Three-Stage Optimization Model for Mobile Robot Path Planning

Suresh. K.S.[1*], R.Venkatesan[2], S.Venugopal[3]
[1,2]School of Computing, SASTRA Deemed University, Thanjavur, India
[3]Director, National Institute of Technology, Nagaland, India
Email: kssuresh@cse.sastra.edu

Abstract

In various real-time situations, the Mobile Robot Path Planning Problem (MRPPP) is one of the most prevalent study fields. In this article, a hybrid path planning technique for mobile robots is used and tested in various environment settings. This approach adopts pre-improvisation and post-improvisation methods for achieving the optimized path with quality with less computational cost. An MRPPP is resolved using a hybrid of the Artificial Potential Field (APF) method and the Multi-Objective Genetic Algorithm (MOGA). The suggested hybrid technique divides the implementation process into three stages. An Artificial Potential Field (APF) algorithm is used to find all feasible paths between the start and destination locations in an environment in order to construct the initial population. By calculating the artificial field produced by obstacles and the target, collision-free paths are created. An optimal solution path is extracted from the initial population of candidate paths using the population-based evolutionary method. In this article, the non-dominated sorting genetic algorithm II (NSGA-II) is applied to identify the optimal solution by concurrently maximizing all the objectives. In the end, the Three Phase Path Refinement Technique(TPRT) is employed to smoothen the derived optimal path. The suggested approach is employed in several maps and the results are compared with other path-planning algorithms to demonstrate the effectiveness of the system.
Keywords: Multi-objective Genetic Algorithm, Artificial Potential Field Algorithm, Smoothening of the path, Quality Initial Population, Path Planning Algorithm.

1. Introduction

Mobile robot path planning improves efficiency, safety, and effectiveness in a variety of industries and domestic applications. The major goal of this article is to use a hybrid technique to ascertain the MRPPP's best course of action inside the MOGA framework. While using the hybrid technique, the objectives must be specified and need to be maximized or decreased. The following factors are taken into consideration in order to arrive at the consequent path while determining the solution to the MRPP issue.

The length of the path is always viewed as a significant objective in the MRPP problem. Hence the time it takes to reach the target, which is the primary objective of this problem, is influenced by the length of the navigational path.

In a practical situation, the number of turns made by each specific path throughout the traversal in addition to the length is objective. The degree of turning from one place to another in the navigational path is used to quantify the smoothness of the path, which is considered another objective. This objective is preferred since a lesser degree of turning or a larger degree of turning might cause the Mobile Robot to tumble. So, the safety (Hidalgo-Paniagua et al., 2016) of the Mobile Robot is associated with this objective (Hidalgo-Paniagua et al., 2015). The quantity and intensity of turns taken on the subjective paths are directly connected to another objective energy consumption (Elhoseny et al., 2018) needed to get to the desired location.

The fitness values are estimated for the paths for the objectives then the Pareto optimality principle is employed, which is defined as a non-dominated strategy. For the selection phase of the MOGA, elitism is applied to improve the quality of the population in each generation.

As the last stage, TPRT is applied to smoothen the path which will reduce the turnings further, as a result, reduce the energy consumption. As a whole, this proposed methodology ensures the quality of the path for the traversal of the mobile robot in all aspects.

2. Related works

The advancement of robotics, where MRPPP is a prominent research topic, played an inevitable impact on the automation of industries. In order to reach the goal from the source point by the mobile robot with the higher performance, a lot of studies have been conducted. As substantially all engineering-related features are involved in the robotic research domain, there is a span to attain goals in a variety of methods. Due to the complexity of the RPP, it is classified as an NP-hard problem (P. Raja, 2012), making it unattractive to deduce the solution path using a deterministic approach (E. Khanmirza, M. Haghbeigi, 2017), even if it may produce an accurate solution if one exists.

The non-deterministic strategies offer an optimum solution to get the solution within a given amount of time or generations, maybe with a close to ideal solution. The quality of the solution is influenced by the operators used to generate the solution in population-based meta-heuristic algorithms (Suresh et al., 2013). The solution obtained by using evolutionary algorithms is close to optimum since it outperforms deterministic approaches (Suh et al., 2011) for the huge solution space.

A population-based technique called the Multi-Objective Genetic Algorithm searches a large randomly generated solution space for the optimal solution (Suresh et al., 2017). Due to the intrinsic properties of the algorithms, the genetic algorithm (GA) approach is the most trustworthy search technique for complicated optimization problems. For MRPP, exhaustive experiments are performed to identify the optimal path using GA with multiple modifications in the execution phases and the number of generations, while only taking into account one objective as the length of the path.

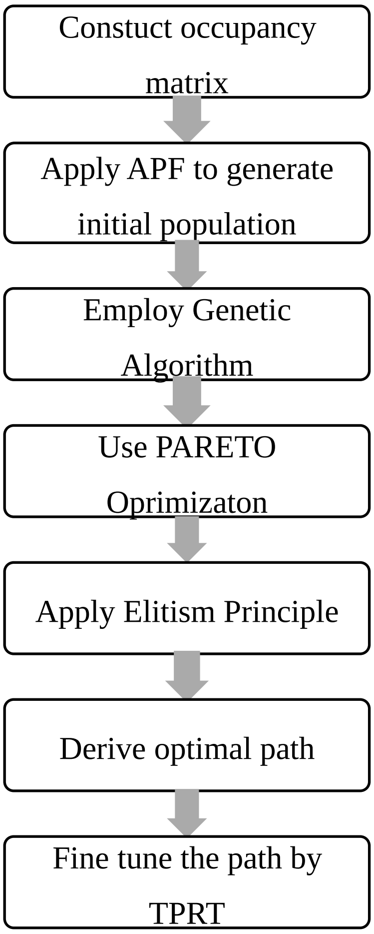


Figure 1 Flowchart for proposed TSOM-RPP model

If there are many objectives to be taken into consideration, they are instead added weight for the objectives or used as a constraint. The result of these methodologies provides only one optimal solution. It requires expertise in the relevant domain since even a minor variation from the value may impact the effectiveness of the solution.

By using MOGA, a group of non-dominated solutions may be created, allowing the decision-makers to make their choice based on their preferences and the trade-off between the objectives. Nevertheless, using the Pareto optimization technique for such a large population multiplies the computing complexity of the process.

The suggested technique makes use of the Pareto optimality principle, but to get around some of its limitations, it is combined with an alternative method called the Artificial Potential Field algorithm, which is used to generate the most efficient initial population. Using Pareto optimization lowers the computing cost as the original population was reduced.

The phases of the proposed TSOM-RPP are introduced in the following sections, which also include an illustration of the APF algorithm, a description of how each phase of the GA for MRPP is implemented, an implementation of Pareto optimization, a discussion of the principle of elitism, explanation for the TPRT, results from experiments for various environments, a list of inferences drawn from the results, and a conclusion.

3. Three-Stage Optimization Model for Robot Path Planning(TSOM-RPP)

The phases of the proposed TSOM-RPP are illustrated in **Figure 1**. It starts with initial population solution space creation and ends with applying the TPRT for the smoothening of the obtained optimal path.

Stage 1: Apply the APF algorithm to generate the initial population

Stage 2: Employ the Pareto optimization technique and elitism

Stage 3: Exercise the Three Phase path Refinement Technique(TPRT) for the derived path

This proposed model aims to improve the quality without increasing the computational cost. The first stage improvises the quality of the initial population at the same time stage 3 is further improvising the quality of the derived path to reduce the energy consumption.

Stage 1: Apply the APF algorithm to generate the initial population

The Artificial Potential Field process yields a potential field matrix by allocating appropriate potential value to each environment node, beginning with the destination node(Siming et al., 2018), which assists in finding all potential paths to reach the goal point from the starting location(Zafar & Mohanta, 2018). The generated solution paths establish the solution space to apply GA.

Stage 2: Employ the Pareto optimization technique and Elitism principle

Using Pareto Optimization(Konak et al., 2006), the population resulting from the genetic operations is ordered according to the number of dominance(Deb et al., 2010). The sets of ranked paths that are expected to include the best paths as a result of the objective evaluation are used to choose the necessary number of optimal paths. To rank the top paths to a solution, Pareto optimization is used. Each generation will get the better-qualified paths due to the Elitism principle for the selection of the offspring to the next generation.

Stage 3: **Exercise the Three Phase path Refinement Technique(TPRT)**

The smoothing procedure is applied to the derived optimal path after using GA. This technique is approached in three phases to get a smooth path.

Typically, the initial solution space is rather noisy because of the randomly created paths. These randomly generated paths demand a considerable amount of time to converge to the optimal path if GA is applied to them. To enhance the path qualitatively, the suggested technique was applied in three steps. APF is used in the first stage of the three-stage model that is being suggested, and the MOGA is implemented using the Pareto optimality principle. Compared to randomly generated paths, the initial population produced by APF has a path of higher quality. The APF produces an initial population for GA, which will shorten the convergence time. While applying GA, multiple objectives are considered such as length, number of turns in the path, and degree of diversion as penalty.

The initial population is produced from a potential grid that is constructed to deploy APF. The target node with the highest potential value, and as it moves through its neighboring nodes, the potential value steadily declines. If the adjoining node's potential values are the same, the subpaths will emerge. The output of APF is subjected to GA after producing each potential path. The fitness value is determined for each path by adding objective values. The total number of cells traversed to arrive at the desired location is accounted for by the objective length. For each path, the number of turns and degree of departure from the prior direction are considered as other objectives to be satisfied.

When there are many objectives, the Pareto optimization principle is used to assess how effective the approach is. Non-dominated paths, or the paths where no other possible paths can dominate it without lowering at least one of the objective values, constitute the Pareto front. In light of all other objectives, no alternative solution is therefore superior. Based on its non-dominated feature, paths are graded. Another concept that aims to improve GA performance by transferring the quality paths to the upcoming generation is called elitism. The convergence time is therefore minimized.

The normalized distance between two neighboring solutions concerning the objective values is added up to assess the variety of the non-dominated paths. The least crowded solutions are those that have the greatest value. The outcomes are examined with existing technologies and for the provided maps.

The length of the path may be taken into account as a goal for solving the MRPPP, and the other objectives may be taken into consideration as model constraints. Yet, in this instance, there will only be one solution found, preventing decision-makers from considering the benefits and drawbacks of a larger number of options with multiple objectives. As a result, the Pareto optimality principle(A & Panchu, n.d.) is used in this suggested technique to find a solution for the MRPPP with multiple objectives (Ajeil et al., 2020; Geetha et al., 2011; Li et al., 2016; Mahmud et al., 2019).

Equally spaced nodes that form square grids constitute the environment(Bae et al., 2019). The illustration depicts the navigational area of the mobile robot environment, which includes open space, obstacles, the robot's starting point, and its target point. The environment in this diagram is represented by n × n unit-length discretized square grids.

4. Implementation

The principle of the APF is, the robot is influenced by the fields produced by goals and obstacles. The repulsive force is generated by obstacles and the attractive force is generated by goals.

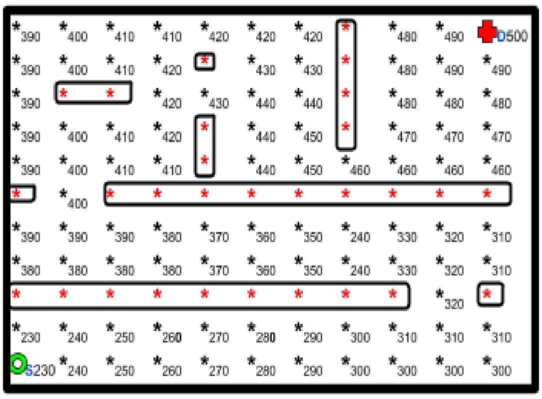


Figure 2 Potential Grid for the sample environment

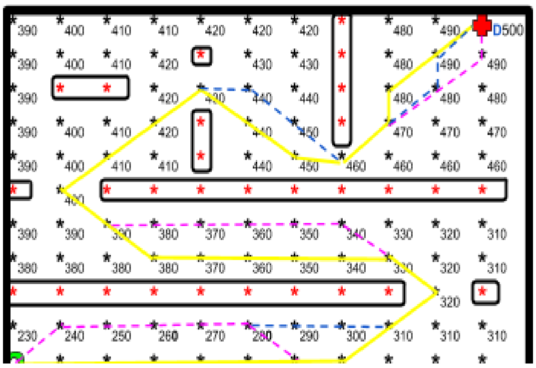


Figure 3 Sample paths generated through the potential grid

APF is implemented by constructing a potential grid. From that initial pop is generated. Arbitrarily chosen, the largest potential value is assigned to the target node[**Figure 2** and

Figure 3]. While Traversing through its adjacent nodes, potential decreases gradually. If the adjacent node has the same potential, leads to the sub-paths. GA is applied to the resulting initial population.

The potential of neighbors starting from the destination $P_{node} = P_{large} - D_{step}$

4.1. Fitness Value Estimation

Three objectives are taken to measure the fitness of the i^{th} path namely the length of the path(L_i), the number of turns in the path(T_i), and the degree of turns of the path(D_i).

Initially, Li = 0, Ti = 0, Di = 0

Length (Li) = number of cells in the i^{th} path

Number of turns of the path Pi where n = number of cells on the path $T_i = \sum_{j=2}^{n-1} A_j$

$$A_j = \begin{cases} 0 & \text{if } \angle A_{j-1}A_jA_{j+1} = 0° \text{ or } 360° \\ 1 & \text{else} \end{cases}$$

A_j = Angle formed by three consecutive cells

Deviation of the path P_i = $D_i = \sum_{j=2}^{n-1} A_j$

Where n = number of cells on the path

$$A_j = \begin{cases} 4 \text{ if } \angle A_{j-1}A_jA_{j+1} = 180° \\ 3 \text{ if } \angle A_{j-1}A_jA_{j+1} = 135° \text{ and } 225° \\ 2 \text{ if } \angle A_{j-1}A_jA_{j+1} = 90° \text{ and } 270° \\ 1 \text{ if } \angle A_{j-1}A_jA_{j+1} = 45° \text{ and } 315° \\ 0 \text{ if } \angle A_{j-1}A_jA_{j+1} = 360° \end{cases}$$

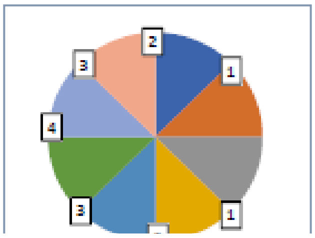


Figure 4 Penalty calculation for Deviation

Using the above functions the objective value is calculated for each objective in turn fitness value is calculated as the sum of the objective values. Specifically, the Degree of deviation from the previous direction is considered a penalty and the calculation is shown in **Figure 4**.

4.2. Pareto Optimization

When several competing objectives are taken into consideration during an optimization process, Pareto optimality is a measure of efficiency (Lavin, 2015). If there is no alternative path that improves the performance of any of its target parameters without compromising at least one of the other criteria, the process is known as Pareto optimization(Li et al., 2016). If no alternative path dominates it, a path is said to be Pareto optimum or non-dominated. The set of all non-dominated paths(Xue, 2018) taking into account a trade-off between all objectives constitutes the Pareto front.

4.3. Implementation Of Elitism

Genetic algorithms use operations like crossover, mutation, and selection to create offspring of the following generation. A philosophy known as elitism (Deb et al., 2002) is used to improve the effectiveness of the GA. The fundamental idea of elitism is to pass on the best members of the present generation to the following generation to retain the quality of the generation. This principle also shortens the convergence period as opposed to the nonelitism principle. A small percentage of the fittest candidates are copied to the following generation, by the elitism principle.

4.4. Crowding Distance

Deb (Deb et al., 2002) devised the crowding distance (CD), a technique that is applied in the NSGA-II (Non-dominated Sorting Genetic Algorithm II) (Xue, 2018). By estimating the density of the solution in the Pareto optimum front, this method assures the diversity of solutions. In a multi-objective environment, the CD is calculated by adding the normalized distances for each individual between two neighboring solutions, related to all objectives (Ahmed & Deb, 2013). The least crowded solution is the one with the largest overcrowding distance value.

$$cd_m^i = \frac{(f_m^{i-1} - f_m^{i+1})}{(f_m^{max} - f_m^{min})}$$

where i = i^{th} solution's fitness value

m = m^{th} objective

Therefore cd_m^i - distance from the neighbor's for the i^{th} solution is calculated

f_m^{i-1} - Fitness value of the $(i-1)^{th}$ (previous) solution for the m^{th} objective

f_m^{i+1} - Fitness value of the $(i+1)^{th}$ (next) solution for the m^{th} objective

f_m^{max} - Maximum fitness value of the m^{th} objective

f_m^{min} - Minimum fitness value of the m^{th} objective

$$CD_i = \sum_{m=1}^{M} cd_m^i$$

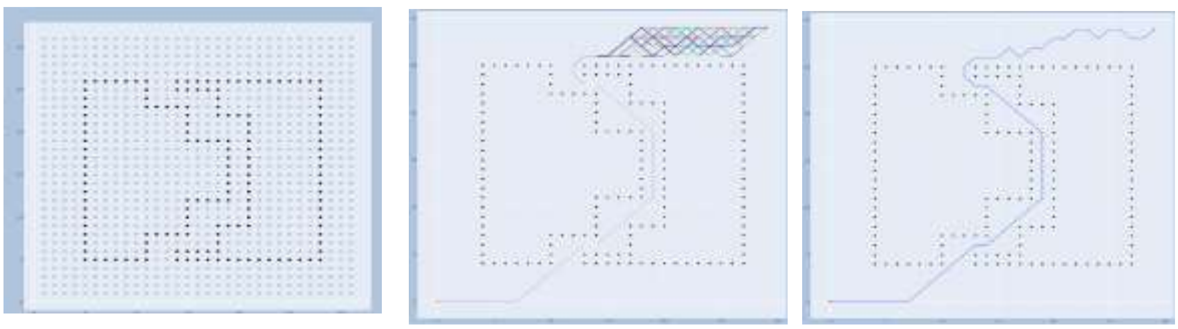
where CD_i = sum of crowding distance of i^{th} solution for all m objectives.

By estimating all the crowding distance values, by comparing the CDi values the one which satisfies the below condition will be rejected.

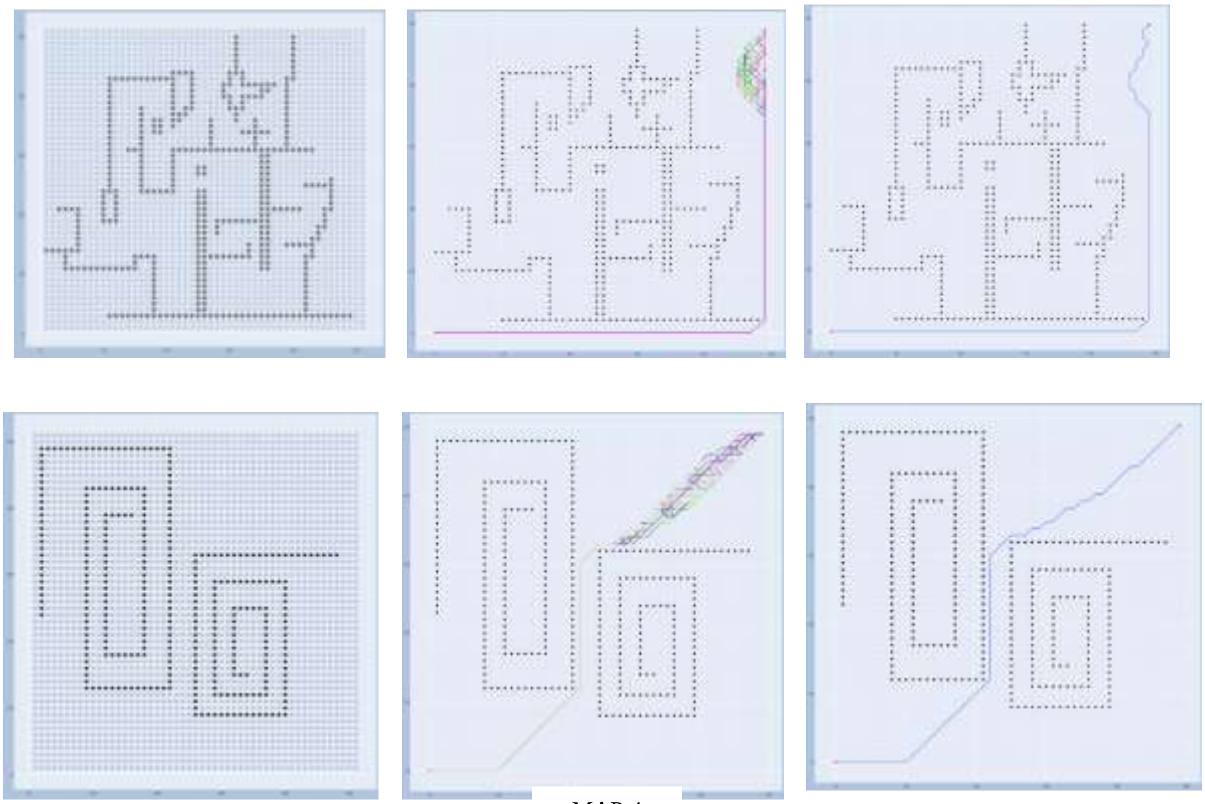
$$CDi < CDj$$

where CDi and CDj are crowding distances of two different solutions.

MAP 1



MAP 2



MAP 4

Figure 5 Sample Environments, Initial paths and optimized

5. **Implementation of the Three-Stage Optimization Model**

According to the proposed methodology, the first stage is generating the initial population by applying the APF algorithm. In this stage, the potential has to be assigned for each cell from the maximum for the destination. Arbitrarily for the destination, the highest potential value is assigned as 3000 and all cells occupied with obstacles are assigned with the highest negative potential as -3000. The neighbor cells are decremented by 10 for each layer.

In the second stage, the Pareto optimization technique and elitism are employed. 50 generations are taken for the implementation. At each generation, the genetic operators such as crossover, mutation, and selection operators are applied which are specific to the path planning solution space. The paths could not be selected randomly as in other domain problems. to employ genetic operators.

Table 1 Comparison of average objective values for 8 environment maps

Map 1	A*	Dijkstra	EGA	APF	HYBRID
Length	44.6	55.18	39.53	31	31
Penalty	49	73	49	55	36.39
Turns	270	300	300	228	159.12

Map 2	A*	Dijkstra	EGA	APF	HYBRID
Length	250.28	282.81	249.36	240	240
Penalty	660	933	615	139	127
Turns	600	585	615	588	527.38

Map 3	A*	Dijkstra	EGA	APF	HYBRID
Length	142.57	200.45	131.4	115	115
Penalty	214	352	190	142	128.41
Turns	795	990	615	588	544.67

Map 4	A*	Dijkstra	EGA	APF	HYBRID
Length	50.04	52.87	49.46	43	43
Penalty	22	23	21	92	76.54
Turns	390	405	360	384	324.65

Map 5	A*	Dijkstra	EGA	APF	HYBRID
Length	77.91	90.6	69.91	56	56
Penalty	31	18	28	92	82.76
Turns	690	360	420	384	352

Map 6	A*	Dijkstra	EGA	APF	HYBRID
Length	70.7	77.18	69.18	59	59
Penalty	113	129	99	102	90.47
Turns	465	525	555	444	410.28

Map 7	A*	Dijkstra	EGA	APF	HYBRID
Length	202.21	238.74	199.87	164	164
Penalty	146	143	143	137	125.1
Turns	555	600	720	552	516.24

Map 8	A*	Dijkstra	EGA	APF	HYBRID
Length	160.92	187.69	159.17	132	132
Penalty	358	419	351	131	119
Turns	540	630	555	540	490.55

8 MAPS are taken for experiment and analyze the proposed methodology and also to compare the efficiency of the other methods with objective values for the individual objectives and fitness values for the paths. Sample configuration maps are shown in figure MAP 1 to MAP 4 are shown in the figures with a potential grid size of 30 x 30.

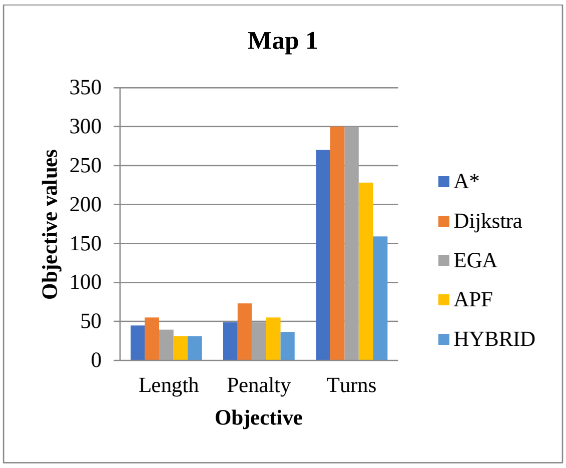
The figures[**Figure 5**] illustrate the environment map for 4 sample workspaces for the mobile robot, the corresponding paths generated by applying the APF algorithm, optimized path generated after employing the Pareto optimality and elitism principles for MOGA.

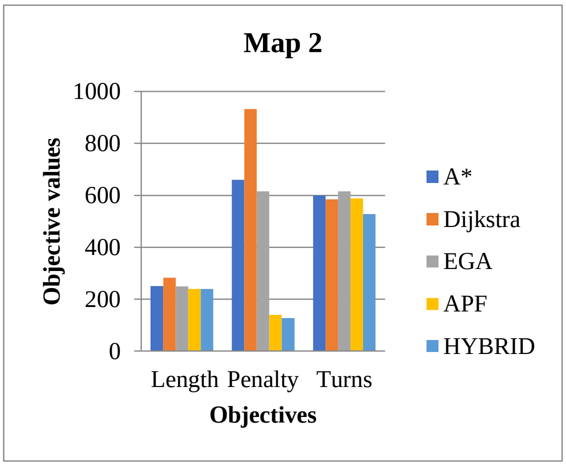
6. **Results and discussion**

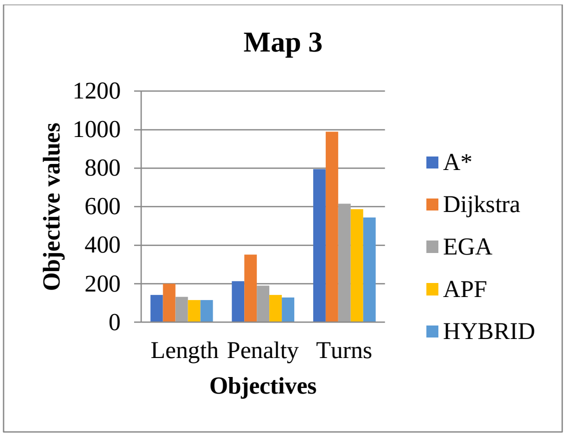
The results obtained by applying the algorithms are illustrated in **Table 1** as well as through the chart, from the results produced, it is clearly evident that the proposed algorithm is more efficient compared to the other four algorithms namely A*, Dijkstra, EGA, and APF for different maps.

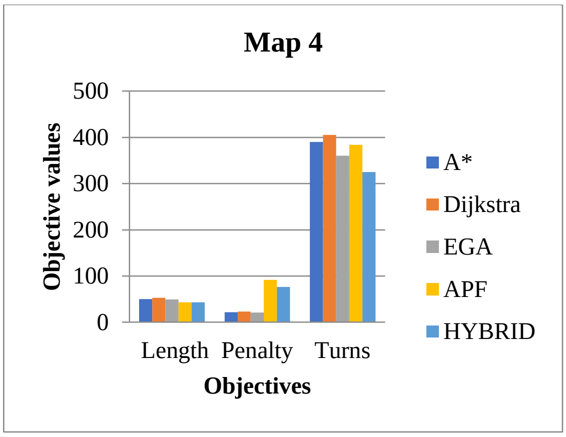
7. Comparison analysis

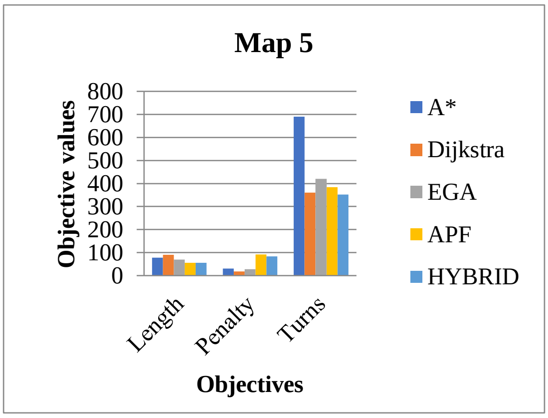
The figures in **Figure 6** are the illustration of the comparison of average objective values for the individual objectives of all the given 8 environmental maps.

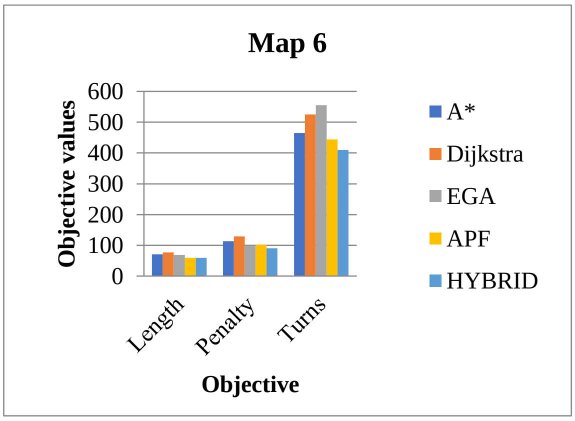

Map 1
Objective values
350
300
250
200
150
100
50
0
Length
Penalty
Turns
Objective
A*
Dijkstra
EGA
APF
HYBRID


Map 2
Objective values
1000
800
600
400
200
0
Length
Penalty
Turns
Objectives
A*
Dijkstra
EGA
APF
HYBRID


Map 3
Objective values
1200
1000
800
600
400
200
0
Length
Penalty
Turns
Objectives
A*
Dijkstra
EGA
APF
HYBRID


Map 4
Objective values
500
400
300
200
100
0
Length
Penalty
Turns
Objectives
A*
Dijkstra
EGA
APF
HYBRID


Map 5
Objective values
800
700
600
500
400
300
200
100
0
Length
Penalty
Turns
Objectives
A*
Dijkstra
EGA
APF
HYBRID


Map 6
Objective values
600
500
400
300
200
100
0
Length
Penalty
Turns
Objective
A*
Dijkstra
EGA
APF
HYBRID

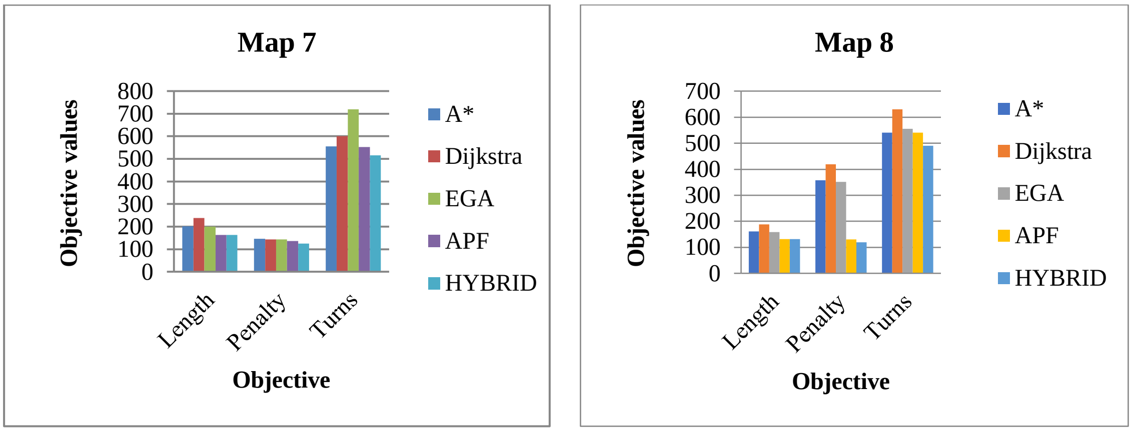


Figure 6 Comparison charts of the average objective values for 8 environments

The results obtained from the popular algorithms that dominated the field of path planning such as A*, Dijkstra, EGA(enhanced GA) and APF are judged against the proposed three-stage optimized model. In comparison with the average of other algorithms with the proposed method for different maps.

For better observation, the average of the results of other methods is compared with the proposed hybrid model results which is shown in **Figure 7** that the improved qualified paths are generated by the method.

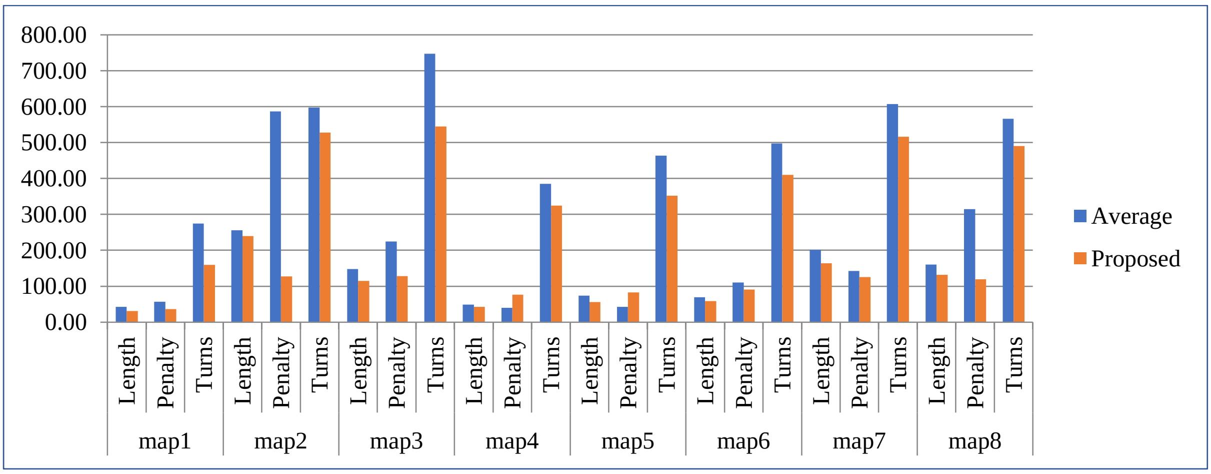


Figure 7 Comparison of the average of the results of 4 methods vs TSOM-RPP

8. Three Phase Path Refinement Technique (TPRT)

The restricted population size or limitation in the number of generations are directly influencing the overall computational cost. In a complex environment, the optimal path derived by employing the above two stages may not be highly qualified. In this sense, it may derive the path with artifacts that can be removed by applying the TPRT. This is highly supportive to reduce the energy consumption for the mobile robot. Because the energy consumption of the mobile robot is contributed by the length of the path and turnings of the path. TPRT addresses both of the said problems and refines the derived path to get a smooth path with the reduced turning of the path. TPRT is implemented on the path which takes the raw path as input and generates the smooth path for the traversal of the mobile robot.

8.1. Algorithm for the TRPT

Phase 1:

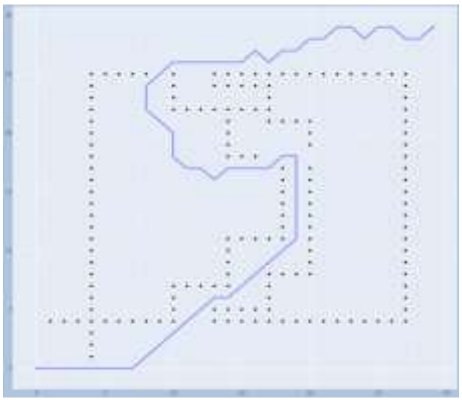
1. Initialize smoothened_path list to starting node
2. Set last_node as starting node
3. While last_node is not the destination do,
    a. Initialize adjacent_nodes to nodes adjacent to the last_node
    b. Remove obstacles from adjacent_nodes
    c. Sort adjacent_nodes in ascending order based on the distance between the node and destination
    d. For (temp_node in adjacent_nodes),
        i. If temp_node is present in the raw_path,
            1. Add temp_node to the smoothened_path
        ii. Else,
            1. Check if the temp_node satisfies the following conditions.
                a. Temp_node should not be an obstacle
                b. Lines drawn parallel to the x-axis or y-axis passing through the temp_node must intersect with the raw_path
                c. The line drawn between the temp_node and intersection node should not have any obstacles
                d. The intersection node should not be already present in the smoothened_path
            2. If the temp_node satisfies the given condition,
                a. Add temp_node to smoothened_path
                b. Break iteration
            3. Else,
                a. If the intersection_node is already present in smoothened_path,
                    i. Continue iteration
                b. If there are multiple intersecting nodes,
                    i. Choose the intersecting node which is nearest to the destination
                c. Construct a straight path between the last_node and chosen intersection node.
                d. Add all the nodes from the straight path to smoothened_path
                e. Break iteration
    e. Set last_node as the last node of smoothened_path

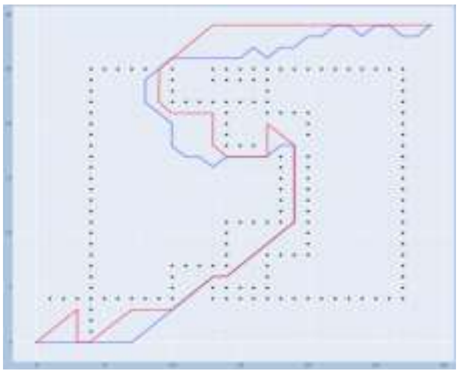
Phase 2:

4. To further reduce the artifacts during the previous process, follow the below steps.
5. Initialize clean_path to starting node
6. Set last_node to starting node
7. While last_node is not the destination, do
    a. Draw lines parallel to the x-axis and y-axis passing through the last_node
    b. Find the nodes of intersection between the lines and smoothened_path
    c. Choose the node which is farthest from the last_node such that a line drawn between the chosen node and last_node does not pass through any obstacles and is not already present in clean_path
    d. If no nodes pass the above criteria,
        i. Add the node that comes after last_node in smoothened_path to final_path
        ii. Set last_node as the last node of clean_path
        iii. Continue iteration
    e. Construct a straight path between the chosen node and last_node
    f. Add all the nodes in the straight path to clean_path
    g. Set last_node as the last node of clean_path

Phase 3:

8. To further optimize the path, follow the below steps.
9. Initialize final_path to starting node
10. Set last_node to starting node
11. While last_node is not the destination, do
    a. Draw diagonal lines passing through last_node.
    b. If either of the lines intersects with nodes in clean_path, such that there is no obstacle in a line between the intersection nodes and last_node
        i. Choose the intersection node which is farthest from the last_node
        ii. Construct a straight path between the intersection node and last_node
        iii. Add all nodes from the straight path to final_path
    c. Else,
        i. Add the next node from clean_path to final_path
    d. Set last_node as the last node of final_path





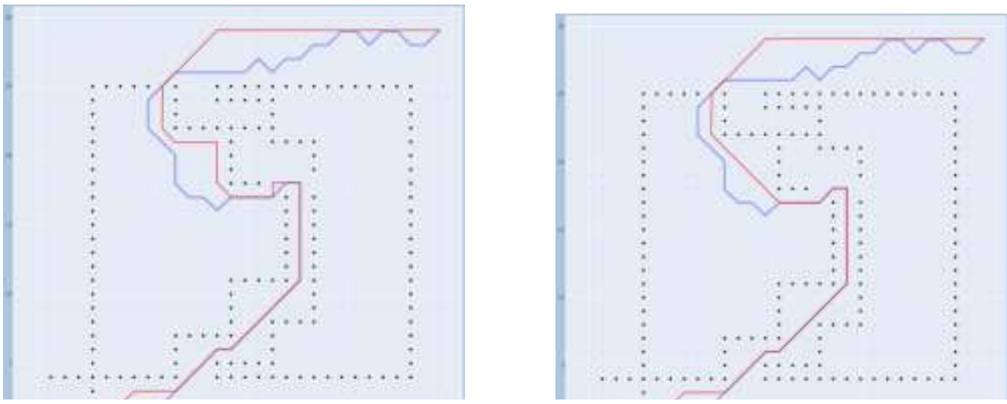


Figure 8 Illustration for the phases of TPRT from raw path

The TPRT algorithm is illustrated through the figures [**Figure 8**]. The first one is the raw path derived after two stages which is the input for the TPRT. If there are concave structures of obstacles in the map, phase 1 of the algorithm TPRT tends to create artifacts as shown in the second one. Phase 2 of the TPRT removes the artifacts of the path which is shown in the 3rd figure. Phase 3 is used to shorten the path by taking a diagonal path where ever possible to reach the destination that is demonstrated in the 4th figure of **Figure 8**.

The overall quality of the path is maintained without compromising the performance through this hybrid model.

9. Conclusion

A model is proposed to determine the path for the mobile robot which hybridizes the APF and non-dominated Pareto analysis. In this path planning algorithm, while implementing the MOGA, length, number of turns that exist in the path, and deviation of the path from the previous direction. The generated path has undergone another technique called the TPRT to smoothen the path. The results obtained from the popular algorithms that dominated the field of path planning such as A*, Dijkstra, EGA(enhanced GA) and APF are compared and analyzed. From the study, it is inferred that the proposed hybrid algorithm is producing better results for individual objective levels. In conclusion, the average percentage of improvement in objective length is 14.75%, the penalty calculated by deviation is 12.10%, and the in terms of the number of turns is 17.51%. Considering the overall fitness value, the average percentage of improvement in fitness value is 14.79%. The TPRT is used to remove the turns on the derived path which will further increase the safety and energy efficiency of the mobile robot.

References

A, B., & Panchu, P. (n.d.). Multi-objective Optimization of Mobile Robot Path Planning Using Genetic Algorithm. *IOSR Journal of Mechanical and Civil Engineering*, 2278–1684.

Ahmed, F., & Deb, K. (2013). Multi-objective optimal path planning using elitist non-dominated sorting genetic algorithms. *Soft Computing*, *17*(7), 1283–1299. https://doi.org/10.1007/s00500-012-0964-8

Ajeil, F. H., Ibraheem, I. K., Sahib, M. A., & Humaidi, A. J. (2020). Multi-objective path planning of an autonomous mobile robot using hybrid PSO-MFB optimization algorithm. *Applied Soft Computing Journal*, *89*. https://doi.org/10.1016/j.asoc.2020.106076

Bae, H., Kim, G., Kim, J., Qian, D., & Lee, S. (2019). Multi-robot path planning method using reinforcement learning. *Applied Sciences (Switzerland)*, *9*(15). https://doi.org/10.3390/app9153057

Deb, K., Member, A., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). *A fast and elitist multi-objective genetic algorithm:NSGAII*. *6*(2), 182–197.

Deb, K., Sinha, A., Korhonen, P. J., & Wallenius, J. (2010). An interactive evolutionary multiobjective optimization method based on progressively approximated value functions. *IEEE Transactions on Evolutionary Computation*, *14*(5), 723–739. https://doi.org/10.1109/TEVC.2010.2064323

E. Khanmirza, M. Haghbeigi, M. N. and S. D. (2017). A Comparative Study of Deterministic and Probabilistic Mobile Robot Path Planning Algorithms. *5th RSI International Conference on Robotics and Mechatronics (ICRoM)*, 534–539. https://doi.org/10.1109/ICRoM.2017.8466197

Elhoseny, M., Tharwat, A., & Hassanien, A. E. (2018). Bezier Curve Based Path Planning in a Dynamic Field using Modified Genetic Algorithm. *Journal of Computational Science*, *25*. https://doi.org/10.1016/j.jocs.2017.08.004

Geetha, S., Chitra, G. M., & Jayalakshmi, V. (2011). Multi objective mobile robot path planning based on hybrid algorithm. *ICECT 2011 - 2011 3rd International Conference on Electronics Computer Technology*, *6*, 251–255. https://doi.org/10.1109/ICECTECH.2011.5942092

Hidalgo-Paniagua, A., Vega-Rodríguez, M. A., & Ferruz, J. (2016). Applying the MOVNS (multi-objective variable neighborhood search) algorithm to solve the path planning problem in mobile robotics. *Expert Systems with Applications*, *58*, 20–35. https://doi.org/10.1016/j.eswa.2016.03.035

Hidalgo-Paniagua, A., Vega-Rodríguez, M. A., Ferruz, J., & Pavón, N. (2015). MOSFLA-MRPP: Multi-Objective Shuffled Frog-Leaping Algorithm applied to Mobile Robot Path Planning. *Engineering Applications of Artificial Intelligence*, *44*, 123–136. https://doi.org/10.1016/j.engappai.2015.05.011

Konak, A., Coit, D. W., & Smith, A. E. (2006). Multi-objective optimization using genetic algorithms: A tutorial. *Reliability Engineering and System Safety*, *91*(9), 992–1007. https://doi.org/10.1016/j.ress.2005.11.018

Lavin, A. (2015). *A Pareto Front-Based Multiobjective Path Planning Algorithm*. http://arxiv.org/abs/1505.05947

Li, M., Yang, S., & Liu, X. (2016). Pareto or Non-Pareto: Bi-criterion evolution in multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, *20*(5), 645–665. https://doi.org/10.1109/TEVC.2015.2504730

Mahmud, M. S. A., Abidin, M. S. Z., Mohamed, Z., Rahman, M. K. I. A., & Iida, M. (2019). Multi-objective path planner for an agricultural mobile robot in a virtual greenhouse environment. *Computers and Electronics in Agriculture*, *157*(August 2018), 488–499. https://doi.org/10.1016/j.compag.2019.01.016

P. Raja. (2012). Optimal path planning of mobile robots: A review. *International Journal of the Physical Sciences*, *7*(9), 1314–1320. https://doi.org/10.5897/ijps11.1745

Siming, W., Tiantian, Z., & Weijie, L. (2018). Mobile Robot Path Planning Based on Improved Artificial Potential Field Method. *2018 IEEE International Conference of Intelligent Robotic and Control Engineering (IRCE)*, *2*(1), 29–33.

Suh, W. J., Park, C. S., & Kim, D. W. (2011). Heuristic vs. Meta-heuristic optimization for energy performance of a post office building. *Proceedings of Building Simulation 2011: 12th Conference of International Building Performance Simulation Association*, 704–711.

Suresh, K. S., Ravichandran, K. S., Ananthakrishnan, S., & Venugopal, S. (2017). Adaptive strategy to improve the efficiency of robot path planning using population based algorithm. *International Journal of Mechanical Engineering and Technology*, *8*(8).

Suresh, K. S., Vaithiyanathan, V., & Venugopal, S. (2013). Emergence of Meta Heuristic algorithm principles in the field of robot motion planning-A survey of paradigm shift. *International Journal of Applied Engineering Research*, *8*(20 SPEC. ISSUE), 2717–2722.

Xue, Y. (2018). Mobile Robot Path planning with a non-dominated sorting genetic algorithm. *Applied Sciences (Switzerland)*, *8*(11). https://doi.org/10.3390/app8112253

Zafar, M. N., & Mohanta, J. C. (2018). Methodology for Path Planning and Optimization of Mobile Robots: A Review. *Procedia Computer Science*, *133*. https://doi.org/10.1016/j.procs.2018.07.018