A simulation tool for vertical transportation systems using python

Conference Paper · May 2018

DOI: 10.1109/ICBIR.2018.8391205

Pruk Sasithong
Chulalongkorn University

Sanika Krishnamali Wijayasekara
Chulalongkorn University

W. Asdornwised
Chulalongkorn University

L. Wuttisittikulkij
Chulalongkorn University

A Simulation Tool for Vertical Transportation Systems using Python

Nathaporn Chaosangket
Faculty of Electrical Engineering
Chulalongkorn University
Bangkok, Thailand
nathaporn.chao@gmail.com

Pruk Sasithong
Faculty of Electrical Engineering
Chulalongkorn University
Bangkok, Thailand
pruk.sasithong@gmail.com

Sanika K. Wijayasekara
Faculty of Electrical Engineering
Chulalongkorn University
Bangkok, Thailand
wijayasekarasanika@gmail.com

Widhyakorn Asdornwised
Faculty of Electrical Engineering
Chulalongkorn University
Bangkok, Thailand
widhyakorn.a@chula.ac.th

Lunchakorn Wuttisittikulkij
Faculty of Electrical Engineering
Chulalongkorn University
Bangkok, Thailand
wlunchak@chula.ac.th

Pisit Vanichchanunt
Faculty of Engineering
King Mongkut's University of Technology
North Bangkok, Bangkok, Thailand
v_pisit@hotmail.com

Muhammad Saadi
Department of Electrical Engineering
University of Central Punjab
Lahore, Pakistan
muhammad.saadi@ucp.edu.pk

***Abstract*—This paper presents a software development tool which is capable of simulating the vertical transportation systems within buildings using elevators. The simulation tool is composed of a passenger arrival model for a single or group of elevators, their control system, and a graphical user interface (GUI). The developed GUI can display statistical information of the passengers traffic pattern and also the animation of elevator cars with the number of passengers inside. The current version allows both individual and batch Poisson arrival processes which could represent up-peak traffic in the morning and after lunch of office buildings respectively. The motion of elevator cars has been precisely modeled with real parameters of velocity, acceleration, distance, and jerk. However, only the basic nearest car algorithm is implemented for elevator group control systems. We believe, the developed simulator not only has its utility for educational purpose but also can serve as a software platform for future research on more advanced elevator group control systems.**

***Keywords*—vertical transportation, elevator group control, simulation tool, poison batch, kinematic**

I. INTRODUCTION

A skyscraper refers to the high-rise building which is taller than 100 m. Bangkok ranks 14 in the world in terms of the number of skyscrapers in a metropolitan city which makes Bangkok ahead of London and Seoul [9]. Vertical transportation refers to the means of travelling between floors in a building which includes elevators, escalators, stairs and etc. In the high rise buildings, elevators are by far the most popular choice by the commuters, because they provide fast, economical and convenient movement from one floor to another. Skyscrapers usually have multiple elevators which are designed according to the logistic requirements, however, as per the needs of modern era, these elevators need to be energy efficient, user friendly and time efficient. In literature, various studies have been conducted on their design, energy efficiency and cost effectiveness [1], [2], [3], [4].

In this paper, we developed a simulation tool in Python which simulates a vertical transportation system. The simulation model is potentially capable of simulating various scheduling algorithms. In order to evaluate the performance of the elevators which are in-line with the real elevator's motion, elevator's kinematic variables are incorporated into the simulation model. For the users to experience the visualization, a graphical user interface (GUI) is developed. The developed tool will help understanding the basic principles of how a real elevator's work in an ease to catch manner and serve as a starting point for further research and development.

The rest of the paper is organized as follows. In Section II, the elevator components like arrival model, control model and elevator kinematics are discussed. In Section III, the methodology of our work is described in detail. In Section IV, results and discussion are given and the paper is concluded in Section V.

II. ELEVATOR COMPONENTS

In this section, we discuss the three main important components for elevator which are the arrival model, a well known nearest car algorithm and elevator kinematics [1], [5]. Moreover, some important components of the group of elevator systems; such as, elevator velocity, arrival time, distance, jerk, and accelerator are discussed.

A. Arrival Model

The most critical component in elevator simulation is the arrival model. In order to make this study comprehensive, two

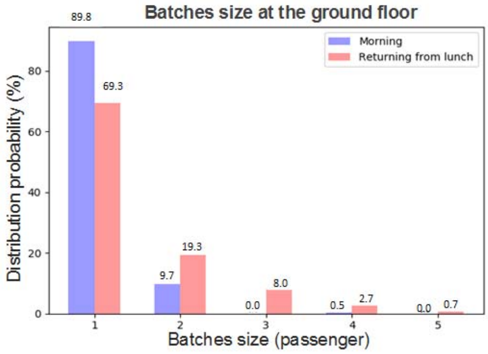

Fig. 1 The relationship between the batch size and probability in the morning and after lunch time at an Office building.

arrival models are investigated which are individual Poisson arrival model and batch Poisson arrival model [6]. Up-peak traffic is considered for this study which is the period in which the passengers move up from the ground floor and this up-peak traffic is analyzed in two periods i.e. in the morning and after lunch time. As shown in Fig. 1, the tendency of arrival in the morning is like an individual Poisson arrival process because most people used to go to their offices independently. However, during the lunch time, the distribution of the arrival process are more like a batch Poisson arrival process [7], since the people tend to go for lunch in groups. The batch Poisson arrival rate (λ_{batch}) can be expressed as follows.

$$\lambda_{batch} = \frac{\lambda}{b_{avg}} \tag{1}$$

where λ is the total arrival rate and b_{avg} is the average batch size, which is defined as the average number of passengers in each batch.

B. Nearest Car

The nearest car (NC) control algorithm [5] is a widely known as a basic algorithm in elevator group control systems. The position of the elevator, direction of the landing call and direction of the elevator are the three main factors which used in this algorithm. Figure of Suitability (FS) is a function of above three factors. Primarily, FS is a numerical value which is used for elevator selection upon user request. Higher the value FS is, more likely the elevator will be selected to serve a passenger. When there are more than one elevators with the same FS, the nearest elevator will be assigned to serve a passenger. In case of new arrival, four rules are checked to calculate the FS value which are listed below:

- **Rule 1:** If the elevator is moving towards the landing call and in the same direction as the landing call then $FS = (N + 2) - d$.
- **Rule 2:** If the elevator is moving towards the landing call but in the opposite direction to the landing call then $FS = (N + 1) - d$.

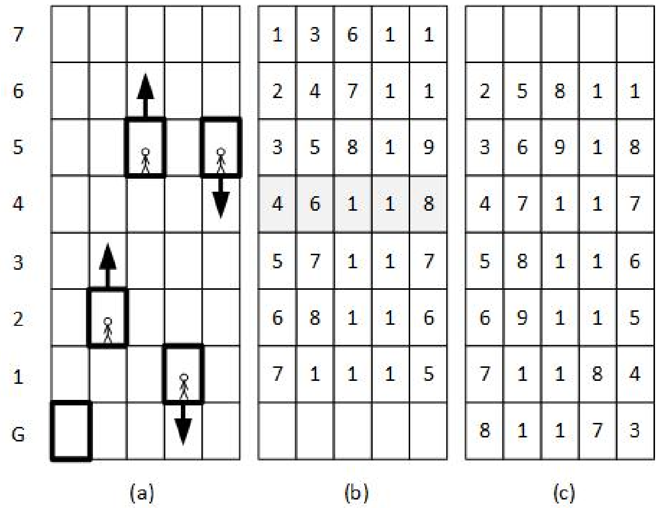

Fig. 2 Execution example of NC algorithm.

- **Rule 3:** If the elevator is not moving (IDLE) then $FS = (N + 1) - d$.
- **Rule 4:** If the elevator is moving away from the landing call then $FS = 1$.

Where d is the distance between the landing call and the present position of the elevator and N is the number of floors in the building.

Fig. 2 presents the scenarios with eight stories building and five operational elevators to illustrate the process of the elevator selecting procedure. In Figure 2b and Figure 2c, the calculated FS values are used for elevator selection against every possible call. Figure 2b considers the scenario of descending landing call from every floor except the ground floor while Figure 2c considers the scenario of ascending landing call from every floor except the top floor. For example, assume a passenger from the fourth floor wants to go down to the first floor. The FS values in the fourth row of Figure 2b are calculated based on the descending landing call for each elevator. As a result, the elevator which has the highest FS value will be selected as the serving elevator for this request. For this case, the 5^{th} elevator will be assigned to this passenger request. The values of FS against this call are calculated as follows:

- **Elevator 1:** is not moving (IDLE), we apply Rule 3. Since $d = 4$ floors, we obtain $FS = (7 + 1) - 4 = 4$.
- **Elevator 2:** is moving up (different direction) towards the landing call. Here, we apply Rule 2. Since $d = 2$, we obtain $FS = (7 + 1) - 2 = 6$.
- **Elevator 3:** is moving up (different direction) away from the landing call therefore Rule 4 is applied. So, $FS = 1$.
- **Elevator 4:** is moving down (different direction) away from the landing call, we apply Rule 4. Hence, $FS = 1$.
- **Elevator 5:** is moving down (same direction) towards the landing call, therefore we apply Rule 1. When $d = 1$, $FS = (7 + 2) - 1 = 8$.

As a result, the 5^{th} elevator has the maximum FS value ($FS = 8$) compared to other elevators therefore, the 5^{th} elevator is assigned to pick the descending landing call passenger from the 4^{th} floor.

C. Elevator Kinematics

The elevator kinematic notations used in this work are given in Table I. In elevator kinematic calculations, four parameters are considered, i.e., traveling distance, velocity, acceleration, and jerk. Jerk is the derivation of the acceleration that can affect feeling of passengers in the elevator and is considered as an unfavorable action in elevator transportation. For example, if the elevator is assigned to move up and the acceleration changes instantly, then a big jerk will be felt by the passengers in an elevator. This feeling will make them uncomfortable. Modern elevators require jerk to be as small as possible in order to have smooth elevator operation. Depending on the relative values of these four parameters, four different cases can be classified into three categories [8]. It should be noted that the case where the elevator reaches the maximum velocity before the maximum acceleration is usually not considered for as it would not be a sensible design [5].

1) The elevator reaches maximum velocity and maximum acceleration.
2) The elevator reaches maximum acceleration but does not reach maximum velocity.
3) The elevator does not reach maximum velocity and maximum acceleration.

As a result of the above three conditions and mathematical derivation in [8], the analytic equations of traveling distance $d(t)$, velocity $v(t)$, acceleration $a(t)$, and Jerk $j(t)$ as functions of traveling time can be stated belows:

1) Elevator reaches both maximum velocity and maximum acceleration. If $d \geq \frac{a^2v+v^2j}{ja}$, then the traveling time $t = \frac{d}{v} + \frac{a}{j} + \frac{v}{a}$.
2) Elevator reaches maximum acceleration but does not reach maximum velocity. If $\frac{2a^8}{j^2} \leqslant d < \frac{a^2v+v^2j}{ja}$, then the traveling time $t = \frac{a}{j} + \frac{\sqrt{a^3+4dj^2}}{\sqrt{aj}}$.
3) Elevator does not reach both the maximum velocity and maximum acceleration. If $d < \frac{2a^8}{j^2}$, then the traveling time $t = \sqrt[3]{\frac{32d}{j}}$.

In Fig. 3, the distance, velocity, acceleration, and jerk as functions of time t are plotted for all of these three cases by the elevator kinematic equations.

III. METHODOLOGY

A. Proposed Simulation Model

There are three main states of the elevator:

- IDLE - Each elevator does not move up, down, and has no request to handle.
- STOPPING - Each elevator stops at the destination floor to pick up or drop the passengers. In this state, there are four sub states which are opening the door, unloading the passengers, loading the passengers and closing the door.

TABLE I: List of the symbols for kinematic calculating.

Symbols	Definition
d	traveling distance (m)
D(t)	travelling distance at time t (m)
v	maximum velocity (m/s)
V(t)	velocity at time t (m/s)
a	maximum acceleration (m/s^2)
A(t)	acceleration at time t (m/s^2)
j	maximum jerk (m/s^3)
J(t)	jerk at time t (m/s^3)

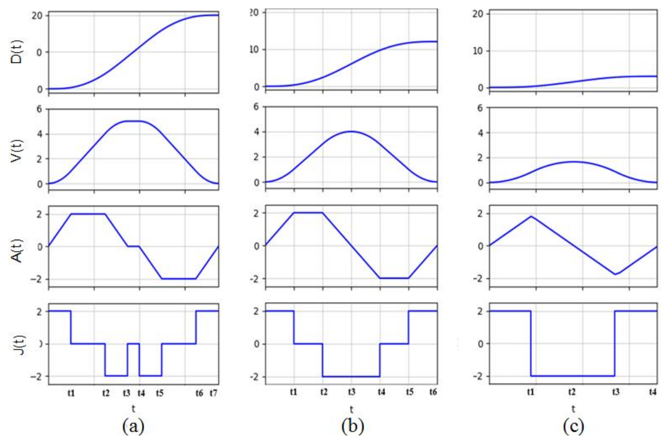

Fig. 3 (a) Elevator reaches maximum velocity and acceleration (b) Elevator reaches maximum acceleration, but does not reach maximum velocity (c) Elevator does not reach maximum velocity and acceleration [8].

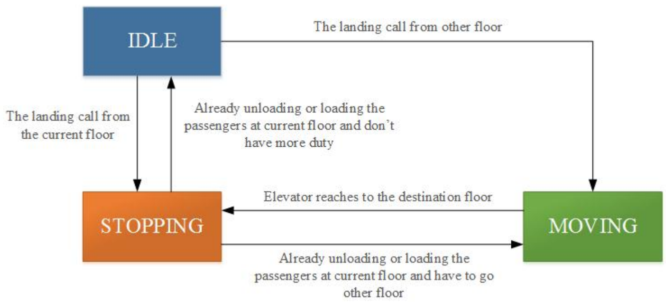

Fig. 4 Relationship among the states of elevator.

- MOVING - Each elevator is in motion (up or down).

As shown in Fig. 4, each elevator state will be changed according to the passenger request.

There are seven main processes in our Python simulation model coding:

1) **Passenger Arrival** - generate passengers into the system with the arrival time, landing call, and destination to go. In particular, passenger arrival time is Poisson distributed. Uniform probability distribution is generated for landing call and destination to go.

2) **Passenger** - collects the data of each passenger which contains the arrival time, departure time, time to get into the elevator, arrival floor, and destination floor.
3) **Waiting Area** - create the passengers when the arrival time equals to system time and then assigns them to the waiting area by separating them into two cases based on the requested direction; Ascending Landing Call (Up) or Descending Landing Call (Down).
4) **ALift** - collects all the data of each elevator that contains the state, moving case, position, number of passengers in the elevator, and the duties for picking up the passenger.
5) **Scheduler** - assign the actions to the elevator group control system.
6) **State** - update the state for all elevators.
7) **Animation** - shows how the system work.

The simulation starts when the passengers are arriving and waiting at the waiting areas. When the arrival time of passenger equals to the system time, **Scheduler** will get the elevator status from **ALift** and decide whether to assign any one of the elevators to serve the passenger requests or not. Next, the system will update all of the elevators state in order to continue for the other requests. This will continue until the end of the simulation time. Fig. 5 shows the illustrated flow chart of our simulation procedure.

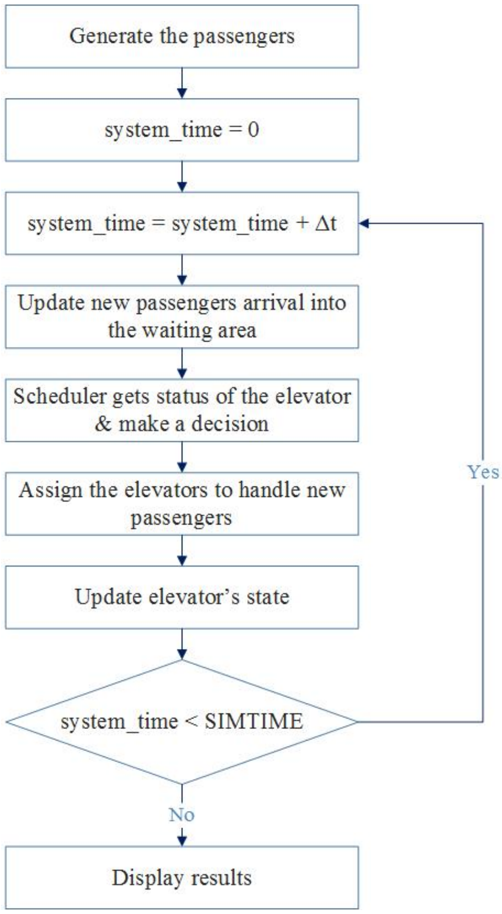

Fig. 5 Flow chart of our coding procedure.

IV. SIMULATION RESULTS AND DISCUSSION

As mentioned before in Section II, first, we will present the arrival results of the scenarios considered for both individual Poisson arrival and batch Poisson arrival processes as denoted in Equation (1). These are the scenarios which the passengers arrive before/after and during the lunch time, respectively. The simulated passenger arrival rate is set at 0.4 passengers per second with 1800 seconds of simulation time. The resultant graph is shown in the Fig. 6. As depicted in Fig. 6, for the longer period of time, the number of passengers in both models increase correspondingly. As presented in the magnified view, both models give reasonable results, this way, our simulation is realistic enough to identify the real traffic model in any scenarios.

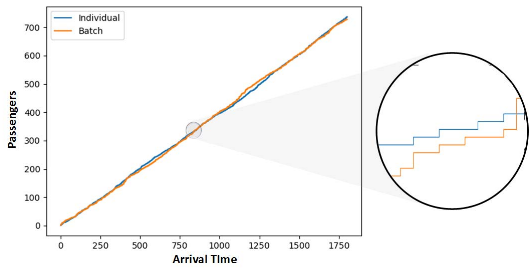

Fig. 6 Our simulation results individual and batch Poisson models.

The specification of our simulation is shown in Table II. The elevator group control system we study is a commercial buildings (offices) with 20 floors and 6 elevators. In the simulation, we study the passenger arrivals in the morning (Up-peak Traffic) at each floor as a Poisson distribution and simulated it for 3600 seconds of simulation period. The simulation platform interface is shown in Fig.7. It shows the real operation which elevators will move up or down to pick up the passengers from each floor according to their landing calls and bring them to the destination floors. The display represents the information of the number of passengers in the waiting area along with the direction that they need to move and the elevators in operations. The light pink colored columns show the route of each elevator car where the moving boxes represent the elevator cars. In order to show the movement direction of the elevator cars, the edge colors of elevator cars are changed to green or red to indicate up and down directions respectively. The first two columns displayed outside the main box shows that which elevator is assigned to pick up the passengers of each floor.

Simulation results of our experimatial elevator group system are measured with the following performance metrics:

- **Waiting time:** is the time interval between the passenger arrival and entering to the elevator.
- **Service time:** is the time interval between the passenger entering to the elevator and departing from the elevator.

TABLE II: Specifications of elevator simulation.

Items	Value
Number of Floors	20
Number of Elevator	6
Floor Distance (m)	4
Maximum Velocity (m/s)	2.5
Maximum Acceleration (m/s^2)	1
Maximum Jerk (m/s^3)	2
Capacity (person/elevator)	24
Door Time (s)	1.5
Passenger Time (s)	1.5

- **Total time:** is the passenger time spent in the system (Waiting time + Service time).

Results in Fig. 8 illustrates that the waiting time increases with the arrival rate. This means that as passengers arrive at higher rates, they have to wait for longer period before an elevator is available to pick them up. According to [10] the building is designed to have a good elevator service, the average waiting time should not exceed 25 seconds. Thus, the maximum allowed arrival rate of this system is 0.25 passengers/second, an equivalence of 900 passengers/hour. It is important to mention that if the arrival rate is above 0.35 passengers/second, the waiting time will start to increase dramatically. This is the situation where the passenger arrival rate is higher than the elevator system can handle, such that passengers will get accumulated in front of the waiting area and get increase with time. In contrast to the waiting time, the service time behaves differently; although the service time increases with the arrival rate, the increases are gradual and get saturated at the higher arrival rates and starting around 0.35 passengers/second. At low arrival rates, the service time includes the door opening and closing time, passenger time (the time that a passenger walks in and out the elevator) and traveling time from arrival floor to the destination floor. At high arrival rates, elevators will stop more often between floors causing longer traveling time between floors. For the example, the passenger who arrives at the ground floor and destines at the top floor will spend non-stop traveling time of 33.4 seconds whereas when the elevator has to stop every floor in between, the traveling time will increase to 85.5 seconds. This is the one of the main reasons why the service time increases with the arrival rates. Finally, based on the total time in Fig. 8, if the system is operated at arrival rate of 0.25 passengers/second a passenger will spend on average 95 seconds in the system.

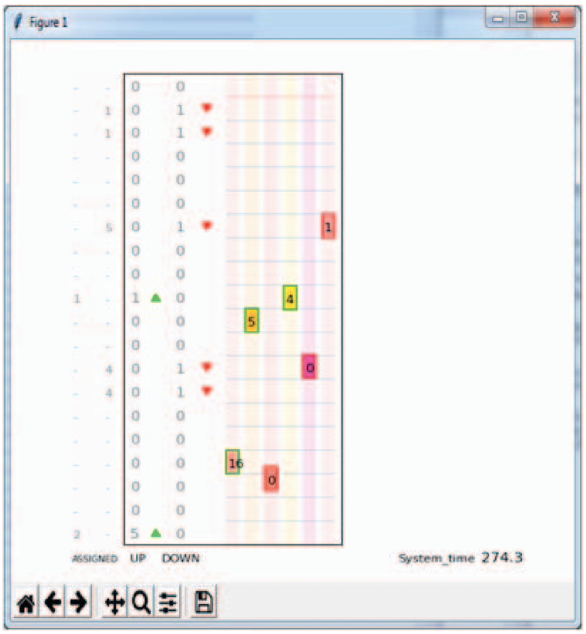

Fig. 7 Our simulation of elevator group control system.

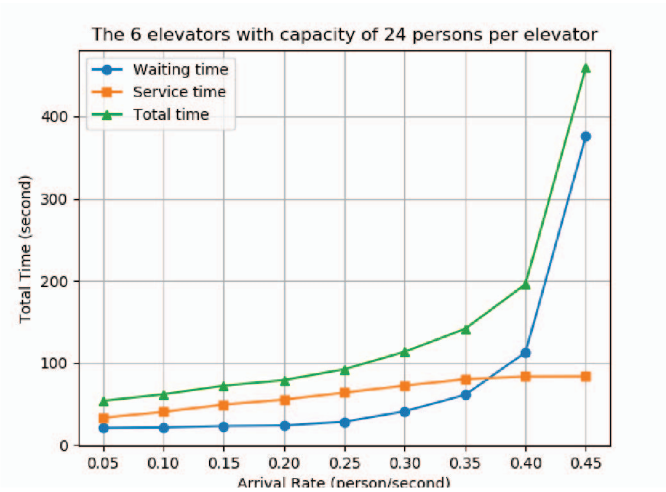

Fig. 8 Result from the simulation by varying the arrival rate.

V. CONCLUSIONS

The main contribution of this paper is the development of a software simulation tool written in Python for performance evaluation of vertical transportation systems with elevator group control. Our simulation model is composed of passenger arrival processes, a group of elevators and the nearest car scheduling algorithm. The proposed model takes into account of several key parameters, namely elevator kinematics, the number of elevators, the number of floors and the elevator capacity. Also, the simulation model is capable of animating the elevators in motions and passenger delivery in details almost the same as the real elevator systems. As a result, the simulator can precisely model the operation of elevator systems, and serves as a useful tool for measuring the elevator performance with respect to waiting time as well as service time and other important performance metrics. We believe that this software simulator can be effectively applied for future research in more advanced elevator group control systems.

REFERENCES

[1] J. R. Fernandez and P. Cortes, A Survey of Elevator Group Control Systems for Vertical Transportation, IEEE Control Systems, Vol. 35, No. 4, pp. 38-55, 2015.

[2] V. C. Galpin and S. T. Rock, A Lift Simulation Prototype, Wiley Online Library, 1995.

[3] R. Braun, Need a Lift? An Elevator Queuing Problem, United Technologies Research Centre, 14 August 2003, pp. 1-28.

[4] N. Pariyatdulapak, Development of a Simulation for Vertical Transport Systems, Senior Project Report, Chulalongkorn University, 2016.

[5] G. C. Barney, Elevator Traffic Handbook: Theory and Practice. Taylor Francis, 2003.

[6] J. Dallas, A Systematic Methodology for the Generation of Lift Passengers under a Poisson Batch Arrival Process. [online] Peters-research.com, 2017.

[7] J.-M. Kuusinen, J. Sorsa, M.-L. Siikonen, and H. Ehtamo, Study on the Arrival Process of Lift Passengers in a Multi-Storey Office Building, Building Serv. Eng. Res. Technol., Vol. 33, No. 4 (2012), pp. 437-449.

[8] R. Peters, Ideal Lift Kinematics, Available at: https://www.peters-research.com/index.php/support/articles-and-papers/53-ideal-lift-kinematics, 1995.

[9] Emporis.com, Cities with the most skyscrapers. [ONLINE] Available at: https://www.emporis.com/statistics/most-skyscraper-cities-worldwide. [Accessed 30 March 2018].

[10] Adsimulo.com, Lift Performance Criteria. [ONLINE] Available at: https://adsimulo.com/support/adsimulo-university/lift-performance-criteria/. [Accessed 30 March 2018].

[11] S. Markon, K. Aoki, M. Nakagawa, and T. Sudo, Recent Trends in Elevator Group Control Systems, International Technical Conference on Circuits/Systems, Computers and Communications, 2008, pp. 697-700.

[12] J. Sorsa, H. Hakonen, and M. Siikonen, Elevator Selection with Destination Control, International Congress on Vertical Transportation Technologies, 2005, pp. 1-13.