# Shortest Path Algorithms for Smart City Emergency Routing: Scalability and Scenario-Based Analysis

Aleksandra Stojanova Ilievska, Natasha Stojkovikj,
Limonka Koceva Lazarova, Arta Aliu, Krume Tasev, Teona Aleksova
Faculty of Computer Science, Goce Delcev University, Stip, North Macedonia

***Abstract*—Shortest path algorithms play an essential role in intelligent transportation systems, emergency response planning, and numerous Smart City services. This study investigates the applicability of three well-established graph-based algorithms, namely Dijkstra, Bellman-Ford, and A*, within an emergency transportation framework inspired by the urban road network of Skopje. A two-stage evaluation methodology was adopted. First, scalability experiments were performed on weighted graphs containing 50 to 500 nodes to examine computational efficiency and algorithm behavior as network size increases. Subsequently, a Smart City transportation model was developed and implemented using Java and JavaFX, enabling the simulation and visualization of several emergency scenarios, including ambulance dispatching, road disruptions, and route unavailability situations. The experimental results indicate that all investigated algorithms successfully identify shortest feasible routes when transportation paths exist, whereas the heuristic guidance employed by A* reduces the number of explored nodes and improves execution times, particularly in larger graph instances. The proposed framework illustrates how algorithmic analysis can be complemented by simulation-based case studies to support intelligent transportation applications and emergency management decision-making. Future research will focus on incorporating real-time traffic information, stochastic travel times, and larger transportation networks to provide more realistic emergency routing environments.**

***Keywords*—Shortest path algorithms; dijkstra; bellman-ford; A*; smart city; emergency routing; intelligent transportation systems; simulation**

## I. Introduction

The increasing urbanization observed worldwide has considerably intensified the demand for efficient transportation systems capable of supporting sustainable, resilient, and safe city development. Consequently, the concept of Smart Cities has attracted significant attention from researchers, policymakers, and industry practitioners. Smart Cities rely on the integration of digital technologies, intelligent infrastructures, and data-driven decision-making processes to improve urban services and enhance citizens' quality of life. Among the numerous application domains, intelligent transportation systems have emerged as one of the most important components of modern Smart Cities due to their direct impact on mobility, accessibility, and emergency response capabilities [1]–[3].

Emergency medical services represent a particularly challenging aspect of urban transportation management. In many situations, even relatively small delays in ambulance dispatching and patient transportation may significantly affect treatment outcomes and public safety. Therefore, identifying efficient routes under dynamically changing traffic conditions has become an important research topic within intelligent transportation systems, emergency management, and Smart City initiatives [1]–[3].

Graph theory provides a natural framework for modeling transportation networks, where intersections, districts, or critical facilities can be represented as nodes, while roads connecting neighboring locations are modeled as weighted edges. Within this context, shortest path algorithms have been extensively investigated and successfully applied in navigation systems, communication networks, logistics, and autonomous decision-support applications [4], [5]. Classical methods such as the Bellman-Ford algorithm remain useful for handling graphs containing negative edge weights, whereas Dijkstra's algorithm is widely recognized for its efficiency in solving shortest path problems in networks with non-negative costs [6]–[9].

Heuristic search techniques have further expanded the applicability of graph-based optimization methods. The A* algorithm combines accumulated path costs with heuristic estimations, enabling a substantial reduction of the explored search space while preserving optimality under admissible heuristic functions [10]. Several studies have demonstrated the effectiveness of heuristic approaches in robotic navigation, autonomous systems, and dynamic path-planning applications [11]–[14].

In addition to navigation tasks, shortest path algorithms have found numerous applications in traffic management, ride-sharing services, logistics optimization, disaster response planning, and healthcare transportation systems [4], [5]. Their ability to support routing decisions under different operational constraints makes them particularly suitable for Smart City environments, where transportation networks are often characterized by congestion, limited accessibility, and dynamically changing conditions [8], [9]. As urban mobility becomes increasingly complex, the need for efficient and adaptive routing mechanisms continues to grow.

Advances in computational capabilities and visualization tools have facilitated the development of interactive environments for studying graph algorithms and evaluating their practical performance. Comparative analyses involving multiple shortest path techniques can provide valuable insights into their computational characteristics, scalability, and suitability for various emergency transportation scenarios.

Such analyses may contribute to the design of more efficient intelligent transportation systems and support decision-making processes in time-critical situations [10]–[14].

Although numerous studies have addressed shortest path algorithms and their theoretical properties, comparatively fewer investigations have examined their behavior within integrated Smart City environments, particularly in emergency transportation scenarios involving road closures, route unavailability, and varying operational conditions. Existing research often focuses either on algorithmic improvements or on large-scale intelligent transportation infrastructures, while experimental comparisons conducted within realistic urban emergency settings remain relatively limited [4], [5], [13]–[15].

Motivated by these observations, this study presents a comparative analysis of Dijkstra, Bellman-Ford, and A* algorithms within a Smart City emergency transportation framework inspired by the urban road infrastructure of Skopje. The proposed methodology combines scalability experiments with a case study involving ambulance dispatching scenarios, road blockage simulations, and emergency routing analysis. Furthermore, a visualization environment implemented using Java and JavaFX enables intuitive representation of transportation networks, optimal routes, and unavailable road segments. The obtained results provide additional insight into the applicability, advantages, and limitations of shortest path algorithms in intelligent transportation and emergency response systems.

The main contributions of this study can be summarized as follows:

- A comparative evaluation of Dijkstra, Bellman-Ford, and A* algorithms under increasing graph sizes;
- The development of a Smart City emergency transportation model inspired by the road network of Skopje;
- The simulation of ambulance dispatching scenarios, including road closures and unavailable routes;
- The implementation of a visualization framework for representing transportation networks and routing results;
- An analysis of algorithm scalability and computational performance under different experimental conditions.

## II. RELATED WORK

Shortest path problems have been extensively investigated over the past decades and continue to represent one of the most important optimization problems in graph theory, operations research, and artificial intelligence. Their practical importance extends beyond theoretical studies, encompassing applications in navigation systems, communication networks, logistics, transportation planning, and autonomous decision-support systems. Comprehensive surveys available in the literature provide an overview of shortest path variants, algorithmic improvements, and emerging application domains, emphasizing their continuing relevance in modern intelligent environments [4], [5].

Several studies have focused on improving classical shortest path algorithms to address the growing complexity of transportation networks. Although Dijkstra's algorithm remains one of the most widely used approaches for graphs with non-negative edge weights [7], numerous modifications have been proposed to improve memory utilization, computational efficiency, and scalability. Recent investigations demonstrated that alternative storage mechanisms and implementation strategies can reduce computational costs, particularly in sparse or large-scale networks [8], [9]. Comparative analyses involving Dijkstra, greedy search, and heuristic methods indicate that algorithm performance is highly dependent on network characteristics, graph density, and routing constraints [12].

Heuristic search methods have attracted considerable attention due to their ability to reduce the explored search space while preserving solution optimality under admissible heuristic functions. The A* algorithm remains one of the most influential heuristic techniques and has been successfully applied in robotic navigation, autonomous systems, game artificial intelligence, and path-planning applications [10]. Several studies reported that modified heuristic strategies can significantly improve navigation efficiency in dynamic and partially known environments, while replanning techniques further enhance adaptability when transportation conditions change unexpectedly [11], [13], [14].

The rapid development of Smart City initiatives has stimulated substantial research efforts toward intelligent transportation systems, urban mobility optimization, and data-driven decision support. Smart transportation infrastructures increasingly rely on interconnected devices, Internet of Things technologies, and real-time information exchange mechanisms to improve traffic management and mobility services [16], [17]. More recently, cooperative transportation frameworks and adaptive path-planning approaches have been proposed to reduce travel times, alleviate congestion, and support efficient traffic flow management [18]. Systematic reviews indicate that intelligent transportation systems are expected to play a key role in future urban environments by integrating sensing technologies, communication platforms, optimization algorithms, and predictive analytics [19].

Emergency medical transportation remains a particularly important research topic because transportation delays may directly affect patient outcomes and overall public safety. Previous studies investigated robust route planning strategies capable of operating under uncertain traffic conditions, demonstrating the importance of adaptive routing mechanisms for emergency response applications [20]. Considerable attention has also been devoted to ambulance redeployment and dispatching problems. Heuristic methods, approximate dynamic programming techniques, and metaheuristic approaches have been shown to improve vehicle availability and reduce response times in dynamically changing environments [21]–[24]. Furthermore, extensive reviews of emergency medical service systems highlight the growing need for optimization-based decision support tools that can efficiently allocate resources and support operational planning [25]–[27].

Simulation techniques have become increasingly important for evaluating transportation systems under realistic operating conditions. Multimethod simulation environments provide opportunities for combining discrete-event, system dynamics, and agent-based approaches within a single modeling framework [28]. Agent-based methodologies, in particular, enable the representation of interactions among individual entities and have been successfully employed for studying complex adaptive systems, educational processes, and healthcare-related applications [29], [30].

Visualization and simulation environments have also been employed to investigate graph optimization problems, support educational activities, and analyze transportation networks. Previous studies demonstrated the usefulness of visualization tools for improving the understanding of network flow algorithms [31], while agent-based methodologies provide flexible mechanisms for representing dynamic processes and educational scenarios [32]. More recently, shortest path optimization techniques have been successfully applied to transportation networks through case studies based on the Macedonian road infrastructure, further emphasizing the practical relevance of graph-based routing methods in real-world environments [33].

Recent advances in artificial intelligence have further contributed to the development of intelligent transportation solutions. AI-enhanced traffic simulations, predictive analytics, and adaptive traffic control mechanisms have demonstrated their potential to improve urban mobility, increase road safety, and support more efficient emergency response operations [34], [35]. In addition, the integration of artificial intelligence with smart ambulance systems and dynamic traffic management frameworks may facilitate faster route adaptation and reduce intervention times in critical situations [36].

Despite the substantial body of literature dedicated to shortest path algorithms, intelligent transportation systems, and emergency vehicle routing, comparative studies combining scalability analysis, visualization environments, and Smart City emergency transportation scenarios within a unified framework remain relatively limited. Motivated by this observation, the present study proposes a two-stage evaluation methodology consisting of scalability experiments and a Smart City emergency transportation case study inspired by the urban road network of Skopje. The proposed framework aims to provide a better understanding of the applicability, efficiency, and limitations of shortest path algorithms in intelligent transportation and emergency management systems.

## III. Theoretical Background and Algorithmic Foundations

Graph theory provides the mathematical foundation for numerous optimization problems, including shortest path determination, network analysis, transportation planning, and intelligent decision-support systems. A graph can be represented as

$$G = (V, E) \tag{1}$$

where, ($V$) denotes the set of vertices (nodes) and ($E$) represents the set of edges connecting pairs of vertices. In weighted graphs, each edge is assigned a numerical value corresponding to distance, travel time, cost, or another optimization criterion.

The shortest path problem aims to determine a path between two vertices such that the total accumulated edge weight is minimized. Due to its broad applicability, this problem has been extensively investigated in transportation systems, robotics, communication networks, and artificial intelligence applications.

### A. Dijkstra's Algorithm

Dijkstra's algorithm [5] is one of the most widely used exact shortest path algorithms for weighted graphs with non-negative edge costs. The algorithm iteratively selects the unvisited vertex with the smallest tentative distance and relaxes its neighboring vertices until the shortest distances to all reachable nodes are determined.

The computational complexity of Dijkstra's algorithm depends on the underlying data structure used for maintaining the priority queue. When implemented using a binary heap, its time complexity is given by

$$O\big((V + E)\log V\big) \tag{2}$$

where, ($V$) denotes the number of vertices and ($E$) represents the number of edges. The corresponding space complexity is

$$O(VE) \tag{3}$$

Due to its efficiency and guaranteed optimality, Dijkstra's algorithm remains widely adopted in transportation planning and navigation systems.

### B. Bellman-Ford Algorithm

Bellman-Ford is another classical shortest path algorithm capable of processing graphs containing negative edge weights [6]. The algorithm repeatedly relaxes all graph edges and can additionally detect negative cycles within the network.

Its worst-case computational complexity is $O(VE)$

while the required memory complexity remains $O(V)$.

Although Bellman-Ford generally requires more computational resources than Dijkstra's algorithm, its ability to handle more general graph structures makes it valuable in communication networks and cost-sensitive routing problems.

### C. A* Algorithm

The A* algorithm combines graph search techniques with heuristic guidance to accelerate shortest path computation [10]. Instead of exploring all possible nodes uniformly, A* prioritizes nodes that appear more promising according to an evaluation function.

The evaluation function is defined as

$$f(n) = g(n) + h(n) \tag{4}$$

where, $(g(n))$ represents the accumulated cost from the source node to the current node, while $(h(n))$ denotes an estimate of the remaining distance to the target node.

For A* to guarantee optimality, the heuristic function should satisfy admissibility and consistency conditions. An admissible heuristic never overestimates the true shortest distance to the destination, whereas a consistent heuristic satisfies.

$$h(n) \leq c(n,m) + h(m) \quad (5)$$

for every neighboring node pair $((n,m))$.

Common heuristic functions include Euclidean and Manhattan distances, which are frequently employed in transportation and navigation applications. Under these assumptions, A* preserves optimality while significantly reducing the number of explored nodes compared to uninformed search methods.

Its worst-case computational complexity remains comparable to Dijkstra's algorithm, although practical implementations often exhibit considerably better performance due to heuristic guidance.

### *D. Shortest Path Algorithms in Intelligent Transportation Systems*

Shortest path algorithms constitute an essential component of intelligent transportation systems and Smart City infrastructures [23-25]. Typical applications include route planning, emergency vehicle dispatching, traffic optimization, autonomous navigation, and mobility management [26–28].

In emergency response scenarios, minimizing travel time is particularly important, as delayed interventions may significantly affect public safety and healthcare outcomes. Consequently, evaluating shortest path algorithms within realistic transportation environments represents an important research direction for intelligent transportation systems and emergency management applications.

## IV. Proposed Smart City Emergency Transportation Framework

### *A. Problem Definition*

Efficient emergency response remains one of the major challenges faced by modern Smart Cities. Delays in ambulance dispatching and patient transportation may significantly affect healthcare outcomes, making route optimization an essential component of intelligent transportation systems.

In this study, the emergency routing problem is formulated as a shortest path problem in a weighted graph representing an urban transportation network. The objective is to determine the fastest route for an ambulance traveling from an arbitrary starting location to a hospital while considering potential road closures and varying transportation conditions.

### *B. Smart City Transportation Network*

To evaluate the applicability of shortest path algorithms in realistic urban environments, a transportation model inspired by the road network of the city of Skopje was developed.

The transportation infrastructure is represented as an undirected weighted graph described by Eq. (1), where nodes correspond to urban districts and edges denote road segments connecting neighboring regions. Edge weights represent estimated travel times between adjacent locations.

The transportation network was constructed using several districts of Skopje, namely Aerodrom, Centar, Karpos, Vlae, Gjorce Petrov, Bit Pazar, Kisela Voda, and a hospital location. Connections between neighboring districts were modeled as weighted edges, where the assigned values represent approximate travel times in minutes.

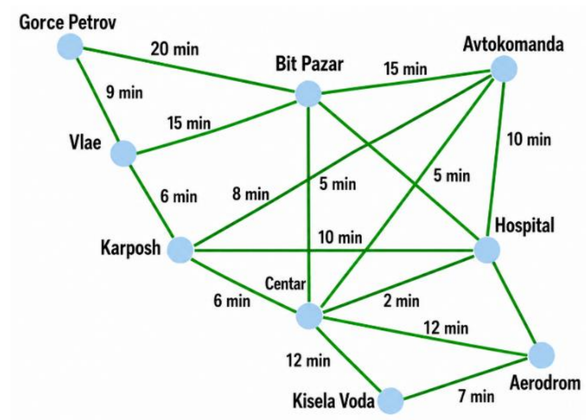


Fig. 1. Smart City transportation network inspired by the urban road infrastructure of Skopje.

Fig. 1 presents the Smart City transportation network used throughout the experimental evaluation.

### *C. Framework Architecture and Implementation*

The proposed framework consists of four major modules:

- Graph Representation Module;
- Routing Algorithm Module;
- Visualization Module;
- Experimental Analysis Module.

The Smart City simulation environment was implemented using Java and JavaFX, while Python was employed for supplementary algorithmic validation and scalability assessment.

The graph representation module stores nodes, edge weights, predecessor information, and route availability status. The routing module integrates Dijkstra, Bellman-Ford, and A* algorithms, whereas JavaFX enables visualization of transportation networks, optimal routes, and blocked road segments. Bellman-Ford was included in the scenario-based evaluation as a classical reference algorithm widely used in shortest-path research. Although the investigated transportation network contains only non-negative edge weights, Bellman-Ford provides a useful baseline for comparing computational behaviour and routing consistency with Dijkstra and A*. It was intentionally excluded from the scalability experiments because its higher computational complexity would have resulted in substantially longer execution times for repeated evaluations on large randomly generated graphs without providing additional methodological insight under the considered experimental conditions. Fig. 2 illustrates the interaction among the main modules of the proposed Smart City emergency transportation framework.

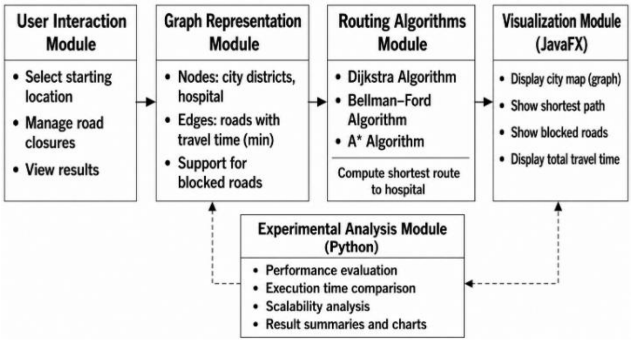


Fig. 2. Architecture of the proposed Smart City emergency transportation framework.

*1) Research hypotheses:* To guide the experimental evaluation, the study investigates the following research hypotheses:

H1. The A* algorithm achieves lower computational cost than Dijkstra's algorithm while preserving optimal route quality in transportation networks with non-negative edge weights.

H2. The proposed Smart City emergency routing framework is capable of correctly identifying feasible shortest routes as well as detecting situations in which no valid route exists due to road disruptions.

H3. Combining scalability experiments with representative Smart City emergency scenarios provides a comprehensive evaluation of both computational performance and practical applicability of classical shortest-path algorithms.

### *D. Dynamic Road Availability and Emergency Routing Scenarios*

Emergency transportation systems frequently operate under dynamically changing conditions. Traffic incidents, road maintenance, construction activities, or unexpected events may temporarily block important transportation links. To simulate such situations, selected road segments can be excluded from the transportation network before executing shortest path algorithms, enabling the generation of alternative routes whenever the initially optimal path becomes unavailable.

The term dynamic in this study refers to the evaluation of multiple transportation scenarios with different road availability configurations rather than continuous real-time updates of edge weights during route computation. Each scenario represents a static snapshot of the transportation network under a particular emergency condition.

Four representative emergency scenarios were considered to investigate algorithm behavior under varying transportation conditions.

*1) Scenario 1:* The ambulance vehicle is initially positioned in the Aerodrom district and dispatched toward the hospital under normal traffic conditions.

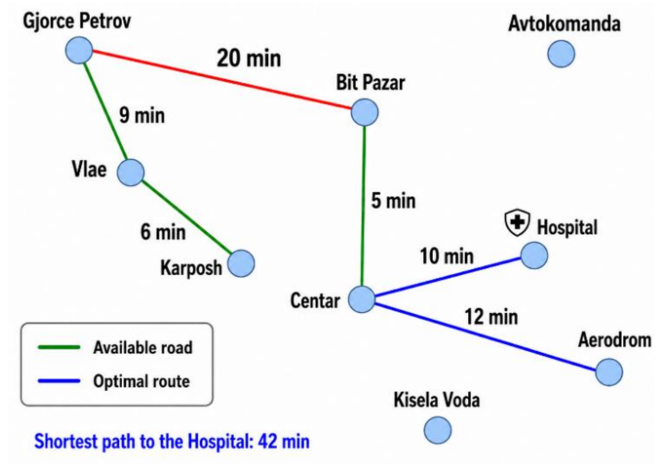


Fig. 3. Ambulance dispatching from Aerodrom district under normal traffic conditions.

*2) Scenario 2:* The ambulance vehicle starts from Vlae while maintaining full road accessibility.

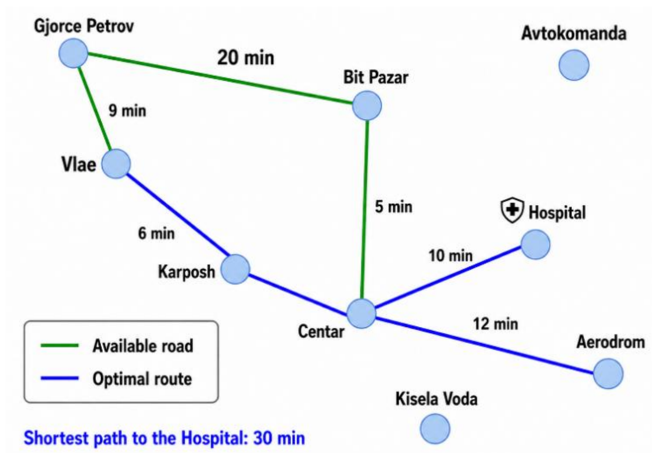


Fig. 4. Ambulance dispatching from Vlae district.

*3) Scenario 3:* Road segments leading to the hospital are intentionally blocked, resulting in the absence of a feasible transportation path.

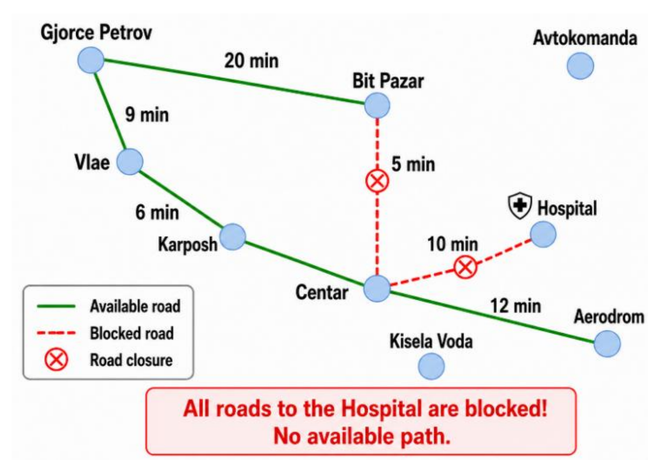


Fig. 5. Road closure scenario with unavailable routes to the hospital.

*4) Scenario 4:* The ambulance vehicle departs from Gjorce Petrov under regular transportation conditions.

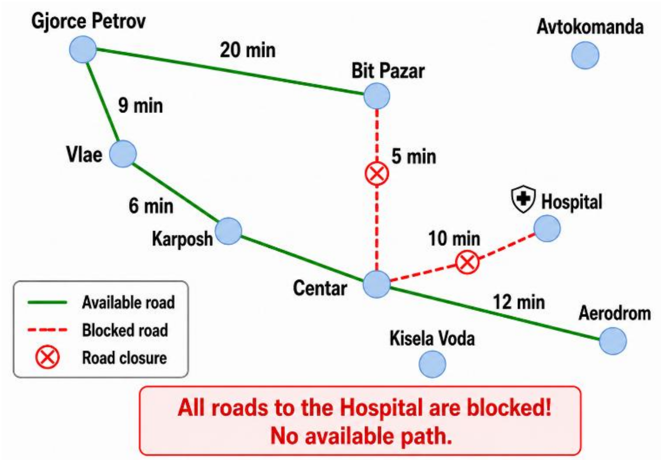


Fig. 6. Ambulance dispatching from Gjorce Petrov district.

The developed framework provides a flexible environment for investigating shortest path algorithms under realistic emergency transportation scenarios and serves as the basis for the experimental evaluation presented in the following sections.

## V. Experimental Evaluation and Results

### *A. Scalability Assessment of Shortest Path Algorithms*

To complement the Smart City emergency transportation scenarios, an additional set of experiments was conducted to evaluate the scalability and computational efficiency of shortest path algorithms under increasing graph sizes. These experiments were designed to assess the potential applicability of the investigated algorithms in larger Smart City transportation networks, where the number of intersections and road segments may substantially exceed those considered in the Skopje-inspired case study.

Dijkstra's and A* algorithms were implemented in Python 3.11 using a binary min-heap priority queue. Random weighted graphs containing 50, 100, 200, and 500 nodes were generated programmatically to simulate transportation networks of varying complexity. Edges were generated by considering all possible node pairs, where an edge was created with a probability (density) of 0.3. Edge weights were assigned as random integer values between 1 and 10, representing abstract transportation costs. The generated graphs were stored using adjacency-list representations to support efficient shortest-path computations. Priority queue operations in both Dijkstra's and A* implementations were performed using Python's built-in heapq module, which provides efficient insertion and extraction of the minimum-cost node. Since the investigated graph sizes were relatively moderate, no specialized memory optimization techniques were required. For each graph size, both algorithms were executed once on the generated weighted graph, and the execution time together with the number of visited nodes was recorded. The experiments were performed on a workstation equipped with an Intel i5 processor and 16 GB RAM.

Bellman-Ford was intentionally excluded from the scalability assessment because its computational complexity O(VE) makes its evaluation on large randomly generated graphs considerably more computationally demanding than Dijkstra and A*. Since the generated graphs contain only non-negative edge weights, including Bellman-Ford would not provide additional methodological insight beyond the scenario-based comparison presented in the Smart City case study.

A simple admissible heuristic based on the absolute difference between the current node identifier and the target node identifier, $h(n) = | \text{goal} - n |$, was adopted for the A* algorithm. Since the randomly generated scalability graphs do not contain spatial coordinates, conventional Euclidean or Manhattan distance heuristics were not applicable. Instead, the selected heuristic provides a simple and deterministic estimate that enables a consistent comparison between A* and Dijkstra under identical experimental conditions. Although relatively simple, this heuristic was sufficient to reduce the search space and improve computational performance.

Fig. 7 illustrates the execution time trends observed for Dijkstra's and A* algorithms as the graph size increases from 50 to 500 nodes. The presented values correspond to the recorded execution times for each generated graph instance. The obtained results indicate that the heuristic guidance employed by A* contributes to lower execution times and fewer visited nodes, particularly in larger graph instances.

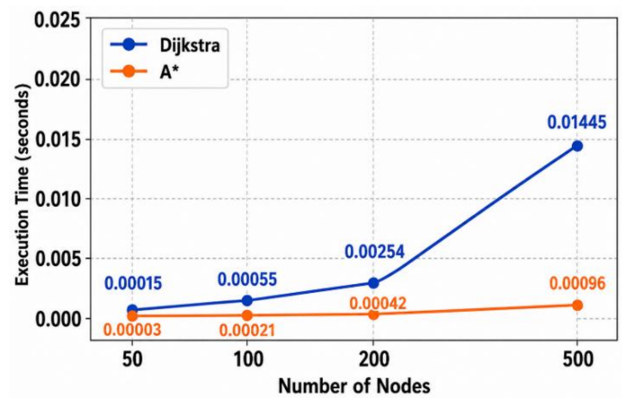


Fig. 7. Execution time comparison between Dijkstra and A* algorithms under increasing graph sizes.

Table I summarizes the obtained performance results.

TABLE I. Algorithm Performance Under Increasing Graph Sizes

| Nodes | Algorithm | Time(s) | Nodes Visited |
|---|---|---|---|
| 50 | Dijkstra | 0.00015 | 89 |
| 50 | A* | 0.00003 | 3 |
| 100 | Dijkstra | 0.00055 | 207 |
| 100 | A* | 0.00021 | 11 |
| 200 | Dijkstra | 0.00254 | 446 |
| 200 | A* | 0.00042 | 5 |
| 500 | Dijkstra | 0.01445 | 1233 |
| 500 | A* | 0.00096 | 7 |

The obtained results demonstrate that both algorithms successfully identified shortest paths for all tested graph instances. However, the increase in graph size had a considerably stronger impact on Dijkstra's algorithm, which explored a significantly larger portion of the graph. In contrast, A* consistently visited fewer nodes and maintained lower execution times due to heuristic guidance. The performance differences become particularly noticeable for graphs containing 200 and 500 nodes, suggesting that heuristic-based approaches can provide substantial advantages when shortest path computations are performed in large-scale transportation networks.

### B. Smart City Emergency Routing Scenarios

Following the scalability assessment, the algorithms were evaluated within the proposed Smart City emergency transportation framework.

Four representative ambulance dispatching scenarios were investigated under different traffic conditions and road availability assumptions.

Scenario 1 considers an ambulance vehicle departing from the Aerodrom district under normal traffic conditions.

Scenario 2 examines emergency transportation originating from Vlae while maintaining complete road accessibility.

Scenario 3 simulates a road closure situation in which all transportation links leading to the hospital become unavailable.

Scenario 4 analyzes ambulance dispatching from Gjorce Petrov under regular transportation conditions.

Fig. 3–6 present visual representations of the considered emergency scenarios.

TABLE II. RESULTS OBTAINED FOR EMERGENCY ROUTING SCENARIOS

| *Scenario* | *Starting District* | *Travel Time (min)* | *Path Available* |
|---|---|---|---|
| Scenario 1 | Aerodrom | 42 | Yes |
| Scenario 2 | Vlae | 30 | Yes |
| Scenario 3 | Road closure | – | No |
| Scenario 4 | Gjorce Petrov | 34 | Yes |

Table II summarizes the routing outcomes obtained for the investigated emergency transportation scenarios. The results indicate that the proposed routing framework successfully identified the shortest available routes whenever feasible transportation paths existed. In the road closure scenario, the framework correctly identified the absence of a valid path to the hospital, demonstrating its capability to handle dynamically changing transportation conditions.

Each Smart City emergency routing scenario was executed once because the transportation network topology, edge weights, and algorithm implementations are deterministic. Consequently, identical input conditions always produce identical routes and travel times. Therefore, variance measures were not reported, as no stochastic variability is present in the adopted experimental setup.

The conducted experiments provide complementary insights into the applicability of shortest path algorithms in intelligent transportation systems. The scalability assessment evaluates computational performance under progressively larger graph instances, whereas the Smart City case study demonstrates the practical applicability of the proposed routing framework under representative emergency transportation scenarios. Since the Smart City evaluation is deterministic and scenario-based, it is intended to illustrate routing behaviour rather than to provide statistical performance benchmarking.

The scalability assessment highlights the computational benefits of heuristic-guided search, particularly when dealing with larger transportation networks. At the same time, the Smart City simulations demonstrate the practical applicability of these algorithms in realistic emergency response scenarios involving ambulance dispatching, road disruptions, and dynamic traffic conditions.

Taken together, the obtained results suggest that combining classical shortest path algorithms with Smart City transportation models offers a flexible environment for studying emergency routing strategies and evaluating intelligent transportation solutions under various operating conditions.

## VI. DISCUSSION

The results obtained from both the scalability assessment and the Smart City case study provide useful insights into the applicability of shortest path algorithms in emergency transportation systems. By combining algorithm-oriented experiments with scenario-based simulations, the proposed framework enables an evaluation of both computational performance and practical suitability for time-critical routing tasks.

The scalability experiments revealed that the computational burden of shortest path algorithms increases with graph size. Although Dijkstra's algorithm consistently identified optimal routes, its execution time and the number of visited nodes increased substantially for larger graph instances. In contrast, the heuristic guidance incorporated into the A* algorithm limited unnecessary exploration of the search space and resulted in lower execution times. These differences became particularly evident for graphs containing 200 and 500 nodes, indicating that heuristic-based approaches may offer significant benefits when shortest path computations are repeatedly performed in large transportation networks.

For relatively small transportation networks similar to the proposed Skopje-inspired model, all investigated algorithms produced comparable routing decisions and identified identical optimal routes. Therefore, the primary differences among the algorithms are reflected in computational efficiency and scalability rather than in route optimality within moderately sized urban networks.

The Smart City emergency scenarios complement the scalability analysis by illustrating how shortest path algorithms behave under conditions that more closely resemble real-world transportation environments. The investigated scenarios demonstrated that feasible ambulance routes can be efficiently identified when transportation links remain available, while the framework was also capable of recognizing situations in which no valid route to the hospital exists due to road disruptions.

Such functionality is especially important in emergency response systems, where traffic incidents, construction activities, or unexpected events may rapidly alter transportation conditions.

An additional advantage of the proposed framework is its adaptability. The Skopje-inspired transportation network employed in this study was intentionally designed as a proof-of-concept case study rather than a complete representation of an urban transportation infrastructure. Its primary objective was to demonstrate the applicability of the proposed routing framework under representative emergency scenarios while maintaining a clear interpretation of algorithmic behaviour. Although the simplified network limits the direct generalization of the obtained results to large metropolitan transportation systems, the proposed methodology is independent of network size and can be readily extended to larger urban environments, multiple healthcare facilities, and transportation systems supported by continuously updated traffic information. Furthermore, more sophisticated heuristic functions, traffic prediction models, machine learning techniques, and real GIS or OpenStreetMap data may be incorporated to support adaptive routing decisions and improve overall system responsiveness.

Despite the encouraging results, several limitations should be acknowledged. The scalability assessment was intentionally limited to execution time and the number of visited nodes, as these metrics provide a direct comparison of computational efficiency across the investigated algorithms. Additional performance indicators, including memory consumption, edge relaxations, priority queue operations, and scalability under varying graph densities, represent promising directions for future research. The transportation network employed in this study represents a simplified proof-of-concept model of an urban environment and therefore does not capture the full topological complexity of real metropolitan transportation systems. Furthermore, the heuristic function adopted for the A* implementation was intentionally selected to remain simple to facilitate a transparent comparison with the investigated classical algorithms. Future studies may therefore focus on integrating richer transportation datasets, advanced heuristic strategies, and hybrid optimization methods capable of supporting more realistic emergency management scenarios. Although the proposed framework considers multiple emergency scenarios with different road availability conditions, route computation is performed on static network snapshots. Incorporating continuously changing traffic conditions, dynamic edge weights, and real-time rerouting mechanisms would further improve the realism and applicability of the proposed framework. In addition, the framework could be extended by incorporating more advanced routing techniques, including ALT (A*, Landmarks, Triangle inequality), Contraction Hierarchies, Bidirectional Dijkstra, and time-dependent shortest-path algorithms to further improve routing efficiency in large-scale transportation networks. Finally, comparisons with established routing platforms, such as OpenStreetMap-based routing engines and GraphHopper, would provide additional practical validation and facilitate benchmarking under real-world transportation conditions.

Nevertheless, the obtained conclusions should be interpreted with caution due to the relatively small size of the investigated transportation network and the simplified assumptions adopted in the simulation model.

Overall, the findings suggest that combining classical shortest path algorithms with Smart City transportation models provides a flexible and informative framework for investigating emergency routing strategies and evaluating intelligent transportation solutions under a wide range of operating conditions.

## VII. CONCLUSION

This study presented a comparative investigation of three well-established shortest path algorithms, namely Dijkstra, Bellman-Ford, and A*, within the context of Smart City emergency transportation systems. The proposed framework combined scalability experiments with scenario-based simulations inspired by the urban road network of Skopje, allowing both computational characteristics and practical routing capabilities to be evaluated.

The scalability assessment demonstrated that the performance differences among the investigated algorithms become more pronounced as graph size increases. While Dijkstra's algorithm consistently produced optimal solutions, the heuristic guidance employed by A* reduced the number of explored nodes and resulted in shorter execution times for larger transportation networks. Bellman-Ford preserved its ability to handle graphs containing negative edge weights, although this flexibility was accompanied by increased computational requirements.

The simulated emergency transportation scenarios further illustrated the applicability of shortest path algorithms in ambulance dispatching and route planning tasks. The developed framework successfully identified feasible routes under normal operating conditions and detected situations in which transportation links became unavailable due to road closures. In addition, the visualization environment facilitated an intuitive interpretation of routing decisions and transportation network states, which may be useful for both educational purposes and preliminary decision-support analyses.

Although the presented transportation model represents a simplified approximation of an urban environment, the obtained results indicate that graph-based routing methods remain effective tools for supporting emergency transportation planning in Smart City settings. Future work may focus on incorporating real-time traffic information, stochastic travel times, multiple healthcare facilities, and adaptive optimization techniques, including machine learning approaches, to provide more realistic and responsive emergency management solutions.

Overall, the presented framework demonstrates that graph-based routing approaches can provide valuable support for emergency transportation planning and may serve as a basis for future intelligent transportation applications.

## Acknowledgment

The authors sincerely appreciate the financial assistance received from Goce Delcev University, Stip, through its Individual Scientific Fund. This support played a crucial role in the successful completion of this research.

## References

[1] E. F. Z. Santana, A. P. Chaves, M. A. Gerosa, F. Kon, and D. S. Milojicic, "Software Platforms for Smart Cities: Concepts, Requirements, Challenges, and a Unified Reference Architecture," ACM Computing Surveys, vol. 50, no. 6, Art. no. 78, 2017.

[2] S. E. Bibri and J. Krogstie, "Smart Sustainable Cities of the Future: An Extensive Interdisciplinary Literature Review," Sustainable Cities and Society, vol. 31, pp. 183–212, 2017.

[3] V. Albino, U. Berardi, and R. M. Dangelico, "Smart Cities: Definitions, Dimensions, Performance, and Initiatives," Journal of Urban Technology, vol. 22, no. 1, pp. 3–21, 2015.

[4] A. Madkour, W. G. Aref, F. Ur Rehman, M. A. Rahman, and S. Basalamah, "A Survey of Shortest-Path Algorithms," arXiv preprint arXiv:1705.02044, 2017.

[5] L. Di Puglia Pugliese, D. Ferone, P. Festa, and F. Guerriero, "Shortest Path Tour Problems," in Encyclopedia of Optimization, Cham, Switzerland: Springer International Publishing, 2022, pp. 1–4.

[6] R. Bellman, "On a Routing Problem," Quarterly of Applied Mathematics, vol. 16, no. 1, pp. 87–90, 1958.

[7] E. W. Dijkstra, "A Note on Two Problems in Connexion with Graphs," Numerische Mathematik, vol. 1, no. 1, pp. 269–271, 1959.

[8] J. Wang, Y. Liu, B. Tang, X. Jin, and Y. Xiao, "Improved Dijkstra Algorithm Based on a New Storage Method," in Proceedings of the International Conference on Neural Networks, Information and Communication Engineering (NNICE), vol. 11933, SPIE, 2021, pp. 54–58.

[9] Y. Huang, Q. Yi, and M. Shi, "An Improved Dijkstra Shortest Path Algorithm," in Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering (ICCSEE 2013), Atlantis Press, 2013, pp. 226–229.

[10] P. E. Hart, N. J. Nilsson, and B. Raphael, "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," IEEE Transactions on Systems Science and Cybernetics, vol. 4, no. 2, pp. 100–107, 1968.

[11] S. Koenig and M. Likhachev, "D* Lite," in Proceedings of the 18th National Conference on Artificial Intelligence (AAAI), Edmonton, Canada, 2002, pp. 476–483.

[12] M. R. Wayahdi, S. H. N. Ginting, and D. Syahputra, "Greedy, A-Star, and Dijkstra's Algorithms in Finding Shortest Path," International Journal of Advances in Data and Information Systems, vol. 2, no. 1, pp. 45–52, 2021.

[13] F. Duchoň, A. Babinec, M. Kajan, P. Beňo, M. Florek, T. Fico, and L. Jurišica, "Path Planning with Modified A Star Algorithm for a Mobile Robot," Procedia Engineering, vol. 96, pp. 59–69, 2014.

[14] S. Koenig and M. Likhachev, "Fast Replanning for Navigation in Unknown Terrain," IEEE Transactions on Robotics, vol. 21, no. 3, pp. 354–363, 2005.

[15] L. R. Ford Jr., Network Flow Theory, Paper P-923, RAND Corporation, Santa Monica, CA, USA, 1956.

[16] M. Batty, K. W. Axhausen, F. Giannotti, A. Pozdnoukhov, A. Bazzani, M. Wachowicz, G. Ouzounis, and Y. Portugali, "Smart Cities of the Future," *The European Physical Journal Special Topics*, vol. 214, no. 1, pp. 481–518, 2012.

[17] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for Smart Cities," *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 22–32, 2014.

[18] Y. Regragui and N. Moussa, "A Real-Time Path Planning for Reducing Vehicles Traveling Time in Cooperative-Intelligent Transportation Systems," *Simulation Modelling Practice and Theory*, vol. 123, Art. no. 102710, 2023.

[19] M. A. Hassan, R. Javed, F. Granelli, X. Gen, M. Rizwan, S. H. Ali, H. Junaid, and S. Ullah, "Intelligent Transportation Systems in Smart City: A Systematic Survey," in *Proc. 2023 International Conference on Robotics and Automation in Industry (ICRAI)*, Islamabad, Pakistan, 2023, pp. 1–9.

[20] R. J. Szczerba, P. Galkowski, I. S. Glickstein, and N. Ternullo, "Robust Algorithm for Real-Time Route Planning," IEEE Transactions on Aerospace and Electronic Systems, vol. 36, no. 3, pp. 869–878, 2000.

[21] C. J. Jagtenberg, S. Bhulai, and R. D. van der Mei, "An Efficient Heuristic for Real-Time Ambulance Redeployment," Operations Research for Health Care, vol. 4, pp. 27–35, 2015.

[22] V. Schmid, "Solving the Dynamic Ambulance Relocation and Dispatching Problem Using Approximate Dynamic Programming," European Journal of Operational Research, vol. 219, no. 3, pp. 611–621, 2012.

[23] M. Gendreau, G. Laporte, and F. Semet, "A Dynamic Model and Parallel Tabu Search Heuristic for Real-Time Ambulance Relocation," Parallel Computing, vol. 27, no. 12, pp. 1641–1653, 2001.

[24] P. Toth and D. Vigo, Eds., Vehicle Routing: Problems, Methods, and Applications. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2014.

[25] R. Aringhieri, M. E. Bruni, S. Khodaparasti, and J. T. van Essen, "Emergency Medical Services and Beyond: Addressing New Challenges Through a Wide Literature Review," Computers & Operations Research, vol. 78, pp. 349–368, 2017.

[26] D. Bandara, M. E. Mayorga, and L. A. McLay, "Optimal Dispatching Strategies for Emergency Vehicles to Increase Patient Survivability," International Journal of Operational Research, vol. 15, no. 2, pp. 195–214, 2012.

[27] M. S. Maxwell, M. Restrepo, S. G. Henderson, and H. Topaloglu, "Approximate Dynamic Programming for Ambulance Redeployment," INFORMS Journal on Computing, vol. 22, no. 2, pp. 266–281, 2010.

[28] A. Borshchev, The Big Book of Simulation Modeling: Multimethod Modeling with AnyLogic 6. AnyLogic North America, 2013.

[29] C. M. Macal and M. J. North, "Tutorial on Agent-Based Modeling and Simulation," in Proceedings of the Winter Simulation Conference, 2005.

[30] S. C. Brailsford, "Tutorial: Advances and Challenges in Healthcare Simulation Modeling," in Proceedings of the 2007 Winter Simulation Conference, 2007, pp. 1436–1448.

[31] N. Stojkovic, M. Kocaleva, A. Stojanova, I. Janeva, and B. Zlatanovska, "Visualization of Ford-Fulkerson Algorithm," Balkan Journal of Applied Mathematics and Informatics, vol. 2, no. 2, pp. 7–20, 2019.

[32] N. Stojkovikj, L. Koceva Lazarova, A. Stojanova, M. Miteva, B. Zlatanovska, and M. Kocaleva, "Application of Agent-Based Modelling in Learning Process," Informatica, vol. 48, no. 1, 2024.

[33] N. Stojkovikj, S. Fileva, A. Stojanova Ilievska, L. Koceva Lazarova, C. Martinovska Bande, and V. Kokalanov, "Shortest Path Optimization in Transportation Networks Using Dijkstra's Algorithm: A Case Study of the Macedonian Road Network," Balkan Journal of Applied Mathematics and Informatics, vol. 9, no. 1, 2026.

[34] M. M. Haque, "Systematic Review on the Impact of AI-Enhanced Traffic Simulation on US Urban Mobility and Safety," ASRC Procedia: Global Perspectives in Science and Scholarship, vol. 1, no. 1, pp. 833–861, 2025.

[35] R. R. Saxena, "Artificial Intelligence in Traffic Systems," arXiv preprint arXiv:2412.12046, 2024.

[36] K. Chidananda, K. R. Pradeep, K. P. Mayuri, and S. Mythri, "AI-Driven Smart Ambulance and Dynamic Traffic Control System for Optimized Emergency Response," in Proceedings of the 2025 Control Instrumentation System Conference (CISCON), 2025, pp. 1–10.