NEURAL NETWORKS BASED ADAPTIVE APPROACH FOR PATH PLANNING AND OBSTACLE AVOIDANCE FOR AUTONOMOUS MOBILE ROBOT (AMR)

INTERNATIONAL JOURNAL OF RESEARCH IN COMPUTER APPLICATIONS AND ROBOTICS
ISSN 2320-7345

NEURAL NETWORKS BASED ADAPTIVE APPROACH FOR PATH PLANNING AND OBSTACLE AVOIDANCE FOR AUTONOMOUS MOBILE ROBOT (AMR)

Najmuddin Aamer[1], Dr. Ramachandran S[2]

[1]*Research Scholar, V.M.U. Salem (Tamil Nadu), E-mail: najmuddinaamer@gmail.com*
[2] *SJB Institute of Technology, Bangalore. E-mail: ramachandr@gmail.com*

Abstract: - Autonomous robot navigation is an improvised method where stable, dynamic, known and unknown environmental parameters are used. In this work, an efficient algorithm is presented to improve the navigation performance of the robot in terms of movement of robot, obstacle identification, learning the environmental conditions and achieving the destination or target for known and unknown types of environments. This work focuses on the path planning and obstacle avoidance modelling for the robot. Proposed model considers dynamic obstacles environment also to show the obstacle avoidance efficiency of the model. MATLAB tool is used for the simulation and validation of algorithm. Obtained results from the simulation shows the robustness and adaptability of the proposed model. Simulation results show that when the proposed algorithm for robot is not applied that time path following error was 32% and in another case when the algorithm is applied the error is reduced to 4.7% for the same environment.

Keywords*:* Autonomous Mobile Robot Navigation, Path Planning, Obstacle Avoidance, Neural Network.

1. Introduction

Recently Autonomous robots have created much significance for late years because of their capacity to perform generally difficult undertakings in risky or remote situations. Now a day autonomous robots have been adequately utilized as a part of different regions of engineering researches, for example, aviation research, atomic exploration, product designing and so on. The major target in the robotic research is to discover an impact freeway from a given begin position to predefined target point. Path planning or discover way issue is surely understood in mechanical autonomy and it assumes an imperative part in the route of autonomous mobile robot. Navigation which is a procedure or action to arrange and coordinate a course or path is an errand that an autonomous robot must do effectively keeping in mind the end goal to move securely starting with one area then onto the next area without getting lost or crashing into different objects. The three general issues of route are localization of robot, planning the path and movement control. Between these three issues, it can be contended that path planning is a standout amongst the most imperative issues in the route process. Path planning empowers the determination and recognizable proof of a suitable way for the robot to navigate in the workspace zone [1]. Path planning techniques are delegated neighbourhood and global relying on the encompassing environment. In global path planning the encompassing environment is totally known not versatile robot so the way went by the portable robot is predefined, where as in neighbourhood way arranging the environment is totally obscure or halfway known not the versatile robot. So different sensors are utilized to see the data about the encompassing environment and arrangement the movement as needs be [2]. Recently various techniques have been developed for collision-free robot navigation system. These algorithms are based on Ant colony optimization, genetic algorithm and fuzzy logic method. For efficient robot navigation path planning is the main issue in these researches. Y. Hu et al. proposed a genetic algorithm for path planning of the autonomous robot. This algorithm utilizes problem-specific genetic algorithm to improve the efficiency of path planning and collision free movement of the robot [3]. Robot path planning has been an extensive research region, and numerous routines have been produced to handle this issue [4], for example, global C-space strategies [5], [6] potential field systems [7], [8] and neural systems approaches [9], [10]. Every technique has its own quality over others in specific angles. For the most part, the principle troubles for path planning issue are computational multifaceted nature, neighbourhood ideal, and flexibility. Analysts have dependably been looking for option and more productive approaches to take care of the issue. There is doubtlessly path planning can be saw as a optimization issue (e.g., short distance) under specific requirements (e.g., the given environment and impact free condition). Despite the fact that autonomous robot navigation is broadly utilized by individuals today, there are still numerous difficulties for individuals to outline great autonomous robot navigation. By and large, there are three principle challenges. The principal test is the obscure and element environment. Because of confinements of sensor capacities, a robot is extremely hard to know the worldwide environment amid its development. Resulting, the robot does not have a worldwide guide for the most part. Subsequently, it chooses its activities as per the sensor information of its neighbourhood surroundings. The second test is the sensor's impact commotion. As a rule, sensors can without much of stretch get wrong information estimation in a complex environment, which makes the robot select a wrong activity. The last test is the restriction of computational ability of the robot. In such circumstances, a productive route calculation is critical to explore the robot in a complex environment [11]. The methodology introduced in [12] is one of the first where static and moving obstructions are maintained a strategic distance from, in light of their present positions and speeds with respect to the robot. The moves are produced by selecting robot speeds outside of the speed impediments that would incite a crash at some future time. Arranging in the speed space makes it conceivable to consider the robot elements. This approach has been adjusted in [13] to the auto like robot kinematic display, and stretched out in [14] to consider unusually moving impediments. This has been finished by utilizing reachability sets to discover coordinating imperatives in the speed space, called Velocity Obstacle Sets. Another pioneer technique that has roused numerous others is the Dynamic Window [15] that is achieved specifically from the robot's elements, and is particularly intended to manage compelled speeds and increasing velocities. The strategy comprises two stages: initial, a legitimate subset of the control space is produced, and afterward an ideal arrangement (driving the robot with most extreme impediment leeway) is looked for inside of it. A progressive's speculation outline represents moving obstacles speeds and shapes is exhibited in [16], where a union of polygonal zones comparing to the non-permissible speeds controls the robot, and counteracts impacts. In [17], the Dynamic Window has been incorporated in a graph search method for path planning, to drive the robot directions inside of a global way. A planning methodology is additionally utilized as a part of [18], where the probability of hindrance positions is information to a rapidly investigating Random Tree technique.

Obstacle Avoidance Robots have been a dynamic range of innovative work in the course of recent decades. The sorts of obstacles can be arranged into two sorts.

Stationary Obstacles: Dividers, substantial furniture like beds, tables, seats and so forth fall into this class.

Non-Stationary Obstacles: These obstructions result from "confusion" in nature. The positional directions of these problems continue evolving. In our application, recognized impediments are added to the interim guide, and thus prepared like stationary hindrances until the impermanent readings are at long last extrapolated and overwritten by the new readings. Along these lines, the constant hindrances can likewise be taken consideration by briefly considering them as stationary problems.

Direction Following: A reference point on the robot must take after a direction in the Cartesian space (i.e., a geometric way with a related timing law) beginning from a given starting arrangement. One way to deal with hindrance shirking is divider after strategy. Here robot route depends on moving close by dividers at a predefined separation. On the off chance that a deterrent is experienced, the robot views the hindrance pretty much as another divider after its shape until it may continue its unique course. An all the more normally utilized strategy for impediment evasion depends on nervous recognition. The line joining the two edges is considered to speak to one of the deterrent's boundaries. A drawback with the edge recognition system is the need of the robot to stop before a snag keeping in mind the end goal to take into consideration a more accurate estimation.

Our proposed model to deal with the issue of navigation to objective focuses is a two stage process. At the point when the robot's way is unhindered, it explores straight towards the target, utilizing its global coordinates of position evaluation to guide it. At the point when the robot experiences impediments, it takes after utilizes the curve of the obstacles to avoid it. This calculation assesses the robot's position in connection to the obstructions and objective and figures out if it should continue to keep taking after the obstacle or whether it is safe to walk specifically towards the objective. Due to the instability in the robot's position and the trouble of figuring out if

an impediment is static or element, this calculation does not include any type of global memory of the robot's position. This implies that in some neurotic circumstances, the robot may come back to the initial position. The proposed algorithm incorporates a level of haphazardness to help the robot out of these sorts of circumstances.
The paper is organized as follows. In the next section, we discuss related work on navigation and obstacle avoidance of robots. In Section 3, we introduce models of obstacle avoidance control and physical motion, and prove that they guarantee passive safety and passive friendly safety with stationary as well as moving obstacles. We then model uncertainty explicitly and prove the safe motion plan. Section 4 concludes the paper.

2. Related work

In this section we discuss about the recent works which have been done in the field of robotics for path planning and obstacle avoidance for autonomous mobile robot control.
As we have discussed earlier that obstacle avoidance is the critical task for the mobile robots. In order to improve this Yaghmaie, F.A.; Mobarhani et al. proposed an escaping algorithm for mobile robots. This strategy depends on power field system and it is gathered that the robot performs SLAM and self-sufficient route in dynamic environment with no predefined data about element obstructions. The development of dynamic obstacle is anticipated by Kalman channel and is utilized for collision recognition reason. At the time of impact discovery, a changing power is added to repulsive and appealing strengths comparing to the static environment and leads robot to maintain a strategic distance from collision. Besides, a protected turning edge is characterized to guarantee safe route of the robot [19].
Sugiyama, S.; Yamada et al. [20] proposed the hypothesis and the simulation of an enhanced speed potential methodology for path planning by which a versatile robot abstains from using so as to stand and/or moving obstructions the hydrodynamic potential. This potential capacity for way arranging is practical for managing a versatile robot to stay away from a subjectively moving obstruction and to achieve the objective continuously without discovering the nearby most extreme or least focuses in all cases. In this hypothesis, there are two issues. One is that a portable robot quickens quickly when it keeps away from a moving obstruction. The other is that a versatile robot has an irregular speed when it is passing a moving deterrent. A circle field, which is gotten by utilizing the conformal change, and a revision capacity, which produces the consistent speed field, are introduced in the past potential capacity to adapt to the trouble. Accordingly, a portable robot can progressively maintain a strategic distance from a moving hindrance from further away, and can be securely guided without fast increasing speed.
Qiao Liu; Yong-posse Lu et al. [21] proposed a strategy for keeping away from the obstacle for versatile robots. As per their system firstly, the dynamic mathematical statements of Autonomous Mobile Robot were built. Besides, as indicated by the quantity of hindrances, the maintaining a strategic distance from conduct was contemplated, and a path planning taking into account fuzzy logic was additionally proposed. The point between the impediment and the target, and the distance between the deterrent and Autonomous Mobile Robot were inputs of the controller. Taking into account over, a fuzzy controller was utilized to alter the moving heading of Autonomous Mobile Robot. Finally, a genetic algorithm was utilized for streamlining looking of parameters in outline of the controller; the seeking parameters incorporated the 5 times 5 ensuing variables of the control principle table, the base parameters of triangular enrolment capacities and scaling elements.
Another procedure to stay away from impediments when a portable robot is exploring in a semi-organized environment is proposed in view of tangential escape. Such a methodology, the tangential escape, is considered to permit the robot to explore from a beginning stage to a destination indicate without impacting any obstruction in its way. The deviation from any deterrent is performed by break point that makes the new robot introduction digression to the obstacle’s fringe. Above all else, the security of such a control framework is tended to, and the conclusion is that the robot dependably gets any reachable last destination. In the spin-off, two executions of such a methodology are talked about and looked at. In the first the sensorial mechanical assembly locally available the portable stage is a ring of ultrasonic sensors, while in the second one it is a laser sensor that conveys 180 territory estimations covering a semi-circle before the robot [22].
Monica-Claudia, D. et al. [23] proposed a new method to overcome the obstruction avoiding problem for mobile robots. The proposed method permits the robot to add to a neighbourhood self-sufficient obstruction evading conduct each time when a more elevated amount engine charge that is driving the robot (e.g. go ahead/in reverse) place it in up and coming threat to impact. The arrangement we proposed for a robot with 36 equally conveyed infrared (IR) sensors is an extremely basic one, construct just in light of an insignificant artificial neural system (ANN) prepared with a back propagation-like calculation.
Berglund, T.; Brodnik, An et al. [24] concentrated on the issue of programmed era of smooth and snag maintaining a strategic distance from planar ways for productive direction of self-sufficient mining vehicles. In this system a four wheel four-apparatus enunciated vehicles considered and expect that from the earlier learning of the mine divider surroundings as polygonal chains is put away. Processing quartic uniform B-spline curves, minimizing fade and flow variety, staying in any event at a proposed safety edge separation from the mine

dividers, fast ways are arranged. The issue of path planning arrangements for the independent vehicle in environment with moving and stationary obstacles is considered. An algorithm taking into account altered Kohonen guideline and behavioural cloning (MKBC) is produced. The MKBC calculation, as change of RBF neural system, utilizes the preparation values as weighting qualities, rather than values from the past time case. This empowers a wise framework to gain from samples to control a robot vehicle, for this situation, to maintain a strategic distance from stationary or moving hindrance. Critical normal for the MKBC calculation is polynomial multifaceted nature, while most other way arranging calculations are exponential [25].

Portable robots have been broadly utilized as a part of investigation and route to manage the versatile robot moving from the beginning position to the sought objective where static, dynamic and obscure environment is included. The earth is recognized by variable territory and snags that may hinder the development of the robot in coming to the craved destination [2]. Portable robot depends on sensors to get data about their surroundings [3]. Some of these essential practices are Goal Seeking, Obstacle Avoidance and Wall Following. Various inquires about have been completed by adjusting Artificial Intelligence procedures to enhance the execution of the versatile robot route as far as the exactness in staying away from obstructions, briefest way voyaged and add up to time devoured. The Artificial Intelligence systems incorporates Fuzzy Logic (FL) [4-12], Artificial Neural Network (ANN) [13-18], Genetic Algorithm (GA) [19-25], and/or the mix of a couple of them. Every technique has its own quality over others in specific viewpoints. In the most recent decade, hereditary calculations have been generally utilized as option strategy to produce the ideal way by exploiting its solid streamlining capacity. Despite the fact that the current calculations [19-21] have fast pursuit and high inquiry quality, there are six issues connected with the current strategies. Firstly, the starting populace contains numerous infeasible ways, which have negative impact on the execution of the hereditary calculation. Also, there are not adequate heuristic information based hereditary administrators. Thirdly, after every era, offspring's might contain infeasible way. Moreover, Qing Li et al. [19-21] simply control the versatile robot from the beginning position to the craved objective and don't mull over the circumstances when portable robot is caught inside intense "U" or "V" formed obstructions. In the situations when versatile robot experiences intense "U" or "V" formed hindrances, portable robot will be caught and not able to turn out from the caught environment. Fifthly, amid the procedure of directing the portable robot going from the beginning position to the objective position, versatile robot may experience with some dynamic hindrances. The current calculations don't think seriously about these circumstances. Ultimately, there is no reported constant execution to demonstrate the pertinence of the calculation in the real portable robot.

3. PROPOSED MODEL

In this section we discuss about the proposed model. In order to achieve the efficient path planning and obstacle avoiding model, first we develop the path planning algorithm for the autonomous robot.

3.1 Problem formulation

In this section we discuss the problem formulation for an autonomous robot in terms of motion planning and obstacle avoidance.

At the first stage mobile robot is formulated based on the state variables which are position of robot, orientation of robot, linear velocity and angular velocity.

The position of robot is given by (x, y, α) where (α) is the orientation parameter. ω and v represents the angular and linear velocities of the robot. To formulate the problem, angular acceleration and linear acceleration are considered as constant. Angular acceleration constant is denoted as ang_ω and linear acceleration constant is lin_v. In order to get the smooth movement and low tension in robot structure, these parameters are tending to be low.

The control of the robot is done by using command variable for angular velocity and linear velocity which are denoted as cmd_v and cmd_ω. They take discrete qualities {−1, 0, 1} and characterize if, for a given time, the speeds must be decremented, not changed or augmented. The augmentation of the speeds will be given by the acceleration constant.

$$ang(\omega)(t_n) = \omega(t_0) + \int_{t_0}^{t_n} ang_\omega \, cmd_{\omega(t)} \, dt \qquad \text{Eq. (1)}$$

$$lin(v)(t_n) = v(t_0) + \int_{t_0}^{t_n} ang_{linear}\, cmd_{linear(t)}\, dt \quad \text{Eq. (2)}$$

Angular directions and position of the robot is modeled by

$$\alpha(t_n) = \alpha(t_0) + \int_{t_0}^{t_n} \omega(t)dt \quad \text{Eq. (3)}$$

$$\mathcal{X}(t_n) = x(t_0) + \int_{t_0}^{t_n} v(t)\cos(\alpha)(t)dt \quad \text{Eq. (4)}$$

$$\mathcal{Y}(t_n) = y(t_0) + \int_{t_0}^{t_n} v(t)\sin(\alpha)(t)dt \quad \text{Eq. (5)}$$

At any point of time the robot is controlled discretely for a given time interval Δt then these equations can be rewritten as

$$\mathcal{X}(t_n) = x(t_0) + \sum_{i=0}^{(n-1)} \int_{t_i}^{t_i+\Delta t} \left(v(t_i) + \Delta_{t_i}^{t} v\right) \cos\left(\alpha(t_i) + \Delta_{t_i}^{t} \alpha\right) dt \quad \text{Eq. (6)}$$

For linear velocity

$$\Delta_{t_i}^{t} v = ang_v cmd_v(t)(t - t_i) \quad \text{Eq. (7)}$$

$$\Delta_{t_i}^{t} \omega = ang_{v(t_i)} cmd_v(t)(t - t_i) + \frac{1}{2} ang_\omega(t_i) cmd_\omega(t_i)(t - t_i)^2 \quad \text{Eq. (8)}$$

The problem in a velocity space can be formulated by using the angular velocity command and linear velocity command.

$$\langle (cmd_{v_{t0}}, cmd_{\omega_{t0}}), (cmd_{v_{t1}}, cmd_{\omega_{t1}}) \ldots . (cmd_{v_{tn}}, cmd_{\omega_{tn}}) \rangle \quad \text{Eq. (8)}$$

The formulated problem is able to cause in the movement of the robot from initial position to the destination position.

3.2 Kinematic model

The kinematic model for proposed robot navigation under rolling and non-slipping constraints is gives as

$$q_1 = v_1 \times \cos\theta \quad \text{Eq. (9)}$$

$$q_2 = v_1 \times \sin\theta \quad \text{Eq. (10)}$$

$$q_3 = v_2 \quad \text{Eq. (11)}$$

The above given equations presents the mathematical model for robot kinematics, in these equations longitudinal velocity is denoted by v_1 which is applied to the wheel. v_2 is the angular deflection which is provided to the robot wheels. v_1 and v_2 depends on the velocity constants which are denoted by ρ and σ. ρ is the velocity constant for forward velocity and σ is the angular constant for angular deflection of the wheel.

$$q = \begin{bmatrix} a_x \\ b_y \\ \alpha_c \end{bmatrix} \quad \text{Eq. (12)}$$

$$\hat{q} = \begin{bmatrix} \widehat{a_x} \\ \widehat{b_y} \\ \widehat{\alpha_c} \end{bmatrix} \quad \text{Eq. (13)}$$

q is the origin position of the robot and $\hat{q}$ is the reference position of the robot. Position of center of mass is denoted by a_x, b_y along the X and Y co-ordinate.
The velocity is denoted by V

$$V = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} v_1 \\ \alpha_c \end{bmatrix} \quad \text{Eq. (14)}$$

The start position of robot is given as (x, y, α) and the target points are given as (x_t, y_t, α_t) with respect to reference plane of the trajectory.

3.2 Path planning

Figure 1 depicts the path planning problem. A three wheeled omnidirectional robot is considered. Suppose that the robot at the starting point A will go to the target point B. the shortest path is the straight line (AB). This line is called the desired path or DP. There are static obstacles and moving obstacles near or on the desired path. The robot has to reach point B while avoiding collisions with obstacles. The strategy studied here will be a planning motion along the direction of the DP, while avoiding obstacles perpendicularly to the DP.

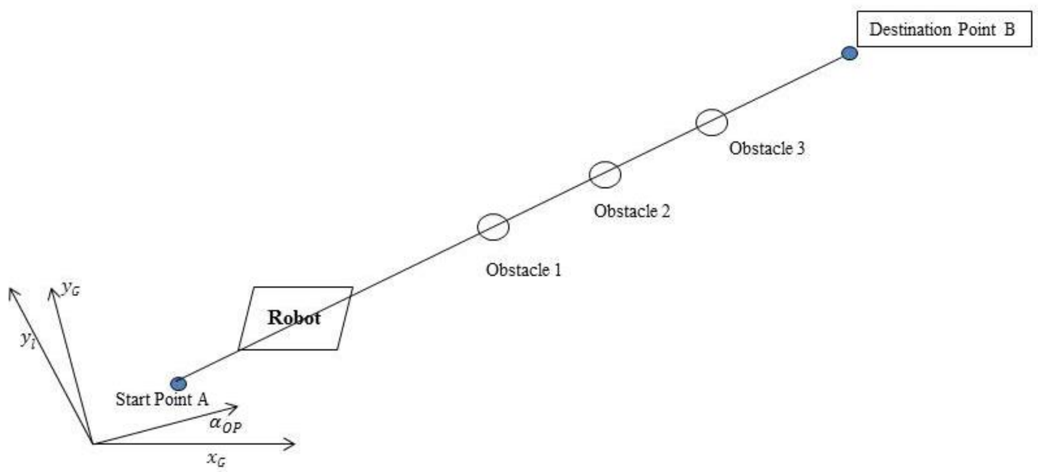


Figure 1. Path Planning and Obstacle Avoidance

As shown in the figure 1 x_G , y_G and o are the global fixed coordinate for the plane.
Point A is the beginning stage and point B is the destination purpose of the robot. The straight line AB denotes to the Optimal Path (OP) of the robot movement. The nearby facilitate framework is taken to be $x_l o y_l$, which has the same cause point as the worldwide direction framework and its hub is parallel to the OP.
The beginning stage and the destination point and the other movement conditions for the robot are constantly changed to nearby coordinates. The position and the speed of the robot are figured in nearby organizes and afterward changed back to global directions to shape the robot way. The purpose behind having OP directions is that arranging the movement in neighborhood directions is more clear than in global directions, as the movement along the x_l course is pre-arranged, while the movement in the y_l heading will be dictated by the given calculation of hindrance evasion in nearby coordinates.

Vol.3 Issue 12, Pg.: 66-79
December 2015

The angle α_{OP} of local coordinate axis x_{local} and global coordinate x_{global} is denoted by

$$\alpha_{OP} = atan2(y_b - y_a, x_b - x_a) \qquad \text{Eq. (15)}$$

In above given equation $atan2$ is the inverse tangent value for the four quadrant which is varied from $[-\pi, \pi]$. The position of the global coordinates A and B is given by $X_{globalA} = (x_a, y_a)$ and $X_{globalB} = (x_b, y_b)$. The position of robot or obstruction can be computed in the optimal by equation 8

$$X_{local} = Rot_{local}^{global} X_{global} \qquad \text{Eq. (16)}$$

Rot_{local}^{global} is the rotational matrix which can be denoted as:

$$Rot_{local}^{global} = \begin{bmatrix} cos\alpha_{OP} & sin\alpha_{OP} \\ -sin\alpha_{OP} & cos\alpha_{OP} \end{bmatrix} \qquad \text{Eq. (17)}$$

Computation of robot velocity or obstacle velocity can be performed by using equation (10)

$$\hat{X}_{local} = \widehat{Rot_{local}^{global}} X_{global} \qquad \text{Eq. (18)}$$

$\hat{X}_{local}$ and X_{global} is the velocity of local and global coordinates.

3.3 Proposed Model for local coordinates

Optimal path for local coordinates is computed using 2nd order polynomial function. In order to compute optimal path coordinates, the time interval for initial position of robot and the destination position is given by Δt_A and Δt_B and can be computed as

$$\Delta t_A = \Delta t_B = \dot{X}_{lrob} / \ddot{X}_{lrob} \qquad \text{Eq. (18)}$$

$$\Delta t_c = \dot{X}_{lrob} / \ddot{X}_{lrob} \qquad \text{Eq. (19)}$$

Distance between source point and destination can be computed as

$$d = \frac{1}{2} \ddot{X}_{lrob} t^2 \qquad \text{Eq. (20)}$$

By substituting (19) into (20) gives

$$\Delta X_A = \Delta X_B = \ddot{X}_{lrob} / 2\ddot{X}_{lrob} \qquad \text{Eq. (21)}$$

$$\Delta X_{cD} = D - 2\Delta x_{cA} \qquad \text{Eq. (22)}$$

By solving equation (18) and (22)

$$t_B = \frac{D}{x_{lB}} + \dot{x}_{lrob} / \ddot{x}_{lrob} \qquad \text{Eq. (23)}$$

Total distance t_D to be travelled by robot to avoid the obstacle can be computed as $t_D = t_B - \Delta t_{BD}$.
The motion of robot according to the optimal path is calculate as :

$$x_l - x_{lA} = \int_{t_A}^{t_B} \dot{x}_l dt = \int_{t_A}^{t_C} \ddot{x}_{lrob} t\, dt + \int_{t_C}^{t_D} \dot{x}_{lrob} dt - \int_{t_D}^{t_B} \ddot{x}_{lrob} t\, dt \qquad \text{Eq. (22)}$$

So,

$$x_l - x_{lA} = \begin{cases} \frac{1}{2}\ddot{x}_{lrob} t^2 & t < t_C \\ \frac{\dot{x}_{lrob}^2}{2\ddot{x}_{lrob}} + \dot{x}_{lrob}\left(t - \frac{\dot{x}_{lrob}}{\ddot{x}_{lrob}}\right) & t_C \le t < t_D \\ D - \frac{\dot{x}_{lrob}^2}{2\ddot{x}_{lrob}} + \dot{x}_{lrob}\left(t - \frac{D}{\ddot{x}_{lrob}}\right) - \frac{1}{2}\ddot{x}_{lrob}\left(t - \frac{D}{\ddot{x}_{lrob}}\right)^2 & t_D \le t < t_B \end{cases} \qquad \text{Eq. (23)}$$

3.4 OBSTACLE AVOIDANCE MODELLING

The essential technique of obstruction evasion is to utilize the idea of the Configuration Space. The fundamental thought of Configuration Space is to take a robot and its workspace, and decrease the robot to a solitary point, while extending the hindrances in the workspace to make note of the state of the robot. The first workspace is changed into Configuration-Obstacles and Configuration-Free space. The request of this thought is that issues of path planning or of situating a robot or article inside of a workspace are lessened to issues including a solitary point. It is less demanding, for instance, to manage the crossing point of a solitary point with Configuration-Space obstructions, instead of with convergences of impediments and robot in a Cartesian workspace.

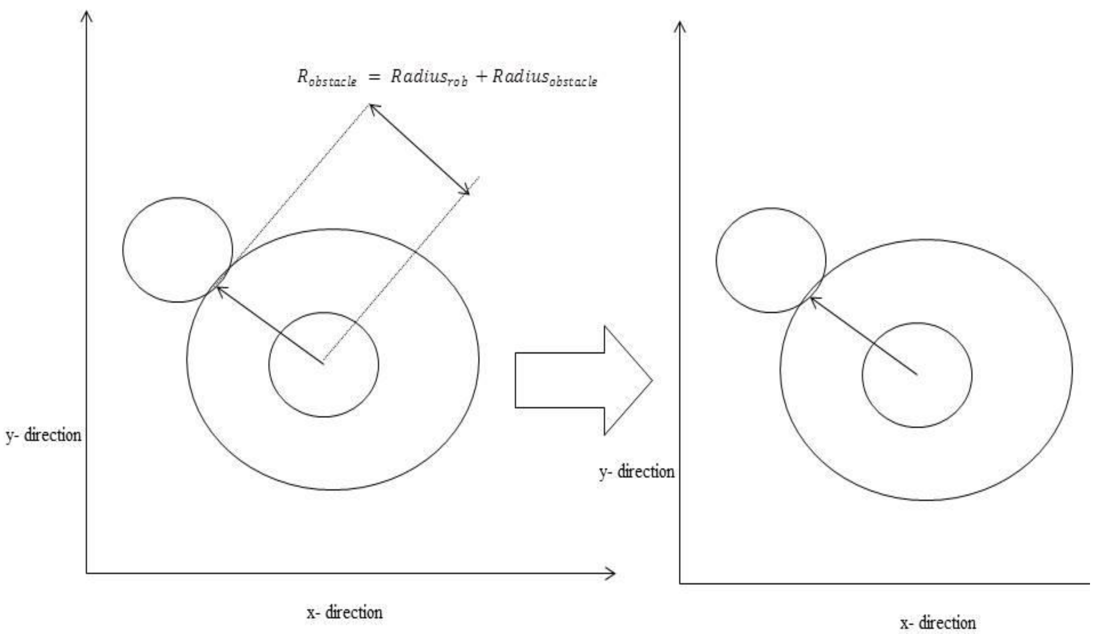


Figure 2. Simplification of Obstacle

The above given figure is the representation of obstacle modeling and simplification. According to the proposed model obstacle is modeled as

$$R_{obstacle} = Radius_{rob} + Radius_{obstacle} \qquad \text{Eq. (24)}$$

The robot is then considered as a point. A robot being able to avoid an obstacle is equivalent to the point robot being capable of avoiding the expanded obstacle.

In our proposed model the robot is considered as a point and it is able to avoid to the obstacle which is equal to the robot and then it is capable to avoid the incremented obstacle. Figure 3. Shows the overall flow diagram for the proposed model

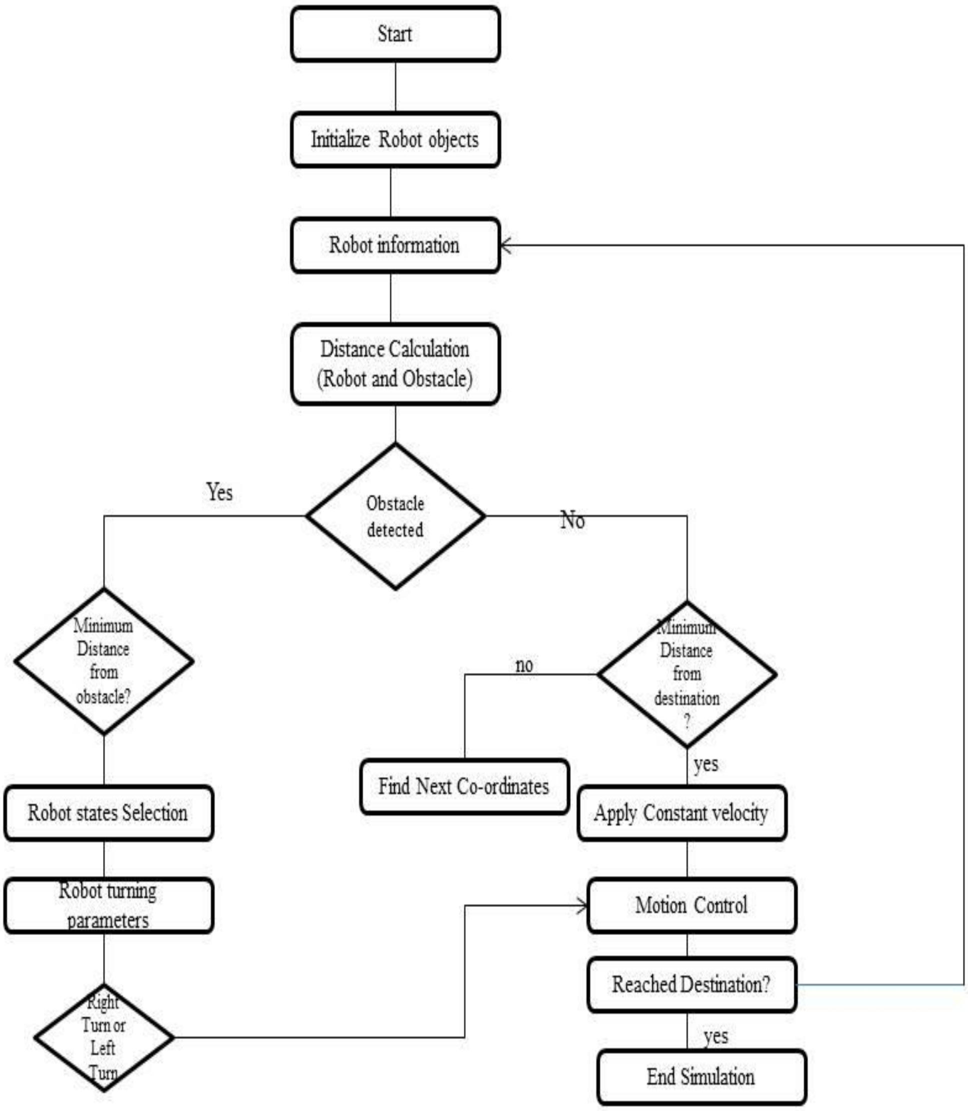


Figure 3. Overall Architectural Flow Chart of the Proposed Module

DISTANCE COMPUTATION FOR COLLISION

The distance of checking crash D_{thres} can be characterized as limit for identifying impediments. At the point when the relative separation with all impediments is longer than D_{thres} , the likelihood of crash with deterrents is not checked, and the robot will continue proceeding onward the ideal way until its relative separation with a obstacle is not as much as or equivalent to D_{thres} , where the robot begins checking conceivable impacts with obstructions.

Expect the greatest speed x_{lobs} of a obstruction in x heading is the same as the most extreme pace of the robot x_{lrob} . For the situation when the deterrent moves towards the robot along the ideal path, both at their most extreme speed, the relative velocity between a robot and an obstruction can be composed as:

$$\dot{x}_{lobs.rob} = \dot{x}_{lrob} - \dot{x}_{lObs} = 2\dot{x}_{lrob} \qquad \text{Eq. (25)}$$

$\dot{x}_{lobs.rob}$ is the representation of relative velocity between robot and obstruction

$\dot{x}_{lrob}$ is the notation of relative velocity of robot

$\dot{x}_{lObs}$ gives the relative velocity of obstacle.

Maximum velocity and acceleration of the robot in y direction is given as y_{lrob} and $y1_{lrob}$ respectively, then the time taken to reach the destination d_{exp} can be computed as

$$\Delta t = \begin{cases} \sqrt{\left(\frac{2d_{exp}}{y1_{lrob}}\right)} \\ \frac{d_{exp}}{y_{lrob}} + y_{lrob}/2y1_{lrob} \end{cases} \quad \text{Eq. (26)}$$

Returning to the Actual path
Relationship between the actual path and robot, to retrieve the actual path can be given as

$$Path_{Actual} = y_{ldir} - y_{ladir} \quad \text{Eq. (27)}$$

Where y_{ldir} and y_{ladir} refer to the coordinates of the robot and the starting point A in direction respectively. After reaching to the returning point, robot starts to increase or decrease the velocity to return to the desired path. This velocity is in y- direction which can be written as

$$y_{ldir} = \begin{cases} y_{ldir} + y_{lrob}\Delta t_s \\ y_{ldir} - y_{lrob}\Delta t_s \\ y_{ldir} \\ 0 \end{cases} \quad \text{Eq. (28)}$$

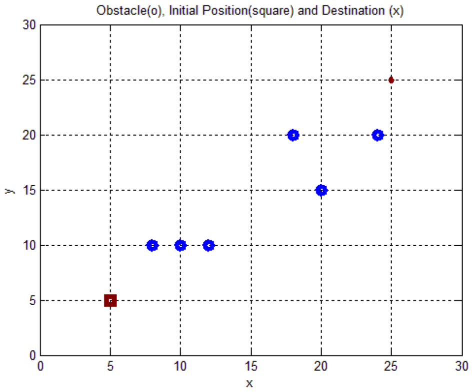
4. Result and Discussions.

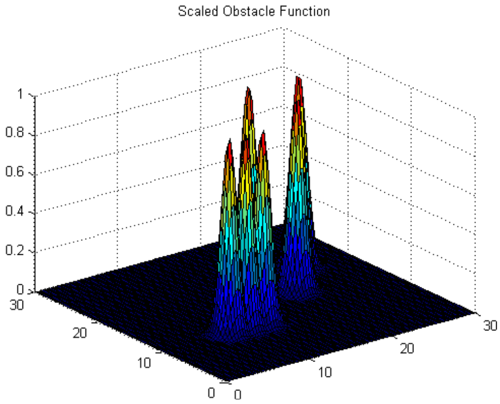
In this section, we discuss about the performance of the proposed autonomous robot navigation algorithm. The framework environment utilized is windows-7® undertakings 64-bit working framework with 8GB of RAM. We have utilized MATLAB tool to simulate programming dialect for the proposed work and directed trial study on taking after parameter for obstacle avoidance proficiency and path planning and less time consumption.

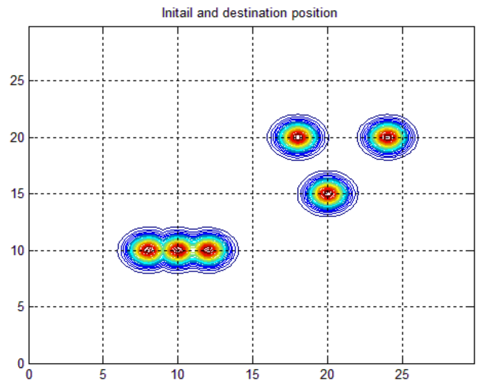
During implementation, our first step is to develop a region in which obstacles are placed to check the performance of the proposed model. For this region we have considered x and y coordinate as (0, 30) and (0, 40) in the first case. To achieve the destination, we have considered 500 iterations with the step size of 0.1. Autonomous robot has a sensing range of 1 meter in the predefined region. In this range it can sense 16 obstacles for each iteration.

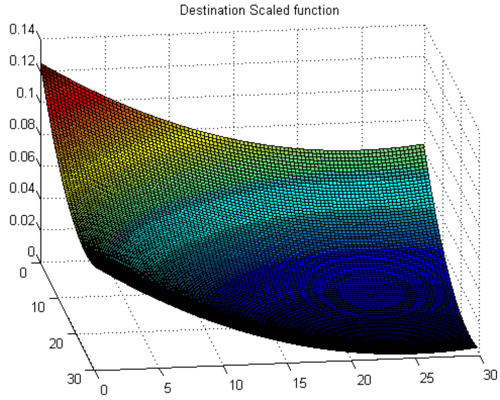
Angle of the obstacles is computed when it starts moving until it reaches to the destination points. In the below given figure 4, obstacle avoidance and path planning results are depicted. According to the first step, source and destination positions are assigned for robot to follow the path. In the second step obstacles are placed in the path as discussed in earlier section. These obstacles are placed with respect to the x and y coordinates as shown in Fig..4 (a). The obstacles are identified by the robot based on the obstacle function; this is based on the sensing factor of the robot which is shown in Fig..4. (b). Fig..4.(c) represents the contour representation of the initial and destination positions of the robot. Fig..4.(d) shows the scaled goal function for the autonomous robot to achieve the destination. By using this scaling of goal function the contour function of the source and destination is achieved which is shown in Fig..4.(e). Similarly contour is computed for the obstacles for planning the path and to avoid the obstacles. Based on the iteration the robot's performance in x-direction and y-direction is given in Fig. 4.(f) and final path to avoid is shown in Fig..4.(g) . By utilizing all these resources, the designed robot model is able to avoid the obstacle and able to reach at the destination in the less time.

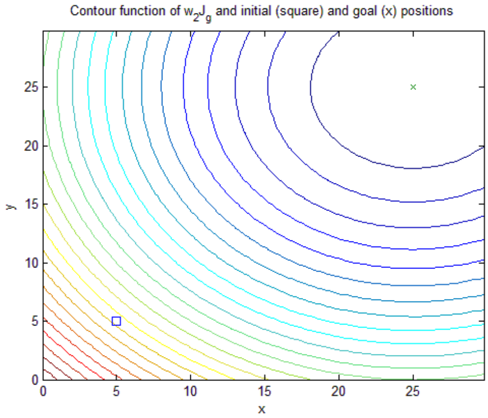
Vol.3 Issue 12, Pg.: 66-79
December 2015

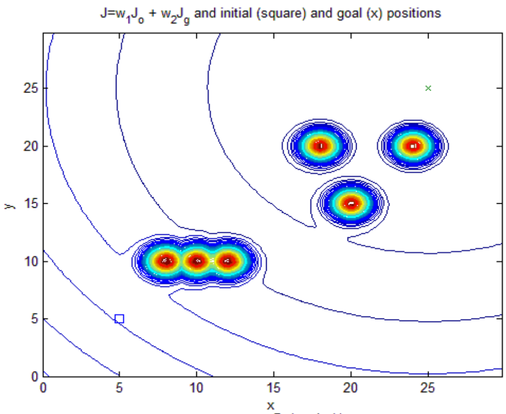


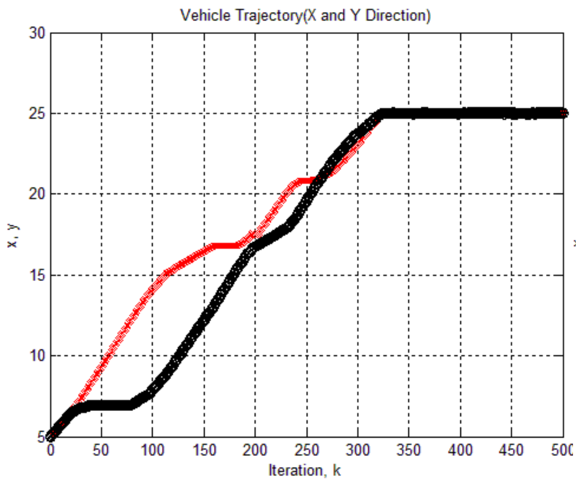


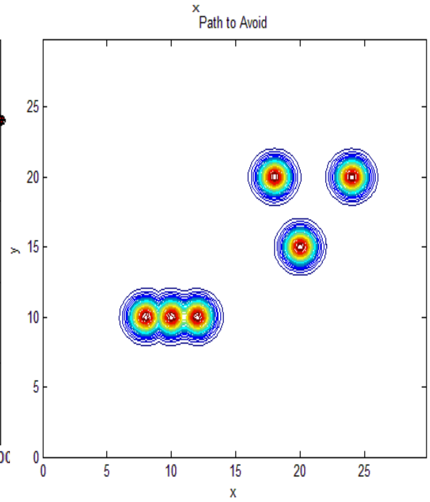












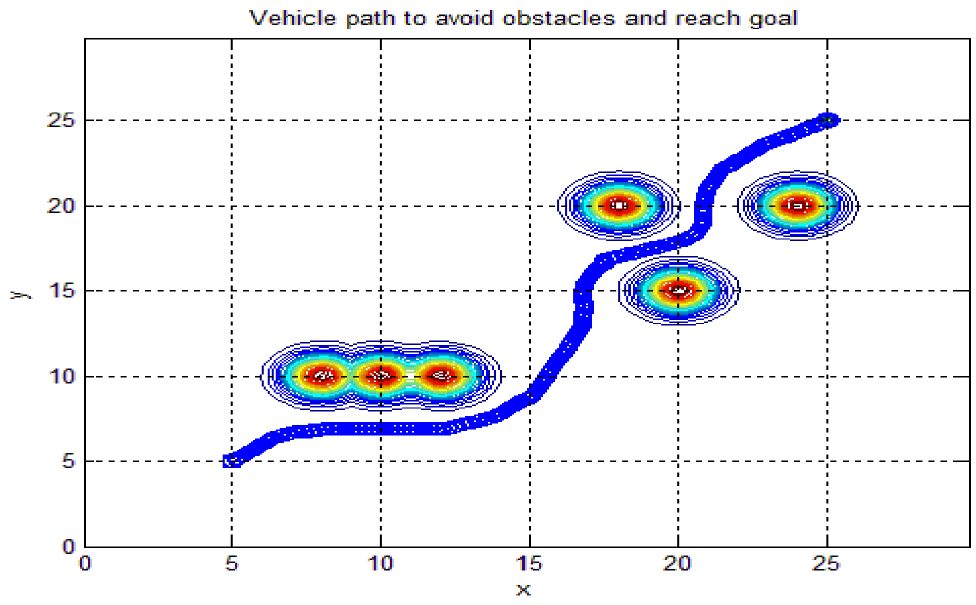


Figure 4. Path planning and obstacle avoidance of the proposed model

During implementation of case 2, our first step is to develop a region in which obstacles are placed to check the performance of the proposed model. For this region we have considered x and y coordinate as (0, 50) and (0, 50) in the second case. To achieve the destination, we have considered 600 iterations with the step size of 0.1. Autonomous robot has a sensing range of 1 meter in the predefined region. In this range it can sense 16 obstacles for each iteration.

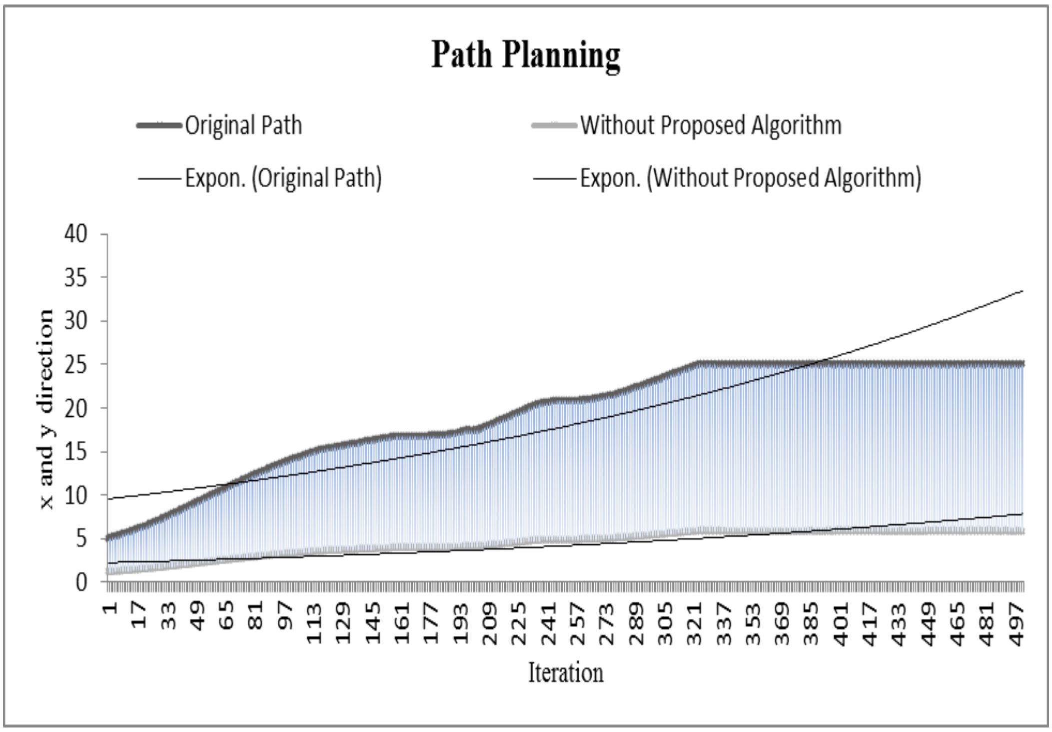


Figure 5. Path following performance in x and y direction without applying proposed algorithm

Above given figure 5 shows the performance of the autonomous robot without applying the proposed algorithm. Without proposed algorithm the robot path varies when it encounters any obstacle and gives the error path which is shown exponentially in the figure.

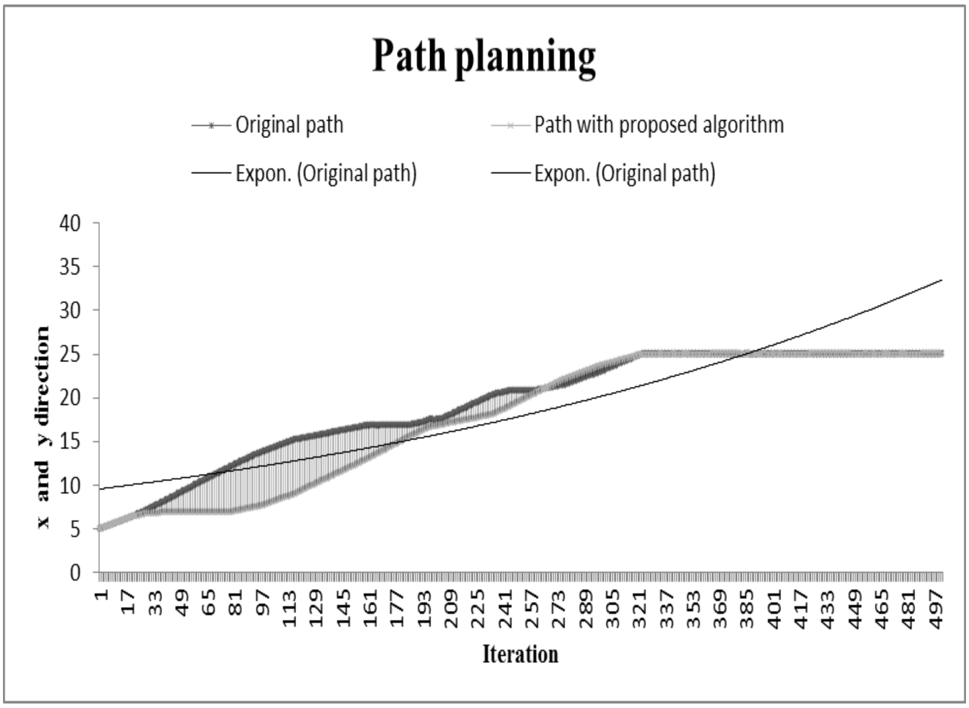


Figure 6. Path following performance in x and y direction with proposed algorithm.

Similarly, above given figure 6 show the performance of the proposed path planning algorithm with respect to original path. By applying the proposed algorithm, the error induced due to the obstacles is removed and it follows the original path efficiently.

4. Conclusion

In this work a new approach is developed for the efficient navigation of the robot. The proposed method is the hybrid model which comprises global approach for motion estimation along with the actual path and 2nd order of polynomial functions. For avoiding the collisions, distance measurement module is developed based on the gaps between the obstacle and robot. Different simulation results were performed to show the capability of the proposed algorithm. It was found that the proposed algorithm for navigation can be adaptable to any kind of complex environments.

Experimental results, carried out in simulated environments, demonstrate that our approach can be positively affective in mobile robot navigation for different kinds of robots and sensors, when compared to previous works. Therefore, our proposed globalized navigation algorithm can yield significant navigation results – at less training time and lower sensor costs.

REFERENCES

[1] Chaomin Luo, Jiyong Gao, Xinde Li, Hongwei Mo, Qimi Jiang, "Sensor-based autonomous robot navigation under unknown environments with grid map representation," in IEEE Symposium on Swarm Intelligence(SIS), 2014, pp. 1-7, 9-12 Dec. 2014

[2] Sariff, N., Buniyamin, N., "An Overview of Autonomous Mobile Robot Path Planning Algorithms," in 4th Student Conference on Research and Development(SCOReD) 2006., pp. 183-188, 27-28 June 2006

[3] Y. Hu and S. X. Yang (2004), “A knowledge based genetic algorithm for Path Planning of a mobile robot”. Proc. of IEEE Intl. Conf. on Robotics and Automation, pp. 4350-4355.

[4] J. C. Latombe, “Robot motion planning” Kluwer Academic Publishers Norwell, MA, USA 1991, ISBN:079239206X.

[5] T. Lozano-Perez, “Spatial planning: A configuration approach,” IEEE Trans. on Computers, Vol. C-32, No. 2, pp. 108-120, Feb 1983.

[6] M. Sharir, “Algorithmic motion planning in robotics,” IEEE Trans. on Computer, Vol. 22, No. 3, pp. 9-19, March 1989

[7] P. Khosla, R. Volpe, “Superquadric artificial potentials for obstacle avoidance and approach,” in Proc. of IEEE Intl. Conf on Robotics and Automation, Vol. 3, Philadelphia, PA, USA, April 1988, pp. 1778-1784.

[8] E. Rimon, D. E. Doditschek, "Exact robot navigation using artificial potential fields," IEEE Trans. on Robotics and Automation, Vol. 8, Issue 5, pp 501–518
[9] S. X. Yang, M. Meng, "An efficient neural network approach to dynamic robot motion planning," Proceedings of the 7th IEEE International Workshop on Neural Networks, Vol. 13, pp. 143- 148, 2000.
[10] S. X. Yang, M. Meng, "An efficient neural network method for real-time motion planning with safety consideration," Robotics and Autonomous Systems, Vol. 32, pp. 115-128, 2000.
[11] Xiyang Song, Huangwei Fang, Xiong Jiao, Ying Wang, "Autonomous mobile robot navigation using machine learning," in IEEE 6th International Conference on Information and Automation for Sustainability (ICIAfS), 2012, pp.135-140, 27-29 Sept. 2012
[12] P. Fiorini, Z. Shiller, "Motion Planning in Dynamic Environments Using Velocity Obstacles", in Int. Journal of Robotics Research, Vol. 17, No. 7, 1998, pp. 760 – 772
[13] D. Wilkie, J. Van den Berg, D. Manocha, "Generalized Velocity Obstacles", IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2009.
[14] A. Wu, J P. How, "Guaranteed infinite horizon avoidance of unpredictable, dynamically constrained obstacles", in Autonomous Robots Vol. 32, No. 3, 2012, pp. 227-242
[15] D. Fox, W. Burgard, S. Thrun, "The Dynamic Window approach to obstacle avoidance", in IEEE Robotics and Automation Magazine, Vol. 4, No. 1, 1997, pp. 23–33.
[16] B. Damas, J. Santos-Victor, "Avoiding Moving Obstacles: the Forbidden Velocity Map", IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2009
[17] M. Seder, I. Petrovic, "Dynamic window based approach to mobile ´ robot motion control in the presence of moving obstacles", IEEE Int. Conf. on Robotics and Automation, 2007
[18] C. Fulgenzi, A. Spalanzani, C. Laugier, "Probabilistic motion planning among moving obstacles following typical motion patterns", in IEEE/RSJ Int. Conf. on Intelligent RObots and Systems, 2009.
[19] Yaghmaie, F.A. Mobarhani, A. Taghirad, H.D., "A new method for mobile robot navigation in dynamic environment: Escaping algorithm," in First RSI/ISM International Conference on Robotics and Mechatronics (ICRoM), 2013, pp.212-217, 13-15 Feb. 2013
[20] Sugiyama, S., Yamada, J., Yoshikawa, T., "Path planning of a mobile robot for avoiding moving obstacles with improved velocity control by using the hydrodynamic potential," in IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS), 2010, pp.1421-1426, 18-22 Oct. 2010
[21] Qiao Liu, Yong-gang Lu, Cun-xi Xie, "Fuzzy Obstacle-avoiding Controller of Autonomous Mobile Robot Optimized by Genetic Algorithm under Multi-obstacles Environment," in Intelligent Control and Automation, 2006
[22] Ferreira, A., Pereira, F.G., Bastos-Filho, T.F., Sarcinelli-Filho, M., Carelli, R., "Avoiding Obstacles in Mobile Robot Navigation: Implementing the Tangential Escape Approach," in IEEE International Symposium on Industrial Electronics, 2006, Vol.4, pp. 2732-2737, 9-13 July 2006
[23] Monica-Claudia, D., Marius, D.D., "A bio-inspired solution for a local autonomous, reflex, obstacle avoiding behavior," in 10th International Symposium on Signals, Circuits and Systems (ISSCS), 2011, pp.1-4, June 30 2011-July 1 2011
[24] Berglund, T., Brodnik, A., Jonsson, H., Staffanson, M., Soderkvist, I., "Planning Smooth and Obstacle-Avoiding B-Spline Paths for Autonomous Mining Vehicles," in IEEE Transactions on Automation Science and Engineering, , Vol.7, No.1, pp.167-172, Jan. 2010
[25] Kulic, R., Vukic, Z., "To avoid unmoving and moving obstacles using MKBC algorithm Path planning," in IEEE International Conference on Mechatronics, ICM 2009., pp.1-6, 14-17 April 2009, pp. 726-736

A Brief Author Biography

Najmuddin Aamer – Completed his Bachelor's in Electronics & Communication Engineering from KBN College of Engineering, Gulbarga University, Gulbarga in 2001 and Master's in Information and Communication Systems from Malnad College of Engg, Hassan, Visvesvarya Technological University, Belgaum in 2005. Currently pursuing Ph.D. From Vinayaka Missions University, Salem, Tamil Nadu, India. His research areas include Mobile Robot Path Planning Using Soft Computing Techniques, Artificial Intelligence, Neural Networks, Reconfigurable Architecture's, FPGA and CPLD.

Dr. Ramachandran S **–**Currently working as a Professor in Electronics & Communication Engineering Department, SJB Institute of Technology, Bangalore. Received his M.tech from Indian Institute of Technology (IIT, Kanpur) in 1979 and Ph.D. from Indian Institute of Technology (IIT, Madras) in 2001. His researches interests VLSI / FPGA Design, Video Processing, Medical Electronics, Controls, and He has a number of papers in international journals and conferences and he has wide academic as well as industrial experience of over 30 years. He has also written a book on Digital VLSI Systems Design, published by Springer Verilog, Netherlands.