Safe Path Planning of Mobile Robot in a known Dynamic Environment

Hasan Mujtaba, Gajendra Singh, Pallavi Gupta

Abstract: Path planning in mobile robot navigation is an advanced method of calculating the safe and obstacle free path in static and dynamic environments are involved between source point to destination. Real time path planning method defines that how a robot can make a decision when some unknown obstacle gets encountered in the path of navigation for a dynamic situation. At the point when an obstruction comes in the way of route, the robot must choose another and safe way to advance towards the objective by evading any impact. This study is focused on exploring the algorithm that gives the safe and shortest path when an obstacle changes the environment. By using A* algorithm in MATLAB simulation the probability of collision with obstacle and robot get increased. In this simulation work a new approach of path planning has been found by placing the virtual obstacles in the environment. A new obstacle get influence in the path of navigation, using virtual obstacle boundary around the new obstacle a short and safe path get evaluated which is collision free or low risk path . The purpose for this paper is to create a dependable and smooth direction in a real time domain with impediments and to manage the robot towards the target without hitting the obstacles also considering the size of the robot.

Keywords: Safe navigation, A* algorithm, Path planning, Virtual obstacles.

I. INTRODUCTION

The path planning algorithm has a very significant role to plan an appropriate path from source to destination. There are implicit and explicit methods are available for path optimization [1]. Implicit methods specify the desired and dynamic behavior of the robot like potential field algorithm. The disadvantage of this approach is that local minima of the potential field function trap the robot in U shaped obstacle [2]. Explicit methods provide the robot path between the initial point and final target. Explicit methods focus on finding discrete collision-free environment between the start and goal location. These methods consist mainly of two components of algorithms, the road-map method that takes account of the visibility graph method, voronoi diagram roadmap algorithm [3] and cell-decomposition methods [4].

The very first example of search-based planning was Dijkstra's Algorithm, which was simply a Breadth-First search [5]. In 1968, the A* algorithm was introduced which uses an 'Admissible Heuristic' to narrow the search [6]. As long s a better-informed heuristic is not used, the *A** algorithm will find the solution as fast as or faster than any other method. *A** is a static algorithm, which means that when the configuration space changes such as when an obstacle or the perception of an obstacle changes, the old path is validated and the *A** algorithm must be re-run from scratch [6]. In 1992, Christos Alexopoulos and Paul M. Griffin [7] considered two problems for path planning of a mobile robot. The first problem was to find the shortest-time, collision-free path for the robot in the presence of stationary obstacles in two dimensions. The second problem was to determine a collision-free path for a mobile robot in an environment of moving obstacles. The environment was modelled in space-time and the collision-free path was found by the variation in A* algorithm. But it has not defined the safe navigation of mobile robot with safe distance "Dsafe" from the obstacle. In 2007, M. Fu and B. Xue[8] proposed one kind of A* algorithm based on restricted searching area (RA*) to compute the fastest path which simplifies the discrete-time dynamic network model according to the real character of the urban traffic system. They also analyzed the performance of RA* and validates its efficiency with contrast to other routing planning algorithms. But it also not validates the safe planning of mobile robot. In 2011, Chia-Jun Yu et al.[9] proposed a path planning method for autonomous mobile robots in a known indoor environment. A traditional A* algorithm modified by a weighted cost function was proposed. The factors of distance and safety were considered simultaneously in the cost function so that the path planning can let the mobile robot reach its goal safely and quickly. Some simulation results were presented to illustrate the proposed method has a good path planning for mobile robots. The proposed method has also been implemented and tested on a real mobile robot. The experiment results illustrate that the proposed method can let the autonomous mobile robot have a safe path-planning in a known environment. Jong-Hun Park and Uk-Youl Huh[10] in 2016 has developed extensive simulations that the developed SGPP (safe global path planning) method outperforms the classical A* and PRM (probabilistic roadmap) algorithms. It improves the time and length of the robot path in comparison to classical A* Algorithm.

II. PROBLEM STATEMENT

The existing algorithms has following drawbacks
(1) The existing algorithm doesn't consider unknown obstacle i.e: dynamic obstacle encounter in the path of mobile robot.

(2) It doesn't consider about the versatile size of a robot. If the robot is bigger it is hard to travel through thin sections or passages
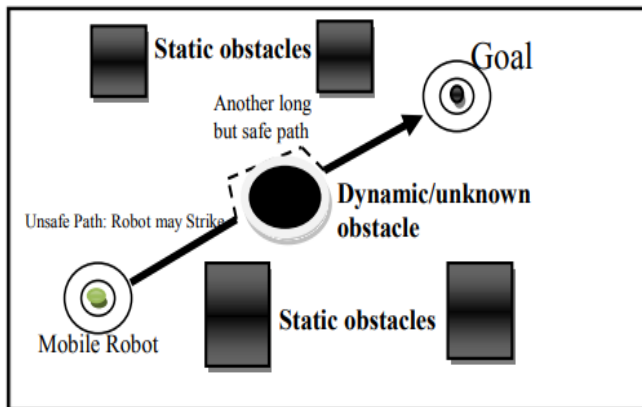

Fig. 1.Example of Problem Statment

A real challenge for the mobile robot in real time path planning is to find the shortest path from the start point to the destination in presence of static as well as dynamic obstacles. In the presence of static obstacles environment, the mobile robot moves around the obstacle and reaches the goal. This path has to approach towards the minimum distance cost or in other words it should be the shortest potential distance. A* is the shortest distance calculating algorithm that uses predefined search technique to join the least-cost points from the start point to the destination. A* path planning algorithm has been used in guiding the autonomous mobile vehicle for direction control to optimize the shortest distance from source to destination. It is useful to recognize the algorithm conceptually. So it can appreciate to use more effectively to calculate the path in static or dynamic arena. The basic function of A* algorithms are including with heuristic function $H(n)$ i.e:

$$F(n) = G(n) + H(n) \tag{1}$$

$n = (X_n, Y_n)$ is the current expanding node or cell
$F(n)$ = This function represents the minimum expense of the cell or node between every one of the path from source S to destination G
$G(n)$ = G cost is the actual development value of the present path from the start node S to the nth node.
$H(n)$ = The heuristic estimation of minimum cost of a path from the current node to the goal node G.
The addition of $H(n)$ in basic function, enable the further information about the graph, makes this algorithm exceptionally efficient for a node to surrounding nodes queries. In robotics, A* is mainly applicable on the grid based graph. Heuristic value $H(n)$ is to calculate the distance between the present cell to the goal cell by taking into account all the obstacles in closed set. If such information about the graph is stored in algorithm, it can be used to recalculate the path towards the goal node or destination. This concept of reducing the number of current expansions node required to arrive at the easier solution compared to the Dijkstra's algorithm or other algorithms. For finding the heuristic functions used for the grid environment, some famous heuristics distances are Manhattan distance, diagonal distance or Euclidean distance. Here for calculating shortest distance and avoiding the obstacles for the fixed arena, we are using the Manhattan distance or Euclidean distance to optimize $H(n)$. Because our purpose is to save the robot from

static or unwanted obstacles to find out the shortest and safest path on the squared grid. The concept of virtual obstacle also introduced for avoiding the dimensions of AMRV.

III. CONCEPTS AND PROPOSED METHODOLOGY

The concept behind this project to identify an efficient path which involves to optimize the multiple objective functions in the path planning of a mobile robot .This experimental work gives a smooth and a real time path planning approach towards the target without hitting the obstacles also consider the size or area of the mobile robot. The proposed system also considers those obstacles which come suddenly in the path of robot known as dynamic obstacle. The Proposed methodology involves smart solution for navigating a mobile robot in an indoor environment. The steps includes following

STEP A: Shortest Path Planning Approach
STEP B: Concept of Virtual Obstacles
STEP C: Concept of Safe Distance (D_{safe})
STEP D: Unknown Obstacles Algorithm
STEP E: Simulation Result and Calculation

 A.: Shortest Path Planning Approach: This consider the minimum distance calculation using A* path planning algorithm for calculating the minimum cost of entire path. But the main drawback is it does not consider the size of a mobile robot and robot can be damage by unsafe edges.

B: Concept of Virtual Obstacles : To avoid the damage of a mobile robot from static obstacle a new approach has been developed by putting the line of virtual obstacle around the static obstacles in my previous work .

C*: Concept of Safe Distance (D_{safe})*: In my present work the above two approach already being included with the concept of D_{safe} is involved which consider the margins of safe distance from the static obstacles.

D: Unknown Obstacles Algorithm: This is the fourth step which consider the unknown obstacles or dynamic obstacles that encountered in the path of a mobile robot or which changes the environment, this modified algorithm states that the robot has to be stop when an obstacle comes in the path of of robot and it will generate a new path by refreshing the algorithm as shown in the simulation result Fig9

E: Simulation Result and Calculation: The final steps involve the simulation results and calculations, Fig 4 and Fig 5 shows the results of shortest distance calculation and the unknown obstacles in the path of mobile robot without considering the size of robot, Fig 8 and Fig 9 shows the simulation results by putting the virtual obstacles around the static obstacles which validate that if the size of robot get increases it will be safe. The calculation part consider few assumption like horizontal and vertical distance of each cell , calculation of Euclidian and Manhattan distance, on that assumption the calculation of parameter like distance , velocity and acceleration shown in Table I and Table II , which verify that there is no incremental changes in the above parameters by putting the virtual obstacles around the static obstacles.

IV. SAFE DISTANCE OR MARGIN (D_{SAFE})

A safe distance or margin is all about the current position of an AMR in the free space that ensure its safe distance from the surroundings of static or unwanted obstacles on the grid, it also execute to all the possible margins that would allow to maintain the minimum safety distances of the robot from the obstacle found in the path of source to destination. Finally, the safe distance or margin contains that minimum D_{safe} required for stopping of the mobile robot to prevent its collision with an static or if there are the encounter of some new or unknown obstacle. A safe distance margin (D_{safe}) configuration is generated in order to ensure the passive safety of a mobile robot in the path planning towards the goal. This passive safety is obtained by considering the single virtual cell around the static or dynamic obstacle. To optimize this concept of virtual obstacle in the free space or environment, there is some parameters has to be taken under consideration. These parameter free space configuration (C), of a mobile robot in two dimensional (2D) spaces on the grid of 50x50 matrix divided into two parts ie: free space configuration (C_{free}) and obstacles space configuration (C_{obs}). The virtual obstacles are introduced in the surrounding of static or dynamic obstacle that is (C_{vobs}).
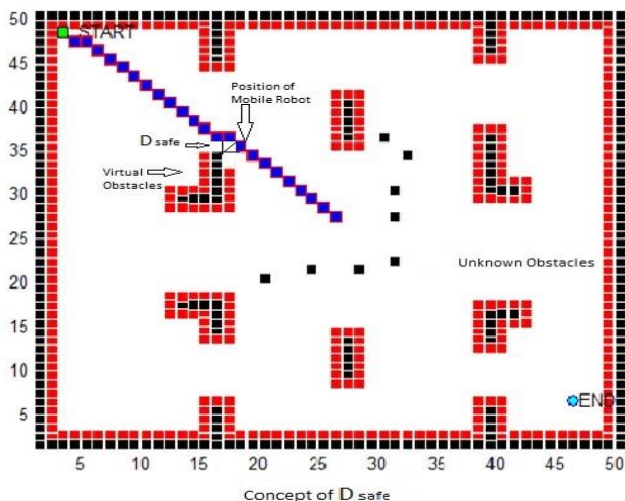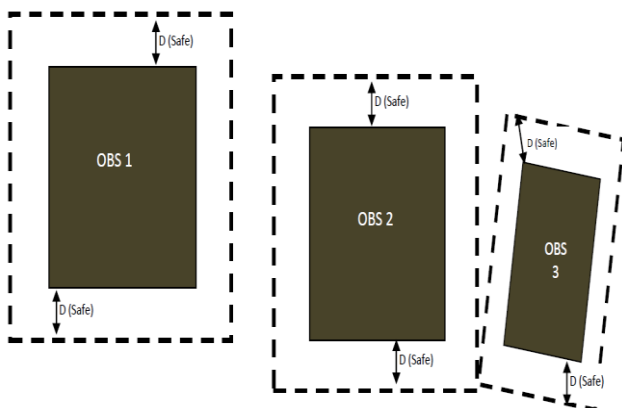

Fig. 2.Dynamic Obstacles in Environment -

V. CONCEPT OF D_{SAFE}

To prevent the collision of the robot with unidentified objects encountered in the corner obstacle the concept of safe distance is required.

Fig. 3.Concept of D(safe)

Let consider the Configuration space of mobile robot is $C(confg)$ then the $C=C_{free} + C_{obs}$
$$C_{obs} =C_{static} +C_{virt}$$
Now let consider the $R(m)$ is the current position of mobile robot. $C_{virt}= \{m \varepsilon C: R(m) \cap C_{obs} \neq \phi\}$
Let n is the no obstacles
$$C_{free} =C- U_{(1\ to\ n)}C_{virt}$$
Now consider risk space in C (Workspace)
Assumption1
Vm =Maximum speed of robot $=D_B/T$
D_B= Distance travelled by robot
Ar =Acceleration of robot= $Vm /T = Vm^2 / D_B$
Mobile robot finds the unsafe corner in the path
C_{virt} (The safe space)
Assumption 2
$Vm(virt)$ =Maximum speed of robot $=D_{virt} /T$
$D(virt)$= Distance travelled by robot after
putting the virtual obstacle.
Av=Acceleration of robot= $Vmv /T = Vmv^2 / D_B$
On the basis of above analysis a new algorithm has-been developed we consider the dynamic obstacle in the environment.
Assumption 3
$Vm(virt)> V_{obs\ (Dynamic)}$ it means considering that the maximum speed robot greater than the dynamic obstacle So the environment get refreshed and a new path has been developed .An algorithm for checking the new obstacle found in the work space. Putting the virtual obstacle.

VI. UNKNOWN OBSTACLE ALGORITHM 1 FOR AMRV

a. Q= For stop the robot
b. R= For run the robot
c. Consider the Step no18. path is created by back tracking method
d. Press Q
e. Create new obstacle=N_{OBS}
f. Create virtual obstacle=N_{VIRT}
g. N_{OBS} ε Closed list
h. N_{VIRT} ε Closed list
i. If $N_{OBS=}$Goal node (G)
j. Go to step no19
k. Otherwise go to step no5 in algorithm2

VII. UNKNOWN OBSTACLE ALGORITHM 2 FOR AMRV

In the proposed method, following steps has been used
1. Create a map grid or nodes.
2. Consider virtual obstacles around the static
3. Create "open list" and "closed list" in reset condition
4. Recognize the entire static& virtual placed obstacle.
5. Place begin point and end point
6. IF Goal (G) is ε open list, then go to step no: 18 go to next step
7. Calculate the F(n) cost of adjacent cells .
8. Pick the smallest value of F(n) from open list "S_k" move it into close list ={S_k}
9. IF "S_k" ε Goal (G) go to step no: 18otherwise go to next step

2436

10. Set Q=1.
11. IF Q ε close list ∪ obstacle list ∪ virtual list then go next step otherwise go to step no15.
12. Set Q= Q+1
13. IF Q<= 8 then go to no.11otherwise go to next
14. Close = Close ∪{S_k}, Open list =Open -{S_k} and go to no.8
15. IF Node(Q) ε open list the go next step otherwise go to no.16
16. Update F(n) value and go to no.12
17. Place it in "open list" and calculate F(n) value got step no 12
18. Calculate the path using back tracking method
19. Stop

VIII. SIMULATION RESULTS

The MATLAB simulators adopted for the mobile robot. Result shows the drawback of A* algorithm. There is the maximum probability is to hit with the static obstacle. In Fig4 there is no any unknown obstacle in the environment the robot move in the fixed static obstacle environment. On the basis of result found in the static environment some parameters of the mobile robot has been evaluated in terms of Distance (Ɛ), Execution time (ms), Velocity (V) and Acceleration (ω) of practical vehicle. These parameter are useful to calculate other parameter like Deviation angle, Radius of wheel etc.
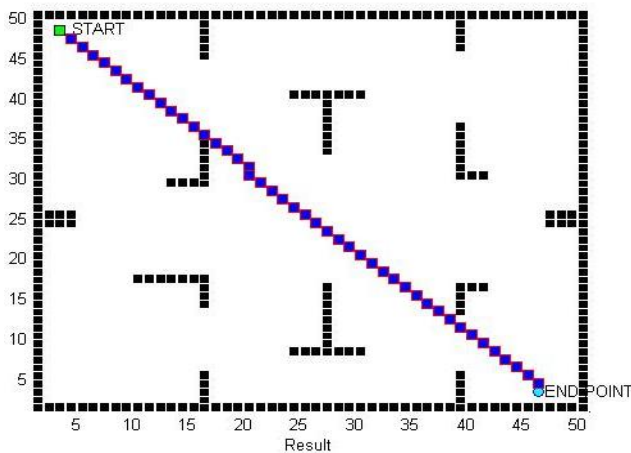

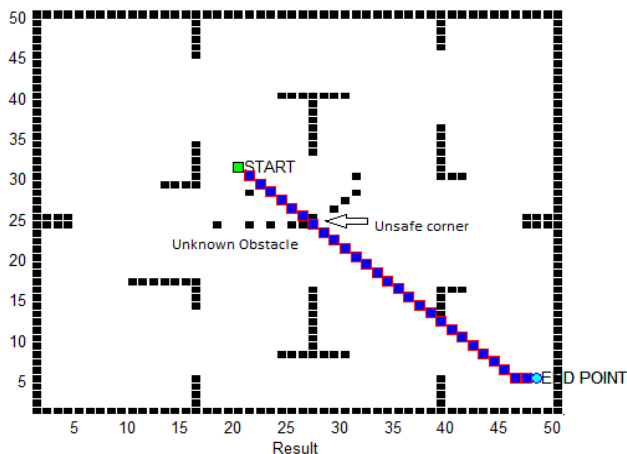**Fig. 4.Simulation Result of A* algorithm**

Fig. 5.Simulation result with dynamic obstacles

IX. CALCULATION OF TRAVELLED DISTANCE

(a)Calculation of Euclidian (Ɛ) distance

Let consider the *assumption1* general consideration of mobile robot as the result shown in Fig4 and Fig5 for the calculation of let consider Euclidian (Ɛ) distance travelled by the mobile robot in given Fig6
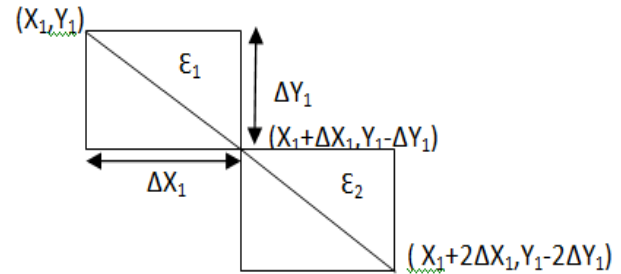, consider the two travelled cell in enhanced form of diagram.

Fig. 6.Calculation of Euclidian Distance

Let consider start point of mobile robot is (X_1,Y_1) or (X_g, Y_g) are the goal position. In x direction robot travel ΔX distance and in Y direction ΔY distance in this condition if robot move to another cell the coordinate in x and y direction are $(X_1+\Delta X_1)$ and $(Y_1-\Delta Y_1)$ the distance travelled by the robot $Ɛ_1$.

$$Ɛ_1=\sqrt{(X_1 + \Delta X - X_1)^2 + (Y_1 - \Delta Y - Y_1)^2}$$
(2)
$$Ɛ_1=\sqrt{(\Delta X)^2 + (\Delta Y)^2}$$
(3)
Similarly
$$Ɛ_2=\sqrt{(X_1 + 2\Delta X - X_1)^2 + (Y_1 - \Delta Y - Y_1)^2}$$
(4)
$$Ɛ_2=\sqrt{4(\Delta X)^2 + 4(\Delta Y)^2}$$
(5)
From equ(1) and equ(5)
$$Ɛ_2=2\sqrt{(\Delta X)^2 + (\Delta Y)^2} = 2 Ɛ_1$$
Hence the total Euclidian distance
$$Ɛ_i = \sum_{n=1}^{N} n Ɛ_1$$
(6)

(b)Calculation of Manhattan distance

In Fig7 there is Manhattan distance travelled by the robot

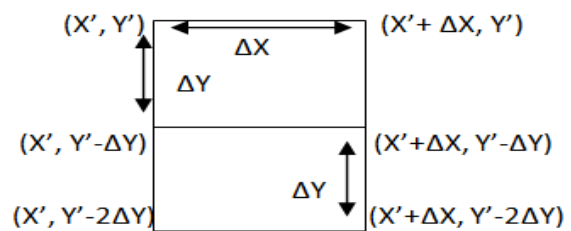**Fig7.Calculation of manhattan distance**

Manhattan distance=$Ɛ'=\{Y' -(Y'-2\Delta Y)\}$ (7)
$Ɛ'=2\Delta Y$ then $Ɛ_m =\sum_{m=1}^{M} mƐ'$ (8)
Hence the total distance
$Ɛ= Ɛ_i+ Ɛ_m$ (9)

In the second part of *assumption2 and 3* by placing virtual obstacle around the static obstacles the concept of safety space in unknown environment have been introduced for evaluating the proposed method some experiment has been done. In Fig.8 we are placing only virtual line around the static obstacles
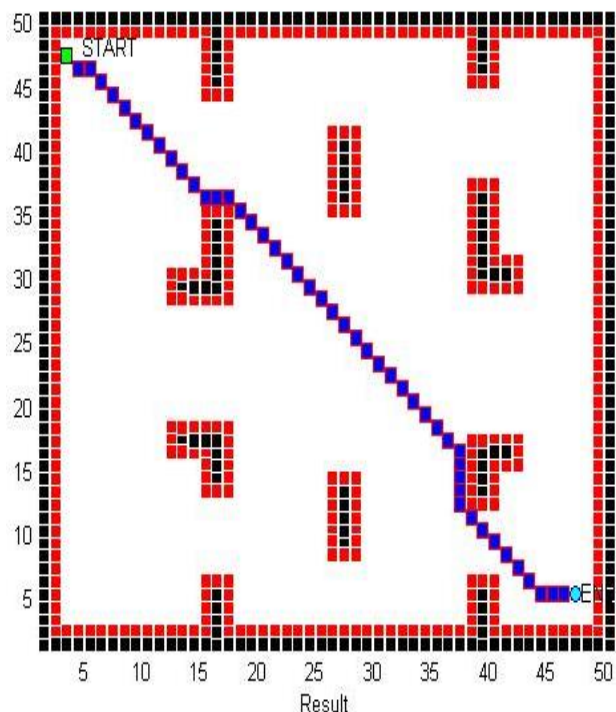

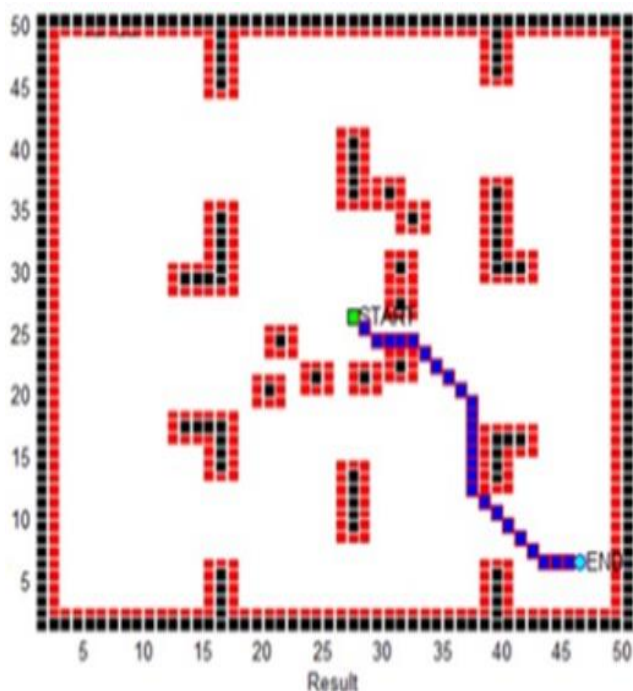**Fig8. Simulation result with vitual obstacle environment**

Fig 9. Placing vitual obstacle in dynamic environment

Let consider $\Delta X = 1cms$ and $\Delta Y = 1cms$ of each cell. In the above analysis mobile robot moves with constant velocity and acceleration. On placing the virtual obstacle and

influencing the new obstacle in the environment the due to increment of virtual cells in the manhattans distance get increased so it will take longer time to reach the destination which reduced the velocity and the acceleration of robot but the path is more secure compare to the result found in Fig4and 5

Table- I: Results with static obstacles

Sr.No	Distance(d) cms	Time(sec)	Velocity	Acceleration
1	60cms	17sec	3.5 cm/s	0.2 cm/s^2
2	62cms	18sec	3.5cm/s	0.2cm/s^2

Table- II: Results with virtual and dynamic obstacles

S.No	Distance(d) cms	Time(sec)	Velocity	Acceleration
1	64cms	20sec	3.2cm/s	0.16 cm/s^2
2	70cms	23sec	3.04cm/s	0.13cm/s^2

X. CONCLUSION

In this work, we have studied different path planning methods; some are good in terms of shortest path and some are good in terms of safest path. Every method has its own advantages and limitations. The Voronoi Diagram gives the safest path but it cannot search the shortest path similarly the Visibility Graph search gives the shortest path but it cannot search the safest path. For planning the path of the mobile robot, it is necessary to consider both the requirements. For this purposed A* algorithm was used which can be applied in the approximate cell decomposed map. In this strategy the arena of the robot is adjusted by putting virtual lines around the obstacles. These virtual obstacles are practically free space but the robot cannot move in this space because this space is treated as obstacle to it. After the execution of the proposed algorithm it has been observed that the execution time has increased when we place the virtual obstacle and change the environment by new obstacle considered as dynamic obstacles. It conclude that the proposed research the for dynamic environment the speed of mobile robot must practically slower due to the safety issue but it must be required that it should be greater than the speed of dynamic obstacles.

REFERENCES

1. Elena Garcia, Maria Antonia Jimenez, Pablo Gonzalez de Santos and Manuel Armada "The Evolution of Robotics Research" Industrial Robotics to Field and Service Robotics IEEE Robotics Automation Magazine IEEE Robotics & Automation Magazine ,Volume14 ,Issue 1, Pages90 - 103April 2007.
2. O.Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," Int. J. Robot. Res., vol. 5, no. 1, pp. 90–98, 1986.
3. J.F. Canny, The Complexity of Robot Motion Planning, Cambridge, MA:MIT Press, 1988.
4. J.T. Schwartz and M. Sharir, "On the 'piano movers The case of two-dimensional rigid polygonal body moving amidst polygonal barriers," Commun. Pure Appl. Math., vol. 36, pp. 345–398, 1983.
5. Cormen, T., Leiserson, C., Rivest, R., and Stein, C "Introduction to Algorithms" Second Edition. MIT Press and McGraw-Hill, 2001.

6. Hart, P. E., Nilsson, N. J., and Raphael, B. (1968), "A Formal Basis for the Heuristic Determination of Minimum Cost Paths", IEEE Transactions on Systems Science and Cybernetics, pp. 100-107.
7.   Christos Alexopoulos and Paul M. Griffin (1992), "Path Planning for a Mobile Robot", IEEE transactions on Systems, Man and Cybernetics, vol. 22, no. 2, pp. 318 – 322.
8. Fu, M. and Xue, B. (2007), "A Path Planning Algorithm Based on dynamic Networks and Restricted Searching Area", IEEE International Conference on Automation and Logistics, pp. 1193-1197.
9. Chia-Jun Yu; Yi-Hong Chen; Ching-Chang Wong (2011), "Path Planning Method Design for Mobile Robots", Proceedings of SICE Annual Conference, pp.1681 – 1686
10. Jong-Hun Park and Uk-Youl Huh"Path Planning for Autonomous Mobile Robot Based on Safe Space"Journal of Electr Eng Technol.2016, pp. 1921-718

AUTHORS PROFILE

Hasan Mujtaba did his B.sc Engineering in ECE from magadh university in year 2002. He has done his M.Sc Engineering form NIT Patna and currently pursuing his Ph.D. in ECE from Sharda University Greater Noida. He has total Experience of 13 years in Academic.

Dr. Gajendra Singh presently working as dean academics at ABES-EC Ghaziabad .He did his B.Tech. in Electrical Engineering from University of Roorkee (now IIT, Roorkee) in 1976. He has done his M.Tech from IIT, Kharagpur and Ph.D. from IIT Delhi.

Dr. Pallavi Gupta presently working as an Associate Professor in ECE Department of Sharda University, Greater Noida. She did her B.Tech. in Electronics and Communication Engineering and M. Tech in Digital Communication & Ph.D. in Electronics Engineering from Aligarh Muslim University. She has total Experience of 13 years in Academic & 2 years of research.