# Calculation of the Round-Trip Time under Destination Group Control using Offline Batch Allocations and Real-Time Allocations

**Albert So[1], Lutfi Al-Sharif*[2]**

**[1]School of Science and Technology, The University of Northampton, St. George's Avenue, Northampton NN2 6JD, UK**

**[2]Mechatronics Engineering Department, School of Engineering, The University of Jordan, Amman 11942, Jordan**

*** Corresponding author, e-mail address: lutfi.alsharif@outlook.com**

## Abstract

The use of Destination Group Control (DGC), or Hall Call Allocation (HCA), in elevator traffic system group control is the current trend in intelligent and advanced supervisory control and is expected to dominate the market in the future. In the conventional elevator traffic design process, designers usually start with a simple calculation in order to obtain a conceptual estimate of the suggested design prior to moving onto simulation. But due to the lack of a suitable set of equations for elevator traffic calculation for DGC systems, designers are forced to carry out the elevator traffic design process for a system controlled by DGC solely by using simulation. Due to the dependence on the simulator and the algorithms it uses, different simulation packages will produce different resultant designs. Thus, the motivation for this paper is to use calculation in order to achieve more transparency and repeatability in the design of DGC systems.

In order to enable the designer to carry out a calculation for the DGC system, equations are needed to evaluate the values of *H* (the highest reversal floor) and *S* (the expected number of stops in a round trip) in order to evaluate the value of the round-trip time under destination group control. Although equations are available to compute the highest reversal floor, *H*, and expected number of stops, *S*, of a DGC system, these equations assume idealized conditions. If designers use them to design the elevator traffic system, the design will be under-sized and inadequate. They do not take into consideration many of the practical implementation issues and non-ideal conditions such as: unequal floor population, real time call allocation of calls to elevator cars, and the different floor to sector arrangements, as well as the practicalities of allocating elevator cars to sectors.

In this paper, more detailed consideration is given to the estimation of *H* and *S* under both offline and real-time call allocations. Three methods of sectoring are suggested to take care of different combinations of the number of floors, the number of elevators, the car capacity and the floor population distribution. The results of this research would help the designer carry out a more reasonable and practical calculation of the round-trip time under DGC and thus arrive at a transparent and repeatable elevator traffic design.

The designer is still expected to move onto a simulation phase in order to understand the effect of the group controller on the system performance.

## 1. INTRODUCTION

Elevator group control is critical to the optimal operation of elevator traffic systems under general traffic conditions and is probably the most important mathematical problem to be solved in elevator systems. It is a very demanding task as it has, even for the simplest of situations, an excessively large number of possible solutions. It has been studied in a lot of detail, an example of which can be found in [1-15]. The aim of the elevator group control algorithm is to find the solution that optimizes a certain parameter of interest. The optimization could involve one or more of the following: maximizing the handling capacity, minimizing the average waiting time or the average travelling time.

Group control algorithms can be sub-divided into two main categories in accordance with the type of prevailing traffic:

1. General traffic group control algorithms: These group control algorithms are applied under any mix of traffic patterns (incoming; outgoing; inter-floor). Although some of the ideas presented in this paper could be applied to these group control algorithms, they will not be discussed in any further detail as they are deemed to be beyond its scope.

2. Up-peak group control algorithms: These group control algorithms are used in cases where most of the passengers are entering the building from the main entrance and heading to the occupant floors above ([2], [3], [16-21]).

In the last 20 years, new elevator group control algorithms have become available for use under up-peak traffic conditions (i.e., that fall within the second category above) and they form an important aspect of elevator traffic management. A number of these algorithms have evolved over that period of time.

Regardless of the different variations of up-peak group control algorithms, they can be viewed as different forms of sectoring. Sectoring is a control technique by which the floors in the building are grouped into virtual groups, referred to as sectors, to which individual elevators are allocated. Landing calls originating to and from within a sector are allocated to the elevator car associated with that sector. It has been shown that sectoring compared to conventional control in general can be applied in one of the following two scenarios [21]:

1. Reducing the car loading while still handling the same arrival rate. This reduction in car loading is accompanied by a reduction in passenger travelling time and an increase in passenger waiting time.

2. Increasing the handling capacity of the elevator traffic system in order to enable it to handle a higher arrival rate, without any change to the car loading. This increase in the handling capacity is accompanied by a larger increase in passenger waiting time, if not properly handled. *It is this second scenario that really makes sectoring a powerful tool that can be used to prevent an elevator system collapsing under heavy arrival rates.*

These up-peak algorithms can be generally classified into two broad categories: static sectoring (in which the sizes and compositions of the sectors are fixed) and dynamic sectoring (in which the sizes and compositions of the sectors change continuously).

Under static-sectoring, the building is split up during the up-peak period into a number of sectors of fixed size (hence the term static-sectoring) usually comprising contiguous floors. The static sectoring system can be further subdivided into two types: static-sectoring-staticallocation and static-sectoring-dynamic-allocation (such as Otis's Channeling system [2]). In the static-allocation variant, each elevator is permanently assigned to a specific sector and is always dispatched to that sector. The destination sector floors of each elevator are known by the waiting passengers in the main entrance. This has the advantage of providing the convenience of familiarity for the passengers and occupants of the building (passengers are creatures of habit and like to wait for and board the same group of elevators every day!). However, as the same elevator is serving the same sector all the time, the population of the sectors cannot be made equal and the size of the lower sectors must be made larger than the upper sectors (in order to equalize the handling capacity of the different sectors). In the dynamic-allocation variation, the elevators are assigned dynamically to the sectors and different elevators will serve different sectors in consecutive round trips. The destination sector floors assigned to the elevator are revealed to the waiting passengers in lobby as soon as (or just before) the elevator arrives in the main lobby. This allows the use of equal size sectors as different elevators will serve different sectors and the handling capacity of the sectors can be equalized. However, the convenience of having the same elevator serving the same sector is lost as far as passengers are concerned. Fortune [3] referred to the static-allocation and dynamic-allocations variations as fixed and rotational respectively. There are instances where the passengers prefer the use of static-allocation over dynamic allocation in buildings where static sectoring is used ([3], [4]). Dynamic sectoring operates in a similar way but with the difference that the sizes of the sectors change continuously ([5], [6], [7], [8], [9]).

This classification is shown in a 'tree' format in Figure 1 below.

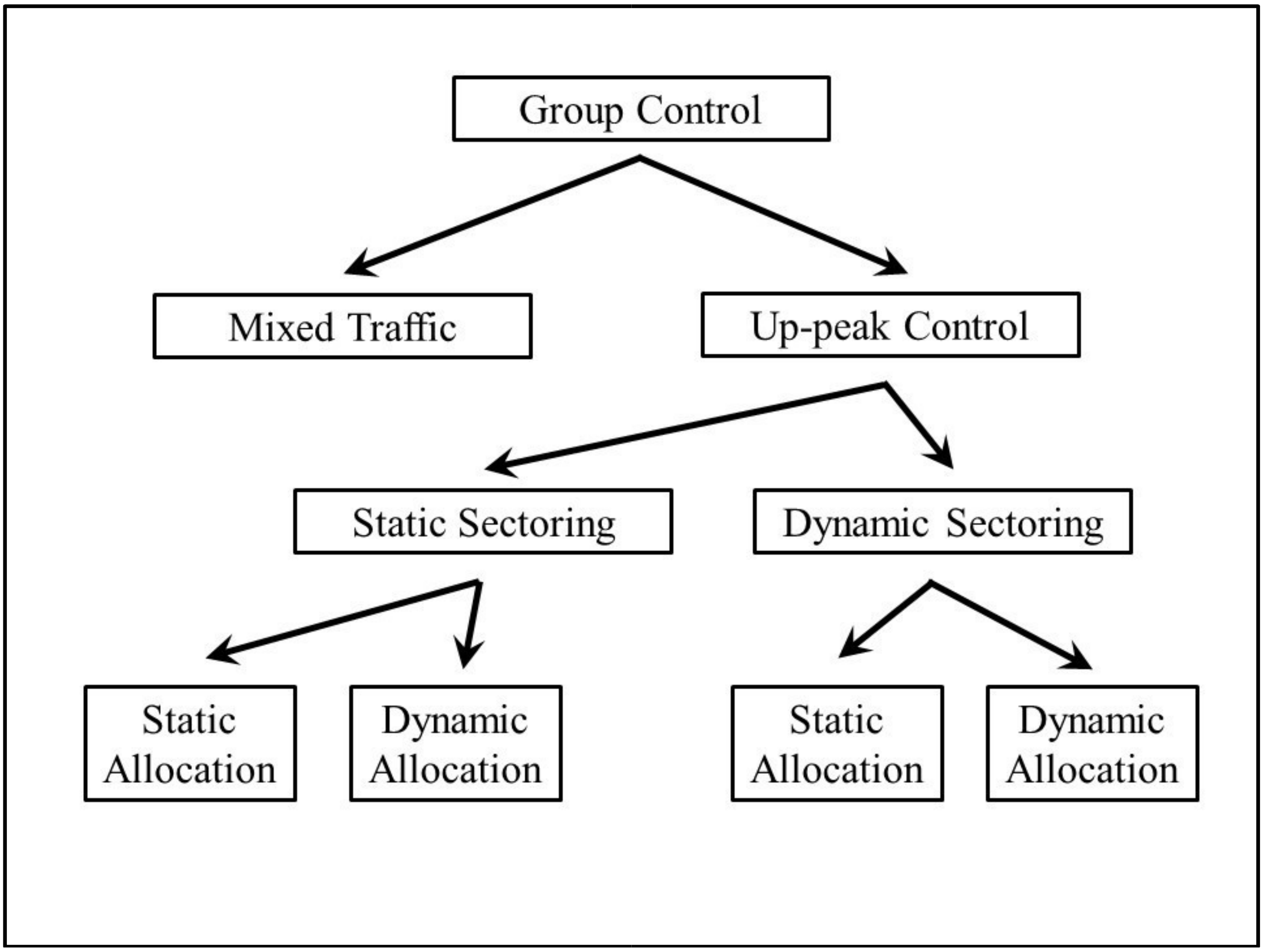


Figure 1: 'Tree' diagram of the different Up peak group control algorithms.

Destination Group Control (DGC) systems allow the passengers to register their destinations prior to boarding the elevator ([10], [11], [22-24]). The group control system can thus allocate the landing call to the most suitable elevator in the group and inform the passenger waiting in the lobby. As the elevator has more prior information, it is possible to make a better allocation decision. DGC has been used in the elevator industry for more than twenty-five years. It is sometimes called the "intelligent dispatcher" as a conventional system using relays or noncomputer based electronic circuits cannot employ it. In common with all sectoring systems, it helps to boost the up-peak handling capacity of a system while reducing the passenger travelling time and increasing the average waiting time [21].

Destination group control can be viewed as an example of dynamic-sectoring-dynamic allocation. Hence under DGC, the size and composition of the sectors change in every round trip as well as the allocation of the elevators to the sectors. The main feature that distinguishes DGC from the general category of sectoring is the fact that passenger destinations are known prior to the passengers boarding the elevator cars. Destination group control is the subject of this paper.

In order to enable an objective repeatable comparison of different up-peak traffic group control algorithms, it is necessary to have an evaluation mechanism of their effectiveness. The use of equations in evaluating elevator group control algorithms has been very limited ([17-18]).

Simulation has traditionally been the tool that is used to assess and compare elevator group control algorithms. However, simulation can suffer from a number of disadvantages such as the lack of repeatability, reproducibility, transparency and convergence ([25-26]).

A detailed design of the elevator traffic system using DGC is conventionally carried out by means of computer simulation. But very often, like conventional designs involving the standard collective group control, designers may sometimes want to have a quick and general overview as to how well the DGC traffic control system is performing. That is conventionally given by an estimated Round-Trip Time (*RTT*), Handling Capacity (*HC%*) and interval (*INT*) during the up-peak period. Both *HC%* and *INT* depend on the *RTT*.

The following round trip time (*RTT*) equation has been very widely used: $RTT = 2Ht_v + (S+1)(T-t_v) + 2Pt_p$ (where $t_v$ is the one floor cycle time under rated speed; $T$ is the performance time as defined in CIBSE Guide D [27]; $P$ is the average number of in-car passengers; $t_p$ is the average passenger transfer time). The structure of this equation clearly emphasizes the fact that the value of *RTT* heavily depends on the values of *H* (the highest reversal floor) and *S* (the probable number of stops). Estimating the values of *H* and *S* is pivotal to the work in this paper.

It is however, worth noting the *H* and *S* approach in calculating the *RTT* is based on two basic assumptions:

1. The rated speed is attained in one floor journey.
2. The floor heights are equal.

Nevertheless, the use of the *H* and *S* method for calculating the *RTT* is so practical, effective and insightful that the authors felt that the benefits of using the *H* and *S* approach in evaluating the round trip far outweigh any inaccuracy arising from the fact that the two conditions are not met in a design.

Once *H* and *S* are reasonably estimated, all other parameters can be found accordingly, and a preliminary system design can be completed. Although *H* and *S* can be found by computer simulations, they keep on varying unless thousands of simulations are repeated by using the Monte Carlo method ([28-31]). Furthermore, not all designers could conveniently access a powerful simulation software.

**Raison d'être of this paper**

The main aim of this paper is to develop a systematic approach for evaluating the round-trip time using a lookup table for expected values of *H* and *S*. Traditionally, lookup tables for the value of *H* and *S* have only depended on the values of *N* (number of floors in the building) and *P*. They did not include a value for the number of sectors. The new set of lookup tables to be developed in this paper introduces a new parameter: $S_n$ (the number of sectors). So, under this new approach, the values of *H* and *S* depend on three (rather than two parameters): *N*, *P* and $S_n$.

Section 2 of this paper lays the theoretical background for estimating *H* and *S* as idealized optimal benchmarks. However, it is well accepted that such hypothetical values only exist under the most favorable conditions and cannot be produced in reality. For this reason, section 3 explains why the real-world values of *H* and *S* deviate from the idealized optimal benchmark values. Section 4 introduces different approaches to splitting a building into sectors and evaluates the values of *H* and *S* for different buildings. Sections 5 and 6 introduce the methodology used in the Monte Carlo simulation method to find the value of *H* and *S* for integral value, and non-integral values of *N*/*L* respectively, where *L* is the number of elevators in the group, and for the condition when floor population is not uniform. In Section 6, a new general method, called fractional floor sectoring, is proposed to deal with sectoring a building under any choices of $N$, $L$, $P$, elevator capacity car-by-car, and unequal floor population distribution. A numerical example is given in section 7 to show how sectoring can be carried out by our newly developed method under any combinations of $N$, $L$, $P$, car capacity and equal/unequal floor population distribution. Section 8 presents a set of steps that can be followed in order to design a traffic system using the $H$ and $S$ lookup table. Conclusions are drawn in section 9.

## 2. CLASSICAL ESTIMATION OF *H* AND *S* AS IDEALIZED OPTIMAL BENCHMARK (IOB)

The most classical equations of $S$ and $H$ under DGC were derived by Schroeder in 1990 [12] and shown as equation (1) and (2). Schroder assumed that the DGC system assigns destination calls up to one and half round trips and the number of served floors is $2N$ instead of $N$ with a conventional control system, where $N$ is the number of floors above the main terminal. And furthermore, all destination calls are assumed to be distributed evenly between all elevators. The expected number of stops during an up-peak trip for all cars is calculated based on a huge elevator of size, $LP$, where $L$ is the number of elevators and $P$ is the number of passengers in each elevator. Then, the $S$ for one car is obtained by dividing such value by $L$.

$$S = \frac{2N}{L}\left[1-\left(1-\frac{1}{2N}\right)^{LP}\right] \quad \text{versus} \quad S_c = N\left[1-\left(1-\frac{1}{N}\right)^{P}\right] \qquad (1)$$

$$H = N - \sum_{i=1}^{N-1}\left(\frac{i}{N}\right)^{S} \quad \text{versus} \quad H_c = N - \sum_{i=1}^{N-1}\left(\frac{i}{N}\right)^{P} \qquad (2)$$

Here, $S_c$ is the expected number of stops under conventional collective control and $H_c$ is the highest reversal floor under conventional collective control. Barney [17] stated that in reality, the allocation of the landing calls to the cars in the group by DGC would have to be done in real time and the look-ahead capability of the system might have to be restricted to a smaller number of cars. So, a factor "$k$" was introduced to estimate $S$ where $k$ can take on values of 2, 3, 4 up to $L$.

$$S = \frac{2N}{k}\left[1-\left(1-\frac{1}{2N}\right)^{kP}\right] \qquad (3)$$

Sorsa [23] modified the equation for *H*, as shown in equation (4) by taking *k* = *L*.

$$S = \frac{N}{L}\left[1-\left(1-\frac{L}{N}\right)^{LP}\right] \qquad (4)$$

Al-Sharif et al. [16] derived two new equations for *S* and *H*, as shown in equations (5) and (6). He further proved that equation (4) is very close to equation (5) although they were derived by different approaches.

$$S = \frac{N}{L}\left[1-\left(1-\frac{L}{N}\right)^{P}\right] \qquad (5)$$

$$H = \frac{N}{2}\frac{L+1}{L} - \sum_{i=1}^{\frac{N}{L}-1}\left(\frac{iL}{N}\right)^{P} \qquad (6)$$

All these equations were derived without considering the problems encountered in allocating cars to serve landing calls under a real-time situation. In other words, they are based on offline allocation of landing calls. Al-Sharif et al [20] suggested the procedures for real time allocation of landing calls and gave a numerical example by computer simulation to demonstrate that but they did not derive equations for *S* and *H* to describe such real time allocation for the convenient application by designers. The work mentioned in this article is a follow-up of such consideration.

## 3. WHY DO THE ACTUAL *H* AND *S* DIFFER IN PRACTICE FROM THE IOB?

Regarding offline allocation of landing calls, it is assumed that, within one epoch, all the *LP* number of passengers and their destinations are well known. The term epoch is borrowed from the field of neural networks. An epoch is the cycle during which one full batch of passengers who can fully occupy all *L* number of cars arrives at the main terminal, are allocated to the elevator cars. Under the situation of DGC, it assumed that all passengers within such epoch have already registered their destinations and are aware of their designated car that they need to board and quietly wait at the lobby of the main terminal for their designated car. Then, all these passengers are ready to board whenever the designated car arrives. And all the elevators are available to take all such available passengers and depart. Furthermore, there are a number of ways to “sector” a building, each with different values of *H* and *S* even under offline allocation.

Under a real time situation, allocation of cars to landing calls is done passenger by passenger. The passenger is impatient enough to wait for the availability of all passengers in an epoch to register their destinations in one go. The DGC system must give an immediate response to a passenger who just keyed in his/her destination floor by allocating him/her the chosen car. A number of rules have to be respected in this algorithm [20]:

1. Although one epoch is still considered, the registered passenger destinations are revealed to the DGC controller sequentially, one at a time.
2. Each allocation has to be made as soon as it arises, preferably within one to two seconds.
3. Once made, the allocations cannot be altered later, even if it becomes obvious that a better allocation could have been made to improve the performance of the system.
4. Each elevator can accommodate a maximum number of *P* passengers. Once *P* passengers have been allocated to an elevator, no more passengers can be allocated to it, and passengers must be allocated to other elevators in the group.
5. The building is divided into sectors where one elevator is assigned to one or more sectors. Every landing call to a particular floor within a sector is served by the corresponding elevator. This is somehow similar to the concept of static sectoring but how a building is sectored and how a sector is assigned to an elevator keep on varying in a DGC system.

In coming sections, we first look into details of *H* and *S* estimation under offline allocations by using different types of sectoring, and then move on to discuss them under real time allocations.

## 4. *H* AND *S* OF THREE DIFFERENT TYPES OF SECTORING: CONTIGUOUS, BOTTOM/TOP, SLICED UNDER OFFLINE ALLOCATIONS

A building with with $N$ +1 number of floors, from 0/F, 1/F, to $N$/F, $N$ being even, served by a group of $L$ elevators where $N/L$ being an even integer as well, is assumed. They are shown as: $L$ number of elevators - L(1), L(2), …, L($i$), … L($L$), $L$ number of sectors - S(1), S(2), …, S($i$), …, S($L$), and each sector is served by one elevator, i.e. S($i$) is served by L($i$).

Three different types of sectoring are possible. There are shown in Figure 2.

Type 1 sectoring is called “Contiguous sectoring” which was described in [20]. The first sector consists of 2/F, 3/F, …, ($N/L$) /F; the second sector consists of ($N/L$+1)/F, ($N/L$+2)/F … (2$N/L$) /F and so on. Table 1 shows how the building is “sectored”.

**Table 1 Type 1: Contiguous Floor Sectoring (there are *N*/*L* number of floors for each sector)**

| Sector | From | To |
|---|---|---|
| 1 | 1 | $N/L$ |
| $i$ | $(i-1)(N/L)+1$ | $i(N/L)$ |
| $L$ | $(L-1)(N/L)+1$ | $N$ |

Type 2 sectoring is called “Bottom/Top sectoring”, which is newly proposed in this article. The first sector includes $N/(2L)$ pairs of floors, i.e. 1/F, N/F, (L+1)/F, (N-L)/F, and so on. Table 2 shows how the building is “sectored” under this type of sectoring.

**Table 2 Type 2: Bottom/Top Sectoring (still *N*/*L* number of floors for each sector)**

| Sector | 1st pair | | | *k*th pair | | | *N*/(2*L*)th pair | |
|---|---|---|---|---|---|---|---|---|
| | Lower Floor | Upper Floor | | Lower Floor | Upper Floor | | Lower Floor | Upper Boundary |
| 1 | 1 | $N$ | | $(k-1)L+1$ | $N-(k-1)L$ | | $(N/(2L)-1)L+1$ | $N/2+L$ |
| $i$ | $i$ | $N-(i-1)$ | | $(k-1)L+i$ | $N-(k-1)L-(i-1)$ | | $(N/(2L)-1)L+i$ | $N/2+L-(i-1)$ |
| $L$ | $L$ | $N-(L-1)$ | | $(k-1)L+L$ | $N-(k-1)L-(L-1)$ | | $N/2$ | $N/2+1$ |

Table 2 can be re-arranged into a consecutive ascending order, as shown in Table 3. “M” represents the middle floor of all the sectors, equal to (($N$+1)/2)/F, which may not be an integer.

**Table 3 Type 2: Re-arrangement into Consecutive Order of Table 2**

| Sector | 1st floor served | 2nd floor served | lower *k* th floor served | *N*/(2*L*) th floor served | M | *N*/(2*L*)+1 th floor served (= upper floor of *N*/(2*L*) th pair in Table 2) | *N*/(2*L*)+2 th floor served (= upper floor of *N*/(2*L*)-1 th pair in Table 2) | upper *k* th floor served | *N*/*L* th floor served (= upper floor of the 1st pair in Table 2) |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | *L*+1 | (*k*-1)*L*+1 | (*N*/(2*L*)-1)*L* + 1 | (*N*+1 )/2 | *N*/2+*L* | *N*/2+2*L* | *kL* | *N* |
| *i* | *i* | *L*+*i* | (*k*-1)*L*+*i* | (*N*/(2*L*)-1)*L* + *i* | (*N*+1 )/2 | *N*/2+*L*-(*i*-1) | *N*/2+2*L*-(*i*-1) | *kL*-(*i*-1) | *N*-(*i*-1) |
| *L* | *L* | 2*L* | (*k*-1)*L*+*L* | (*N*/(2*L*)-1)*L* + *L* | (*N*+1 )/2 | *N*/2+1 | *N*/2+*L*+1 | *kL*-(*L*-1) | *N*-(*L*-1) |

Type 3 sectoring is called "Staggering sectoring", which is also newly proposed in this article. The first floor belongs to the first sector, second floor to the second sector, third floor to the third sector and so on. After *L* number of floors has been assigned, the (*L*+1)/F belongs to the first sector again. Table 4 shows how the building is "sectored" under this type of sectoring.

**Table 4 Type 3: Staggering Sectoring (also *N*/*L* number of floors per sector)**

| Sector | 1st floor served | 2nd floor served | *k*th floor served | *N*/*L* th floor served |
|---|---|---|---|---|
| 1 | 1 | *L*+1 | (*k*-1)*L*+1 | *N*-*L*+1 |
| *i* | *i* | *L*+*i* | (*k*-1)*L*+*i* | *N*-*L*+i |
| *L* | *L* | 2*L* | (*k*-1)*L*+*L* | *N* |

The *H* and *S* of all three types of sectoring are different, as derived below. To avoid confusion, *H* under DGC of each sector is called $H_{local}$ while *H* under DGC of the whole building is called $H_{des}$ in the remaining part of this article. For each sector, there are *N*/*L* number of floors served by one elevator and the population of every floor is assumed to be constant.

### 4.1 Type 1 Contiguous sectoring

For offline allocations, the $H_{des}$ and $S$ were derived in [16], as shown in equations (5) and (6) above. They are reproduced here for ease of reference.

$$S = \frac{N}{L}\left[1 - \left(1 - \frac{L}{N}\right)^{P}\right] \quad (5)$$

$$H_{des} = \frac{N}{L} + \frac{1 - L}{2L} - \sum_{k=1}^{\frac{N}{L}-1}\left(\frac{kL}{N}\right)^{P} \quad (6)$$

It should be noted that $S$ is identical among all sectors but the local $H$ varies sector by sector. Equation (6) gives the average of $H_{local}$ of all sectors. Since Type 2 is rather non-linear, Type 3 is discussed first.

### 4.2 Type 3 Staggered sectoring

Since the population of every floor is equal, the common $H_{local}$ equation of each sector is still applicable but a conversion is needed as the floor spacing is not contiguous anymore. As shown in Table 4, S($i$) includes the $i$th floor, ($L$+$i$)th floor, and so on. $H_{local}$ of the $i$th sector, though within the range from 1 to $N/L$, may not be an integer. It has to be matched to a range between the $i$th floor as the lower boundary of the sector and the ($N$-$L$+$i$) th floor as the upper boundary of the sector. $H_{local}$ is the $H$ of a sector irrespective of its location; $H_{eff}$ is the real $H$ of a sector by considering its lowest floor not being equal to the first floor. $H_{des}$ is then the average of all $H_{eff}$. Equation (5) can still be used for $S$ in this type of sectoring.

$$H_{local} = \frac{N}{L} - \sum_{k=1}^{\frac{N}{L}-1}\left(\frac{kL}{N}\right)^{P}$$

$$H_{eff}(i) = (i - L) + L H_{local} = (i - L) + L\left[\frac{N}{L} - \sum_{k=1}^{\frac{N}{L}-1}\left(\frac{kL}{N}\right)^{P}\right]$$

$$H_{des} = \square^{i=}{}_{1} HL_{eff}(i) = L1 \quad {}_{L1}\square\square\square(i - L) + N - L\square \quad {}_{NL-11}\square\square\ kLN\ \square\square\square_{P}\square\square\square\square$$

(7)

$i=\square$ $k=\square$

$\square$ $\square$

$\square$ $\frac{N}{}-1$ $P\square$

$$= L1\ \square\square\square\ L(L+1) - L_2 + NL - L_2 \quad {}_{L1}\square\square\ kLN\ \square\square\square\ \square\square\square$$

2 $k=\square$

$\square$ $\square$

$N$ $N$ $N$

$$kL\square = N - L - 1 - L\ \square\square\ \square$$

$$= L2 \pm 1 - L + N - L\square_{Lk=-11}\square\square\square\ N\ \square\square_{P}2\ \square_{Lk=-11}\square\ kLN\ \square\square_{P} = N + 12 - L\square\square\square\ 12 + \square_{L-11}\square\square\ kLN\ \square\square\square_{P}\square\square\square\square$$

$\square$ $k=\square$

$\square$ $\square$

### 4.3 Type 2 Bottom/Up Sectoring

For this type, the $H_{local}$ equation in equation set (7) above can still be used but the conversion formula is a bit complicated and non-linear. Matching is shown in Table 5.

**Table 5 Conversion from $H_{local}$ of the *i*th Sector to Real Floor Level for Type 2**

| Condition of $H_{local}$ | $H_{local}$ Range | Matched to Real Floor Range |
|---|---|---|
| < *N*/(2*L*) | [1, *N*/(2*L*) ) | [ *i*, (*N*/(2*L*)-1)*L*+*i* ] |
| = *N*/(2*L*) | *N*/(2*L*) | (*N*+1)/2 |
| > *N*/(2*L*) | (*N*/(2*L*) , *N*/*L*] | [ *N*/2+*L*-(*i*-1) , *N*-(*i*-1) ] |

It is generally believed that $H_{local}$ should always be higher than *N*/(2*L*) in real practice and therefore the third row of the table above is usually applicable. $H_{des}$ is derived as follows in equation set (8). Again, equation (5) can still be used for *S* in this type of sectoring.

$H_{eff}(i) = H_{local} \frac{}{L}\left(N - 2L\right) + 2L - (i - 1) N$

$H_{eff}(i)$ $N-$

$H_{des} = \sum_{i=1}^{L} \frac{1}{L} \quad \frac{1}{L} \quad \frac{N}{L} - \sum_{k=1}^{L-1} \frac{kL}{N} \quad NL\left(N - 2L\right) + 2L - (i - 1)$

$i=$

$N^{-1}$ $P$

$$= \frac{N}{L} - \sum_{k=1}^{L} \frac{kL}{N} \quad NL\left(N - 2L\right) + 2L + 1 - \left(\frac{L^2+1}{}\right) \tag{8}$$

$N$

$= -\frac{NL}{} -1\, kL \quad L\left(N - 2L\right) + 3L + 1$

$-$

$L$ $k=1$ $N$ $N$ 2 2

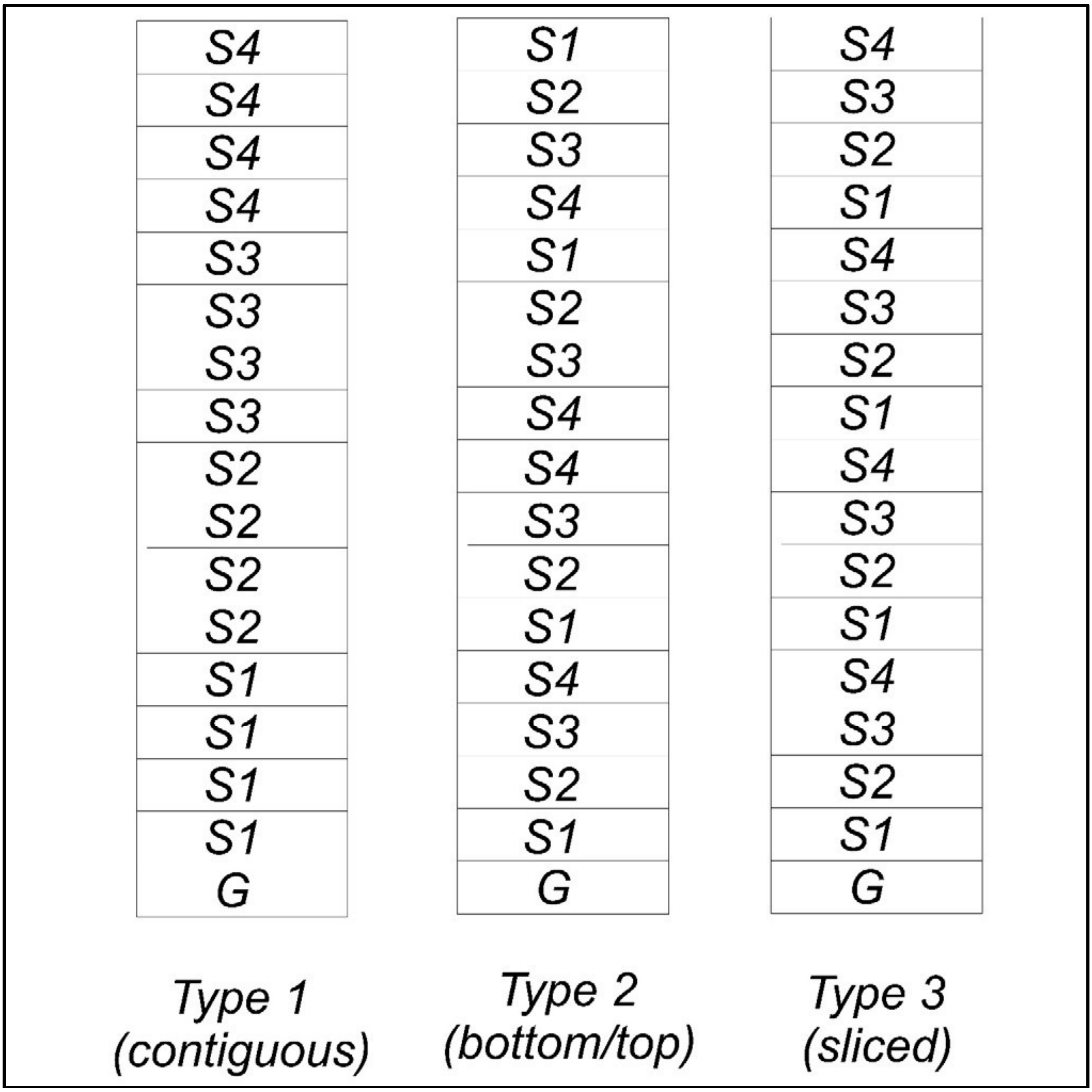


Figure 2: The three different types of sectoring.

### 4.4 Numerical Comparison between $H_{des}$ of three Types

Table 6 gives a quick comparison between the $H_{des}$ of three types of sectoring based on some selected values of *N, L* and *P*. It can be seen that $H_{des}$ of Type 1 is always the lowest, thus an IOB, while $H_{des}$ of Type 2 and Type 3 are similar with Type 2 being always the highest.

**Table 6 Quick Comparison between $H_{des}$ of three Types of Sectoring**

| *N* | *L* | *P* | $H_{des}$ Type 1 | $H_{des}$ Type 2 | $H_{des}$ Type 3 |
|---|---|---|---|---|---|
| 24 | 4 | 8 | 14.72 | 21.76 | 21.40 |
| 24 | 4 | 20 | 14.97 | 22.43 | 22.39 |
| 12 | 3 | 12 | 7.97 | 10.95 | 10.90 |
| 48 | 6 | 16 | 27.87 | 44.92 | 44.73 |

In the next section, we shall start to look at deviations of *H* and *S* under real time allocations, versus offline allocations.

## 5. METHODOLOGY 1: MONTE CARLO SIMULATION TO FIND THE ACTUAL *H* AND *S* (INTEGRAL VALUES OF *N*/*L*) UNDER REAL TIME ALLOCATIONS

To further study the variation of $H_{des}$ and *S* of DGC, a combination of common *P*, *N* and *L* in the industry was chosen and a Monte Carlo simulation with at least three thousand epochs for each case was conducted to evaluate the actual highest reversal floor, $H_{real}$, and the actual number of stops, $S_{real}$, and to compare them with calculated $H_{des}$ and *S* of the three types of sectoring.

The steps of the algorithm for real time allocations were discussed in [20] and they are duplicated here for easy reference.

1. Once a passenger arrives at the main terminal and registers his/her destination floor, the corresponding elevator of that sector consisting of the destination floor is assigned to this landing call.

2. This process continues until *P* number of landing calls has been assigned to a particular elevator.

3. When a passenger's destination is a floor belonging to a sector, say S(*i*), corresponding to a saturated elevator, the sector above, S(*i*+1), is considered. Here, the classical "contiguous sectoring approach" is adopted. If the sector above also corresponds to a saturated elevator, the sector further above is considered until the consideration reaches the topmost sector, S(*L*).

4. If all elevators of all sectors above have been saturated, the sector below, S(*i*-1), is considered, followed by S(*i*-1) until S(1) is reached.

5. Consideration is given to one epoch with *LP* number of passengers at time, with *L* number of elevators each with a capacity of *P* number of passengers. The full allocation will always be successful. However, cross-sector allocations heavily affect the actual $H_{des}$ and *S* and hence the *RTT*.

Table 7 shows the results of calculation and simulation. *HType*1 is in effect the Ideal Optimal Benchmark. *SType*123 is calculated based on equation (5) and it is a constant for all three types of sectoring. *S* is calculated based on equation (3) from [32] where *k* = *L*, just for reference.

**Table 7 $H_{des}$ and *S* based on Calculation and Monte Carlo Simulation**

| *P* | *N* | *L* | *HType* 1 | *HType* 2 | *HType* 3 | *Hreal* | *HType* 2 / *HType* 1 | *HType* 3 / *HType* 1 | *Hreal* / *HType* 1 | *SType* 123 | *S*(Barney) | *Sreal* | *Sreal* / *SType* 123 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | | | |
| 8 | 24 | 3 | 15.5 | 21.9 | 21.6 | 17.5 | 1.41 | 1.39 | 1.13 | 5.3 | 7.7 | 5.7 | 1.09 |
| 8 | 24 | 4 | 14.7 | 21.8 | 21.4 | 17.0 | 1.48 | 1.45 | 1.15 | 4.6 | 6.0 | 5.3 | 1.15 |
| 8 | 24 | 6 | 13.9 | 21.2 | 20.9 | 16.1 | 1.52 | 1.50 | 1.16 | 3.6 | 4.0 | 4.5 | 1.26 |
| 8 | 24 | 8 | 13.5 | 20.4 | 20.2 | 15.9 | 1.52 | 1.50 | 1.18 | 2.9 | 3.0 | 4.0 | 1.37 |
| 8 | 20 | 4 | 12.3 | 18.1 | 17.8 | 14.5 | 1.47 | 1.44 | 1.17 | 4.2 | 5.0 | 4.9 | 1.18 |
| 8 | 20 | 5 | 11.9 | 17.7 | 17.5 | 14.1 | 1.49 | 1.47 | 1.18 | 3.6 | 4.0 | 4.5 | 1.26 |
| 8 | 20 | 10 | 11.0 | 15.5 | 15.5 | 13.0 | 1.41 | 1.41 | 1.18 | 2.0 | 2.0 | 3.3 | 1.63 |
| 8 | 16 | 2 | 11.5 | 14.8 | 14.6 | 12.6 | 1.28 | 1.26 | 1.09 | 5.3 | 7.1 | 5.6 | 1.07 |
| 8 | 16 | 4 | 9.9 | 14.3 | 14.1 | 11.3 | 1.44 | 1.42 | 1.15 | 3.6 | 4.0 | 4.4 | 1.22 |
| 8 | 16 | 8 | 9.0 | 12.5 | 12.5 | 10.5 | 1.39 | 1.39 | 1.17 | 2.0 | 2.0 | 3.1 | 1.55 |
| 8 | 10 | 2 | 7.3 | 9.3 | 9.1 | 8.0 | 1.27 | 1.25 | 1.09 | 4.2 | 4.9 | 4.7 | 1.12 |
| 8 | 10 | 5 | 6.0 | 8.0 | 8.0 | 6.9 | 1.33 | 1.33 | 1.15 | 2.0 | 2.0 | 3.0 | 1.48 |
| 13 | 24 | 3 | 15.8 | 22.5 | 22.4 | 18.0 | 1.43 | 1.42 | 1.14 | 6.6 | 8.0 | 7.4 | 1.13 |
| 13 | 24 | 4 | 14.9 | 22.2 | 22.1 | 17.6 | 1.49 | 1.48 | 1.18 | 5.4 | 6.0 | 6.6 | 1.22 |
| 13 | 24 | 6 | 14.0 | 21.4 | 21.4 | 16.6 | 1.53 | 1.53 | 1.18 | 3.9 | 4.0 | 5.3 | 1.36 |
| 13 | 24 | 8 | 13.5 | 20.5 | 20.5 | 16.0 | 1.52 | 1.52 | 1.19 | 3.0 | 3.0 | 4.5 | 1.51 |
| 13 | 20 | 4 | 12.4 | 18.4 | 18.3 | 14.5 | 1.48 | 1.47 | 1.16 | 4.7 | 5.0 | 5.9 | 1.26 |
| 13 | 20 | 5 | 12.0 | 17.9 | 17.9 | 14.3 | 1.50 | 1.49 | 1.20 | 3.9 | 4.0 | 5.2 | 1.34 |
| 13 | 20 | 10 | 11.0 | 15.5 | 15.5 | 13.1 | 1.41 | 1.41 | 1.19 | 2.0 | 2.0 | 3.5 | 1.73 |
| 13 | 16 | 2 | 11.8 | 15.2 | 15.1 | 12.9 | 1.29 | 1.28 | 1.09 | 6.6 | 7.8 | 7.2 | 1.09 |
| 13 | 16 | 4 | 10.0 | 14.5 | 14.4 | 11.8 | 1.45 | 1.44 | 1.18 | 3.9 | 4.0 | 5.1 | 1.31 |
| 13 | 16 | 8 | 9.0 | 12.5 | 12.5 | 10.6 | 1.39 | 1.39 | 1.18 | 2.0 | 2.0 | 3.4 | 1.70 |
| 13 | 10 | 2 | 7.4 | 9.4 | 9.4 | 8.3 | 1.27 | 1.26 | 1.11 | 4.7 | 5.0 | 5.4 | 1.15 |
| 13 | 10 | 5 | 6.0 | 8.0 | 8.0 | 7.1 | 1.33 | 1.33 | 1.18 | 2.0 | 2.0 | 3.2 | 1.61 |
| 17 | 24 | 3 | 15.9 | 22.7 | 22.7 | 18.4 | 1.43 | 1.43 | 1.16 | 7.2 | 8.0 | 8.4 | 1.17 |
| 17 | 24 | 4 | 15.0 | 22.4 | 22.3 | 17.5 | 1.50 | 1.49 | 1.17 | 5.7 | 6.0 | 7.2 | 1.25 |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 17 | 24 | 6 | 14.0 | 21.5 | 21.5 | 16.5 | 1.53 | 1.53 | 1.18 | 4.0 | 4.0 | 5.5 | 1.40 |
| 17 | 24 | 8 | 13.5 | 20.5 | 20.5 | 16.3 | 1.52 | 1.52 | 1.21 | 3.0 | 3.0 | 4.7 | 1.56 |
| 17 | 20 | 4 | 12.5 | 18.4 | 18.4 | 14.5 | 1.48 | 1.48 | 1.16 | 4.9 | 5.0 | 6.3 | 1.29 |
| 17 | 20 | 5 | 12.0 | 18.0 | 18.0 | 14.1 | 1.50 | 1.50 | 1.18 | 4.0 | 4.0 | 5.4 | 1.37 |
| 17 | 20 | 10 | 11.0 | 15.5 | 15.5 | 13.2 | 1.41 | 1.41 | 1.20 | 2.0 | 2.0 | 3.6 | 1.79 |
| 17 | 16 | 2 | 11.9 | 15.3 | 15.3 | 13.4 | 1.29 | 1.29 | 1.13 | 7.2 | 7.9 | 8.0 | 1.12 |
| 17 | 16 | 4 | 10.0 | 14.5 | 14.5 | 11.8 | 1.45 | 1.45 | 1.18 | 4.0 | 4.0 | 5.4 | 1.35 |
| 17 | 16 | 8 | 9.0 | 12.5 | 12.5 | 10.8 | 1.39 | 1.39 | 1.20 | 2.0 | 2.0 | 3.6 | 1.79 |
| 17 | 10 | 2 | 7.5 | 9.5 | 9.5 | 8.3 | 1.27 | 1.26 | 1.12 | 4.9 | 5.0 | 5.7 | 1.18 |
| 17 | 10 | 5 | 6.0 | 8.0 | 8.0 | 7.2 | 1.33 | 1.33 | 1.20 | 2.0 | 2.0 | 3.3 | 1.66 |
| 20 | 24 | 3 | 15.9 | 22.8 | 22.8 | 18.4 | 1.43 | 1.43 | 1.15 | 7.4 | 8.0 | 8.7 | 1.17 |
| 20 | 24 | 4 | 15.0 | 22.4 | 22.4 | 17.8 | 1.50 | 1.50 | 1.19 | 5.8 | 6.0 | 7.4 | 1.27 |
| 20 | 24 | 6 | 14.0 | 21.5 | 21.5 | 16.8 | 1.54 | 1.53 | 1.20 | 4.0 | 4.0 | 5.7 | 1.44 |
| 20 | 24 | 8 | 13.5 | 20.5 | 20.5 | 16.3 | 1.52 | 1.52 | 1.21 | 3.0 | 3.0 | 4.8 | 1.60 |
| 20 | 20 | 4 | 12.5 | 18.5 | 18.5 | 14.8 | 1.48 | 1.48 | 1.19 | 4.9 | 5.0 | 6.5 | 1.32 |
| 20 | 20 | 5 | 12.0 | 18.0 | 18.0 | 14.4 | 1.50 | 1.50 | 1.20 | 4.0 | 4.0 | 5.6 | 1.41 |
| 20 | 20 | 10 | 11.0 | 15.5 | 15.5 | 13.4 | 1.41 | 1.41 | 1.22 | 2.0 | 2.0 | 3.7 | 1.87 |
| 20 | 16 | 2 | 11.9 | 15.4 | 15.4 | 13.2 | 1.29 | 1.29 | 1.11 | 7.4 | 8.0 | 8.3 | 1.12 |
| 20 | 16 | 4 | 10.0 | 14.5 | 14.5 | 11.9 | 1.45 | 1.45 | 1.19 | 4.0 | 4.0 | 5.5 | 1.38 |
| 20 | 16 | 8 | 9.0 | 12.5 | 12.5 | 11.0 | 1.39 | 1.39 | 1.22 | 2.0 | 2.0 | 3.7 | 1.85 |
| 20 | 10 | 2 | 7.5 | 9.5 | 9.5 | 8.2 | 1.27 | 1.27 | 1.10 | 4.9 | 5.0 | 5.9 | 1.19 |
| 20 | 10 | 5 | 6.0 | 8.0 | 8.0 | 7.2 | 1.33 | 1.33 | 1.20 | 2.0 | 2.0 | 3.4 | 1.68 |
| 26 | 24 | 3 | 16.0 | 22.9 | 22.9 | 18.5 | 1.44 | 1.43 | 1.16 | 7.8 | 8.0 | 9.4 | 1.21 |
| 26 | 24 | 4 | 15.0 | 22.5 | 22.5 | 18.0 | 1.50 | 1.50 | 1.20 | 5.9 | 6.0 | 7.8 | 1.31 |
| 26 | 24 | 6 | 14.0 | 21.5 | 21.5 | 16.7 | 1.54 | 1.54 | 1.19 | 4.0 | 4.0 | 6.0 | 1.49 |
| 26 | 24 | 8 | 13.5 | 20.5 | 20.5 | 16.2 | 1.52 | 1.52 | 1.20 | 3.0 | 3.0 | 5.0 | 1.65 |
| 26 | 20 | 4 | 12.5 | 18.5 | 18.5 | 15.1 | 1.48 | 1.48 | 1.20 | 5.0 | 5.0 | 6.8 | 1.36 |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 26 | 20 | 5 | 12.0 | 18.0 | 18.0 | 14.5 | 1.50 | 1.50 | 1.21 | 4.0 | 4.0 | 5.8 | 1.46 |
| 26 | 20 | 10 | 11.0 | 15.5 | 15.5 | 13.2 | 1.41 | 1.41 | 1.20 | 2.0 | 2.0 | 3.9 | 1.94 |
| 26 | 16 | 2 | 12.0 | 15.5 | 15.4 | 13.4 | 1.29 | 1.29 | 1.12 | 7.8 | 8.0 | 8.8 | 1.13 |
| 26 | 16 | 4 | 10.0 | 14.5 | 14.5 | 11.8 | 1.45 | 1.45 | 1.18 | 4.0 | 4.0 | 5.7 | 1.43 |
| 26 | 16 | 8 | 9.0 | 12.5 | 12.5 | 10.9 | 1.39 | 1.39 | 1.21 | 2.0 | 2.0 | 3.8 | 1.88 |
| 26 | 10 | 2 | 7.5 | 9.5 | 9.5 | 8.3 | 1.27 | 1.27 | 1.11 | 5.0 | 5.0 | 6.0 | 1.21 |
| 26 | 10 | 5 | 6.0 | 8.0 | 8.0 | 7.3 | 1.33 | 1.33 | 1.21 | 2.0 | 2.0 | 3.5 | 1.74 |
| | | | | | | | | | | | | | |
| | | | | | Maximum | | 1.54 | 1.54 | 1.22 | | | | 1.94 |
| | | | | | Minimum | | 1.27 | 1.25 | 1.09 | | | | 1.07 |
| | | | | | Mean | | 1.42 | 1.42 | 1.17 | | | | 1.40 |
| | | | | | Standard Deviation | | 0.086 | 0.086 | 0.035 | | | | 0.237 |

Table 7 shows that actual $H_{des}$ based on simulation is always between that of Type 1 and Type 2/3. And $S_{real}$ is also larger than the $S$ of Type 1,2 or 3 as they are all equal under equal floor population distribution. These are in accordance with the Idealized Optimal Benchmark (IOB) equations, i.e. Type 1, cannot be used for a quick design guideline, or otherwise, the system would be under-designed as the *RTT* is always longer under real time allocations versus offline allocations. Table 7 would be a good tool for designers to quickly see the framework of an elevator system using DGC. The designer first of all makes use of the IOB equations, i.e. Type 1, to get $H_{des}$ and $S$. Table 7 is then consulted in order to extract the appropriate multiplier factor for finding the actual value of $H_{des}$ and $S$. Using these two multipliers, the effective value of the round-trip time can be calculated.

Designers may simply use the two means of ratios as stipulated in Table 7, i.e. 1.17 for $H$ and 1.40 for $S$. To further help designers to avoid looking up long tables and doing interpolation, two formulae were generated by regression so that the two ratios could be calculated easily from $P$, $N$ and $L$, namely the $H_{ratio} = H_{real}/H_{Type1}$ and $S_{ratio} = S_{real}/S_{Type123}$.

$$H_{ratio} = 0.00230886P + 0.00000026P^2 + 0.00099067N + 0.00000416N^2 + 0.008227L + 1.06997237 \quad (9)$$

$$S_{ratio} = 0.00953577P + 0.00000001P^2 + 0.00000001N + 0.03906110L + 0.00335103L^2 + 0.90219785 \quad (10)$$

It appears that $H_{ratio}$ is rather insensitive to $L^2$ while $S_{ratio}$ is insensitive to $P^2$, $N$ and $N^2$.

## 6. METHODOLOGY 2: DEALING WITH NON-INTEGRAL VALUES OF *N/L* AND NONUNIFORM DISTRIBUTION OF FLOOR POPULATION

So far, we have been dealing with situations when *N/L* is always an integer. Second, we have been assuming that the distribution of floor population is always uniform. But it is usually not the case as the choices of number of floors, *N*, and the number of elevators, *L*, are quite arbitrary. Most buildings are usually multi-tenanted, resulting in non-uniform distribution of floor population. In this section, a new method is to be introduced so that designers to carry out a sectoring design based on the appropriate values of of *N*, *L* and floor population distribution to arrive at the estimation of highest reversal floor $H_{local}$, expected number of stops, *S*, and the *RTT* of every sector. Then, the *RTT* of the whole system with all *L* sectors can be found by averaging. In this way, traffic design for HCA (hall call allocation) or DGC systems can be handled, all by calculation, to reveal a general trend without the need to perform simulations that are confined to selected cases. Before that, a much simpler approach is first introduced.

### 6.1 $H_{des}$ Evaluation when *N/L* is non-integral

First, we go back to equation (6) where the $H_{des}$ is evaluated.

$$H_{des} = \frac{N(L+1)}{2L} - \sum_{j=1}^{N/L} \left[ 1 - \frac{jL}{N} \right]^{P} \quad (6)$$

If *N/L* is not an integer, we can employ the largest integer smaller than *N/L*, i.e. truncate (*N/L*) or the smallest integer larger than *N/L*, i.e. truncate (*N/L*) + 1. Table (8) shows the evaluation of $H_{des}$ based on some typical values of *N*, *L* and *P* based on equation (6). Those shaded rows belong to cases when *N/L* is an integer. It can be seen that when *N/L* is not an integer, the resultant $H_{des}$ differs less between the choice of using either the largest integer < *N/L* or smallest integer > *N*/*L*. More precisely, it is possible to use a weighted average to arrive at the $H_{des}$ based on interpolation because the exact position of *N*/*L* between the two integers is well defined.

This approach also seems applicable to the evaluation of *S* by using the standard equation (5). Again, two *S*'s are evaluated, one for the highest floor below *N*/*L*, i.e. truncate (*N*/*L*), and one for the lowest floor above *N*/*L*, i.e. truncate (*N*/*L*) + 1.

$$S = N_x\left[1-\left(1-\frac{1}{N_x}\right)^P\right] \qquad (11)$$

where $N_x = \text{truncate}(N/L)$ or $\text{truncate}(N/L)+1$

It is enough to evaluate *S* once because all sectors look identical, under an equal floor population distribution.

There are two issues here. First, if *N*/*L* is a multiple of 0.5, this approach still makes sense due to the symmetry between sectors, but not otherwise. Second, if the floor population distribution is not uniform, even if *N*/*L* is an integer, sectors do not share equal number of floors or otherwise, some sectors may need to handle too many passengers and some too less.

The goal of this paper is to present a method for designers working on HCA or DGC systems by calculation alone, just like the conventional practice of handling a single sector building. So, a more general methodology is necessary to create reasonable sectors of any choices of *N*, *L* and floor population distribution. And this is called "fractional floor sectoring".

**Table 8 Evaluation of $H_{des}$ of typical combinations of *N*, *L* and *P***

| ***N*** | ***L*** | ***P*** | ***N*/*L* chosen** | **$H_{des}$** | | ***N*** | ***L*** | ***P*** | ***N*/*L* chosen** | **$H_{des}$** |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | |
| 10 | 3 | 13 | 3 | 6.662 | | 10 | 3 | 16 | 3 | 6.665 |
| 10 | 3 | 13 | 4 | 6.643 | | 10 | 3 | 16 | 4 | 6.657 |
| 12 | 3 | 13 | 4 | 7.976 | | 12 | 3 | 16 | 4 | 7.99 |
| 14 | 3 | 13 | 4 | 9.309 | | 14 | 3 | 16 | 4 | 9.323 |
| 14 | 3 | 13 | 5 | 9.277 | | 14 | 3 | 16 | 5 | 9.305 |
| 16 | 3 | 13 | 5 | 10.61 | | 16 | 3 | 16 | 5 | 10.638 |
| 16 | 3 | 13 | 6 | 10.568 | | 16 | 3 | 16 | 6 | 10.611 |
| 18 | 3 | 13 | 6 | 11.901 | | 18 | 3 | 16 | 6 | 11.944 |
| 20 | 3 | 13 | 6 | 13.235 | | 20 | 3 | 16 | 6 | 13.278 |
| 20 | 3 | 13 | 7 | 13.185 | | 20 | 3 | 16 | 7 | 13.244 |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 10 | 4 | 13 | 2 | 6.25 | | 10 | 4 | 16 | 2 | 6.25 |
| 10 | 4 | 13 | 3 | 6.245 | | 10 | 4 | 16 | 3 | 6.248 |
| 12 | 4 | 13 | 3 | 7.495 | | 12 | 4 | 16 | 3 | 7.498 |
| 14 | 4 | 13 | 3 | 8.745 | | 14 | 4 | 16 | 3 | 8.748 |
| 14 | 4 | 13 | 4 | 8.726 | | 14 | 4 | 16 | 4 | 8.74 |
| 16 | 4 | 13 | 4 | 9.976 | | 16 | 4 | 16 | 4 | 9.99 |
| 18 | 4 | 13 | 4 | 11.226 | | 18 | 4 | 16 | 4 | 11.24 |
| 18 | 4 | 13 | 5 | 11.194 | | 18 | 4 | 16 | 5 | 11.222 |
| 20 | 4 | 13 | 5 | 12.444 | | 20 | 4 | 16 | 5 | 12.472 |
| 10 | 5 | 13 | 2 | 6 | | 10 | 5 | 16 | 2 | 6 |
| 12 | 5 | 13 | 2 | 7.2 | | 12 | 5 | 16 | 2 | 7.2 |
| 12 | 5 | 13 | 3 | 7.195 | | 12 | 5 | 16 | 3 | 7.198 |
| 14 | 5 | 13 | 2 | 8.4 | | 14 | 5 | 16 | 2 | 8.4 |
| 14 | 5 | 13 | 3 | 8.395 | | 14 | 5 | 16 | 3 | 8.398 |
| 16 | 5 | 13 | 3 | 9.595 | | 16 | 5 | 16 | 3 | 9.598 |
| 16 | 5 | 13 | 4 | 9.576 | | 16 | 5 | 16 | 4 | 9.59 |
| 18 | 5 | 13 | 3 | 10.795 | | 18 | 5 | 16 | 3 | 10.798 |
| 18 | 5 | 13 | 4 | 10.776 | | 18 | 5 | 16 | 4 | 10.79 |
| 20 | 5 | 13 | 4 | 11.976 | | 20 | 5 | 16 | 4 | 11.99 |

### 6.2 A General Algorithm of Sectoring for Real Time Allocations

To provide a general algorithm to create sectors for a building under DGC, all choices of *N*, *L*, *P* and floor population distribution must be taken care of. It should be noted that normally, under a HCA or DGC system, the number of sectors is equal to the number of elevators available. However, elevators could be statically or dynamically allocated to different sectors during operation (one epoch at a time). When the elevators serving the building have identical car capacities (which is usually the case) and the floor population distribution is uniform, the sizes of the sector populations are equal. Thus, the floor populations of al the sectors are equal and the number of floors in each sector are equal.

If the car carrying capacities of the elevators are unequal or if the floor population distribution varies floor by floor, the size of the sectors is unequal. Normally, buildings are served by elevators of equal car capacity for better maintenance and are assumed to have equal floor population if a particular distribution has not been provided by either the building owner or the architect during the design stage. In this general method, no special case assumption are made in advance.

The whole idea of splitting a building into sectors is shown in Figure 3. MT is the main terminal (which is usually the ground floor) with *N* number of floors above. Suppose the floor population of the *N* floors is given by $U_1$, $U_2$, …,$U_N$ where *U* (total population of the building) = $U_1 + U_2 + \ldots + U_N$. Every sector consists of contiguous number of floors, Type 1 sectoring as the IOB. A boundary floor is shared by two contiguous sectors. And the floor number of this boundary floor is designated as $f(i, i+1)$ between sectors S(*i*) and S(*i*+1) and the population of such a boundary floor is also denoted by $U_{f(i,i+1)}$. The $L$ number of sectors, equivalent to $L$ number of elevators, S(1) (with a total population of $SP(1)$), S(2) (with a total population of $SP(2)$), …, S($i$) (with a total population of $SP(i)$), …, S($L$) (with a total population of $SP(L)$), are then defined by:

S(1) : 1F, 2F, …, $f(1,2)$ F where $U_1 + U_2 + \ldots + r(1,2)*U_{f(1,2)} = SP(1)$ or $= U/L$ (when contract capacity of every elevator is identical to each other, which is the normal case.)

⋮

S($i$) : $f(i\text{-}1,i)$ F, ($f(i\text{-}1,i)$+1) F , …, $f(i,i+1)$ F where $(1-r(i-1, i))*U_{f(i-1,i)} + U_{f(i-1,i)+1} + \ldots + r(i,i+1)*U_{f(i,i+1)} = SP(i)$ or $= U/L$ (when contract capacity of every elevator is identical to each other) and so on for $i \in \{2, 3, \ldots, N-1\}$.

⋮

S($L$) : $f(L\text{-}1,L)$ F, ($f(L\text{-}1,L)$+1) F, …, $N$ F where $(1-r(L-1, L))*U_{f(L-1,L)} + U_{f(L-1,L)+1} + \ldots + U_N = SP(L)$ or $= U/L$ (when contract capacity of every elevator is identical to each other, which is normally the case).

Here, $r$ is the ratio, $0 \leq r < 0.999$, that indicates how the population of the boundary floor is divided between two contiguous sectors, the lower sector and the upper sector, and it is estimated based on a ratio of the population of that floor belonging to the lower sector to the total population of that floor. Therefore, $r$ cannot be equal to 100% because if the highest floor of a sector is solely owned by that sector, the boundary floor would be one floor higher.

In other words, the lowest floor of any sector must either be solely or partially owned by that sector while there could be zero population at the highest floor of that sector. This arrangement has been adopted for the convenience of being able to derive a formula for the highest reversal floor using an analytical approach. This formula will be discussed later in this paper. If the lowest floor of the $j$th sector is solely owned by the $j$th sector, $f(j-1, j)$ is this lowest floor and $r(j1, j) = 0$. This is much better than the other way round because if we allow the lowest floor of a sector to be vacant, it is more difficult to evaluate the highest reversal floor of that sector accurately. Moreover, it should be noted that the sum of population of all sectors must eventually be equal to the total population of the whole building, i.e. $SP(1) + SP(2) + \ldots + SP(i) + \ldots + SP(L) = U_1 + U_2 + \ldots + U_N = U$.

After all sectors are defined, $S(i)$ (expected number of stops) and $H(i)$ (highest reversal floor) of the $i$th sector can be given by traditional equations involving unequal floor population, as shown in equation sets (12) and (13). Finally, the round-trip time of the $i$th sector, $RTT(i)$ can be found by equation (14). And these follow the IOB defined in [16].

$$S(1) = f(1,2) - {}_{f} \quad {}_{(1,2)} \quad 1 \qquad SPU(^{j}1) \; P \rightarrow r(1,2)*U_{f(1,2)}$$

$$- \quad \text{where } U_{f(1,2)} \text{ here } _{j=1}$$

$$S(i) = f(i,i+1) - f(i-1,i) + 1 - _{f(\ i,i+1)} \quad 1 \quad SPU(^{j}i) \ _{P}$$

$$\rightarrow \left(1 - r(i-1,i)\right) U_{f(i-1,i)} \qquad (12)$$

$$- \quad \text{where } U_{f(i-1,i)} \text{ here}$$

$$_{j=f(i-1,i)}$$

for $i \in \{2,3,...,L-1\}$ and $U_{f(i,i+1)}$ here $\rightarrow r(i,i+1) U_{f(i,i+1)}$

$$N$$

$$S(L) = N - f(L-1,L) + 1 - \quad 1 - \overline{\quad} SPU(_{j}L) \ _{P} \quad \text{where } U_{f(L-1,L)} \text{ here } \rightarrow \left(1\text{-}r(L-1,L)\right) U_{f} \ _{(L-1,L)}$$

$$_{j=f(L-1,L)}$$

$$_{f(1,2)-1}$$

$$H(1) = f(1,2) - \quad _{j=1} \quad _{i=j1} \overline{\quad} SPU(_{i}1) \ _{P}$$

$$P$$

$$_{f(i,i+1)-1}$$

$$H(i) = f(i,i+1) - \quad _{j} \overline{\quad} SP^{U}(_{i}i) \qquad (13)$$

$$_{j=f(i-1,i)} \ _{i=-f(i\ 1,\ )i}$$

for $i \in \{2,3,4,...,L-1\}$

$$H(L) = N - _{j=f} \quad _{NL--1\ L} \quad _{i=fLj-\ L} \overline{\quad} SPU(_{i}L)$$

$$( \quad 1, ) \qquad ( \quad 1, )$$

For all $U$'s of the boundary floors in equation (13), the same definition in equation (12) applies. Finally, the *RTT* of every sector is evaluated by equation (14).

$$RTT(i) = 2H(i)t_{v} + (S(i)+1)(T - t_{v}) + 2Pt_{p} \qquad (14)$$

where $t_v = d_f / v$ is the time to travel one floor under rated speed; $T$ is the performance time (= $t_o + t_c + t_f(1)$ as conventionally defined); $P$ is the average number of passengers in the car during up-peak; $t_p$ is the passenger transfer time. They are all explained in full details in CIBSE Guide D [27].

**6.3 The Last Step - Methodology to evaluate f(i,i+1) and r(i,i+1), i running from 1 to L-1** This is the most important step of sectoring. Once this step is completed, *H*(i) and *S*(i) and then the *RTT*(*i*) of the *i*th sector can be evaluated in a straight forward manner by using conventional formulae (12) to (14).

We start from the 1st floor.

Check the *k*th floor where:

$$U_1 + \cdots + U_k \le SP(1) \text{ (or } U/L) \quad \text{and}$$

$$U_1 + \cdots + U_k + U_{k+1} > SP(1) \text{ (or } U/L)$$

Then, the $(k+1)$th floor $= f(1,2)$; and

$$r(1,2) = \frac{SP(1) \text{ (or } U/L) - (U_1 + \cdots + U_k)}{U_{k+1}}$$

Then, continue to check the 2nd sector by resetting $k$ to $f(1,2)$ and searching $k$ such that

$$(1 - r(1,2))U_{f(1,2)} + U_{f(1,2)+1} \cdots + U_k \le SP(2) \text{ (or } U/L) \quad \text{and}$$

$$(1 - r(1,2))U_{f(1,2)} + U_{f(1,2)+1} \cdots + U_k + U_{k+1} > SP(2) \text{ (or } U/L) \text{ Then,}$$

$(k+1)$th floor $= f(2,3)$; and

$$r(2,3) = \frac{SP(2) \text{ (or } U/L) - \left[(1 - r(1,2))U_{f(1,2)} + U_{f(1,2)+1} \cdots + U_k\right]}{U_{k+1}}$$

The whole procedure continues until $f(L-1,L)$ and $r(L-1,L)$ are identified.

In addition to the usefulness of this method to create sectors under any values of *N*, *L*, *P*, carby-car elevator capacity, and floor population distribution, there is one more advantage which is real time allocation of cars to serve landing calls generated at the main terminal. Once all boundary floors and the ratios *r*(*i*, *i*+1) are identified based on a knowledge of floor population distribution, $U_i$, $i$ = 1, …, *N*, *H* and *S* can be computed by the conventional formulae of unequal floor population, shown in equation sets (12) and (13). And the $H_{ratio}$ and $S_{ratio}$ of formulae (9) and

(10) can be further applied to produce a more reasonable and practical $H_{des}$ and $S$ for real time allocations.

As mentioned before, there are $L$ number of sectors for $L$ number of elevator cars. Any elevator can either be statically assigned to a particular sector or be dynamically assigned to sectors on the basis of alternative epochs. The latter method is to balance the *RTT* of different elevators of different sectors when every sector has the same total population. Dynamic allocation is invisible to the passengers demanding service under a DGC operational mode. What the passenger knows is the car assigned to his or her particular landing call and there is no way to judge whether such an assignment is optimal or not.

Obviously, a passenger going to a particular floor within a sector but not on the boundary floor is assigned to the car of that sector. If the landing call is to a boundary floor, say $f(i,i+1)$, of the $i$th and ($i$+1)th sectors. A dice is thrown with a probability of $r(i,i+1)$ to the lower sector and a probability of (1-$r(i,i+1)$) to the higher sector to determine which sector to which the landing call is assigned. That is a very fair arrangement.

**Figure 3: General Methodology of Sectoring a Building**

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | |
| $N$th F | | | | | | $U_N$ |
| | | | | | | |
| $f(L\text{-}1,L)$ th F | | | | | | $(1-r(L-1,L))*U_{f(L-1,L)}$ |
| | | | | | | |
| | | | | | | |
| $f(i,i+1)$th F | | | | $r(i,i+1)*U_{f(i,i+1)}$ | | |
| | | | | | | |
| $f(i-1,i)$th F | | | | $(1-r(i-1,i))*U_{f(i-1,i)}$ | | |
| | | | | | | |
| $f(2,3)$ th F | | $r(2,3)*U_{f(2,3)}$ | | | | |
| | | | | | | |
| | | $U_{f(1,2)+1}$ | | | | |
| $f(1,2)$ th F | $r(1,2)*U_{f(1,2)}$ | $(1-r(1,2))*U_{f(1,2)}$ | | | | |
| | | | | | | |

| | | | | | | |
|---|---|---|---|---|---|---|
| 2nd F | $U_2$ | | | | | |
| 1st F | $U_1$ | | | | | |
| | MT | MT | | MT | | MT |
| | L(1) | L(2) | | L($i$) | | L($L$) |
| | | | | | | |

## 7. NUMERICAL EXAMPLE

Three numerical examples are presented here to illustrate how the values of *H* and *S* of each sector and the average *H* and *S* of the whole system can be calculated. For simplicity, it is assumed that the passenger arrival rate to every floor is constant. The building has 18 or 19 floors above the ground floor or main terminal, served by three elevators, each having a capacity (assumed to be equal to *P*) of 13 passengers. It is assumed that the car is always full during up-peak. Population of every floor is either uniformly 30 passengers or non-uniform.

### 7.1 *N*/*L* is an integer and equal floor population distribution

```
N =  18, L =   3, U =   540 , CC =  13
U of every floor =  30,  30,  30,  30,  30,  30,  30,  30,  30,  30,  30,  30,  30,  30,  30,  30,  30,  30,
Sector  1 , LB =   1 , UB =   7 , Ratio(  1,  2) =  0.00
Sector  2 , LB =   7 , UB =  13 , Ratio(  2,  3) =  0.00
Sector  3 , LB =  13 , UB =  18

 Sector =  1 , UU =  30  30  30  30  30  30   0   0   0   0
 Sector =  2 , UU =  30  30  30  30  30  30   0   0   0   0
 Sector =  3 , UU =  30  30  30  30  30  30   0   0   0   0

Sector  1 , S =  5.4 , H =  5.9
Sector  2 , S =  5.4 , H =  11.9
Sector  3 , S =  5.4 , H =  17.9
Average S =  5.4 , Average H =  11.9
```

Every floor has a population of 30 passengers, a total building population of *U* = 18 x 30 = 540 passengers. It can be seen that Sector 1 occupies from 1/F to 7/F, 7/F being the boundary floor between Sector 1 and Sector 2. But there is no passenger at 7/F that belongs to Sector 1 due to $r(1,2) = 0$. So, effectively, Sector 1 is from 1/F to 6/F. Sector 2 is from 7/F to 13/F where 13/F is the boundary floor between Section 2 and Sector 3. Similarly, there is no passenger at 13/F that belongs to Section 2 due to $r(2,3) = 0$. Effectively, Section 2 is from 7/F to 12/F. Obviously, Sector 3 is from 13/F to 18F. The results are reasonable based on common sense as the situation is rather straight forward. *UU* is the number of passengers of each floor belonging to the corresponding sector.

### 7.2 *N*/*L* is not an integer and equal floor population distribution

```
N =  19, L =  3, U =  570 , CC =  13
U of every floor =  30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30,
Sector  1 , LB =  1 , UB =  7 , Ratio( 1, 2) =  0.33
Sector  2 , LB =  7 , UB =  13 , Ratio( 2, 3) =  0.67
Sector  3 , LB =  13 , UB =  19

 Sector =  1 , UU = 30  30  30  30  30  30  10  0  0  0
 Sector =  2 , UU = 20  30  30  30  30  30  20  0  0  0
 Sector =  3 , UU = 10  30  30  30  30  30  30  0  0  0

Sector  1 , S =  5.9 , H =  6.5
Sector  2 , S =  6.0 , H =  12.7
Sector  3 , S =  5.9 , H =  18.9
Average S =  5.9 , Average H =  12.7
```

Total building population *U* = 19 x 30 = 570 as there are 19 floors above main terminal. It can be seen that the boundary floors are at 7/F and 13/F, and *r*(1,2)=0.33, *r*(2,3)=0.67. That means 33% of the 30 passengers at 7/F belong to Sector 1 while the remaining 67% of the 30 passengers belong to Section 2. When a passenger wants to go to 7/F, a dice is played with a probability ratio of 0.33 and 0.67 for Sector 1 and Sector 2 respectively to determine whether this passenger shall be assigned to the elevator of Sector 1 or Sector 2.

### 7.3 *N*/*L* is not an integer and unequal floor population distribution

```
N =  19, L =  3, U =  604 , CC =  13
U of every floor =  34, 35, 27, 35, 32, 26, 28, 31, 35, 35, 27, 35, 35, 30, 34, 27, 30, 35, 33,
Sector  1 , LB =  1 , UB =  7 , Ratio( 1, 2) =  0.44
Sector  2 , LB =  7 , UB =  13 , Ratio( 2, 3) =  0.65
Sector  3 , LB =  13 , UB =  19

 Sector =  1 , UU = 34  35  27  35  32  26  12  0  0  0
 Sector =  2 , UU = 16  31  35  35  27  35  23  0  0  0
 Sector =  3 , UU = 12  30  34  27  30  35  33  0  0  0

Sector  1 , S =  5.9 , H =  6.5
Sector  2 , S =  5.9 , H =  12.8
Sector  3 , S =  5.9 , H =  18.9
Average S =  5.9 , Average H =  12.7
```

There are 19 floors above main terminal and three elevators, with a total population of 604. Since the floor population distribution does not vary significantly, the two boundary floors are still 7/F and 13/F but *r*(1,2) and *r*(2,3) cannot be visualized by common sense.

It is obvious that this method has the advantage that it can handle all three types of situations, with any combination of *N*, *L*, *P*, and floor population distribution.

## 8. APPLICATIONS IN THE DESIGN OF DESTINATION GROUP CONTROL SYSTEMS UNDER REAL TIME ALLOCATION

Based on the discussion so far, a designer of a DGC system can follow the following steps. For destination control, the number of sectors is equal to the number of elevators, and sectors comprise contiguous floors (i.e. Type 1 sectoring).

8.1 Understand the population distribution of every floor.

8.2 Determine whether *N*/*L* is an integer or not.

8.3 Evaluate the (*L*-1) boundary floors for all sectors, *f*(1,2), *f*(2,3),…., *f*(*L*-1,*L*) and calculate their ratios of population demarcation, *r*(1,2), *r*(2,3), …, *r*(*L*-1,*L*).

8.4 Based on the new floor population distribution, *UU*, of every sector, evaluate the *H* and *S* of every sector and assign this as the IOB value for H and S (from the formulae).

8.5 Use Table (7) to find out *Hratio* and *Sratio* directly if *N*/*L* is an integer or by interpolation, or by using formulae (9) and (10).

8.6 Apply the two ratios to the *H* and *S* of every sector, i.e. every elevator, obtained in step (8.4) to improve accuracy due to real time landing call allocations and non-uniform passenger arrival rate to each floor in one typical epoch.

8.7 Calculate the average *H* and average *S* and then the average *RTT* of all elevators.

8.8 Calculate the interval and handling capacity as if it were a conventional collective control system.

8.9 Revise the elevator characteristics to get the best interval and handling capacity desired.

## 9. CONCLUSIONS

Using the value of the round-trip time based on estimating the values of the average number of stops (S) and the highest reversal floor (H) remains a widely used method in elevator traffic design. Many elevator traffic system designers find the concept of S and H very intuitive and insightful.

This paper develops a methodology for evaluating the elevator round trip time under incoming traffic conditions and destination group control. A set of equations for H and S under destination group control has previously been derived for idealized optimal benchmark (IOB) conditions. However, it is accepted that the values of H and S from these equations cannot be used for the design of elevator traffic system as they under-estimate the actual values of the round-trip time as they ignore the practical issues related to allocating calls to elevator cars under real time conditions. There are a number of reasons for the difference, the most important of which is the real time allocation of the landing calls to the elevators in the group.

This paper uses the Monte Carlo simulation method in order to develop a set of lookup tables for the values of H and S under destination group control. These lookup tables provide a ratio for the increase in the values of H and S as a percentage compared to the value of H and S under the idealized optimal benchmark conditions (IOB). They can be used by elevator traffic system designers in order to find the value of the round-trip time under destination group control systems and thus offer a practical approach to designing elevator traffic systems under destination group control. In addition, curve fitting has been applied to the data in the table to allow an equation-based approach to finding the values of H and S based on the number of floors above the main entrance (N), the number of passengers boarding the elevator car in a round trip (P) and the number of elevators in the group (L).

## REFERENCES


[1] Jana Koehler, Daniel Ottiger, "An AI-based approach to destination control in elevators", AI Magazine, volume 23, number 3, 2002, Fall 2002, American Association for Artificial Intelligence, pp 59 – 78.

[2] Bruce Powell, "Important issues in up peak traffic handling", in Elevator Technology 4, International Association of Elevator Engineers, Proceedings of Elevcon '92, May 1992, Amsterdam, The Netherlands, pp 207-218.

[3] Fortune, J.W., 2002, "Electronic Up-Peak Booster Options", in *Elevator Technology 12*, proceedings of Elevcon 2002, Milan, Italy, June 2002, The International Association of Elevator Engineers, pp 84-90.

[4] Christy, Theresa, 2014, "An Evolution of Lift Passenger Grouping", in *Elevator Technology 20*, proceedings of Elevcon 2014, Paris, France, 8-10 July 2014, pp 41-50.

[5] W. L. Chan and Albert T. P. So, "Dynamic zoning for intelligent supervisory control", International Journal of Elevator Engineering, Volume 1, 1996, pp 47-59.

[6] W. L. Chan and Albert T. P. So, "Dynamic zoning in elevator traffic control", in Elevator Technology 6, International Association of Elevator Engineers, Proceedings of Elevcon '95, March 1995, Hong Kong, pp 133-140.

[7] Zhonghua Li, Zongyuan Mao and Jianping Wu, "Research on Dynamic Zoning of Elevator Traffic Based on Artificial Immune Algorithm", 8th International Conference on Control, Automation, Robotics and Vision", Kumming, China, 6th-9th December 2004.

[8] Albert T.P. So, J.K.L. Yu, "Intelligent supervisory control for lifts: dynamic zoning", Building Services Engineering Research & Technology, Volume 22, Issue 1, 2001, pp. 14-33.

[9] Suying Yang, Jianzhe Tai, Cheng Shao, "Dynamic partition of elevator group control system with destination floor guidance in up-peak traffic", Journal of Computers, Volume 4, Number 1, January 2009, pp 45-52.

[10] Richard Peters, "Understanding the benefits and limitations of destination dispatch", in Elevator Technology 16, The International Association of Elevator Engineers, the proceedings of Elevcon 2006, June 2006, Helsinki, Finland, pp 258-269.

[11] Rory Smith, Richard Peters, "ETD algorithm with destination dispatch and booster options", in Elevator Technology 12, International Association of Elevator Engineers, Proceedings of Elevcon 2002, Milan, Italy, 2002, pp 247-257.

[12] Schröder J. (1990), "Advanced dispatching: destination hall calls + instant car-to-call assignments: m10", Elevator World, March, pp. 40-46.

[13] Schröder J. (1990), "Elevatoring calculation: probable stops and reversal floor: "m10" destination hall calls + instant car assignment", Elevator World, April, pp. 52-54.

[14] Fernandez, Cortes. "A survey of Elevator Group Control Systems for Vertical Transportation". *IEEE Control Systems Magazine*.

[15] Al-Sharif L. Introduction to Elevator Group Control (METE XI). *Lift Report* 2016; 42(2):59-68.

[16] Al-Sharif L, Hamdan J, Hussein M, Jaber Z, Malak M, Riyal A, AlShawabkeh M and Tuffaha D. Establishing the Upper Performance Limit of Destination Elevator Group Control Using Idealised Optimal Benchmarks. *Building Services Engineering Research and Technology*, 2015; 36(5):546-566. September 2015.

[17] G.C. Barney, "Uppeak revisited", in Elevator Technology 4, International Association of Elevator Engineers, Proceedings of Elevcon '92, Amsterdam, The Netherlands, May 1992, pp 39-47.
[18] G.C. Barney, "Uppeak, down peak and interfloor performance revisited", in Elevator Technology 9, Proceedings of Elevcon '98, Zurich, Switzerland, published by the International Association of Elevator Engineers, pp 31-40.
[19] Lutfi Al-Sharif, Mohamed Hussein, Moh'd Malak, Daoud Tuffaha, "The Use of Numerical Methods to Evaluate the Performance of Up Peak Group Control Algorithms", Proceedings of Elevcon 2014, Paris, 8th to 10th July 2014, pp 327-337.
[20] Al-Sharif L. Riyal A., Jaber Z. and Hamdan J., "Assessing the up-peak performance of destination elevator group control systems using real-time allocation of landing calls", Int. J. Industrial and Systems Engineering, Vol. 25, No. 4, 2017, pp. 443-460.
[21] Al-Sharif L, Yang Z S, Hakam A and Abd Al-Raheem A. Comprehensive analysis of elevator static sectoring control systems using Monte Carlo simulation. Building Serv. Eng. Res. Technol. 2018. Published online 8th January 2018. DOI: 10.1177/0143624417752644
[22] Joris Schroeder, "Elevatoring calculation, probable stops and reversal floor, "M10" destination halls calls + instant call assignments", in Elevator Technology 3, Proceedings of Elevcon '90, International Association of Elevator Engineers, 1990, pp 199-204.
[23] Janne Sorsa, Henri Hakonen, Marja-Liisa Siikonen, "Elevator selection with destination control system", in Elevator Technology 15, International Association of Elevator Engineers, Proceedings of Elevcon 2005, Peking, China.
[24] Jorg Lauener, "Traffic performance of elevators with destination control", Elevator World, September 2007, pp 86-94.
[25] Lutfi Al-Sharif and Mohamed D. Al-Adhem, "The current practice of lift traffic design using calculation and simulation", *BUILDING SERV ENG RES TECHNOL* July 2014 vol. 35 no. 4, pp 438-445, published online 26 September 2013, doi: 10.1177/0143624413504422.
[26] Lutfi Al-Sharif, Mohamed D. Al-Adhem, "Overview and Comparison of the Four Methods of Elevator Traffic Analysis and Design", Elevator World, Virtual Elevator Event 2012, 11th July 2012.
[27] CIBSE, "CIBSE Guide D: Transportation systems in buildings", published by the Chartered Institute of Building Services Engineers, Fifth Edition, 2015.
[28] Lutfi Al-Sharif, Osama F. Abdel Aal, Ahmad M. Abu Alqumsan, "Evaluating the Elevator Passenger Average Travelling Time under Incoming Traffic Conditions using Analytical Formulae and the Monte Carlo Method", Elevator World, June 2013, pp 110-123.
[29] Bruce A. Powell, "The role of computer simulation in the development of a new elevator product", Proceedings of the 1984 Winter Simulation Conference, page 445-450, 1984.
[30] C. M. Tam, Albert P. C. Chan, "Determining free elevator parking policy using Monte Carlo simulation", International Journal of Elevator Engineering, Volume 1, 1996.
[31] Lutfi Al-Sharif, Hussam Dahyat, Laith Al-Kurdi, "The use of Monte Carlo Simulation in the calculation of the elevator round trip time under up-peak conditions", *Building Services Engineering Research and Technology, volume 33, issue 3 (2012) pp. 319– 338, doi:10.1177/0143624411414837*.
[32] Barney G C and Al-Sharif L. *Elevator Traffic Handbook: Theory and Practice*. 2nd Edition. Routledge, Abingdon-on-Thames, United Kingdom. 2016.