MPRI-JERT | INTERNATIONAL JOURNAL OF ENGINEERING & RESEARCH TECHNOLOGY
VOL. 10 NO.5 DEC-2019 ISSN: 1074-4741

ROBOT PATH TRAILING IN AN INTERMEDIATE ENVIRONMENT USING A COMPUTATIONAL MODEL

***A. T. SALAWUDEEN **I. F. EGBUJO **M. L. IMAM *N. S. USMAN **K. C. ZAKKA **P. J. NYABVO**

**Department of Computer Engineering, University of Ahmadu Bello University, Zaria. **Department of Electrical and Electronics Engineering, University of Jos, Jos, Plateau State.*

Abstract

This paper presents an optimal path planning technique for a mobile robot in a dynamic environment. One of the major problems in robotics engineering is motion planning. Since robots are not intelligent and have no intelligent means of perceiving its environment, it usually fundamental to develop an obstacle free path capable of guiding the robot from a goal point to a target point. Therefore, this paper presents a computational base model for robot path planning in a dynamic obstacle environment. The simulation scenario was formulated with a [redacted] *constrained robot position and an unknown obstacle position. Simulation were carried out in Matlab R017a and results showed that the proposed method is efficient in comparison with previous techniques.*

KEYWORDS: *SAO, SDA, Robot, Dynamic Environment, Path Planning*

Introduction

Robot path planning problem is a typical example of an oversimplified real-world problem. Given a starting and final position of a robot, the path planning problem is to seek for a set of motions which will enable the robot to move between the two positions without colliding or been obstructed or stopped by an obstacle [1]. The robot has to identify the optimal path and be able to locate the final position (target) in the shortest amount of time. This is usually a difficult task to achieve since the robot does not have an intelligent means of identifying the nature of the environment. Researchers have over the years employed various CI algorithms [2-4] for optimal path planning. However, selecting a CI technique which will provide the optimal path planning when compared with all other algorithms has always been a critical issue [5, 6]. This is because there is no know CI algorithm optimized for path planning only.

Robot motion or path planning which is also called the "Piano mover's problem" is a process of breaking down the desired movement into some discrete motions that satisfy certain movement constraints and optimize some performance criterion [7]. Developing practical control algorithms for these kinds of problems have been very challenging due to the complexities involved. Mobile robots are primarily employed for repetitive tasks, program execution and some distance waypoint navigation [8]. When an autonomous robot navigates from an initial point to a destination point in an environment, it is necessary to plan a feasible path avoiding obstacles in its way while satisfying some criterion of autonomy requirements such as thermal, energy, time and safety [9]. Due to these complexities, the piano mover problem search environment has been classified into the static and dynamic environment [10].

Static environment

In this environment, the positions of the obstacles are fixed and the robot already has the prior knowledge of these obstacles as it trails the pre-planned path hoping to avoid a collision. Since the environmental information is available to the robot, the robot can plan the optimal path a priori.

Dynamic environment

In the dynamic environment, the nature and conditions of the obstacles are not known. In most cases, some of the obstacles are in a fixed position while some are moving in a random direction. This usually makes a collision-free movement of the robot in this environment very difficult.

In this paper, we consider an intermediate environment where the initial positions of the robot, goal and obstacles are generated randomly at the initial stage. As the robot explore the search the position its position is updated while that of the obstacle remain fixed. We formulated a path minimization cost function which we optimized using Particle Swarm Optimization, PSO [11] and Smell Detection Agent, SDA [12].

The rest of the paper are organized as follows: Section II presents the methods employed for successful implementation of the proposed method. Details of results are discussed in section III while section IV presents conclusion.

INTERNATIONAL JOURNAL – ERT
VOL. 10 NO.5 DEC-2019 ISSN: 1074-4741

METHODS

In this section, the methods which we employed in formulating the path planning objective function was discussed.

A. Objective Funtion Formulation

The path planning objective function which we adopted in this paper is given in eq. (1) [13]

$$f(x) = \min \sum_{i=1}^{N} D(\Delta x, \Delta y) \qquad (1)$$

Where $D(\Delta x, \Delta y)$ is Euclidean distance between every coordinate position of the robot in the optimization hyper space. Details on the formulation of this objective function can be found in [14]. Eq. (1) serves as evaluation criteria for the optimality of the solution obtained from the PSO and SDA. The modelled environment showing the randomized positions of the robot, goal and obstacles is given in Fig. 1

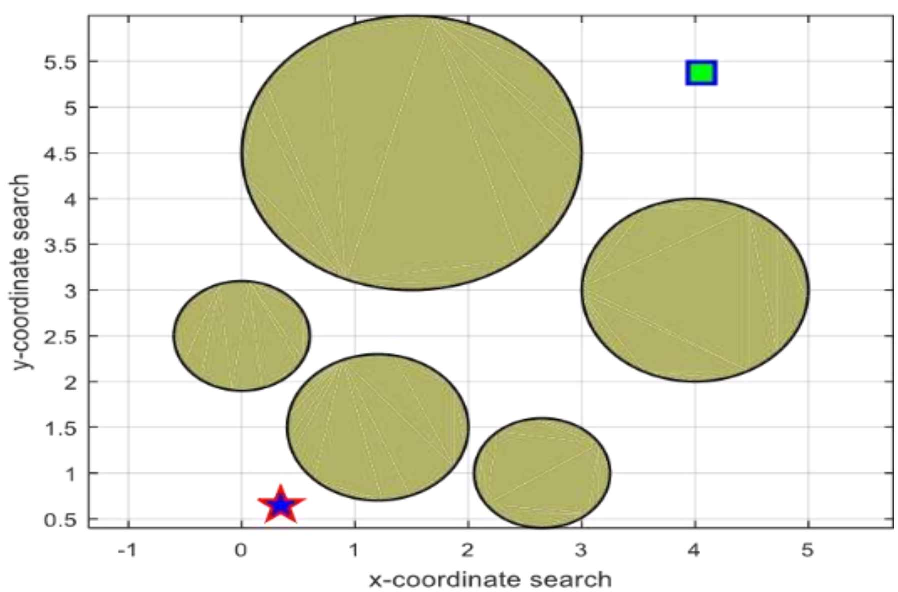

Fig. 1: Path Planning Environments

Fig 1, is modelled to mimic the practical representation of the intermediate environment. In the figure, the robot is represented as the blue star shape with red outline. The environment also contains five circled obstacles made of different diameters and the goal is represented as the green square with blue outline. In the environment the position of the robot, the goal and the obstacles vary every time the PSO and SDA algorithms are run. In each case, the position of the robot is such that, there is no direct line of site between the robot and the goal, and the robot is always positioned below the obstacles while the goal is always positioned above the obstacles. This way, the complexity of the search environment during robot exploration is made as complex and realistic as possible.

The coordinate positions of the robot (x_{robot}, y_{robot}) were randomly generated and the position is restricted by the constrain given in eq. (2)

$$0 \le x_{robot} \le 1$$
$$0 \le y_{robot} \le 1$$
(2)

Just like the robot, the position of the goal was also generated randomly and is restricted by the constrain given in eq. (3 - 4)

$$3 \le x_{goal} \le 5$$
(3)
$$4 \le y_{goal} \le 6$$
(4)

The number of obstacles were then initialized and each obstacle coordinate is given in the Table 1.

Table 1 Coordinate positions of obstacles

Coordinate	Obstacle 1	Obstacle 2	Obstacle 3	Obstacle 4	Obstacle 5
$x_{obstacle}$	1.5	4.0	1.2	2.65	0
$y_{obstacle}$	4.5	3.0	1.5	1.0	2.5

Since the obstacles considered in this paper are circle, the radius of each obstacle was selected as 1.5m, 1.0m, 0.8m, 0.6 and 0.6m for Obstacle 1, Obstacle 2, Obstacle 3, Obstacle 4 and Obstacle 5 respectively. The planning cost function is then optimized until the termination criteria is met.

The PSO and SDA parameters given in Table 2 were adopted for the path planning simulation. However, the number of search dimension or decision variables were selected to be 3. This is purely the choice of the researcher as the decision variable influence the number of points the robot should evaluate at every iteration.

Nevertheless, more decision variables may improve the path planning cost, but have negative influence on the time the robot takes to reach the goal. The parameters used for simulating the path planning model along with the SAO parameters is given in Table 2:

Table 2 Algorithms Parameters

SN	Parameters	Value	Unit
1	Population	250	-
2	Dimension	3	-
3	Iteration	500	-

Table 2 shows some of the parameters used during the algorithm's simulation on the developed model of the robot path planning. These parameters were selected carefully by the according to the problems under consideration. Since there is no empirical parameter established in literature for solving path planning problems, these parameters can be varied to improve the optimality of the planned path.

RESULTS DISCUSSION

The path planning environment contains a total of five obstacles of different sizes strategically placed, such that, the robot has no direct line of sight with the goal. This makes it trivial for the smell agent algorithm to plan an optimized robot path easily. The optimum path obtained by the PSO and SDA are shown in Fig 2 and Fig 3 respectively.

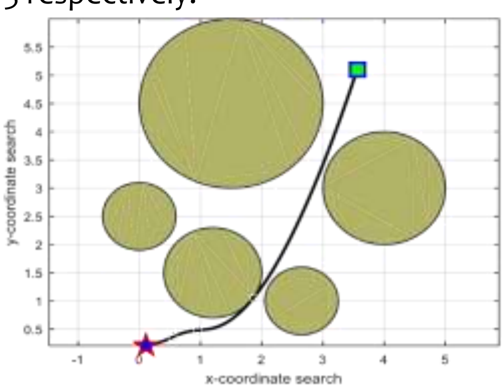

Fig 2: Optimized Path Using PSO algorithm

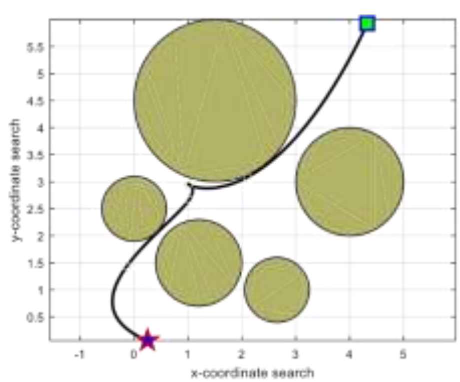

Fig 3: Optimized Path Using SAD algorithm

From Figure 2, it can be observed that the PSO obtained an optimal obstacle free path from the source to the goal point. This show the efficiency of the PSO for obstacle free robot path planning. The SDA also obtained an obstacle free path between the source and destination point. Unlike the PSO, the SDA obtained a much complicated and longer path for the robot. Though this path is longer, the robot can also move successfully from the starting position to goal position in an online manner without any obstruction by the obstacles. Table 3 shows the path planning fitness function obtained by each algorithm and the time it takes the algorithms to plan the optimized paths.

Table 3 Fitness Value for each Algorithm

SN	Algorithm	Fitness Value	Time (Sec)
1	PSO	6.4209	86.7963
2	SDA	10.5642	77.1479

Comparing the performance of the algorithms given in Table 3, it can be observed that the fitness value obtained by PSO is much better than that obtained by SDA. However, the SDA attained its optimum cost in less computational time than that of the PSO.

CONCLUSION

This paper presents Robot path planning technique in an intermediate environment using Particle Swarm Optimization and Smell Detection Agent Optimization. The developed path planning technique was implemented in MATLAB simulation environment. Results shown the proposed technique can effectively plan an optimized obstacle free path for a mobile robot. When the results of the two algorithms were compared, results showed that PSO performed better than SDA in terms of cost function whereas, the computational From Figure 2, it can be observed that the PSO requirements of SDA is slightly better than that of obtained an optimal obstacle free path from the source PSO.

REFERENCES

M. Amir and A. M. Bruckstein, "Minimizing Travel in the Uniform Dispersal Problem for Robotic Sensors," in *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, 2019, pp. 113-121: International Foundation for Autonomous Agents and Multiagent Systems.

V. Roberge, M. Tarbouchi, and G. J. I. T. o. I. I. Labonté, "Comparison of parallel genetic algorithm and particle swarm optimization for real-time UAV path planning," vol. 9, no. 1, pp. 132-141, 2012.

A. Tharwat, M. Elhoseny, A. E. Hassanien, T. Gabel, and A. J. C. C. Kumar, "Intelligent Bézier curve-based path planning model using Chaotic Particle Swarm Optimization algorithm," vol. 22, no. 2, pp. 4745- 4766, 2019.

T. T. Mac, C. Copot, D. T. Tran, R. J. R. De Keyser, and A. Systems, "Heuristic approaches in robot path planning: A survey," vol. 86, pp. 13-28, 2016.

S. MahmoudZadeh, D. M. Powers, and R. B. Zadeh, "State-of-the-Art in UVs' Autonomous Mission Planning and Task Managing Approach," in *Autonomy and Unmanned Vehicles*: Springer, 2019, pp. 17-30.

Z. Zhou, J. Ding, H. Huang, R. Takei, and C. J. A. Tomlin, "Efficient path planning algorithms in reachavoid problems," vol. 89, pp. 28-36, 2018.

W. Chen *et al.*, "Including robustness in multi-criteria optimization for intensity-modulated proton therapy," vol. 57, no. 3, p. 591, 2012.

J. Ni, K. Wang, H. Huang, L. Wu, and C. Luo, "Robot path planning based on an improved genetic algorithm with variable length chromosome," in *Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), 2016 12th International Conference on*, 2016, pp. 145-149: IEEE.

R. Rashid, N. Perumal, I. Elamvazuthi, M. K. Tageldeen, M. A. Khan, and S. Parasuraman, "Mobile robot path planning using Ant Colony Optimization," in *Robotics and Manufacturing Automation (ROMA), 2016 2nd IEEE International Symposium on*, 2016, pp. 1-6: IEEE.

A. Tuncer, M. J. C. Yildirim, and E. Engineering, "Dynamic path planning of mobile robots with improved genetic algorithm," vol. 38, no. 6, pp. 1564- 1572, 2012.

M.-P. Song and G.-C. Gu, "Research on particle swarm optimization: a review," in *Proceedings of 2004 International Conference on Machine Learning and Cybernetics (IEEE Cat. No. 04EX826)*, 2004, vol. 4, pp. 2236-2241: IEEE.

R. A. Ammal, P. Sajimon, and S. Vinodchandra, "Application of Smell Detection Agent Based Algorithm for Optimal Path Identification by SDN Controllers," in *International Conference on Swarm Intelligence*, 2017, pp. 502-510: Springer.

E. Masehian and D. Sedighizadeh, "A multi-objective PSO-based algorithm for robot path planning," in *2010 IEEE International Conference on Industrial Technology*, 2010, pp. 465-470: IEEE.

S. Mirjalili, J. S. Dong, and A. Lewis, "Ant Colony optimizer: theory, literature review, and application in AUV path planning," in *Nature-Inspired Optimizers*: Springer, 2020, pp. 7-21.