AUTONOMOUS GROUND WHEELED ROBOT MOTION PLANNING IN STATIC ENVIRONMENT USING TLBO ALGORITHM

[1]Preeti Sharma, [2]Harsh Pandey

[1]M. Tech. Student
[2]Associate Professor
[1,2]Department of Mechanical Engineering,
[1,2]Dr. C. V. Raman University Kota, Bilaspur, PIN - 495113, India

Abstract: Navigation and obstacle avoidance both are most crucial elements for a mobile robot, and this article focuses on the same. This article concentrates on the analysis of various intelligent navigation approaches. These approaches are skilled enough to navigate mobile robot autonomously in static environments. This article describes the optimization of a collision-free path in various static environments. For this, we are introducing the novel approach for the mobile robot to guide along its predefined target. This approach is Teacher Learner based Optimization (TLBO) algorithm. TLBO algorithm is based on the teaching and learning mode of teacher and learner respectively. It optimizes the path selection of mobile robot by taking Euclidean distance and steering angle as the key factor. Numerical computer simulations are the key of this paper to show the effectuality of the recommended standard path-planning control project.

***Index Terms* – Autonomous ground wheeled robot, Navigation, Static obstacle, Teacher Learner based Optimization algorithm, Numerical computer simulation.**

I. INTRODUCTION

Navigation is all about determining an efficient path to the desired location by avoiding collisions the travel taken and conserving energy. Wheeled robot navigation is thus a technique in which wheeled robot is guided between obstacles to reach the goal from starting point. Demand for the wheeled robot is expanding continuously as in increase of complexity, advancement of hardware and researches at dangerous places. The necessity for wheeled robots has been increased as it becomes impossible and too dangerous to interact with some situations like the fire, nuclear test, etc. for the human. Therefore, the replacement of human by wheeled robot become necessary to protect lives and to execute dangerous and delegate works.

Global navigation and local navigation are the two branches of navigation in the study area of wheeled robotics [1]. For the local navigation problems, vision (camera) sensors, ultrasonic range finder, and other sensors are annexed to the wheeled robot. With the help of these, the wheeled robot or powerbot itself is capable of deciding or controlling its motion and orientation. Various researchers have efficaciously utilized different algorithms like Genetic algorithm (GA) [2], Fuzzy logic controller (FLC) [3], Ant colony optimization (ACO) algorithm [4], Neural network (NN) [5], Neuro-fuzzy [6], Particle swarm optimization algorithm (PSO) [7], and simulated annealing (SA) algorithm [8], etc., to solve local navigation problems. Similarly, for the global navigation problem, the initial step is to gather the knowledge of the environment. Many kinds of researches have efficiently applied different methods to solve the problem on global navigation. Some of them are Dijkstra algorithm [9], Voronoi graph [10, 11], graph [13], Cell decomposition method [12], Grids [14], etc. According to literature survey, this is the first research to apply TLBO algorithm for wheeled robot navigation in the various environments.

II. DESIGN OF TEACHER LEARNER BASED OPTIMIZATION (TLBO) ALGORITHM FOR WHEELED ROBOT NAVIGATION

Rao et al. [15] have proposed TLBO (Teaching-learning-based optimization) by taking care of all above facts. His proposed algorithm didn't call for any algorithm-specific parameters. Optimization researchers recognized TLBO as better optimization approach and were adopted by them as it calls for only similar governing specifications such as population size and number of generation for it's working. TLBO algorithm is influenced by two main factors and those as (i) teaching process and (ii) learning process. So, as obvious this algorithm consists of two elemental situations of learning: (i) learning through teacher (Teacher Phase) (ii) Learning through interaction between the learners (Leaner Phase). As described above to solve any algorithm we require some governing parameters like population and design variables. TLBO extracts collection of learners as population and various subjects provided to the learner are extracted as various design variables for the optimization challenges. A learner's conclusion is comparable to the 'fitness' value for the given optimization problem. The teacher is accepted as the best solution for the whole population. Parameters that are used in the target function of the defined optimization challenges are design variables, and the best result describes the prime value of the target function. The basic steps of the algorithm, which is implemented for autonomous ground wheeled robot are given below: -

(a) Basic steps for optimizing path in the predefined environment using TLBO are given below: -

Step 1. Input start point, goal point, and workspace coordinate.
Step 2. Investigate the environment
Step 3. Receive wheeled robot information
Step 4. Minimal distance calculation between the robot and the object

$$r = (X^2 + Y^2)^{1/2} \quad (1)$$

Step 5. Are there any obstacles between robot and final point?
Step 6. If no, follow the same minimal path. If yes, find the obstacle position, turn right/left according to obstacle position and again find the minimal distance between robot and goal.

Step 7. Repeat step from 4 to 6 till the robot reach the goal.

Step 8. Report optimal solution and exit.

Block diagram of path planning using TLBO is shown in Figure 1.

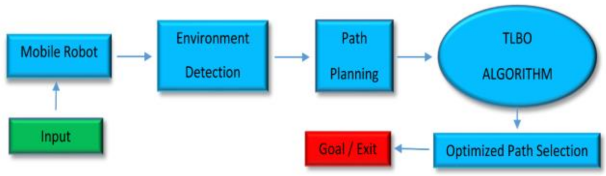

Figure 1: Basic steps in finding shortest path using TLBO

(b) Enhancement of Fitness Function in TLBO

Main objective of path planning is to get minimum path distance between the start point and goal with the shunning of hindrances in between the path. So the challenge in path planning is not finding the shortest path, it is finding the optimum path length in which path length is minimum along with the collision-free and less computational cost. In TLBO we are going to optimize path planning by taking two fitness function into consideration at once.

i. Fitness Function on The Basis of Path Length

ii. Fitness Function on The Basis Steering Angle to Avoid Collision.

To optimize the objective function, we need to select path length which is minimum and avoid the obstacle at the same time.

To get the minimum distance fitness function is taken as Euclidean distance.

$$S = \sum_{i=1}^{s} \sqrt{(x(i+1) - x(i))^2 + (y(i+1) - y(i))^2} \quad (2)$$

Where $x(i+1)$ & $x(i)$ are the new and old position of wheeled robot in x direction respectively, $y(i+1)$ & $y(i)$ are new and old position of wheeled robot in y direction respectively.

Fitness function "F" is described as the summation of f_1 and f_2. f_1 is fitness function for length and f_2 for shunning hindrances.

$$F = f_1 + f_2 \quad (3)$$

Where, $f_1 = S$ (Euclidean distance)

$f_2 = \theta$ (Steering angle to avoid obstacles)

The above equation finds shortest path selection of robot between start and target point.

III. Simulation Results

We have used MATLAB software to ensure the efficiency of the proposed algorithms. Various iteration has been performed to obtain the result precisely and is shown for evidence. Simulations are carried out in the MATLAB GUI (Graphical user interface) using different shapes and positions of the obstacles. In the same shape and position of obstacles, the positions of start point and goal have also been changed to ensure the robustness of the applied algorithms. When the robot comes close to any obstacle, the algorithms will be activated after sensing it. It decreases the speed of the robot and modifies its steering angle to avoid shunning with the hindrances and reach the given target. It is very important to turn at an appropriate angle and follow the most efficient path to reach the target. These are necessary to improve the efficiency of wheeled robot concerning path length, computation cost, obstacle avoidance and travel time. Therefore, the priority of proposing any algorithm is to get minimum path length with minimum possible safest distance from the obstacle in the environment.

After 60-70 and 4-5 number of runs/attempts of the given algorithms following above results are obtained The start point (S) of the robot is (50, 50) and the goal (G) is (180, 200).

- In the first run of the algorithm, wheeled robot follows Euclidean distance and doesn't get turned even obstacle comes in between the trajectory. (See the Figure 2).
- In the second run, the robot steered, but the steering angle was not sufficient to avoid collision. (See the Figure 3).
- In next run, robot and it avoided the obstacle, but while doing this path length get increased and time taken to perform the task is also increased. (See the Figure 4).
- After many runs, we obtained the best result as it is avoiding the obstacles with minimum possible and safest steering angle to ensure the minimum path length. (See the Figure 5).
- Table 1 describes the path length (in cm) and time taken (sec) to perform the respective tasks.

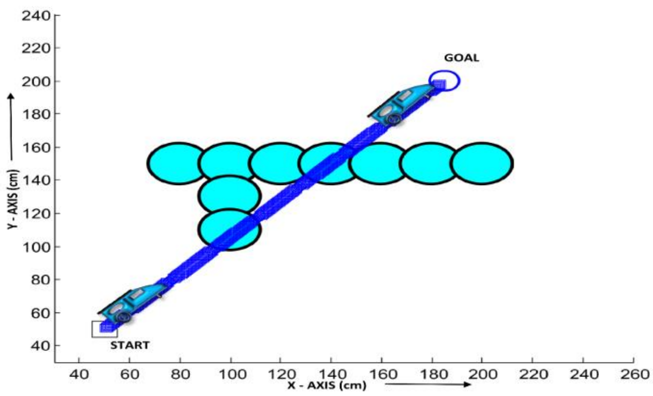

Figure 2: Route achieved by wheeled robot using TLBO algorithm (1st case) (S (50, 50) and G (190, 200))

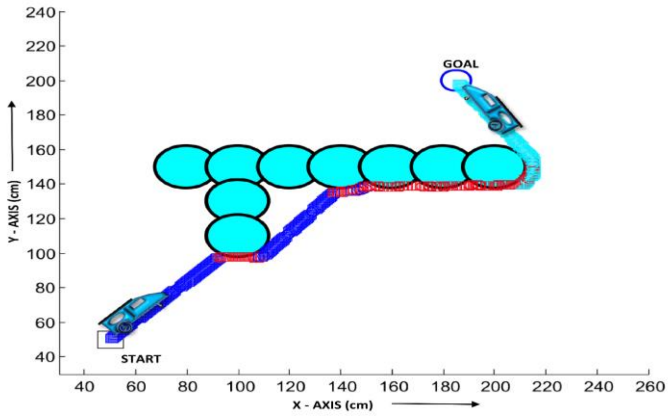

Figure 3: Route achieved by wheeled robot using TLBO algorithm (2nd case) (S (50, 50) and G (190, 200))

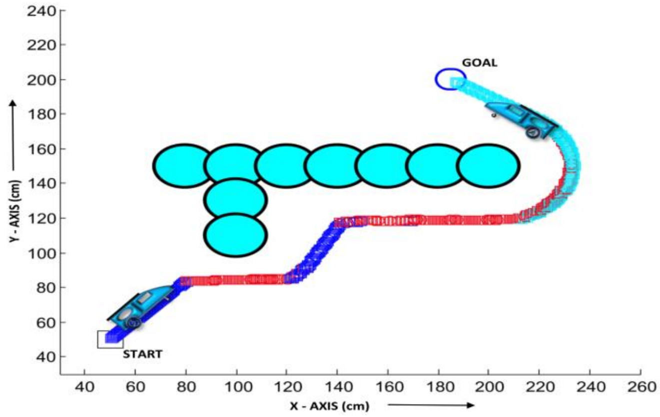

Figure 4: Route achieved by wheeled robot using TLBO algorithm (3rd case) (S (50, 50) and G (190, 200))

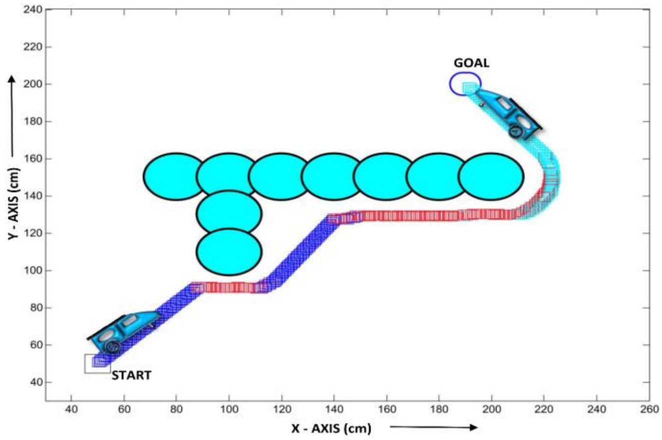

Figure 5: Route achieved by wheeled robot using TLBO algorithm (4th case) (S (50, 50) and G (190, 200))

Table 2: Comparison of different iteration for TLBO on the basis of path length and time taken

Figure no.	Path Length (cm)	Time Taken (sec)	Avoided Obstacle
Figure 2	198.49	18.2	NO
Figure 3	296.14	23.7	NO
Figure 4	360.33	37.3	YES
Figure 5	329.68	30.23	YES

IV. Conclusion And Future Scope

The conclusion part of the manuscript has been briefly described below: -

- In this article, the two algorithms TLBO has been proposed for wheeled robot navigation and path optimization.
- Both algorithms have been successfully implemented in various simulation environments.

- Finally, the simulation studies approved that current navigational algorithms can be qualified for any kind of complicated environments.
- In future, proposed algorithms can be implemented for dynamic obstacle avoidance and can also be used for multiple robot navigation and obstacle avoidance.
- In addition, the reader of this article can replace these developed algorithms through other nature-inspired algorithm for further research work.

REFERENCES

[1] Ni, J., Wu, L., Fan, X., and Yang, S. X. 2016. Bioinspired intelligent algorithm and its applications for mobile robot control: a survey. Computational intelligence and neuroscience, 1: 1-15.

[2] Ghorbani, A., Shiry, S., and Nodehi, A. 2009. Using genetic algorithm for a mobile robot path planning. In IEEE International Conference on Future Computer and Communication, Malaysia, 164-166.

[3] Montaner, M. B., and Ramirez-Serrano, A. 1998. Fuzzy knowledge-based controller design for autonomous robot navigation. Expert Systems with Applications, 14(1): 179-186.

[4] Garcia, M. P., Montiel, O., Castillo, O., Sepúlveda, R., and Melin, P. 2009. Path planning for autonomous mobile robot navigation with ant colony optimization and fuzzy cost function evaluation. Applied Soft Computing, 9(3): 1102-1110.

[5] Engedy, I., and Horváth, G. 2010. Artificial neural network based local motion planning of a wheeled mobile robot. In IEEE International Symposium on Computational Intelligence and Informatics (CINTI), Hungary, (2010), 213-218.

[6] Zhu, A., and Yang, S. X. 2007. Neurofuzzy-based approach to mobile robot navigation in unknown environments. IEEE Transactions on Systems, Man, and Cybernetics, 37(4): 610-621.

[7] Ahmadzadeh, S., and Ghanavati, M. 2012. Navigation of mobile robot using the PSO particle swarm optimization. Journal of Academic and Applied Studies (JAAS), 2(1): 32-38.

[8] Miao, H., and Tian, Y. C. 2013. Dynamic robot path planning using an enhanced simulated annealing approach. Applied Mathematics and Computation, 222: 420-437.

[9] Soltani, A. R., Tawfik, H., Goulermas, J. Y., and Fernando, T. 2002. Path planning in construction sites: performance evaluation of the Dijkstra, A∗, and GA search algorithms. Advanced engineering informatics, 16(4): 291-303.

[10] Takahashi, O., and Schilling, R. J. 1989. Motion planning in a plane using generalized Voronoi diagrams. IEEE Transactions on robotics and automation, 5(2): 143-150.

[11] Bhattacharya, P., and Gavrilova, M. L. 2008. Roadmap-based path planning-using the voronoi diagram for a clearance-based shortest path. IEEE Robotics & Automation Magazine, 15(2): 58-66.

[12] Park, K. H., Kim, Y. J., and Kim, J. H. 2001. Modular Q-learning based multi-agent cooperation for robot soccer. Robotics and Autonomous Systems, 35(2): 109-122.

[13] Masehian, E., and Amin-Naseri, M. R. 2004. A voronoi diagram-visibility graph-potential field compound algorithm for robot path planning. Journal of Robotic Systems, 21(6): 275-300.

[14] Weigl, M., Siemiåatkowska, B., Sikorski, K. A., and Borkowski, A. 1993. Grid-based mapping for autonomous mobile robot. Robotics and Autonomous Systems, 11(1): 13-21.

[15] Rao, R. V., Savsani, V. J., & Vakharia, D. P. 2011. Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems. Computer-Aided Design, 43(3): 303-315.