# A Genetic Algorithm for Controlling Elevator Group Systems

P. Cortes, J. Larrañeta, and L.Onieva

Ingeniería Organización. Escuela Superior Ingenieros. Seville University, Camino de los Descubrimientos s/n. Sevilla 41092. Spain
pca@esi.us.es

**Abstract.** The efficient performance of elevator group system controllers becomes a first order necessity when the buildings have a high utilisation ratio of the elevators, such as in professional buildings. We present a genetic algorithm that is compared with traditional controller algorithms in industry applications. An ARENA simulation scenario is created during heavy lunchpeak traffic conditions. The results allow us to affirm that our genetic algorithm reaches a better performance attending to the system waiting times than THV algorithm.

## 1 Introduction

The installation of synchronized elevator groups in professional use buildings (offices, hospitals or hotels) is an usual practice. The large utilisation ratio of the elevators makes necessary the implementation of such systems in order to give quality to the users-passengers and energetic efficiency to the building managers.

In this situation, it is usual to select the system waiting time as the goal to attain an efficient system performance. Also, the maximum waiting time has to be had in account as an additional limitation. The system waiting time includes the waiting time for the lift in the hall plus the trip time inside the lift.

When dealing with elevator systems is a usual practice to consider the following assumptions (where most of them are evident assumptions). Each hall call is attended by only one cabin. The maximum number of passengers being transported in the cabin is bounded by its capacity. The lifts can stop at a floor only if it exists a hall call or a cabin call in that floor. The cabin calls are sequentially served in accordance with the lift trip direction. A lift carrying passengers cannot change the trip direction.

The traditional type of controllers implemented in the industry follows simple dispatch rules that make use of an IF-ELSE logical commands set. Among these dispatch rules, the THV is one of the most habitual algorithms. The THV assigns the hall call to the nearest lift in the adequate trip direction [1].

Recently, advanced methods have been proposed showing better performance. Examples of them are the *Optimal Routing algorithm*, the *Dynamically Adaptive Call Allocation* (DACA) and the *Adaptive Call Allocation* (ACA) [2] that are based on Dynamic Programming. Also Fuzzy Logic has been proved as a valuable alternative when evaluating a large amount of criteria in a flexible manner. The fuzzy elevator

J. Mira (Ed.): IWANN 2003, LNCS 2687, pp. 313-320, 2003.

group control system [3] and the Fuzzy Elevator Group Controller with Linear Context Adaptation [4] are some examples. Bio-inspired systems [5] and [6] have been revealed successful capacities. Here we propose a genetic algorithm based on a hall call allocation strategy to identify the chromosomes of the population individuals.

The rest of the paper deals with the simulation model in section 2, the genetic algorithm characteristics in section 3, the main results of the simulations showed in section 4 and the conclusions in the final section.

## 2 Simulation Model

We have made use of the ARENA v.5.0 software to simulate the possible event set. ARENA is a powerful interactive visual modelling system that makes use of the SIMAN language. The initial model consists of an animation zone and a module logical zone that can be divided into one controller zone, one passenger zone and two elevator zone for each of the cabins.

The ***animation zone*** is defined by the `Arrive` and `Depart` modules, which regulate the arrivals and departures of the passengers at the system.

In the ***controller zone***, one entity is created by lift to travel around the logical zone. When the passengers come into the lift, the passengers are joined to the `controller` entity shaping one only entity at the same time as holding all the particular individual entities attributes.

The ***passenger zone*** consists of the allocation of the `UpDown` attribute (1 if the passenger goes up and 2 otherwise) that is stated as function of the `Origin` and `Destination` attributes. So, the passenger is sent to the waiting queue if it exists. Otherwise the hall call allocation procedure is done by means of the correspondent optimisation algorithm (our genetic algorithm by the case).

For each one of the ***elevator zones***, when the lift arrives at a floor the subsequent actions must be checked and done if necessary: lift waits for calls, passengers leaves the lift, passengers come into the lift, lift allocation in case of full capacity, cabin call allocation and call evaluation.

When the lift arrives at a floor, the `state` of the lift is evaluated. If the lift `state` is set to zero, the lift is stopped and will have access to the `Waiting_for_Calls` submodule. If the lift is not stopped and it is carrying passengers, it inputs into the `Leaving_the_Lift` submodule. If it is not carrying passengers, it inputs into the `Taking_Passengers` submodule after a `Delay` to simulate the opening doors time (we use the delay variable `Time_Doors` (2.5 seconds).

When the lift arrives at a floor, the `Arrival_Evaluation` submodule presents three options: the lift continues up, the lift continues down or the lifts starts the deceleration process (preparing to stop). We use the `LDX (Transporter ID, unit number)` as an ARENA proprietary variable allowing to know the floor in which the lift is. We load this data in the variable `Level`.

After updating `Level`, if the lift is going up and the lift is in the ground floor, the lift is sent to the first floor; if the lift is going down and the lift is in the highest floor, the lift is sent to the last but one floor. Otherwise the simulation model checks if the lift has stopped in the floor that `Level` indicates (this data can be checked by means