A-star Algorithm & Hybrid A-star Algorithm

Haoxian Cai*

School of Mechanical and Electronic Control Engineering, Beijing Jiaotong University, Beijing, China

*Corresponding author: 23222031@bjtu.edu.cn

Abstract. A-star Algorithm and Hybrid A-star Algorithm are universally utilized in automatic path planning. However, based on previous research, in various scenarios, the two algorithms have potential issues in the generation of an optimized path. This essay proposes a review of previous work and conclusions to objectively assess the performance of each algorithm in various aspects, ranging from computational time, success rate, efficiency, trajectory smoothness, to versatility among diverse circumstances. These aspects are crucial in practical applications of autonomous navigation, where the balance between speed, reliability, and adaptability directly impacts system performance. After the analysis on benefits and drawbacks of each algorithm, a comprehensive conclusion is drawn that A-Star has features of wide adaptability, high efficiency and low computational speed in complex conditions, while Hybrid A-star Algorithm provides precise trajectory planning based on car limitations with smooth driving but attached with lower success rate in some conditions and additional expense on hardware. This makes Hybrid A-star more suitable for structured environments where kinematic feasibility and smooth manoeuvring are prioritized. Both algorithms face distinct issues and are recommended to be utilized in distinct fields based on the conclusion drawn. The final recommendations emphasize that algorithm selection should be scenario-dependent, considering trade-offs between performance and resource requirements.

Keywords: A-star Algorithm; Hybrid A-star Algorithm; Trajectory smoothness; Performance.

1. Introduction

Path planning has been an emerging applied technology in various industries, ranging from Automated Guided Vehicle (AGV) to robotic-arm motion, since it shortens the trajectory length and avoids collision with obstacles [1,2]. With the development of automation, instant planning with interactions to changing environments or constant obstacles is necessary. Therefore, to realize autonomous path planning, diverse algorithms, including the A-star algorithm, are employed [3,4]. Rapidly Rising Tree (RRT) algorithm, genetic algorithm are widely accepted in path planning [3,4]. Among all the above, A-star is preferred in diverse fields since it detects an optimized path by detecting adjacent nodes, while these nodes are evaluated by the sum of the heuristic function(h(n)) and geometric function(g(n)) [5]. Based on the fundamental theory of the A-star algorithm, the Hybrid A-star algorithm is proposed for path solution for vehicles that contain constraints of chassis and turning radius [6]. In different circumstances, however, the two algorithms above have distinct flaws in actual implementation. Despite its simplicity, a relative study points out the A-star algorithm's rising execution time on a large scale when the map size gradually extends and the generated path is not the optimized solution [7,8]. Although the Hybrid A-star algorithm expands node searching by curve paths, research shows its error on trajectory planning when Euclidean distance immensely differs from actual distance, which affects the computational time and success rate in real planning [6].

Both algorithms meet potential issues when several factors are considered. Therefore, for assessing both algorithms, the concrete performance of the A-star and the Hybrid A-star algorithm requires a qualitative evaluation. In this study, an objective review of two algorithms is proposed by analysing from advantages to disadvantages. In summary, the contributions of this paper are to conduct a literature review on the A-star Algorithm and the Hybrid A-star Algorithm, and draw a conclusion while commenting on future work.

The rest of the paper is organized as follows. First, the definitions of the two algorithms are introduced. Then, each algorithm is discussed based on previous studies. Finally, a comprehensive conclusion is provided.

2. Definition

2.1. A-Star Algorithm

The basic equation of the A-star Algorithm consists of two functions: the heuristic function (h(n)) and the geometric function (g(n)). The sum of the two functions is named the "cost function"; it is defined as follows [8]:

$$f(n) = h(n) + g(n) \tag{1}$$

Geometric function(g(n)) is an objective function that reflects the linear distance between the starting point and the location of the specified spot.

Heuristic function (f(n)) is a customized function that is designed according to requirements. In regular situations, the Manhattan distance or Euclidean distance between the specified spot and goal is applied for the function [9,10]. Manhattan distance refers to the sum of the length difference in the x direction and the y direction. For spot n to be calculated, the function is demonstrated as:

$$h_{Manhattan}(n) = |\ x_{goal} - x_n\ | + |\ y_{goal}\ - y_n\ | \tag{2}$$

Euclidean distance refers to the linear distance between operated spot and goal, shown as [8]:

$$h_{Euclidean}(n) = \sqrt{(x_{goal} - x_n)^2 + (y_{goal} - y_n)^2} \tag{3}$$

2.2. Hybrid A-star Algorithm

The Hybrid A-star Algorithm has the same fundamental mechanism as the A-star Algorithm, which both search the optimized nodes by comparison of the cost function. Constraint based on the car structure is considered in functions for a better prediction of vehicle motion. Autonomous vehicles with Hybrid A will search for possible locations by curved paths. With limitations of turning radius, the widest curve path leads to the furthest node away from the car body.

3. Algorithms

This section mainly focuses on two algorithms. By inheriting and analysing previous work, an analysis of each algorithm is done in every unit.

3.1. Features of A-star

From relative studies, it is drawn that the A-star algorithm has a strong versatility under different circumstances. In various fundamental experiments, A-star has performed well under various specific tasks, which prove its effectiveness. Initially, A-star can be utilized for a more stable output. Foaled et al. have claimed that A-star can reach a higher success rate than other algorithms, such as HPA* [11]. Due to the principle of searching 8 adjacent nodes, A-star is enabled to move in 8 directions forward and backward in freedom, which significantly enhances its ability of self-correction [12]. Furthermore, the structure of the A-star Algorithm is determined to be efficient [13]. The study conducted by Zhou et al. points out that A-star automatically avoids unreachable nodes and quickly calculates the shortest path [14]. This is explained by Wayahdi et al. that no overestimation exists in the evaluation of cost value when A-star operates, where work is reduced by statistical filtering [14-18]. Eventually, the A-star algorithm can flexibly switch its heuristic function belonging to Section II according to actual requirements [7,14-16]. Only when h(n) is larger than the value of g(n) may it lead to the inefficiency of A-star [14]. Relevant conclusions drawn in studies and research have shown that the A-star Algorithm can deal with various scenarios while keeping a high success rate. The feasibility of changing the heuristic function also enhances its ability to adapt unknown environment. To wind up, A-star Algorithm possesses a strong adaptability with a stable output and a lower memory occupation in chips, which is friendly to algorithm manipulation on vehicles and robots. This conclusion is proved by an experiment conducted by Duchoň et al., where datasets of compared algorithms in a specific configuration are demonstrated in Table 1 [7]:

Table 1. Comparison of Computational time adopted from work by Duchoň et al. [7]

Algorithms	Computational time [ms]
A*	229
Basic Theta*	998
Phi*	1019
JPS(A*)	21

From Table 2, it is shown that A-star Algorithms and their derived algorithm with modified heuristic functions significantly outperform the rest of the algorithms in computational time. The advantage in computational time allows hardware with poor conditions to run efficiently, as it is mentioned above.

Nevertheless, the A-star Algorithm has exposed its critical disadvantage, reversely to the conclusion above, for the poor efficiency when situations occur in complicated scenarios. A-star Algorithm performs weakly in solution of objectives with higher complexity or with diverse targets. The algorithm has poor efficiency among many samples compared. The listed data in Table 2 offers a clear vision on the comparison of efficiency between A-star with its variations and other advanced algorithms [11]:

Table 2: Advantages beyond A-star from work by Foaed et al. [11]

Algorithm	Result
GAMMA (Genetic Algorithm)	85% efficiency increase
FAR (Flow Annotation Replanning)	86% path length decrease
Jump points	Over 300% node expansion speed increase
OA (Optimal Anytime)	67% efficiency increase
TASS	300% node expansion speed increase

Several factors lead to the huge gap in execution time in comparison. The extension of the map size can lead to a huge difference in execution time compared to other improved algorithms. When multiple conditions relative to map features or specific tasks are changed, the computational time of the algorithm can be hugely affected on a tremendous scale. The study conducted by Duchoň has demonstrated that the cost of average computational time for most computer chips is over an hour when the map area reaches 60,000 cells, which seriously hinders the popularization of A-star among robots with poor condition in hardware conditions [7]. Meanwhile, as it lacks specified modifications on the heuristic function for pursuing the least computational time in multiple scenarios, among various layouts with different targets, the long computational time of the A-star algorithm has been demonstrated by comparison in other varieties of tasks, rather maze-solving problem. In bidirectional graph searching, for instance, Breadth First Searching (BFS) significantly outperforms A-star in computational time than that of A-star as the map size is larger than a specific value [17-19]. The utilization of the A-star Algorithm produces redundant trajectories, which additionally costs execution time during robot manipulation [1,20]. In the experiment of AGV route planning by Tang et al. “sawtooth path” and “cross path” for unnecessary U-turns are generated in the visualized experiment results, which enormously retards progress on standby computation in real-time planning, as generated sharp trajectory leads to lower turning speed [1,14].

The factors shown above have provided a comprehensive answer to why A-star takes longer in general situations. Relative statistic tells that the A-star Algorithm requires extra modification according to the needs of operators, as the algorithm performs under expectation in a huge gap either in execution time or optimized path. Therefore, the A-star Algorithm shows its lack of capability in quick solution for complicated tasks. For developers who intend to apply fast-computing route planning in complicated scenarios, it is of great necessity to innovate a complex heuristic function to mitigate time cost.

3.2. Features of Hybrid A-star

The Hybrid A-star Algorithm has its advancement in unmanned driving as it considers limitations of the turning radius compared to the A-star Algorithm. The study by Li et al. claims that the selection of a successor can be adjusted under the constraints of the vehicle and generates extra nodes by subdivision of the turning radius [21]. Penalty on acceleration based on Voronoi potential field is also implemented for smoother motion [22,23]. Thus, the algorithm produces a smoother route and reduces damage to the chassis caused by jerky turning, braking, etc. The hybrid A-star has a great advantage in public transportation and commercial services since it effectively mitigates sudden motion and satisfies the demand from people. This is demonstrated by Sheng et al., as Figure 1 depicts the smoothness difference of Hybrid A and A-star in the route [24]:

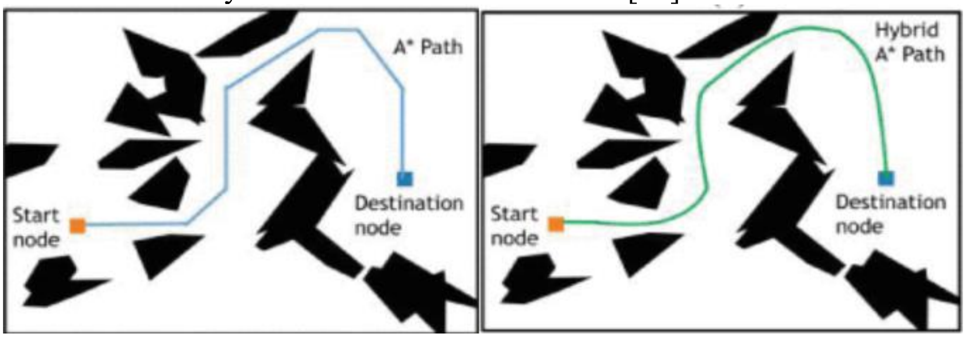

Fig.1: Smooth difference between A-star and Hybrid A-star in complex configuration by Sheng et al. [24]

As shown in the previous study, multiple obtuse angles occur in the trajectory from A-star, while those of Hybrid A are replaced by smooth curve paths. For four-wheel vehicles, manipulation by the Hybrid A-star Algorithm can effectively protect the stability of car motion as the path from A-star generates higher centripetal force when driving, with potential for damage.

The shortcomings of the Hybrid A-star Algorithm are also revealed in several aspects. One of the most critical issues is the possibility of output failure during execution. Research conducted by Sedighi et al. has shown that when Euclidean distance from start to goal massively differs from actual distance, Hybrid A-star is prone to fail [6]. Narrow aisles can also lead to extension in finding invalid spots before a valid successor is detected, while under some circumstances, the algorithm might be trapped without generation of a complete start-to-goal trajectory [24]. Subsequent problems such as having difficulty on modification of heuristic function etc. can similarly influence on the rate of success for an expected output, as heuristic function in Hybrid A-star is usually much more complicated than that of A-star, which is often expressed in quadratic polynomials or even polynomials in higher order for advanced performance [6, 25]. Kinematics and dynamic algorithms of non-linear equations for the pursuit of the optimized path are harder to compute than normal linear functions, occupying extra chip memory at the same time [23,26,27]. Findings of previous work have revealed that Hybrid A-star requires high-performance equipment to support a massive amount of computation during the process. Extra sensors are necessary to enhance verification of surroundings and import sufficient information to microchips for avoiding the execution failure of the algorithm. Therefore, it is drawn that Hybrid A-star is unfriendly to poorly equipped vehicles and robots without a PID controller and occupies a lot of resources for computation, while the trajectory planning is possible to fail.

4. Conclusion

From the analysis of two algorithms, the benefits and drawbacks of each are sufficiently reviewed. It is drawn that the A-star Algorithm is more versatile in various scenarios with low memory occupation for its simple structure, while the computational time for complex cases increases on a tremendous scale. The hybrid A-star Algorithm generates a smoother trajectory since it takes the limitations of turning curvature and car acceleration into account, bringing passengers a comfortable journey with less jerky motion in the vehicle. However, Hybrid A-star itself consumes a massive memory and thus requires hardware with better performance. Besides, its versatility is not as good as that of the A-star Algorithm.

Both algorithms expose their issues when facing specific problems, urging for better modifications on their basic structure to approach a reliable, efficient, and fast-computing algorithm for future applications, especially within the rapid popularization of “Internet of Things” around the globe. Evolution is necessary for better efficiency in various situations.

Future applications are seen according to the results with clear limitations: For the rising balance car industry among new generations, this experiment provides a valuable conclusion of its function on automatic planning in a limited space and equipment. Automatic guidance of a balance car with relatively poor hardware may adopt A-star in places of finite size, such as a pocket park, yard, parking lot, etc., as it conserves computational resources and plans an optimized trajectory. The two-wheel industry can also greatly benefit from this result in the global market as automation has been gaining popularizing in developing countries. As for Hybrid A-star, it has the potential to serve its function in future applications of intelligent on-road vehicles such as public buses, private cars. The development of hardware backups and the popularization of Hybrid A-star are anticipated to be widely used around the globe. This study brings a new vision from the analysis of previous work.

References

[1] G. Tang, C. Tang, C. Claramunt, X. Hu and P. Zhou. Geometric A-Star Algorithm: An Improved A-Star Algorithm for AGV Path Planning in a Port Environment. In IEEE Access, 2021, vol. 9, pp. 59196-59210.

[2] T. Kunz, U. Reiser, M. Stilman and A. Verl. Real-time path planning for a robot arm in changing environments. 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2010, pp. 5906-5911.

[3] X. Li, X. Hu, Z. Wang and Z. Du. Path Planning Based on Combinaion of Improved A-STAR Algorithm and DWA Algorithm. 2020 2nd International Conference on Artificial Intelligence and Advanced Manufacture (AIAM), 2020, pp. 99-103.

[4] L. Schmid, M. Pantic, R. Khanna, L. Ott, R. Siegwart and J. Nieto. An Efficient Sampling-Based Method for Online Informative Path Planning in Unknown Environments. IEEE Robotics and Automation Letters, 2020, vol. 5, no. 2, pp. 1500-1507.

[5] Munir R and Lidya L 1998 Algoritma dan Pemrogaman (Bandung: Informatika)

[6] Sedighi S, Nguyen D V, Kuhnert K D. Guided hybrid A-star path planning algorithm for valet parking applications[C]. IEEE, 2019 5th international conference on control, automation and robotics (ICCAR). 2019, 570-575.

[7] Duchoň F, Babinec A, Kajan M, et al. Path planning with modified a star algorithm for a mobile robot[J]. Procedia engineering, 2014, vol 9, pp. 9-69.

[8] C. Ju, Q. Luo and X. Yan. Path Planning Using an Improved A-star Algorithm. 2020 11th International Conference on Prognostics and System Health Management (P+HM-2020 Jinan), 2020, pp. 23-26.

[9] L. Greche, M. Jazouli, N. Es-Sbai, A. Majda and A. Zarghili. Comparison between Euclidean and Manhattan distance measure for facial expressions classification. 2017 International Conference on Wireless Technologies, Embedded and Intelligent Systems (WITS), 2017, pp. 1-4.

[10] Liwei Wang, Yan Zhang and Jufu Feng. On the Euclidean distance of images. In IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005, vol. 27, no. 8, pp. 1334-1339.

[11] Daniel Foead, Alifio Ghifari, Marchel Budi Kusuma, Novita Hanafiah, Eric Gunawan. A Systematic Literature Review of A* Pathfinding, Procedia Computer Science, 2021, Volume 179, pp. 507-514, ISSN 1877-0509.

[12] Oluwaseun Opeyemi Martins, Adefemi Adeyemi Adekunle, Olatayo Moses Olaniyan, Bukola Olalekan Bolaji. An Improved multi-objective a-star algorithm for path planning in a large workspace: Design, Implementation, and Evaluation. Scientific African, 2022, vol. 15, e01068, ISSN 2468-227.

[13] J. Yu, J. Hou and G. Chen. Improved Safety-First A-Star Algorithm for Autonomous Vehicles. 2020 5th International Conference on Advanced Robotics and Mechatronics (ICARM), 2020, pp. 706-710.

[14] Y. Zhou, X. Cheng, X. Lou, Z. Fang and J. Ren. Intelligent Travel Planning System based on A-star Algorithm," 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chongqing, China, 2020, pp. 426-430.

[15] Wayahdi M R, Ginting S H N, Syahputra D. Greedy, A-Star, and Dijkstra's algorithms in finding shortest path[J]. International Journal of Advances in Data and Information Systems, 2021, 2(1): 45-52.

[16] A. Candra, M.A. Budiman, & K. Hartanto. Dijkstra's and A-Star in Finding the Shortest Path: a Tutorial. IEEE International Conference on Data Science, Artificial Intelligence, and Business Analytics (DATABIA), 2020, pp. 28-32,

[17] S. Mahadewi, K.R. Shylaja, & M.E. Ravinandan. Memory Based A-Star Algorithm for Path Planning of a Mobile Robot. International Journal of Science and Research (IJSR), 2014, vol. 3, issue 6, pp. 1351-1355.

[18] Z. Lin, K. Wu, R. Shen, X. Yu and S. Huang. An Efficient and Accurate A-Star Algorithm for Autonomous Vehicle Path Planning. In IEEE Transactions on Vehicular Technology, 2024, vol. 73, no. 6, pp. 9003-9008.

[19] Kumar N. Bidirectional Graph Search Techniques for Finding Shortest Path in Image Based Maze Problem (Master's thesis, Punjab Technical University). 2019, pp. 1411-1414.

[20] J. Liu, J. Yang, H. Liu, X. Tian and M. Gao. An improved ant colony algorithm for robot path planning. Soft Comput., 2017, vol. 21, no. 19, pp. 5829-5839..

[21] H. Li, Z. Chen, Z. Huang and Z. Gao. Research on Automatic Parking Path Planning Based on Bidirectional Hybrid A-Star Algorithm. 2024 11th International Forum on Electrical Engineering and Automation (IFEEA), 2024, pp. 1251-1254.

[22] H. Zheng, M. Dai, Z. Zhang, Z. Xia, G. Zhang and F. Jia. The Navigation Based on Hybrid A Star and TEB Algorithm Implemented in Obstacles Avoidance. 2023 29th International Conference on Mechatronics and Machine Vision in Practice (M2VIP), 2023, pp. 1-6.

[23] J. Ziegler, M. Werling and J. Schroder. Navigating car-like robots in unstructured environments using an obstacle sensitive cost function. 2008 IEEE Intelligent Vehicles Symposium, 2008, pp. 787-791.

[24] W. Sheng, B. Li and X. Zhong. Autonomous Parking Trajectory Planning With Tiny Passages: A Combination of Multistage Hybrid A-Star Algorithm and Numerical Optimal Control. In IEEE Access, 2021, vol. 9, pp. 102801-102810.

[25] R. Song, T. Chen, J. Pan and Y. Peng. Adaptive Path Planning for Amphibious Vehicles Based on Enhanced Hybrid A-Star Algorithm. 2024 8th International Conference on Electrical, Mechanical and Computer Engineering (ICEMCE), 2024, pp. 1584-1589.

[26] Qiu D, Li X, Yang H, Zhu X. Time-optimal global trajectory planning for autonomous valet parking: An improved hybrid A-star algorithm-based optimization control approach. International Journal of Advanced Robotic Systems. 2025, 22(2).

[27] C. Li, D. Yu, W. Lu and M. Li. Variable-curvature hybrid A-star search for AMR path planning in limited space. 2021 3rd International Symposium on Robotics & Intelligent Manufacturing Technology (ISRIMT), 2021, pp. 61-65.