A Computer-Aided Design for the Specification and Performance Evaluation of Elevator Systems

Y.C. Cho[a], J.H. Park[a], H.C. Chang[a], Zavarin Gagov[a], W.H. Kwon[a]

C.S. Kim[b], P.H. Han[b] and J.H. Seo[b].

[a] *Control Information Systems Lab., School of Electrical Engineering, Seoul National University, Seoul, 151-742, Korea*

[b] *Building Systems Research Laboratory, LG-OTIS Inc. Inchon, Korea*

Key Words: Computer-aided design, elevator system, simulation, genetic algorithms

ABSTRACT

This paper presents a computer-aided design tool (ESES-Tool) which helps system designer to decide the specification of elevator systems and predict system performances given building environment and objectives. Owing to various possible combinations of building conditions such as group divisions, available service floors and car dynamics, the searching space is inherently huge, which makes it difficult to decide appropriate (or optimal) specification of the elevator system. A genetic algorithm is employed to find near-optimal elevator specifications and to reduce the searching time. The fast simulation with event-driven techniques tests the searching result. Also, system performances such as mean, max, variance of HRT (hall call response time) and passengers' service time are also analyzed under several system options at the simulation stage of ESES-tool.

1. INTRODUCTION

A large number of modern elevator systems have been installed in most high-rise buildings. Though elevator machine itself has been improved fast in accordance with the development of mechanical and electrical engineering, the planning of elevator installation has usually depended on the reference of map tables or experiences of experts. But as for high-rise buildings, the appropriate installation of elevators is so important to attain overall success of the elevator system (Terazono and Matsukura 1994). Especially, as the demands of end-users, i.e. building owners or managers, become more and more diverse and complex, traditional map table method (CIBSE 1993) is not suitable to decide the "specification of the elevator system" simply called *elevator specification* in this paper.

Given a building condition such as available floors, population, building purpose, a number of elevator group, etc., the number of all the possible elevator specifications is very huge even in a small sized building. Thus, it is necessary to predict the system performance for each elevator specification and to efficiently find best candidates among them. After the decision of the elevator specification, it should be tested whether the elevator specification satisfies the detail performance objectives of the system.

Previous computer-aided design (CAD) was mainly focused on traffic analysis and performance evaluation of elevator control policies (Barney 1977). In this paper, we propose a computer-aided design tool to search the optimal elevator specification as well as to evaluate its performance. This tool consists of two design stages: one stage is called "*elevating stage*" and the next stage is "*simulation stage*". The function of elevating stage is to find near-optimal elevator specifications when building condition and system objective are given. Owing to various combination of building conditions, it is difficult to search the optimal elevator specification among huge state space. A genetic algorithm is employed to find near-optimal elevator specifications and to reduce the searching time in the CAD tool. The result of elevating stage is stored in a file and used as input data at the simulation stage. The function of simulation stage is not only to test the basic performance of the selected elevator specification but also to evaluate its detail performance under other additional options such as bank division, group control algorithms, etc. Also, the evaluated result at the simulation stage is feeded back to the elevating stage in order to find more optimal elevator specifications. The whole procedure of the suggested CAD tool (called *ESES-tool* standing for Elevating and Simulation of Elevator Systems) is shown in Fig 1.

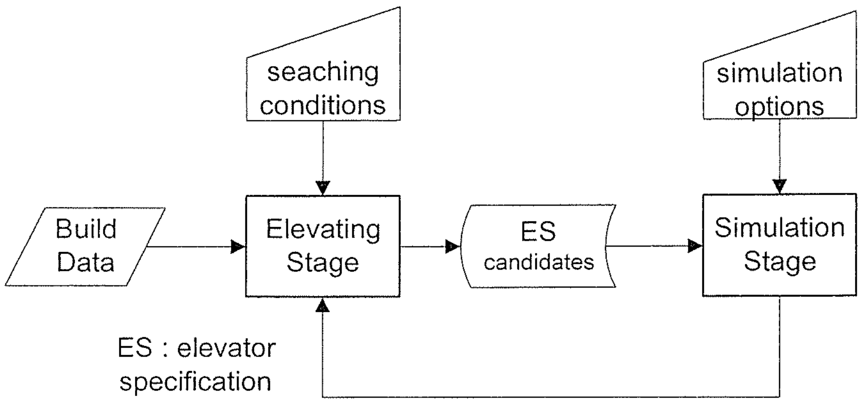

Fig. 1. The procedure of ESES-Tool

In Section 2, the function of the elevating stage is explained and an efficient searching method is suggested using genetic algorithm (GA). In Section 3, the function and unique features of the simulation stage is explained. In Section 4, the result is summarized.

2. ELEVATING : OPTIMAL DESIGN OF ELEVATOR SYSTEMS

The most important problem in the elevating stage of ESES-Tool is how many groups should be installed in the building and how many elevators should be assigned in each group to achieve the reasonable performance. To determine the appropriate number of groups and the number of elevators, the traffic demand must be pre-estimated. The traffic demand can be expressed by a ratio of the number of passengers using elevators during five minutes to the number of total population in the building, which ratio is dependent on the kind and purpose of the building.

In elevating stage, two kinds of objectives should be satisfied: quantitative and qualitative objectives. The quantitative objective is expressed handling capacity (*HC*), i.e. the number of passengers to be transported during five minutes. *HC* is formulated with round trip time (*RTT*) and the number of passengers served in a round trip (*P*) as follows:

$$HC = \frac{(5\,\mathrm{min}) \times (60\,\mathrm{sec/min}) \times P}{RTT} \qquad (1)$$

Next, we must consider the qualitative objective related to the waiting times of passengers on the floors, which are dependent on the group control employed in the elevator system. But, at this elevating stage, we simplify the passengers' waiting time in a rough way to be independent on the individual group control. Note that the detail evaluation is performed at the simulation stage. The qualitative objective is represented with average operation interval (*INT*) as follows:

$$INT = \frac{RTT}{N} \qquad (2)$$

where, N is the number of elevators in a group to be determined.

2.1 Input and Output of Elevating Stage

ESES-Tool receives system data and searching conditions and reference table data as input data as shown in Table 1. System data consist of building data and basic elevator group data.

Table 1. Input of Elevating stage

Category	Contents	
System Data	Building Data	Building type, the number of floors, the height of each floor, population of each floor and the number of groups.
	Elevator Group Data	Door type, door width and available service floors for each group
Searching Conditions	Searching scope of the number of elevators, car capacity and car speed. Desired values of handling capacity (HC) and average operation interval (INT). The number of candidated elevator specification.	
Reference Table data	Building type table, speed table, door table and capacity table	

And, the output of elevating stage is both elevator specifications of best candidates and their performance (*HC, INT*). The components of elevator specification are shown in Table 2.

Table 2. Output of Elevating Stage

Category	Contents
Elevator specification of each candidate	The number of cars in each group.
	Car speed in each group (we assume the speed of cars in one group are identical.)
	Car capacity in each group (we assume the speed of cars in one group are identical.)
Basic performance of each candidate	Handling capacity (*HC*) and the average operation interval (*INT*) of each group

2.2 Estimation of Round Trip Time

The performance measures to be estimated are the handling capacity (*HC*) and the average operation interval (*INT*). They both are calculated separately for each group given by eqn (1) and eqn (2) respectively. According to the two equations, the round trip time *RTT* should to be calculated firstly under the up-peak traffic situation. *RTT* is the sum of the traveling time in normal service area (T_n), the traveling time in express area (T_e), the door access time (T_d), passenger delay time (T_p), and the losing time (T_l) :

$$RTT = T_n + T_e + T_d + T_p + T_l \tag{3}$$

T_n is the time spent during the movement of elevator in the normal service area. We assume the speed of elevator persists in a constant except acceleration/deceleration times. Then, T_n can be formulated with the car dynamics and number of stops (S_u for upword and S_d for downword) in a round trip which is calculated by the formulas from (Gaver and Powell 1971). On the other hand, since the elevator does not stop in the express area, T_e is only dependent on the mean distance of the express area. The door access time (T_d) is proportional to the number of stops in a round trip. The passenger delay time (T_p) is a function of the number of passengers (P) served in a round trip and the number of stops (S). Besides P and S, door width also influences the delay time. But, detailed formulation of T_p is omitted here. The losing time implies the unexpected overhead occurring during the stop of elevator, caused by the passengers movement and door operations. We assume here that the losing time is proportional to the sum of T_d and T_p. By experiments, the losing time is approximately 10 percent of the sum of T_d and T_p.

2.3 An Efficient Searching Algorithm

The handling capacity (*HC*) and the average operation interval (*INT*) is calculated for every distinct elevator specification from the available searching conditions. Assume that the number of available elevators is *M*, the number of available car speeds is *V*, the number of available car capacities is *E* and the number of groups is *G*. Then, the total amount of distinct specifications is $M^G \times V^G \times E^G$. This implies the amount of searching processes is increasing exponentially with respect to the number of groups. For example, $M=V=E=4$ and $G=3$ generates $2^{18}=262144$.

In this section, we present an efficient searching algorithm based on the genetic algorithm (GA) that can be performed in a polynomial time. The genetic algorithm is the process of constructing mother gene set to have smaller performance errors with the iterative process of crossover, mutation and evaluation (Gen and Cheng 1997). As the iteration goes further, the mother gene set is converging into the minimum error group. The mutation process is needed to avoid the localization problem by accepting the randomly chosen genes as the parent genes of the next generation.

The typical gene structure is shown in Fig. 2 where each chromosome of the gene becomes the elevator specification of an elevator group (i.e. number, speed and capacity of the group).

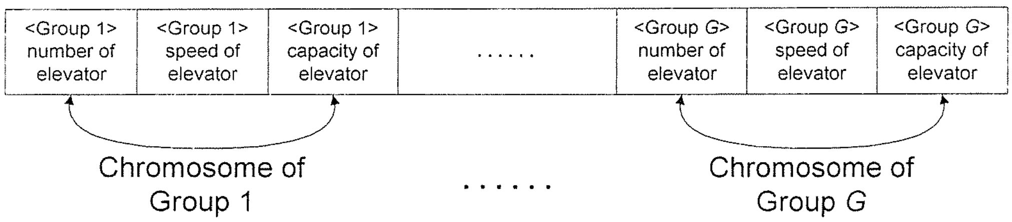

Fig 2. The structure of a gene

The genetic algorithm consists of the four basic operations: the selection of parent genes from a mother gene set, crossover, mutation and the creation of new mother gene set.

(a) Selection of the parent genes from a mother gene set
Among the mother gene set, select two parent genes that have the minimum and the maximum performance error. Here, performance error is defined as follows:

$$err = | HC_e - HC | \tag{4}$$

where, HC_e is the estimated handling capacity and *HC* is the desired handling capacity.

(b) Crossover
Let two parent genes be g_1 and g_2. The children genes are generated through a crossover of g_1 and g_2. The scheme of the crossover is as follows: For each chromosome field of a group, we generate two children gene such that the value of each chromosome field becomes the average between those of g_1 and g_2 without any change of the other chromosomes. Then, for one chromosome, two children genes are generated and for the number of groups *G*, the total children genes of $2\times3\times G=6G$, are generated for a pair of parents.

(c) Mutation
The mutation process is to insert a randomly chosen gene as a parent gene. With the existing two parent genes, all the three parent genes are selected. Thus, the combination of two parents among three parents are three cases and the total children genes generated by the three parents are ${}_3C_2\times6G = 18G$.

(d) Creation of new mother gene set

Among original mother gene set and children genes, new mother gene set is created with the genes which has smallest performance errors, *err*, in eqn (4) and satisfy $INT_e > INT$ as many as the number of candidated elevator specification. Here, INT_e is the estimated average operation interval and *INT* is the desired value. The searching algorithm with GA is shown in Table 3.

Table 3. Searching Algo. using GA

```
Input : initial mother gene set Mg, number of candidates and building conditions
Output : Mg
While (not done or i < Max_Iteration)
        Step 1 : Select two parent genes from Mg ;
        Step 2 : (Mutation) insert a randomly chosen gene to parent genes;
        Step 3 : (Crossover) make children gene set Mc;
        Step 4 : Compute HCe and INTe for each child gene;
        Step 5 : make new mother group M'g from Mg and Mc;
        Step 6 : if (Mg == M'g ) done=TRUE;
                 Else Mg ← M'g ;
        Step 7 : i = i+1;
  End
```

In a crossover, total $18 \times G$ of children genes are generated and used for the calculation of performance error. Additionally three parent genes are added for the calculation. Thus, $(18G+3)$ calculations are performed at one iteration time. The maximal process count of the genetic algorithm is $Max\_Iteration \times (18 \times G+3)$. The comparison between the blind search (BS) and the genetic algorithm (GA) is shown in Table 4 by setting $MAX\_Iteration = 1000$. Figure 3 shows a captured window of the elevating stage in ESES-Tool.

Table 4. Comparison of the two searching algorithms

TEST #	Search Method	Total number of processes	Time consumption (On Pentium 200Mhz)
1	BS	8503056	~ 40 min
	GA	75000	8 sec
2	BS	262144	45 sec
	GA	14600	< 1 sec

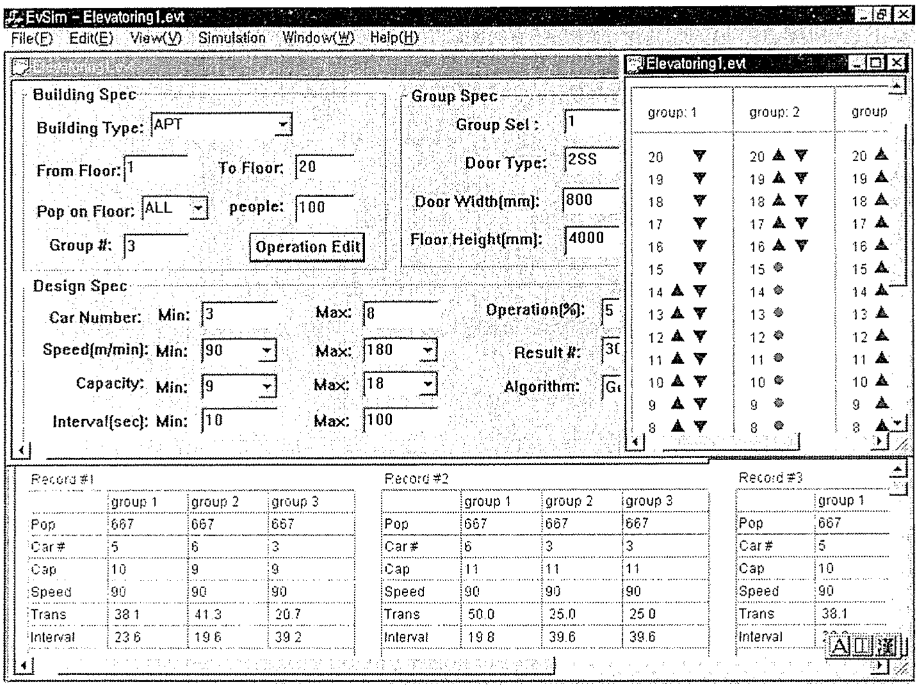

Figure 3. A window for elevating stage in ESES-Tool

3. SIMULATION : PERFORMANCE EVALUATION

At simulation stage, system performances of each elevator specification obtained at the elevating stage are evaluated. In addition to the basic performance evaluation, detail system performance is tested under various system options.

3.1 Input of Simulation Stage

The input of the simulation stage of ESES-Tool is listed in Table 5. Especially, the hall call and car call bank division is necessary to divide the normal service area and express service area. Many modern buildings are employing the bank division options.

Table 5. Input of Simulation stage

Category	Contents
Building Data	Number of floors, height, Number of cars in each group.
Traffic Data	Population, Concentration, Riding/alighting time, traffic patterns
Car Dynamics Data	Max Speed, Max acceleration, Jerk, door access time
System Options	Hall call/car call bank division, group control algorithm options

3.2 Features of Simulation Kernel

To reduce the simulation time under various system options, we adopt event-driven simulation method where system behaviors are classified with a number of events and states (Cassandras 1993). The transitions between the states are determined by the condition of current state. In the simulation kernel, Event scheduler manages the list of events in the order of its occurrence time as shown in Fig 4.

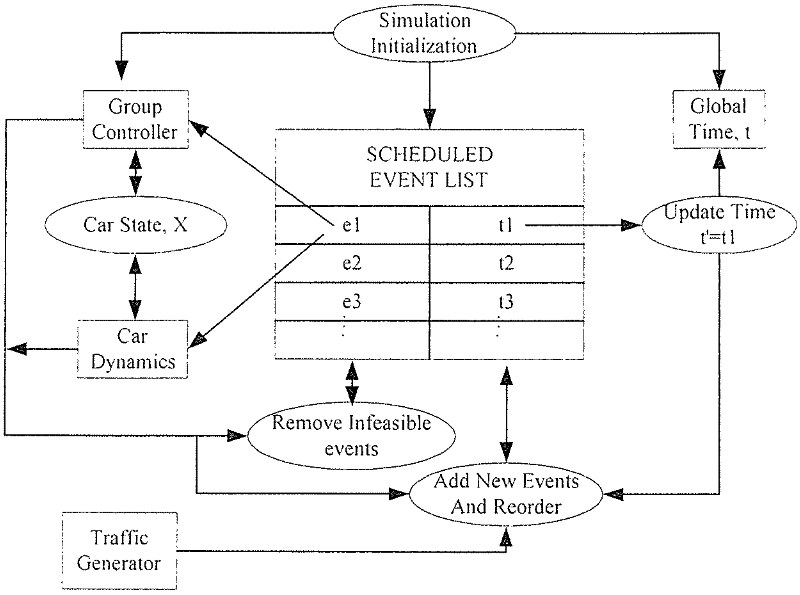

Figure 4. Structure of Event Scheduler

The traffic generator in Fig. 4 generates random passenger events labeled PASS_EVNT with attributes of occurrence time, origin floor and destination floor according to the traffic pattern. In addition, the number of bank is attached to the PASS_EVNT based on the bank division. The event makes a following event labeled HC_EVNT. In HC_EVNT event, the optimal car is selected according to the group control algorithm. Some kinds of algorithms can be chosen according to the request of the end-users. The following event is AHC_EVNT in which the hall call assignment information is delivered to the corresponding car. Each car manages the list of assigned hall call and car calls. The car itself generates CAR_EVNT. This event can have several attributes according to the car direction, door status. CAR_EVNT generates another CAR_EVNT after the car reaches to the next service floor.

The traveling time between two floors is calculated based on the *Molz' formulae* of ideal lift kinematics (Molz 1991) which enable minimum travel times, taking to account maximum values of jerk, acceleration and speed. If the traveling distance is too short for the elevator contract speed or acceleration given at the simulation input stage, the maximum speed and acceleration attained during the trip may be calculated.

In addition, the ESES-tool supports both visual animation mode and quick time mode. In visual animation mode we can validate the correctness by eyes and, in quick time mode, we can obtain the results promptly without visualization delay. The result of the simulation stage contains many performance indices as follows: passenger service time, passenger journey time, hall call response time (HRT), long-HRT, etc. After simulation, the information is recorded in files. The another advantage of simulation stage is that we can compare several group control algorithms with standard reference algorithms and analyze each control algorithm. We implemented three reference algorithms called worst distance assignment (WDA), minimum sum assignment (MSA) and load balance assignment (LBA) as well as new group control algorithm (Cho et al 1999). Figure 5 shows a captured window at the elevating stage in ESES-Tool.

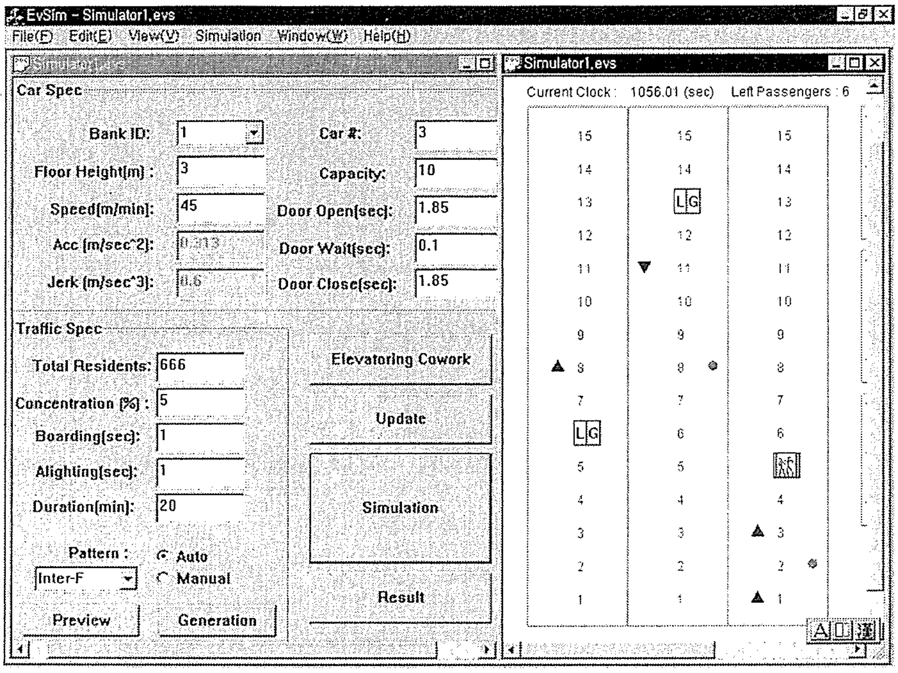

Figure 5. A window for simulation stage in ESES-Tool

4 CONCULSION

This paper proposed a computer-aided design tool (ESES-Tool) in order to find optimal specifications of elevator systems given specific building environments and evaluate their performances. This tool consists of two design stages: *elevating stage* and *simulation stage*. At the elevating stage, a genetic algorithm is adopted in order to find near-optimal elevator specifications which reduced search times dramatically. The result of the elevating stage is used as input data to the next simulation stage. The function of simulation stage is not only to test the basic performance of the selected elevator specification but also to evaluate the detail performance under other additional options. ESES-Tool will help system designers to plan the elevator installation more easily and to react faster the demands of end-users, i.e. building owners or building managers. Additional advantage of ESES-Tool is that this tool can be employed to analyze traffic flows or to develop new elevator group control algorithms.

REFRENCES

1. Barney, G.C. (1977). *Lift Traffic Analysis Design and Control*, Peter Peregrinus Ltd.
2. Cassandras C.G. (1993). Discrete Event Systems: Modeling and Performance Analysis, Irwin Inc, Boston.
3. Cho, Y.C., K.H. Kim *et al* (1999). Optimal Group Control of Elevator Systems by Statistic Approximation of Hall Call Waiting Times," *Proceedings of 14th IFAC World Congress*, China, pp. 103-108.
4. CIBSE (1993). *CIBSE Guide D: Transportation Systems in Buildings.*
5. Gaver, D.P. and Powell B.A. (1971). Variability in round trip times for an elevator car during up peak, *Transportation Reseach*, Vol 4, no. 4, pp.301-307.
6. Gen, N. and Cheng, R. (1997). *Genetic Algorithms and Engineering Design*, John Wiley & Sons, Inc. NY, pp. 3-40.
7. Molz H. (1991). On the ideal kinematics of lifts, *Elevatori*, no. 1, pp. 41-46.
8. Terazono, N. and Matsukura, Y (1994). *Elevator High Technology: The Challenge of the Highest Speed in the World*, Ohmsha Ltd., Japan.