https://doi.org/10.20965/jaciii.2010.p0487

Paper:

A Double-Deck Elevator Systems Controller with Idle Cage Assignment Algorithm Using Genetic Network Programming

Shingo Mabu, Lu Yu, Jin Zhou, Shinji Eto, and Kotaro Hirasawa

Graduate School of Information, Production and Systems, Waseda University
2-7 Hibikino, Wakamatsu-ku, Kitakyushu, Fukuoka 808-0135, Japan
E-mail: {mabu@aoni., yulu1006@fuji., zhoujin@asagi., eto@suou., hirasawa@}waseda.jp,
[Received October 9, 2009; accepted April 22, 2010]

So far, many studies on Double-Deck Elevator Systems (DDES) have been done for exploring more efficient algorithms to improve the system transportation capacity, especially in a heavy traffic mode. The main idea of these algorithms is to decrease the number of stops during a round trip by grouping the passengers with the same destination as much as possible. Unlike what occurs in this mode, where all cages almost always keep moving, there is the case, where some cages become idle in a light traffic mode. Therefore, how to dispatch these idle cages, which is seldom considered in the heavy traffic mode, becomes important when developing the controller of DDES. In this paper, we propose a DDES controller with idle cage assignment algorithm embedded using Genetic Network Programming (GNP) for a light traffic mode, which is based on a timer and event-driven hybrid model. To verify the efficiency and effectiveness of the proposed method, some experiments have been done under a special down-peak pattern where passengers emerge especially at the 7th floor. Simulation results show that the proposed method improves the performance compared with the case when the cage assignment algorithm is not employed and works better than six other heuristic methods in a light traffic mode.

Keywords: double-deck elevator systems, evolutionary computation, genetic network programming

1. Introduction

To meet the demand of transportation capacity in high-rise buildings without adding any more elevator installation spaces, Double-Deck Elevator Systems (DDES) have been invented in 1970's. Due to its specific hardware configuration, i.e., two decks vertically connected in one shaft, the transportation capacity of DDES can be significantly improved in a pure up-peak traffic pattern, where passengers are efficiently grouped at the lobby floors. On the other hand, some specific features caused by its hardware configuration make the DDES controller much more intractable giving poor performances in some traffic patterns other than the pure up-peak traffic pattern. A lot of research has been done not only on the control algorithm but also on the hardware of DDES. Recently, more and more DDESs have been installed in high-rise buildings all over the world by employing some collective approaches [1] as well as some hardware enhancements like Destination Floor Guidance System (DFGS), monitoring cameras, and so on.

The elevator control system is a classical NP hard problem. It is the problem of finding the best solution from all feasible solutions. It is very hard to find the global optimal. Many Artificial Intelligence (AI) technologies [2–5] have been applied to these kinds of problems. In recent years, many heuristic methods with learning and evolution have been found to be preferable for realizing such systems. As typical examples, Fuzzy Logic, Neural Network (NN), Reinforcement Learning (RL), Genetic Algorithm (GA) were successfully applied to elevator systems, and solutions are obtained, that can satisfy the demands of customers. Genetic Network Programming (GNP) which is an extension of GA and Genetic Programming (GP) was also verified to be applicable to the DDES in our past studies. GNP can realize a rule based elevator control system due to its directed graph structure, which makes the elevator system more flexible in different traffics. Unlike a linear solution that must cover multiple episodes, e.g., produced by a genetic algorithm, a directed graph can encapsulate the naturally recurring patterns in elevator operation for reuse throughout the problem instance. Also, the reusability of nodes makes the structure of GNP more compact than the tree structure of GP. Most of the studies on DDES control algorithms [6] including our past research [7, 8] have focused on the improvement of DDES performances in the heavy traffic mode, where all cages almost always keep moving to assigned calls. However, an overall evaluation of DDES is to be made not only in the heavy, but also in the light traffic mode. Since some cages become idle in the light traffic mode, how to dispatch these idle cages, which is seldom considered in the heavy traffic mode, becomes important when developing the controller of DDES. In this paper, we propose a DDES controller with idle cage assignment algorithm using GNP for the light traffic mode. Some experiments are done to verify the efficiency of the proposed method using a DDES simulator.

This paper is organized as follows. An overview of DDES is given in the next section, and the details of the proposed method are described in section 3. Section 4

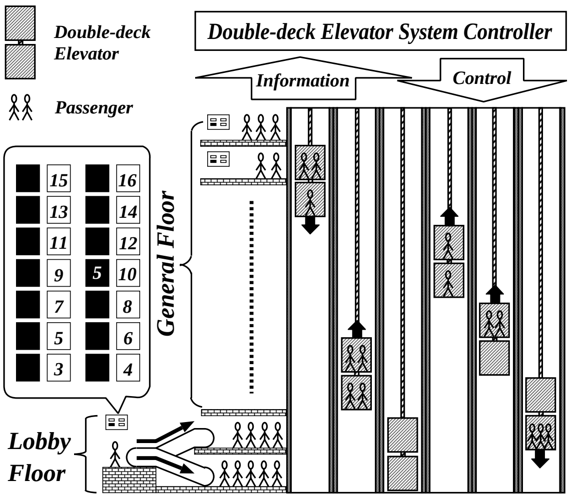


Fig. 1. Outline of DDES.

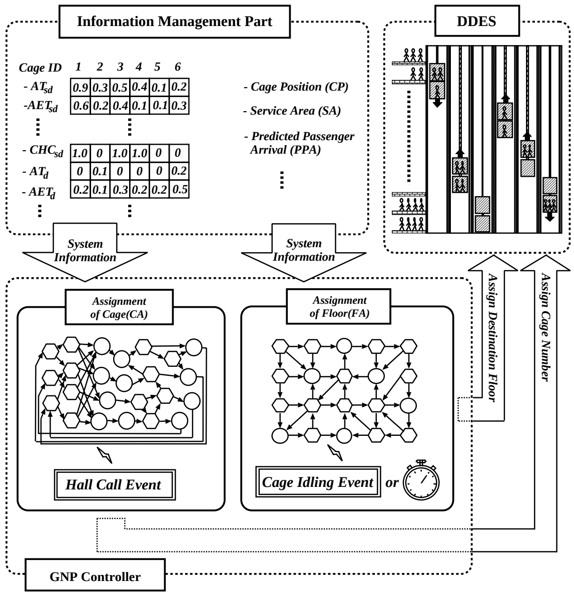


Fig. 2. Outline of the proposed method.

shows the simulation results and some discussions. Finally, some conclusions are made in section 5.

2. Outline of Double-Deck Elevator Systems

The outline of Double-deck Elevator Systems (DDES) is shown as **Fig. 1**. In DDES, two cages are connected vertically in each shaft whose size is the same as the conventional Single-deck Elevator Systems (SDES). It allows that the passengers at two consecutive floors could be serviced simultaneously. To guide the passengers efficiently, one or more escalators are usually installed at the entrance of the lobby floors. In order to obtain more information on passengers' destination, the conventional hall/cage call system has been replaced by Destination Floor Guidance System (DFGS) [9, 10], where the cage call buttons inside the cage are removed and DFGS is set at each floor instead of up/down buttons. The passenger could input there destination before they enter into the cages. Then, the system guides the passenger to an elevator that will be stopping at their destination floor.

There are three kinds of running modes in DDES. They are (1) *Double running mode*, in which the upper/lower cage serves only odd/even floors respectively; (2) *Semi-double running mode*, in which both two cages can serve any floor except for the two lobby floors; and (3) *Single running mode*, in which one of cages is set to be out of service. Note that semi-double running mode can provide a more flexible service while it makes the control algorithm more complex.

In our past studies, a GNP controller of DDES has been proposed for the semi-double running mode, and its performance has been verified under a moderately heavy traffic density, where all cages always keep moving to assigned calls. The traffic density, however, does not keep high during the work hours in typical high rise office buildings. Some cages become idle when the DDES runs in a light traffic mode, and they are usually requested to stay there until they are assigned to a new hall call. An idle cage assignment algorithm of how to dispatch these idle cages is proposed in this paper for some performance improvements.

3. Idle Cage Assignment Algorithm for DDES Controller Using GNP

Since how to dispatch the idle cages is important for a DDES controller in a light traffic mode, we added an floor assignment algorithm to the DDES controller using Genetic Network Programming (GNP). **Fig. 2** shows the outline of the proposed method, where the GNP controller consists of two parts, i.e., Cage Assignment algorithm (CA) and Floor Assignment algorithm (FA). The *immediate assignment policy* is employed in Cage Assignment algorithm as our past studies did, that is to say, the optimal cage is assigned based on the current information of DDES such as the values of evaluation items of each cage shown in the left side of "Information Management Part" (for more details, see [8]) and the assignment is not changed later. Contrary to this kind of event-driven model, Floor Assignment algorithm is proposed by a timer and event-driven hybrid model, that is, FA is invoked by a preset timer or a cage idling event to assign the idle cage to the optimal destination floor where it should move to.

3.1. Overview of GNP

As a new evolutionary computation method, Genetic Network Programming (GNP) has been proposed around ten years ago. In contrast to the string structure of Genetic Algorithm (GA) and the tree structure of Genetic Programming (GP), GNP has a directed graph structure.

The efficiency and effectiveness of GNP have been verified in several studies [11, 12].

The evolutionary process in GNP is almost the same as the one in other evolutionary computation methods, which is briefly described as follows.

- Initialize the population of the first generation.
- Evaluate each individual based on the fitness after task execution.
- Generate the population of the next generation by executing the genetic operations.
- Iterate 2–3 until the terminal condition is satisfied.

In this paper, "Uniform Crossover,"[1] "Branch Mutation"[2] and "Elite Preservation" are used as the genetic operators of GNP.

3.2. Evaluation Items

To determine the optimal floor of idle cages, several evaluation items, i.e., $X \in \{CP, SA, PPA\}$ are proposed as follows. They will be used in the judgment nodes of GNP, and their functions are described later.

Cage Position (CP) To avoid the bunching mode [13] of elevator group systems, which is reportedly linked to a poor performance, the positions of all cages are considered when determining the optimal floor for the idle cages.

Service Area (SA) With the same reason, the service area of each cage is defined for Floor Assignment algorithm since the states of each cage including the moving direction and moving speed are also some important factors.

Predicted Passenger Arrival (PPA) In an ideal case, the idle cages should be dispatched to the floor, where new passengers would arrive in the near future if we could precisely predict the next passenger arrival. PPA is another valuable factor of the proposed evaluation.

3.3. Main Algorithm

Figure 3 shows the flowchart of the proposed GNP controller. After the controller is started, GNP will be invoked by some events including the call event and cage idling event or a preset timer. The Cage Assignment algorithm (CA) is invoked and the cage to serve is determined when a hall call with DFGS occurs by a passenger. On the other hand, the Floor Assignment algorithm (FA) is invoked either when a cage becomes idle or the timer makes a time up. Since there might be more than one idle cages in DDES, each idle cage will be checked whether to stay at the current position or move to a specified floor by FA.

1. Some nodes in two parent GNPs are selected by probability P_c, then the branches from the corresponding nodes are exchanged and two new offspring GNPs are generated.
2. Some branches in a GNP are selected by probability P_m, then the connections of them are changed randomly and a new GNP is generated.

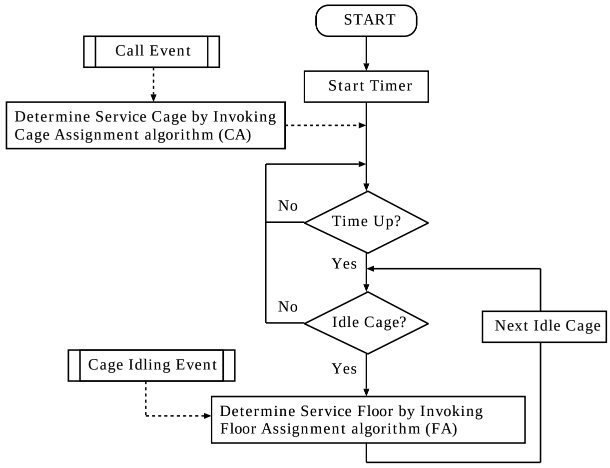


Fig. 3. Flowchart of the proposed GNP controller.

Since the Floor Assignment algorithm is the main point of this paper, we employ the same Cage Assignment algorithm which has been proposed in our past research [8].

When the FA of the GNP controller is invoked, the current position of the idle cage is judged firstly with 3 results, i.e., {Base, General-Low, General-High}, which represents where the idle cage is. Then, the processing nodes on the transition route of GNP in the floor selection part are activated and the evaluation values of the candidate floors are calculated based on the following Eq. (1). This evaluation function to be maximized can be calculated with different evaluation item X until the node transition transfers to the floor assignment part.

$$e(f) = \sum_{p \in P} w_p \cdot X_p(f), \qquad (1)$$

where,

P : set of suffixes of nodes transited in the floor selection part (P is determined by node transition)

w_p : weights of the floor selection processing node p (w_p is optimized during the evolutionary process)

$X_p(f)$: evaluation function of floor f at floor selection processing node P

The evaluation functions $X_p(f)$ of floor f are shown in **Fig. 4**. In $CP(f)$, f_0 represents the current position of the cage running upward. Since the area behind the upward running cage in the figure is very hard to serve as floor assignment floors by this cage in the near future, so, the high priority is given to the floors behind the upward running cage. On the other hand, the low priority is given to the floors ahead the running cage. The function should be reversed when the cage runs downward. In $SA(f)$, f_0 represents the current position of the cage running upward, and f_1 represents the next stop of the cage. Since the floors around the next stop of the cage are serviced by the cage, its function value is set to the lowest one. In $PPA(f)$, the

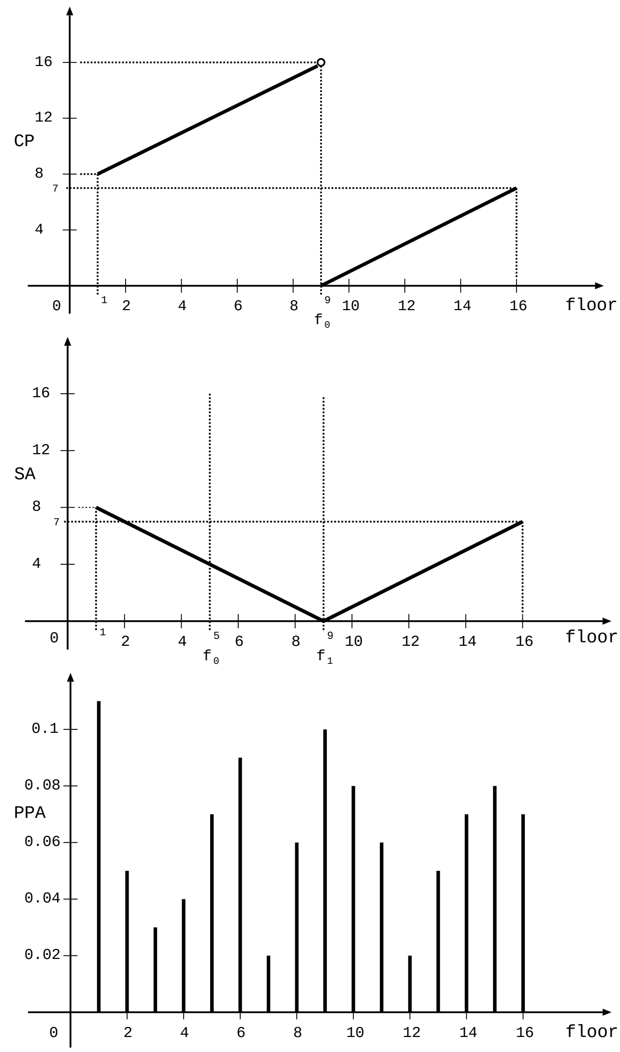


Fig. 4. Evaluation functions of floors.

rate of the passengers emerged at each floor during the past 30 minutes is used to predict the next passenger arrival. The higher the rate of the floor is, the more the floor is considered as the service floor of the idle cage. The sum of the rate is 1.0. In order to balance the influence of *PPA* with *CP* and *SA*, the value of *PPA* is multiplied by 100 in advance.

The following Eq. (2) is used to determine the service floor of the idle cage when the floor assignment node is activated.

$$f = \arg \max_{f \in F} e(f), \quad \ldots\ldots\ldots\ldots \quad (2)$$

where, F : set of the number of floors.

3.4. Node Functions

The node functions in the proposed method are defined as follows.

Idle Cage Position Judgment Node

- Judge the current position of the idle cage ({Base, General-Low, General-High}).

Candidate Floor Selection Node

- Calculate the evaluation values of the candidate floors based on Eq. (1).

Floor Assignment Node

- Assign the idle cage to serve floor f.

3.5. Fitness Function

The same items, which have been proposed in our past research, are employed to evaluate the fitness F of each FA individual. As shown in Eq. (3), the first two items, average waiting time and maximum waiting time, are minimized for better performance. The third item is minimized to provide more comfortable riding service [8], while the last one is minimized to eliminate the loop gene of GNP [8].

$$F = \frac{1}{N}\sum_{n=1}^{N}(t_n)^2 + w_t \cdot (t_{\max})^2 + w_c \cdot (n_c)^2 + w_l \cdot (n_l)^2, \quad (3)$$

where,

N : total number of passengers

t_n : waiting time of n-th passenger

$t_{\max}$: maximum waiting time among N passengers

n_c : total number of passengers experiencing one cage service

n_l : number of loops of GNP per one hour evaluation

w_t, w_c, w_l : weighting coefficients

Since the items in the function are expected to minimize, the smaller the fitness value of the individual is, the better the performance becomes. The weights are determined empirically referring to [8], which means that they are determined by finding the best values using various simulations, in other words, by considering the balance among the average waiting time, the maximum waiting time, the ratio of the long waiting and average system time to be explained in section 4.3.

Moreover, the loop of GNP occurs when the accumulation of the time delays of nodes and transitions exceed the time delay threshold value, which should be avoided.

4. Simulation Results and Discussions

4.1. DDES Simulator

The DDES simulator was built based on the specifications shown in **Table 1**. All events are simulated in detail by using 0.1 *second* time unit. In each time unit, the events of passengers such as arriving at floors, pushing the button of DFGS, getting on and off the cage, are generated according to the O/D table shown in **Table 2**, which represents a typical down peak pattern. In this paper, a more complicated traffic pattern is simulated by setting the 7th

Table 1. Specifications of DDES simulator.

Items	Value
Number of Floors	16
Number of Shafts (Cages)	6 (12)
Floor Distance [m]	4.5
Max. Velocity [m/s]	2.5
Max. Acceleration [m/s^2]	0.7
Jerk [m/s^3]	0.7
Cage Capacity [person]	20
Time for Opening Door [s]	2.0
Time for Closing Door [s]	2.3
Time for Riding [s/person]	1.0
Passenger Density [person/h]	200

Table 2. Relative traffic flows.

Dest. Floor / Origin Floor	LF	7F	ReGF
Lobby Floor(LF)	–	2	2
7th Floor(7F)	25	–	2
Rest of General Floors(ReGF)	10	1	1

floor, where a high passenger arrival rate is set compared to the other general floors. **Table 3** shows the running parameters of the proposed method.

Table 3. Running parameters of floor assignment GNP.

Items	Value
Generation	300
Population Size	300
—Crossover	120
—Mutation	170
—Elite	10
Node Size	30+Initial Node
Crossover Rate	0.1
Mutation Rate	0.01
Evaluation Time [hour]	2
w_t, w_c, w_l	0.007, 0.001, 0.6

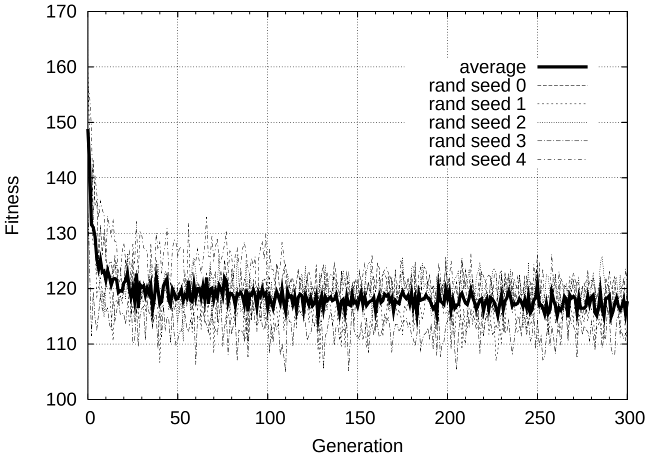


Fig. 5. Fitness curves of the proposed method.

4.2. Fitness Curves

Figure 5 shows the fitness curves of the best individuals during the evolutionary process of GNP with floor assignment individuals. In order to reduce the influence of random noises, we did the experiments using 5 random seeds. Fitness curves of all 5 random seeds are listed in **Fig. 5** as well as the average one. Note that the population of GNP controller was optimized generation by generation and converged to a certain level at the latter generations of the evolutionary process.

4.3. Performance Comparisons

To confirm the generalization ability of the proposed method, the best individual of each rand seed obtained in the above evolutionary process was tested on the same DDES simulator for 30 times with 2 simulated hours. To verify the efficiency and effectiveness of the proposed method, the performance comparisons have been done firstly between the proposed method and the method called *Non-FA Method*. In *Non-FA Method*, there is only cage assignment algorithm, i.e., the elevator system without floor assignment algorithm for the idle cage, that is to say, the cage will stay at the last floor after it serves all registered hall and cage calls.

Moreover, there are six other heuristic methods proposed for some further performance comparisons in this paper. They are *SA Method*, *SA+CP Method*, *SA+CP+PPA Method*, *Fixed-FA(1F) Method*, *Fixed-FA(7F) Method*, *Fixed-FA(7F*) Method* and *Fixed-FA(16F) Method*. In *SA Method*, only $SA(f)$ is used in the proposed method to determine the service floor of the idle cage. Similarly, $SA(f)+CP(f)$ and $SA(f)+CP(f)+PPA(f)$ are used in $e(f)$ of Eq. (1) in *SA+CP Method* and *SA+CP+PPA Method*, respectively. On the other hand, the idle cages are dispatched to the 1st floor automatically in *Fixed-FA(1F) Method*, while 7th floor in *Fixed-FA(7F) Method*[3] and 16th floor in *Fixed-FA(16F) Method*. In addition, *Fixed-FA(7F*) Method*, an enhanced version of *Fixed-FA(7F) Method*, is considered, where 1 of 6 cages is dispatched to the lobby floor, 1 to the 16th floor and the remaining 4 to the 7th floor.

The performances of each method are listed in **Table 4**, where the best individual obtained during the evolutionary process under 200 persons/h is tested not only under 200 persons/h, but also under 100 and 300 persons/h to confirm its generalization ability. There are four performance criteria, *LWR*, *AWT*, *AST* and *MWT* [14–17]. *LWR* represents the ratio of the long waiting, i.e., the ratio of passengers who wait more than 60 s. *AWT* represents the

3. This method is efficient as expected since the rate of passenger emergence at the 7th floor is much larger than the one at other floors as shown in **Table 2**

Table 4. Performance comparison of different methods in simulations.

Passenger Density	Methods	LWR	Imp.	AWT	Imp.	AST	Imp.	MWT	Imp.
100	Proposed Method	**0.01**	**75%**	**9.9**	**9.2%**	**31.1**	**3.1%**	29.9	24.7%
	Non-FA Method	0.04	0%	10.9	0%	32.1	0%	39.7	0%
	SA Method	0.02	50%	11.9	-9.2%	33.6	-4.7%	34.8	12.3%
	SA+CP Method	0.03	25%	11.8	-8.3%	33.5	-4.4%	36.9	7.1%
	SA+CP+PPA Method	0.03	25%	12.8	-17.4%	34.5	-7.5%	39.8	-0.3%
	Fixed-FA(1F) Method	0.06	-50%	18.2	-67%	39.7	-23.7%	47.3	-19.1%
	Fixed-FA(7F) Method	0.02	50%	11.6	-6.4%	32.5	-1.2%	30.2	23.9%
	Fixed-FA(7F*) Method	0.02	50%	12.4	-13.8%	33.1	-3.1%	**29.6**	**25.4%**
	Fixed-FA(16F) Method	0.09	-125%	16.6	-52.3%	38.3	-19.3%	48.6	-22.4%
200	Proposed Method	**0.01**	**75%**	**10.3**	**17%**	**33.3**	**6.5%**	36.8	18.4%
	Non-FA Method	0.04	0%	12.4	0%	35.6	0%	45.1	0%
	SA Method	0.02	50%	11.9	4%	35.6	0%	41.5	8%
	SA+CP Method	0.02	50%	12.1	2.4%	35.8	-0.6%	43.5	3.5%
	SA+CP+PPA Method	0.04	0%	13.0	-4.8%	36.7	-3.1%	44.6	1.1%
	Fixed-FA(1F) Method	0.05	-25%	16.9	-36.3%	40.6	-14%	48.3	-7.1%
	Fixed-FA(7F) Method	0.02	50%	11.6	6.5%	34.3	3.7%	35.3	21.7%
	Fixed-FA(7F*) Method	0.02	50%	12.1	2.4%	34.9	2%	**35.1**	**22.2%**
	Fixed-FA(16F) Method	0.03	25%	15.3	-23.4%	39.1	-9.8%	47.9	-6.2%
300	Proposed Method	**0.04**	**20%**	**10.9**	**12.2%**	**35.5**	**0.3%**	46.7	-3.6%
	Non-FA Method	0.05	0%	12.4	0%	35.6	0%	45.1	0%
	SA Method	0.06	-20%	12.3	0.4%	37.6	-5.8 %	52.0	-15.5%
	SA+CP Method	0.08	-60%	12.6	-2.0%	38.0	-6.8 %	55.1	-22.2%
	SA+CP+PPA Method	0.05	0%	13.3	-7.8%	38.8	-8.9%	53.3	-18.2%
	Fixed-FA(1F) Method	0.05	0%	16.3	-31.6%	41.8	-17.3%	50.5	-12.0%
	Fixed-FA(7F) Method	0.03	40%	11.8	5.1%	35.8	-0.6%	42.2	6.3%
	Fixed-FA(7F*) Method	0.04	20%	12.2	1.3%	36.4	-2.3 %	**41.3**	**8.5%**
	Fixed-FA(16F) Method	0.1	-100%	14.5	-17.0%	39.9	-12.2%	56.3	-25.0%

Note: $Imp.$ is defined by $\frac{Value_{\text{Non-FA}} - Value_x}{Value_{\text{Non-FA}}}$, $x \in$ {Proposed Method, SA, SA+CP, SA+CP+PPA, Fixed-FA(1F), Fixed-FA(7F), Fixed-FA(7F*), Fixed-FA(16F)}.

average waiting time of all passengers during the test period. *AST* represents the average system time of all passengers, which is the sum of the average waiting time and the average travelling time. MWT represents the maximum waiting time during the test period.

Table 4 shows that the proposed method outperforms the *Non-FA Method* and six other heuristic methods except MWT by the *Fixed-FA(7F) Method* and *Fixed-FA(7F*) Method* on all four performance criteria. The $Imp.$ columns of each performance criteria show the percentage improvement of each method comparing with the *Non-FA Method*. The worst performances of the *Fixed-FA(1F) Method* on all four criteria suggest that inappropriate floor assignment for the idle cage, in this case the idle cage is dispatched to the 1st floor regardless of the down-peak traffic pattern, will deteriorate the performance to some extent. Note that the performances of *SA Method* and *SA+CP Method* are almost the same, which suggests that only adding *CP* to *SA* for the floor assignment does not impact the overall performances a lot. Furthermore, the performances of *Fixed-FA(7F) Method* and *Fixed-FA(7F*) Method* show that the latter one underperforms on AWT and AST though it works a bit better on MWT than the former one. This is reasonable since dispatching all idle cages to the 7th floor, where the rate of passenger emergence is very large, in *Fixed-FA(7F) Method* could shorten the AWT and AST, while dispatching one of idle cages to the lobby floor and one to the 16th floor in *Fixed-FA(7F*) Method* could shorten the MWT by servicing many more floors.

In order to compare the performances of the different methods, we used the statistical test analysis of variance, named ANOVA. It gives a statistical test on whether the means of several groups, in our case, 9 different methods, are all equal or not. We used 5% level of significance. The various quantities in ANOVA are summarized in **Table 5**. DF is the degree of freedom for the sum of the squares between different methods. F value is the ratio of the model mean square to the error mean square. $Pr(F)$ is the probability to judge how strong the hypothesis that all the means are equal is supported. The smaller the $Pr(F)$ is, the stronger the hypothesis is supported. From **Table 5**, it is found that AWT in all 9 different methods are not equal, under the three kind of passengers' density. Also, the same statistical tests have been done for AST. Then, the Student-Newman-Keuls (SNK) is used to compare all

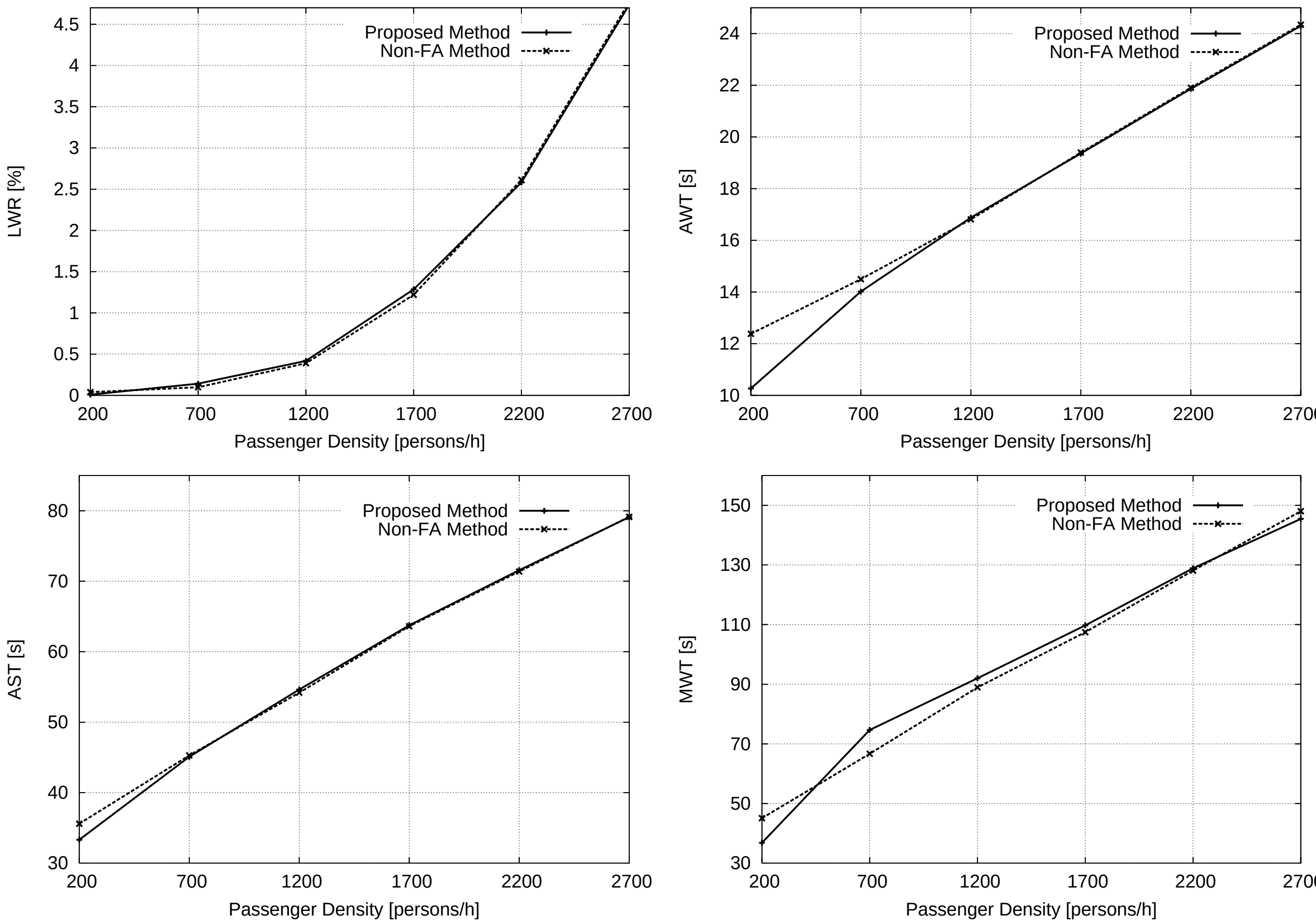


Fig. 6. Performance comparisons under different passenger density.

Table 5. ANOVA of different methods.

	DF	Sum of Sq	Mean Sq	F Value	Pr(F)
Passenger Density 100					
AWT	8	290.683	36.335	1894.445	0.0000000
AST	8	333.793	41.724	1587.426	0.0000000
Passenger Density 200					
AWT	8	165.133	20.642	889.259	0.0000000
AST	8	214.002	26.750	785.322	0.0000000
Passenger Density 300					
AWT	8	103.761	12.970	413.797	0.0000000
AST	8	186.420	23.302	501.630	0.0000000

the means based on ANOVA. The means are ordered from small to large and divided into several subsets. **Table 6-8** shows where the difference between two subsets is statistically significant.

In addition, as mentioned earlier, the idle cage assignment algorithm embedded controller is proposed especially for a light traffic mode where some idle cages occur. That is to say, the proposed method would not contribute to the system performances when it is employed in a heavy traffic mode, because there are almost no idle cages in such a mode. Furthermore, the idle cage assignment algorithm even deteriorate the performance a bit when there are only few idle cages in a moderate traffic mode, because the cage movement due to the idle cage assignment weakens the flexibility of the idle cage movement in terms of not serving the new passenger in two directions, which could be implicitly linked to a larger MWT. **Fig. 6** confirms the above discussions, where experimental results under various passenger densities are done. The frequencies of idle cage occurrence in each passenger density are listed in **Table 9**. Also, **Fig. 6** shows that the proposed method works well under the passenger density less than 450 persons/h.

5. Conclusions

In this paper, we proposed an idle cage assignment algorithm of the DDES controller for the light traffic mode where some idle cages exist. Three evaluation items are proposed to determine the service floor for the idle cage, and it is selected with optimized weights during the node transition of the proposed GNP. Fitness curves show that the evolutionary process was implemented generation by generation. The best individuals are firstly applied to a

Table 6. Student-Newman-Keuls in 100 person/h.

<Average Waiting Time>

100 person/h	Subset for alpha = 0.05							
Method	1	2	3	4	5	6	7	8
Proposed	9.9							
Non-FA		10.9						
Fixed-FA(7F)			11.6					
SA+CP				11.8				
SA				11.9				
Fixed-FA(7F*)					12.4			
SA+CP+PPA						12.8		
Fixed-FA(16F)							16.6	
Fixed-FA(1F)								18.2
Pr(F)	1.0	1.0	1.0	0.27	1.0	1.0	1.0	1.0

<Average System Time>

100 person/h	Subset for alpha = 0.05							
Method	1	2	3	4	5	6	7	8
Proposed	31.1							
Non-FA		32.1						
Fixed-FA(7F)			32.5					
Fixed-FA(7F*)				33.1				
SA+CP					33.5			
SA					33.6			
SA+CP+PPA						34.5		
Fixed-FA(16F)							38.3	
Fixed-FA(1F)								39.7
Pr(F)	1.0	1.0	1.0	1.0	0.80	1.0	1.0	1.0

Table 7. Student-Newman-Keuls in 200 person/h.

<Average Waiting Time>

200 person/h	Subset for alpha = 0.05						
Method	1	2	3	4	5	6	7
Proposed	10.3						
Fixed-FA(7F)		11.6					
SA			11.9				
SA+CP			12.1				
Fixed-FA(7F*)			12.1				
Non-FA				12.4			
SA+CP+PPA					13.0		
Fixed-FA(16F)						15.3	
Fixed-FA(1F)							16.9
Pr(F)	1.0	1.0	0.07	1.0	1.0	1.0	1.0

<Average System Time>

200 person/h	Subset for alpha = 0.05						
Method	1	2	3	4	5	6	7
Proposed	33.3						
Fixed-FA(7F)		34.3					
Fixed-FA(7F*)			34.9				
Non-FA				35.6			
SA				35.6			
SA+CP				35.8			
SA+CP+PPA					36.7		
Fixed-FA(16F)						39.1	
Fixed-FA(1F)							40.6
Pr(F)	1.0	1.0	1.0	0.11	1.0	1.0	1.0

Table 8. Student-Newman-Keuls in 300 person/h.

<Average Waiting Time>

300 person/h	Subset for alpha = 0.05						
Method	1	2	3	4	5	6	7
Proposed	10.9						
Fixed-FA(7F)		11.8					
Fixed-FA(7F*)			12.2				
SA			12.3				
Non-FA			12.4				
SA+CP				12.6			
SA+CP+PPA					13.3		
Fixed-FA(16F)						14.5	
Fixed-FA(1F)							16.3
Pr(F)	1.0	1.0	0.34	1.0	1.0	1.0	1.0

<Average System Time>

300 person/h	Subset for alpha = 0.05						
Method	1	2	3	4	5	6	7
Proposed	35.5						
Non-FA	35.6						
Fixed-FA(7F)	35.8						
Fixed-FA(7F*)		36.4					
SA			37.6				
SA+CP				38.0			
SA+CP+PPA					38.8		
Fixed-FA(16F)						39.9	
Fixed-FA(1F)							41.8
Pr(F)	0.63	1.0	1.0	1.0	1.0	1.0	1.0

Table 9. Frequency of idle cage occurrence.

Passenger Density	200	700	1200	1700	2200	2700
Freq. of Idle Cage	4.5	3.7	1.9	0.7	0.3	0.2

Note: The frequency of idle cage occurrence represents the number of idle cages per minute.

light traffic mode (100, 200 and 300 persons/h) and compared with the *Non-FA Method* and six other heuristic methods. The efficiency and effectiveness of the proposed method have been verified by the performance comparisons. Furthermore, the proposed method has been applied to various traffic modes from light to heavy to clarify its applicable conditions. In addition, ANOVA is used to prove that the difference of the performance is not caused by the selection of different samples. The proposed method has been shown to have essential differences to the alternative methods for this domain.

References:

[1] M. -L. Siikonen, "Double-deck elevators: Savings in Time and Space," Elevator World, Vol.46 No.7, pp. 65-69, 1998.

[2] H. Aoki and K. Sasaki, "Group supervisory control system assisted by artificial intelligence," Elevator World, pp. 70-80, Feb. 1990.

[3] C. B. Kim, K. A. Seong, H. Lee-Kwang, J. O. Kim, and Y. B. Lim, "A fuzzy approach to elevator group control system," IEEE Trans. Syst., Man, Cybern., Vol.25, pp. 985-990, 1995.

[4] D. Levy, M. Yadin, and A. Alexandrovitz, "Optimal control of elevators," Int. J. Syst. Sci., Vol.8, No.3, pp. 301-320, 1977.

[5] Y. W. Ho and L. C. Fu, "Dynamic Scheduling Approach to Group Control of Elevator Systems with Learning Ability," In Proc. of the 2000 IEEE Int. Conf. on Robotics and Automation, San Francisco, CA, pp. 2410-2415, April 2000.

[6] J. Sorsa, M. -L. Siikonen, and H. Ehtamo, "Optimal control of double-deck elevator group using genetic algorithm," Int. Trans. in Operations Research, Vol.10, No.3, pp. 103-114, 2003.

[7] J. Zhou, L. Yu, S. Mabu, K. Hirasawa, J. Hu, and S. Markon, "Double-deck Elevator Systems using Genetic Network Programming with Reinforcement Learning," In Proc. of 2007 IEEE Congress on Evolutionary Computation (CEC2007), pp. 2025-2031, Singapore, Sept. 2007.

[8] K. Hirasawa, T. Eguchi, J. Zhou, L. Yu, and S. Markon, "A Double-Deck Elevator Group Supervisory Control System Using Genetic Network Programming," IEEE Trans. on Systems, Man and Cybernetics, Part C, Vol.38, No.4, pp. 535-550, Aug. 2008.

[9] J. Koehler and D. Ottiger, "An AI-based approach to destination control in elevators," AI Magazine, Vol.23, No.3, pp. 59-79, 2002.

[10] S. Tanaka, Y. Innami, and M. Araki, "A study on objective functions for dynamic operation optimization of a single-car elevator system with destination hall call registration," In Proc. IEEE Int. Conf. on Systems, Man and Cyvernetics, 2004.

[11] T. Eguchi, K. Hirasawa, J. Hu, and N. Ota, "A Study of Evolutionary Multiagent Models Based on Symbiosis," IEEE trans. on Systems, Man and Cybernetics, Part B, Vol.35, No.1, pp. 179-193, Feb. 2006.

[12] S. Mabu, K. Hirasawa, and J. Hu, "A Graph-Based Evolutionary Algorithm: Genetic Network Programming and Its Extension Using Reinforcement Learning," Evolutionary Computation, MIT Press, Vol.15, No.3, pp. 369-398, 2007.

[13] G. Barney, "Elevator Technology," Ellis Horwood Limited, West Sussex, England, 1986.

[14] G. Barney, "Elevator Traffic Handbook," Spon Press, 2003.

[15] R. D. Peters, "The theory and practice of general analysis of lift calculations," In Proc. of the ELEVCON 1992, pp. 197-206, 1992.

[16] G. R. Strakosch, "Vertical Transportation: Elevators and Escalator," New York: Wiley, 1983.

[17] K. Sasaki, S. Markon, and M. Nakagawa, "Elevator group supervisory control system using neural networks," Elevator World, No.2, pp. 81-86, 1996.

Name:
Shingo Mabu

Affiliation:
Assistant Professor, Graduate School of Information, Production and Systems, Waseda University

Address:
2-7 Hibikino, Wakamatsu-ku, Kitakyushu, Fukuoka 808-0135, Japan

Brief Biographical History:

2003 Received the M.E. degree from Kyushu University
2006 Received Ph.D. degree Waseda University
2006-2007 Visiting Lecturer at Waseda University
2007- Assistant Professor at Waseda University

Main Works:

• "A Graph-Based Evolutionary Algorithm: Genetic Network Programming (GNP) and Its Extension Using Reinforcement Learning," Evolutionary Computation, MIT press, Vol.15, No.3, pp. 369-398, 2007.

Membership in Academic Societies:

• The Institute of Electrical and Electronics Engineers, Inc. (IEEE)
• The Society of Instrument and Control Engineers (SICE)
• The Institute of Electrical Engineers of Japan (IEEJ)

Name:
Lu Yu

Affiliation:
Ph.D. Student, Graduate School of Information, Production and Systems, Waseda University

Address:
2-7 Hibikino, Wakamatsu-ku, Kitakyushu, Fukuoka 808-0135, Japan

Brief Biographical History:
2005 Received the B.E. degree from Beijing Institute of Technology, China
2007 Received the M.E. degree form Waseda University, Japan
2007- Ph.D. Student at Graduate School of Information, Production and Systems, Waseda University, Japan

Main Works:
• "Double-deck Elevator Group Supervisory Control System Using Genetic Network Programming with Ant Colony Optimization," Proc. of IEEE Congress on Evolutionary Computation 2007 (CEC2007), Singapore, pp. 1015-1022, Sept. 2007.
• "A Double-deck Elevator Group Supervisory Control System with Destination Floor Guidance System using Genetic Network Programming," IEEJ Trans. EIS, Vol.127, No.7, pp. 1115-1122, July 2007.

Membership in Academic Societies:
• The Institute of Electrical and Electronics Engineers, Inc. (IEEE)
• The Association for Computing Machinery (ACM)

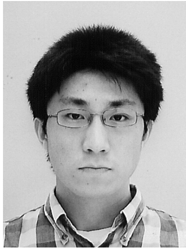
Name:
Jin Zhou

Affiliation:
Graduate School of Information, Production and Systems, Waseda University

Address:
2-7 Hibikino, Wakamatsu-ku, Kitakyushu, Fukuoka 808-0135, Japan

Brief Biographical History:
1996 Joined BHH Corp.
2006 Received M.E. degree from Waseda University
2009 Received Ph.D. degree Waseda University

Main Works:
• "A study of applying Genetic Network Programming with Reinforcement Learning to Elevator Group Supervisory Control System," in Proc. of IEEE Congress on Evolutionary Computation 2006, Vancouver, Canada, pp. 10392-10398, July 2006.



Name:
Shinji Eto

Affiliation:
Graduate School of Information, Production and Systems, Waseda University

Address:
2-7 Hibikino, Wakamatsu-ku, Kitakyushu, Fukuoka 808-0135, Japan

Brief Biographical History:
2005 Received the M.E. degree from Waseda University
2008 Received Ph.D. degree Waseda University
2009- Visiting Research Associate at Waseda University

Main Works:
• "Realizing Functional Localization using Genetic Network Programming with Importance Index," J. of Advanced Computational Intelligence and Intelligent Informatics, Vol.10, No.4, pp. 555-566, 2006.

Membership in Academic Societies:
• The Institute of Electrical and Electronics Engineers, Inc. (IEEE)
• The Society of Instrument and Control Engineers (SICE)
• The Institute of Electrical Engineers of Japan (IEEJ)

Name:
Kotaro Hirasawa

Affiliation:
Professor, Graduate School of Information, Production and Systems, Waseda University

Address:
2-7 Hibikino, Wakamatsu-ku, Kitakyushu, Fukuoka 808-0135, Japan

Brief Biographical History:
1966 Joined Hitachi Corp.
1992- Professor at Kyushu University
2002- Professor at Waseda University

Main Works:
• "A Double-Deck Elevator Group Supervisory Control System Using Genetic Network Programming," IEEE Trans. on Systems, Man and Cybernetics, Part C, Vol.38, No.4, pp. 535-550, July 2008.
• "A Study of Evolutionary Multiagent Models Based on Symbiosis," IEEE Trans. on Systems, Man and Cybernetics, Part B, Vol.35, No.1, 2006.
• "A Function Localized Neural Network with Branch Gates Neural Networks," Neural Networks, Vol.16, No.10, pp. 1461-1481, December 2003.
• "Chaos Control on Universal Learning Networks," IEEE Trans. on Systems, Man, and Cybernetics, PART C, Vol.30, No.1, pp. 95-104, January 2000.

Membership in Academic Societies:
• The Institute of Electrical and Electronics Engineers, Inc. (IEEE)
• The Association for Computing Machinery (ACM)
• The Society of Instrument and Control Engineers (SICE)
• The Institute of Electrical Engineers of Japan (IEEJ)