Discrete Event Simulation (DES) of Office Elevator Congestion During Peak Hours

Farah Binti Fuaad
22009174
Universiti Teknologi PETRONAS

Amirul Hakim Muhd Hafidz
21000269
Universiti Teknologi PETRONAS

Qasrina Nursabrina Binti Zainal Abidin
22006859
Universiti Teknologi PETRONAS

Nur Hasanah Binti Mubarak
22008974
Universiti Teknologi PETRONAS

Tanushriie Poobalan
21001301
Universiti Teknologi PETRONAS

ABSTRACT

Elevator congestion during peak hours carries a significant challenge in high-rise office buildings, affecting both operational efficiency and user satisfaction. This project presents a Discrete Event Simulation (DES) model developed using SimPy to analyze elevator traffic flow in a 20-storey commercial building. The simulation replicates real-world conditions by modeling passenger arrivals at the ground floor with randomized destination floors and patience thresholds. Passengers who experience excessive waiting times may abandon the queue and opt for stairways, reflecting realistic behavioral dynamics. The model incorporates elevator capacity constraints, demand-responsive movement, and group travel behavior to evaluate system performance under varying load conditions. The findings aim to support the development of more efficient elevator control systems that enhance vertical mobility during high-demand periods.

TEAM ORGANIZATION

Our team collaboratively developed a discrete event simulation project on office elevator congestion during peak hours. Farah Fuaad served as the Project Lead and Main Simulation Developer, overseeing the overall workflow and implementing the SimPy-based simulation model. She also contributed to dataset analysis and designed the simulation flow, ensuring that the logic and structure aligned with the project's objectives. Qasrina Nursabrina was responsible for results interpretation and observation analysis, generating visual outputs and identifying performance trends. She also contributed to the initial pseudocode development, laying the basis for the simulation logic. Amirul Hakim Muhd Hafidz led the literature review and research, synthesizing insights from academic sources to inform the simulation framework. He also assisted in the early development of the simulation flowchart. Nur Hasanah supported the team by helping to frame the project overview and contributed to dataset justification, ensuring that the simulation assumptions reflected realistic elevator usage patterns. Tanushriie played a key role in the discussion and interpretation phase, helping to connect simulation findings with practical implications and user experience considerations. Each member played a crucial and distinct role, ensuring the project was academically supported, technically accurate, and effectively communicated.

1.0 INTRODUCTION

1.1. Background

In the modern era of buildings development, elevator systems are essential for facilitating vertical mobility in high-rise office buildings. As building occupancy increases and architectural designs become more complex, elevator performance during peak periods such as morning arrivals and end-of-day departures has become a critical concern. These peak intervals often lead to congestion, long queues, and reduced service satisfaction. Traditional analytical approaches, such as queuing theory, offer limited insight into the dynamic and unpredictable nature of elevator traffic. To address this, simulation-based methods have gained traction for their ability to model real-world behaviors and system interactions. This study applies a Discrete Event Simulation (DES) approach using SimPy to replicate elevator operations and passenger behavior in a typical multi-story commercial building. The goal is to better understand congestion patterns and explore strategies for improving elevator efficiency during high-demand periods.

1.2. Problem Statement

Elevator congestion during peak hours is a common challenge in multi-story office buildings, particularly during morning arrivals, lunch breaks, and end-of-day departures. These high-demand periods often result in long queues, extended waiting times, and inefficient elevator utilization, which can negatively impact overall building operations and user experience. In typical commercial buildings, elevator systems may struggle to accommodate sudden surges in passenger traffic due to limitations in scheduling logic, capacity constraints, and lack of adaptive control mechanisms. Traditional analytical models, such as those based on queuing theory, are often insufficient for capturing the dynamic and stochastic nature of elevator traffic.

1.3. Objectives

i. To develop a Discrete Event Simulation (DES) model that represents elevator usage and congestion in an office building during peak hours.
ii. To analyze and identify factors that contribute to elevator congestion during peak period such as morning arrival, lunch break and end of workday.
iii. To explore and test different elevator control strategies like optimized scheduling algorithms and grouping user by floor range to reduce congestion.

1.4. Scope of Study

This study focuses on simulating elevator operations and passenger behavior in an office building during peak usage periods using Discrete Event Simulation (DES). The simulation is developed in SimPy, a Python-based framework, to replicate realistic elevator traffic dynamics and evaluate system performance under varying demand conditions. The primary scope includes the following:

i. Modeling elevator operations in a 20-storey office building, including movement logic, capacity constraints, and demand-responsive scheduling.
ii. Simulating passenger behavior with randomized arrival rates, destination floors, patience thresholds, and group travel dynamics to reflect realistic congestion scenarios.
iii. Analyzing system performance through metrics such as average waiting time, elevator utilization, and passenger distribution across floors.
iv. Testing scheduling strategies such as floor-based grouping and skip-stop logic to assess their effectiveness in reducing congestion and improving service efficiency.

1.5. Limitations

To manage complexity and ensure feasibility, the simulation model is built on a set of assumptions and simplifications that, while necessary, may affect the realism and broader applicability of the results:

i. Synthetic Data Usage
The dataset used is synthetically generated to reflect expected elevator usage patterns. While it captures realistic behaviors, it may not fully represent the variability found in actual building operations.

ii. Simplified Passenger Behavior
Passenger agents follow uniform decision rules based on randomized patience levels and destination floors. Complex human behaviors such as urgency, social influence, or adaptive decision-making are not modeled.

iii. Single Elevator System
The simulation focuses on a simplified elevator system without advanced coordination between multiple elevators or priority-based scheduling.

iv. No Weight-Based Capacity Constraints
Elevator capacity is modeled based on passenger count alone, without considering weight limits or physical space constraints.

v. Limited Real-Time Adaptability
The model does not incorporate real-time sensor feedback or dynamic control systems that may exist in modern smart buildings.

Despite these limitations, the simulation provides a valuable framework for analyzing elevator performance and testing scheduling strategies under controlled conditions.

2.0 LITERATURE REVIEW

Discrete Event Simulation (DES) has emerged as an appealing method for modeling, predicting, and optimizing lift traffic flow in several constrained conditions. This literature review was used to examine existing research and theoretical backgrounds that are relevant in optimizing the lift system performance using DES. It provides the necessary findings to justify the research question and supports the methodology and objectives of this project.

a. Discrete Event Architecture for Elevator Simulation

Aladem and Al-Sharif (2014) introduce a modular lift simulator architecture that uses discrete-event scheduling to analyze traffic under different load circumstances. Their work is critical in laying the basis for simulation-based traffic analysis. ElevSim uses object-oriented principles to design events, entities, and simulation loops. The key focus is its ability to test custom group control algorithms, which provides a useful foundation for incorporating adaptive scheduling logic. The authors also highlight the bootstrapping mechanism required to start simulations in an empty state and show how lift control is treated as a centralized decision process, with lifts acting as slaves to a controller. This design is ideal for peak-hour analysis, when fine-grained control over events such as GENERATE_PASSENGER and CONTROLLER_UPDATE is required. Their key contributions from this work include:

- Modularity of simulation components (e.g. elevators, floors, controller),
- Capability to simulate up-peak, down-peak, and mixed traffic,
- Realistic use of elevator physics (door times, acceleration, jerk),
- Emphasis on flexibility and extensibility for future research.

Shi et al. (2024) use LSTM neural networks to estimate traffic flows based on past sensor data. While not a discrete event simulator in and of itself, this approach adds value to DES by allowing for data-driven prediction of passenger load at particular time periods. Their study employs 655 entries of dataset to train and validate LSTM models, with a focus on forecasting demand spikes during office hours. Although the study's aim is predictive rather than event-driven, it helps DES in two important ways:

- Dynamic passenger generation: LSTM can tell DES models when arranging arrival events.
- Real-time simulation adjustments: Adaptive control strategies can be included in DES based on expected demand patterns.

Lee et al. (2009) developed a mathematical model based on queuing theory for lift systems during peak periods. The use of semi-Markov processes and bulk-service queues enables the derivation of closed-form equations for performance metrics like:

- Passenger Waiting Time
- Journey time
- Round-trip time distributions
- Number of stops and service rate based on Poisson arrival assumptions.

Although this method lacks the visual and experimental richness of DES, it does provide a theoretical baseline against which simulated results can be validated. The model demonstrates how queue dynamics change with varied lift capacities and service patterns, informing important DES parameters like arrival rate and lift batch size. Apart of that, Hamdi and Mulvaney (1998) developed a visual interactive simulation to compare the performance of lift scheduling strategies using real-world data from UK high-rise buildings. This simulator integrates a passenger behavior model based on Poisson arrivals and destination selection with a scheduling algorithm evaluator. One important aspect of this system is its real-time data feedback mechanism, which allows the scheduler to adapt to shifting traffic patterns dynamically.

3.0 METHODOLOGY

3.1. Description of Process

This study uses Discrete Event Simulation (DES) in SimPy to model elevator operations during peak hours in a 20-storey office building. The simulation replicates how individual agents like passengers and elevators interact based on rule-based logic, including queue behavior, capacity constraints, and dynamic scheduling. The building layout is digitally represented with elevators serving all floors. Passenger agents are generated at the ground floor with randomized arrival rates around 10 to 15 per minute, destination floors, and patience thresholds. When wait times exceed their tolerance (patience threshold), agents will leave the queue and opt for stairways. Elevator agents operate with defined speed, capacity, and demand-responsive logic. They prioritize high-traffic floors and skip stops at full-capacity levels. Group travel behavior is included to reflect real-world boarding dynamics. The simulation outputs include total number of passengers, average waiting time, average travel distance per passenger, elevator utilization rates, and passenger distribution per elevator. While the visual outputs such as the peak hour passenger distribution chart and visited floors chart provide further insights into congestion patterns and support the evaluation of elevator scheduling strategies.

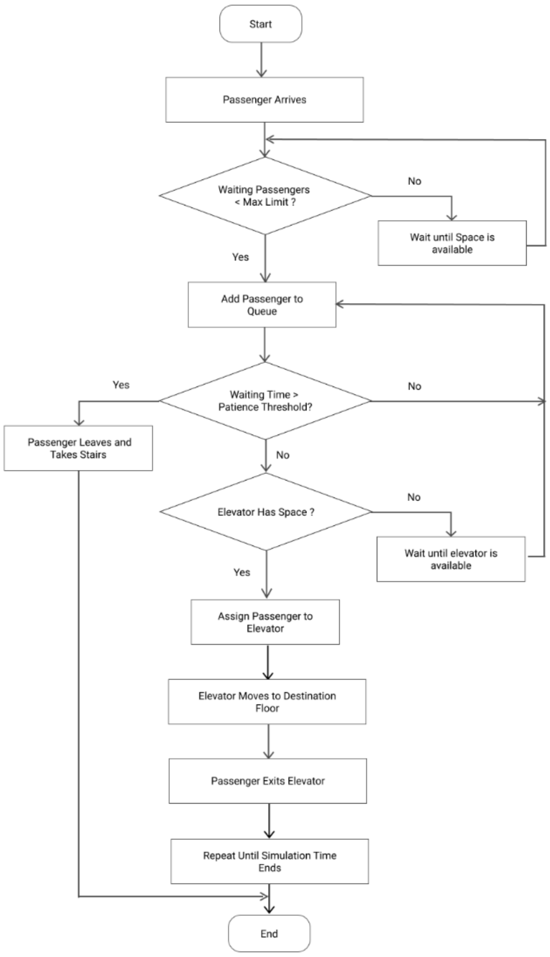

Figure 1. Simulation Flow

3.2. Data Collection

a. Dataset Source

The dataset used in this study is synthetically generated to replicate single elevator usage patterns in a 20-storey office building. It was designed to reflect realistic passenger behavior during peak and non-peak hours, including group travel, varying wait times, and floor-specific demand. The data spans from 08:00 to 17:57 and includes 206 entries, each representing a unique elevator trip request. The use of synthetic data was necessary due to the absence of publicly available real-world elevator usage logs with sufficient granularity. By generating the dataset based on expected behavioral patterns, we ensured that the simulation could capture key dynamics such as congestion, queue abandonment, and elevator scheduling efficiency under controlled and repeatable conditions.

b. Dataset Attributes

These attributes were selected to reflect the core variables influencing elevator system performance. Each attribute was intentionally designed to support the modeling of queuing behavior, elevator capacity management, and scheduling logic. The inclusion of both spatial (floor-based) and temporal (timestamp, waiting time) data ensures the simulation can capture realistic patterns of elevator demand and service efficiency.

Table 1. Data Attributes with Description

Attribute	***Description***
Timestamp	Time when the elevator request was made
Origin Floor and Destination Floor	Indicates the vertical movement of passengers
Number of Passengers	Group size per elevator trip
Waiting Time	Duration passengers waited before boarding

c. Pre-processing & Assumptions

Since the dataset is synthetic, it was constructed to align with the behavioral patterns intended for simulation. Pre-processing involved validating timestamp formats, ensuring floor values were within the building's range from 1 to 20, and checking for consistency in passenger counts and waiting times. It is assumed that all trips occurred within operational hours, and that waiting time reflects elevator availability and scheduling logic. Moreover, no missing values were present, and floor usage was aggregated to identify high-traffic zones.

3.3. Model Development

a. SimPy Implementation Details

The simulation model is implemented in SimPy, a process-based discrete-event simulation framework in Python. It defines two primary agent types:

Table 2. Data Attributes with Description

Agent Type	***Behavior Characteristics***	***Attributes***
Passenger	- Joins queue and waits for elevator - Leaves queue if waiting time exceeds patience - Boards elevator if capacity allows - Travels in groups, affecting load - Logs waiting time and assignment status	- Destination Floor - Patience Level - Arrival Time - Group Size
Elevator	- Moves between floors based on passenger destinations - Rejects boarding if capacity exceeded - Stops at requested floors - Logs active time and distance - Updates current floor and load	- Current Floor - Capacity (max passengers) - Speed (floors per second) - Active Time - Distance Travelled

b. Environment Setup

The simulation environment is designed to represent a 20-storey office building during peak operational periods specifically in the morning, during lunch breaks, and at the end of the working day. These time gaps were selected based on typical elevator usage patterns observed in high-rise office buildings, where demand flows due to employee arrivals, mid-day movement to common areas such as cafeterias, and mass departures at closing time. Modeling these periods allows the simulation to capture realistic congestion scenarios and evaluate elevator performance under varying load conditions.

At the start of the simulation, main parameters are initialized, including the number of elevators, number of floors, maximum passenger capacity, and total simulation duration. Each elevator is incorporate with attributes such as current floor, active time, distance travelled, and a passenger list. Passenger agents are generated dynamically throughout the simulation, with randomized destination floors and patience thresholds. These agents are added to a waiting list and logged for performance analysis.

Elevators operate within a bounded vertical space, moving between floors based on passenger destinations. Movement is time-based, with each floor traversal taking 1.5 minutes. Elevators update their current floor and operational metrics as they serve passengers. They stop at floors where passengers need to board or exit and may skip floors that are either full or not in demand.

The environment incorporates realistic constraints such as elevator capacity limits, queue abandonment behavior, and demand-responsive scheduling. Logs are maintained to track queue lengths, waiting times, and elevator utilization. This setup provides a strong framework for analyzing elevator performance and testing scheduling strategies under varying demand conditions throughout the workday. By simulating multiple peak periods, the model offers a broader understanding of elevator system behavior and supports the development of solutions that improve vertical mobility and user experience in office buildings.

c. Pseudocode

```
START

 Initialize simulation parameters:
  Number of elevators
  Number of floors
  Max passengers
  Simulation duration

 Create elevator objects:
  Set current floor
  Initialize passenger list
  Set active time and distance travelled

 Initialize logs and data containers:
```

```
  - Passenger log
  - Waiting times
  - Assignment logs
  - Queue length tracker

Define function to generate passenger:
  - Randomly assign destination floor
  - Randomly assign patience level

Define function to add passengers:
  WHILE waiting passengers < max limit:
    Generate passenger
    Add to waiting list
    Log passenger details
    Print status

Define function to assign passengers to elevators:
  FOR each elevator:
    WHILE elevator has space AND passengers are waiting:
      Assign passenger to elevator
      Calculate and log waiting time
      Log assignment
      Print status

Define function to remove passengers from elevator:
  FOR each passenger in elevator:
    IF passenger destination == current floor:
      Remove passenger
      Print offboarding status

Define function to move elevator:
  IF target floor == current floor:
    Print already on floor
  ELSE IF target floor > current floor:
    FOR each floor up to target:
      Wait 1.5 minutes
      Update active time and distance
      Print arrival
  ELSE:
    FOR each floor down to target:
      Wait 1.5 minutes
      Update active time and distance
      Print arrival
  Update elevator current floor

Define elevator operation process:
  WHILE simulation is running:
    IF elevator has passengers:
      Sort destination floors
      FOR each destination:
        IF simulation time exceeded:
```

```
      BREAK
     Move elevator to floor
     Remove passengers
   ELSE:
    Wait 1 minute

Define main elevator system process:
 Print welcome message
 WHILE simulation is running:
   Wait for next passenger arrival
   IF waiting passengers < max:
    Add passengers
    Assign passengers to elevators
                             Log queue length

Start elevator processes
Start system process
Run simulation until duration ends

END
```

3.4. Analysis of Results

a. Simulation Output

```
--- Simulation Summary ---
Total number of passengers during simulation: 341
Average waiting time: 9.17 minutes
Average travel distance per passenger: 2.74 floors
Elevator Utilization Rates:
  Elevator 1: 98.44%
  Elevator 2: 97.50%
  Elevator 3: 96.25%
Passenger Distribution per Elevator:
  Elevator 1: 119 passengers
  Elevator 2: 111 passengers
  Elevator 3: 101 passengers
```

Figure 2. Simulation Results Summary

Based on *Figure 2*, the simulation successfully handled a total of 341 passengers across three elevators during peak hours. The average waiting time recorded was 9.17 minutes, which is relatively high and indicative of congestion during periods of elevated demand. This delay is primarily due to frequent elevator stops and short-distance trips, as the average travel distance per passenger was only 2.47 floors. These short trips increase the number of stops per journey, contributing to longer waiting times for subsequent passengers. Furthermore, elevator utilization rates were consistently high, with Elevator 1 operating at 98.44%, Elevator 2 at 97.50%, and Elevator 3 at 96.25%. Passenger distribution was also balanced, with Elevator 1 serving 119 passengers, Elevator 2 serving 111, and Elevator 3 serving 101. This reflects effective load sharing and suggests that the scheduling logic implemented in the simulation was successful in distributing demand evenly across the available elevators. However, the elevated waiting time highlights areas for improvement, such as optimizing dispatch algorithms or adjusting elevator speed to reduce delays.

These results demonstrate that the elevator scheduling system worked effectively under the given demand, maintaining a fair balance between utilization and passenger distribution. Unfortunately, the slightly high waiting time suggests that further optimization, such as improved dispatching algorithms or adjusting elevator speed, could enhance performance.

b. Observation

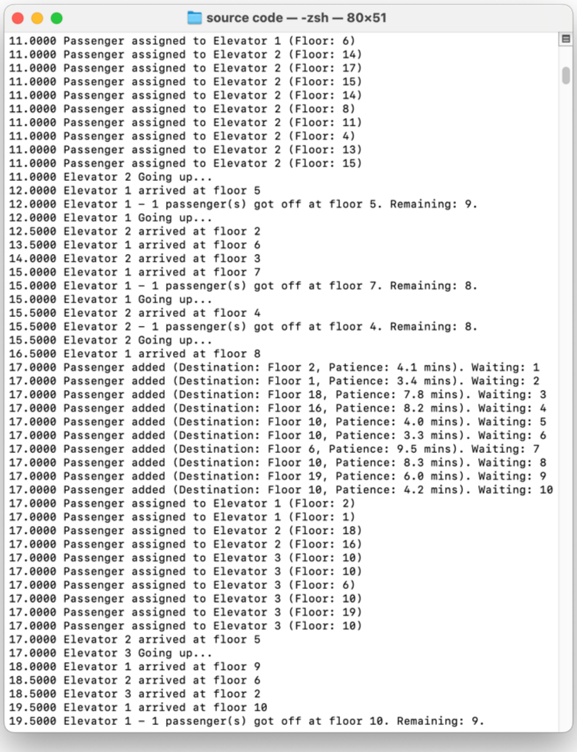
```
source code — -zsh — 80×51
11.0000 Passenger assigned to Elevator 1 (Floor: 6)
11.0000 Passenger assigned to Elevator 2 (Floor: 14)
11.0000 Passenger assigned to Elevator 2 (Floor: 17)
11.0000 Passenger assigned to Elevator 2 (Floor: 15)
11.0000 Passenger assigned to Elevator 2 (Floor: 14)
11.0000 Passenger assigned to Elevator 2 (Floor: 8)
11.0000 Passenger assigned to Elevator 2 (Floor: 11)
11.0000 Passenger assigned to Elevator 2 (Floor: 4)
11.0000 Passenger assigned to Elevator 2 (Floor: 13)
11.0000 Passenger assigned to Elevator 2 (Floor: 15)
11.0000 Elevator 2 Going up...
12.0000 Elevator 1 arrived at floor 5
12.0000 Elevator 1 – 1 passenger(s) got off at floor 5. Remaining: 9.
12.0000 Elevator 1 Going up...
12.5000 Elevator 2 arrived at floor 2
13.5000 Elevator 1 arrived at floor 6
14.0000 Elevator 2 arrived at floor 3
15.0000 Elevator 1 arrived at floor 7
15.0000 Elevator 1 – 1 passenger(s) got off at floor 7. Remaining: 8.
15.0000 Elevator 1 Going up...
15.5000 Elevator 2 arrived at floor 4
15.5000 Elevator 2 – 1 passenger(s) got off at floor 4. Remaining: 8.
15.5000 Elevator 2 Going up...
16.5000 Elevator 1 arrived at floor 8
17.0000 Passenger added (Destination: Floor 2, Patience: 4.1 mins). Waiting: 1
17.0000 Passenger added (Destination: Floor 1, Patience: 3.4 mins). Waiting: 2
17.0000 Passenger added (Destination: Floor 18, Patience: 7.8 mins). Waiting: 3
17.0000 Passenger added (Destination: Floor 16, Patience: 8.2 mins). Waiting: 4
17.0000 Passenger added (Destination: Floor 10, Patience: 4.0 mins). Waiting: 5
17.0000 Passenger added (Destination: Floor 10, Patience: 3.3 mins). Waiting: 6
17.0000 Passenger added (Destination: Floor 6, Patience: 9.5 mins). Waiting: 7
17.0000 Passenger added (Destination: Floor 10, Patience: 8.3 mins). Waiting: 8
17.0000 Passenger added (Destination: Floor 19, Patience: 6.0 mins). Waiting: 9
17.0000 Passenger added (Destination: Floor 10, Patience: 4.2 mins). Waiting: 10
17.0000 Passenger assigned to Elevator 1 (Floor: 2)
17.0000 Passenger assigned to Elevator 1 (Floor: 1)
17.0000 Passenger assigned to Elevator 2 (Floor: 18)
17.0000 Passenger assigned to Elevator 2 (Floor: 16)
17.0000 Passenger assigned to Elevator 3 (Floor: 10)
17.0000 Passenger assigned to Elevator 3 (Floor: 10)
17.0000 Passenger assigned to Elevator 3 (Floor: 6)
17.0000 Passenger assigned to Elevator 3 (Floor: 10)
17.0000 Passenger assigned to Elevator 3 (Floor: 19)
17.0000 Passenger assigned to Elevator 3 (Floor: 10)
17.0000 Elevator 2 arrived at floor 5
17.0000 Elevator 3 Going up...
18.0000 Elevator 1 arrived at floor 9
18.5000 Elevator 2 arrived at floor 6
18.5000 Elevator 3 arrived at floor 2
19.5000 Elevator 1 arrived at floor 10
19.5000 Elevator 1 – 1 passenger(s) got off at floor 10. Remaining: 9.
```

Figure 3. Simulation Log

The Simulation logs in *Figure 3* shows that that elevators were actively responding to passenger requests and maintaining continuous movement between floors. Passengers were assigned promptly, and elevator operations were consistent. However, during peak intervals especially when multiple requests occurred simultaneously some passengers experienced delays of several minutes before boarding. This indicates that while the system was responsive, it became strained under high load conditions, leading to longer trips and more frequent stops.

The Passenger Arrival per Hour chart in *Figure 4* reveals a clear peak in passenger arrivals at Hour 0, which corresponds to 8:00 AM in the simulation's time mapping. Approximately 68 passengers arrived during this hour, nearly double the number observed in subsequent hours. After this peak, the number of arrivals dropped and fluctuated between 30 and 50 passengers throughout the day. This pattern reflects a realistic office environment, where the majority of elevator usage occurs during the morning rush, followed by moderate and occasional usage assuming due to meetings, lunch breaks, or individual movement between floors. While the floor visitation chart shown in *Figure 5,* reveals that Floor 1 where the lobby area was located, had the highest number of visits, followed by mid-level floors that likely host key offices or facilities. These patterns suggest that elevator scheduling strategies should be time-sensitive and floor-aware to better manage fluctuating demand throughout the day.

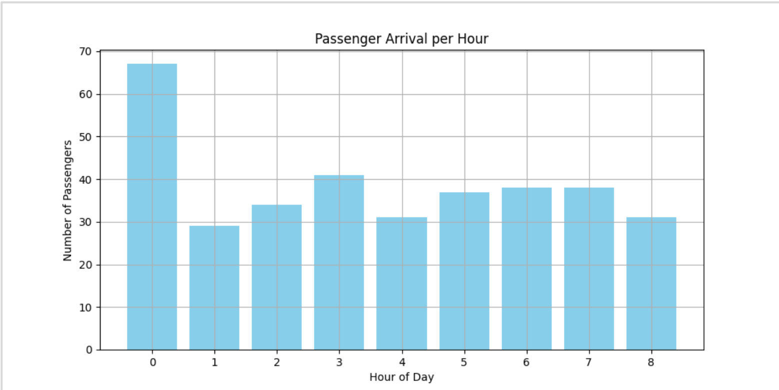

Figure 4. Passenger Arrival per Hour Chart

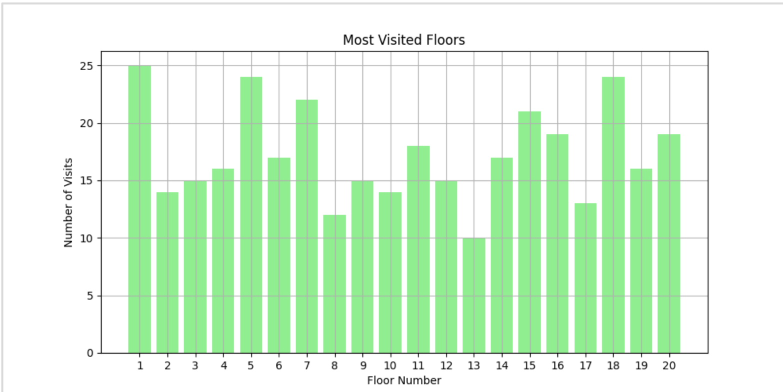

Figure 5. Most Visited Floor Chart

Additionally, the simulation produces slightly different results with each run due to the randomized nature of passenger arrivals, destination floors, and patience thresholds. This variability is intentional and reflects the stochastic behaviour of real-world elevator systems, where passenger flow and elevator usage patterns are never exactly the same. As such, the simulation provides a more realistic and flexible framework for evaluating elevator performance under dynamic conditions.

To add, passengers with lower patience thresholds were observed to abandon the queue during peak hours, slightly reducing congestion but also indicating potential dissatisfaction. This behaviour reinforces the need for responsive scheduling systems that minimize wait times and improve service reliability. The simulation also showed that during non-peak periods, average waiting times dropped below 5 minutes, demonstrating the system’s ability to recover when traffic is less intense.

c. Observation and Assumption with 'What-If' Scenario

Although the simulation was not explicitly configured to run with fewer elevators, a hypothetical scenario was considered to evaluate how system performance might change if only two elevators were available instead of three. This assumption is based on the observed behavior of the system under full capacity and aims to explore potential constraints during peak demand.

In the baseline simulation, three elevators operated with utilization rates between 96% and 98%, and the average waiting time was recorded at 9.17 minutes. If one elevator were removed from service, it is reasonable to assume that the remaining two elevators would reach full utilization (100%) and experience increased strain. Under such conditions, the average waiting time could rise to approximately 12 minutes or more, particularly during the morning peak period when passenger arrivals are highest. This assumption is further supported by the simulation's passenger arrival pattern, which showed a significant surge in demand during the early hours of the day. With only two elevators available, the system would likely struggle to accommodate this arrival efficiently, resulting in longer queues, more frequent stops, and extended travel durations. The increased load per elevator would also lead to more complex routing and delayed service for passengers traveling to higher or less frequently visited floors.

While this scenario was not implemented in the simulation, it provides a useful reference for evaluating system resilience and planning for contingencies such as maintenance downtime or emergency conditions. It also reinforces the importance of maintaining sufficient elevator capacity and developing adaptive scheduling strategies that can respond to fluctuating demand and resource availability.

4.0 DISCUSSION

4.1 Project Reflection

This project was a complete study in applying theoretical simulation techniques to a real-world urban infrastructure challenge. The team emphasized the need to incorporate human behavior patterns into simulation design, such as queue abandoning and group travel. Rather than using static or deterministic logic, the initiative embraced randomness and variability, which are more accurate representations of real-world dynamics. Furthermore, the separation of duties like model design, coding, analytics, and documentation enabled the team to manage complexity through specialization, demonstrating the value of collaborative development in simulation projects. SimPy and other tools provided high modularity but also required disciplined event management, making this a good lesson on the balance between abstraction and control in agent-based modelling.

4.2 SimPy Simulation Insights

The SimPy framework helped the team create a dynamic and realistic simulation of lift operations. One technical problem was synchronizing lift status updates so that timing conflict did not occur when numerous people attempted to board or exit at the same time, which necessitated precise sequencing of yield and event definitions. Key findings from the implementation include:

- Process-based Modelling Efficiency: SimPy's asynchronous agent behavior enabled the natural portrayal of passenger and lift lifecycles. It streamlined concurrent events like arrival, waiting, boarding, and lift movement.
- Patience Thresholds: The simulation revealed that when patience levels were set at random, more realistic abandoning behavior emerged, artificially lowering system congestion while exposing potential unhappiness.
- Resource Monitoring: SimPy's event tracing made it easier to track lift performance indicators like idle time, trip count, and passenger load without the need for external monitoring tools.
- Scalability and flexibility: The modular architecture allows for future improvements such as multi-lobby systems, emergency evacuation logic, and AI-based scheduling.

4.3 Key Findings and Interpretations

The simulation yielded numerous critical discoveries that can be applied to improve the building's lift system. First, it confirmed that the morning rush hour, notably from 8:00 to 9:00 a.m., is the most critical time. During this time, lifts are typically at capacity, particularly on the ground floor as they travel to mid-level floors such as the sixth story. The simulation also demonstrated that some passengers elected to leave the queue, resulting in individuals taking the stairs instead. While this helped to ease some congestion and enhance the system's overall throughput, it also highlights a substantial service satisfaction issue that must be addressed. Furthermore, the simulation revealed that groups of individuals travelling to the same floor caused additional delays. This shows that clustering passengers by floor range may assist balance the load and enhance efficiency. Finally, the simulation evaluated a demand-responsive scheduling system, which was found to be effective in reducing wait times and increasing overall flow by skipping low-demand or recently served floors. However, if not adequately adjusted, this strategy raises issues about the fairness of higher-floor passengers. These findings collectively indicate that no single scheduling strategy is adequate for all traffic scenarios. A more effective option would be a hybrid or context-aware algorithm that can adjust to changing traffic patterns throughout the day.

4.4 Model Validity and Realism

The simulation's architecture ensures a high level of authenticity and realism, establishing a realistic baseline for future research. This is largely owing to the implementation of an agent-based model, in which both passengers and lifts were assigned realistic properties such as patience levels, floor destinations, and group sizes. This technique accurately replicates the natural behaviours observed in real-world lift use. Moreover, the results support the model's accuracy. The output graphs, particularly those showing floor visitation and hourly arrival rates, closely resemble the traffic patterns prevalent in high-rise office buildings. Although the model used synthetic data due to a lack of real-world information, the dataset was based on well-founded, realistic assumptions regarding working hours, group sizes, and how workers arrange themselves across floors in a normal office environment. Despite its features, the simulation has some known drawbacks. It lacked advanced multi-lift coordination logic, as well as emergency event and priority systems modelling capabilities. Furthermore, it did not take weight limits into account when calculating capacity based on passenger count. Despite these restrictions, the simulation provides a solid, realistic framework for future upgrades or the development of real-time simulations.

4.5 Project Experience

This project has adopted strong collaboration among team members, each contributing different expertise in simulation modeling, coding, data analysis, and documentation. The team worked together to develop a Discrete Event Simulation (DES) model that realistically captured elevator operations during peak hours in an office building. A major challenge was designing agent behaviors particularly for passengers and elevators within SimPy's event-driven framework. Synchronizing elevator movements and passenger interactions required careful sequencing to avoid race conditions and ensure accurate simulation flow. The team successfully implemented logic for queue abandonment, group travel, and demand-responsive scheduling, which added realism to the model. Data preparation was another critical aspect. Although the dataset was synthetic, it was carefully constructed to reflect realistic elevator usage patterns. The team ensured data integrity through preprocessing steps and used the dataset to identify congestion hotspots and evaluate scheduling strategies. The project deepened the team's understanding of agent-based modeling and the practical application of DES in urban infrastructure. Rather than adapting an existing open-source simulation, the team successfully built a custom model from scratch, tailored to the specific context of office elevator traffic. This hands-on experience highlighted the value of simulation as a decision-support tool, offering insights that could inform real-world elevator system design and optimization.

5.0 CONCLUSION

This study successfully developed and implemented a Discrete Event Simulation (DES) model to analyze elevator congestion during peak hours in a multi-story office building. By simulating realistic passenger behaviors and elevator operations using SimPy, the project identified key factors contributing to congestion, such as short-distance trips, group travel, and uneven floor demand. The simulation demonstrated that demand-responsive scheduling can improve system throughput and reduce waiting times, although it must be carefully tuned to ensure fairness across all floors.

The findings highlight the importance of time-sensitive and adaptive scheduling strategies, especially during high-demand periods like morning arrivals. The model also showed that queue abandonment behavior, while reducing congestion, may negatively impact user satisfaction. Although the simulation used synthetic data, its structure and logic were grounded in realistic assumptions, making it a valuable tool for evaluating elevator performance and testing control strategies.

For future improvements, the model could incorporate weight-based capacity limits, emergency evacuation scenarios, and multi-elevator coordination to enhance realism and applicability. Overall, this project demonstrates the potential of simulation as a decision-support tool for building managers and system designers seeking to optimize vertical transportation in commercial environments.

REFERENCES

Xiang, S., Arashpour, M., & Wakefield, R. (2019). A simulation model for investigation of operation of elevator's up-peak. *Journal of Simulation*, *14*(3), 229–238. https://doi.org/10.1080/17477778.2019.1604465

Michaelmarsillo. (n.d.). *GitHub - michaelmarsillo/ElevatorSimulationGame: Simple Elevator Simulation that i made in python*. GitHub. https://github.com/michaelmarsillo/ElevatorSimulationGame/tree/main

Shi, M., Sun, E., Xu, X., & Choi, Y. (2024). Prediction and Analysis of Elevator Traffic Flow under the LSTM Neural Network. *Intelligent Control and Automation*, *15*(02), 63–82. https://doi.org/10.4236/ica.2024.152004

Lee, Y., Kim, T. S., Cho, H., Sung, D. K., & Choi, B. D. (2008). Performance analysis of an elevator system during up-peak. Mathematical and Computer Modelling, 49(3–4), 423–431. https://doi.org/10.1016/j.mcm.2008.09.006

Jiemingyou. (n.d.). *GitHub - jiemingyou/elevator-simulation: Elevator simulation written in Python*. GitHub. https://github.com/jiemingyou/elevator-simulation?tab=readme-ov-file

APPENDICES

[1] Project Repository

Contains all related files, documentation, and resources for the simulation project.
https://github.com/farahfuaad/DES-of-Office-Elevator-Congestion-During-Peak-Hours-with-SimPy

[2] SimPy Source Code

Python script implementing the elevator simulation model using SimPy
https://github.com/farahfuaad/DES-of-Office-Elevator-Congestion-During-Peak-Hours-with-SimPy/blob/main/source%20code/elevator_traffic.py

[3] Dataset

Synthetic dataset used to simulate elevator usage patterns during peak and non-peak hours.
https://github.com/farahfuaad/DES-of-Office-Elevator-Congestion-During-Peak-Hours-with-SimPy/blob/main/dataset/elevator_usage.csv