Analysis of AGVs Routes Considering Disturbances and Unexpected Events

Fernanda Coutinho [*] , Edson Costa , Jorge Barreiros

Article

Analysis of Automated Guided Vehicles Routes Considering Disturbances and Unexpected Events

Fernanda Coutinho [1,2,*] [ID], **Edson Costa** [1] **and Jorge Barreiros** [1,3] [ID]

[1] Coimbra Institute of Engineering, Polytechnic Institute of Coimbra, 3030-199 Coimbra, Portugal; a2020113892@isec.pt (E.C.); jmsousa@isec.pt (J.B.)

[2] Institute of Systems and Robotics, University of Coimbra, 3030-290 Coimbra, Portugal

[3] NOVALINCS, NOVA University of Lisbon, 2829-516 Caparica, Portugal

* Correspondence: fermaco@isec.pt; Tel.: +351 790200

Abstract: Automated Guided Vehicles are self-guided vehicles used in manufacturing and warehouses for material transport. They play a vital role in just-in-time material delivery, ensuring smooth production lines. However, disruptions in their fixed routes can lead to efficiency losses. This paper studies unexpected disruptions in mobile robot routes, assessing the costs of rectification using the E* analysis. The study highlights the significant additional cost of such disruptions, underscoring the importance of their consideration in AGV route planning to enhance system resilience.

Keywords: mobile robots; AGV; route planning

1. Introduction

Automated Guided Vehicles (AGVs) play a pivotal role in flexible manufacturing systems, material transportation, and stock management. Their adoption stems from their ability to enhance productivity, system flexibility, and cost reduction [1]. However, designing efficient AGV systems is intricate due to interrelated decision-making processes and the array of available algorithms and solutions [2].

AGVs employ diverse technologies for workplace navigation, including magnetic or optical guide tracking, Global Positioning System (GPS), and Laser Imaging Detection and Ranging (LIDAR) sensors [3]. Particularly, the organization of the workspace into nodes and arcs, forming a Flow Network Model graph [4], is common in applications where AGVs follow optical or magnetic guides. In this model, nodes denote material pickup and delivery points, and arcs represent predefined paths for AGV movement. The graph-based approach simplifies route planning and optimization.

Algorithms like Bellman-Ford [5], Dijkstra [6] and A* [1] have been applied to determine optimal paths between nodes. While they often prioritize the shortest path, the occurrence of impediments during operation necessitates recalculating routes, potentially leading to costly detours and disruptions of the service. Such scenarios could adversely impact production schedules and necessitate halting manufacturing lines.

This study delves into the emergence of unexpected disruptions in robot trajectories, evaluating the costs associated with rectifying these situations using an approach we named E* analysis. Results make it clear that that the additional cost incurred due to disruptions can be substantial. Therefore, considering this aspect during AGV route identification holds potential for enhancing the resilience of the entire system.

In the subsequent sections, this paper presents a comprehensive analysis of disruptions, explores related works in the field, outlines the proposed methodology for route planning and robustness assessment, and offers insights into the attained results and their implications.

2. Related Work

In the pursuit of optimal paths between origin and destination nodes in a graph, a range of techniques and algorithms have been developed to enhance route planning efficiency and effectiveness. These approaches aim to minimize travel time, prevent collisions with obstacles and other vehicles,

and mitigate the likelihood of congestion. A selection of prominent algorithms and their applications in this context are discussed below.

2.1. Dijkstra's Algorithm

One of the most renowned algorithms for finding the shortest path, Dijkstra's algorithm was formulated by computer scientist Edsger W. Dijkstra in 1956. It addresses the shortest path problem within a graph with non-negative arcs [7]. By choosing the closest node with the lowest weight (arc value), Dijkstra's algorithm systematically processes each node in the graph. The speed of finding a solution depends on the number of nodes, and this algorithm treats the shortest path as the optimal outcome [6,8–13].

2.2. Yen's Algorithm

Yen's algorithm [14,15] can be used to find the k-shortest paths between two nodes. An efficient single shortest path algorithm (like Dijkstra's algorithm) is repeatedly ran on selectively modified versions of the original graph, producing in this way alternative deviations from the single optimal path solution.

2.3. Bellman-Ford's Algorithm

The Bellman-Ford algorithm is employed to compute the shortest distance in a weighted flow network [16]. Unlike Dijkstra's algorithm, Bellman-Ford accommodates negative weights on arcs connecting two vertices. This algorithm is capable of parallel searching and counting of paths, allowing nodes to perform simultaneous processes (multiprocessing). It is especially suitable for scenarios where negative weights are incorporated into the routing decision [5,16–20].

2.4. A Star Algorithm

The A Star (A*) algorithm employs heuristic information to identify an ideal path. Executing node exploration and a heuristic evaluation, this algorithm continually selects the best node for expansion until the destination node is reached [21]. Maintaining both an open list (nodes to be considered) and a closed list (nodes already visited), A* benefits from rapid convergence due to the heuristic approach [1]. A* is extensively utilized for route and path planning problems [22–29].

2.5. Time Enhanced A* Algorithm

A variant of A*, the Time Enhanced A* algorithm, enhances the algorithm's predictive capability in scenarios involving multiple AGVs. Incorporating time as an additional parameter, this algorithm predicts vehicle movements during runtime [30]. The approach constructs trajectories for each vehicle by considering the movements of other mobile robots. This technique facilitates the creation of collision-free routes while dynamically recalculating paths and updating map information iteratively [1,20,31,32].

2.6. Other Algorithms

Additional algorithms, including Probabilistic Roadmaps [20,33–36], Rapidly Exploring Random Trees [1,21,29,32,37,38], and Genetic Algorithms [35,39–42], have been explored for optimal path determination. These algorithms are particularly suited for open spaces and free-form environments.

Ongoing research in this area aims to enhance heuristics and approaches for path selection, especially in graphs with a high number of nodes [11,43].

3. Resilience in the Presence of Pre-Planned Path Disturbances

The concept of robustness considered in this work refers to the capacity to reroute an AGV partially along the pre-planned route, when unforeseen disturbances block parts of pre-planned paths,

with minor or even no cost in terms of increased distance required to reach the target destination. This concept of robustness is different from the one discussed in [44], which is concerned with algebraic connectivity.

The concept of robustness considered in our work is illustrated using the graph in Figure 1a, adapted from [45].

By applying Dijkstra's or A* algorithms, three optimal paths between nodes 1 and 10 can be found, namely:

(i) 1-5-9-10
(ii) 1-5-6-10
(iii) 1-2-6-10

Because all arcs in Figure 1a correspond to 10 meter segments, these paths correspond to alternative 30 meter routes from origin to destination (Figure 1b).

Although they are equivalent from that perspective, considering some specific scenarios of arc disturbance and AGV positioning:

- Consider path (i), represented in red in Figure 1b, with the AGV in arc 1-5 and arc 5-9 becoming unavailable. In this scenario, the AGV can be rerouted successfully via the arc 5-6 and 6-10, leading to the AGV reaching the destination via the route 1-5-6-10, with total distance 30 meter. In this case, the disturbance did not have any actual impact on the total distance that the AGV must travel to reach the destination.
- Consider path (ii), represented in green in Figure 1b, with the AGV in arc 1-5 and arc 5-6 becoming unavailable. Rerouting through 5-9-10 is possible, resulting in a total travel distance of 30 meters from origin to destination. Again, the disturbance has no negative impact on travelled distance.
- Consider path (iii), represented in blue in Figure 1b, with the AGV in arc 1-2 and arc 2-6 becoming unavailable. Four rerouting possibilities exist, based on the AGV's current position and the destination, leading to the AGV travelling the path 1-2-3-7-11-10, 1-2-3-7-6-10, 1-2-1-5-9-10 or 1-2-1-5-6-10. In any one of these cases, the total travelled distance is 50 meter. In this case, the disturbance has an actual negative impact on total travelled distance.
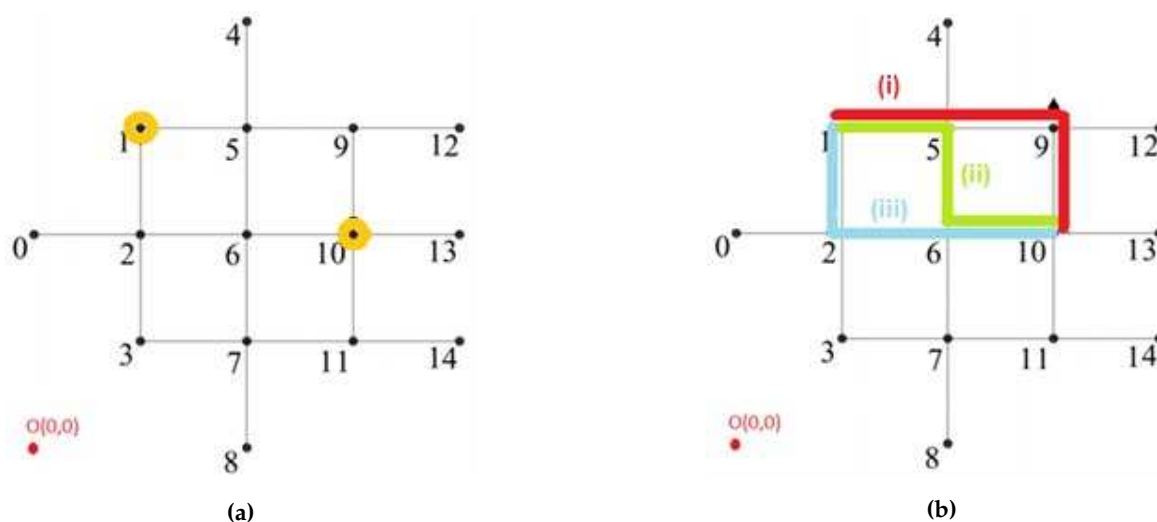

Figure 1. Illustrative example of the concept of robustness. (**a**) Example graph; (**b**) Optimal paths between origin node 1 and destination node 10.

This example makes it clear that some paths are more susceptible to being negatively impacted by unforeseen disturbances on the pre-planned route than others. In our example, we describe paths (i) and (ii) as being more robust than path (iii). One interesting metric is to consider the probability of *accidental robustness*, that is, the probability of randomly selecting a robust path from the entire set of

distance-optimal paths. In the example scenario, two paths out of three are robust, so the probability of accidental robustness is 2/3=66.6%. This is the probability of randomly selecting a robust route if the only criteria used to obtain the candidate routes is path length. Similarly, another metric to consider would be the actual impact in travelled distance when path disturbances occur and non-robust paths are being followed. In the example, travelled distance increases to 50 in non-robust reroutes over 30 in robust routes, so robust routes would only travel on average in 30/50=60% the distance of non-robust reroutes when a disturbance occurs[1].

4. Methodology

The main objective of this work is to evaluate the robustness of AGV paths using the E* analysis, which verifies the impact of unexpected disturbances and obstructions that might unexpectedly appear along the route during operation. The E* analysis evaluates the robustness of the routes by considering the cost of re-routing the AGVs through a new path. In the context of this work, we consider a route to be more robust than other, of similar cost, if the cost of rerouting the AGVs in the presence of a disturbance is lower in the former than in the latter, in terms of total distance travelled, including re-routing movement.

The E* analysis includes the following steps:

1) **Generating candidate paths**. These are the paths that are going to be considered in the analysis.
2) **Generating path disturbances** in each candidate path.
3) **Assessing route repair cost** for each disturbance, dependent on AGV position.

These steps are described in further detail in the next sections.

4.1. Generating Candidate Paths

The first step of E* entails generating a set of paths among which will be considered in the ensuing resilience analysis. This set of paths should include the shortest path or paths, which are most commonly used to determine the preferred routes, as well as possible alternatives. Two approaches that can be considered for generating the candidate set:

– An optimal shortest path solution is obtained using a traditional algorithm (*e.g.*, A*). Alternative paths are then obtained by modification of this initial path using a stochastic approach such as mutation followed by hill-climbing search.
– A k-set of optimal shortest paths is explicitly computed using Yen's algorithm [14,15]. This set might include all alternative possibilities for shortest paths, or also include sub-optimal solutions within a specified threshold of the optimal solution. A hard limit in the size of the candidate set can be imposed for practical purposes.

The first approach ensures at least one route with optimal distance is included in the candidate set. The stochastic process will then generate other variations of this original path. Because this requires computing only a single optimal path, an efficient single solution algorithm such as A* can be used to compute the initial solution. Advantages of this approach are higher efficiency, even for large networks. However, the stochastic algorithm may fail to identify all optimal paths.

In contrast, the second approach entails the systematic generation of all optimal paths using Yen's algorithm. While this ensures all these paths are considered in the E* analysis, Yen's algorithm requires repeated use of Dijkstra's algorithm, which might result in a performance-intensive operation, especially in large graphs.

---

[1]   This analysis is simplified by considering only the specific disturbances and AGV positioning described in the example. In practice, all other possibilities would also be accounted for.

4.2. Generating Path Disturbances

Once the candidate paths are found, disturbances will be applied. To do so, different scenarios are generated for each candidate path. In each such scenario, a single arc along the path is made unavailable, representing an unforeseen disturbance during operation. Scenarios will include disturbances for all possible arcs. Furthermore, each possible position (arc-wise) of the AGV along the path up to the unavailable arc is also considered in an independent scenario. In this way, the disturbance scenario set will contemplate all possible combinations of arc unavailability and positioning of the AGV along the path up to the disturbance. It is not necessary to consider AGV positions beyond the disturbance for the purposes of our analysis, because the target node may be reached by following along the pre-planned route in that scenario.

4.3. Assessing Route Repair Cost

On each of the disturbance scenarios that is identified using the approach discussed in Section 4.2, it is no longer possible for the AGV to follow along the pre-planned path towards the destination node, because one of the required path arcs is no longer available.

For the AGV to progress towards the desired destination, it will be necessary to find an alternative path originating at the current position of the AGV. This alternative route can be found by applying one of the standard optimal path algorithms (such as A* or Dijkstra algorithms). It is possible that a new route may be discovered which does not extend the remaining distance to be covered before reaching the destination; however, this is not always guaranteed. In the latter scenario, there is a risk that the AGV could fail to meet the necessary deadlines for ensuring just-in-time distribution of materials. Such a delay could have a detrimental effect on the overall efficiency of the manufacturing plant. Consequently, we calculate the route repair cost as the additional distance that must now be traversed in comparison to the original path.

5. Results

To assess the gains in robustness when the E* analysis is considered, we devised a set of experiments aimed at determining:

a) What is the magnitude of the improvement obtained by using robust paths vs. non-robust paths, in the presence of disturbances?
b) What is the likelihood of obtaining a robust path when not specifically looking for it?
c) What are the trade-offs of using Yen's algorithm vs. using the stochastic algorithm, particularly with respect to execution time and result quality?

To answer these questions, we applied the E* approach to two sets of data:

- **Input Set A**: Ten AGV paths described in the literature (adapted from [4,10,11,45–51]).
- **Input Set B**: A set of 100 randomly generated graphs, with varying dimensions ranging up to 240 edges and up to 150 nodes.

5.1. Ratio of Robust vs. Non-Robust Paths

To address the first question, results for input set A are presented in Figure 2, representing the number of optimal paths identified by Dijkstra's algorithm and stochastic approach (considering only on the route distance), compared to the number of optimal paths identified after the E* analysis (considering not only the route distance but also the costs of rerouting due to disturbances in the path). As it can be expect, not all paths considered optimal in the former sense are optimal when considering the rerouting costs. By comparing the number of optimal solutions found for each metric, we can determine the probability of obtaining a robust path when not specifically looking for it, as indicated by the percentage displayed in each pair of columns in Figure 2.
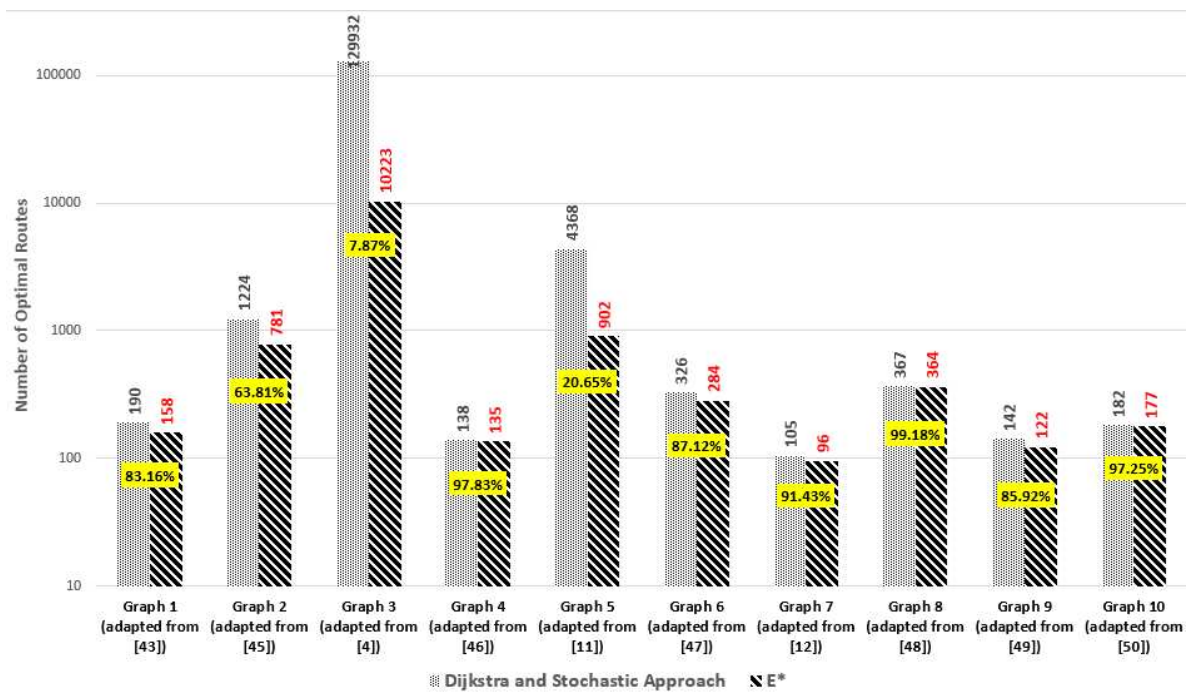
Figure 2. Number of optimal paths obtained by using Dijkstra's algorithm and stochastic approach and E*. The number of optimal solutions is described on a logarithmic scale. The percentage values describe the relation between the number of optimal solutions found according to the E* approach and shortest-path only solutions.

Similar results can be found for Set B in Figure 3. In this case, both the stochastic approach and Yen's algorithm were used to compute the optimal path set, and the ratio of robust vs. non-robust paths is obtained.
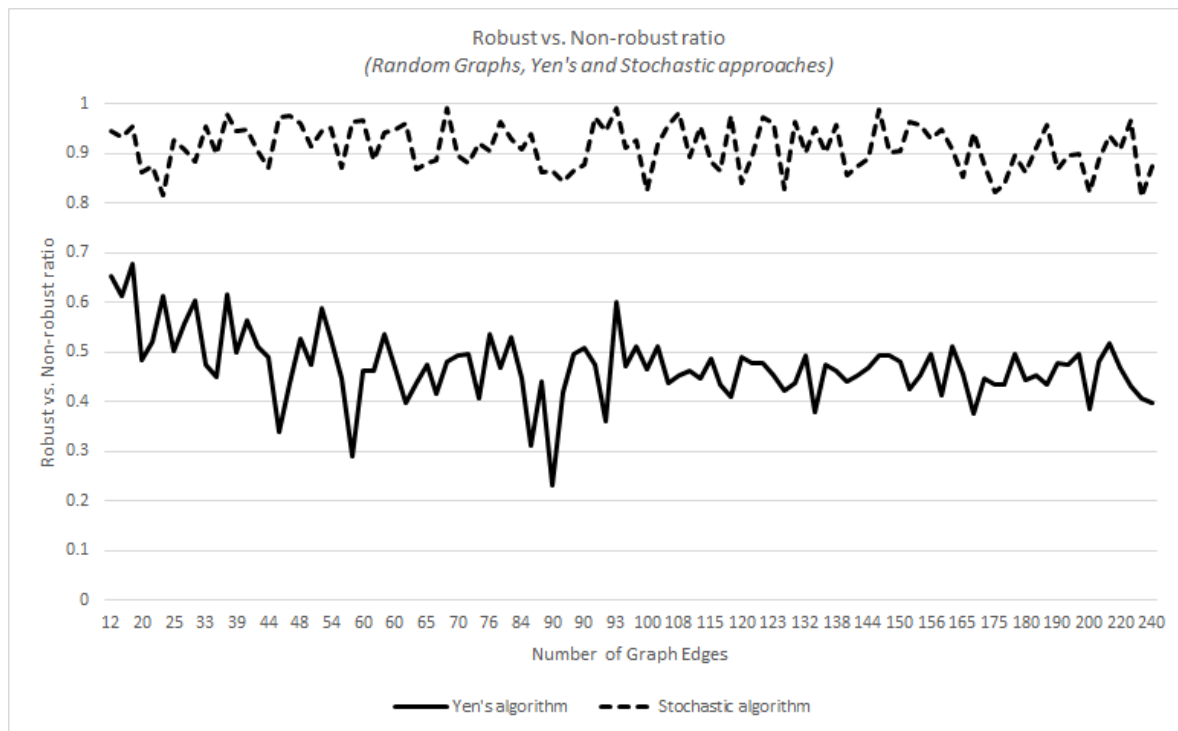
7 of 15

Figure 3. Robust path ratios computed using both Yen's algorithm and the stochastic approach when applied to input set B, composed of 100 randomly generated graphs. The vertical axis represents the ratio expressed in the range 0-1, while the horizontal axis represents the number of edges of the corresponding graph.

5.2. Magnitude of Improvement Achieved by Robust Paths under the Presence of Disturbances

Figure 4 represents the average values of the ratio of the distance travelled under the presence of disturbances when robust paths are used, with respect to the distance travelled when shortest but non-robust paths are used. To obtain these values, disturbances in all possible arcs of the path between the current AGV position and the target destination were considered, that forced the identification of a new path to the destination starting at the current AGV position. Furthermore, to do this, we considered all possible intermediate location of the AGV along the predetermined path.
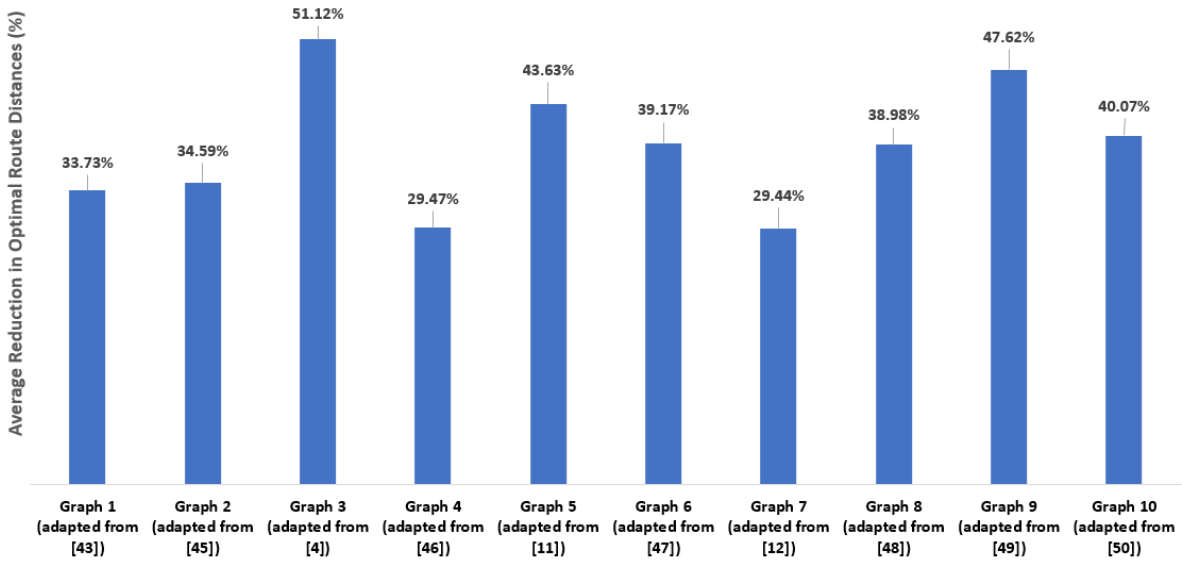
Figure 4. Average reduction distance of rerouted robust paths with respect to the distance of rerouted non-robust paths.

To assess the impact of the path generation approach and also to collect results from a wider range of graphs, a similar analysis was applied to the graphs in input set B. Figure 5 presents the reduction in the length of the path when a disturbance occurs, for both path generation algorithms.
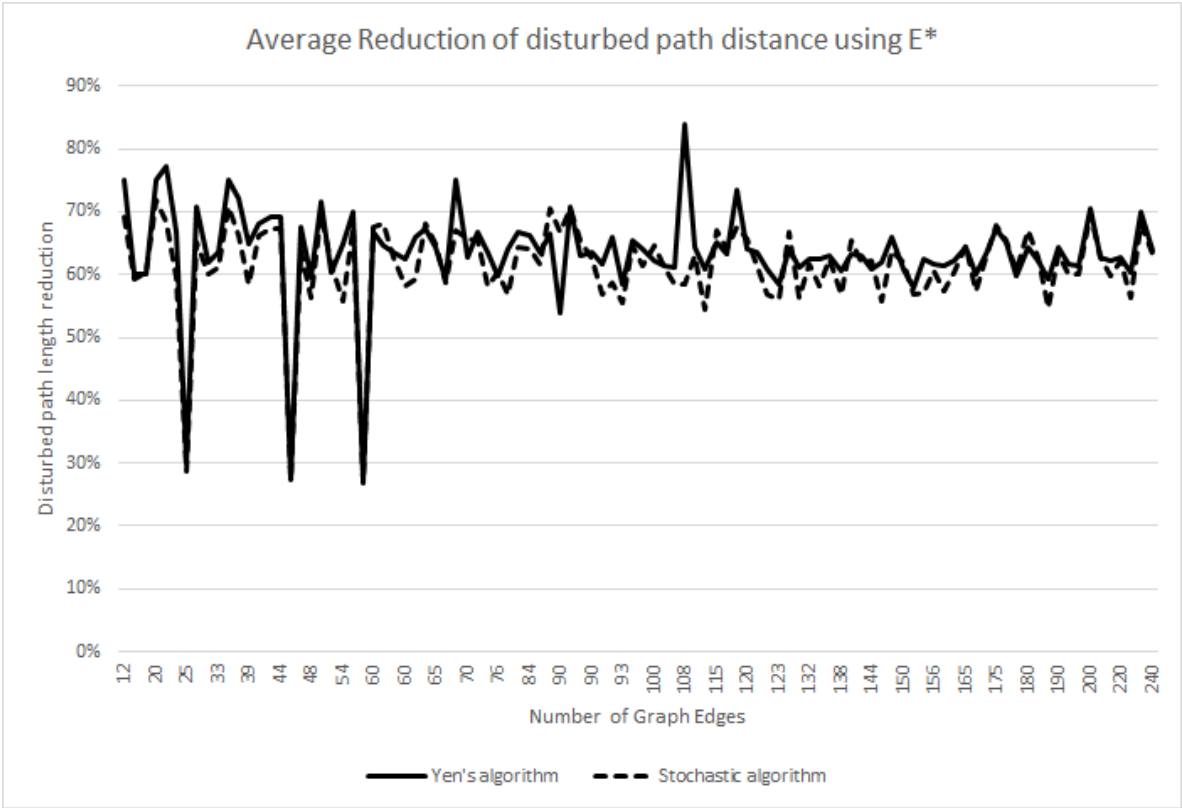

Figure 5. Reduction of path length when a disturbance occurs and the E* approach is used, using both Yen's algorithm and the stochastic approach. Input set B, composed of 100 randomly generated graphs, was used. The vertical axis represents a percentage representing the reduction in path length: the ratio of the disturbed path size when using E* to the disturbed path size when E* is not used. The horizontal axis represents the number of edges of the corresponding graph.

5.3. Execution Time Performance of Yen's and Stochastic Algorithms

 To assess whether there are significant advantages in using the stochastic approach with respect to Yen's algorithm in terms of execution speed we applied both to the graphs in input set B and collected the results. Input set A was not considered for this purpose because of it mostly includes graphs of low to medium dimension, for which execution time differences are not very meaningful. Execution time for all graphs is presented in Figure 6, while Figure 7 presents a zoomed in view of the results for graphs with a lower number of edges.
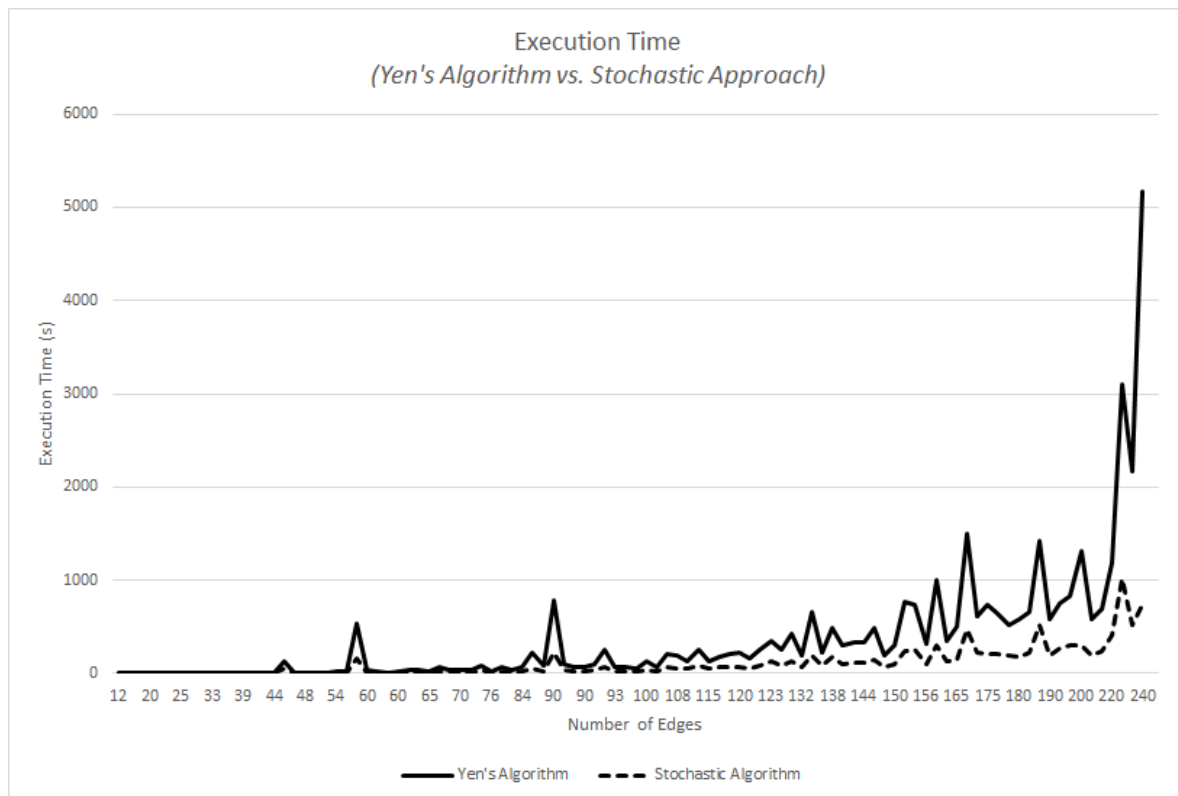

Figure 6. Execution time of path generation using both Yen's algorithm and the stochastic approach applied to input set B, composed of 100 randomly generated graphs. The vertical axis represents the execution time in seconds, while the horizontal axis represents the number of edges of the corresponding graphs.
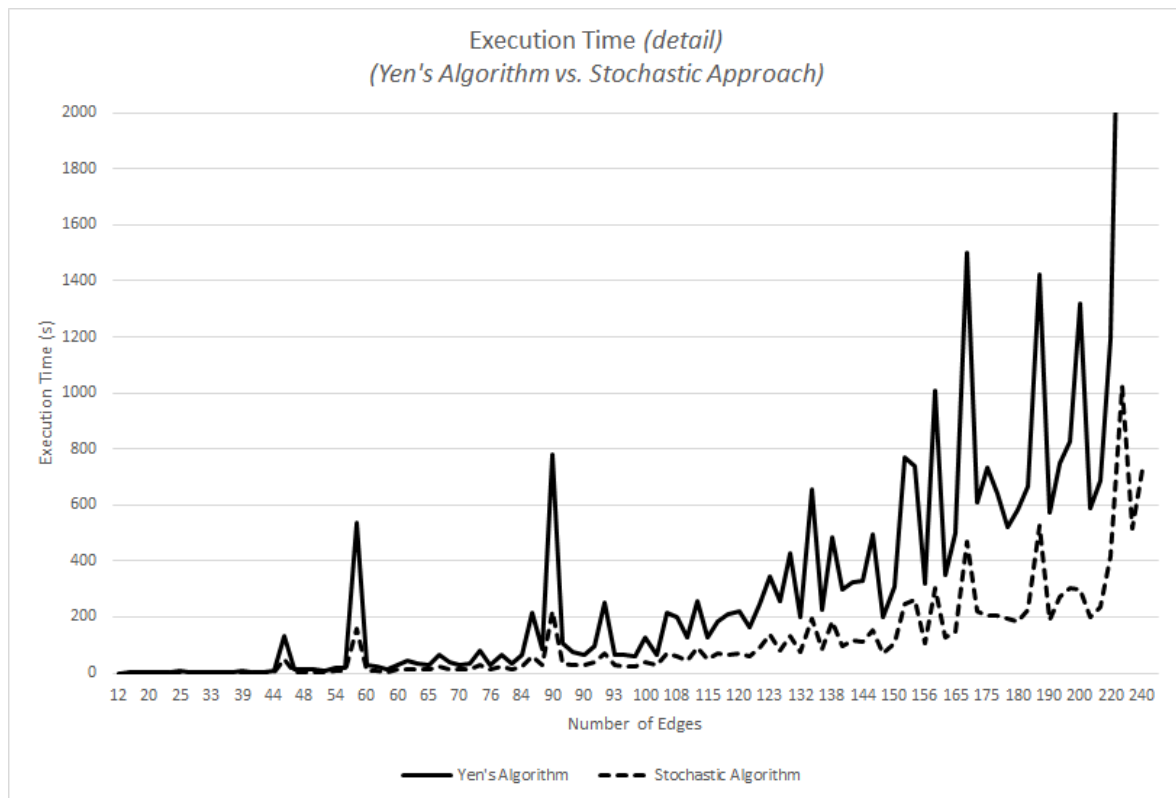
Figure 7. Detail of the information presented in Figure 6, zooming in on the graphs with lower edge count.

6. Discussion

Results in Figure 2, obtained with input set A, demonstrate that path robustness is not ensured if the only criteria for route creating are based on path distance. In average, for all scenarios, 73,42% of all paths generated using only shortest-distance criteria were robust to disturbances. Further analysis of results reveals, however, that while the probability of accidental robustness is reasonably high for simpler graphs with fewer alternative paths, it will decrease rapidly as the complexity of the graph and number of alternative paths increases. In the most complex graph of input set A, only 7.87% of the distance-optimal solutions are also robust. There is, however, a large variance of results, so for definitive conclusions we should also consider similar metrics for input set B, which includes a larger number of randomly generated graphs.

The results disturbed path ratios for input set B can be seen in Figure 3. It can be observed that, when using the stochastic algorithm for generating optimal paths, the observed ratio of robust to non-robust paths was much higher (on average 91%) than those of input set A. In contrast, when Yen's algorithm is used, the same metric drops to an average value of 47%. The values obtained when using Yen's algorithm are most likely more representative of the real ratio, because it generates all distance-optimal paths, while the stochastic algorithm can only provide a best-effort approximation of the distance-optimal path set.

Concerning the average improvement of total distance travelled when re-routing is required due to unexpected disturbances, results in Figure 4 indicate that, when using E* with input set A, the length of the paths subjected to disturbances is reduced to only 61.22% of the solutions based only on distance-optimal criteria. For input set B, no significant difference is found when comparing the results using either distance-optimal path generators (61.5% for the stochastic algorithm and 63.5% when using Yen's algorithm) (See Figure 5). These results are also in line with those that were obtained using input set A. This result demonstrates that significant improvements of performance on disturbance scenarios can be gained if path robustness is considered during route selection.

Results in Figures 6 and 7 demonstrate that, as expected, the stochastic algorithm is faster than Yen's algorithm. The speedup gained from using the stochastic algorithm instead of Yen's algorithm is represented in Figure 8. It can be seen that the speedup slowly improves with increasing number of graph edges, with a speedup of approximately 3 for an average sized graph of input set B.
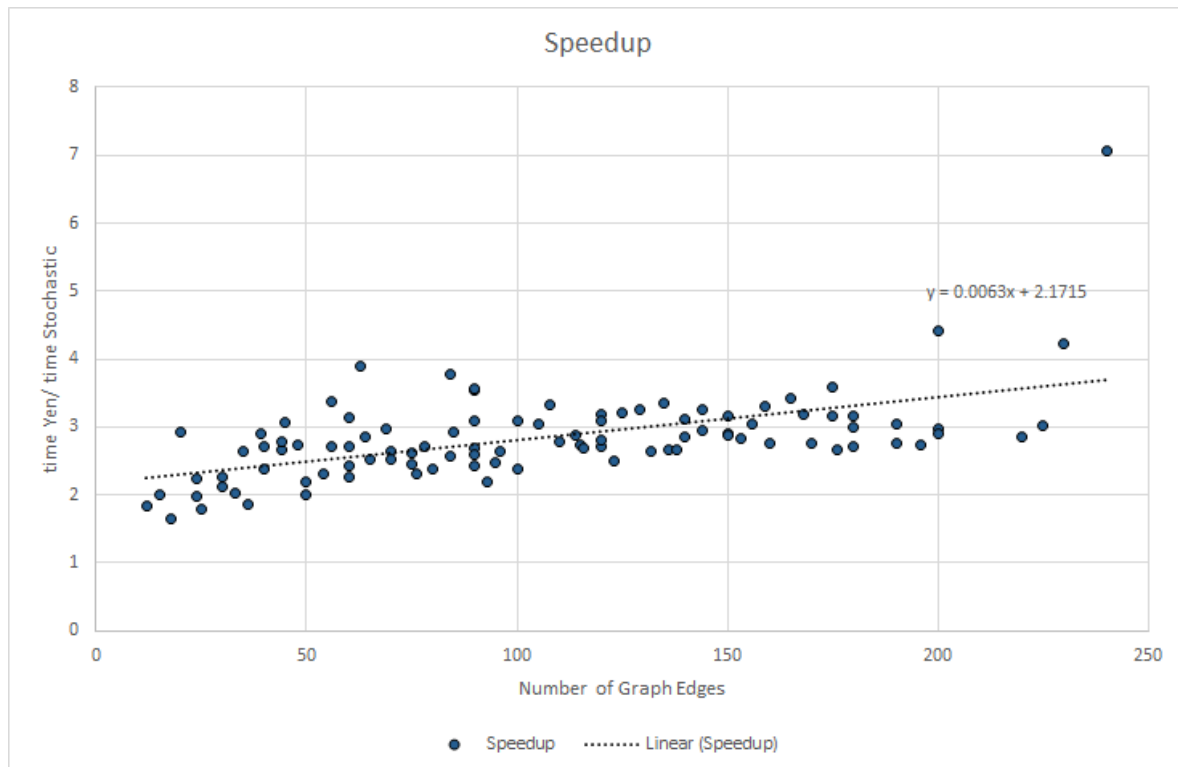

Figure 8. Speedup obtained when using the Stochastic algorithm instead of Yen's algorithm. The vertical axis represents the ratio of execution time in seconds (Yen's to Stochastic algorithm), while the horizontal axis represents the number of edges of the corresponding graphs.

Summarizing our findings, and answering the questions posed at the start of Section 5:

a) *What is the magnitude of the improvement obtained by using robust paths vs. non-robust paths, in the presence of disturbances?*

When in the presence to unforeseen disturbances requiring rerouting an AGV from its current route, we found that the length of robust routes was just slightly over 60% of the corresponding values. This improvement of nearly 40% demonstrates the usefulness of considering robustness in route design.

b) *What is the likelihood of obtaining a robust path when not specifically looking for it?*

While specific results depend on the input set and route generator being used, data in Figure 2 and Figure 3 demonstrate that accidental robustness is generally unlikely to occur when routing only with optimal distance considerations.

c) *What are the trade-offs of using Yen's algorithm vs. using the stochastic algorithm, particularly with respect to execution time and result quality?*

No discernible difference in effectiveness is found when using either approach (Figure 5). Execution time benefits the stochastic approach, with speedups in the range of 2 to 4 for the tested graphs (Figures 6–8). In spite of this improvement, it appears that the benefit is mostly linear which might not be sufficient to scale well for operation in graphs of very high dimension.

7. Conclusions and Future Work

In this paper, we propose the E* analysis approach, that identifies AGV routes with high robustness, in terms of resilience against unexpected disturbances in pre-planned routes. We demonstrated that routes of equivalent distance may have different robustness properties, suggesting that route planning based on distance criteria only may not be generate robust solutions.

High robustness mitigates unforeseen disruptions of planned routes in industrial applications, reducing wasted time and allowing the continued operation of production processes.

Future work can address issues such as improving the efficiency of E*, or exploring its use in other contexts such as scenarios with multiple AGVs and multiple routes.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Costa, M.M.; Silva, M.F. A Survey on Path Planning Algorithms for Mobile Robots. 2019 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC), 2019, pp. 1–7. doi:10.1109/ICARSC.2019.8733623.
2. Duinkerken, M.B.; Ottjes, J.A.; Lodewijks, G. Comparison of Routing Strategies For AGV Systems Using Simulation. Winter Simulation Conference - Netherlands, 2006, pp. 1523–1530.
3. Zhang, Y.F.; Zhang, W.; Liu, S.H.; Chang, H.; Qin, G.; Chen, Y.H. Research on AGV Navigation Simulation in Printing Wisdom Factory. IEEE Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), 2021, pp. 2312–2316. doi:10.1109/IAEAC50856.2021.9390739.
4. Nishi, T.; Akiyama, S.; Higashi, T.; Kumagai, K. Cell-Based Local Search Heuristics for Guide Path Design of Automated Guided Vehicle Systems with Dynamic Multicommodity Flow. *IEEE Transactions on Automation Science and Engineering* **2020**, *17*, 966–980. doi:10.1109/TASE.2019.2952920.
5. Nazarifard, M.; Bahrepour, D. Efficient Implementation of the Bellman-Ford Algorithm on GPU. IEEE 4th International Conference on Knowledge-Based Engineering and Innovation (KBEI), 2017, pp. 773–778.
6. Dijkstra, E.W. A Note on Two Problems in Connexion with Graphs. Numerische Mathematlk l, 1959, pp. 269–296.
7. Candra, A.; Budiman, M.A.; Hartanto, K. Dijkstra's and A-Star in Finding the Shortest Path: A Tutorial. International Conference on Data Science, Artificial Intelligence, and Business Analytics (DATABIA), 2020, pp. 28–32.
8. Parekh, S.; Jha, A.; Dalvi, A.; Siddavatam, I. An Exhaustive Approach Orchestrating Negative Edges for Dijkstra's Algorithm. IEEE 7th International Conference for Convergence in Technology, I2CT. Institute of Electrical and Electronics Engineers Inc., 2022, pp. 1–5. doi:10.1109/I2CT54291.2022.9824795.
9. Wenzheng, L.; Junjun, L.; Shunli, Y. An Improved Dijkstra's Algorithm for Shortest Path Planning on 2D Grid Maps. 2019 IEEE 9th International Conference on Electronics Information and Emergency Communication (ICEIEC). IEEE, 2019, pp. 438–441. doi:10.1109/ICEIEC.2019.8784487.
10. Sungkwan, K.; Hojun, J.; Minah, S.; Dongsoo, H. Optimal Path Planning of Automated Guided Vehicle using Dijkstra Algorithm under Dynamic Conditions. 7th International Conference on Robot Intelligence Technology and Applications (RiTA), 2019, pp. 231–236.
11. Gbadamosi, O.A.; Aremu, D.R. Design of a Modified Dijkstra's Algorithm for finding alternate routes for shortest-path problems with huge costs. 2020 International Conference in Mathematics, Computer Engineering and Computer Science, ICMCECS 2020. Institute of Electrical and Electronics Engineers Inc., 2020. doi:10.1109/ICMCECS47690.2020.240873.
12. Cruz, J.C.D.; Magwili, G.V.; Mundo, J.P.E.; Gregorio, G.P.B.; Lamoca, M.L.L.; Villaseñor, J.A. Items-mapping and route optimization in a grocery store using Dijkstra's, Bellman-Ford and Floyd-Warshall Algorithms. IEEE Region 10 Conference (TENCON) : November 22–25, 2016, Marina Bay Sands, Singapore., 2016.
13. Yin, C.; Wang, H. Developed Dijkstra shortest path search algorithm and simulation. 2010 International Conference on Computer Design and Applications, ICCDA 2010, 2010, Vol. 1. doi:10.1109/ICCDA.2010.5541129.
14. Yen, J.Y. An algorithm for finding shortest routes from all source nodes to a given destination in general networks. *Quart. Appl. Math* **1970**, pp. 526–530. doi:10.1090/qam/253822.

15. Martins, E.Q.; Pascoal, M.M.   A new implementation of Yen's ranking loopless paths algorithm. *Quarterly Journal of the Belgian, French and Italian Operations Research Societies* **2003**, *1*, 121–133. doi:10.1007/s10288-002-0010-2.

16. Adytoma, W.W.; Huda, A.F.; Maylawati, D.S.; Arianti, N.D.; Darmalaksana, W.; Rahman, A.; Ramdhani, M.A. Automatic Text Summarization for Hadith with Indonesian Text using Bellman-Ford Algorithm. 2020 6th International Conference on Computing Engineering and Design (ICCED), 2020, pp. 1–6. doi:10.1109/ICCED51276.2020.9415864.

17. Faychuk, V.; Lavriv, O.; Klymash, M. Investigation of the Bellman-Ford Algorithm Enhanced for Remote Execution. 3rd International Conference on Advanced Information and Communications Technologies (AICT), 2019, pp. 204–208.

18. Weber, A.; Kreuzer, M.; Knoll, A. A Generalized Bellman-Ford Algorithm for Application in Symbolic Optimal Control. European Control Conference (ECC), St. Petersburg, Russia, 2020, pp. 2007–2014.

19. Schambers, A.; Eavis-O'Quinn, M.; Roberge, D.V.; Tarbouchi, D.M. Route Planning for Electric Vehicle Efficiency Using the Bellman-Ford Algorithm on an Embedded GPU. 4th International Conference on Optimization and Applications (ICOA), 2018, pp. 1–6.

20. Khanmirza, E.; Haghbeigi, M.; Nazarahari, M.; Doostie, S. A Comparative Study of Deterministic and Probabilistic Mobile Robot Path Planning Algorithms. 5th RSI International Conference on Robotics and Mechatronics (ICRoM), Tehran, Iran, 2017, pp. 534–539.

21. Wang, J.; Chi, W.; Li, C.; Wang, C.; Meng, M.Q. Neural RRT: Learning-Based Optimal Path Planning. *IEEE Transactions on Automation Science and Engineering* **2020**, *17*, 1748–1758. doi:10.1109/TASE.2020.2976560.

22. Damle, V.P.; Susan, S. Dynamic Algorithm for Path Planning using A-Star with Distance Constraint. 2022 2nd International Conference on Intelligent Technologies, CONIT 2022. Institute of Electrical and Electronics Engineers Inc., 2022. doi:10.1109/CONIT55038.2022.9847869.

23. Kuang, H.; Li, Y.; Zhang, Y.; Feng, Y. Improved A-star Algorithm based on Topological Maps for Indoor Mobile Robot Path Planning. IEEE 6th Information Technology and Mechatronics Engineering Conference, ITOEC 2022. Institute of Electrical and Electronics Engineers Inc., 2022, pp. 1236–1240. doi:10.1109/ITOEC53115.2022.9734323.

24. Jianqin, L.; Xiao, G. Research on improved A-star algorithm for global path planning of unmanned logistics vehicles.   14th International Conference on Intelligent Human-Machine Systems and Cybernetics, IHMSC 2022. Institute of Electrical and Electronics Engineers Inc., 2022, pp. 44–47. doi:10.1109/IHMSC55436.2022.00019.

25. Guo, J.; Huo, X.; Guo, S.; Xu, J.   A Path Planning Method for the Spherical Amphibious Robot Based on Improved A-star Algorithm.   2021 IEEE International Conference on Mechatronics and Automation, ICMA 2021. Institute of Electrical and Electronics Engineers Inc., 2021, pp. 1274–1279. doi:10.1109/ICMA52036.2021.9512805.

26. Huang, D.; Zhao, Y.; Li, Q.; Wu, T. Research on Path Planning of Mobile Robot Based on Improved A-Star Algorithm.   Proceedings - 2022 International Conference on Informatics, Networking and Computing, ICINC 2022. Institute of Electrical and Electronics Engineers Inc., 2022, pp. 251–255. doi:10.1109/ICINC58035.2022.00058.

27. Song, Y.; Ma, P. Research on Mobile Robot Path Planning Based on Improved A-star Algorithm. 2021 International Conference on Electronic Information Engineering and Computer Science, EIECS 2021. Institute of Electrical and Electronics Engineers Inc., 2021, pp. 683–687. doi:10.1109/EIECS53707.2021.9588002.

28. Zhang, Z.; Wang, S.; Zhou, J. A-star algorithm for expanding the number of search directions in path planning. 2nd International Seminar on Artificial Intelligence, Networking and Information Technology, AINIT 2021. Institute of Electrical and Electronics Engineers Inc., 2021, pp. 208–211. doi:10.1109/AINIT54228.2021.00049.

29. Zhou, Q.; Liu, G. UAV Path Planning Based on the Combination of A-star Algorithm and RRT-star Algorithm. IEEE International Conference on Unmanned Systems, ICUS 2022. Institute of Electrical and Electronics Engineers Inc., 2022, pp. 146–151. doi:10.1109/ICUS55513.2022.9986703.

30. Santos, J.; Costa, P.; Rocha, L.; Vivaldini, K.; Moreira, A.; Veiga, G. Validation of a Time Based Routing Algorithm Using a Realistic Automatic Warehouse Scenario. Second Iberian Robotics Conference. Advances in Intelligent Systems and Computing, 2015, pp. 81–92.

31. Santos, J.; Costa, P.; Rocha, L.F.; Moreira, A.P.; Veiga, G. Time Enhanced A*: Towards the development of a new approach for Multi-Robot Coordination. IEEE International Conference on Industrial Technology (ICIT), Seville, Spain,, 2015, pp. 3314–3319.

32. Chen, G.; Hou, J.; Dong, J.; Li, Z.; Gu, S.; Zhang, B.; Yu, J.; Knoll, A. Multiobjective Scheduling Strategy with Genetic Algorithm and Time-Enhanced A*Planning for Autonomous Parking Robotics in High-Density Unmanned Parking Lots. *IEEE/ASME Transactions on Mechatronics* **2021**, *26*, 1547–1557. doi:10.1109/TMECH.2020.3023261.

33. Ravankar, A.A.; Ravankar, A.; Emaru, T.; Kobayashi, Y. HPPRM: Hybrid Potential Based Probabilistic Roadmap Algorithm for Improved Dynamic Path Planning of Mobile Robots. *IEEE Access* **2020**, *8*, 221743–221766. doi:10.1109/ACCESS.2020.3043333.

34. Sudhakara, P.; Ganapathy, V.; Sundaran, K. Probabilistic roadmaps-spline based trajectory planning for wheeled mobile robot. International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), Chennai, India, 2017, pp. 3579–3583.

35. Santiago, R.M.C.; Ocampo, A.L.D.; Ubando, A.T.; Bandala, A.A.; Dadios, E.P. Path planning for mobile robots using genetic algorithm and probabilistic roadmap. IEEE 9th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM), Manila, Philippines, 2017, pp. 1–5.

36. Yang, J.; Dymond, P.; Jenkin, M. Practicality-based probabilistic roadmaps method. Canadian Conference on Computer and Robot Vision, CRV 2011, 2011, pp. 102–108. doi:10.1109/CRV.2011.21.

37. Chen, Q.; Wang, M. An improved RRT*algorithm for mobile robots path planning. Proceedings - 2022 Chinese Automation Congress, CAC 2022. Institute of Electrical and Electronics Engineers Inc., 2022, Vol. 2022-January, pp. 4334–4339. doi:10.1109/CAC57257.2022.10055894.

38. Chen, Z.; Zhang, X.; Wang, L.; Xia, Y. A Fast Path Planning Method Based on RRT Star Algorithm. 3rd International Conference on Consumer Electronics and Computer Engineering (ICCECE), Guangzhou, China. Institute of Electrical and Electronics Engineers (IEEE), 2023, pp. 258–262. doi:10.1109/iccece58074.2023.10135365.

39. Changan, L.; Xiaohu, Y.; Chunyang, L.; Guodong, L. Dynamic Path Planning for Mobile Robot Based on Improved Genetic Algorithm. *Chinese Journal of Electronics* **2010**, *19*.

40. Long, Y.; Su, Y.; Zhang, H.; Li, M. Application of Improved Genetic Algorithm to Unmanned Surface Vehicle Path Planning. IEEE 7th Data Driven Control and Learning Systems Conference (DDCLS), Enshi, China, 2018, pp. 209–212.

41. Liu, J.; Xi, B.; Chen, S.; Gao, F.; Wang, Z.; Long, Y. The Path Planning Study of Autonomous Patrol Robot based on Modified Astar Algorithm and Genetic Algorithm. 34th Chinese Control and Decision Conference (CCDC), Hefei, China. Institute of Electrical and Electronics Engineers Inc., 2022, pp. 4713–4718. doi:10.1109/CCDC55256.2022.10033990.

42. Naderan-Tahan, M.; Manzuri-Shalmani, T. Efficient and safe path planning for a Mobile robot using genetic algorithm. IEEE Congress on Evolutionary Computation, CEC, 2009, pp. 2091–2097. doi:10.1109/CEC.2009.4983199.

43. Asani, E.O.; Ayoola, J.; Okeyinka, A.E. A Preliminary Study on the Complexity of Some Heuristics for Solving Combinatorial Optimization Problems. *International Journal of Engineering Research and Technology* **2019**, *12*, 1615–1620.

44. Jamakovic, A.; Mieghem, P.V. On the Robustness of Complex Networks by Using the Algebraic Connectivity. International Conference on Research in Networking. Springer Berlin Heidelberg, 2008, pp. 183–194.

45. Jia, F.; Ren, C.; Chen, Y.; Xu, Z. A System Control Strategy of a Conflict-free Multi-AGV Routing based on Improved A* Algorithm. 24th International Conference on Mechatronics and Machine Vision in Practice (M2VIP), 2017.

46. Sun, S.; Gu, C.; Wan, Q.; Huang, H.; Jia, X.; Fellow, I. CROTPN Based Collision-Free and Deadlock-Free Path Planning of AGVs in Logistic Center. 15th International Conference on Control, Automation, Robotics and Vision (ICARCV) - Singapore, 2018, pp. 1685–1691. doi:10.0/Linux-x86_64.

47. Li, Q.; Liu, G.; Zhang, W.; Zhao, C.; Yin, Y.; Wang, Z. A Specific Genetic Algorithm for Optimum Path Planning in Intelligent Transportation System. 6th International Conference on ITS Telecommunications, Chengdu, China, 2006, pp. 140–143.

48. Guo, H.; Hou, X.; Cao, Z.; Zhang, J. GP3: Gaussian Process Path Planning for Reliable Shortest Path in Transportation Networks. *IEEE Transactions on Intelligent Transportation Systems* **2022**, *23*, 11575–11590. doi:10.1109/TITS.2021.3105415.

49. Sun, Q.; Wan, W.; Chen, G.; Feng, X. Path planning algorithm under specific constraints in weighted directed graph. ICALIP 2016 - 2016 International Conference on Audio, Language and Image Processing. Institute of Electrical and Electronics Engineers Inc., 2017, pp. 635–640. doi:10.1109/ICALIP.2016.7846631.

50. Rojas, J.L.; Gonzalez, J.G.; Boutleux, E.; Niel, E. Path search algorithm minimizing interferences with envisaged operations in a pipe network. 2009 European Control Conference, ECC 2009. Institute of Electrical and Electronics Engineers Inc., 2014, pp. 2682–2687. doi:10.23919/ecc.2009.7074811.

51. Pramudita, R.; Heryanto, H.; Handayanto, R.T.; Setiyadi, D.; Arifin, R.W.; Safitri, N. Shortest Path Calculation Algorithms for Geographic Information Systems. Fourth International Conference on Informatics and Computing (ICIC), Semarang, Indonesia, 2019, pp. 1–5.