An Analysis of Various Optimization Techniques for Elevator Scheduling

Pranav Nataraj Devaraj
School of Computer Science and Engineering
Vellore Institute of Technology
Chennai, India
pranav.nataraj2022@vitstudent.ac.in

Surya Adhikari
School of Computer Science and Engineering
Vellore Institute of Technology
Chennai, India
surya.adhikari2022@vitstudent.ac.in

Aaryav Gangrade
School of Computer Science and Engineering
Vellore Institute of Technology
Chennai, India
aaryav.gangrade2022@vitstudent.ac.in

A. Ilavendhan
Assistant Professor
School of Computer Science and Engineering
Vellore Institute of Technology
Chennai, India
ilavendhana62@pec.edu

***Abstract*—Optimization of elevator scheduling is a challenging and important problem that affects the efficiency and comfort of vertical transportation in high-rise buildings. Elevator scheduling enables elevator systems to respond to user requests in an optimal manner to save the user's time. With the increase in urbanization there are more high-rise buildings being constructed and the need for efficient solutions to the problem of elevator scheduling are becoming more and more important. In recent years, many researchers have proposed various methods to improve the performance of elevator systems, such as genetic algorithms, swarm intelligence, other machine learning models and the use of advance information. We use different criteria to evaluate the performance of these algorithms, such as waiting time, travel time, and energy consumption. The focus, however, is on the reduction of waiting and travel times. We review the advantages and disadvantages of each approach and discuss their complexity and scalability.**

***Keywords*— Group Elevator Scheduling, Waiting Time, Travel Time, Traffic Patterns.**

I. Introduction

In an era where urbanization is rapidly increasing, elevators have become an indispensable part of modern living. From towering skyscrapers to everyday residential buildings, elevators play a pivotal role in ensuring efficient vertical transportation. The optimization of elevator algorithms, which dictate the movement of elevators and allocation of resources, has garnered significant attention from researchers, engineers, and building operators. The aim is clear: to enhance elevator performance, reduce waiting times, energy consumption, and ultimately improve the quality of life for millions of people worldwide.

Elevator optimization is a multifaceted challenge, influenced by several factors such as traffic patterns, building architecture, and user behavior. Consequently, researchers have explored a multitude of approaches to address these complexities and improve elevator system performance. This research paper embarks on a comprehensive exploration of these diverse optimization strategies,

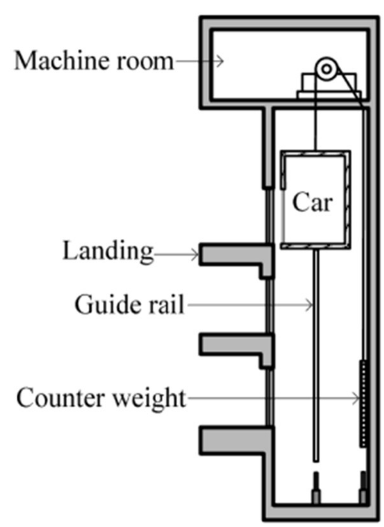

Fig 1.: Schematic of Traction Elevator. Image credit to *thecontructor.org*.

aiming to provide a clear and structured analysis of their strengths, weaknesses, and real-world applicability.

Both hardware and software solutions can be developed for this problem. Hardware solutions mainly work by increasing the efficiency of human interactions with elevators or by increasing the capabilities of the elevator. On the other hand, software solutions aim to increase efficiency by making sure that every elevator is always operating to.

its maximum capability and serving the users in the most efficient manner. This is called elevator scheduling.

Understanding the intricacies of elevator scheduling optimization is not only of academic interest but also holds immense practical significance. By shedding light on the comparative performance of these approaches, this research aims to guide building designers, engineers, and stakeholders in making informed decisions to enhance elevator systems, ultimately contributing to improved urban mobility and a more sustainable built environment. It holds the potential to save builders money by reducing the number of elevators required to meet demand. Needless to say it will also benefit the users by saving the time they spend waiting and travelling.

II. Evaluating Criteria Of Algorithms

There are many characteristics of an elevator scheduling algorithm that can be used to assess the performance of an algorithm. These characteristics can also be used to categorize the algorithms based on which characteristics they optimize and make it easy for us to identify the correct algorithms for the correct situations.

A. Waiting Time

Waiting time is a term commonly used in CPU Scheduling algorithms. In the context of this paper, it is the amount of time that a user who requires the elevator needs to wait after requesting for it. It is calculated as the amount of time passed from when the user submits the request to use an elevator till the elevator reaches the floor the user is on, and the user can enter it. This definition holds true for most scheduling techniques. It can be represented by the following expression.

$$W = t_1 - t_0 \tag{1}$$

Where W is the waiting time, t_0 is the time at which the user called the elevator and t_1 is the time at which the elevator arrived at the user's floor to serve their request.

B. Travel Time

In the context of this paper, it is the amount of time that an elevator takes to fulfill the request of a user. It is calculated as the amount of time passed from when the user entered an elevator till they are taken to their desired floor and exit the elevator. This definition holds true for most scheduling techniques.

$$Tr = t_f - t_i \tag{2}$$

Where Tr is the travel time, t_i is the time at which the user entered the elevator and t_f is the time at which the elevator reaches the user's desired floor and the user exits the elevator.

C. Energy Efficiency

This criterion considers how much energy is saved in the operation of the elevator system by using an algorithm as compared to other algorithms. This metric is measured differently in different optimization techniques.

III. Situation Variables

There are many different variables to consider when implementing an elevator system. These choices are influenced by many factors such as the architecture of the building, financial budget, foot traffic and availability.

A. Number of Elevators

It is important to note that we are trying to optimize *systems* of elevators, this may amount to optimizing a single elevator or multiple elevators. We must use all the available elevators to optimize their performance as a whole, not individually.

Fig 2.: Elevator Operating Interface. Image by *Freepik.com*.

B. Input Interface

There are many different interfaces which companies have developed for users to operate elevators. For instance, in order to call the elevator, the user may only have to indicate the direction they want to travel in, i.e., upwards or downwards, and then once inside the elevator specify the destination floor. While in another system with a different type of interface they may have to specify which floor, they want to travel to while calling the elevator. Fig.2 shows the most common type of elevator interface.

C. Weight Capacity

The weight capacity can be a crucial factor to consider when there is very heavy traffic in which case the lift is almost always at capacity weight. In these cases, completing the current requests must take priority over responding to new requests.

D. Number of Floors

The number of floors can also be an important consideration. Elevators can be assigned certain floors to operate on, or they could be limited by hardware to operate only on certain floors.

E. Traffic Pattern

The traffic pattern if studied can provide insight into what algorithms best suit the needs of the users. We can also have different algorithms for different times of the day based on the traffic pattern.

IV. Optimization Techniques

A. Machine learning

There are many machine learning algorithms which are used in many innovative optimization solutions. Machine learning has gained popularity in many research areas across the globe and elevator scheduling is not an exception. Researchers have come up with many innovative uses of machine learning in elevator scheduling. We will discuss four such algorithms.

Simulated Annealing (SA) stands as a heuristic approach extensively used for addressing optimization problems. It draws its inspiration from the annealing process employed in metallurgy, where metals are gradually cooled to reduce their potential energy. In SA, an iterative movement is harnessed, driven by an adaptable temperature parameter. The core mechanism of the algorithm involves a continuous comparison between the outcomes of the objective function operating with the current and neighboring points within the domain. Notably, SA allows for occasional uphill movements, enabling it to escape local optima and explore the search space more comprehensively. SA has gained wide applicability in various optimization problems, including the Elevator Dispatching Problem (EDP). In the context of EDP, SA is actively deployed with the objective of determining the optimal elevator route to minimize passenger journey time. This algorithm is put to the test within a case study, and fitness values are computed to assess its performance.

Genetic Algorithm (GA) represents another optimization technique frequently applied to the EDP. GA draws its inspiration from the principles governing natural selection and genetics. The algorithm embarks on its journey with an initial population comprising prospective solutions, each represented as chromosomes. These chromosomes undergo genetic operations, including crossover and mutation, to generate offspring. which, in this context, pertains to the average journey time for all passengers. The fittest The fitness of each chromosome is rigorously evaluated based on the objective function, individuals are earmarked for the formation of the subsequent generation, while their less fit counterparts are eliminated from the pool. This process is reiterated across multiple generations until an optimal solution is unearthed. Notably, GA has showcased remarkable promise in tackling the EDP, often outperforming other algorithms concerning fitness value. The results from [4] are shown in Fig.3 and Table I.

The Particle Swarm Optimization Algorithm (PSO) represents a population-based optimization technique that finds inspiration in the social behavior of birds flocking or fish schooling. In the realm of PSO, a cluster of particles steps in as stand-ins for potential solutions. Each particle assumes a defined position and velocity within the search space. These particles embark on a quest to attain superior positions, driven by their own experiences and those of the top-performing particle within the swarm. Through iterative updates to their positions and velocities, the particles collectively strive to unearth the optimal solution. PSO, when wielded in the EDP, often necessitates a transformation of the problem into an arithmetic configuration. The fitness values of the particles come under rigorous scrutiny, with the average journey time emerging as the key metric. It's worth noting that PSO frequently delivers competitive results when tackling the EDP, although in scenarios featuring a higher number of passengers, it may require more iterations to yield optimal solutions.

Whale Optimization Algorithm (WOA) is a relatively novel optimization algorithm that takes its inspiration from the hunting practices of humpback whales. WOA takes its initial steps with a set of randomly selected solutions, each represented as a search agent. With each iteration, these search agents undertake position updates grounded in either a randomly chosen search agent or the best solution encountered thus far. The algorithm meticulously balances the act of exploration and exploitation by modulating a parameter referred to as 'a,' which undergoes a decrement from 2 to 0. When ‘a’ holds a, value greater than 1, a random search agent is chosen for position updates, while when ‘a’ dwindles below 1,the best solution takes center stage.

WOA, when harnessed for the EDP, entails the transformation of the problem into an arithmetic mold. The algorithm's performance undergoes evaluation, predominantly centered around fitness values. Interestingly, it’s noted that WOA occasionally lags in terms of fitness value performance when measured against other algorithms.

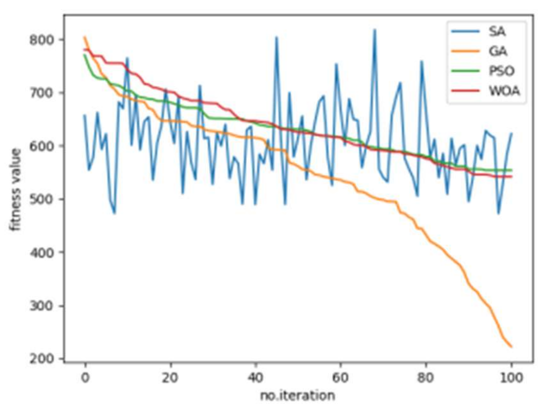

Fig 3.: Convergence of Fitness Value During Training. Graph credit to [1].

In summary, the four optimization algorithms pressed into service to resolve the EDP—Simulated Annealing, Genetic Algorithm, Particle Swarm Optimization Algorithm, and Whale Optimization Algorithm—each bring to the table a distinct approach to resolving optimization problems. They are adeptly molded to the demands of the EDP by means of a problem transformation and rigorous evaluation of fitness values to gauge their efficacy. Among this quartet, Genetic Algorithm emerges as the front-runner in terms of fitness value, with the remaining algorithms offering competitive results.[1][3][5][6]

B. Real-Time Matrix Iterative Algorithm

This solution, presented in [2], aims to improve the operation speed and utilization efficiency of elevators in high-rise buildings. It introduces a dynamic matrix iterative model and utilizes indoor navigation technology to help users quickly find elevators. The algorithm also defines data filtering criteria for user waiting time and elevator remaining space. By optimizing the reservation elevator groups, the algorithm maximizes elevator utilization, reduces waiting time for users, and enhances the overall efficiency of elevator operations.

The algorithm helps users find elevators quickly by incorporating indoor navigation technology. It utilizes the UWB (Ultra-Wideband) algorithm for indoor navigation, allowing users to locate nearby elevators efficiently. This feature enhances user convenience and improves the overall user experience in high-rise buildings.

The algorithm optimizes the system by implementing various steps. It involves data pre-processing to filter and optimize reservation elevator data. It considers time optimization by prioritizing users who are close to the elevator and can meet the time filtering rules.

TABLE I. Fitness Values Of ML Models

Performance Index	SA	GA	PSO	WOA
Avg. Fitness value in 5 Runs	531.4s	279.1s	600s	620s
Optimal Fitness Value	472.3s	222.3s	553.3s	541.3s

The algorithm also ensures automatic load limit to prevent elevator overloading. Additionally, it incorporates hierarchical management optimization for high-rise buildings, dividing elevators into high, medium, and low segments. This model enhances coordination, response time, and adaptive functionality. The algorithm utilizes matrix iteration to calculate the number of people on each floor, the number of people waiting, and the capacity matrix. This optimization scheme aims to maximize elevator utilization, reduce waiting time, and improve the overall efficiency of elevator operations.

For a single elevator, the time complexity of the algorithm is determined by factors such as the total number of reservations, the total number of floor destinations, and the round-trip cycle. On the other hand, the complexity of the elevator group is influenced by the total number of people, the total number of floor destinations, and the round-trip period of each elevator. The document [2] presents the complexity formula for the elevator group. It also states that the complexity of the elevator group is positively correlated with the number of tasks. The research concludes that the complexity of the elevator group falls within a range, with the lower limit being the complexity of a single elevator and the upper limit approaching m times the complexity of a single elevator.

Overall, the real-time matrix iterative optimization algorithm aims to enhance the operation speed and utilization efficiency of elevator groups. It incorporates indoor navigation technology, data filtering criteria, and hierarchical management optimization to improve the overall efficiency of elevator operations in high-rise buildings.[2]

C. Advance Information

This approach discusses the optimization of group elevator scheduling with advance traffic information. It addresses the challenge of developing new scheduling methods that effectively utilize this information. The paper [7] presents a two-level formulation to solve the problem, with passenger-to-car assignment at the high-level and single car dispatching at the low-level. The formulation takes into account various traffic patterns, complicated car dynamics, and the combinatorial explosion of the search space.

To improve the flexibility of elevators, a new door action control method is proposed, which allows elevators to make decisions about when to close the door without relying solely on door dwell time. The methodology also incorporates the use of simulation models to embed detailed car dynamics for performance evaluation.

The high-level optimization involves optimizing passenger-to-car assignment based on advance traffic information. A hybrid nested partitions and genetic algorithm method is developed for this purpose, which can be extended to solve a generic class of sequential decision problems. At the low-level, a trip-based heuristic is used to optimize single car dispatching for individual cars.

Numerical results demonstrate the effectiveness of the proposed method in terms of solution quality, computational efficiency, and the value of advance information and the new door action control method. Research also highlights the need for further improvement to reduce CPU time for online implementation.[7]

D. Monte Carlo Algorithm

The first step in the solution presented in [4] is, a high-rise residential building with 20 floors and 4 independent elevators is selected as the research object. To measure the number of passengers, an Automatic Counter Module (ACM) based on an infrared sensor is created. This module records the number of passengers on different floors, providing data for analysis.

Next, the probability of residents on different floors taking the elevator during the peak period is calculated. The researchers use the Monte Carlo Algorithm, which involves generating random numbers that follow a Poisson distribution with a specified lambda value. These random numbers represent the people coming to the elevators. The probability of taking the elevator is then determined based on this data.

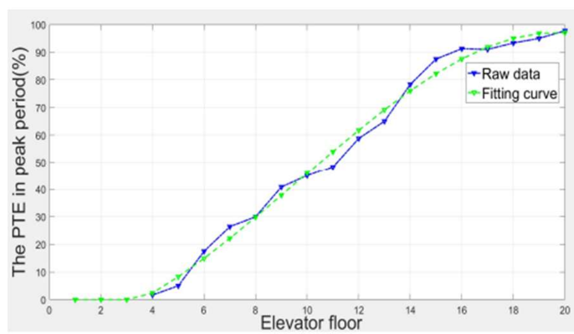

Fig. 4. Probability of Taking Elevator. Graph credit to [4].

Next, the probability of residents on different floors taking the elevator during the peak period is calculated. The researchers, in [4], use the Monte Carlo Algorithm, which involves generating random numbers that follow a Poisson distribution with a specified lambda value. These random numbers represent the people coming to the elevators. The probability of taking the elevator is then determined based on this data.

To further refine the optimization model, a cubic polynomial fitting curve is generated using the least square method. This curve, Fig.4., represents the probability distribution of taking the elevator with the floor as the independent variable. The Monte Carlo Algorithm is then applied again, incorporating the probability of taking the elevator obtained from the fitting curve. This allows for a more accurate evaluation of the elevator scheduling parameters.

The optimized model is compared with the existing model in terms of elevator running efficiency and the number of people waiting for the elevator. The results show that considering the probability of taking the elevator significantly improves the accuracy of the model. The existing model has a relative error of 10%-20% compared to the optimized model, and the number of passengers transported below 10 floors is significantly reduced.

Overall, the methodology involves data acquisition through the ACM, probability calculation using the Monte Carlo Algorithm, curve fitting using the least square method, and evaluation of the optimized model compared to the existing model. This approach allows for a more accurate and effective optimization of the elevator system. [4]

TABLE II. COMPARISON OF TECHNIQUES

Optimization Technique	**Advantages**	**Disadvantages**
Machine Learning	They offer a data driven optimization allowing for adaptive decision making.	They require large datasets to train.
	They can handle complex non-linear relationships between variables.	The underlying decision-making logic may be hard to discern.
	They can learn from historical data and improve over time.	They require high quality datasets. Badly curated datasets may result in inaccuracy and biased results.
	They can optimize multiple factors like waiting time, travel time and energy efficiency at the same time.	The computation can be time taking and heavy.
Matrix Iterative Algorithm	This solution incorporates hardware solutions.	May require advanced technology and Infrastructure.
	This solution considers data filtering factors such as users' proximity to elevators and load capacity of elevators.	Is dependent on data used for data filtering.
		Needs to be updated periodically.
Advanced Information	Is based on real time information	May require extra infrastructure and data acquisition.
	Shows better response to changing traffic conditions	Relies on accuracy of data acquisition technology.
Monte Carlo Algorithm	Enables accurate modeling and prediction of the algorithm's behavior.	May require large number of iterations to yield sound results.
	It can handle complex situations and take into account multiple factors.	Implementation requires advanced statistical and mathematical knowledge.
	Can incorporate uncertainty into optimization process thus yielding more realistic results.	Understanding the underlying working may tough due to complexity of algorithm.

In Table II we have noted all the pros and cons of the different optimization techniques developed by researchers. Upon comparison it can be said that each optimization technique has its own conditions under which it can outperform the other techniques. Each technique has its own limitations as well. One must consider all the situation variables and the main goals of optimization in order to choose the most optimal technique for any use case. Further research is being conducted in this field to improve solutions for specific use cases and in developing solutions which can be applied to a wide range of use cases.

VI. CONCLUSION

In conclusion, this research paper provides a comprehensive exploration of various approaches to elevator algorithm optimization. The paper examines both traditional and advanced techniques, including genetic algorithms, swarm intelligence, Advanced information and others. Overall, this research paper contributes to the field of elevator optimization by providing valuable insights into different approaches and their real-world applicability. By guiding building designers, engineers, and stakeholders, this study aims to enhance urban mobility and contribute to a more sustainable built environment.

REFERENCES

[1] "Investigation of Optimization Techniques on the Elevator Dispatching Problem" by Shaher Ahmed, Mohamed Shekha, Suhaila Skran, and Abdelrahman Bassyouny, published in arXiv:2202.13092 in 2022.

[2] Chen, W.; Zheng, B.; Liu, J.; Li, L.; Ren, X. A Real-Time Matrix Iterative Optimization Algorithm of Booking Elevator Group and Numerical Simulation Formed by Multi-Sensor Combination. *Electronics* **2021**, *10*, 3179. https://doi.org/10.3390/electronics10243179

[3] "A Survey of Optimization Methods from a Machine Learning Perspective" by Jiajin Li, Zhe Wang, and Xiaoyu Liu, published in arXiv:1906.06821 in 2019.

[4] Y. Zhang, H. Zhao, S. Huang and F. Guo, "An elevator scheduling optimization model considering the probability of taking the elevator based on Monte Carlo Algorithm," 2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, China, 2020, pp. 1-4, doi: 10.1109/ITOEC49072.2020.9141641.

[5] Hanif, M., Mohammad, N. (2022). Performance Analysis of Particle Swarm Optimization and Genetic Algorithm in Energy-Saving Elevator Group Control System. In: Arefin, M.S., Kaiser, M.S., Bandyopadhyay, A., Ahad, M.A.R., Ray, K. (eds) Proceedings of the International Conference on Big Data, IoT, and Machine Learning. Lecture Notes on Data Engineering and Communications Technologies, vol 95. Springer, Singapore. https://doi.org/10.1007/978-981-16-6636-0_38

[6] X. Wang, "Intelligent Scheduling System for Elevators Based on Particle Swarm Optimization Algorithm," 2022 IEEE 5th Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), Chongqing, China, 2022, pp. 1163-1166, doi: 10.1109/IMCEC55388.2022.10019817.

[7] J. Sun, Q. -C. Zhao and P. B. Luh, "Optimization of Group Elevator Scheduling With Advance Information," in IEEE Transactions on Automation Science and Engineering, vol. 7, no. 2, pp. 352-363, April 2010, doi: 10.1109/TASE.2009.2024242.

[8] L. -D. Van, Y. -B. Lin, T. -H. Wu and Y. -C. Lin, "An Intelligent Elevator Development and Management System," in IEEE Systems Journal, vol. 14, no. 2, pp. 3015-3026, June 2020, doi: 10.1109/JSYST.2019.2919967.

[9] S. -Y. Chou, D. A. Budhi, A. Dewabharata and F. E. Zulvia, "Improving elevator dynamic control policies based on energy and demand visibility," 2018 3rd International Conference on Intelligent Green Building and Smart Grid (IGBSG), Yilan, Taiwan, 2018, pp. 1-4, doi: 10.1109/IGBSG.2018.8393578.

[10] L. -D. Van, Y. -B. Lin, T. -H. Wu and T. -H. Chao, "Green Elevator Scheduling Based on IoT Communications," in IEEE Access, vol. 8, pp. 38404-38415, 2020, doi: 10.1109/ACCESS.2020.2975248.