

Comparing the novel Dhouib-Matrix-SPP to genetic algorithm for autonomous mobile robot path planning problem

Souhail Dhouib¹, Danijela Pezer², Tole Sutikno^{3,4}

¹OLID Research Laboratory, Higher Institute of Industrial Management, University of Sfax, Sfax, Tunisia

²University Department of Professional Studies, University of Split, Split, Croatia

³Master Program of Electrical Engineering, Faculty of Industrial Technology, Universitas Ahmad Dahlan, Yogyakarta, Indonesia

⁴Embedded System and Power Electronics Research Group, Yogyakarta, Indonesia

Article Info

Article history:

Received Oct 25, 2025

Revised Apr 3, 2026

Accepted Apr 19, 2026

Keywords:

Artificial intelligence

Automation and mobile robots

Computational intelligence

Genetic algorithms

Metaheuristic

Optimization

Shortest path problem

ABSTRACT

Autonomous mobile robots (AMRs) are becoming integral to applications ranging from industrial automation to urban mobility. A core challenge in deploying AMRs effectively is the path planning problem determining an optimal and collision-free path from a start to a goal location within a given environment. This paper proposes a novel method, Dhouib-Matrix-SPP (DM-SPP), that enhances path planning efficiency and adaptability for AMRs operating in different statistical environment. Basically, DM-SPP is developed to unravel the shortest path in a graph and based on columns-rows structure with polynomial computational time. Here, the DM-SPP method is adapted to plan the shortest feasible path between two positions while avoiding obstacles. In order to prove the validity of the proposed DM-SPP method, it is applied to different environments and compared to different case studies taken from the literature. The simulation results show that the DM-SPP method was able to find, with a significantly lower number of iterations, the optimal solutions in comparison with other results obtained by the genetic algorithm (GA) method. DM-SPP presents an overall average improvement in computation time of (37882.55%) compared to the GA, which can reduce search and execution time.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Souhail Dhouib

OLID Research Laboratory, Higher Institute of Industrial Management, University of Sfax

Sfax, Tunisia

Email: souhail.dhouib@gmail.com

1. INTRODUCTION

Mobile robots are more and more present in everyday life and at the same time their application reaches into different spheres of life and today they are successfully implemented in the field of health care [1], the military, in search and rescue (earthquakes, volcanoes, and hurricanes) in [2], [3] production, storage [4], logistics [5], and household [6]. Considering the growing popularity of using mobile robots, the goal is to use them as efficiently as possible. Mobile robots should mostly be used in situations dangerous for people, such as monitoring a dangerous environment. One of the most important tasks is thus to ensure a sub-optimal path from the starting point to the target point that we want to reach. Optimization is the most often done regarding to the path length or reducing time which is necessary to get from the starting position to the final position, while avoiding obstacles that can lead to a collision. As safety is a very important item in production, it must not be missing here either, and it is necessary to ensure the shortest path/time by respecting precautionary measures and safely avoiding obstacles on the path.

The working environment of the robot can be completely known, partially known, completely unknown, and static or dynamic. Each path has specific limitations (given that it is a collision-free path) depending on the area of its application. The optimal path can be observed in terms of distance (shortest path), time (shortest execution time), energy (minimum energy consumption). Path planning is an extremely important issue in mobile traffic, because the robot must perform many tasks that ensure its navigation for a given environment. The robot uses sensors to obtain information about obstacles on the path to the target position.

The best path planned is the path that gives the lowest costs. The method used to find a path for an object that is moving should cover the shortest path, and at the same time be economical, and ensure movement without collision with obstacles placed on that path, i.e., without collision. The simplest way to avoid obstacles is to create a map of the environment, which can be known or unknown, mark obstacles and choose a path using a graph algorithm. Path planning of a mobile robot is classified as an NP-hard problem and is similar to the traveling salesman problem.

The path planning problem can be divided into two groups, i.e., local and global path planning. Global planning only deals with basic path planning and does not include localization and path tracking. Global mobile robot planning is an important issue in the field of robotics, given the degree of production automation that is necessary both today and in future (by given the importance of process automation). Local planning implies finding a path from the initial to the final position in an unknown environment, while global path planning enables finding sub-optimal and optimal solutions in a known environment using different optimization algorithms such as ABC [7], ACO [8], GA [9], PSO [10], Dijkstra A* algorithm [11]-[13], and rapidly random tree* (RRT*) in [14], [15]. These methods require a complicated parameters setting with a considerable computational time to optimize the mobile robot path problem.

In this paper, the deterministic (using zero parameter) Dhouib-Matrix-SPP (DM-SPP) algorithm [16] is proposed for solving the shortest path problem of a mobile robot. DM-SPP is a fast method that can reduce research time and execution time as well as energy consumption. In this paper, the DM-SPP algorithm was tested and a study is conducted to compare its results with the results available in the literature [17]-[19].

The paper is organized as follows: section 2 provides insight into previously published papers by various researchers. Section 3 presents the method DM-SPP which explains the proposed algorithm, section 4 presents the study, results and discussion of the proposed method in relation to the various case studies conducted by the genetic algorithm (GA) method. Section 5 gives a summary of contributions and also the suggestions for future research work.

2. RELATED WORKS

Tool path planning occupies an important place in the field of artificial intelligence and mobile robotics and is still being researched today, by developing new methods and improving existing ones. While we are planning the robot path it is necessary to ensure the shortest possible path without collision or safe path [20]. In this study, the problem of ensuring a safe path and at the same time the shortest path for one or more mobile robots in a static global environment was considered. For the case of planning the path of one robot, a comparison of the results obtained with the improved objective function of the GA for different coefficients of safety parameters was performed. For the case of planning the path of several mobile robots, different maps of size 500×500 that simulate the real environment were applied. It was observed that the level of complexity of the map increases as the number of obstacles increases and the distance between them decreases. In the paper, the safety parameter was used as a fitness function and the influence of the safety parameter on the optimality and length of the path was shown.

The path planning problem using a GA in a static environment is proposed by Lamini *et al.* [21]. An improved crossover operator is proposed that avoids premature convergence and offers solutions (feasible path) with better fitness value than its parents and also converges faster. The fitness function takes into account the distance, safety and power consumption of the mobile robot. The environment is modeled as a 2D matrix, where 0 represents the free position while 1 represents the occupied position. The proposed approach is tested in 4 different environments with map sizes of (29×30), (13×30), (5×9), and (60×50) grid fields. The proposed GA finds the optimal path with fewer iterations compared to other methods and significantly reduces the number of runs due to a fitness function that favors straight line paths over zigzag lines.

Path planning of mobile robots in a dynamic environment with various obstacles using an improved GA algorithm is discussed [22]. Through local search, feasible paths are found, then these paths are optimized in order to find the shortest path. By finding the optimal path, the defined points are smoothed to avoid obstacles from the environment. GA operators propose to accelerate the evolution of individual populations on the path. The robot was driven with different number of obstacles (5, 8, and 10) in the environment and the results were compared with the traditional GA method. It was found that the path achieved by the proposed method is shorter and smoother compared to the traditional GA method. The

proposed method shows that the chromosomes generated by the planned path have better characteristics and faster calculation time in the solution space.

Research by Kwaśniewski and Gosiewski [23] use an advanced GA to determine the optimal path of a mobile robot avoiding obstacles in an unknown environment. Two versions were used, one based on a 2D map created by the robot, while a B-spline of the second degree was used for the path marking model. The proposed algorithm converges much faster than traditional graph algorithms. The return path is also represented in the main algorithm, and enables return to complicated paths when only one way in and out is possible (dead end). The return path from the starting position is also ensured if it is not possible to reach the destination in case of blocking by obstacles. The selection of the number of control points is complex due to the parameters of the improved algorithm. A smaller number of control points enables faster calculation, but is not suitable for complex roads, while the opposite is true for a larger number, which is not suitable because it requires a longer calculation time. The algorithm proved successful for most runs. The proposed algorithm can be improved to find the optimal path in 3D maps, which could be used, for example, to avoid collisions with 3D printers.

A GA for global path planning of a mobile robot in a known environment [24] uses a modified data structure to generate an optimal path. There are two phases of the algorithm, one starts the individuals to develop and checks the availability of chromosomes on the way to the destination. The best individuals are selected and moved to the second stage and then developed. Individuals that survive generate the optimal path. The method was tested in an area of 26×26 m and proved to be very effective compared to conventional methods as it quickly finds the optimal solution.

Path planning of a mobile robot in a known environment with the help of a modified GA that uses a modified genetic operator to find the best population is investigated in [25]. In this study, it is proposed to move the robot in eight directions instead of the usual four in a known environment consisting of a network in which each node represents a parameter. A set of nodes actually represents an individual, and only the best individuals form a new generation of parents. The optimal solution is achieved by eliminating long paths from the group of acceptable solutions. The proposed method was compared with other GA approaches and proved to be effective in terms of convergence speed, total time and path length.

The possibilities of planning the path of mobile robots in static and dynamic environments were investigated in [26]. A dynamic environment presents an additional challenge in the field of trajectory planning for mobile robots. To manage the dynamic environment, the proposed algorithm uses two selection operators: Elitist and Tournament and a new intelligent crossover operator and also five types of mutations. Chromosomes with variable length are used due to less memory usage. Population diversity is achieved by generating part of the initial population randomly and the rest of the population with one-point crossing to enable intelligent crossing. The proposed algorithm uses the memory with random immigrants technique to manage the dynamic environment. The implemented algorithm successfully finds suboptimal and optimal solutions in a short time compared to the known algorithms A* and MGA.

The trajectory path optimization of a mobile robot based on the principle of the robot's digital twin is a research paper proposed by Liu *et al.* [27]. The physical robot provides information about real movement data in space while the movement path is tested in the virtual environment and the virtual robot dynamically adapts to that path. Path optimization was performed with the help of a traditional GA, and the emphasis in the paper was placed on innovative (digital twin) technologies in the field of mobile robotics.

It is also possible to plan the path of a mobile robot based on image processing with the help of a visibility graph and a GA [28], which achieved valid results compared to other available methods for solving this problem. The GA starts the search from a binary image that represents the environment in which the robot will move. The visibility graph is generated based on key points that are calculated for each obstacle based on image processing and geometric measurements. Collision is avoided by using the image dilation operation. The optimization of the path of the mobile robot, which was created on the basis of the visibility graph, was carried out with the help of a GA. The disadvantage of this method was the trajectory with a greater number of obstacles on the way and lower image resolution, which requires a longer execution time.

Planning the material transportation by a mobile robot with the aim of achieving the shortest path without collision was solved by [29], according to the principle of the traveling salesman problem. An improved adaptive genetic simulated annealing algorithm (IAGSAA) is proposed, which enables the avoidance of local optimal solutions and premature solutions and improved generation of the initial population. The proposed algorithm IAGSAA through the mutation phase uses a simulated annealing strategy to accelerate convergence, and adaptive crossover and mutation enable population diversity. The test results showed that IAGSAA is stable because the frequency of optimal solutions increased and the standard deviation of the path length decreased. Also, the generation of the initial population has been improved compared to the standard GA. The proposed IAGSAA includes the advantages of the adaptive GA and the genetic simulated annealing algorithm and achieves good results even though it is somewhat more complex than the mentioned algorithms.

In the paper [30], an improved GA was proposed for solving the robot path planning problem in the case when the path is not smooth or has too many turns. The improvement is reflected in the increase in convergence speed by adding an elite retention strategy compared to the basic GA. Evaluation objectives such as turn angle and turn time are introduced, and crossover and mutation probabilities are optimized. Through testing, it was determined that the improved algorithm reduces the number of turns on the path and the turning angle, also reduces the number of starts and accelerates the convergence of the algorithm. The proposed algorithm is suitable for a static environment with obstacles and significantly reduces the energy consumption of the robot.

In a dynamic environment, navigating with a mobile robot is a challenge given the obstacles on the way. The successful implementation of the adapted GA according to [31] enables finding the optimal path with the help of a controller that ensures movement from the start point to the target point without collision. The role of the controller is to avoid obstacles in the environment and enable the planning of a new optimal path in order to avoid a collision. Virtual walls have been created to avoid repeating the path that needs to be crossed. Through testing, the algorithm proved to be robust and flexible and is intended for implementation in a real environment and application on the Team AmigoBot™ robot.

Path planning of a mobile robot in a static environment with and without obstacles was considered using a GA to optimize the path length. In the obstacle-free study [32], the robot was allowed to visit the same point more than once, and GA was used to select the optimal path. The algorithm was tested with different numbers of population size 10, 20, and 50, and the proposed algorithm found the optimal path in all cases. In the obstacle study, three different environments with various obstacle locations were considered (indoor environment, moderately scattered environment, complex scattered environment). Through testing, it was determined that with an increase in the number of generations, the algorithm finds the shortest path, but the calculation is more complex.

In the study proposed in [33], a modified GA was developed for globally planned trajectories during mobile robot navigation. The specificity of the algorithm is that no mutation operator was used, thus reducing the execution time. The algorithm efficiently steers the robot from start to finish without collision along an optimal or suboptimal path by reading the environment map. The proposed algorithm achieves optimal solutions in complex environments, but also in closed environments (there is no path between the starting position and the goal).

Research by Li *et al.* [34] propose an improved multi-objective GA to solve the problem of defining the trajectory of mobile robots in a static global environment. The generation of the initial population is done with the help of the heuristic method of median insertion, which improves the quality of the initial population and accelerates the convergence of the algorithm. Selection, crossover and mutation operators are created in such a way that they use layered methods (single-point crossover method, and the eight-neighborhood-domain single-point mutation method). Indicators were used to establish a multi-criteria objective function: length of path, safety and energy consumption. As the main advantages of the proposed algorithm, it is possible to single out the avoidance of getting stuck in a local optimum and the shortening of the optimal path length by 17% in relation to the classical GA method.

In addition to minimizing the trajectory of the mobile robot and reducing the distance from the starting point to the target point, optimization can also be carried out regarding to the smoothness of the path [35] which implies a minimum number of rotations. In this way, energy consumption is also reduced, because the rotations require a lot of power. In the paper, a modified GA is proposed, which aims to achieve the aforementioned. A traditional algorithm was used to generate the shortest path solution, and then three additional operators were added to ensure the improvement of path characteristics (path correction operator, vertex identification operator and gene redistribution operator). The algorithm was tested in the MATLAB environment and the results show that the minimization of path edges and path length was achieved with the improved approach.

In addition to the approaches mentioned, algorithms based on knowledge are also applied to solve the path optimization of mobile robots. The GA based on the knowledge [36], which uses heuristic knowledge in the form of special genetic operators to display the path of a mobile robot in a complex environment, successfully generates optimal solutions compared to the classic GA method (crossover and mutation operators are used). Tournament selection and elitism were applied. The proposed GA used the same objective function to determine the suitability of solutions (feasible and infeasible) with the help of which obstacles on the path are successfully overcome and collision is avoided. The algorithm is applicable to both static and dynamic environments. Knowledge-based GA are preferred when applied to specific problems.

3. THE DHOUIB-MATRIX-SPP

Tool path very recently, the first author developed a new optimization concept entitled Dhoub-Matrix (DM) where several novel heuristics, metaheuristics, and optimal methods (see Figure 1) are

developed based on matrix navigation. Hence, in this field three metaheuristics are designed and tested: the DM-4 in [37], the DM-3 in [38], and the Far-to-Near in [39]. Furthermore, several heuristics are invented in order to solve the travelling salesmen problem in [40], the assignment problem in [41] and the transportation problem in [42]. Besides, a greedy column-row method entitled DM-SPP to unravel the shortest path problem is considered in [43]-[46] and an optimal method to solve the minimum spanning tree problem is studied in [47].

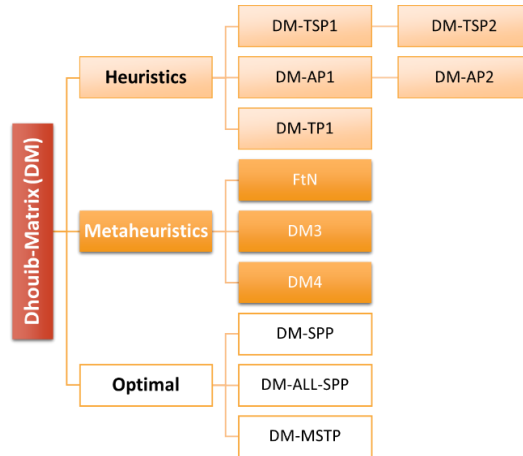


Figure 1. Setting for document template

In this paper the innovative DM-SPP method is compared to GA for the case of mobile robot path planning problems in static environment. DM-SPP is a greedy method based on column-row navigation through distance matrix. At first two additional matrices are added with a single list entitles respectively: The path-memory, the cumulative-cost, and the sum-cost. Indeed, the path-memory is used to archive the shortest movement between vertices, the cumulative-cost is beneficial for the navigation through the vertices and the sum-cost is an intermediate list helpful to simplify the transaction via the cumulative-cost. Figure 2 illustrates the five steps of DM-SPP (a step-by-step application of DM-SPP with more clarification are presented in [16]).

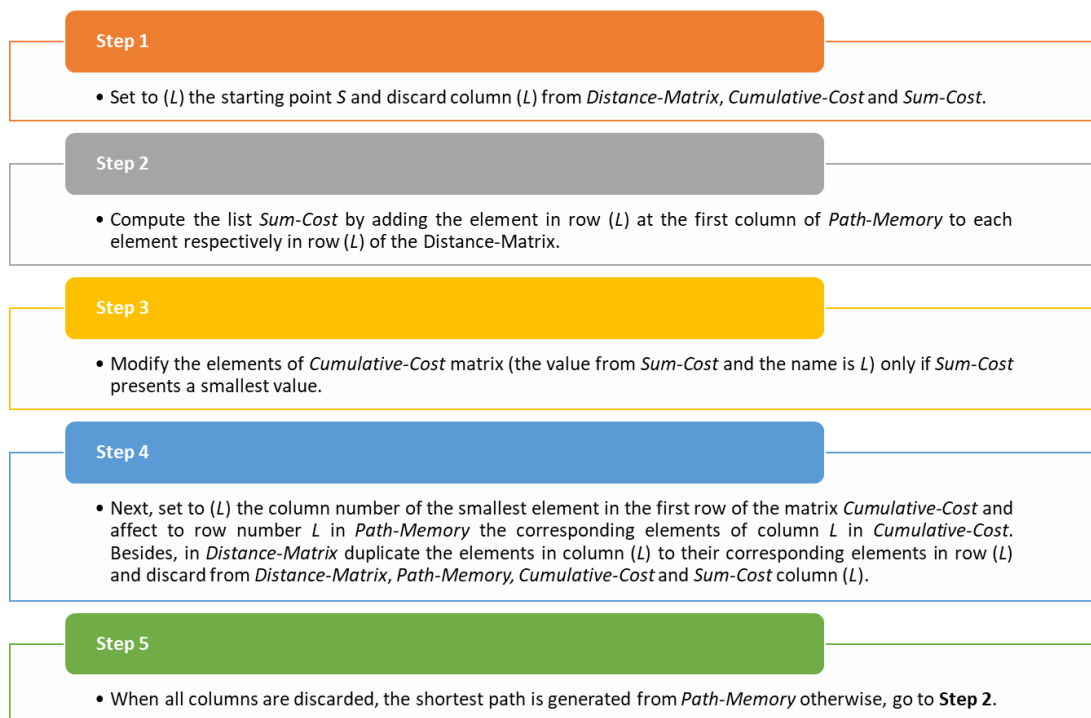


Figure 2. The five steps of DM-SPP method

4. COMPUTATIONAL RESULTS

In this section, the feasibility, performance and superiority of the proposed DM-SPP method is simulated on Python platform and contrasted to GA and its derivatives. Where a laptop is used with Intel® Core™ i7-1255U 1.70 GHz processor, 16.0 Go running memory and Windows 10 computer system environment. DM-SPP is deterministic method (it generates the same result in each iteration) however it is iterated fifty time in order to compute the average computation time (this time cover the time to create the Grid Map, the Graph and the optimal trajectory). DM-SPP is compared to:

- The standard genetic algorithm (SGA) developed by [17],
- The programmed cell death evolutionary algorithm (PCDA) developed by [17],
- The multi-population migration genetic algorithm (MPMGA) developed by [17],
- The traditional genetic algorithm (TGA) developed by [18],
- The improved genetic algorithm (IGA) developed by [18],
- The A* developed by [18],
- The GA developed by [19],
- The ant colony optimization (ACO) developed by [19],
- The GA combined with the ACO (GA-ACO) developed by [19],
- The hybrid method (AGAACO) developed by [19].

4.1. Case study 1: 25×25 grid map

This example is taken from [17] where a square (25×25) grid map is developed. This grid map represents the library area of (30×30 m) divided in to (625) squares of (1.2×1.2 m) each and the unfeasible areas (red squares) represent the walls, desks, and bookcases. The proposed DM-SPP method rapidly (in 0.0488 second) generates the shortest trajectory (30.38473) with eight neighbor movements (a clear representation of the generated solution by DM-SPP is depicted in Figure 3).

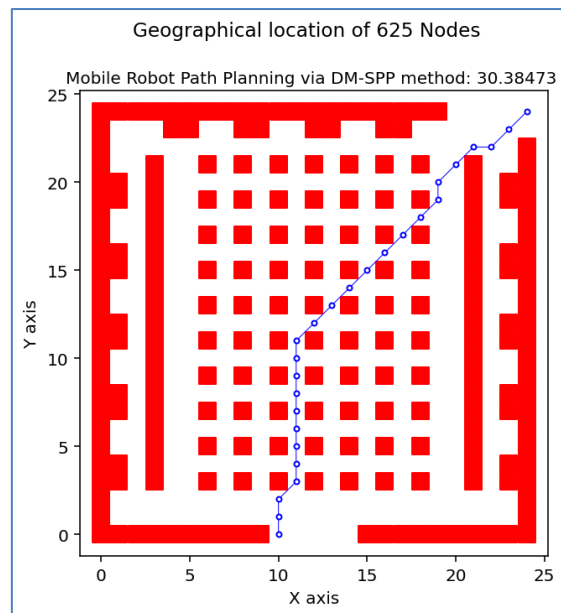


Figure 3. The solution generated by DM-SPP for the 25×25 case study

The generated result by DM-SPP and the other algorithms (SGA, PCDA, and MPMGA) developed by [17], are summarized in Table 1. Obviously, DM-SPP is the rapid method: DM-SPP represents an improvement of (4547.13%) to SGA, (3882.38%) to PCDA, and (5504.92%) to MPMGA (see Figure 4).

Table 1. Test results of the 25×25 grid map

Methods	Distance	Run time (s)	% distance	% CPU
SGA	32.5383	2.2678	7.09	4547.13
PCDA	31.5877	1.9434	3.96	3882.38
MPMGA	30.8085	2.7352	1.39	5504.92
DM-SPP	30.3847	0.0488	0.00	0.00

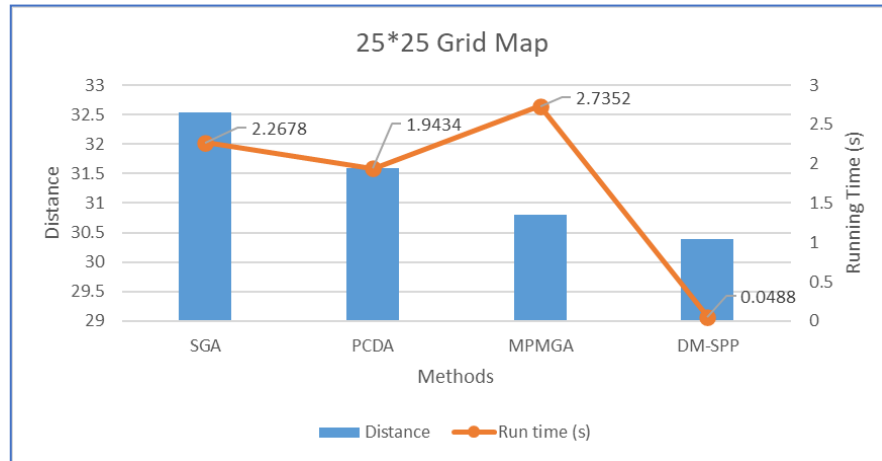


Figure 4. Comparing DM-SPP to other methods developed in the literature using two distance and running time evaluation criteria

4.2. Case study 2: 50x50 grid map

Let us consider a second environment (see Figure 5) represented by (50x50) grid map originally developed by [17]. DM-SPP quickly (in 0.5408 second) creates the trajectory (73.3968). The Table 2 gathers all generated result by DM-SPP and the other algorithms (SGA, PCDA, and MPMGA) developed by [17]. Once more, DM-SPP is the faster method: it gives a percentage improvement of (18786.58%) to SGA, (17746.84%) to PCDA, and (22198.56%) to MPMGA (see Figure 6). An obstacle is added (let assume it is suddenly find), DM-SPP will modify the trajectory of the mobile robot as its represented in Figure 7.

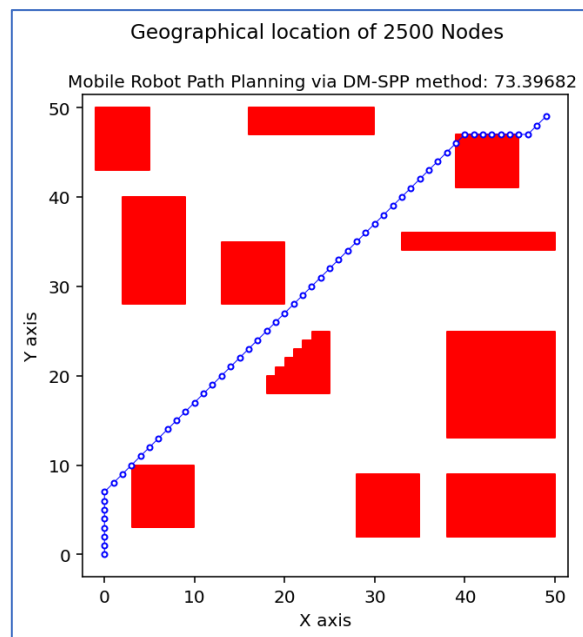


Figure 5. The solution generated by DM-SPP for the 50x50 case study

Table 2. Test results of the 50x50 grid map

Methods	Distance	Run time (s)	% distance	% CPU
SGA	76.9703	102.1386	4.87	18786.58
PCDA	76.2673	96.5157	3.91	17746.84
MPMGA	72.6975	120.5906	-0.95	22198.56
DM-SPP	73.3968	9.3526	0.00	0.00

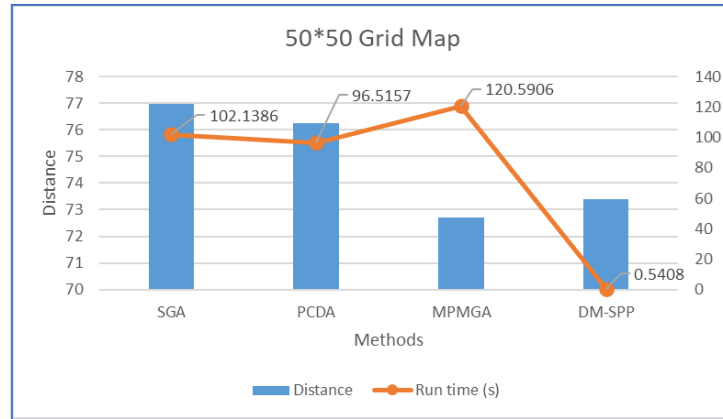


Figure 6. Comparing DM-SPP to SGA, PCDA, and MPMGA methods

4.3. Case study 3: 30×30 grid map

Another example (30×30 grid map) is considered in this section. This example was studied in [18] where three methods are developed (TGA, IGA, and A*). The trajectory generated by DM-SPP is clearly illustrated in Figure 8.

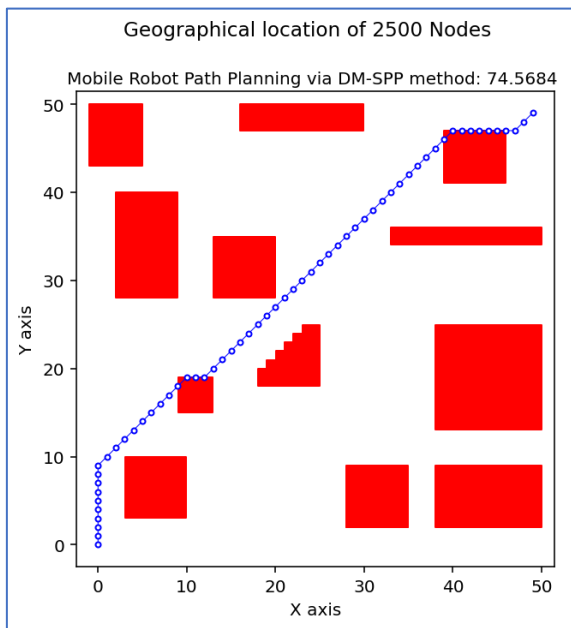


Figure 7. The solution generated by DM-SPP for the 50×50 case study with added obstacle

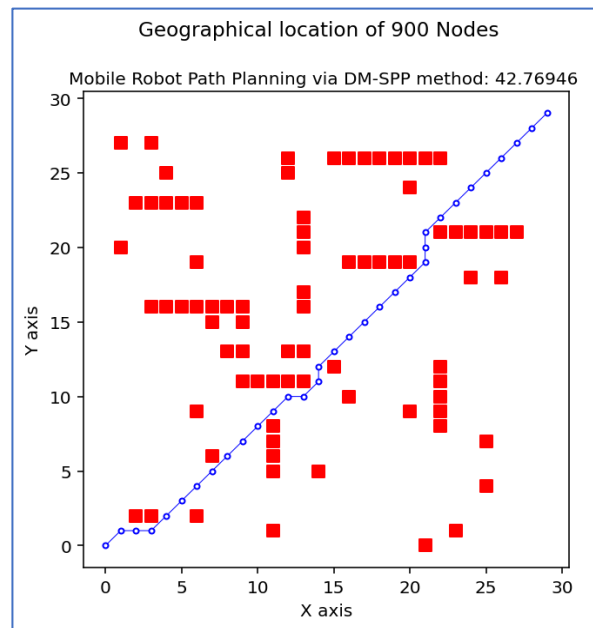


Figure 8. The solution generated by DM-SPP for the 30×30 case study

DM-SPP quickly generates the optimal solution (42.769) in (0.096) second which is represents a speed improvement of (907.29%) to the TGA, (531.25%) to the IGA, and (933.33%) to the A*. Table 3 gathers all results generated by DM-SPP and other methods developed in literature. Another time, DM-SPP generates the best accurate solution (see Figure 9).

Table 3. Test results of the 30×30 grid map

Methods	Distance	Run time (s)	% distance	% CPU
TGA	43.941	0.967	2.740	907.29
IGA	42.769	0.606	0.000	531.25
A*	43.355	0.992	1.370	933.33
DM-SPP	42.769	0.096	0.000	0.00

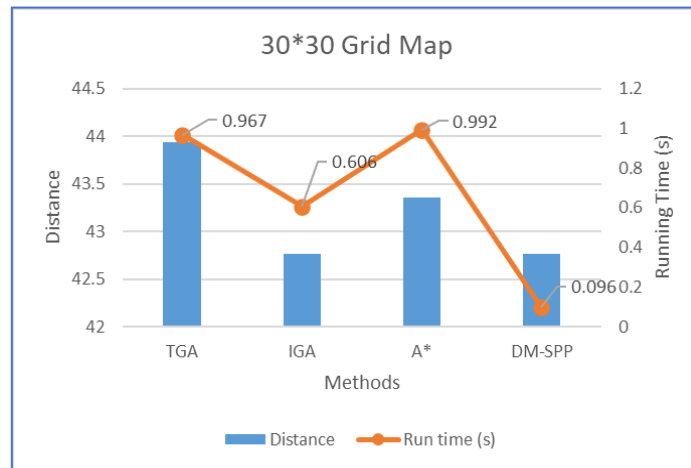


Figure 9. Comparing DM-SPP to TGA, IGA, and A* methods

4.4. Case study 4: 20x20 simple grid map

In this section, a (20x20) simple grid map is taken from [19]. The trajectory generated by DM-SPP is clearly illustrated in Figure 10. DM-SPP quickly generates the optimal solution (28.041) in (0.02) second. While, AGAACO generates the minimal solution (27.616) but it requires (13.59) second and it is not stable method (the average distance is 39.32). Whereas, DM-SPP is rapid (generate the solution (28.041) after just (0.02) second) and robust method (DM-SPP is deterministic technique with null standard deviation). Table 4 gathers all results generated by DM-SPP and other methods developed in literature. Another time, DM-SPP generates rapidly the best accurate solution (see Figure 11). Again, DM-SPP generates a huge improvement percentage: (86650.00%) to GA and (80950.00%) to ACO.

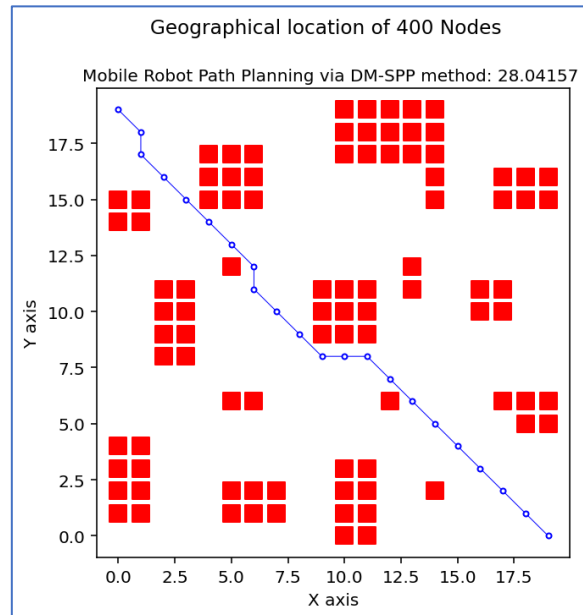


Figure 10. The solution generated by DM-SPP for the (20x20) simple case study

Table 4. Test results of the simple grid map

Methods	Distance	Run time (s)	% distance	% CPU
GA	45.26	17.35	61.41	86650.00
ACO	50.45	16.21	79.92	80950.00
GA-ACO	44.67	14.68	59.31	73300.00
AGAACO	39.32	13.59	40.23	67850.00
DM-SPP	28.04	0.02	0.00	0.00

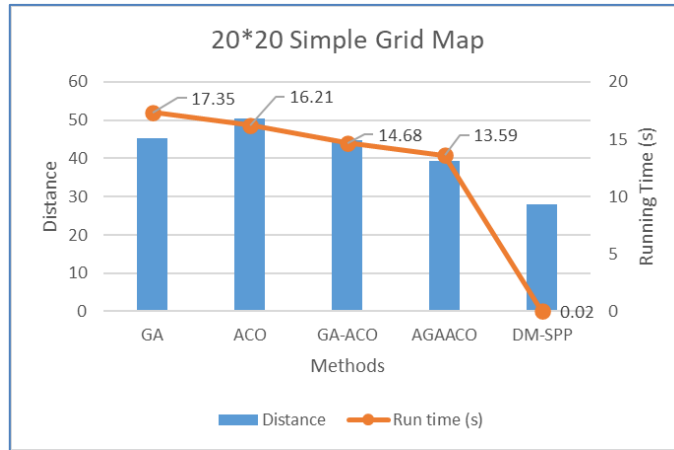


Figure 11. Comparing DM-SPP to GA, ACO, GA-ACO, and AGAACO methods for simple environment

4.5. Case study 5: 20×20 complex rid map

The example of (20×20 complex grid map) is considered. This example was studied in [19], where four methods are tested (GA, ACO, GA-ACO, and AGAACO). The trajectory generated by DM-SPP is clearly illustrated in Figure 12.

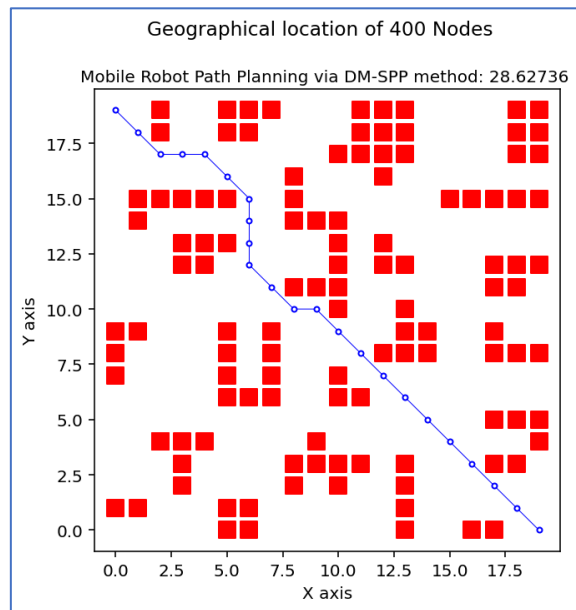


Figure 12. The solution generated by DM-SPP for the (20×20) complex case study

DM-SPP quickly generates the optimal solution (28.627) in (0.02) second. Table 5 gathers all results generated by DM-SPP and other methods developed in literature. Obviously, DM-SPP is the fastest method. Once more, DM-SPP generates the best accurate solution rapidly (see Figure 13) with a great CPU improvement of (99200%) to GA, (92600.00%) to ACO, (83800.00%) to GA-ACO, and (73250.00%) to AGAACO.

Table 5. Test results of the complex envirnment

Methods	Distance	Run time (s)	% distance	% CPU
GA	49.87	19.86	74.19	99200.00
ACO	49.32	18.54	72.27	92600.00
GA-ACO	45.37	16.78	58.47	83800.00
AGAACO	43.78	14.67	52.92	73250.00
DM-SPP	28.63	0.02	0.00	0.00

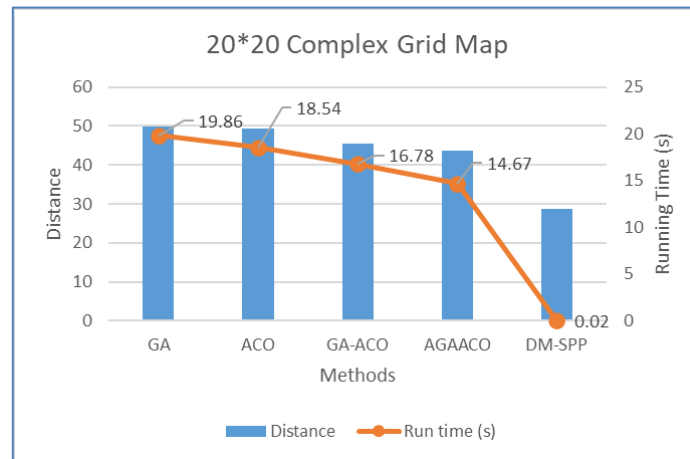


Figure 13. Comparing DM-SPP to GA, ACO, GA-ACO, and AGAACO methods for complex environment

To summarize this section: DM-SPP outperforms GA because it is deterministic method using zero parameter. Whereas, GA is stochastic method using several parameters (population size, number of generations or iterations, crossover probability and mutation probability). In the first case study (25×25 grid map), DM-SPP method was compared to the SGA [17], and achieves more favorable results considering the trajectory length of the mobile robot, i.e., it finds the shortest feasible trajectory with an improvement compared to SGA in the amount of 7.09%. DM-SPP also achieves a significantly shorter run time (CPU), i.e. it achieves the optimal solution in 0.0488 s, which represents an improvement of 4547.13% compared to SGA.

In the second case study (50×50 grid map) that was conducted, DM-SPP also achieves better results (improvement in the amount of 4.87% regarding to the distance) compared to the results of tested SGA method according to [17], and the time to reach the optimal solution is significantly faster (18786.58 expressed in percentages).

The third case study (30×30 grid map) includes a comparison of results with the TGA method and the IGA method according to [18]. The results show that DM-SPP achieves equal results compared to the improved algorithm (IGA) considering of trajectory length, but faster finds the optimal solution with (531.25%) improvement. DM-SPP achieves the optimal solution with an improvement of (2.74%) compared to the traditional method (TGA). DM-SPP also finds the optimal solution faster than TGA (907.29 expressed as a percentage).

The fourth case study (20×20 simple grid map) according to [19] compares the results of DM-SPP with the results achieved by the GA method. DM-SPP finds the shortest feasible trajectory with an improvement compared to GA in the amount of (61.41%). Run time (CPU) is also significantly shorter achieved with DM-SPP and is more favorable by (86650%) compared to the GA method.

The fifth study (20×20 complex grid map) proved to be the most successful in terms of results for the DM-SPP method, where the result of finding the shortest trajectory was (74.19%) more favorable than the GA method developed in [19]. Run time (CPU) is also significantly shorter achieved with DM-SPP and is more favorable by (99200%) compared to the performed GA. From the above experimental results, it can be concluded that DM-SPP is significantly accurate and faster than GA.

5. CONCLUSION

In the field of mobile robotics, an important task is to find the optimal path (at the same time, it is possible to optimize in terms of the shortest path, reduced energy consumption or the shortest time spent) from the starting point to the destination while avoiding collision (ensuring a safe path). Various methods are used to solve this type of problem, and in this paper, an overview of previous research in the field of path planning in mobile robotics with the help of the GA method (traditional and modified) is given. Also, a new heuristic method DM-SPP was presented, which characteristics were compared with the aforementioned characteristics of the GA, and the obtained results were analyzed. DM-SPP is an efficient method, in terms of efficiency and speed of convergence, which doesn't require any of parameters in relation to the GA method. For instance, GA method for the successful operation needs the selection of genetic operators and parameters which determine the behavior of these operators. For this reason, DM-SPP proved to be a fast method and simpler to implement (DM-SPP over GA in terms of efficiency, convergence speed, and ease of

implementation). DM-SPP achieves optimal with a significantly lower number of iterations compared to the aforementioned GA method, which can be seen from the analysis of different cases from published research. In this paper, the DM-SPP method is studied based on a single objective function (the shortest distance) and in statistical environment. Accordingly, the simulation of DM-SPP for the case of multi-objective, dynamic environments, multi-robot systems and hardware implementation will be conducted in depth next studies.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Souhail Dhouib	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Danijela Pezer	✓	✓			✓	✓			✓	✓	✓	✓	✓	✓
Tole Sutikno	✓	✓			✓	✓			✓	✓	✓	✓	✓	✓

C : Conceptualization

M : Methodology

So : Software

Va : Validation

Fo : Formal analysis

I : Investigation

R : Resources

D : Data Curation

O : Writing - Original Draft

E : Writing - Review & Editing

Vi : Visualization

Su : Supervision

P : Project administration

Fu : Funding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author, upon reasonable request.

REFERENCES




- [1] Ş. Yıldırım and S. Savaş, "Design of a Mobile Robot to Work in Hospitals and Trajectory Planning Using Proposed Neural Networks Predictors," in *Cioboată, International Conference on Reliable Systems Engineering (ICORSE), Lecture Notes in Networks and Systems*, 2022, pp. 32–45, doi: 10.1007/978-3-030-83368-8_4.
- [2] A. Davids, "Urban Search and Rescue Robots: From Tragedy to Technology," *IEEE Intelligent Systems*, vol. 17, no. 2, pp. 81–83, Mar. 2002, doi: 10.1109/MIS.2002.999224.
- [3] C. Sampedro, A. Rodriguez-Ramos, H. Bavlé, A. Carrio, P. de la Puente, and P. Campoy, "A Fully-Autonomous Aerial Robot for Search and Rescue Applications in Indoor Environments using Learning-Based Techniques," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 95, no. 2, pp. 601–627, Aug. 2019, doi: 10.1007/s10846-018-0898-1.
- [4] V. Digani, L. Sabattini, C. Secchi, and C. Fantuzzi, "Ensemble Coordination Approach in Multi-AGV Systems Applied to Industrial Warehouses," *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 3, pp. 922–934, 2015, doi: 10.1109/TASE.2015.2446614.
- [5] G. Fragapane, R. de Koster, F. Sgarbossa, and J. O. Strandhagen, "Planning and control of autonomous mobile robots for intralogistics: Literature review and research agenda," *European Journal of Operational Research*, vol. 294, no. 2, pp. 405–426, 2021, doi: 10.1016/j.ejor.2021.01.019.
- [6] E. Falcone, R. Gockley, E. Porter, and I. Nourbakhsh, "The personal rover project: The comprehensive design of a domestic personal robot," *Robotics and Autonomous Systems*, vol. 42, no. 3–4, pp. 245–258, 2003, doi: 10.1016/S0921-8890(02)00379-2.
- [7] P. K. Baghel and A. K. Mishra, "A New Approach for a Car Navigation System," *International Research Journal of Engineering and Technology (IRJET)*, vol. 03, no. 09, pp. 377–381, 2016.
- [8] S. Mirjalili, "Ant Colony Optimization," *Studies in Computational Intelligence*, vol. 780, pp. 33–42, 2019.
- [9] X. P. Zeng, Y. M. Li, and J. Qin, "A dynamic chain-like agent genetic algorithm for global numerical optimization and feature selection," *Neurocomputing*, vol. 72, no. 4–6, pp. 1214–1228, 2009, doi: 10.1016/j.neucom.2008.02.010.
- [10] D. Wang, D. Tan, and L. Liu, "Particle swarm optimization algorithm: an overview," *Soft Computing*, vol. 22, no. 2, pp. 387–408, 2018, doi: 10.1007/s00500-016-2474-6.
- [11] A. Ammar, H. Bennaceur, I. Châari, A. Koubâa, and M. Alajlan, "Relaxed Dijkstra and A* with linear complexity for robot path planning problems in large-scale grid environments," *Soft Computing*, vol. 20, no. 10, pp. 4149–4171, 2016, doi: 10.1007/s00500-015-1750-1.
- [12] S. Koenig, M. Likhachev, and D. Furcy, "Lifelong Planning A*," *Artificial Intelligence*, vol. 155, no. 1–2, pp. 93–146, May 2004,

- doi: 10.1016/j.artint.2003.12.001.
- [13] S. M. Persson and I. Sharf, "Sampling-based A* algorithm for robot path-planning," *International Journal of Robotics Research*, vol. 33, no. 13, pp. 1683–1708, Nov. 2014, doi: 10.1177/0278364914547786.
- [14] H. Hoang, A. K. Tran, L. N. T. Tran, M. H. Le, and D. T. Tran, "A shortest smooth-path motion planning for a mobile robot with nonholonomic constraints," in *2021 International Conference on System Science and Engineering (ICSSE)*, Ho Chi Minh City, Vietnam: IEEE, Aug. 2021, pp. 145–150, doi: 10.1109/ICSSE52999.2021.9538414.
- [15] W. Wang, H. Deng, and X. Wu, "Path planning of loaded pin-jointed bar mechanisms using Rapidly-exploring Random Tree method," *Computers and Structures*, vol. 209, pp. 65–73, 2018, doi: 10.1016/j.compstruc.2018.08.016.
- [16] S. Dhouib, "An optimal method for the Shortest Path Problem: The Dhouib-Matrix-SPP (DM-SPP)," *Results in Control and Optimization*, vol. 12, pp. 1–12, Sep. 2023, doi: 10.1016/j.rico.2023.100269.
- [17] J. Ma, Y. Liu, S. Zang, and L. Wang, "Robot Path Planning Based on Genetic Algorithm Fused with Continuous Bezier Optimization," *Computational Intelligence and Neuroscience*, vol. 25, pp. 1–10, Feb. 2020, doi: 10.1155/2020/9813040.
- [18] K. Hao, J. Zhao, K. Yu, C. Li, and C. Wang, "Path planning of mobile robots based on a multi-population migration genetic algorithm," *Sensors*, vol. 20, no. 20, pp. 1–23, Oct. 2020, doi: 10.3390/s20205873.
- [19] X. Zhang, Y. Zhao, N. Deng, and K. Guo, "Dynamic Path Planning Algorithm for a Mobile Robot Based on Visible Space and an Improved Genetic Algorithm," *International Journal of Advanced Robotic Systems*, vol. 13, no. 3, May 2016, doi: 10.5772/63484.
- [20] D. Sahu and A. K. Mishra, "Mobile robot path planning by genetic algorithm with safety parameter," *International Journal of Engineering Science and Computing*, vol. 7, no. 8, pp. 14723–14727, 2017.
- [21] C. Lamini, S. Benhlama, and A. Elbekri, "Genetic Algorithm Based Approach for Autonomous Mobile Robot Path Planning," *Procedia Computer Science*, vol. 127, pp. 180–189, 2018, doi: 10.1016/j.procs.2018.01.113.
- [22] W. Rahmaniari and A. E. Rakhmania, "Mobile Robot Path Planning in a Trajectory with Multiple Obstacles Using Genetic Algorithms," *Journal of Robotics and Control (JRC)*, vol. 3, no. 1, pp. 1–7, 2022, doi: 10.18196/jrc.v3i1.11024.
- [23] K. K. Kwaśniewski and Z. Gosiewski, "Genetic algorithm for mobile robot route planning with obstacle avoidance," *Acta Mechanica et Automatica*, vol. 12, no. 2, pp. 151–159, 2018, doi: 10.2478/ama-2018-0024.
- [24] D. Kang, H. Hashimoto, and F. Harashima, "Path Generation for Mobile Robot using Genetic Algorithm," *IEEE Transactions on Electronics, Information and Systems*, vol. 117, no. 2, pp. 102–109, 1997, doi: 10.1541/ieejieiss1987.117.2_102.
- [25] P. Bhanaria, M. Pachori, and P. K. Pandey, "Modified Genetic Algorithm-based Robot Path Planning to Avoid Static Obstacles Collision," *New Frontiers in Communication and Intelligent Systems*, pp. 811–817, 2023, doi: 10.52458/978-81-95502-00-4-79.
- [26] H. Alsouly and H. Bennaceur, "Enhanced genetic algorithm for mobile robot path planning in static and dynamic environment," in *IJCCI 2016 - Proceedings of the 8th International Joint Conference on Computational Intelligence*, 2016, pp. 121–131, doi: 10.5220/0006033401210131.
- [27] X. Liu *et al.*, "Genetic Algorithm-Based Trajectory Optimization for Digital Twin Robots," *Frontiers in Bioengineering and Biotechnology*, vol. 9, Jan. 2022, doi: 10.3389/fbioe.2021.793782.
- [28] F. M. Santa, F. H. M. Sarmiento, and H. M. Ariza, "Path Optimization for Mobile Robots using Genetic Algorithms," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 13, no. 2, pp. 657–662, 2022, doi: 10.14569/IJACSA.2022.0130277.
- [29] J. Jiang, X. Yao, E. Yang, and J. Mehnen, "An Improved Adaptive Genetic Algorithm for Mobile Robot Path Planning Analogous to TSP with Constraints on City Priorities," in *IEEE World Congress on Computational Intelligence 2020*, 2020.
- [30] Z. Guo and L. Mo, "Robot path planning based on improved genetic algorithm," *Frontiers in Computing and Intelligent Systems*, vol. 2, no. 3, pp. 78–81, Feb. 2023, doi: 10.54097/fcis.v2i3.5314.
- [31] S. C. Yun, S. Parasuraman, and V. Ganapathy, "Dynamic path planning algorithm in mobile robot navigation," in *2011 IEEE Symposium on Industrial Electronics and Applications*, Langkawi, Malaysia: IEEE, Sep. 2011, pp. 364–369, doi: 10.1109/ISIEA.2011.6108732.
- [32] I. Al-Taharwa, A. Sheta, and M. Al-Weshah, "A mobile robot path planning using genetic algorithm in static environment," *Journal of Computer Science*, vol. 4, no. 4, pp. 341–344, Apr. 2008, doi: 10.3844/jcssp.2008.341.344.
- [33] N. A. Shiltagh, K. S. Ismail, and Z. Q. Habeeb, "Best Path Planning Algorithm for Mobile Robot Based on Modified Genetic Algorithm," *Engineering and Technology Journal*, vol. 32, no. 4, pp. 986–1006, 2014, doi: 10.30684/etj.32.4a.14.
- [34] K. Li, Q. Hu, and J. Liu, "Path Planning of Mobile Robot Based on Improved Multiobjective Genetic Algorithm," *Wireless Communications and Mobile Computing*, 2021, doi: 10.1155/2021/8836615.
- [35] F. Abushmala and I. Abuhadrous, "Adaptable Genetic Algorithm Path Planning of Mobile Robots Based on Gene Reallocating," *Advances*, vol. 3, no. 3, pp. 125–131, 2022, doi: 10.11648/j.advances.20220303.22.
- [36] Yanrong Hu and S. X. Yang, "A knowledge based genetic algorithm for path planning of a mobile robot," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, New Orleans, LA, USA: IEEE, 2004, pp. 4350–4355, doi: 10.1109/ROBOT.2004.1302402.
- [37] S. Dhouib, "Multi-Start Constructive Heuristic Through Descriptive Statistical Metrics: The Dhouib-Matrix-4 (DM4) Metaheuristic," *International Journal of Operational Research*, vol. 50, no. 2, 2024, doi: 10.1504/ijor.2021.10045069.
- [38] S. Dhouib, "Novel Metaheuristic Based on Iterated Constructive Stochastic Heuristic: Dhouib-Matrix-3 (DM3)," *Applied Computational Intelligence and Soft Computing*, pp. 1–10, Dec. 2021, doi: 10.1155/2021/7761993.
- [39] S. Dhouib, "Hole Drilling Route Optimization in Printed Circuit Boards Using Far-to-Near Metaheuristics," *International Journal of Strategic Engineering*, vol. 5, no. 1, pp. 1–12, Jun. 2022, doi: 10.4018/ijose.301568.
- [40] S. Dhouib, A. Zouari, S. Dhouib, and H. Chabchoub, "Integrating the artificial bee colony metaheuristic with Dhouib-Matrix-TSP1 heuristic for holes drilling problems," *Journal of Industrial and Production Engineering*, vol. 40, no. 3, pp. 177–187, Apr. 2023, doi: 10.1080/21681015.2022.2158499.
- [41] S. Dhouib, "Novel optimization method for unbalanced assignment problems with multiple jobs: The Dhouib-Matrix-AP2," *Intelligent Systems with Applications*, vol. 17, pp. 1–8, Feb. 2023, doi: 10.1016/j.iswa.2023.200179.
- [42] S. Dhouib, "Solving the Single-Valued Trapezoidal Neutrosophic Transportation Problems through the Novel Dhouib-Matrix-TP1 Heuristic," *Mathematical Problems in Engineering*, pp. 1–11, Nov. 2021, doi: 10.1155/2021/3945808.
- [43] S. Dhouib, "Shortest path planning via the rapid Dhouib-Matrix-SPP (DM-SPP) method for the autonomous mobile robot," *Results in Control and Optimization*, vol. 13, pp. 1–12, Dec. 2023, doi: 10.1016/j.rico.2023.100299.
- [44] S. Dhouib, N. Béji, D. Kallel, and S. Dhouib, "A Computational Time Analysis of Dhouib-Matrix-SPP versus Particle Swarm Optimization Metaheuristics for Grid-based Path Planning," *Statistics, Optimization and Information Computing*, vol. 15, no. 4, pp. 2575–2586, Jan. 2026, doi: 10.19139/soic-2310-5070-3259.
- [45] S. Dhouib and H. Chabchoub, "Optimizing the Shortest and Safety Pathways in Infectious Disease Context via the Novel Dhouib-Matrix-SPP Method," *Applied Computational Intelligence and Soft Computing*, no. 1, pp. 1–8, Jan. 2026, doi: 10.1155/acis/4490913.




- [46] S. Dhoui, "Fast Method for the Mobile Robot Path Planning Problem: The DM-SPP Method," *Statistics, Optimization and Information Computing*, vol. 15, no. 1, pp. 516–528, Dec. 2025, doi: 10.19139/soic-2310-5070-3013.
- [47] S. Dhoui, "Innovative method to solve the minimum spanning tree problem: The Dhoui-Matrix-MSTP (DM-MSTP)," *Results in Control and Optimization*, vol. 14, pp. 1–12, Mar. 2024, doi: 10.1016/j.rico.2023.100359.

BIOGRAPHIES OF AUTHORS






Souhail Dhoui    is a full professor at the University of Sfax, Tunisia. He is listed as one of the World's Top 2% Scientists in the main field "Information & Communication Technologies", the first subfield of "Artificial Intelligence" and the second subfield of "Operations Research" (by Stanford University and Elsevier). He is the inventor of a new optimization concept, namely the "Dhoui-Matrix", under which several innovative artificial intelligence methods are developed. He can be contacted at email: souhail.dhoui@gmail.com.



Danijela Pezer    is a senior lecturer at the University Department of Professional Studies at the University of Split, Split, Croatia. She is currently the Head of the Mechanical Engineering Department. She received the award for the best scientific paper in the category of Young Researchers (Dubrovnik, Croatia, 2017) and also for the best paper presentation (Kecskemét, Hungary, 2014) as Young Researcher. She is also a member of Editorial and Reviewer Board in several International Scientific Journals. Her research interests include the fields of artificial intelligence, optimization, polymer processing, and additive manufacturing. She can be contacted at email: dpezer@oss.unist.hr.



Prof. Ir. Tole Sutikno, Ph.D., MIET, IPM., ASEAN Eng.    is a full professor in the Department of Electrical Engineering at Universitas Ahmad Dahlan (UAD) in Yogyakarta, Indonesia. He has held this position since 2023, having previously served as an associate professor from 2008. He earned his bachelor's degree from Universitas Diponegoro in 1999, his master's degree from Universitas Gadjah Mada in 2004, and his Ph.D. in Electrical Engineering from Universiti Teknologi Malaysia in 2016, where his doctoral research focused on advanced digital power electronics and intelligent control systems. From 2016 to 2021, he served as the Director of the Institute for Scientific Publishing and Publications (LPPI) at UAD, where he led initiatives to strengthen research visibility, journal management, and international collaboration in scholarly publishing. Since 2024, he has served as the head of the master's program in electrical engineering at UAD, following his leadership of the undergraduate program in electrical engineering in 2022. He is also the founding leader of the Embedded Systems and Power Electronics Research Group (ESPERG), which actively collaborates with both national and international institutions on topics such as fault-tolerant embedded systems, FPGA-based control, and renewable energy integration. He is widely acknowledged for his contributions to digital design, industrial electronics, motor drives, robotics, intelligent systems, and AI-based automation. His interdisciplinary research emphasises practical deployments in industrial and healthcare contexts, covering FPGA applications, embedded systems, power electronics, and digital libraries. He has published 380 peer-reviewed articles in high-impact journals and conferences indexed by Scopus. As of 2025, Google Scholar indicates over 6,000 citations, with an h-index of 36 and an i10-index of 174. In recognition of his global research impact, he has been listed among the Top 2 of Scientists Worldwide by Stanford University and Elsevier BV from 2021 to the present, a distinction based on standardized citation metrics across all scientific disciplines. He can be contacted at email: tole@te.uad.ac.id.