CHAPTER 5: CONCLUSION

Autonomous mobile robotic systems are employed in structured environments in industrial applications and unstructured/unconfined environments in applications such as defence, mining, hospitals, etc. Efficient path planning algorithms are required for structured, off-line situations as well as unstructured, on-line situations. In off-line path planning situations, path length or travel time or energy consumed or combinations of such parameters may be considered as optimization criteria. In on-line path planning situations even though the optimization criteria may be the same viz. path length or travel time, what is critical is the computation time required to plan a collision-free path segment. Therefore, this work fulfills the twin objectives of developing effective off-line path planning algorithm which gives the shortest path and developing computationally efficient on-line path planning algorithm based on a comprehensive mathematical model of collision-avoidance considering kinematic and dynamic constraints of mobile robot.

5.1. OFF-LINE PATH PLANNING

For a given known environment, in order that the mobile robot reaches its target, an initial population of valid alternative paths is generated. A path is valid if it does not interfere with any obstacle (static or moving). In the process of generating the required size of population, an invalid path interfering with a stationary obstacle is converted into valid one by gliding along the edges of the amplified obstacle. In case of an invalid path interfering with a moving obstacle, it is omitted instead of attempting to convert into valid path as it will cause undue delay in initial population generation itself.
The valid paths are then subjected to evaluation function to calculate its fitness. PSO technique is used by the algorithm to select collision-free optimal paths. The evaluation function is designed to have shortest path length as an objective. The proposed algorithm is computationally efficient because of quick convergence of solution due to assured quality of initial population. The effectiveness of the proposed algorithm has been shown by taking into account a variety of environments and also by comparison of routes obtained with GA in place of PSO. The obstacles in the environment are designed to contain convex, concave and curved obstacles. Results show that the algorithm using PSO delivers marginally shorter path lengths. More importantly, in terms of computation time over 29% improvement has been achieved when compared to GA.

In addition, the performance of the algorithm is compared with a published literature (Wang et al.) for computational efficiency using four simulated environments as presented in chapter 3. The environments are designed to contain multiple obstacles which are sparsely located or cluttered. Comparative performance of the algorithm shows that a minimum of 1.3% reduction in path length and a maximum of 8.9% reduction in path length are achieved for sparse and cluttered dynamic environments respectively. More importantly, the computation time required by the proposed algorithm is only around 1% of the time required by Wang et al. algorithm. Also, the path obtained by the proposed algorithm has less number of path segments. The comparison of results show that the proposed algorithm is very effective for cluttered dynamic environments.
Also, the ruggedness of the algorithm is demonstrated using Monte Carlo simulation trials. Five different tasks are designed for each of the four simulated environments to check collision-avoidance. In all the 20 cases, resulting from 5 different tasks in 4 different environments, target is reached successfully negotiating the interfering obstacles. After 1000 trials each in the 20 different situations, only around 1% deviation is found in optimal path length. Thus the ruggedness of the algorithm is proved.

The salient features of the proposed off-line path planning algorithm which yields better quality paths are listed below:

1. An open initial search space without limiting to the vertices of the enlarged obstacles results in better paths with minimum path segments.

2. Generation of valid paths using direction concept without using any special genetic operators results in reduced computation time.

3. Modelling of path is done with variable number of segments at the time of population generation. Path length, in terms of number of discrete points, can be 2 at the minimum and $n + 2$ at the maximum where $n$ is the total number of vertices of all the obstacles in the environment. Thus, the path is modelled according to the varying complexity of environments.

4. Particle Swarm Optimization technique with real strings is used by the algorithm for choosing the shortest path. Real coded formulation of the problem is much
easier than the binary representation. This eliminates the need for coding and decoding procedures resulting in improved computational efficiency.

5.2. ON-LINE PATH PLANNING

An effective on-line path planning algorithm for mobile robots in dynamic environments has been developed. A comprehensive mathematical model is established which considers all the current on-line information of robot as well as nearing obstacles. The proposed algorithm combines the mathematical model for collision-avoidance and evolutionary PSO technique to plan an optimal collision-free path satisfying both kinematic and dynamic constraints of robot. The effect of constraining obstacles is also taken into consideration while negotiating the most imminent obstacle. The proposed algorithm does not require any separate recovery mode approach to escape from trap situations. The algorithm is applicable to environments having moving targets also.

The effectiveness of the proposed algorithm has been demonstrated by considering a variety of environments, as presented in chapter 4. Similar to off-line situations, considered in chapter 3, here also the environments have been designed to be sparse in two cases and cluttered in the remaining two cases containing obstacles of different sizes.

In addition, the performance of the algorithm is compared with Min et al. algorithm for computational efficiency using the same four simulated environments. Comparative performance of the algorithm shows that the proposed algorithm is computationally more efficient and effective in reaching targets along shortest possible paths. Around 30%
reduction in path length and 70% reduction in computation time are achieved for environments with cluttered dynamic obstacles.

Also, the ruggedness of the algorithm has been demonstrated using Monte Carlo simulation trials. Five different tasks are considered for each sparse and cluttered simulated environment to check collision-avoidance while successfully reaching the respective targets. Results such as computation time per instant of the robot and its standard deviation are presented for 1000 simulation trials for all the 20 cases. In terms of path length as well as computation time, less than 1% deviation has been achieved in all the cases.

The salient features of the proposed on-line path planning algorithm to yield shorter paths in quicker time are summarized below:

1. An accurate mathematical model has been developed considering instantaneous velocity of robot as well as nearing obstacles to negotiate cluttered moving obstacles.

2. The influence of other constraining obstacles while negotiating the most imminent obstacle is also included in the model to take care of further deviation required instead of merely avoiding the imminent threat.

3. Kinematic and dynamic constraints of robot are also built in the mathematical model.
4. The need for trap recovery is eliminated as the path is planned tangential to the collision zone.

5.3. FURTHER SCOPE

Computationally efficient off-line path planning as well as on-line path planning algorithms have been developed for mobile robots in dynamic environments. However, there is scope for further improvements. For off-line path planning, the objective may have multiple optimization criteria with varying weightages for improving quality of paths. For on-line path planning, this work can further be extended to applications involving multiple robots moving towards different targets or same targets, which can also be moving.