UDC 004
DOI: 10.15587/1729-4061.2025.323746

DEVELOPMENT OF AUTONOMOUS VEHICLE NAVIGATION IN UNSTRUCTURED ENVIRONMENTS: THE IMPACT OF IMPLEMENTING A PATH PLANNING ALGORITHM ON AUTONOMOUS VEHICLES

Desi Windi Sari
Doctor of Electrical Engineering, Student
Doctoral Program in Engineering Science**
Suci Dwijayanti
Doctor of Electrical Engineering, Associate Professor*
Bhakti Yudho Suprapto
Corresponding author
Doctor of Electrical Engineering, Associate Professor*
E-mail: bhakti@ft.unsri.ac.id
*Department of Electrical Engineering**
**Universitas Sriwijaya
Palembang-Prabumulih str., KM 32,
Inderalaya, Indonesia, 30662

The path planning system has been identified as an efficient method for optimizing navigation, reducing energy consumption, and ensuring safety in autonomous vehicles. Various studies have been conducted on algorithms such as Ant Colony Optimization (ACO), Artificial Potential Field (APF), and the A* algorithm. However, only a few studies have evaluated the effectiveness of each of these algorithms, especially for autonomous vehicles implemented in real-road scenarios. Thus, this study aims to assess the effectiveness of ACO, APF, and A* in identifying the optimal path, computational efficiency, and execution time for generating routes in an autonomous vehicle. Experiments were conducted using road coordinate data from the Universitas Sriwijaya campus, representing suburban road conditions in Indonesia. The results showed that the A* algorithm excels in finding optimal routes, with an average path length of 0.48 km and a 100 % success rate. This is due to its heuristic, Euclidean-based approach. Meanwhile, ACO achieved an average path length of 0.57 km with a 100 % success rate, whereas APF achieved 0.36 km with a 41 % success rate. ACO demonstrated varied route performance due to its probabilistic nature, while APF generated paths more quickly but often failed in complex environments due to local minimum traps. Regarding computation time, an increase in distance leads to a longer route formation time for APF and A*, respectively. However, in ACO, route distance does not directly determine the time required for route formation, as the algorithm incorporates a probability factor in the process. This study confirms that A* is more optimal for global path planning, whereas APF is better suited for local path planning. These findings provide valuable insights into the development of autonomous vehicle navigation in unstructured environments

Keywords: ant colony optimization, artificial potential fields, A*, autonomous vehicles, path planning

Received 28.01.2025
Received in revised form 12.03.2025
Accepted 02.04.2025
Published 29.04.2025

How to Cite: *Sari, D. W., Dwijayanti, S., Suprapto, B. Y. (2025). Development of autonomous vehicle navigation in unstructured environments: the impact of implementing a path planning algorithm on autonomous vehicles. Eastern-European Journal of Enterprise Technologies, 2 (3 (134)), 16–24. https://doi.org/10.15587/1729-4061.2025.323746*

1. Introduction

There are several factors to consider when designing an autonomous vehicle, including road detection, object detection, and navigation. Navigation, implemented through a path planning system, is a crucial component for autonomous vehicles to optimize energy usage and minimize negative environmental impacts [1, 2]. In this context, path planning algorithms determine the optimal route an agent should take from a starting point to a destination while minimizing specific criteria, such as distance or travel time, and avoiding obstacles in the environment.

Path planning can be classified into two categories based on the level of environmental information available: global path planning, which utilizes comprehensive prior knowledge, and local path planning, which relies on real-time sensor data [3–5]. Global and local path planning (Fig. 1) are distinguished by the availability of complete environmental information. In global path planning, the autonomous vehicle has full knowledge of the environment before starting. Conversely, in local path planning, vehicles have little to no prior environmental information before navigation begins [6].

Path planning for autonomous vehicles serves as a reference for navigation during operation. Battery-powered autonomous vehicles must consider the optimal route to conserve energy. To achieve the most accurate route planning, it is necessary to review various algorithms to ensure their implementation in real-time applications for autonomous vehicles, especially in unstructured suburban areas in Indonesia. In addition, if there is an obstacle on the road, the system must be able to handle it effectively. Therefore, such studies are scientific relevant for the implementation of autonomous vehicles.

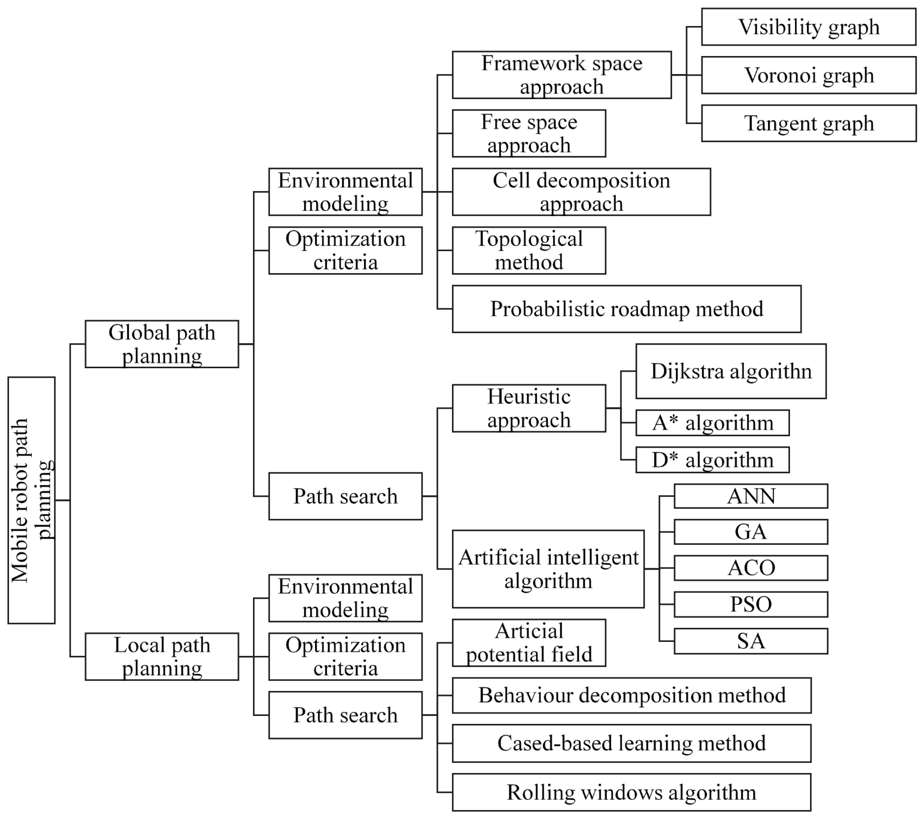


Fig. 1. Classification of path planning [6]

2. Literature review and problem statement

Several methods have been proposed for determining global path planning. The paper [7] applied ant colony optimization (ACO) together with fuzzy logic for global path planning in a mobile robot. This study effectively enhances the performance of the Ant Colony Optimization (ACO) algorithm in global path planning for mobile service robots. Through the integration of fuzzy control, optimized initial pheromone distribution, and a dynamic pheromone update and evaporation mechanism, the proposed algorithm demonstrates superior performance compared to conventional ACO in terms of convergence, path efficiency, and system stability. However, this study was based on simulations and has not been implemented in a real-world scenario. Additionally, the proposed algorithm was only compared with the original ACO.

Additionally, [8] developed a more efficient path planning method for AUVs in complex underwater environments. The integration of ACO for travelling sequence optimization and A* for local path planning provides a balance between efficiency and optimality of the solution. Nevertheless, this study was conducted in a static environment without considering dynamic obstacles.

Another study [9] compared the proposed algorithm with four other algorithms: JADE, TVPSO, Gravitational Search (GS), and Modified Genetic Algorithm (mGA) in various simulation scenarios and Monte-Carlo experiments. The simulation results show that HNTVPSO-RBSADE is superior in finding the optimal path compared to other methods. Monte-Carlo Experiment with 50 independent trials proved that this algorithm can generate more efficient paths with competitive computation time. However, the proposed method was only implemented in a mobile robot, and it was not compared with non-evolutionary methods.

Compared to TVPSO, it improves the path quality by 9.75 %, although TVPSO is about 1.38 % faster in computation time. while [10] proposed GA-ACO hybrid algorithm, from the simulation result this algorithm significantly improves the efficiency of intelligent vehicle path planning. By optimizing ACO and GA and combining them, it produces shorter, smoother, and faster paths than traditional methods. However, in highly complex environments, the reduction in the number of turns is not as effective as in simple scenarios.

Meanwhile [11] develops an improved A-Star algorithm with modifications to its heuristic function, testing result shows that this improve research successfully improves the efficiency of autonomous vehicle path planning by improving the A-Star heuristic function using the Artificial Potential Field method. The proposed algorithm is able to significantly reduce the number of turning points, thus improving the smoothness of vehicle motion. However, this method has a higher computational complexity than traditional A-Star, especially in more complex environments.

Besides global path planning, it is also important to consider local path planning when determining routes. The paper [12] integrating Membrane Computing, Evolutionary Algorithms, and Artificial Potential Fields (memEAPF) algorithm. Experiments were conducted in 12 benchmark environments, under both static and dynamic conditions. Tests were also conducted using parallel processing to assess the efficiency of the algorithm. The experiment result showed this research is able to produce more optimized paths in less computation time. This approach is proven to be effective in both static and dynamic environments, and can utilize parallel processing to improve efficiency. However, it was only implemented in simulations, so real-world validation is still needed. Another study, [13] integrated A* dan Artificial Potential Field (APF). Experiments were conducted in simulation using the MORAI simulator, which replicates the Chungbuk National University (CBNU) campus environment, and tested under straight road, winding road, and scenarios with static and dynamic obstacles. Experimental result showed this algorithm able to produce safer and smoother paths with processing times that still fulfil real-time requirements. Nevertheless, this study was focused on static simulated environments; thus, real-time performance should be explored. Meanwhile, [14] proposed a new approach, Augmented Reality-based APF (AR-APF), which extends the robot's perception by detecting the minimum possible local traps and creating virtual obstacles to direct it to a more optimal path. From the Experiments were conducted in a laboratory environment using Husarion ROSbot 2.0 PRO with Robot Operating System (ROS) proved this algorithm more effective than the standard APF in handling local minimum problems. By extending the robot's perception using Augmented Reality, this algorithm enables smoother and more efficient navigation. However, it was only tested in a small-scale environment, and real-time implementation should be considered. While [15] proposed an approach that combines APF, ACO, and Velocity Obstacle (VO) for submarine path planning in three-dimensional underwater scenarios. Testing in semi-physical simulation, this algorithm proved superior to previous methods in terms of global path planning and dynamic obstacle avoidance. The use of Velocity Obstacle Method allows submarines to avoid collisions more effectively. However, the results should be validated in the real world, especially in cases involving non-threatening dynamic obstacles.

From the previous studies mentioned above, ACO [7, 8, 10], APF [12–15], and A* [8, 11, 13] are the most widely used path planning algorithms. However, in these studies, testing has primarily been confined to simulations where paths are pre-determined and have not yet been directly applied to autonomous vehicles. Additionally, these investigations often take place in idealized and static environments. In real-world scenarios, autonomous vehicles navigate complex and dynamic environments, particularly in developing countries like Indonesia, where road infrastructure is still underdeveloped and construction is often haphazard, leading to irregularities, especially in suburban areas. Furthermore, global mapping applications such as Google Maps currently do not include small roads in residential areas or specific regions, resulting in a lack of research in the field of path planning in Indonesia. Thus, it is necessary to conduct a comprehensive comparison of these algorithms in suburban areas, such as those found in Indonesia.

3. The aim and objectives of the study

The aim of the study is to development of autonomous vehicle navigation in unstructured environments, focusing on their effectiveness in identifying the optimal path, computational efficiency, and execution time. To achieve this aim, the following objectives are accomplished:

– to evaluate the performance of three path planning algorithms: ACO, APF, and A* for implementation in autonomous vehicles;

– to utilize paths obtained from unstructured suburban areas in Indonesia.

4. Materials and methods

4. 1. Object and hypothesis of the study

The object of this study is the longitude and latitude data of a road at Universitas Sriwijaya, representing a suburban area in Indonesia. This road will later become the main route for the implementation of autonomous vehicles. Thus, this study is expected to be beneficial in determining the appropriate algorithm for path planning. It is assumed that this road will be obstructed by obstacles to evaluate the effectiveness of the three compared algorithms – ACO, APF, and A* – in finding the best route. The simplification of the obstacle is that the road is closed or impassable.

4. 2. Ant colony optimization (ACO)

The ant colony optimization (ACO) algorithm is a computational paradigm designed to solve optimization problems by drawing inspiration from the behavior of insect colonies, particularly ants. Initially introduced by Moyson and Manderick and later popularized by Marco Dorigo, ACO belongs to the category of bioinspired metaheuristics – algorithms influenced by natural phenomena and biological systems. Specifically, ACO tackles optimization problems by emulating the foraging behavior of ant colonies, making it distinct from other approaches [16, 17].

This algorithm is primarily used to solve combinatorial optimization problems. The key processes of the ACO algorithm include:

– solution construction: artificial ants build solutions incrementally by traversing paths within the solution space. The selection of the next element in the solution is probabilistic, influenced by pheromone trails and heuristic information;

– pheromone update: after all ants have completed solution construction, pheromone trails are updated to reflect solution quality. This involves the evaporation of existing pheromones and the deposition of new pheromones based on the quality of the solutions found.

The probability of an ant to move from one point (i) to another point (j) is given by the following equation:

$$P_{ij}^{K}(t)=\begin{cases}\dfrac{\tau_{ij}^{\alpha}(t)\eta_{ij}^{\beta}(t)}{\sum_{s\in allowed}\tau_{is}^{\alpha}(t)\eta_{is}^{\beta}(t)}, s\in \text{allowed}_k,\\ 0,\ \ s\in \text{allowed}_k,\end{cases} \quad (1)$$

where α and β are the weights of two heuristic functions. allowed_k is the next movable point; $\tau_{ij}(t)$ is the pheromone concentration heuristic function, and $\eta_{ij}(t)$ is the distance heuristic function, defined as follows:

$$\eta_{ij}(t)=\frac{1}{d_{ij}}, \quad (2)$$

$$d_{ij}=\sqrt{(x_i-x_j)^2+(y_i-y_j)^2}. \quad (3)$$

When all ants complete their path searches, the pheromones they leave behind will naturally evaporate, gradually diminishing the pheromone levels. ρ in (4) – evaporation rate, falls within the range ($0<\rho<1$), decreasing the pheromone concentration left by the ants. The pheromone update equation is expressed as follows:

$$\tau_{ij}(t+1)=(1-\rho)\tau_{ij}(t)+\Delta\tau_{ij}(t), \quad (4)$$

and:

$$\Delta\tau_{ij}(t) = \sum_{k=1}^{m} \Delta\tau_{ij}^{k}(t), \quad (5)$$

where $\Delta\tau_{ij}(t)$ is the increase in pheromone on the path (i, j) in the current cycle; $\Delta\tau_{ij}^{k}(t)$. is the increase in pheromone in the current cycle, ant k traverses the path (i, j). Assuming all ants search complete paths and subsequently update pheromones, the ant cycle model can be updated with the following equation:

$$\Delta\tau_{ij}^{k}(t) = \\ = \begin{cases} \dfrac{Q}{L_k}, \text{ if } ant\, k \text{ travels through path } (i,j), \\ 0, \text{ if } ant\, k \text{ does not travel through path } (i,j), \end{cases} \quad (6)$$

where L_k – total length of all paths taken by ant k in the current cycle [10] and Q – pheromone intensity. This pheromone intensity slightly influences the search results.

The ACO algorithm involves several parameters that influence its performance, including the number of ants, initial pheromone levels, evaporation rate, and number of iterations [18, 19]. Careful parameter tuning is essential to achieving optimal performance. ACO has been successfully applied to various problems, such as image edge detection and workforce planning, demonstrating its effectiveness in finding high-quality solutions [20].

4. 3. Artificial potential fields (APF)

The APF algorithm operates on gradients, modeling vehicles as points that navigate through a potential field. These points are attracted toward the goal while being repelled by obstacles. The primary advantage of this approach is its ability to rapidly generate collision-free paths. However, a significant limitation is its susceptibility to becoming trapped in local minima, where better solutions may exist but remain undiscovered by the algorithm. This issue arises because the APF algorithm relies on gradients with zero slope, causing it to terminate upon reaching such states and preventing further exploration of new search spaces [21].

The APF algorithm comprises three key components: attractive force, repulsive potential force, and total potential force [22, 23]. In path planning using the APF method, the primary objective is to generate a path that allows agents (such as robots) to reach their final destination while avoiding obstacles. The fundamental equations used in APF-based path planning include:

$$F_{atr} = K_{atr} \times (goal - position), \quad (7)$$

where F_{atr} – attractive force, K_{atr} – scaling factor that determines the strength of the attractive force, *goal* represents the destination position, and *position* denotes the agent's current position.

Meanwhile, the repulsive force, which pushes the agent away from obstacles, is calculated using the following equation:

$$F_{rep} = \sum_i K_{rep} \times \frac{1}{d_i^2} \times \widehat{d}_i, \quad (8)$$

where K_{rep} – scaling factor that determines the strength of the repulsive force, d_i – unit vector pointing from the agent to obstacle i, d_i^2 – squared distance between the agent and the obstacle. The summation symbol Σ accounts for all obstacles surrounding the agent.

Another fundamental principle in the APF method for path planning is the total force. This total force is the sum of the attractive and repulsive forces and dictates the agent's movement. It is expressed as:

$$F_{total} = F_{atr} + F_{rep}. \quad (9)$$

Finally, the agent's movement is determined by the total force acting upon it.

4. 4. A-Star

The A-Star (A*) algorithm was initially developed by Peter Hart, Nils Nilsson, and Bertram Raphael in 1968. Widely utilized in route searching algorithms, A* computes the total cost from one node to another, making it particularly effective for finding the shortest route to a destination while also estimating the remaining distance. This algorithm optimizes search efficiency by incorporating heuristics to determine the fastest path [24, 25].

The A* algorithm determines the shortest distance to a destination by evaluating the generated nodes [26] In its calculations, the algorithm considers the total number of nodes traversed to reach the destination from a given location. The cost function is computed using the following equation:

$$F(n) = g(n) + h(n), \quad (10)$$

where $F(n)$ – total costs associated with reaching node n; $g(n)$ – cost incurred from the initial node to node n, and $h(n)$ – heuristic cost from node n to the destination node.

4. 5. Experimental setup

The experiment utilizes a dataset containing latitude and longitude coordinates of roads within the Universitas Sriwijaya campus in Indralaya, with routes depicted in Fig. 2. This campus route was chosen as it reflects conditions similar to those of suburban areas in Indonesia where autonomous vehicles operate.

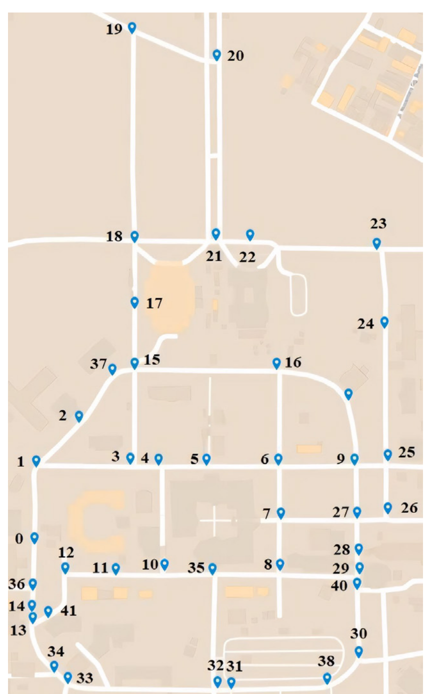


Fig. 2. Universitas Sriwijaya Indralaya campus route map

The comparison of the three algorithms is based on parameters such as route distance and the speed of route formation. Testing begins by inputting the initial and final points. The parameters of the path planning algorithms used are detailed in Table 1.

Table 1

Testing parameters

ACO	A*	APF
The amount of pheromones = 50 ants	Searching using Euclidean heuristic with the formula	Itera-tion = 1000
The amount of itera-tion = 30 Iteration	$d=\sqrt{(x_1-x_2)^2+(y_1-y_2)^2}$	The maximum distance for obstacle verifi-cation = 10
The maximum num-ber of pheromone steps = 50		Repulsive Strength = 1
Evaporation rate = 0.1		Attractive Strength = 1
Alpha = 0.7		
Beta = 0.3		

Table 1 represents all the parameters used in this study during the experimental setup for ACO, A, and APF.

5. Results of comparative testing

5. 1. Evaluation of the performance of ACO, APF, and A* algorithms

In the initial testing phase, the autonomous vehicle operates along the route shown in Fig. 2 without encountering any obstacles that could divert it from the starting point to the destination. Table 2 presents data on the distance costs of 10 out of 182 sample routes generated by the ACO, APF, and A* path planning algorithms. These results were obtained using the parameters shown in Table 1.

Table 2

Testing result

No.	Initial point	Destination	Distance cost ACO (km)	Distance cost APF (km)	Distance cost A* (km)
1	Auditorium (17)	Unsri Monument (21)	0.08	0.08	0.08
2	Unsri Monument (21)	Faculty of Social Science and Politics (25)	0.4	0.39	0.38
3	Faculty of Social Science and Politics (25)	Faculty of Economics (28)	0.34	0.34	0.33
4	Faculty of Economics (28)	Faculty of agriculture (33)	0.44	0.45	0.44
5	Faculty of Engineering (0)	Unsri Monument	0.56	0.55	0.55
6	Faculty of Mathematics and Natural Science (14)	Faculty of Economics (28)	0.62	0.57	0.57
7	Faculty of agriculture (33)	Faculty of Social Science and Politics (25)	1.32	0.78	0.78
8	Library UNSRI (5)	Faculty of Mathematics and Natural Science (14)	0.51	0.51	0.5
9	Auditorium (17)	Faculty of Teacher Training and Education (31)	1.02	Failed	0.81
10	Faculty of Medicine (15)	Faculty of Teacher Training and Education (31)	0.7	0.69	0.68

Table 2 reveals that the three algorithms perform equally well for short-distance routes, such as from the Auditorium to the UNSRI Monument. However, for longer routes with multiple intersections, such as from the Auditorium (point 8) to the Faculty of Teacher Training and Education (point 31), as shown in Fig. 3, the APF algorithm fails to generate a route.

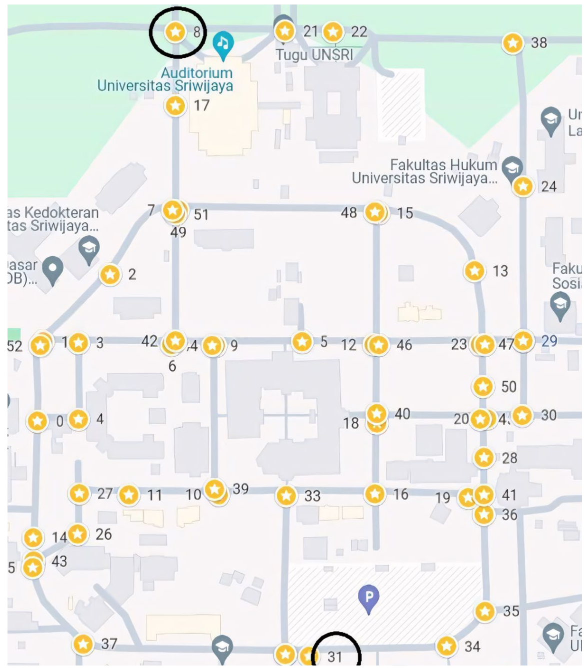


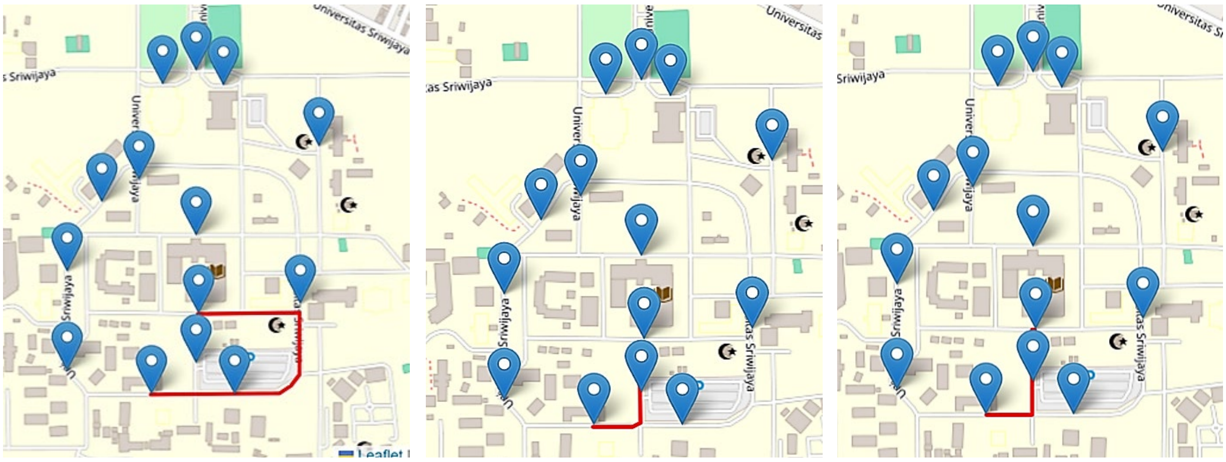
Fig. 3. Maps Auditorium to Faculty of teacher training and education

In these tests, the A* algorithm demonstrates superior performance compared to ACO and APF, covering a more optimal distance of 0.81 km compared to 1.02 km for ACO, while APF fails to generate a path. This advantage stems from A*'s heuristic approach, which utilizes the Euclidean distance. APF's failure to generate a path results from its need to constrain distances between objects and obstacles, a crucial factor in avoiding local minima during path formation. Neglecting this constraint renders APF incapable of generating a viable path.

Overall, based on the analysis of 182 samples, the average distance cost of A* is 0.48 km, which is superior to that of the other algorithms.

Fig. 4 shows one of the paths tested, specifically the path from Fisheries Department to Faculty of Agriculture.

Fig. 4 demonstrates that the longest path obtained using the ACO algorithm (0.62 Km). This discrepancy arises because ACO relies on probabilistic methods. In scenarios with few intersections and short distances, the APF method can identify the optimal route. However, for routes with many intersections and longer distances, the APF method tends to produce longer routes and frequently encounters local minima, leading to failures in route generation. Consequently, the APF method is more suitable for local path planning.



a *b* *c*

Fig. 4. Path planning from fisheries department (35) to agriculture faculty (33) with: *a* – ACO; *b* – APF; *c* – A*

5. 2. Utilization of paths obtained from unstructured suburban areas in Indonesia

In this testing phase, the autonomous vehicle operates along the route shown in Fig. 2, encountering obstacles that could divert it from the starting point to the destination. These typical obstacles may occur in unstructured suburban areas in Indonesia due to certain conditions, such as cone blocks or other barriers that may make it difficult for the autonomous vehicle to navigate. Subsequent testing incorporates obstacles using the map from the Auditorium (17) to the Faculty of Agriculture, with results presented in Table 3.

As depicted in the table, both ACO and A* successfully find alternative routes when encountering an obstacle. Meanwhile, APF fails to re-route in the presence of obstacles. These results highlight APF's limitations in local route formation when autonomous vehicles encounter obstacles. However, A* also demonstrates inefficiency in navigating routes obstructed by obstacles.

In addition to evaluating their ability to find the shortest path, let's also assess the performance of the three algorithms – ACO, APF, and A – in terms of computational efficiency and execution time. In terms of execution time, the time cost was calculated for each algorithm in finding the best routes. The overall execution time for processing all latitude and longitude data is illustrated in Fig. 5.

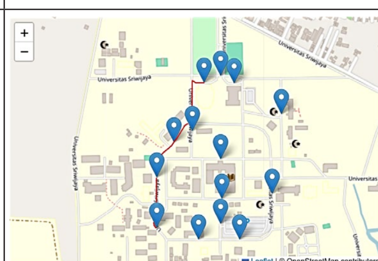
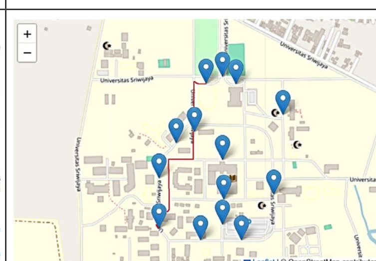
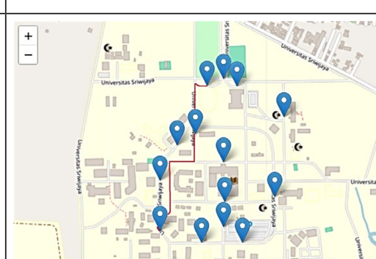
Next, the path length is compared to determine which of the three algorithms produces the shortest route. The graph in Fig. 6 presents a comparison of the total path lengths for the ACO, A*, and APF algorithms.

For APF and A*, an increase in distance leads to a longer route formation time. However, in ACO, route distance does not directly determine the time required for route formation, as the algorithm incorporates a probability factor in the process. This analysis is further supported by the graph illustrating the relationship between distance and route formation speed in Fig. 7.

Fig. 7 shows that the relationship between path length and time for ACO is more scattered compared to APF and A*. Interestingly, A* shows a better pattern, but it requires more time to find the best route.

Table 3

Testing with obstacle

No.	Algorithm	Not blocked	Blocked
1	ACO	 0.67 km	 0.73 km
2	APF	 0.67 km	Failed
3	A*	 0.67 km	 0.73 km

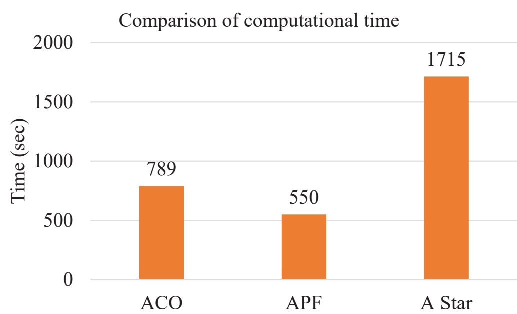


Fig. 5. Comparison computational time

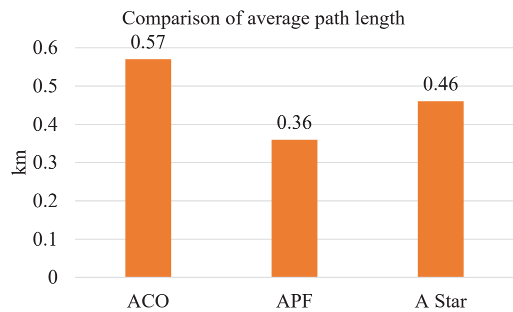


Fig. 6. Comparison of average path length

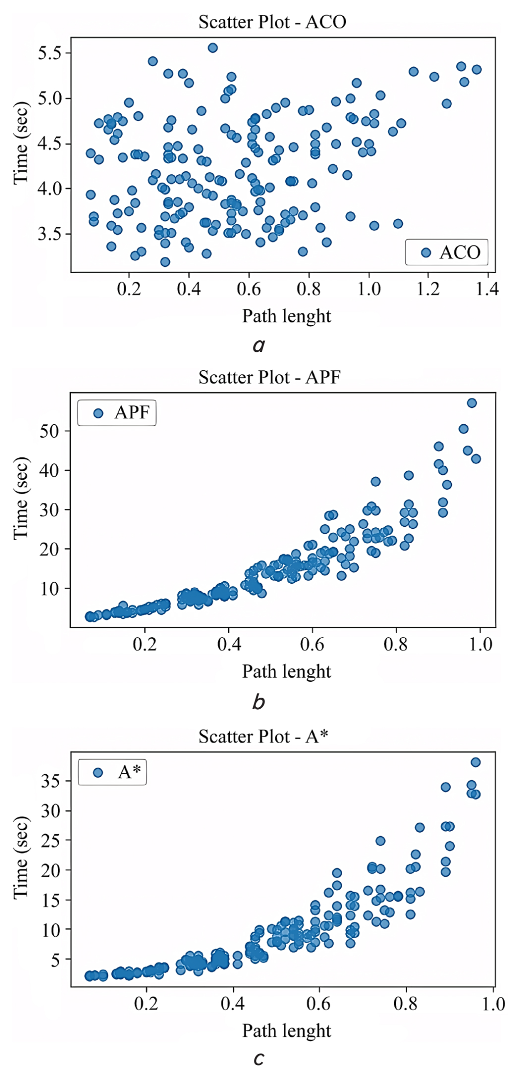


Fig. 7. Distance and speed relation graph: *a* – ACO; *b* – APF; *c* – A*

6. Discussion of results: performance comparison of path planning algorithms in autonomous vehicle navigation

Table 2 shows that ACO, APF, and A* achieved similar performance in cases without obstacles and on short-distance paths. However, as the distance increases, APF may become trapped in a local minimum, leading to failure in finding the shortest path, as shown in Fig. 3. Meanwhile, ACO remains relatively stable for both short and long paths. Nevertheless, its performance is inferior to A*, which relies on a heuristic approach.

Fig. 5 shown where it is evident that APF exhibits superior performance in computational time compared to A* and ACO, but this advantage can be attributed to the initial test's implementation of a maximum obstacle distance limitation per iteration. While this restriction aims to prevent excessively long iteration times, it occasionally results in suboptimal routes and, in some cases, even failure to generate a route. These findings reinforce our earlier observation that the APF algorithm is well-suited for local path planning.

The study results in Fig. 6 show that the A* algorithm performs the best in finding the optimal route compared to ACO and APF. As shown in Table 2, A* produces an average path length of 0.48 km with a 100 % success rate, while ACO has an average of 0.57 km with 100 % success, and APF only 0.36 km but with a 41 % success rate. This superiority of A* can be explained by its Euclidean-based heuristic approach, which allows for more accurate route estimation compared to the probabilistic method of ACO and the gradient-based method of APF.

From Fig. 7, it can be seen that the route plot in ACO is more spread out than in APF, while A* is more compact. As a result, the routes formed by ACO are sometimes longer compared to those generated by APF and A*.

Additionally, as illustrated in Fig. 4, ACO generates longer and more scattered routes like shows in Fig. 7 due to its probabilistic nature. Meanwhile, APF often gets trapped in local minima, causing failures in generating feasible routes in complex environments. This is evident in Table 3, where APF fails when encountering obstacles along the path.

This study compares the performance of three algorithms in real-world environments, unlike many previous studies that only tested these algorithms in simulations or controlled environments. For example: Unlike study [8], which tested ACO in a static environment, this study demonstrates that ACO generates longer routes in more complex environments due to its probabilistic nature. Unlike study [10], which improved APF for unmanned vehicle path planning, this study shows that APF still has limitations in real-world complexity to getting stuck in local minima. Unlike study [12], which only tested A* in simulations, this study confirms that A* remains superior in real-world conditions with various environmental parameters.

The key advantage of this study is its direct application in suburban areas of Indonesia, where unstructured road infrastructure poses unique challenges for autonomous vehicle navigation.

This study successfully addresses several key challenges identified in the Literature Review, namely:

– the lack of direct evaluation of ACO, APF, and A* algorithms in real-world road conditions. This study provides a direct comparison using actual coordinate data from Universitas Sriwijaya, ensuring that the results are more representative of real-world navigation challenges;

– the limitations of previous studies that focused only on ideal simulation environments. By conducting tests in unstructured environments, this study highlights how factors such as probability (ACO), local minima traps (APF), and heuristic efficiency (A*) affect algorithm performance in real-world conditions.

Although this study provides valuable insights, there are several limitations to consider:

– limited test environment: experiments were conducted only on Universitas Sriwijaya's campus, which, although representative of suburban road conditions in Indonesia, may not fully reflect all road conditions in developing countries;

– lack of dynamic factors: this study does not incorporate variables such as traffic flow or pedestrian movement, which could impact algorithm effectiveness in more complex real-world scenarios.

In practical applications, A* is recommended for global navigation, while APF is more suitable for local navigation, as confirmed in this study. ACO, although not optimal in terms of travel distance, can still be used in scenarios where a probabilistic-based approach is preferred. Thus, in future research, it is necessary to further develop ACO to obtain the optimal distance, as ACO demonstrates robustness in finding both short and long distances, with or without obstacles.

7. Conclusions

1. In performance comparison of path planning algorithms:

– this study comprehensively evaluated the performance of three path planning algorithms – ACO, APF, and A* – sing real-world data from an unstructured suburban area in Indonesia. The findings provide insights into the effectiveness, computational efficiency, and execution time of each algorithm;

– the APF algorithm tends to generate longer routes due to its iterative limitation on maximum obstacle distance, which is designed to avoid local minima. However, this restriction also contributes to APF being the fastest in route formation compared to A* and ACO. Route plots in ACO are more dispersed than in APF, while A* produces more direct paths, often resulting in shorter routes than APF and ACO;

– in APF and A*, longer distances lead to slower execution times, whereas in ACO, route distance has minimal impact on execution time due to its probabilistic nature. These differences stem from the inherent characteristics of each algorithm's execution process.

2. ACO and A* have shown a 100 % success rate, with an average distance of 0.57 km and 0.48 km, respectively, in the case of a path with obstacles. However, ACO can be improved to obtain an optimal path since it can adapt to dynamic environments. This study demonstrates that the ACO algorithm can be adopted for autonomous vehicles, though considerations for optimizing its speed remain a focus for future research.

Conflict of interest

The authors declare that they have no conflict of interest in relation to this research, whether financial, personal, authorship or otherwise, that could affect the research and its results presented in this paper.

Financing

The research/publication of this article was funded by DIPA of Public Service Agency of Universitas Sriwijaya 2024. No SP DIPA 023.17.2.677515/2024, On November 24, 2023. In accordance with the Rector's Decree Number: 0013/UN9/LP2M.PT/2024, On May 20, 2024.

Data availability

Data will be made available on reasonable request.

Use of artificial intelligence

The authors confirm that they did not use artificial intelligence technologies when creating the current work.

Acknowledgments

The authors would like to thank Universitas Sriwijaya for its funding and support during this research.

References

1. Alaliyat, S., Oucheikh, R., Hameed, I. (2019). Path Planning in Dynamic Environment Using Particle Swarm Optimization Algorithm. 2019 8th International Conference on Modeling Simulation and Applied Optimization (ICMSAO), 1–5. https://doi.org/10.1109/icmsao.2019.8880434
2. Sari, D. W., Dwijayanti, S., Suprapto, B. Y. (2023). Path Planning for an Autonomous Vehicle based on the Ant Colony Algorithm: A Review. 2023 International Workshop on Artificial Intelligence and Image Processing (IWAIIP), 57–62. https://doi.org/10.1109/iwaiip58158.2023.10462744
3. Jin, Q., Tang, C., Cai, W. (2022). Research on Dynamic Path Planning Based on the Fusion Algorithm of Improved Ant Colony Optimization and Rolling Window Method. IEEE Access, 10, 28322–28332. https://doi.org/10.1109/access.2021.3064831
4. Chen, Y., Luo, W., Wang, M., Su, Y., Zhang, H. (2022). UUV 3D Path Planning Based on PSO-ACO Fusion Algorithm. 2022 37th Youth Academic Annual Conference of Chinese Association of Automation (YAC), 599–604. https://doi.org/10.1109/yac57282.2022.10023579
5. Qin, H., Shao, S., Wang, T., Yu, X., Jiang, Y., Cao, Z. (2023). Review of Autonomous Path Planning Algorithms for Mobile Robots. Drones, 7 (3), 211. https://doi.org/10.3390/drones7030211
6. Zhang, H., Lin, W., Chen, A. (2018). Path Planning for the Mobile Robot: A Review. Symmetry, 10 (10), 450. https://doi.org/10.3390/sym10100450

7. Tao, Y., Gao, H., Ren, F., Chen, C., Wang, T., Xiong, H., Jiang, S. (2021). A Mobile Service Robot Global Path Planning Method Based on Ant Colony Optimization and Fuzzy Control. Applied Sciences, 11 (8), 3605. https://doi.org/10.3390/app11083605
8. Yu, X., Chen, W.-N., Gu, T., Yuan, H., Zhang, H., Zhang, J. (2019). ACO-A*: Ant Colony Optimization Plus A* for 3-D Traveling in Environments With Dense Obstacles. IEEE Transactions on Evolutionary Computation, 23 (4), 617–631. https://doi.org/10.1109/tevc.2018.2878221
9. Tang, B., Zhu, Z., Luo, J. (2016). Hybridizing Particle Swarm Optimization and Differential Evolution for the Mobile Robot Global Path Planning. International Journal of Advanced Robotic Systems, 13 (3). https://doi.org/10.5772/63812
10. Shi, K., Huang, L., Jiang, D., Sun, Y., Tong, X., Xie, Y., Fang, Z. (2022). Path Planning Optimization of Intelligent Vehicle Based on Improved Genetic and Ant Colony Hybrid Algorithm. Frontiers in Bioengineering and Biotechnology, 10. https://doi.org/10.3389/fbioe.2022.905983
11. Zhang, J., Wu, J., Shen, X., Li, Y. (2021). Autonomous land vehicle path planning algorithm based on improved heuristic function of A-Star. International Journal of Advanced Robotic Systems, 18 (5). https://doi.org/10.1177/17298814211042730
12. Orozco-Rosas, U., Montiel, O., Sepúlveda, R. (2019). Mobile robot path planning using membrane evolutionary artificial potential field. Applied Soft Computing, 77, 236–251. https://doi.org/10.1016/j.asoc.2019.01.036
13. Park, C., Kee, S.-C. (2021). Online Local Path Planning on the Campus Environment for Autonomous Driving Considering Road Constraints and Multiple Obstacles. Applied Sciences, 11 (9), 3909. https://doi.org/10.3390/app11093909
14. Szczepanski, R., Bereit, A., Tarczewski, T. (2021). Efficient Local Path Planning Algorithm Using Artificial Potential Field Supported by Augmented Reality. Energies, 14 (20), 6642. https://doi.org/10.3390/en14206642
15. Fu, J., Lv, T., Li, B. (2022). Underwater Submarine Path Planning Based on Artificial Potential Field Ant Colony Algorithm and Velocity Obstacle Method. Sensors, 22 (10), 3652. https://doi.org/10.3390/s22103652
16. I Hamad, I. I., Hasan, M. S. (2020). A Review: On using ACO Based Hybrid Algorithms for Path Planning of Multi-Mobile Robotics. International Journal of Interactive Mobile Technologies (IJIM), 14 (18), 145. https://doi.org/10.3991/ijim.v14i18.16371
17. Che, G., Liu, L., Yu, Z. (2019). An improved ant colony optimization algorithm based on particle swarm optimization algorithm for path planning of autonomous underwater vehicle. Journal of Ambient Intelligence and Humanized Computing, 11 (8), 3349–3354. https://doi.org/10.1007/s12652-019-01531-8
18. Ou, J., Wang, M. (2019). Path Planning for Omnidirectional Wheeled Mobile Robot by Improved Ant Colony Optimization. 2019 Chinese Control Conference (CCC), 2668–2673. https://doi.org/10.23919/chicc.2019.8866228
19. Li, X., Li, Q., Zhang, J. (2022). Research on global path planning of unmanned vehicles based on improved ant colony algorithm in the complex road environment. Measurement and Control, 55 (9-10), 945–959. https://doi.org/10.1177/00202940221118132
20. Fidanova, S. (2021). Ant Colony Optimization and Applications. In Studies in Computational Intelligence. Springer International Publishing. https://doi.org/10.1007/978-3-030-67380-2
21. Zhang, Z., Chen, S., Li, Y., Wang, L., Ren, R., Xu, L. et al. (2022). Local Path Planning of Unmanned Underwater Vehicle Based on Improved APF and Rolling Window Method. 2022 International Conference on Cyber-Physical Social Intelligence (ICCSI), 542–549. https://doi.org/10.1109/iccsi55536.2022.9970658
22. Chen, G., Liu, J. (2019). Mobile Robot Path Planning Using Ant Colony Algorithm and Improved Potential Field Method. Computational Intelligence and Neuroscience, 2019, 1–10. https://doi.org/10.1155/2019/1932812
23. Yang, R., Cheng, L. (2019). Path Planning of Restaurant Service Robot Based on A-star Algorithms with Updated Weights. 2019 12th International Symposium on Computational Intelligence and Design (ISCID), 292–295. https://doi.org/10.1109/iscid.2019.00074
24. Ju, C., Luo, Q., Yan, X. (2020). Path Planning Using an Improved A-star Algorithm. 2020 11th International Conference on Prognostics and System Health Management (PHM-2020 Jinan), 23–26. https://doi.org/10.1109/phm-jinan48558.2020.00012
25. Suprapto, B. Y., Jonathan, J., Dwijayanti, S. (2023). Implementation of the A* Algorithm for Determining the Best Route for an Autonomous Electric Vehicle. 2023 International Workshop on Artificial Intelligence and Image Processing (IWAIIP), 304–308. https://doi.org/10.1109/iwaiip58158.2023.10462790
26. Sun, Y., Fang, M., Su, Y. (2021). AGV Path Planning based on Improved Dijkstra Algorithm. Journal of Physics: Conference Series, 1746 (1), 012052. https://doi.org/10.1088/1742-6596/1746/1/012052