Journal of Applied Data Sciences
Vol. 6, No. 2, May 2025, pp. 1037-1049

Using Evolutionary Optimization Techniques to Improve the Efficiency of Transportation Scheduling

Mohd Khaled Shambour[1,*]

[1]*Faculty of Engineering and Design, Department of Intelligent Systems Engineering, Middle East University, Amman, 11831, Jordan*

(Received: December 9, 2024; Revised: December 19, 2024; Accepted: January 19, 2025; Available online: March 4, 2025)

Abstract

This study addresses the challenge of enhancing transportation efficiency during large-scale events, with a particular focus on the Hajj pilgrimage. Every year, more than two million pilgrims visit Makkah in Saudi Arabia to perform their Hajj rituals. The Haj ritual requires transporting vast numbers of pilgrims within a limited time, compounded by diverse transportation preferences that make timely, optimal scheduling complex. To tackle this, the study employs three optimization algorithms -Harmony Search (HS), Differential Evolution (DE), and Black Widow Optimization (BWO) - to optimize transportation schedules based on individual preferences. A comprehensive mathematical model was developed for this purpose, incorporating both hard and soft constraints that reflect the scheduling requirements and preferences of pilgrims. Experimental results show that the DE algorithm consistently outperforms HS and BWO, achieving the highest mean scores in 100% of scenarios with a population size of 100, 66.7% of scenarios with a population size of 20, and 16.7% of scenarios with a population size of 5. In contrast, BWO struggles to adapt to varying parameter settings, producing consistently lower-quality solutions. DE, in particular, performs exceptionally well with lower crossover probabilities, demonstrating its ability to balance exploration and exploitation effectively. On the other hand, HS yields better results when higher exploration probabilities are used, highlighting its strength in broader search space exploration. In contrast, the performance of BWO remains largely unaffected by variations in exploration and exploitation parameters, leading to consistently inferior solutions. These findings underscore the importance of dynamic parameter tuning for large-scale optimization tasks, suggesting that such approaches are promising for addressing complex scheduling challenges in major events like Hajj.

Keywords: Transportation, Optimal Scheduling, Hajj, Optimization Algorithms, Large-Scale Events

1. Introduction

The need to satisfy human desires is constantly increasing in a rapidly changing environment, driven by technological advances and evolving preferences. Numerous industries are significantly impacted by this trend, including technology, healthcare, education, and commerce. It becomes essential to continuously innovate and manage resources efficiently in order to respond to these dynamic developments. Large-scale events like the Hajj pilgrimage present special difficulties for the planning of transportation and accommodation. Participants' entire experience is greatly enhanced when transportation programs are scheduled efficiently and resources are used optimally.

The Hajj pilgrimage in Saudi Arabia, which attracts a large number of people, requires precise planning and coordination. Among the most significant problems is the bottleneck at Muzdalifah, where pilgrims must stay a night or portion of it before continuing their pilgrimage activities. The limited capacity of Muzdalifah's housing sites limits the number of pilgrims who may be accommodated, hindering the scalability of the pilgrimage capacity in the future. As a result, there is an urgent need for new solutions that can optimize the scheduling process, boost accommodation capacity, and enhance the entire Hajj experience for all participants. Thus, the objective is to find efficient ways to optimize the scheduling procedures for the pilgrim transportation programs during the mega event using evolutionary algorithms such as the Harmony Search (HS) [1], the Black Widow Optimization (BWO) [2], and the Differential Evolution (DE) [3] algorithms. The HS algorithm draws inspiration from musicians' improvisational techniques, and emulates the process of finding the optimal answers by preserving balance and harmony among several program alternatives. It provides a flexible and adaptable scheduling technique that lets pilgrims select the best modes of

*Corresponding author: Mohd Khaled Shambour (m.shambour@meu.edu.jo)
DOI: https://doi.org/10.47738/jads.v5i2.581

transportation for their needs. On the other hand, the BWO algorithm draws inspiration from the hunting skills of black widow spiders, and improves the effectiveness of the proposed method in navigating intricate problem areas and identifying efficient solutions.

This study contributes significantly to improving transportation and major event management by using different approaches to address the unique challenges facing Hajj. The transportation scheduling process focuses on optimizing individual preferences, aiming to accommodate as many preferences as possible to enhance satisfaction and maximize resource utilization. This approach ensures efficient allocation of transportation resources, accommodating the diverse preferences of pilgrim transportation programs. The structure of the paper is outlined as follows. Section 2 provides a review of relevant literature. Section 3 provides the problem description. Section 4 introduces the proposed algorithm frameworks, followed by experimental results and discussion in Section 5. Lastly, Section 6 concludes the paper.

2. Related Work

In recent years, much research has focused on addressing the challenges of transportation management and improving participant experiences during large-scale events. Experts have explored various approaches and techniques to enhance efficiency, satisfaction, and overall participant experience. This section reviews the literature that has contributed to improving transportation and accommodation in large-scale events, with a particular focus on Hajj. Research in this area has explored a range of AI techniques such as optimization algorithms, simulation modeling, and scheduling strategies. Rehman and Felman [4] proposed an interactive approach to schedule groups of pilgrims performing the stoning rituals at the Jamarat building. This method allowed for personalized scheduling, which yielded favorable results when implemented during the Hajj season of 1440 AH. Felman et al. [5] also used camera data from inside the Grand Mosque in Mecca to simulate crowd movement around the Kaaba using the Mass-Motion program. The researchers developed a model that provided critical insights into crowd density and flow patterns, especially in key areas, which enhances crowd safety and better utilization of available resources. To improve the distribution of service points in the holy sites during the Hajj season, Morgan and Al-Khayat [6] used a genetic algorithm to achieve this. The authors proposed to apply the research idea to a set of applications that benefit pilgrims and facilitate the performance of their rituals, such as ambulances, police vehicles, and water bottle distribution stations. Their approach showed practical improvements in the accessibility of the proposed services, and the distribution of ambulances in Arafat was applied as a case study.

Al-Sabban and Ramadan [7] developed a simulation model for a shuttle bus system, recommending an optimal number of buses for the Nafra loop and reducing inter-transmission delays to improve travel times between Arafat, Muzdalifah, and Mina. Hussain et al. [8] also proposed a method to improve shuttle bus operations within the Hajj organization by determining the optimal number and cycles of buses for each office, to improve operational efficiency. Haase et al. [9] presented a scheduling model for the stoning ritual, which accommodates more than 2.3 million variables in less than ten minutes, with minimal deviation from the optimal solution. In another study, Yassin and Khan [10] built a simulation model for the shuttle bus system to the holy sites, which was used to evaluate current operations and recommend performance improvements.

Finally, Felemban et al. [11] studied the movement of the holy sites train for transporting pilgrims. A sophisticated system was applied to improve the scheduling of pilgrims using the train, taking into account spatial and temporal rituals, taking into account various factors such as train movement times, camping sites, road restrictions, and station capacity to prevent crowding and jostling.

Previous studies have studied various aspects of improving services provided to large numbers of people during a specific time period, such as transportation and accommodation at major events, especially during the Hajj season, which is considered one of the largest global gatherings. The proposed methodologies in various service dimensions contribute to enriching the knowledge database to support the decisions of officials and stakeholders, which would contribute to facilitating the management of the complexities of large-scale events and maintaining the security and safety of pilgrims. However, despite the substantial advancements made, there remains ample room for future progress. This research aims to fill a critical gap by providing innovative solutions to improve the movement of pilgrims between Hajj sites. The goal is to increase the efficiency of pilgrims' movement while aligning with their preferences regarding travel times and accommodation, thus enhancing their overall experience and contributing to the smooth running of the event.

3. Problem Description

3.1. The Designed Transportation Programs

The transportation process of a pilgrim group to Muzdalifah took 90 minutes, which serves as the standard unit of timeslot. The total duration for assigning all the pilgrims to the Muzdalifah area is divided into seven timeslots. Specifically, three timeslots are allocated before midnight, three timeslots are allocated after midnight, and one timeslot is allocated after dawn time.

The assignment of timeslots before midnight, after midnight, and after dawn shows the different forms of transportation scheduling activities spanning different periods of the night. Segmenting the entire transportation period into seven timeslots can indeed facilitate the efficient management and scheduling of pilgrim movement to Muzdalifah. This division allows for better coordination and ensures that the assignment process is carried out in an organized and systematic manner.

The division of timeslots has resulted in the identification of five distinct programs for Hajj pilgrims, contingent upon their arrival and departure timings at Muzdalifah. Program 1 encompasses pilgrims who arrive at Muzdalifah before midnight and depart before midnight. Program 2 accommodates those arriving before midnight but departing after midnight. Program 3 caters to pilgrims arriving before midnight and departing after dawn. Program 4 includes those who arrive at Muzdalifah after midnight and depart after midnight. Finally, Program 5 encompasses pilgrims arriving after midnight and departing after dawn.

These subprograms are designed to streamline the scheduling process of pilgrims according to their respective arrival and departure timeslots. From these five main programs, a total of 27 subprograms can be derived based on their distribution across the timeslots.

3.2. Problem Formulation

To ensure the proposed algorithm effectively meets its objective of offering diverse transportation programs to accommodate a large number of pilgrims, several key constraints were implemented to guide its operation. These constraints were designed to help the algorithm generate programs that provide a variety of options, allowing pilgrims to choose schedules that best align with their individual preferences. This approach fosters inclusivity and significantly enhances the overall satisfaction of pilgrims during the Hajj.

The program scheduling process is governed by two types of constraints: hard constraints, which must be strictly satisfied in the final solution, and soft constraints, which allow for some flexibility or minor violations. These constraints are mathematically represented using an assignment function (A), which takes as inputs the following resources: PG (set of pilgrim groups), T (set of available time slots), S (set of Muzdalifah sites), and P (set of pilgrimage programs). The specific constraints are defined as follows:

3.2.1. Hard Constraints:

These represent non-negotiable requirements (e.g., resource availability, task dependencies) that must always be satisfied. The optimization algorithm incorporates these constraints directly into its feasibility check, ensuring that any solution violating these constraints is discarded or penalized with a high cost. This guarantees the generation of feasible solutions. The hard constraints are as follows:

H1. Each group of pilgrims is transported once. This constraint guarantees that each pilgrim group is considered, thereby ensuring fairness and equal treatment throughout the scheduling process.

$$A_{PG}^{S} = A_{PG_j}^{S_i} \qquad \forall\ j \in PG;\ \ i \in S \tag{1}$$

 ISSN 2723-6471

H2. Each timeslot contains only one pilgrim group, avoiding any conflicts or overlaps. This constraint prevents overcrowding or mismanagement of resources within a given timeslot.

$$\mathrm{A}_{\mathrm{PG_j}}^{\mathrm{T_t,S_i}} \neq \mathrm{A}_{\mathrm{PG_k}}^{\mathrm{T_t,S_i}} \qquad \mathrm{t} \in \mathrm{T};\ \mathrm{j} \neq \mathrm{k}; \forall\, \mathrm{j}\,, \mathrm{k} \in \mathrm{PG}; \mathrm{i} \in \mathrm{S} \tag{2}$$

H3. The generated solutions must contain all main transportation programs, ensuring that the percentage of each program does not fall below a certain threshold. This constraint guarantees the representation and availability of all programs in the final scheduling solutions.

$$A_{PG}^{P} = A_{PG_j}^{P_i} \qquad \forall\ i \in P; j \in PG \tag{3}$$

such that $X\left(A_{PG}^{P_i}\right) \geq x_i \quad \forall\ i \in P$, where X indicates the assignment percentages of the main pilgrim groups. The xi represents the percentage of the main program i, such that $x_{i=1}$=0.02, $x_{i=2}$=0.03, $x_{i=3}$=0.05, $x_{i=4}$=0.01, and $x_{i=5}$=0.01.

3.2.2. Soft Constraints:

These represent preferences or desirable conditions (e.g., minimizing task delay, balancing workload) that are not mandatory but improve the quality of the solution if satisfied. The algorithm integrates these constraints into the objective function by assigning weights or penalties. This allows the algorithm to balance trade-offs between competing objectives while prioritizing critical factors. The soft constraints are as follows:

S1. Pilgrims should be distributed among the main programs based on the preferred percentage for each main program.

$Y\left(A_{PG}^{P_i}\right) \cong y_i \quad \forall\ i \in P$, where Y represents the assignments percentage function of pilgrim groups to a particular program, where yi represents the desired proportion of program i, such that $y_{i=1} = 0.2$, $y_{i=2} = 0.5$, $y_{i=3} = 0.2$, $y_{i=4} = 0.01$, and $y_{i=5} = 0.09$.

S2. All timeslots are occupied by pilgrim groups across all Muzdalifah sites, ensuring the comprehensive utilization of the entire area of Muzdalifah.

$$A_{PG_j}^{T_t,S_i} \neq \emptyset \qquad \forall t \in T; \forall i \in S; j \in PG \tag{4}$$

S3. Utilize each Muzdalifah site as much as possible by allocating the largest possible number of pilgrim groups.

$A_{PG}^{S_i} \cong Max(nPG) \quad \forall\ i \in S$, where nPG represents the count of pilgrim groups allocated to a site, denoted as i.

The minimization objective function serves as a metric to assess the quality of the generated solutions and provides guidance in determining the most suitable distribution for transporting pilgrim groups to Muzdalifah sites. This objective function assigns a numerical value to each solution, reflecting the efficiency of the final solution. Since the objective of this study is to achieve the optimal distribution of transportation programs while adhering to as many constraints as possible, the costs associated with violating these constraints are determined based on their significance in attaining the final solution. The objective function accounts for the costs associated with violating both hard and soft constraints is represented as follows:

$$Objective\ Function\ Cost = Cost\ (Hard\ Constraints\ Violations) + Cost\ (Soft\ Constraints\ Violations) \tag{5}$$

$$Cost\ (Hard\ Constraints\ Violations) = 1000 \times (Number\ of\ violated\ hard\ constraints) \tag{6}$$

$$Cost\ (Soft\ Constraints\ Violations) = Cost(S1\ Violations) + Cost(S2\ Violations) + Cost(S3\ Violations) \tag{7}$$

Where:

$$Cost(S1\ Violations) = 10 \times (abs(Actual\ distribution\ percentage\ for\ each\ main\ program - Preferred\ distribution\ percentage\ for\ each\ main\ program))$$

$$Cost(S2\ Violations) = 5 \times (Number\ of\ unoccupied\ timeslots)$$

$$Cost(S3\ Violations) = Number\ of\ timeslots\ without\ a\ new\ sub\ program\ assignment$$

Accordingly, the objective function aims to minimize the total cost, reflecting the goal of finding a solution that minimizes the violations of both hard and soft constraints. This is represented as follows:

$$Min \sum_{h=1}^{3} n_{v_{H_h}} \times W_{H_h} + \sum_{s=1}^{3} n_{v_{S_s}} \times W_{S_s} \tag{8}$$

Where $n_{v_{H_h}}$ represents the violation times for each of the hard constraints (H_1, H_2, H_3), $n_{v_{S_s}}$ denotes the violation times for each of the soft constraints (S_1, S_2, S_3), W_{H_h} and W_{S_s} indicate the violation cost values for the hard and soft constraints, respectively.

4. Proposed Algorithm Framework

This section provides a brief description of HS and BWO algorithms, outlining their operators and characteristics. Additionally, it provides a detailed description of the proposed approach employed in this study.

4.1. Harmony Search Algorithm

The Harmony Search (HS) algorithm is an evolutionary algorithm proposed by Geem et al.[1]. Its procedure involves preserving several solutions during the search process which are cooperated to create a new solution in every improvisation.

The HS algorithm primarily operates through a series of steps [1]: harmony initialization, where a random initial population of harmonies is generated; harmony memory consideration, where a new harmony is constructed either by selecting one from the harmony memory (HM) or through a random search; pitch adjustment, where the values of a new harmony are subject to adjustment; and finally, harmony memory update, where the contents of the HM are updated by incorporating the best harmony in terms of fitness value. The HS algorithm is characterized by its ease of implementation, its stochastic nature inspired by musical improvisation, a balance between exploration and exploitation, versatility in handling different problem types, and proven effectiveness in solving optimization problems.

4.2. Black Widow Optimization Algorithm

The BWO algorithm is an evolutionary optimization method proposed in [2]. The idea of this algorithm is inspired by the natural mating process of black widow spiders. Spiders collaborate by sharing information which facilitates the exchange of insights. This collective effort enables the spiders to generate new solutions for the next iteration of the search procedure.

The BWO algorithm primarily operates through several key steps [12]: population initialization, where an initial random population of potential solutions, often referred to as "spiders", is generated; procreation, where the selection of spiders for reproduction or survival in the subsequent generation is determined; cannibalism, where spiders with lower fitness values are eliminated, while those with higher fitness values persist; and finally, mutation, which introduces small stochastic changes to the genetic makeup of spiders, thereby encouraging diversity within the population. The BWO algorithm is characterized by its simplicity and ease of implementation, efficient navigation of the search space, satisfactory accuracy, and reduced computational complexity.

4.3. Differential Evolution Algorithm

The DE algorithm is an evolutionary optimization method introduced by Storn and Price [3]. It is a stochastic search method that operates by evolving a population of candidate solutions over successive generations. The DE algorithm primarily operates through a series of steps: population initialization, where a random set of candidate solutions is generated within the problem's search space; mutation, which promotes diversity and explores new regions of the search space; crossover, which combines the mutated vector with the current individual to create a trial vector; and finally, selection, where the fitness values of the resulting solution and the original solution are compared, and the solution with the better fitness value is retained. The DE algorithm exhibits several desirable characteristics, including ease of implementation, efficient exploration of the search space, adaptability to various problem types, scalability and flexibility, and a notable ability to avoid local optima.

5. Experimental Results and Discussion

This section presents the transportation simulation results for the three algorithms: HS, BWO, and DE. The performance of each algorithm is evaluated, and the effects of various parameters on outcomes are analyzed. The case study focuses on scheduling pilgrim programs across a consistent set of one hundred sites, which remained unchanged throughout all experiments. Each algorithm was assessed over 30 trials, each involving 1000 iterations.

5.1. Experimental Design

As shown in table 1, six scenarios were developed with different parameter settings for the three algorithms. For each scenario, thirty experiments were conducted with different population sizes of 5, 20, and 100. It is worth noting that the parameter setting for the BWO algorithm was set to 0.44, as recommended in [2]. These experimental settings were chosen to examine the effectiveness of the search, exploration, and exploitation of the HS, BWO, and DEA algorithms. A set of statistical measures was used to compare the performance of the algorithms in producing solutions for all scenarios, including the mean, standard deviation, best and worst fitness values. All experiments were implemented using MATLAB 2020b, which was run on a Windows 11 64-bit operating system with an Intel i7-2.3 GHz processor and 24 GB of RAM.

Table 1. Parameter settings of HS, BWO and DE algorithms

Algorithm		Sc.1	Sc.2	Sc.3	Sc.4	Sc.5	Sc.6
HS	HMCR	0.5	0.5	0.7	0.7	0.9	0.9
	PAR	0.3	0.5	0.3	0.5	0.3	0.5
BWO	Procreate rate	0.5	0.5	0.7	0.7	0.9	0.9
	Mutation rate	0.5	*	0.3	*	0.1	*
DE	Pc	0.5	0.5	0.7	0.7	0.9	0.9
	Pm	0.3	0.5	0.3	0.5	0.3	0.5

* The missing values for BWO in certain scenarios are due to the algorithm's parameter constraints.

5.2. Experimental Results and Discussion

5.2.1. Simulation Results with a 5- Population Size

The experiment results for a population size of 5, as shown in table 2, show that the HS algorithm performs better than the alternative algorithms as it achieves the highest mean scores in all the tested scenarios except scenario 1. The experiment results also show that although HS achieves better average values in most of the scenarios performed, the DE algorithm performs better by producing the best solution in all scenarios compared to both the HS and BWO algorithms. Moreover, BWO shows the lowest performance compared to the others in all the experiment scenarios.

Table 2. Statistical results of fitness values for 30 experimental tests with a population size of 5

		Sc1	Sc2	Sc3	Sc4	Sc5	Sc6
DE	Mean	2.19E+03	2.69E+03	2.51E+03	2.81E+03	2.51E+03	2.78E+03
	Std	4.70E+02	2.12E+02	5.00E+02	1.68E+02	5.07E+02	1.75E+02
	Best	1.68E+03	2.03E+03	1.65E+03	2.18E+03	1.78E+03	2.03E+03
	Worst	2.90E+03	2.84E+03	2.96E+03	2.92E+03	3.06E+03	2.96E+03
HS	Mean	2.41E+03	2.43E+03	2.38E+03	2.42E+03	2.35E+03	2.38E+03
	Std	1.68E+01	1.14E+01	1.50E+01	1.47E+01	4.33E+01	1.11E+02
	Best	2.38E+03	2.41E+03	2.35E+03	2.40E+03	2.29E+03	1.79E+03
	Worst	2.44E+03	2.45E+03	2.41E+03	2.45E+03	2.55E+03	2.43E+03

ISSN 2723-6471

BWO	Mean	2.65E+03	-	2.58E+03	-	2.71E+03	-
	Std	3.73E+01	-	4.19E+01	-	3.17E+01	-
	Best	2.59E+03	-	2.49E+03	-	2.65E+03	-
	Worst	2.72E+03	-	2.66E+03	-	2.76E+03	-

Figure 1, figure 2, figure 3, figure 4, figure 5, figure 6 show the best convergence behavior of the algorithms across different parameter settings, demonstrating their ability to minimize the cost function over search iterations. The DE algorithm consistently outperforms the others, achieving superior solution quality and faster convergence in most scenarios. The graphs also reveal the relatively stable performance of the HS algorithm and the slower convergence rate of the BWO algorithm, which tends to stabilize at higher cost values. These visual results confirm the efficiency and robustness of the DE algorithm in solving optimization problems under different experimental settings.

In terms of consistency, the DE and HS algorithms show substantial variability, with standard deviations ranging from 1.36E+02 to 5.00E+02 and from 1.14E+01 to 1.11E+02, respectively. The high standard deviations suggest that the algorithms' convergence to a solution is not very consistent at small population sizes, though DE consistently achieves the best fitness values in most cases. On the other hand, the BWO algorithm exhibits relatively lower variability, ranging from 3.17E+01 to 4.19E+01, indicating that it produces similar results across different runs.

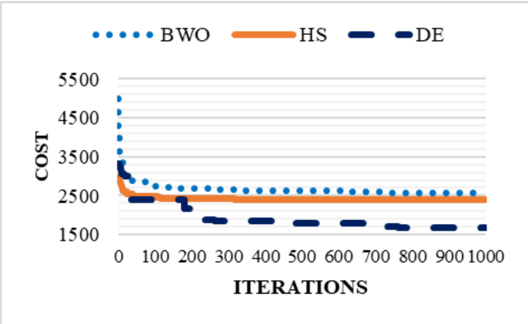

Figure 1. The comparison of convergence rates (Scenario 1 - population size of 5)

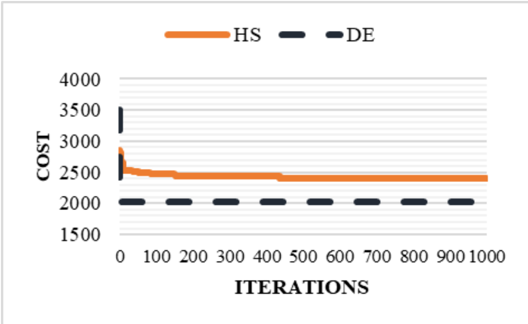

Figure 2. The comparison of convergence rates (Scenario 2 - population size of 5)

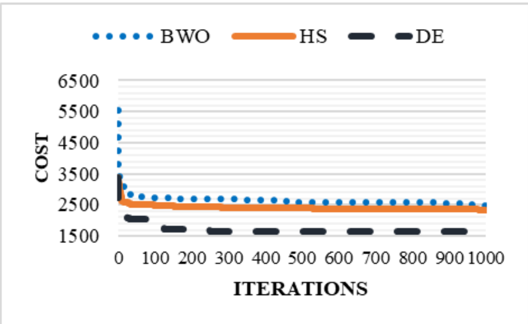

Figure 3. The comparison of convergence rates (Scenario 3 - population size of 5)

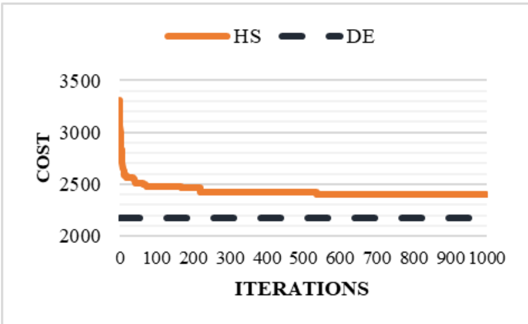

Figure 4. The comparison of convergence rates (Scenario 4 - population size of 5)

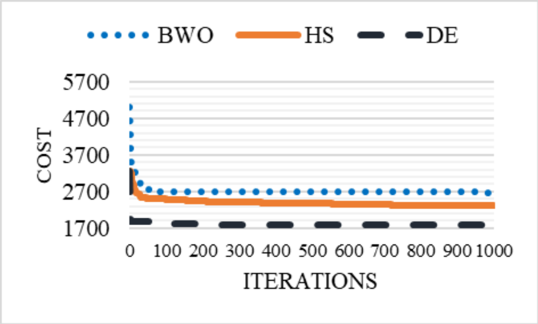

Figure 5. The comparison of convergence rates (Scenario 5 - population size of 5)

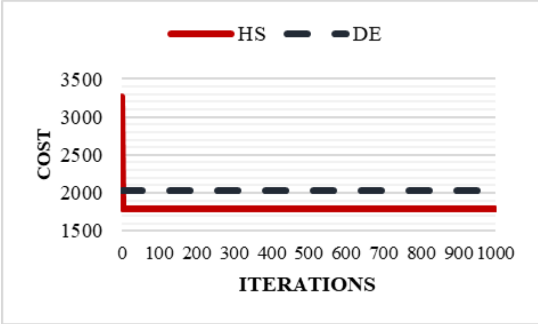

Figure 6. The comparison of convergence rates (Scenario 6 - population size of 5)

5.2.2. Simulation Results with a 20- Population Size

The experimental results for a population size of 20, as shown in table 3, indicate that the DE algorithm achieves the highest average scores in 66.7% of the scenarios, followed by the HS algorithm at 33.3%. Additionally, the results demonstrate that the DE algorithm consistently outperforms the others by providing the best solution across all scenarios (figure 7, figure 8, figure 9, figure 10, figure 11, figure 12). Meanwhile, the BWO algorithm exhibits the lowest performance in every experimental scenario.

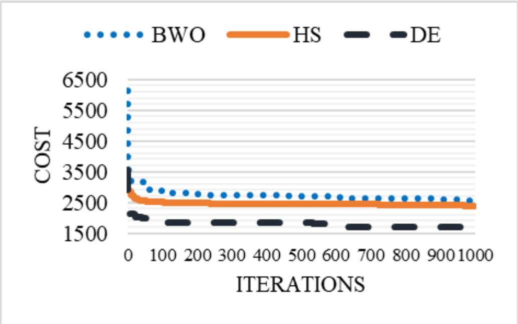

Figure 7. The comparison of convergence rates (Scenario 1 - population size of 20)

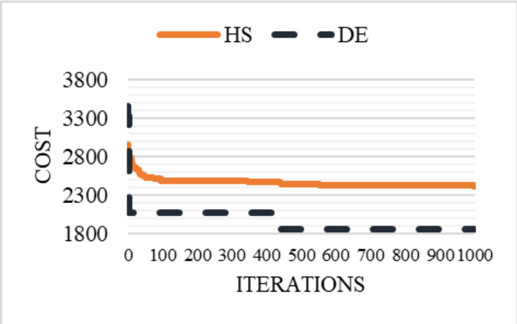

Figure 8. The comparison of convergence rates (Scenario 2 - population size of 20)

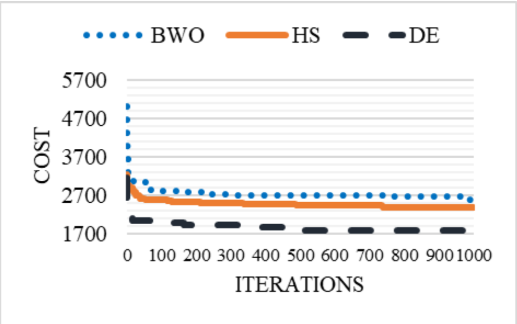

Figure 9. The comparison of convergence rates (Scenario 3 - population size of 20)

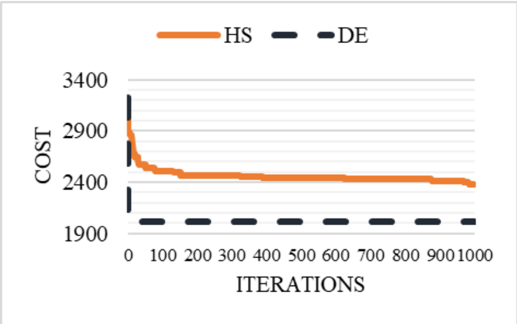

Figure 10. The comparison of convergence rates (Scenario 4 - population size of 20)

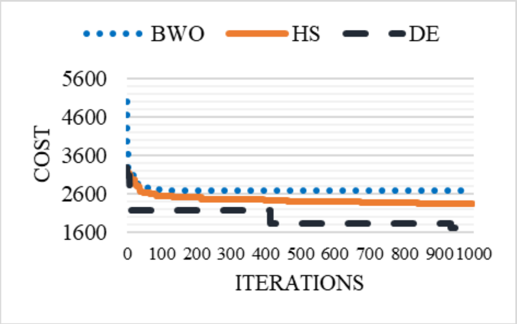

Figure 11. The comparison of convergence rates (Scenario 5 - population size of 20)

Figure 12. The comparison of convergence rates (Scenario 6 - population size of 20)

Table 3. Statistical results of fitness values for 30 experimental tests with a population size of 20

		Sc1	**Sc2**	**Sc3**	**Sc4**	**Sc5**	**Sc6**
DE	Mean	1.83E+03	2.41E+03	1.98E+03	2.58E+03	2.08E+03	2.58E+03
	Std	1.15E+02	3.17E+02	2.72E+02	2.90E+02	3.81E+02	3.45E+02
	Best	1.69E+03	1.86E+03	1.77E+03	2.01E+03	1.71E+03	1.77E+03
	Worst	2.19E+03	2.86E+03	2.85E+03	2.88E+03	2.92E+03	2.95E+03
HS	Mean	2.42E+03	2.44E+03	2.40E+03	2.43E+03	2.37E+03	2.42E+03
	Std	1.73E+01	1.22E+01	1.22E+01	1.66E+01	2.12E+01	2.38E+01
	Best	2.37E+03	2.42E+03	2.38E+03	2.38E+03	2.33E+03	2.40E+03
	Worst	2.45E+03	2.46E+03	2.42E+03	2.46E+03	2.43E+03	2.49E+03
BWO	Mean	2.66E+03	-	2.66E+03	-	2.66E+03	-
	Std	3.08E+01	-	2.87E+01	-	2.64E+01	-
	Best	2.57E+03	-	2.59E+03	-	2.59E+03	-
	Worst	2.71E+03	-	2.73E+03	-	2.73E+03	-

From a variability perspective, DE shows decreased standard deviation values (e.g., 1.15E+02 to 3.81E+02), indicating improved consistency with a larger population size. HS and BWO algorithms maintain low standard deviations, such as 1.22E+01 to 2.38E+01 and 2.64E+01 to 3.08E+01, respectively, demonstrating their consistent performance. These results emphasize the reliability of HS and BWO algorithms for consistent outcomes, while DE's superior solutions highlight its robustness despite slightly higher variability.

5.2.3. Simulation Results with a 100- Population Size

The experimental results for a population size of 100, as presented in table 4, further confirm the superior performance of the DE algorithm over the alternative algorithms. It consistently achieves the highest average scores across all tested scenarios in both mean and best fitness values, with the HS algorithm as the next best performer.

Table 4. Statistical results of fitness values for 30 experimental tests with a population size of 100

		Sc1	**Sc2**	**Sc3**	**Sc4**	**Sc5**	**Sc6**
DE	Mean	1.84E+03	2.11E+03	1.88E+03	2.10E+03	1.90E+03	2.32E+03
	Std	6.59E+01	1.49E+02	8.04E+01	1.69E+02	7.39E+01	3.43E+02
	Best	1.71E+03	1.88E+03	1.70E+03	1.76E+03	1.76E+03	1.78E+03
	Worst	1.98E+03	2.55E+03	2.07E+03	2.55E+03	2.03E+03	2.73E+03
HS	Mean	2.43E+03	2.45E+03	2.44E+03	2.46E+03	2.45E+03	2.48E+03
	Std	2.76E+01	1.93E+01	2.13E+01	2.04E+01	9.12E+01	2.50E+01

ISSN 2723-6471

	Best	2.35E+03	2.38E+03	2.40E+03	2.41E+03	1.98E+03	2.41E+03
	Worst	2.47E+03	2.48E+03	2.46E+03	2.49E+03	2.50E+03	2.51E+03
BWO	Mean	2.75E+03	-	2.74E+03	-	2.73E+03	-
	Std	2.20E+01	-	2.37E+01	-	2.15E+01	-
	Best	2.71E+03	-	2.68E+03	-	2.68E+03	-
	Worst	2.78E+03	-	2.78E+03	-	2.77E+03	-

The best convergence results achieved by the comparison algorithms in different parameter setting scenarios are shown in figure13, figure 14, figure 15, figure 16, figure 17, figure 18.

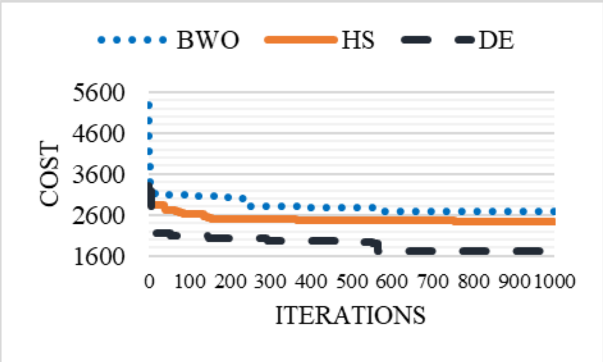

Figure 13. The comparison of convergence rates (Scenario 1 - population size of 100)

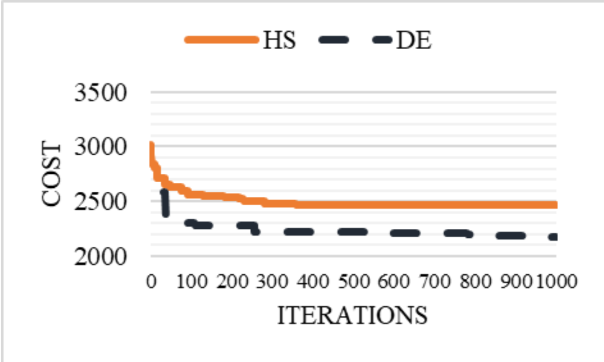

Figure 14. The comparison of convergence rates (Scenario 2 - population size of 100)

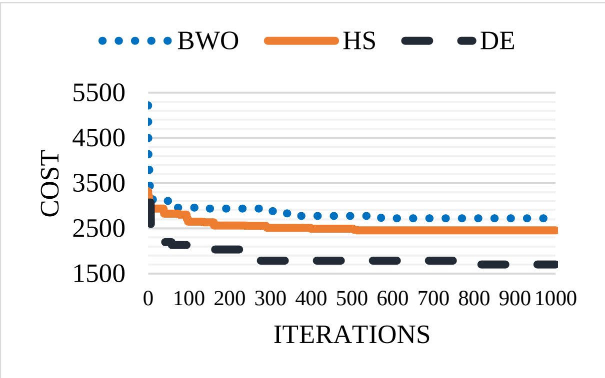

Figure 15. The comparison of convergence rates (Scenario 3 - population size of 100)

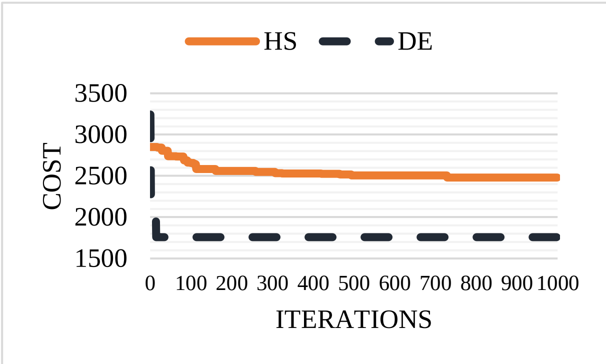

Figure 16. The comparison of convergence rates (Scenario 4 - population size of 100)

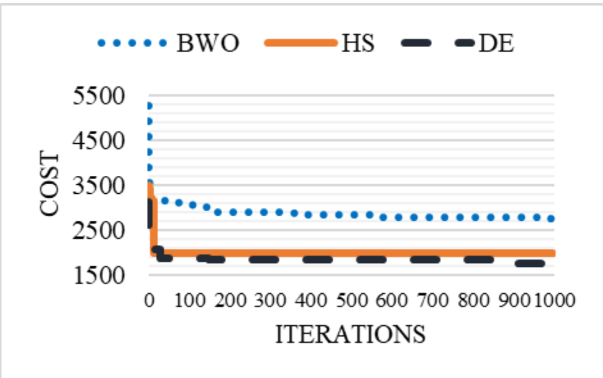

Figure 17. The comparison of convergence rates (Scenario 5 - population size of 100)

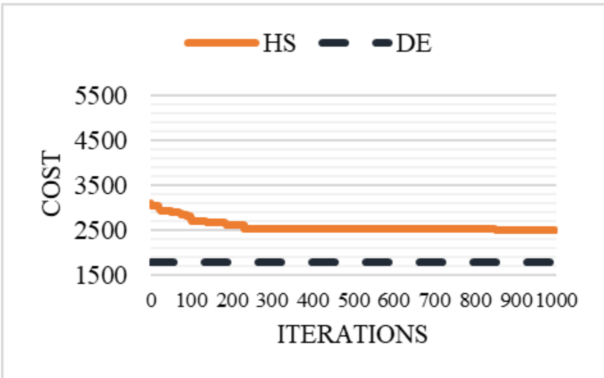

Figure 18. The comparison of convergence rates (Scenario 6 - population size of 100)

The standard deviations of the DE algorithm, ranging from 6.59E+01 to 3.43E+02, indicate moderate to high variability. The larger population size has not significantly improved consistency. HS, however, retains its consistent performance with generally low standard deviations, ranging from 1.93E+01 to 9.12E+01. The results also demonstrated that, even though the BWO algorithm continues to exhibit low variability (2.15E+01 to 2.37E+01), it is unable to achieve competitive optimization results.

5.3. Comparative Analysis

The comparative analysis of the optimization algorithms across all population sizes revealed that the DE algorithm consistently outperformed both the HS and BWO algorithms across most tested scenarios. DE demonstrated superior performance in achieving optimal solutions by effectively balancing exploration and exploitation while handling problem-specific constraints.

In contrast, the BWO algorithm exhibited the lowest performance, struggling to converge efficiently and maintain competitive results, particularly in high-dimensional and complex problem settings. Furthermore, the results suggest that HS and BWO are more robust and reliable in terms of consistent performance. However, while DE often achieved lower mean fitness values, it remains more susceptible to variability in its solution quality across different runs.

These findings underscore the superiority of the DE algorithm over others in the evaluated scenarios while highlighting the limitations of BWO. To address these limitations, potential improvements include dynamically adjusting parameters during the search process to better balance exploration and exploitation, as well as hybridizing BWO with local search techniques or other metaheuristics to enhance its solution refinement capabilities.

Population size significantly affects the trade-off between exploration and exploitation in DE, HS, and BWO algorithms. For DE, larger populations promote exploration by increasing the diversity of solutions, enabling superior performance, while smaller populations risk premature convergence. HS maintains strong performance across different sizes, efficiently balancing exploration and exploitation by taking memory into account. In BWO, larger populations improve exploration but weaken exploitation, while smaller populations favor exploitation but risk trapping local optimums. In general, DE benefits more from larger populations, while HS and BWO are less sensitive to size changes, with BWO leaning toward exploitation. Tailoring population size is essential for problem-specific optimization. Furthermore, most experiments show that DE and HS algorithms perform better in odd-numbered scenarios (i.e., 1, 3, and 5), where lower values of Pm and PAR are observed, compared to even-numbered scenarios. This indicates that the algorithms excel in exploration tasks, as lower values of Pm and PAR are associated with greater diversity in the search process, allowing the algorithms to cover the solution space more effectively. On the other hand, higher values of Pm and PAR, which emphasize exploitation, appear to limit the algorithms' ability to escape local optima and adapt to varying problem landscapes.

Generally, the time complexity of HS, BWO and DE algorithms depends on population size, number of generations, and problem dimensionality. However, the BWO introduces an additional computational overhead due to its unique cannibalism operation. This step requires selecting and removing weaker offspring solutions in each generation, adding to its overall time complexity compared to HS and DE.

Finally, optimization models can be practically implemented in the context of Hajj transportation scheduling. Specifically, optimization algorithms can be embedded within existing transportation management software to leverage real-time data, such as vehicle locations, traffic congestion levels, and demand fluctuations, for dynamic scheduling and route adjustments. These models exhibit scalability, making them suitable for large-scale events like Hajj, and can be adapted to accommodate real-world constraints, including vehicle capacity, time windows, and prevailing road conditions. To ensure successful implementation, pilot programs can be conducted in controlled environments to refine model parameters and address potential challenges such as data reliability and seamless system integration before full-scale deployment. Furthermore, effective stakeholder collaboration with transportation authorities and system developers is crucial to ensure compatibility with existing policies, infrastructure, and technology.

6. CONCLUSION

This study addresses the critical need to enhance transportation efficiency during large-scale events, with a particular focus on Hajj—a complex event involving diverse transportation and accommodation requirements. In this paper, three algorithms—HS, DE, and BWO—have been successfully applied to address the transportation challenges faced during the Hajj pilgrimage, demonstrating their potential for optimizing logistics and improving the efficiency of transportation systems. The experimental results demonstrate that HS performs well on smaller problem sizes, while DE consistently outperforms the other algorithms in large-scale optimization tasks, effectively handling complex requirements. The performance of all algorithms varies with parameter tuning: lower mutation probabilities improve solutions across algorithms, DE thrives with lower crossover probabilities, and HS benefits from higher exploration probabilities. In contrast, BWO exhibits limited adaptability and consistently delivers lower-quality solutions.

The findings provide valuable insights into the strengths and limitations of the HS, DE, and BWO algorithms, highlighting opportunities for further exploration. Future research directions include expanding the algorithm comparison by incorporating widely used methods such as Genetic Algorithms and Simulated Annealing to identify complementary strengths and weaknesses. This will pave the way for hybridization strategies that combine the advantages of different algorithms to enhance overall performance. Furthermore, future research should focus on implementing automated parameter optimization techniques to improve the robustness and efficiency of these algorithms, performing a comprehensive error analysis to investigate the impact of parameter variations and scenario characteristics on algorithm performance, exploring the potential of advanced technology and applications in transportation scheduling and optimizing logistics [13], [14], [15], [16], [17], and conducting a survey of ongoing studies in the field similar to [18], [19], [20] to identify current research trends and potential areas for future collaboration.

7. Declarations

7.1. Author Contributions

Conceptualization: M.K.S.; Methodology: M.K.S.; Software: M.K.S.; Validation: M.K.S.; Formal Analysis: M.K.S.; Investigation: M.K.S.; Resources: M.K.S.; Data Curation: M.K.S.; Writing Original Draft Preparation: M.K.S.; Writing Review and Editing: M.K.S.; Visualization: M.K.S.; All authors have read and agreed to the published version of the manuscript.

7.2. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

7.3. Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

7.4. Institutional Review Board Statement

Not applicable.

7.5. Informed Consent Statement

Not applicable.

7.6. Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] M. K. Shambour, E. Khan, and M. A. Abu-Hashem, "A hybrid scheduling approach for mega event transportation: integrating harmony search and black widow optimization," *PeerJ Computer Science*, vol. 10, no. e2526, pp. 1-27, 2024.

[2] V. Hayyolalam and A. A. Pourhaji Kazem, "Black Widow Optimization Algorithm: A novel meta-heuristic approach for solving engineering optimization problems," *Engineering Applications of Artificial Intelligence,* vol. 87, no.10, pp. 1-28, Jan. 2020.

[3] R. R. Mostafa, A. M. Khedr, Z. Al Aghbari, I. Afyouni, I. Kamel, and N. Ahmed, "An adaptive hybrid mutated differential evolution feature selection method for low and high-dimensional medical datasets," *Knowledge-Based Systems*, vol. 283, no. 1, pp. 9, 2024.

[4] F. Rehman and E. Felemban, “A preference-based interactive tool for safe rescheduling of groups for hajj,” in *Proc. IEEE/ACS Int. Conf. Comput. Syst. Appl. (AICCSA)*, Nov. 2019, vol. 16, no. 11, pp. 1-6.

[5] E. Felemban , K. Qahtani , A. Hawsawi , A. Shahri. “Crowd simulation model for crowd movement in the Holy Mosque”. *17th Scientific Forum for the Research of Hajj.* vol. 15, no. 1, pp. 727-737, 2017.

[6] A. Morgan and K. M. J. Khayyat, “Improving emergency services efficiency during Islamic pilgrimage through optimal allocation of facilities,” *International Transactions in Operational Research*, vol. 29, no. 1, pp. 259–300, Jan. 2022.

[7] S. Al-Sabban, H. Ramadan, “A simulation study of the shuttle-bus pilgrim transportation system between the holy sites for the 1422H Hajj season”. *King AbdulAziz University, Engineering Sciences*, vol. 16, no. 2, pp. 71-93. 2005

[8] O. Hussain, E. Felemban, and F. Rehman, “Optimization of the Mashaer shuttle-bus service in Hajj: Arafat-Muzdalifah case study,” *Information*, vol. 12, no. 12, pp. 496-512, 2021.

[9] K. Haase, M. Kasper, M. Koch, and S. Müller, “A pilgrim scheduling approach to increase safety during the Hajj,” *Operations Research*, vol. 1 67, no. 2, pp. 376-406, 2019.

[10] M. S. Yasein and E. A. Khan, “Optimizing Shuttle-Bus Systems in Mega-Events using Computer Modeling: A Case Study of Pilgrims’ Transportation System,” *IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 14, no. 11, pp. 222-233. 2023.

[11] E. Felemban, A. Fatani and F. U. Rehman, "An Optimized Scheduling Process for a Large Crowd to Perform Spatio-temporal Movements Safely during Pilgrimage," *2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA,* 2019, vol. 8, no. 12, pp. 6049-6051, 2019.

[12] M. A. Abu-Hashem and M. K. Shambour, “An improved black widow optimization (IBWO) algorithm for solving global optimization problems,” *Int. J. Ind. Eng. Comput*., vol. 15, no. 3, pp. 1–16, 2024.

[13] S. Almuaythir, A. K. Singh, M. Alhusban, et al., "Robotics technology: catalyst for sustainable development—impact on innovation, healthcare, inequality, and economic growth," *Discovery Sustainability*, vol. 5, no.12, pp. 1-14, 2024.

[14] E. A. Khan and M. K. Shambour, "Pilgrims Services Optimization During Hajj Mega Event Utilizing Heuristic Algorithms," in *Proc. 24th Int. Arab Conf. Inf. Technol. (ACIT)*, Dec. 2023, vol. 4, no. 12, pp. 1-6. 10, 2023.

[15] S. Govindaraju, M. Indirani, S. S. Maidin, and J. Wei, "Intelligent Transportation System's Machine Learning-Based Traffic Prediction," *Journal of Applied Data Sciences*, vol. 5, no. 4, pp. 1826–1837, 2024.

[16] Z. Ning, S. Sun, X. Wang, L. Guo, S. Guo, X. Hu, B. Hu, and R. Y. K. Kwok, "Blockchain-enabled intelligent transportation systems: A distributed crowdsensing framework," *IEEE Transactions on Mobile Computing*, vol. 21, no. 12, pp. 4201–4217, 2021.

[17] F. Elghaish, S. T. Matarneh, and M. Alhusban,. “The application of “deep learning” in construction site management: A scientometric, thematic, and critical analysis”. *Construction Innovation*, vol. 22, no. 3, pp. 580-603, 2022.

[18] S. S. Maidin, N. Yahya, M. A. bin Fauri Fauzi, and N. S. Bakar, "Current and Future Trends for Sustainable Software Development: Software Security in Agile and Hybrid Agile through Bibliometric Analysis," *Journal of Applied Data Sciences*, vol. 6, no. 1, pp. 311-324, 2025.

[19] S. Kaffash, A. T. Nguyen, and J. Zhu, "Big data algorithms and applications in intelligent transportation system: A review and bibliometric analysis," *International Journal of Production Economics*, vol. 231, no. 7, pp. 18, 2021.

[20] D. Oladimeji, K. Gupta, N. A. Kose, K. Gundogan, L. Ge, and F. Liang, "Smart transportation: An overview of technologies and applications," *Sensors*, vol. 23, no. 8, pp. 1-32, 2023.