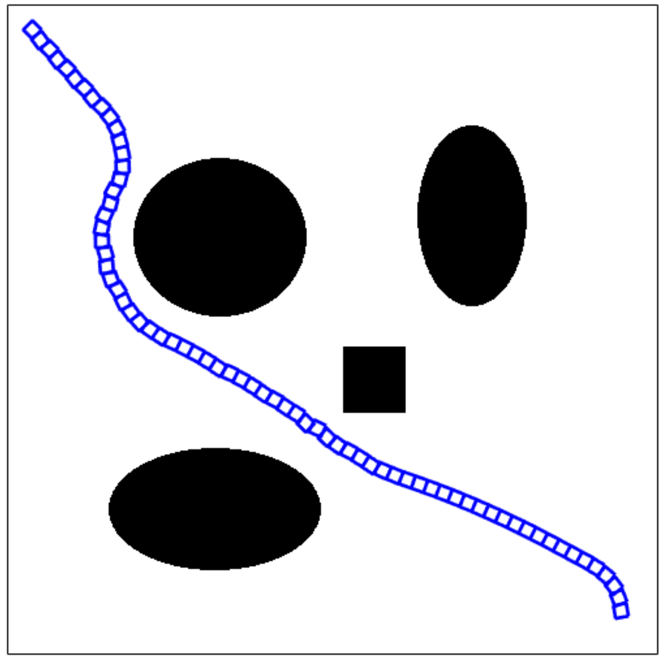
(b)

Figure 5.17 (a) Inputs of the Fuzzy Logic algorithm [31], (b) the artificial potential field generated by the improved Artificial Potential Field algorithm

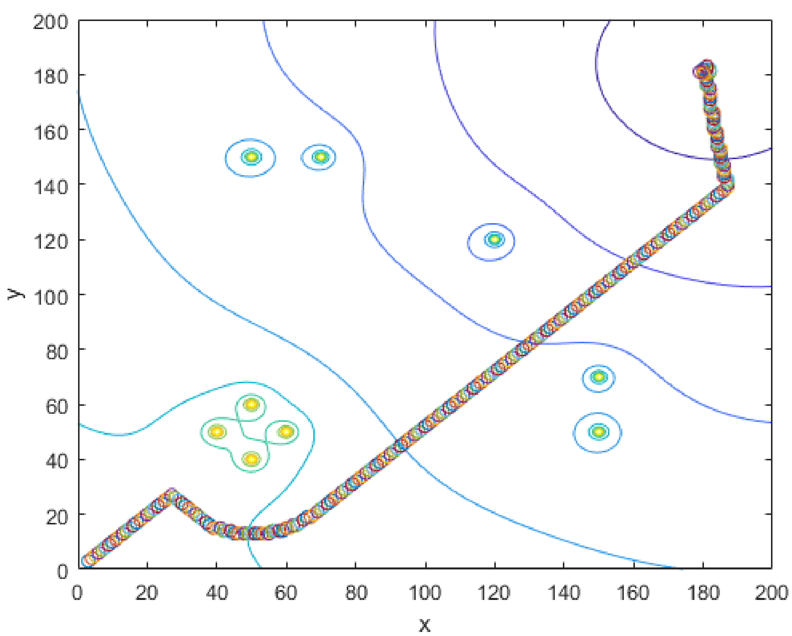
It is observable from the configuration spaces that both maps are same and hence a comparative study can be performed. In order to solve the problem using fuzzy logic, first few inputs which best represent the situation that the robot is currently placed in is selected. The decision of motion is made purely on the basis of these inputs and not the actual scenario. For this problem 6 inputs are selected. These are distance from the obstacle in front, distance from the obstacle at the front left diagonal, distance from the obstacle at the front right diagonal, angle between the heading direction of robot and the goal, distance from the goal and preferred turn. The different inputs are summarized in Fig. 5.17.

The last input, preferred turn indicates whether it would be beneficial to turn clockwise or anti-clockwise, all other inputs ignored. A simple rule is used to set the parameter. If the distance of the front obstacle is more than a given tolerance [31], the robot will turn towards the goal. If the front obstacle is close and a new front obstacle is encountered, turn using the side of the goal is preferred. If the front obstacle is close and the same obstacle as encountered in the previous step is found, the same turn as made previously is repeated.

The fuzzy system produces a single output, which is the steering to make or the immediate angular speed. The fuzzy rules are written such that the robot avoids the obstacles and aligns itself toward the goal. The fuzzy system is a result of a lot of manual tuning of the rules and membership functions, over a wide variety of scenarios [31]. Fig. 5.18 shows the path taken by both the methods.



(a)



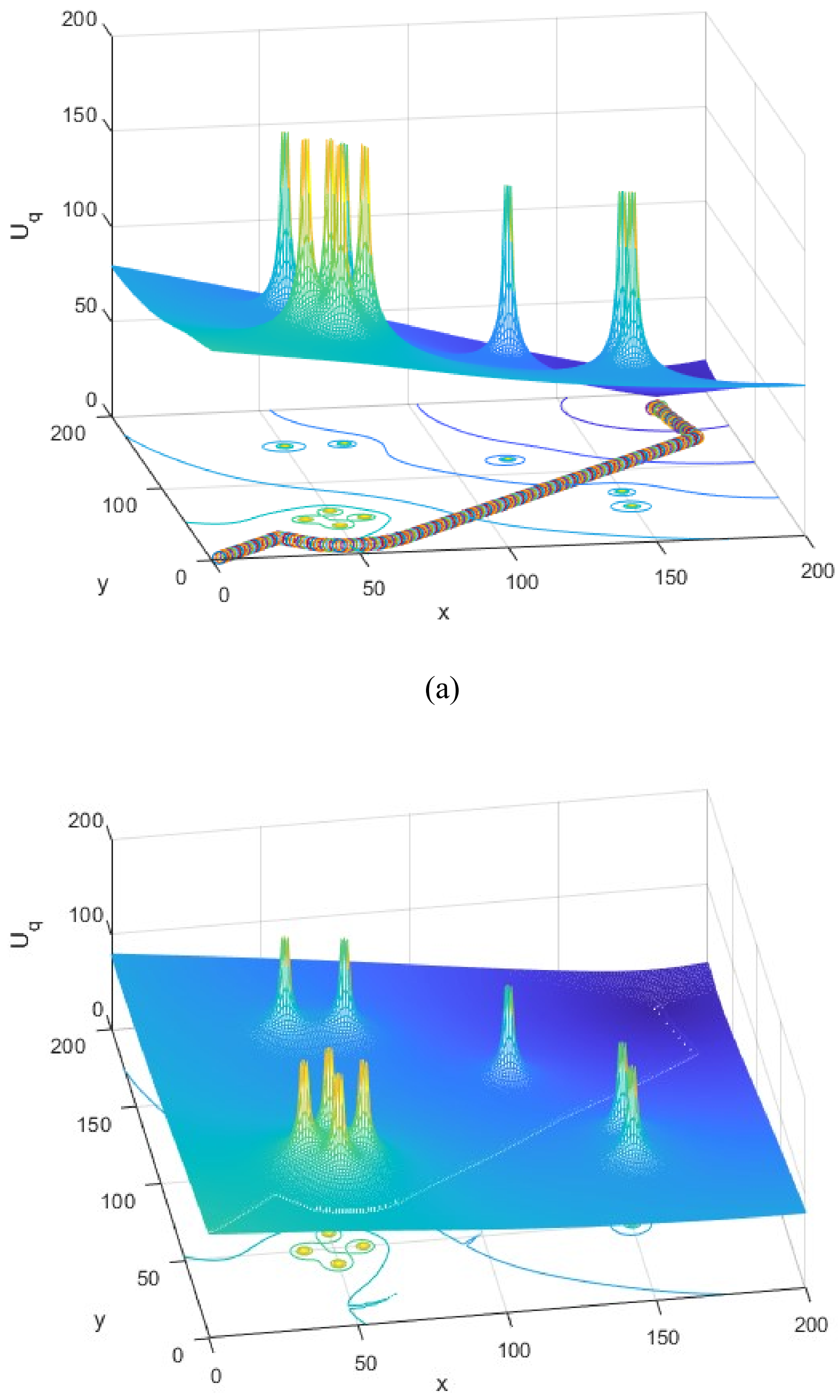
(b)

Figure 5.18 Path taken by the robot in: (a) Fuzzy Logic method, (b) improved Artificial Potential Field method

5.6.1 Advantages of the proposed algorithm over Fuzzy Logic method

From Fig. 5.18 it is observable that in case of the Fuzzy Logic algorithm, the direction of the robot is calculated at every step, which is computationally expensive. On the other hand, the proposed improved Artificial Potential Field algorithm has to develop the artificial potential field of the configurations space, which is computationally light and robust. In a complex configuration space, the Fuzzy Logic method would face difficulties, whereas the proposed method would navigate easily. Further, the path taken by the robot in the improved Artificial Potential Field can be analyzed from the potential field of the configuration space and the modifications performed by the robot to the potential field. Fig. 5.19 shows both the potential field maps.

From Fig. 5.19 it is observed that the proposed improved Artificial Potential Field algorithm can modify the artificial potential field of a configuration space in such an eclectic way that a wide variety of configuration spaces can be navigated using the algorithm. Moreover, the mean time taken by the Fuzzy Logic algorithm is 2.67 seconds, and the mean time taken by the proposed improved Artificial Potential Field algorithm for the same configuration space is 0.34 seconds. It concludes the supremacy of the proposed algorithm over the Fuzzy Logic method. Hence, the proposed improved Artificial Potential Field algorithm is more desirable.



(a)

(b)

Figure 5.19 Potential field of the configuration space: (a) before the robot traversed, (b) after the robot traversed

Chapter 6

CONCLUSIONS AND FUTURE WORK

6.1 Summary

This thesis has proposed a path planning algorithm based on artificial potential field for mobile robots in structured environments. In order to do that the thesis has been oriented in a proper manner which includes discussion on path planning and obstacle avoiding methods, derivation of the theory using mathematics, a detailed simulation study of the proposed path planning method for path planning in real environments, and a comparative study to establish the efficiency of the proposed algorithm over other existing algorithms. The main objective of this research work is to introduce a path planning method to overcome the deadlock problem associated to the traditional one by optimizing the potential field in the traditional Artificial Potential Field method.

In the first part of this thesis, a comprehensive overview of the mobile robot path planning is given, which provides a necessary background for this research. The various path planning categories are described, and a taxonomy of path planning problem is presented. Three fundamental navigation functions – localization, mapping and motion planning are discussed. The present research is basically an optimization problem to reduce the computation time and to enhance the smoothness of the trajectory of the robot.

Then, a detailed literature review was carried out on various robotic path planning methods and the advantages and limitations of those methods were discussed. The ideas of Configuration Space, Free Space, Obstacle Space and Free Path were presented. The methods were categorized on the basis of completeness of the algorithm. On the basis of this review, Artificial Potential Field method was selected as the basic technique for the proposed path planning algorithm.

Further, the Artificial Potential Field method was discussed in detail. A significant drawback of this method is deadlock which was encountered in the present thesis. To overcome the problem of deadlock, the concept of Added Potential was introduced. The process to address the deadlock problem using Added Potential concept was explained.

And finally, the improved Artificial Potential Field algorithm was developed and proposed.

To check the efficiency of the proposed improved Artificial Potential Field method a detailed a simulation study was carried out for a wide variety of configuration spaces such as the map with random obstacles, the cave, the wall, the bug trap and the maze. The algorithm was tested against each of the maps by using MATLAB. It was observed that the robot was able to overcome each of the hurdles arrived at its path. It was also shown that the traditional Artificial Potential Field method failed to reach the goal in each of these complex configuration spaces.

The performance of the proposed algorithm was also examined by a thorough parametric study with an objective to check the versatility of the same for different set of obstacles.

Moreover, a comparative study was carried out to test the efficiency and smoothness of the proposed improved Artificial Potential Field algorithm over a set of traditional algorithms such as Visibility Graph, Rapidly-exploring Random Trees, A*, Genetic Algorithm, and Fuzzy Logic. In each of the test cases the proposed improved Artificial Potential Field algorithm had lower computation time and smoother path. The proposed method excelled in each cases and proved to be more desirable than the other methods.

6.2 Conclusions

In the present work, an improved Artificial Potential Field method was proposed to solve the problem of robotic path planning.

The proposed method efficiently works for a wide variety of configuration spaces such as the map with random obstacles, the cave, the wall, the bug trap and the maze.

It was shown that the harmonic potential field addresses the issue of local minima by application of the superposition principle, maxima principle and minima principle of the harmonic function.

The mathematical background, characteristics and limitations of the traditional Artificial Potential Field method were studied. This method has several features such as good computational speed, capability to develop a smooth path, and compatibility to a wide range of applications. However, this method is inefficient in a region of local minima or a deadlock. The present research was dedicated to find an economical solution to address this issue without generating a high time complexity or a coarse path.

Further to address the local minima and deadlock in a better way, the proposed method has been improved by introducing a novel application of Added Potential (U_{add}). This U_{add} was added to attractive potential (U_{att}) and repulsive potential (U_{reps}) to produce the new improved potential field. This addition does not compromise the computation time and the smoothness of the path trajectory. This results in improved efficiency with no additional computation cost.

In this regard, it is viable to mention that the popular heuristic method like A* could be used to solve the present problem of robotic path planning. However, this computationally heavy method would result in a slow and inefficient solution.

The theoretical findings of the present research stand experimentally verified through a detailed simulation study. At first, the deadlock and goal-non reachability issues associated with the traditional potential field method were examined under different situations and the drawbacks were analyzed. Next, the novel application of Added Potential, proposed in this thesis, was incorporated with the existing method and the simulation results were examined. A comparative study of the traditional method and proposed method was carried out for different cases. It is observed that the novel method could overcome the drawbacks of the traditional method more efficiently. Hence, the mathematical background and concept of the proposed method is verified.

Further, to examine the robustness of the proposed method, a parametric study was carried out for randomly distributed multiple obstacles by varying the number and location/coordinate of the obstacles. The performance criterion of this robustness study was the computation time and the smoothness of the path trajectory. In each case, the proposed method has shown much better performance than the traditional one. In addition to the simulation case study, a comparative runtime analysis with a number of environments was also presented. This analysis showed that the proposed method has the capability of guiding the robot without occurrence of the deadlock problem, which proves the superiority of the present algorithm over others.

Finally, a comparative study is performed between the proposed improved Artificial Potential Field method and a set of conventional path planning algorithms as mentioned in the previous section. The study concludes the supremacy of the proposed work over its counterparts not only in simplicity but also in time consumption and smoothness of the path.

6.3 Future Scope of Work

In future, the research work can be carried out in the following directions.

- The path planning can be further developed to reach the goal through a smoother path avoiding the delays at all the local minima. This will make the path cleaner and linear.
- To develop a more efficient path planning algorithm a new strategy can be adopted, in which the total area would be segmented into smaller parts. To exit from that smaller area would be the primary goal of the robot and this process will be repeated for the total area to achieve the major goal. This segmentation procedure would be highly effective against environments consisting of maze.

BIBLIOGRAPHY

[1] J. Kim, "Real-Time Obstacle Avoidance Using Harmonic Potential Functions Real-Time Obstacle Avoidance Using Harmonic Potential Functions *," *Comput. Eng.*, pp. 1–28, 1992.

[2] R. Daily and D. M. Bevly, "Harmonic potential field path planning for high speed vehicles," *Proc. Am. Control Conf.*, pp. 4609–4614, 2008.

[3] S. Panati, B. Baasandorj, and K. T. Chong, "Autonomous mobile robot navigation using harmonic potential field," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 83, no. 1, 2015.

[4] A. A. Masoud, "Planning with Discrete Harmonic Potential," 1991.

[5] N. Hantos, S. Iván, P. Balázs, and K. Palágyi, *peter corke Robotics book*, vol. 75, no. 1–2. 2015.

[6] O. Montiel, U. Orozco-Rosas, and R. Sepúlveda, "Path planning for mobile robots using Bacterial Potential Field for avoiding static and dynamic obstacles," *Expert Syst. Appl.*, vol. 42, no. 12, pp. 5177–5191, 2015.

[7] T. Chen and Q. Zhang, "Robot motion planning based on improved artificial potential field," *Proc. 2013 3rd Int. Conf. Comput. Sci. Netw. Technol. ICCSNT 2013*, pp. 1208–1211, 2014.

[8] M. Drahansky *et al.*, "Motion Planning for Mobile Robots" *Intech*, vol. i, no. tourism, p. 13, 2016.

[9] T. T. Mac, C. Copot, D. T. Tran, and R. De Keyser, "Heuristic approaches in robot path planning: A survey," *Rob. Auton. Syst.*, vol. 86, pp. 13–28, 2016.

[10] N. Sariff and N. Buniyamin, "An overview of autonomous mobile robot path planning algorithms," *SCOReD 2006 - Proc. 2006 4th Student Conf. Res. Dev. "Towards Enhancing Res. Excell. Reg.*, no. June, pp. 183–188, 2006.

[11] Ronald C Arkin, "Motor Schema — Based Mobile Robot Navigation." *The International Journal of Robotics Research, vol. 8, no. 4, Aug. 1989,* pp. 92–112, doi:10.1177/027836498900800406.

[12] E. Masehian, & D. Sedighizaddeh, "Classic and heuristic approaches in robot motion planning – A chronological review." *Engineering and technology (Vol. 23*, pp. 101–106). Germany: World Academy of Science, 2007.

[13] D. Eppstein, "Geometry in action: Voronoi diagrams." <*http://www.ics.uci. edu/eppstein/gina/voronoi.html*>*,* 1996.

[14] A. Bhattacharyya, E. Singla, & B. Dasgputa, "Robot path planning using silhouette method." *In 13th National conference on mechanisms and machines (NaCoMM07). Bangalore, India,* 2007.

[15] R. Siegwart, I. R. Nourbakhsh, & D. Scaramuzza, “Introduction to autonomous mobile robot (2nd ed.)”. *London, England: The MIT Press*, 2011.

[16] E. Masehian, & D. Sedighizaddeh, “Classic and heuristic approaches in robot motion planning – A chronological review”, *Engineering and technology (Vol. 23, pp. 101–106). Germany: World Academy of Science*, 2007.

[17] B. Zhang, W. Chen, & M. Fei, “An optimized method for path planning based on artificial potential field”, *Proceedings of the sixth international conference on intelligent systems design and applications. IEEE*, 2006.

[18] J. Berger, K. Jabeur, A. Boukhtouta, A. Guitouni, & A. Ghanmi, “A hybrid genetic algorithm for rescue path planning in uncertain adversarial environment”, *Evolutionary computation. IEEE* [pp. 1–8], 2010.

[19] C. Hocaoglu, & A. Sanderson, “Planning multiple paths with evolutionary speciation”, *Evolutionary computation. IEEE* [pp. 169–191], 2001.

[20] S. Li, M. Ding, C. Cai, & L. Jiang, “Efficient path planning method based on genetic algorithm combining path network”, *Genetic and evolutionary computing. IEEE* [pp. 194–197], 2010.

[21] K. Zhang, E. G. Collins, & A. Barbu, “An efficient stochastic clustering auction for heterogeneous robotic collaborative teams”, *Journal of Intelligent & Robotic Systems*, 72(3-4), 541–558, 2013.

[22] O. Montiel, R. Sepúlveda, O. Castillo, & P. Melin, “Ant colony test center for planning autonomous mobile robot navigation”, *Computer Applications in Engineering Education*, 21(2), 214–229, 2013.

[23] Q. Zhang, D. Chen, & T. Chen, “An obstacle avoidance method of soccer robot based on evolutionary artificial potential field”, *International conference on future energy, environment, and materials* [pp. 1792–1798], 2012.

[24] P. Vadakkepat, T. H. Lee, & L. Xin, “Application of evolutionary artificial potential field in robot soccer system”, *IEEE* [pp. 2781–2785], 2001.

[25] P. Vadakkepat, K. C. Tan, & M. L. Wang, “Evolutionary artificial potential fields and their application in real time robot path planning”, *Congress on evolutionary computation. IEEE* [pp. 256–263], 2000.

[26] C. Goerzen, Z. Kong, & B. Mettler, “A survey of motion planning algorithms from the perspective of autonomous UAV guidance”, *Springer – Journal of Intelligent & Robotic Systems*, 65–100, 2010.

[27] N. E. Nawa, T. Hashiyama, T. Furuhashi, & Y. Uchikawa, “A study on fuzzy rules discovery using pseudo-bacterial genetic algorithm with adaptive operator”, *Evolutionary computation. IEEE* [pp. 589–593], 1997.

[28] M. Fallah, F. Akbari, & A. Javan, “Implementation of common genetic and bacteriological algorithms in optimizing testing data in mutation testing”, *IEEE*, 2010.

[29] W.M.M. Tharindu Weerakoon, “Artificial Potential Field and Feature Extraction Method for Mobile Robot Path Planning in Structured Environments”, Kyushu Institute of Technology, 2016

[30] R. Kala, “Code for Robot Path Planning using Genetic Algorithms”, Indian Institute of Information Technology Allahabad, 2014

[31] R. Kala, “Code for Robot Path Planning using Fuzzy Logic”, Indian Institute of Information Technology Allahabad, 2014

[32] J. Latombe, Robot Motion Planning, Boston, MA: Kluwer Academic Publisher, 1991.

[33] Y. Hwang and N. Ahuja, "Gross Motion Planning - A Survey," ACM Computing Surveys, vol. 24, no. 3, pp. 219-291, 1992.

[34] C. Goerzen, Z. Kong and B. Mettler, "A Survey of Motion Planning Algorithms from the Perspective of Autonomous UAV Guidance," Journal of Intelligent and Robotic Systems, vol. 57, no. 1-4, pp. 65-100, 2010.