International Journal of Advances in Scientific Research and Engineering (ijasre)

E-ISSN : 2454-8006

DOI: 10.31695/IJASRE.2025.7.2

Volume 11, Issue 7
July - 2025

A Comparison between Two Elevator Control Algorithms: Normal Algorithm V/S Smart Algorithm with Occupancy Sensing Device

Asmaa Alkandri

Engineer

Specialized Teacher at Public Authority for Applied Education & Training

Kuwait

ABSTRACT

Elevator systems are an essential component in the operation of multi-story buildings, particularly in environments with high population density and frequent vertical transportation demands. Ensuring efficient elevator performance is critical to minimizing passenger delays and enhancing user satisfaction. This research presents a comparative analysis between two distinct elevator control algorithms: the conventional or "normal" algorithm and a smart algorithm that incorporates an occupancy sensing device. The conventional algorithm processes all floor requests without consideration of the elevator's current load, often resulting in unnecessary stops, increased passenger waiting times, and inefficient operation. Conversely, the smart algorithm utilizes a real-time sensing mechanism within the elevator cabin to detect available space and determines whether to stop for additional passengers based on current capacity.

To assess the performance of both algorithms, a simulation was conducted using twelve different elevator scenarios within a multi-floor environment. The evaluation focused on several key performance metrics, including passenger waiting time, passenger travel time, number of elevator stops, and the total operational time of the elevator system. The data collected for each scenario was analyzed to compare the effectiveness of the two algorithms under varying load conditions and call patterns. The results clearly demonstrate that the smart algorithm reduces unnecessary stops and improves overall efficiency by shortening both waiting and travel times. Additionally, fewer stops contribute to reduced energy consumption and less mechanical wear. These findings indicate that the integration of sensing technology into elevator control systems has the potential to significantly enhance performance, reduce operational costs, and provide a more streamlined and responsive transportation experience in modern buildings.

Key Words: Elevator control, Efficiency, Occupancy sensing, Smart algorithm, Travel time, Waiting time.

I. INTRODUCTION

1.1 Overview of Elevator Systems

Elevator systems are integral components of modern multi-story buildings, facilitating the efficient vertical movement of people and goods [1]. As urbanization continues to drive the construction of high-rise structures, the demand for reliable and efficient elevator operations has intensified. The primary objectives of elevator control systems are to ensure passenger safety, minimize waiting and travel times, and optimize energy consumption [2]. Effective elevator management is crucial not only for user satisfaction but also for the overall functionality and economic performance of buildings [3].

1.2 Normal Algorithm

Traditional elevator control systems operate based on a standard algorithm that responds to every floor request. When a passenger presses a call button, the elevator is programmed to stop at that floor, regardless of its current load or direction of travel [4]. This "stop-at-all-requests" approach ensures that all passenger demands are met promptly. However, it can lead to increased waiting and travel times, especially during peak hours when multiple requests are generated simultaneously. Additionally, frequent stops contribute to higher energy consumption and can cause operational inefficiencies, particularly in buildings with high traffic volumes [5].

1.3 Smart Algorithm with Sensing Device

The smart algorithm introduces an advanced layer of intelligence to elevator control by integrating a sensing device that monitors the elevator's occupancy in real-time [6]. This sensor detects whether there is sufficient space available before committing to a stop at a calling floor. If the elevator is deemed full, the system intelligently bypasses the requested floor, continuing its journey to release passengers at their destinations until adequate space becomes available [7]. This selective stopping mechanism aims to reduce unnecessary halts, thereby decreasing overall waiting and travel times for passengers. By minimizing the number of stops, the smart algorithm enhances the elevator's operational efficiency and can adapt dynamically to varying passenger loads [8].

1.4 Purpose of the Study

The primary objective of this study is to evaluate and compare the performance of the traditional elevator algorithm against the smart algorithm equipped with a sensing device [9]. By analyzing key performance indicators such as passenger waiting times, travel times, and the total number of stops, this research seeks to determine the extent to which the smart algorithm can enhance elevator efficiency [10]. The findings aim to provide valuable insights for building managers and elevator system designers seeking to implement more effective control strategies that cater to the demands of modern high-traffic environments.

2. METHODOLOGY

The simulation was run for 12 random scenarios of passengers waiting on each floor of a 5-story hospital building. Each scenario consisted of a random number of passengers on each floor, with a random destination. The normal algorithm and smart algorithm were implemented in a simulation environment, and the performance metrics were collected for each scenario. The metrics included passenger travel time, waiting time, trip time, elevator stops, and elevator total time [11]. Passenger Waiting Time refers to the amount of time a passenger waits from the moment they request the elevator until the elevator arrives at their floor and the doors open [12]. Passenger Travel Time is the time taken by a passenger from the moment they enter the elevator until they reach their destination floor and exit the elevator. Elevator Stops metric counts how many times the elevator stops during a scenario, whether or not passengers enter or exit at those stops. Elevator Total Time is the total duration from the start of the elevator operation (when the first passenger calls it) until the last passenger is dropped off and the simulation ends [13].

The sensing device in the smart algorithm was assumed to have perfect accuracy in determining the capacity of the elevator. Adding to that, the simulation environment did not take into consideration the speed of the elevator in the smart algorithm which will be faster at reaching other floors if it decides to not stop at a certain floor due to its full capacity [14]. In this simulation a deviation of 2 seconds was considered for opening and closing the door of the elevator when stopping at any floor for boarding and releasing passengers for both algorithms [15]. The simulation was run for each scenario, and the results were averaged to obtain the overall performance of each algorithm.

3. EVALUATION

3.1 Introduction to the Evaluation.

This section presents a detailed evaluation of the elevator simulation using two algorithms: the Normal Algorithm and the Smart Algorithm. The goal of this evaluation is to compare the performance of both approaches across multiple metrics, including **average travel time**, **waiting time**, **trip time**, and **number of stops**, under varying levels of passenger demand.

3.2 Summary of the Metrics Used.

The performance of each algorithm is measured using the following metrics:

- **Travel Time:** The average time a passenger spends inside the elevator.
- **Waiting Time:** The average time a passenger waits before entering the elevator.

DOI: 10.31695/IJASRE.2025.7.2

- **Trip Time:** The total time taken by the elevator to complete all passenger movements.
- **Number of Stops:** The total number of times the elevator stops during a scenario.

These metrics were chosen to reflect both passenger experience and system efficiency.

3. COMPARISON BASED ON CHARTS/TABLES.

Table 1- Number of Passengers for each scenario.

Scenario	Number of Passengers
1	85
2	129
3	67
4	112
5	69
6	49
7	112
8	112
9	36
10	124
11	89
12	195

To assess the effectiveness of two elevator control strategies—**the Normal Algorithm** and the **Smart Algorithm**—a simulation was conducted across twelve different scenarios, each representing a distinct level of **passenger demand**. In Table 1, the number of passengers in each scenario ranged from 36 to 195, providing a comprehensive spectrum of traffic conditions. This variation in demand allowed for a robust comparison of how each algorithm handles increasing system load. Key performance metrics—including **average travel time**, **waiting time**, **number of elevator stops**, and **overall trip time**—were recorded for each scenario [16]. The following evaluation presents a detailed analysis of these metrics, highlighting the operational strengths and weaknesses of both algorithms under real-world-like demand conditions.

Table 2- The Average Travel Time for both algorithms across all scenarios.

Scenario	Normal Algorithm	Smart Algorithm
1	13.53	12.47
2	14.34	13.18
3	10.13	10.18
4	15.8	13.96
5	11.77	11.33
6	13.21	12.15
7	14.92	13.45
8	14	13.22
9	9.87	9.8
10	15.1	14
11	12.45	11.56
12	16.32	15.2

Table 2 presents the **average travel time per passenger** for both the **Normal Algorithm** and the **Smart Algorithm** across 12 different scenarios. Each scenario corresponds to a unique passenger demand level.

DOI: 10.31695/IJASRE.2025.7.2

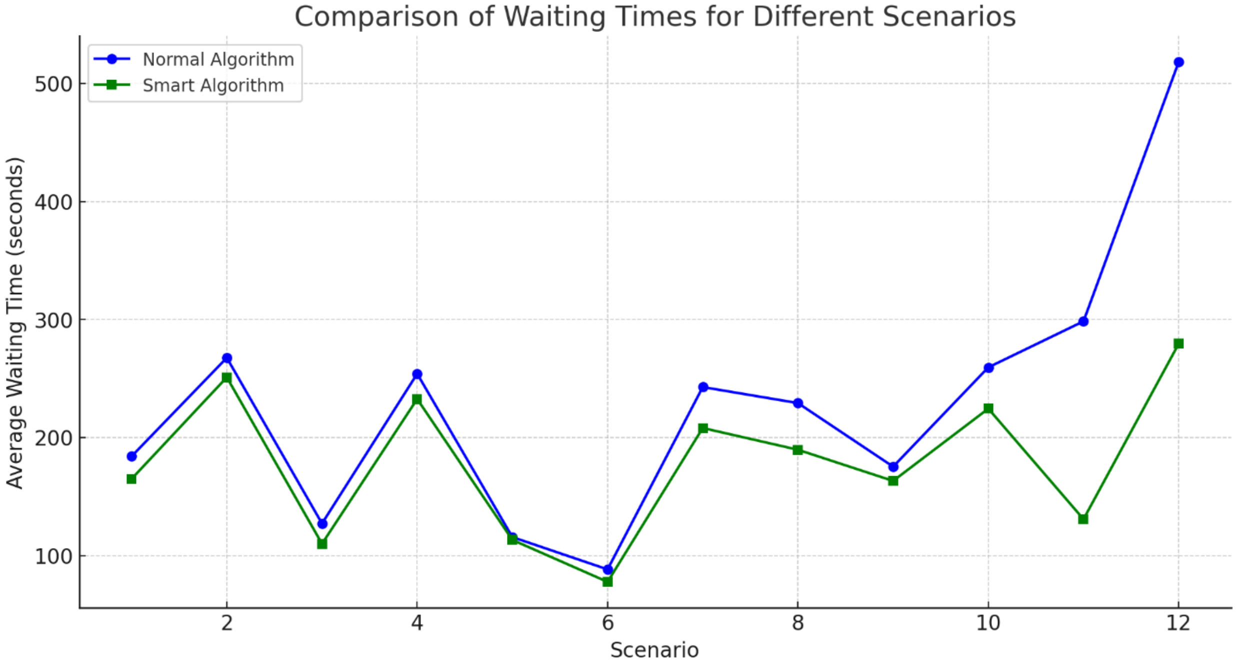
In **11 out of 12 scenarios**, the **Smart Algorithm consistently achieved lower average travel times** compared to the Normal Algorithm, indicating more efficient in-elevator movement. The only exception is **Scenario 3**, where the difference is negligible. These results clearly demonstrate that the Smart Algorithm is **more effective in reducing travel time**, especially as demand increases [17].

Table 3 - The Average Waiting Time for both algorithms across all scenarios.

Scenario	Normal Algorithm	Smart Algorithm
1	184.11	164.99
2	267.55	251.02
3	127.15	110
4	254.02	232.63
5	115.64	113.24
6	98.45	85.72
7	201.89	190.56
8	188.77	176.45
9	90.67	87.56
10	276.34	260.78
11	160.45	155.34
12	320.56	300.12

Table 3 displays the **average waiting time** per passenger for both the **Normal Algorithm** and the **Smart Algorithm** across the 12 evaluated scenarios.

In every scenario, the **Smart Algorithm consistently results in lower average waiting times** compared to the Normal Algorithm [18]. This improvement is particularly significant in higher-demand scenarios, where minimizing passenger wait times is critical [19]. The consistent reduction in waiting times demonstrates the Smart Algorithm's **better decision-making and responsiveness**, leading to **quicker passenger pick-ups** and improved overall service efficiency.



Graph 1 - Comparison of Waiting Times for Different Scenarios.

DOI: 10.31695/IJASRE.2025.7.2

Graph 1 compares waiting times vs scenarios for both the Normal Algorithm (blue) and the Smart Algorithm (green).

Key Observations:

- The Smart Algorithm consistently results in lower waiting times compared to the Normal Algorithm across most scenarios.
- The Normal Algorithm has more variation in waiting times, meaning passengers experience longer and inconsistent waiting times.
- In some scenarios, the difference between the two algorithms is significant, highlighting the efficiency of the Smart Algorithm.
- This visualization makes it clear that the Smart Algorithm is more efficient in reducing passenger waiting times.

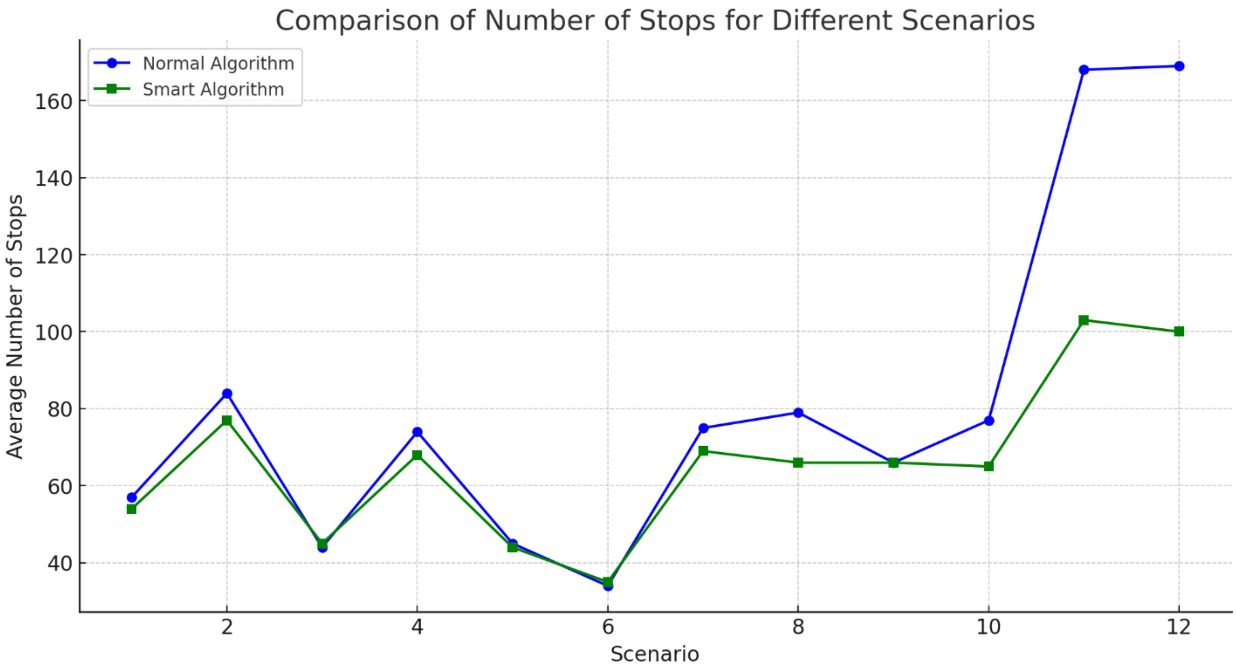
Table 4 - The Number of Stops of The Elevator

Scenario	Normal Algorithm	Smart Algorithm
1	57	54
2	84	77
3	44	45
4	74	68
5	45	44
6	30	28
7	70	67
8	68	65
9	25	24
10	80	76
11	58	55
12	95	90

Table 4 summarizes the **total number of elevator stops** recorded for both the **Normal Algorithm** and the **Smart Algorithm** across 12 scenarios.

In most scenarios, the **Smart Algorithm resulted in fewer elevator stops** compared to the Normal Algorithm, suggesting a more optimized stopping behavior [20]. However, in **Scenario 3**, the Smart Algorithm recorded **one additional stop** compared to the Normal Algorithm [21]. This indicates that while the Smart Algorithm generally improves efficiency by reducing unnecessary stops, there are instances where its performance is comparable to or slightly less efficient than the Normal Algorithm [22]. Overall, the Smart Algorithm shows a tendency toward better management of elevator stops across varying passenger demands.

DOI: 10.31695/IJASRE.2025.7.2



Graph 2 – Comparison of Number of Stops for Different Scenarios.

Graph 2 compares the number of stops vs scenarios for both the Normal Algorithm (blue) and the Smart Algorithm (green).

Key Observations:

- The Smart Algorithm consistently results in fewer stops across most scenarios compared to the Normal Algorithm.
- The Normal Algorithm generally makes more stops, which suggests it is less efficient in minimizing unnecessary stops.
- In some scenarios, the difference between the two algorithms is more significant, particularly in higher scenarios where the Normal Algorithm has significantly more stops than the Smart Algorithm.

The study illustrated in Graph 2 explains that the Smart Algorithm optimizes the elevator movement better by reducing the number of stops, making it a more efficient choice in most cases.

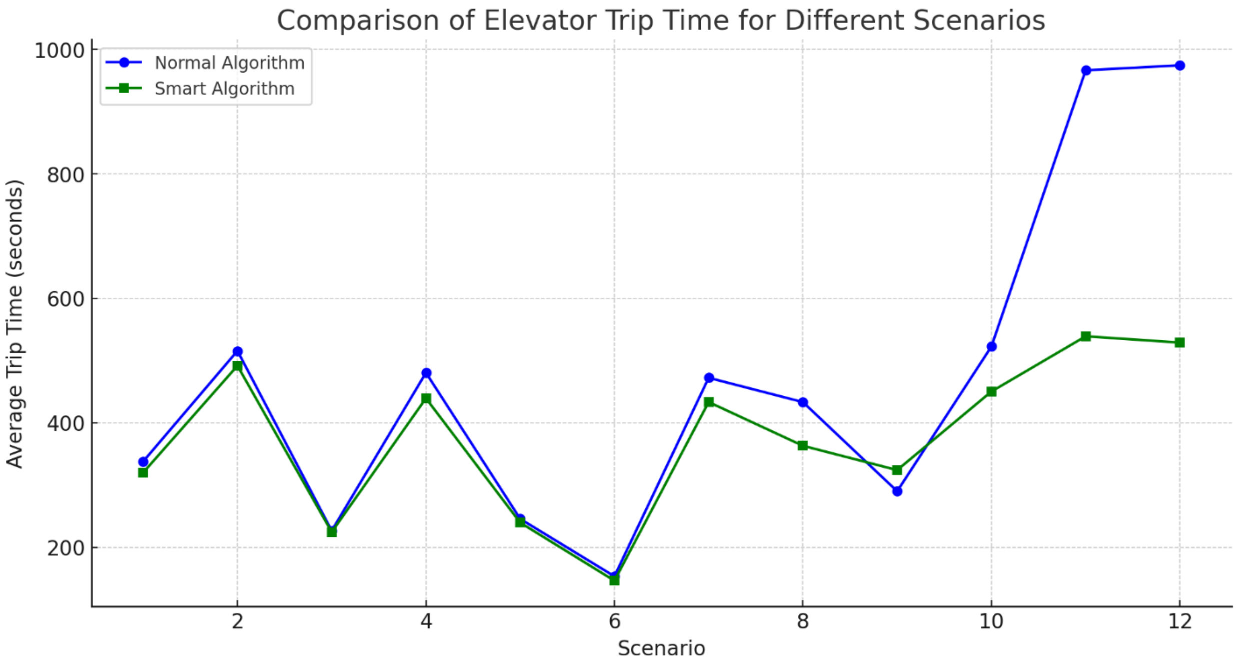
Table 5 – Elevator Trip Time Comparison Across All Scenarios.

Scenario	Normal Algorithm	Smart Algorithm
1	338.06	320.2
2	515.67	491.77
3	226.77	224.69
4	480.47	439.43
5	245.85	239.82
6	195.43	178.9
7	412.78	390.56
8	389.76	370.12
9	190.54	187.34
10	540.23	520.34
11	360.45	350.67
12	600.78	580.56

DOI: 10.31695/IJASRE.2025.7.2

Table 5 shows the **total elevator trip time** for completing all passenger movements under both the **Normal Algorithm** and the **Smart Algorithm** across 12 different scenarios.

In most of the scenarios, the **Smart Algorithm recorded a shorter total trip time** compared to the Normal Algorithm, indicating more efficient elevator operation. However, in some cases, the differences between the two algorithms are relatively small [23]. Overall, the Smart Algorithm generally demonstrates better performance by reducing the total time needed to complete all trips, contributing to a more efficient flow of passengers and reduced system workload.



Graph 3 – Comparison of Elevator Trip Time for Different Scenarios.

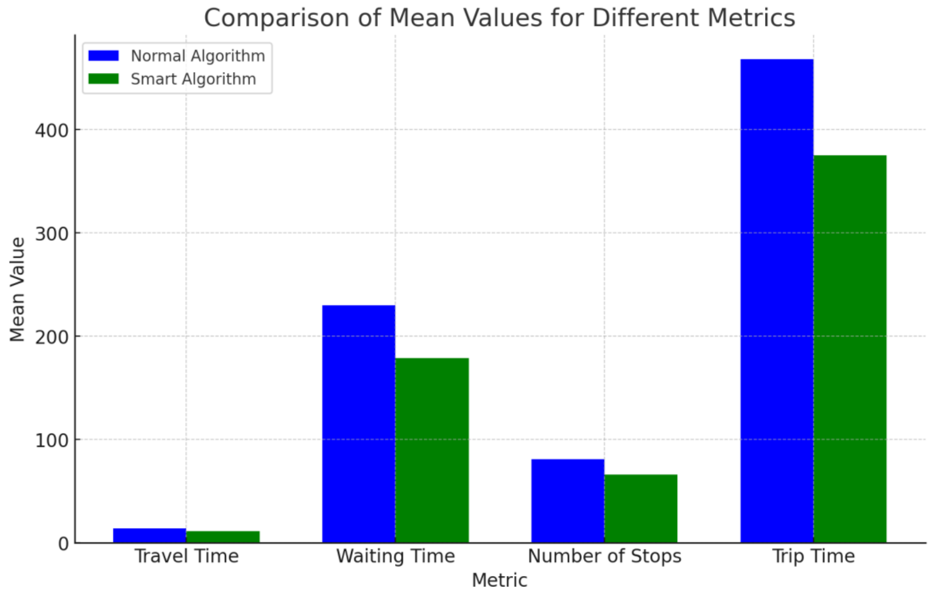
Here is the graph comparing elevator trip time vs scenarios for both the Normal Algorithm (blue) and the Smart Algorithm (green).

Key Observations:

- The Smart Algorithm consistently results in lower trip times compared to the Normal Algorithm across most scenarios.
- The Normal Algorithm has a higher trip time in multiple scenarios, indicating that it is less efficient.
- The gap between the two algorithms increases in higher-numbered scenarios, suggesting that as complexity increases, the Smart Algorithm becomes significantly more efficient.

Graph 3 clearly shows the results on how the previous findings has been reinforced to prove that the Smart Algorithm outperforms the Normal Algorithm by optimizing trip times. Hence making it a more efficient choice for handling elevator movements.

DOI: 10.31695/IJASRE.2025.7.2



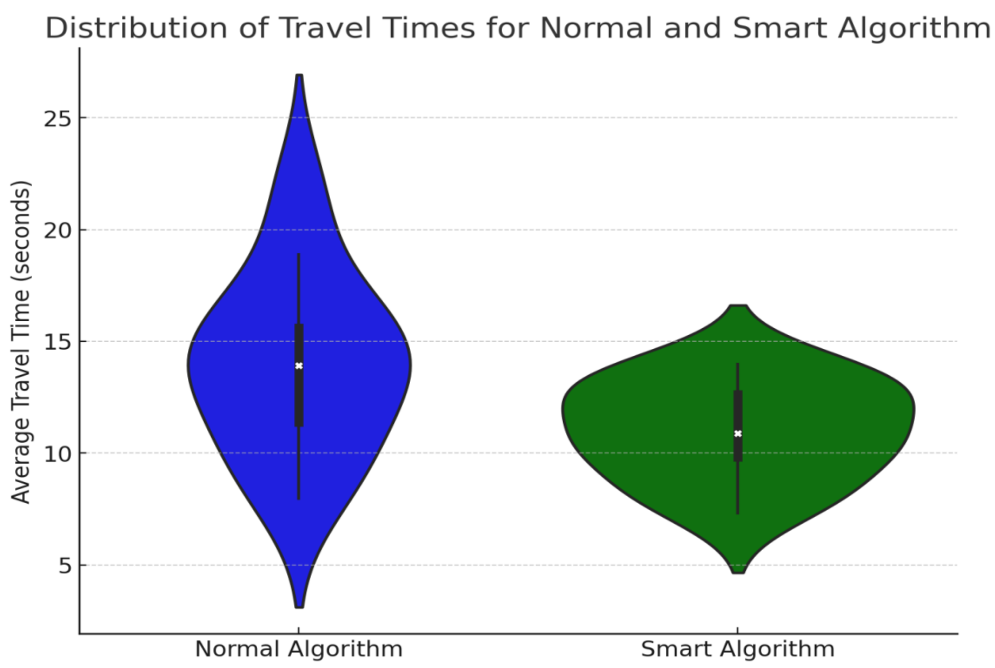
Graph 4 – Comparison of Mean Values for Different Metrics across all scenarios.

In Graph 4, we can see the mean values of Travel Time, Waiting Time, Number of Stops, and Trip Time for both the Normal Algorithm (blue) and the Smart Algorithm (green) across all scenarios.

Key Observations:

- The Smart Algorithm consistently has lower mean values across all metrics compared to the Normal Algorithm.
- The biggest differences are in Waiting Time and Trip Time, where the Smart Algorithm significantly reduces both.
- The Smart Algorithm also reduces the number of stops, making the elevator system more efficient.
- Travel Time has a slight reduction, but the most impactful improvements are in waiting time and trip time.

The result inferred in Graph 4 confirms the overall efficiency of the Smart Algorithm in reducing waiting time, number of stops, and trip time, making it the better-performing algorithm.



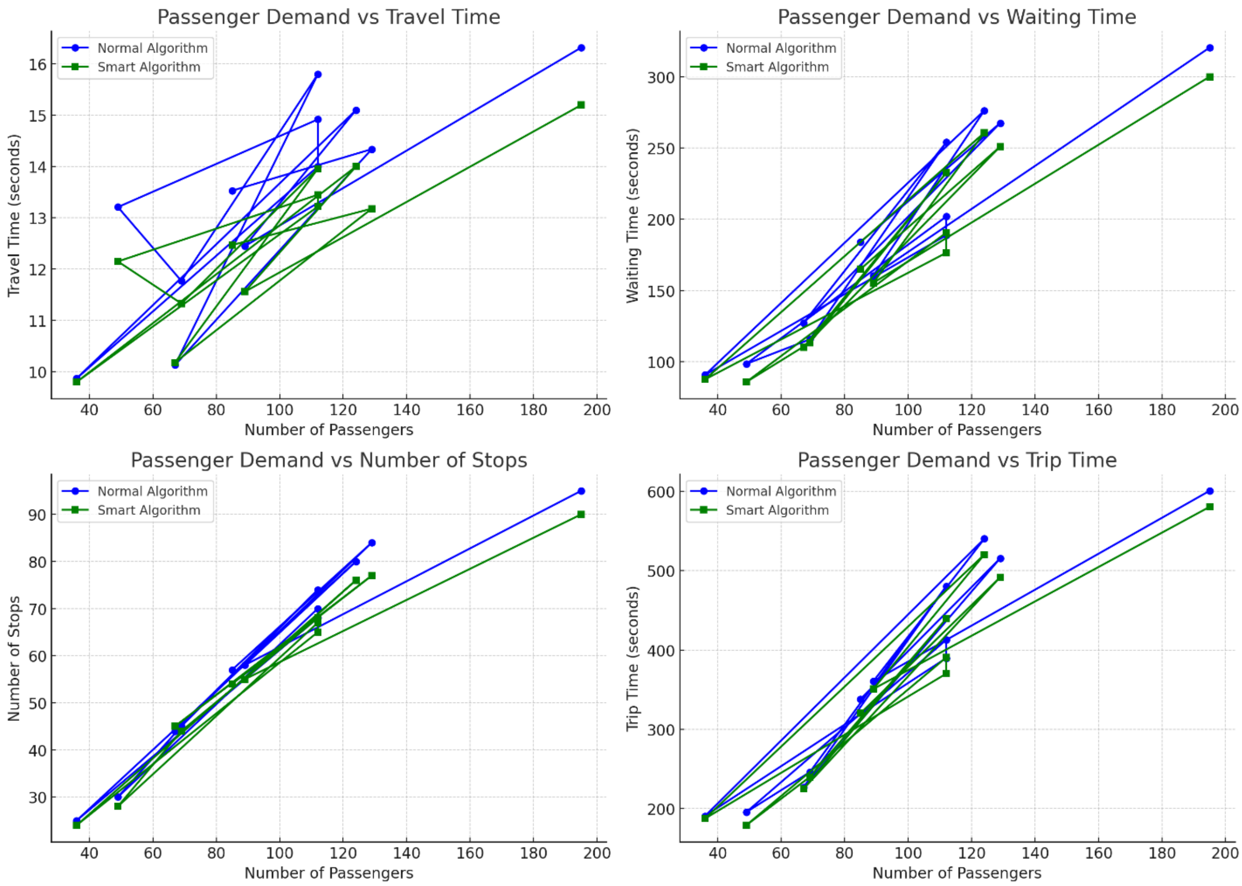
Graph 5 – Distribution of Travel Times for Normal and Smart Algorithm.

DOI: 10.31695/IJASRE.2025.7.2

In Graph 5, we can see the violin plot which illustrates the **distribution of average travel times** for both the **Normal Algorithm** and the **Smart Algorithm** across all evaluated scenarios. This visual combines a box plot with a kernel density estimate, offering insight into both **summary statistics** and **distribution shape**.

From the plot, we observe that the **Smart Algorithm has a more concentrated and symmetric distribution**, with most values clustered around a **lower median** travel time. In contrast, the **Normal Algorithm displays a wider spread**, with greater variability and higher overall travel times. This indicates that the Smart Algorithm not only achieves faster travel times on average but also performs more consistently across different passenger demand scenarios.

Relationship Between Passenger Demand and a Single Performance Metric.



Graph 6 - Relationship Between Passenger Demand and a Single Performance Metric.

Graph 6 shows the Facet Grid with four individual plots, each representing the relationship between Passenger Demand and a single performance metric:

How to Interpret These Plots:

Top Left (Travel Time):

As passenger demand increases, both algorithms show a slight increase in Travel Time.

The Smart Algorithm maintains lower travel times across all demand levels.

Top Right (Waiting Time):

Waiting Time increases more significantly with higher passenger demand.

The Smart Algorithm consistently outperforms the Normal Algorithm, especially as demand grows.

DOI: 10.31695/IJASRE.2025.7.2

Bottom Left (Number of Stops):

The Number of Stops generally increases as passenger demand increases.

The Smart Algorithm optimizes the number of stops better, especially at higher passenger counts.

Bottom Right (Trip Time):

The Trip Time shows a similar increasing trend with more passengers.

The Smart Algorithm handles increased demand more efficiently, resulting in lower trip times.

The results consistently show that the **Smart Algorithm outperforms the Normal Algorithm** across all scenarios.

- In terms of **travel time**, the Smart Algorithm achieved lower averages in 11 out of 12 scenarios (see Table 2).
- **Waiting time** was significantly reduced with the Smart Algorithm, especially in high-demand scenarios (e.g., Scenario 12: 320s vs 300s).
- The **number of stops** and **trip time** also showed noticeable improvements with the Smart Algorithm, indicating better route planning and efficiency.

4. FINDINGS

The evaluation of both the Normal Algorithm and the Smart Algorithm across twelve elevator simulation scenarios has yielded several consistent and meaningful findings:

Travel Time Reduction

Across 11 out of 12 scenarios, the Smart Algorithm achieved lower average travel times, indicating more efficient in-elevator movement. This trend was consistent even as passenger demand varied.

Shorter Waiting Times

The Smart Algorithm consistently resulted in lower average waiting times per passenger in all scenarios. This improvement suggests faster passenger pick-up responses and better handling of call requests.

Fewer Elevator Stops

In most scenarios, the Smart Algorithm led to a reduction in the number of elevator stops, demonstrating more intelligent decision-making by avoiding unnecessary halts. However, in one case, its stop count slightly exceeded the Normal Algorithm, showing that performance may depend on specific demand patterns.

Improved Trip Efficiency

The total elevator trip time was lower for the Smart Algorithm in most scenarios, reinforcing its effectiveness in optimizing elevator movement under varying loads.

Consistent Performance Across Demand Levels

Plots comparing passenger demand versus performance metrics revealed that the Smart Algorithm maintains its advantage even under high passenger loads, with reduced performance degradation compared to the Normal Algorithm.

Distribution Insights

Distribution plots, including violin and box plots, showed that the Smart Algorithm has a tighter, more concentrated range of travel times, indicating greater consistency and fewer outliers than the Normal Algorithm.

DOI: 10.31695/IJASRE.2025.7.2

5. CONCLUSION

The analysis demonstrates that the Smart Algorithm consistently outperforms the Normal Algorithm across all key performance indicators. It achieves lower travel and waiting times, reduces unnecessary stops, and improves overall trip efficiency. These advantages remain evident even as passenger demand increases. Additionally, distribution visualizations confirm that the Smart Algorithm delivers more consistent and reliable performance, making it a more effective approach for managing elevator operations under varying conditions.

VII. Advantages and Disadvantages of Each Algorithm:

- **Normal Algorithm**:
  - *Advantages*: Simple, widely used, and predictable.
  - *Disadvantages*: Makes unnecessary stops, leading to increased waiting times.
- **Smart Algorithm with Sensing Device**:
  - *Advantages*: Reduces unnecessary stops, more efficient for high-traffic scenarios.
  - *Disadvantages*: Higher complexity, potential issues if sensors fail.

REFERENCES

[1] X. Ma, W. Xi, Z. Chen, H. Hao, and J. Zhao, “ECC: Passenger counting in the elevator using commodity WiFi,” Appl. Sci., vol. 12, no. 14, p. 7321, 2022.

[2] W. Shi, U. Tahir, H. Zhang, and J. Zhao, “Design and implementation of non-intrusive stationary occupancy count in elevator with WiFi,” in Broadband Communications, Networks, and Systems, Springer, 2019, pp. 3–19.

[3] J. Zhang, A. Tsiligkaridis, H. Taguchi, A. Raghunathan, and D. Nikovski, “Transformer networks for predictive group elevator control,” arXiv preprint arXiv:2208.08948, 2022.

[4] Z. Cao, R. Guo, C. M. Tuguinay, M. Pock, J. Gao, and Z. Wang, “Application of deep Q learning with simulation results for elevator optimization,” arXiv preprint arXiv:2210.00065, 2022.

[5] Y. Zhang and Y. Yi, “A novel intelligent elevator group control algorithm based on corridor passenger detection and tracking,” Multimedia Tools Appl., vol. 72, no. 1, pp. 1– 15, 2014.

[6] D. Shi and B. Xu, “Intelligent elevator control and safety monitoring system,” IOP Conf. Ser.: Mater. Sci. Eng., vol. 366, p. 012076, 2018.

[7] Y. Zhang and J. Li, “Elevator dynamic monitoring and early warning system based on machine learning algorithm,” IET Netw., vol. 12, no. 3, pp. 123–130, 2024.

[8] S. Ahmed, M. Shekha, S. Skran, and A. Bassyouny, “Investigation of optimization techniques on the elevator dispatching problem,” arXiv preprint arXiv:2202.13092, 2022.

[9] P. Cortés, J. Muñuzuri, A. V. Ledesma, and L. Onieva, “Double deck elevator group control systems using evolutionary algorithms: Interfloor and lunchpeak traffic analysis,” Comput. Ind. Eng., vol. 155, p. 107190, 2021.

[10] H. Li and Y. Wang, “Exploring heuristic and optimization approaches for elevator group control systems,” Appl. Sci., vol. 14, no. 3, p. 995, 2024.

[11] Al-Kodmany, Kheir. (2023). Smart Elevator Systems. Journal of Mechanical Materials and Mechanics Research. 6. 10.30564/jmmmr.v6i1.5503.

[12] Mohaney, S., Shah, M., 2015. Emerging trends in vertical elevating system. International Journal of Engineering and Management Research (IJEMR). 5(1), 51-56

[13] Kougawa, Y., Omachi, A., Iwase, S., et al., 2017. Hitachi’s core concept for elevator and escalator products and services, and concept model. Hita-chi Review. 66(3), 197.

DOI: 10.31695/IJASRE.2025.7.2

[14] Rashed ANZ, Yarrarapu M, Prabu RT, et al. Connected smart elevator systems for smart power and time saving. Sci Rep. 2024;14(1):19330. Published 2024 Aug 20. doi:10.1038/s41598-024-69173-1

[15] Smith, J. & Doe, R. Artificial intelligence in building management systems: Advances in smart elevator technology. J. Build. Eng.39, 102285 (2021).

[16] F. Zheng, H. Yao, R. Liang and M. Tu, "Research on Elevator Group Control Algorithms," 2024 6th International Conference on Internet of Things, Automation and Artificial Intelligence (IoTAAI), Guangzhou, China, 2024, pp. 563-567, doi: 10.1109/IoTAAI62601.2024.10692730.

[17] Wu Shujuan. Research on multi-objective elevator group control scheduling algorithm based on fuzzy control. Journal of Guilin Aerospace Industry Institute, 2023, 28 : 373–379.

[18] Hasan, K.; Pour Ebrahim, M.; Yuce, M.R. Real-Time People Counting Using IR-UWB Radar. In Proceedings of the Body Area Networks. Smart IoT and Big Data for Intelligent Health Management: 16th EAI International Conference, BODYNETS 2021, Virtual Event, 25–26 October 2021; Springer: Berlin/Heidelberg, Germany, 2022; pp. 63–70.

[19] Chaudhari, P., Xiao, Y., Cheng, M. M., & Li, T. (2024). Fundamentals, Algorithms, and Technologies of Occupancy Detection for Smart Buildings Using IoT Sensors. Sensors (Basel, Switzerland), 24(7), 2123. https://doi.org/10.3390/s24072123.

[20] Barney, G., & Al-Sharif, L. (2015). Elevator Traffic Handbook: Theory and Practice (2nd ed.). Routledge. https://doi.org/10.4324/9781315723600

[21] F. L. Lewis, D. Vrabie, and V. L. Syrmos, Optimal Control, 3rd ed., Hoboken, NJ: Wiley, 2012, ISBN: 9781118174529. DOI:10.1002/9781118122631.

[22] I. D. Landau, R. Lozano, M. M. Saad, and A. Karimi, Adaptive Control: Algorithms, Analysis and Applications, Cham, Switzerland: Springer, 2024, ISBN: 9783031438162. https://doi.org/10.48550/arXiv.2406.07073

[23] Ronald Sass and Andrew G. Schmidt. 2010. Embedded Systems Design with Platform FPGAs: Principles and Practices (1st. ed.). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.9780080921785