Journal Title
XX(X):1–12
DOI: 10.1177/ToBeAssigned
www.sagepub.com/
SAGE

Estimation of Elevator Passenger Traffic Based on the Most Likely Elevator Trip Origin-Destination Matrices

Arnaud Malapert[1] and Juha-Matti Kuusinen[2]

Abstract

We present a constraint programming formulation for the elevator trip origin-destination matrix estimation problem using a lexicographic bi-criteria optimization method where least squares minimization is applied to the measured counts and the minimum information or the maximum entropy approach to the whole matrix. An elevator trip consists of successive stops in one direction of travel with passengers inside the elevator. It can be defined as a directed network, where the nodes correspond to the stops on the trip and the arcs to the possible origins and destinations of the passengers. The goal is to estimate the most likely counts of passengers for the origin-destination pairs of every elevator trip occurring in a building that are consistent with the measured boarding and alighting counts and any prior information about the trip matrix. These counts are used to make passenger traffic forecasts which, in turn, are used in elevator dispatching to reduce uncertainties related to future passengers and therefore to improve passenger service level. Artificial test data was obtained by simulations of lunch hour traffic in a typical multi-story office building. This resulted in complex problem instances that enable robust performance and quality testing. The results show that the proposed approach outperforms previous alternatives in terms of quality, and can take an advantage of prior information. In addition, the proposed approach satisfies real time elevator group control requirements for estimating elevator trip origin-destination matrices.

Keywords

vertical transportation; building traffic; origin-destination matrix; constraint programming

Practical application

The elevator trip origin-destination matrix estimation problem is important since it makes it possible to obtain complete information and statistics about the elevator passenger traffic. The statistics can be used to model future passengers which, when taken into account in the elevator group control, helps to improve passenger service level. Simulation experiments show that most of the problems occurring in reality can be solved within a reasonable time considering a real application, and the solving algorithms are relatively easy to implement using available constraint programming tools. Hence, this work is undoubtedly of interest to the building and elevator industry.

Introduction

In high-rise buildings, elevators are typically combined into groups and the elevators in the same group are collectively controlled by an elevator group control system (EGCS). The task of the EGCS is to dispatch the elevators to passengers' requests, e.g., up and down calls, so that they get to their destinations fast and without waiting. In order to achieve this, modern EGCS use advanced mathematical optimization algorithms to minimize passenger waiting or journey time, or both. This optimization problem is called the Elevator Dispatching Problem (EDP)[1;2]. Given a set of pickup requests, a solution to the EDP defines optimal elevator routes, i.e., which requests and in which order each elevator will serve so that the selected service level criterion is optimized.

Making optimal dispatching decisions and maintaining a good passenger service level may be difficult especially during heavy traffic. This is because of the uncertainties related to the current and future traffic demand. For example, the number of passengers waiting behind a pickup request is not in general known, which means that the elevator dispatched to serve the request may not have enough space for all the passengers behind it. Modeling of the uncertainties in the EDP helps to avoid bad dispatching decisions and to improve the passenger service level[3]. The uncertainties are typically modeled based on historical traffic statistics that are constructed from traffic measurements. The EGCS is capable of measuring and storing, e.g., the count of passengers boarding and alighting at each elevator stop and passengers' requests[4–6].

The statistics based on the measurements do not fully describe the passenger traffic in a building, and thus, cannot be used to explicitly model all the uncertainties related to

[1]Université Côte d'Azur, CNRS, I3S, France
[2]KONE Corporation, 02150 Espoo, Finland

Corresponding author:
Arnaud Malapert, Laboratoire I3S - UMR7271 - UNS CNRS
2000, route des Lucioles - Les Algorithmes - bât. Euclide B,
06900 Sophia Antipolis - France
Tel : +33 4 92 94 27 93
Email: arnaud.malapert@unice.fr

the passenger traffic. For example, it is not possible to accurately model how many passengers who board at an origin floor will alight at a given destination floor, i.e., how much free space should there be in the elevator during the journey from the origin to the destination. Modeling of this and other important uncertainties requires the measuring of every passenger journey between every pair of floors in a building. This, however, is not possible with the commonly used measuring devices. Although some dedicated systems such as video cameras exist to identify every passenger, they are not commonly used in EGCS. In addition, the measuring devices are not usually 100% accurate, which makes also the boarding and alighting count measurements uncertain.

The passenger journeys can, however, be estimated by finding the passenger counts for the origin-destination (OD) pairs of every elevator trip occurring in a building. An elevator trip in an up or down direction starts at a stop where passengers board an empty elevator and ends at a stop where the elevator becomes empty again[7]. The OD pairs of the trip define the possible passenger routes that in turn are defined by delivery requests, e.g., calls given inside the elevator car.

An elevator trip can be mathematically defined as a directed network of nodes $N = \{1, 2, \ldots, n\}$, and arcs A defined by the OD pairs $(i, j) \in N$. The nodes correspond to the stops made by the elevator and the arcs to the OD pairs. Figure 1 presents an elevator trip with five nodes, $n = 5$, and four OD pairs that are defined as follows. The passengers who board the elevator at node 1 register a delivery request to node 4, which defines the OD pair $(1, 4)$. The passengers who board the elevator at node 2 register a new delivery request to node 3, which defines the OD pair $(2, 3)$. The passengers who board the elevator at node 2 may be traveling also to node 4, which results in the OD pair $(2, 4)$. The passengers who board the elevator at node 3 do not register new delivery requests, and thus, they must be traveling to node 4, which results in the OD pair $(3, 4)$. Let b_i and a_i denote the measured count of passengers who board and alight the elevator at node or stop i, respectively. They can be measured, e.g., with an electronic load-weighing device[5]. Let X_{ij} denote the unobserved number of passenger journeys or the OD passenger count from the origin node i to the destination node j. The goal is to find the OD passenger count for every OD pair $(i, j) \in A$ of the trip, i.e., the elevator trip OD matrix (ETODM) based on the measured counts.

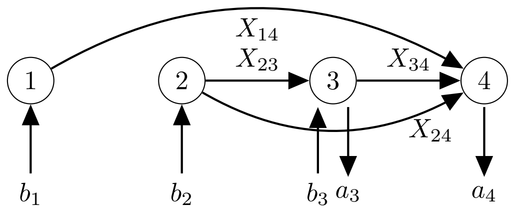


Figure 1. An elevator trip with five nodes and four OD pairs.

An elevator trip is analogous to a single transit route such as a bus line, where there is only one route connecting any OD pair, and counts on the boarding and alighting passengers are collected on all stops on the route[8]. In addition, a single transit route is usually defined in advance, a node forms an OD pair with every other preceding the node and the route has always the same set of OD pairs unless the route or the stops are changed. This means that it is possible to collect many counts on the same route during a given time period, e.g., a rush hour, and use these counts to estimate an OD matrix for the whole period.

The methods for the estimation of a single transit route OD matrix can be roughly divided into two categories. In the first one, it is assumed that the sum of the measured boarding counts is equal to the sum of the measured alighting counts, i.e., the counts are consistent. Then, a typical objective is to estimate an OD matrix that minimizes the distance to a target OD matrix with respect to a suitable distance measure[8–11]. The target OD matrix is usually based on historical data collected, e.g., with a manual survey. In the second category, the counts are not required to be consistent and the objective is to estimate an OD matrix that minimizes the distance between the estimated and measured counts, and usually also the distance to the target OD matrix[12–14]. The estimated boarding or alighting count at a node is the sum of the unobserved OD passenger counts out of or into the node, respectively.

An elevator trip, however, is request driven which means that every elevator trip is unique, having its own set of OD pairs and boarding and alighting counts. Consequently, the methods used to estimate an OD matrix for single transit route are not well suited for the estimation of an elevator trip OD matrix. In addition, especially in new buildings, the estimation must be based only on the measured counts since there is no target OD matrix available.

Kuusinen *et al.*[15] formulate the ETODM estimation problem as a box-constrained integer least squares (BILS) problem, and present algorithms for finding all solutions to the problem. There are three reasons for finding all or at least multiple solutions and the integer constraint. The first one is the natural fact that there cannot be partial passengers, and thus, the OD passenger counts must be positive integers. Actually, more stringent lower bounds can be defined by making the reasonable assumption that each delivery request corresponds to at least one passenger. The second one is related to the quality of the statistics constructed based on the estimated ETODMs. When all or at least multiple solutions are available for every ETODM estimation problem and the final solution is selected randomly for every problem, the statistics are not affected by the algorithm used for solving the problem. In the long term, this strategy results in statistics that model the possible realizations of the passenger traffic better than selecting always the first integer solution found by the algorithm or a continuous solution to every problem. The third reason is that integer OD counts make it possible to model the passenger traffic uncertainties in the EDP as a geometric Poisson process, which results in good performance with respect to passenger service level[16]. Although the ETODM estimation problem is NP-hard[17] (roughly, polynomial time algorithms are unlikely to exist for this problem), the problem instances occurring in reality are often so simple that all or at least many solutions can be found fast enough considering the time constraints of a real EGCS, usually in less than half-second[15].

Kuusinen and Malapert[18] present a constraint programming (CP) based formulation of the ETODM estimation problem. One advantage of the CP[19] approach compared to the BILS approach is that both deterministic and randomized search procedures resulting in a single or multiple optimal solutions can be easily implemented. The reason for studying the effect of randomization is that intuitively, if only some of the optimal solutions can be computed within a real time limit, then a randomized algorithm should result in better quality statistics for the reason that a deterministic algorithm will always favor particular solutions. Indeed, the results suggest that randomization and multiple solutions is a good compromise between solving time and quality. However, the problem with the ETODM estimation problem based only on the least squares objective function is that there are often more than one minimizing ETODM. The final solution can be selected, e.g., randomly or as the average of the minimizing ETODMs, but it would be better to obtain a good quality integer solution directly.

This paper presents a bi-criteria optimization method to estimate the most likely ETODM. The problem is formulated as a CP problem, but in addition to minimizing the least squares deviation between the estimated and measured boarding and alighting counts on the first level, a secondary objective function based on the minimum information or the maximum entropy approach is also optimized on the second level to obtain a single solution directly. The derivation of the minimum information and the maximum entropy objective function for the ETODM estimation problem is based on Van Zuylen and Willumsen[10]. The two objective functions favor solutions where none of the OD pairs of the OD matrix is given more weight, i.e., the passenger flow out of and into the origins and destinations, respectively, is divided evenly between the OD pairs. This is well suited for the ETODM estimation problem since the OD pairs do not have any order of importance. In addition, as mentioned in Van Zuylen and Willumsen[10], the two approaches make full use of the information contained in the measured counts, and prior information from a target OD matrix can easily be included into the problem. The performance of the new bi-criteria optimization methods is studied and compared to some previous methods with respect to solving time and quality of the estimated OD matrix.

The rest of the paper is organized as follows. Section presents the CP formulation, and Section the derivation of the minimum information and the maximum entropy objective. Section describes the process of generating and using the ETODMs in practice or in simulation, and Section the search algorithms. Section presents numerical experiments and results, and Section concludes the paper.

Constraint programming formulation

Continuing the example and mathematical definition from the previous section, let r_i be the node at which a delivery request to the node $i \in N = \{1, 2, \dots, n\}$, $r_i < i$, is registered. If no delivery requests are registered to node i, then $r_i = n + 1$. The elevator capacity, expressed as number of passengers, is denoted with C. The following assumptions are made:

1. At any time, there are less than C passengers in the elevator.
2. At least one passenger boards at node $r_i \neq n + 1$ and alights at node i.
3. Passengers do not alight at a node without a delivery request.
4. Passengers who board at node $i < r_j$, i.e., before the delivery request to node j is registered, do not alight at node j.

The assumptions 2, 3 and 4 imply that the delivery requests are assumed to be accurate, i.e., there are no false requests. This is reasonable since, even if passengers may sometimes accidentally register false requests, the number of such requests is small. The fourth assumption means that the possible destinations of a passenger are defined by the delivery requests that are registered before or at the node where the passenger boards the elevator, which is usually the case in practice. This eliminates some OD pairs, and thus, an elevator trip often includes a smaller number of OD pairs than a single transit route where typically any node i forms an OD pair with any other node j, $i < j$. The set of arcs A is then defined as:

$$A = \{(i, j) \in N^2 \mid i < j \ \wedge \ i \geq r_j\} \ . \tag{1}$$

Let $B_i \in [0, C]$ and $A_i \in [0, C]$ denote the estimated count of passengers who board and alight the elevator at node $i \in N$, respectively. The estimated boarding and alighting counts must be consistent:

$$\sum_{i \in N} B_i = \sum_{j \in N} A_j \ . \tag{2}$$

An elevator trip starts at a stop where passengers board an empty elevator and ends to a stop where the elevator becomes empty again. Hence, at the first node, the estimated boarding count must be at least one and the alighting count zero, and, at the last node, the reverse must hold:

$$A_1 = 0, \ \ B_1 \geq 1, \ \ A_n \geq 1, \ \ B_n = 0 \ . \tag{3}$$

At every node between the first and the last node, at least one passenger either boards or alights:

$$A_i + B_i \geq 1, \quad 1 < i < n \ .$$

This constraint is more accurately stated by considering the delivery requests.
Passengers cannot alight at a node to which there is no delivery request, and thus, at least one passenger must board:

$$r_i = n + 1 \Leftrightarrow A_i = 0 \wedge B_i \geq 1, \quad 1 < i < n \ . \tag{4}$$

At least one passenger boards at node $r_i \neq n + 1$ and alights at node i:

$$r_i \neq n + 1 \Leftrightarrow X_{r_i, i} \geq 1, \quad 1 < i \leq n \ . \tag{5}$$

The unobserved OD passenger counts are related to the estimated boarding and alighting counts through the flow conservation constraints:

$$\sum_{j \mid (i,j) \in A} X_{ij} = B_i, \quad \forall\, i \in N \ , \tag{6}$$

$$\sum_{i \mid (i,j) \in A} X_{ij} = A_j, \quad \forall\, j \in N \ . \tag{7}$$

Let $P_i \in [1, C]$, $i = 1, \ldots, n-1$, denote the number of passengers in the elevator between the nodes i and $i+1$. It is computed as follows:

$$P_1 = B_1, \quad P_{n-1} = A_n, \\ P_i = P_{i-1} + B_i - A_i, \quad 1 < i < n-1 \ . \tag{8}$$

The elevator capacity constraint is always satisfied because of the domain of the variables.

In a real application, the goal is to measure the boarding and alighting counts as accurately as possible. Hence, it can be assumed that the measured counts are closed to the true counts, i.e., possible measuring errors are small. A good solution or an ETODM would then be such that the difference between each estimated and measured count on the elevator trip is small. Such solutions can be obtained by minimizing the least squares (LS) deviation between the estimated and measured counts:

$$LS = \sum_{i \in N} \left[(A_i - a_i)^2 + (B_i - b_i)^2 \right] \ . \tag{9}$$

The goal is to minimize (9) with respect to the constraints (2)-(8).

Finding the most likely ETODM

This section presents the minimum information and the maximum entropy approaches to formulate a secondary objective function and to obtain a single solution directly. The main difference with the approach of Van Zuylen and Willumsen [10] is that the second level optimization problem is solved using CP rather than a dedicated algorithm. Indeed, they obtained the formal solution to the second level problem by the differentiation of its Lagrangian, which results in a multi-proportional problem. Here, the presence of additional constraints prevents the direct application of the Lagrangian method.

Minimum information approach

Since the information contained in the measured boarding and alighting counts is not enough to define a unique ETODM, it seems reasonable to select an ETODM that adds as little information as possible to the knowledge contained in equations (6) and (7). Brillouin [20] defines the information contained in a set of N observations where the state k has been observed n_k times as:

$$I = \log N! \prod_k \frac{q_k^{n_k}}{n_k!}, \tag{10}$$

where q_k is the a priori probability of observing state k. If the observations are boarding counts B_i at floor i, it is possible to define state ij as the state in which a passenger has been traveling on the OD pair (i, j). This implies that:

$$n_{ij}^b = X_{ij}. \tag{11}$$

It is also possible to express the a priori probability q_{ij}^b of observing state ij when boarding at floor i as a function of a priori information about the OD matrix as:

$$q_{ij}^b = \frac{t_{ij}}{\sum_{\Gamma(i)} t_{ik}}, \tag{12}$$

where t_{ij} is the a priori number of passenger journeys or the OD passenger count from origin i to destination j provided, e.g., by a historical OD matrix, and $\Gamma(i) = \{j | (i, j) \in A\}$ is the set of successors of i.

The information contained in the boarding count B_i for the OD pair (i, j) is then:

$$I_{b_i} = -\log B_i! \prod_{\Gamma(i)} \frac{\left(q_{ij}^b\right)^{X_{ij}}}{X_{ij}!}. \tag{13}$$

Using Stirling's approximation, $\log X! = X \log X - X$, it is possible to obtain:

$$I_{b_i} = -B_i \log B_i + B_i - \sum_{\Gamma(i)} X_{ij} \log q_{ij}^b + \\ \sum_{\Gamma(i)} X_{ij} \log X_{ij} - \sum_{\Gamma(i)} X_{ij}, \tag{14}$$

and using the fact that $B_i = \sum_{\Gamma(i)} X_{ij}$ gives:

$$I_{b_i} = -\sum_{\Gamma(i)} X_{ij} \log B_i - \sum_{\Gamma(i)} X_{ij} \log q_{ij}^b + \sum_{\Gamma(i)} X_{ij} \log X_{ij} \\ = \sum_{\Gamma(i)} X_{ij} \log \frac{X_{ij}}{B_i q_{ij}^b}. \tag{15}$$

Similarly, the information contained in the alighting count A_j can be computed using the a priori probability q_{ij}^a of observing state ij when alighting at floor j:

$$q_{ij}^a = \frac{t_{ij}}{\sum_{j|(i,j) \in A} t_{kj}}. \tag{16}$$

Then, summing up over all the OD pairs on the elevator trip gives:

$$I = \sum_A X_{ij} (\log \frac{X_{ij}}{A_j q_{ij}^a} + \log \frac{X_{ij}}{B_i q_{ij}^b}). \tag{17}$$

This is the total information contained in the measured counts and in the unobserved OD passenger counts. The problem of finding an ETODM consistent with the measured counts and adding a minimum of extra information to the them is equivalent to minimizing I subject to the model's constraints.

Maximum entropy approach

This approach defines the most likely ETODM as the one having the greatest number of micro-states associated with it [21]. The number of ways of selecting an ETODM with a total number of passengers or passenger journeys X is then:

$$W = \frac{X!}{\prod_{ij} X_{ij}!}, \tag{18}$$

where $\prod_{ij}$ stands for $\prod_{(i,j) \in A}$. The goal is to find an ETODM that maximizes W or a monotonic function of it. A convenient choice is the logarithmic function:

$$E = \log W. \tag{19}$$

Using Stirling's approximation for $\log X!$ the problem becomes

$$E = X \log X - X - \sum_{ij}(X_{ij} \log X_{ij} - X_{ij})$$
$$= X \log X - \sum_{ij}(X_{ij} \log X_{ij}). \tag{20}$$

Prior information can be included in this model as follows:

$$W' = \frac{X! \prod_{ij} q_{ij}^{X_{ij}}}{\prod_{ij} X_{ij}!}, \tag{21}$$

where

$$q_{ij} = \frac{t_{ij}}{\sum_{ij} t_{ij}}. \tag{22}$$

Then, following steps analogous to above:

$$E = X \log X - \sum_{ij}(X_{ij} \log \frac{X_{ij}}{q_{ij}}). \tag{23}$$

Generating and using the ETODMs

The ETODMs estimated during a given time interval, e.g., 5 or 15 minutes, can be summed up to construct a building OD matrix (BODM). This matrix describes the passenger traffic between every pair of floors in the building. The BODMs of successive intervals form historical traffic statistics that can be used to model the passenger traffic uncertainties. They can also be used by the EGCS to forecast and adapt to the possible changes in the traffic. For example, to learn the traffic in the building, the BODMs of the same time of day or time interval, and usually day of week can be combined, e.g., by exponential smoothing[5].

In a real building, the elevator and passenger traffic events that are used to formulate and generate ETODM estimation problems such as elevator stops, boarding and alighting passenger counts, and passengers' requests are created by actual passenger and elevator movements. In this study, however, the events were obtained from simulations run with the Building Traffic Simulator (BTS)[22]. The advantage of simulation is that traffic, elevator and building parameters can easily be changed to model any kind of building. In addition, the simulation of a whole day traffic takes typically only a couple of minutes, and thus, a large amount of appropriate test data can easily be obtained.

Figure 2 describes the process of generating ETODM estimation problems, solving the problems, constructing the BODM based on the estimated ETODMs and using the BODM in the EDPs both in practice and in simulation. The EGCS is responsible for monitoring the elevator and passenger traffic events, and controlling the elevators based on the EDP solutions. Each elevator typically has its own sensors and signalization devices, e.g., a load-weighing device to count the boarding and alighting passengers and buttons for giving a delivery request to a destination floor, and a control unit that sends this data to the EGCS. The EGCS also receives the pickup requests registered using, e.g., up and down call buttons in the elevator lobbies. Based on these data, the EGCS formulates and solves the ETODM estimation problems and the EDPs. The EDP solutions define the dispatching decisions and elevator routes, i.e., which pickup and delivery requests each elevator should serve and in which order. The EGCS communicates the route of an elevator to its control unit that then executes the route.

In reality, the EDP solution algorithm as well as all the other components of the process are part of the EGCS but, in the figure, they are separated for clarity. In the figure, $\mathcal{P}_p$ corresponds to the pth ETODM estimation problem, $1 \leq p \leq m$, where m is the total number of estimation problems generated, e.g., within a 15-min interval, n_p is the number of nodes in the pth problem, function f maps an OD passenger count obtained by solving the pth problem, X^p_{uv}, to the correct OD pair of the estimated BODM, $\widehat{B}$, and n^B is the total number of floors in the building. B describes the true BODM that contains the true traffic. It is constructed directly from the simulated passengers and can be used in two ways:1) to study the quality of the estimation results or the estimated BODM, and 2) as prior information in the ETODM estimation problems. In reality, the true BODM is not known, but it could be replaced by an estimated BODM when the estimation process has been running for some time. To highlight the fact that simulation is not part of a real elevator system or building, the corresponding components and connections are shown in the figure with gray color and dashed lines.

Search algorithms

The search algorithms summarized in Table 1 are based on a complete backtracking search which consists of a depth-first traversal of the search tree. During the search, an uninstantiated variable X_{ij} representing an OD passenger count is selected at each node of the tree, and the node is extended with new branches that may have to be examined in order to find a solution. The branching strategy determines the order in which the variables and their values are instantiated.

This paper considers only one deterministic variable selection strategy, namely, the *dom* strategy which selects the variable whose domain is the smallest. Other application-independent variable selection strategies exist, but Kuusinen and Malapert[18] showed that they do not provide here significant improvement compared to the simpler *dom* strategy with respect to solving time and quality.

The value selection is based on two classical strategies; *minVal* selects the smallest value of the domain and *randVal* selects a value randomly. They showed that the also classical *maxVal* strategy, which selects the largest value, increases the search time without any improvement in quality, and thus, it is not considered in this paper. For an extensive review on constraint programming, we refer the reader to the Chapter 4 of the "Constraint Programming Handbook"[19] Chapter 4.

Single-criterion optimization

These search algorithms minimize the LS objective function (9) using a *bottom-up* procedure. The *bottom-up* procedure starts with a lower bound lb as the target upper bound which is incremented by one unit until the problem becomes feasible. The first solution found by *bottom-up* is proven optimal. The *bottom-up* procedure is efficient if the lower bound is tight.

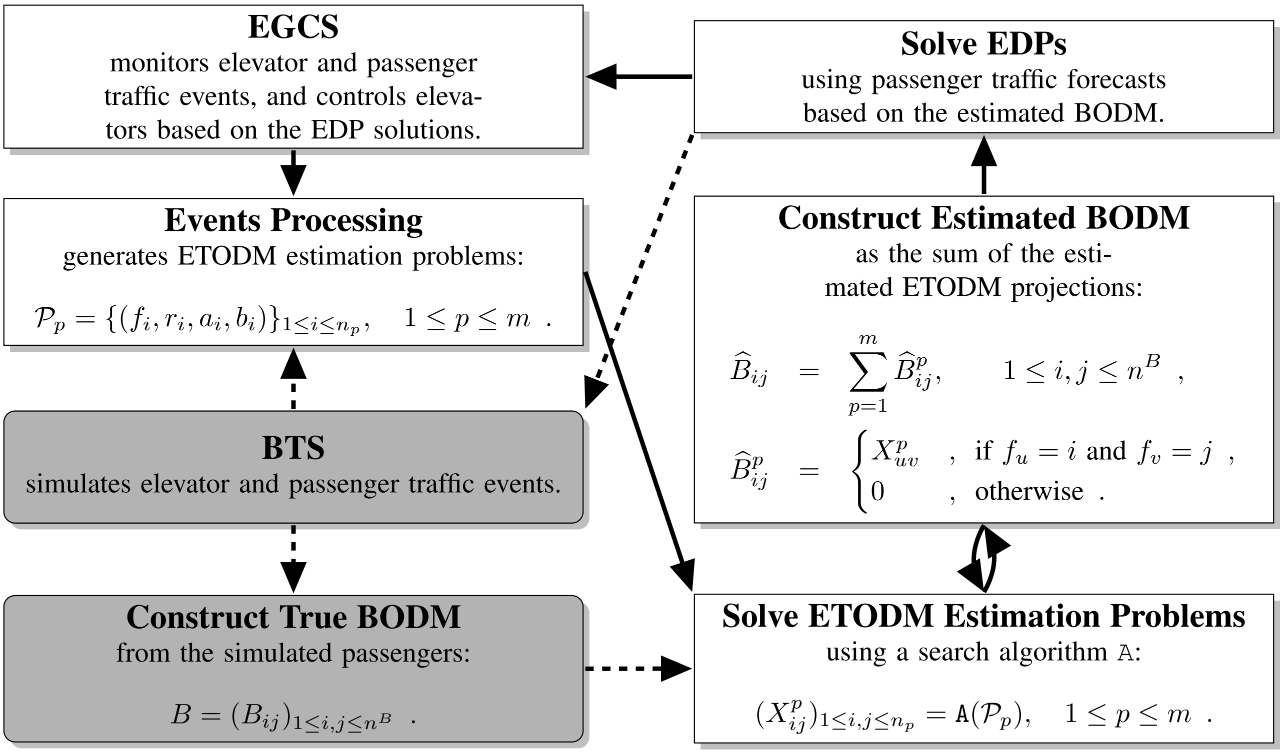


Figure 2. Overview of the process of generating and using the ETODMs in practice and in simulation.

The first two single-criterion search algorithms return the first optimal solution using different value selection strategies; DOM returns the first optimal solution using the strategies *dom* and *minVal*, and RAND returns the first optimal solution using the strategies *dom* and *randVal*.

The *bottom-up* procedure can be easily extended to find all optimal solutions. The next two single-criterion search algorithms use the *dom* and *minVal* strategies to find all solutions; UNIF returns an optimal solution drawn randomly with a uniform distribution among all optimal solutions, and AVG returns the average of all optimal solutions. Indeed, the *randVal* strategy is efficient for the diversification of equivalent solutions, but not needed when searching for all solutions. Note that AVG is the only search algorithm that does not satisfy the integer constraint and produces a fractional ETODM.

Bi-criteria optimization

These search algorithms use a lexicographic method which consists of solving two single-objective optimization problems. They first minimize the LS objective function and then a secondary criterion subject to the constraint that the LS objective remains optimal. The secondary criterion is either the minimum information objective (17) or the maximum entropy objective (23). By nature, the lexicographic method returns pareto-optimal solutions. The lexicographic method was chosen since the measuring errors on the boarding and alighting counts can be assumed to be very small, and thus, the LS objective considered more important than the secondary objectives.

The secondary objective is optimized using the *top-down* procedure which starts with an upper bound, ub, and tries to improve it. If, by luck, the first solution found with the *top-down* procedure is optimal, the optimality still has to be proven. The *top-down* procedure is appropriate for the optimization of the secondary objective because typically the lower bounds on this objective are weak.

The search algorithms use the *dom* and *minVal* strategies; ENT uses the maximum entropy and INF the minimum information as the secondary objective. In addition, some of the algorithms use prior information to put more weight on historically likely OD passenger counts. ENTP and INFP use the prior information contained in the true BODM, and ENTB and INFB use the prior information obtained by summing up all the true BODMs generated for the numerical experiments. Hence, the latter two algorithms are based on imperfect prior information corresponding to the reality better.

Numerical experiments

This section presents computational experiments conducted to evaluate the search algorithms. The emphasis is especially on the performance of the new bi-criteria optimization methods compared to the previous single-criterion alternatives. The implementation is based on `choco` (`http://choco.mines-nantes.fr`) which is an open source java library for constraint programming. All the experiments were conducted on a Linux machine with 32 GB of RAM and a Intel Core i7 processor (6 cores – 3.20GHz). In reality, the EGCS may be executed on an industrial PC with less computing power, which may limit the usage of the search algorithms in practice. However, most of the instances occurring in reality are so simple that they can be solved fast enough also with a less powerful machine[15], and in any case the process executing the ETODM solver can be given a lower priority

Algorithm	Objective(s)	Value selection	Solution selection	Prior info
DOM	LS	*minVal*	First optimal solution	–
RAND1 RAND2	LS LS	*randVal*	First optimal solution	–
UNIF1 UNIF2	LS LS	*minVal*	Random optimal solution	–
AVG	LS	*minVal*	Average optimal solution.	–
ENT INF	Lex(LS,E) Lex(LS,I)	*minVal*	Pareto-optimal solution	–
ENTP INFP	Lex(LS,E) Lex(LS,I)	*minVal*	Pareto-optimal solution	Perfect
ENTB INFB	Lex(LS,E) Lex(LS,I)	*minVal*	Pareto-optimal solution	Imperfect

Table 1. Summary of the search algorithms. All algorithms use *dom* as the variable selection strategy. The search algorithms RAND1 and RAND2 (resp. UNIF1 and UNIF2) correspond to RAND (resp. UNIF) with different random seeds. All algorithms return an integer solution with the exception of AVG which returns a fractional solution.

or a maximum allowed duration based on the performance of the machine so that more critical tasks can be executed immediately when necessary.

This section is organized as follows. Section analyses the characteristics of the true BODMs and of the generated ETODM estimation problem instances. Section analyses the optimal values and the number of optimal solutions over all instances for the LS objective. Section compares the performance of the search algorithms with respect to the real time constraint of the EGCS for solving one ETODM estimation problem. The remaining four subsections study the performance of the algorithms with respect to four quality measures. All four measures compare the estimated BODM, $\widehat{B}$, to the true BODM, B. Section analyzes the absolute deviation $\sum_{i=1}^{n^B}\sum_{j=1}^{n^B}|\widehat{B}_{ij} - B_{ij}|$, where n_B is the number of floors in the building, and Section the LS deviation $\sum_{i=1}^{n^B}\sum_{j=1}^{n^B}(\widehat{B}_{ij} - B_{ij})^2$. Section discusses the accuracy and precision of estimated BODMs, and Section summarizes the numerical results.

Characteristics of the true BODMs

Building Traffic Simulator was used to simulate lunch hour traffic in a 25-story office building with two express floors. Lunch hour is one of the most difficult traffic situations occurring in an office building during a day. The simulated elevator group consisted of eight elevators with the capacity of 21 passengers. To obtain several sets of test data, i.e., true BODMs and the corresponding ETODM estimation problem instances, we repeated the simulation process 10 times with different seeds. The duration of each simulation was 15 minutes which is a suitable interval length for passenger traffic statistics in a real elevator group control application[5].

Naturally, simulation does not involve measurement errors but in practice it is very difficult or even impossible to measure the boarding and alighting counts without errors. The magnitude of the measuring error and the error frequency depends on the accuracy of the measuring device and counting algorithm. For example, with a basic load weighing device and a dedicated algorithm based on such a device, a reasonable assumption about the measuring accuracy is 90% and the magnitude of the error one passenger[23]. Consequently, to simulate measurement errors, one passenger was removed from each boarding and alighting count with 10% probability.

Table 2 gives general information about each of the 10 true BODMs: the number of elevator trips, i.e., ETODM estimation problem instances, the total number of passengers, and the number of OD pairs for which the OD passenger count is non-zero. Since the total number of floors in the simulated building was 25 and there were two express floors, the total number of OD pairs for which the OD passenger count can be non-zero is $25 \times 25 - 25 - 94 = 506$ (all OD pairs - diagonal OD pairs - express floor OD pairs). There is a total of 558 elevator trips with a total of 6675 passengers. The average number of passengers per elevator trip is around 12, and the average number of non-zero OD pairs in a true BODM is around 23.7% when all OD pairs are considered and 29.2% when only those OD pairs for which the OD passenger count can be non-zero are taken into account. In addition to the information given in the table, the number of stops or nodes in an ETODM estimation problem instance varies from 2 to 19 and the number of passengers from 1 to 34.

Each true BODM contains the passengers from a 15-min period which constitutes an upper bound for solving an average of 55 ETODM estimation problem instances. This is approximately 15 seconds per instance. But at the same time, the EGCS is processing more important tasks such as dispatching the elevators to passenger's requests. This means that each ETODM estimation problem instance needs to be solved as fast as possible. The next sections show, however, that the quality of the solutions is more challenging than the time limit.

Characteristics of the optimal solutions

Figure 3a gives the frequency of the optimal value (dark gray) and the number of optimal solutions (light grey) for the LS objective. The latter is calculated for a value range based on the x-axis values $i = 0, 1, 2, \dots, 8$ as $]10^{i-1}, 10^i]$. For instance, the two leftmost columns indicate that, for more than 300 instances, the LS objective is equal to 0 and they have only one optimal solution, $10^0 = 1$.

BODM	1	2	3	4	5	6	7	8	9	10	total
trips	53	50	59	52	54	54	63	60	59	54	558
passengers	692	681	676	597	686	682	721	662	672	706	6675
OD pairs	143	144	148	138	142	158	158	141	162	144	1478

Table 2. General information about the true BODMs.

The LS objective value is very small and close to the trivial lower bound (0) which justifies the choice of the *bottom-up* procedure. Note that errors on the boarding or alighting counts can be present even if the LS objective is equal to 0, for instance, if one passenger is removed from the boarding and the alighting count related to the same OD pair. More than 40% of the instances have multiple optimal solutions, and 5% have more than 1000 optimal solutions. Errors on boarding and alighting counts and multiple optimal solutions are likely to decrease the quality of the estimation results and BODMs.

Solving time

Figure 3b shows a box plot of the solving times of the 558 ETODM problem instances with respect to the search algorithms. The box spans the range of values from the first quartile to the third quartile. The whiskers extend from each end of the box for a range equal to 1.5 times the interquartile range. Any point that lies outside of the range of the whiskers is considered as an outlier which are drawn as individual stars in the figure.

Most of the instances are solved within a few seconds. The search algorithms that return the first optimal solution (DOM, RAND1 and RAND2) are the fastest. The maximum time in the figure is 10 seconds even if there was one instance for which it took more than 10 seconds to find the first optimal solution. Including this instance, there were only four instances for which the other algorithms ends within more than 10 seconds. To conclude, all search algorithms meet the real time constraints of the EGCS. In the worst case, the search can always be interrupted as soon as a solution has been found.

Absolute deviation

Figure 4a shows a heat map of the absolute deviations between the true and the estimated BODMs. Here, the value of a cell is the absolute deviation of a search algorithm (row) for a BODM (column). The color of a cell represents the Z-score of the cell which is the signed number of the standard deviations of the absolute deviation of a search algorithm for a BODM. Z-scores larger than the mean are represented by dark gray whereas Z-scores smaller than the mean are represented by light gray. Briefly, the smaller (lighter) the Z-score (color), the better the search algorithm (row) for a BODM (column) with respect to the absolute deviation. The dendrogram on the left is a tree diagram that illustrates the hierarchical clustering of the search algorithms. The rows or algorithms are ordered from top to bottom in a decreasing order according to the row means, and the leaves of the dendrogram that also correspond to algorithms are given a color according to the grouping in Table 1. Note that the relevance of this empirical grouping is confirmed by the results of the hierarchical clustering.

The search algorithms based on bi-criteria optimization outperform those based on single-criterion optimization. The dendrogram also highlights the fact that single-criterion and bi-criteria optimizations behave differently. The only exception is that AVG outperforms ENT. Bi-criteria optimization with perfect information is the most efficient and the dendrogram shows that it behaves differently than with imperfect information or without prior information. Imperfect information is beneficial for the maximum entropy approach, but not for the minimum information approach. For most BODMs, bi-criteria optimization with perfect information provides the best quality with respect to the absolute deviation.

Least squares deviation

Figure 4b shows a heat map of the LS deviations between the true and the estimated BODMs. Here, the value of a cell is proportional to the root mean square error which is equal to the LS deviation divided by the square of the total number of floors which is 625. AVG outperforms other search algorithms, but it is the only search algorithm that does not satisfy the integer constraint and returns fractional ETODMs that give a real advantage here. Except for AVG, bi-criteria optimization outperforms single-criterion optimization UNIF being the best of the single-criterion based algorithms. As showed by the dendrogram, ENTP and INFP are close to AVG, but they satisfy the integer constraint. More surprisingly, the bi-criteria search algorithms using imperfect information, ENTB and INFB, are not as good as the same algorithms without prior information, INF and ENT. In addition, the maximum entropy approach outperforms the minimum information approach without prior or with imperfect information.

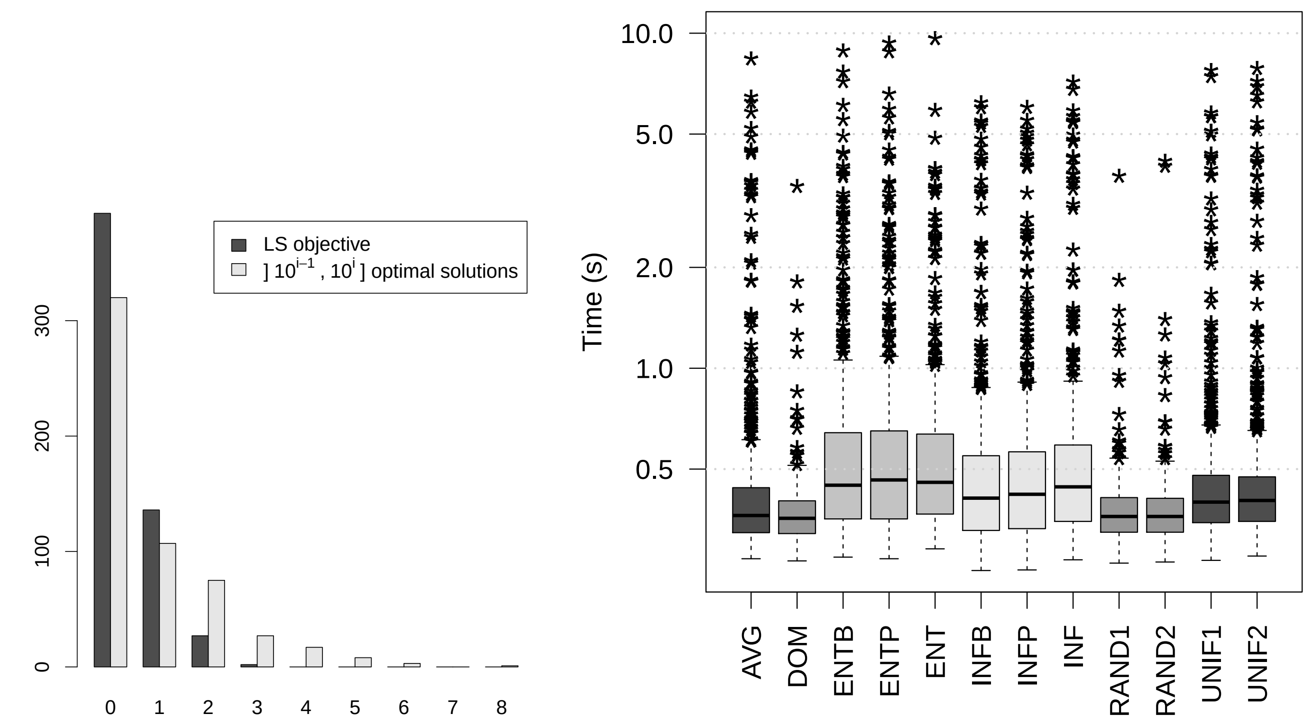
Accuracy and precision

In this section, we focus on the OD pairs rather than on the OD counts. Table 3 describes a classification of the OD pairs based on the true and estimated BODMs. The rows describe the presence of an OD pair in a true BODM while the columns describe the presence of the OD pair in the corresponding estimated BODM. For example, the cell $(B_{ij} = 0, \widehat{B}_{ij} = 0)$ means that OD pair (i, j) is not present, i.e., the corresponding OD passenger count is zero, in the true and in the estimated BODM which results in a true negative classification.

	$\widehat{B}_{ij} = 0$	$\widehat{B}_{ij} > 0$
$B_{ij} = 0$	True Negative (TN)	False Positive (FP)
$B_{ij} > 0$	False Negative (FN)	True Positive (TP)

Table 3. Classification of the OD pairs.

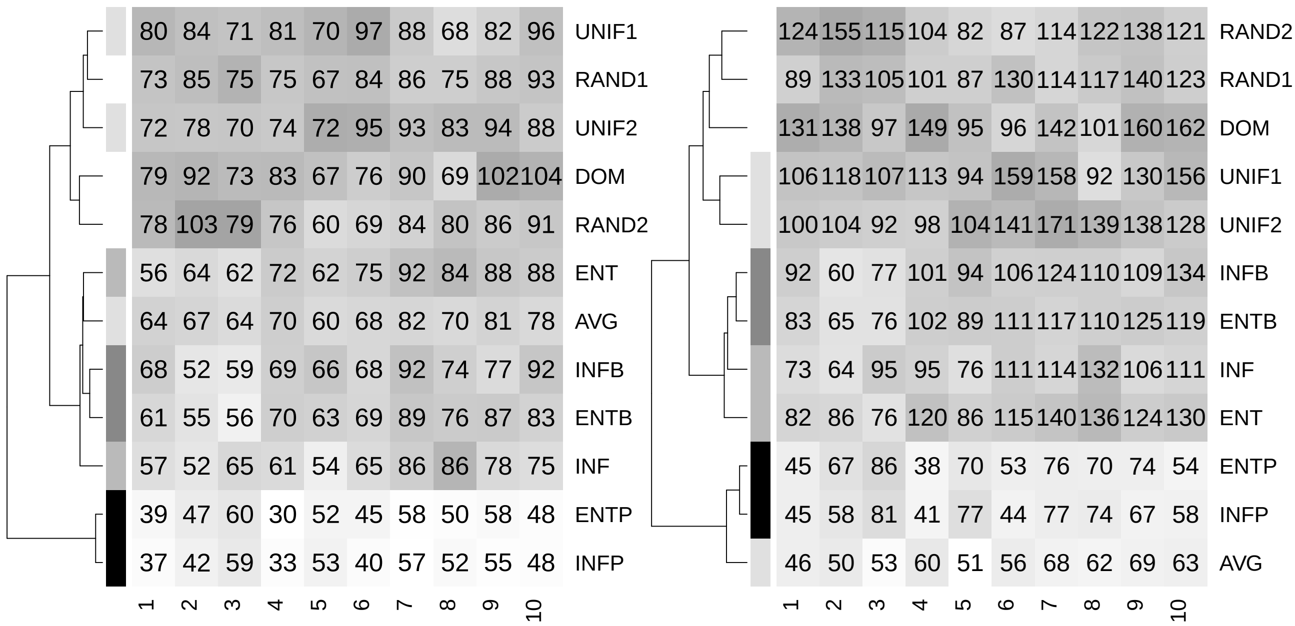
Accuracy and precision are defined based on this classification. Accuracy is the proportion of true results, both



(a) Frequency of the LS objective value and the number of optimal solutions.

(b) Solving time.

Figure 3. Frequency of the LS objective values and the number of optimal solutions, and solving time analysis for the ETODM estimation problem.



(a) Absolute deviation.

(b) LS deviation.

Figure 4. Absolute and LS deviation between the true and estimated BODMs.

true positives and true negatives, among the total number of cases examined:

$$\text{Accuracy} = \frac{\#\text{TN} + \#\text{TP}}{\#\text{TN} + \#\text{TP} + \#\text{FP} + \#\text{FN}} \quad . \tag{24}$$

In this case, an accuracy of 100% means that whenever an OD pair is present (resp. not present) in the true BODM, it is also present (resp. not present) in the estimated BODM.

Precision is defined as the proportion of the true positives against all the positive results, both true positives and false positives:

$$\text{Precision} = \frac{\#\text{TP}}{\#\text{TP} + \#\text{FP}} \ . \tag{25}$$

Precision answers the following question: “If an estimated OD pair is positive, how well does that predict the presence of the OD pair in the true BODM?”.

Figure 5 shows the accuracy and precision of the search algorithms as a scatter plot. Each point represents one BODM estimated by one search algorithm. The search algorithms are grouped to ease the interpretation. Clearly, AVG has the lowest accuracy and precision because of its fractional nature. If a passenger can be assigned to several OD pairs in an ETODM estimation problem, AVG splits the passenger between these OD pairs and therefore decreases accuracy and precision. So despite having low deviations, AVG has a poor predictive power in describing the actual passenger traffic in a building. The search algorithms with perfect information, INFP and ENTP, have the highest accuracy and precision. The other algorithms are slightly less accurate and precise and have about the same predictive power for OD pairs even if their estimation performance with respect to OD counts differs as discussed before.

Summary of the numerical results

Based on the results, bi-criteria search algorithms with perfect information significantly outperform the other search algorithms. In most cases, bi-criteria optimization outperforms single-criterion optimization. Fractional optimization such as AVG has low deviations but also poor accuracy and precision. Imperfect information, on the other hand, does not really improve the BODM quality, but this may not be the case in practice. Here, imperfect information was formed by summing up all the 10 true BODMs into one BODM that does not describe the OD pairs and passenger counts of a particular BODM accurately enough. In practice, imperfect information would correspond, e.g., to a historical BODM of a given time interval. A time series of such matrices is likely to capture typical traffic patterns, e.g., passengers' day-to-day routines such as lunch time habits. Hence, in practice, it may be possible to obtain imperfect information that improves the quality of the estimated BODMs. Overall, the maximum entropy approach slightly outperforms the minimum information approach. Finally, the performance of the search algorithms meets the real time constraints of the EGCS.

Conclusions

We presented a constraint programming formulation for the elevator trip origin-destination matrix (ETODM) estimation problem using a lexicographic bi-criteria optimization method based on least squares minimization of the measured traffic counts and information minimization or entropy maximization of the whole matrix. An elevator trip consists of successive stops in one direction of travel with passengers inside the elevator, and the estimated OD matrix contains the OD passenger counts for the OD pairs of the trip.

The ETODMs estimated for a given time interval are added up to construct the building OD matrix (BODM) of that interval. The passenger traffic in a building can be learned by combining the BODMs of the same day or time interval, and usually day of week. These matrices can be used to make forecasts about future passengers. The forecasts are needed in elevator dispatching to improve the dispatching decisions with respect to future passengers.

An ETODM estimation problem may have many optimal solutions with respect to the least squares minimization, and any of these solutions may correspond to what happened in reality. To obtain robust forecasts, the learned BODMs should describe the possible realizations of the passenger traffic as well as possible. This can be achieved by finding the most likely ETODM based on the information minimization or the entropy maximization approach.

Several test problems were obtained by simulations of lunch hour traffic in a typical multi-story office building. The traffic intensity was adjusted above the handling capacity of the simulated elevator group. This resulted in complex problem instances that enable robust performance testing and comparison of the algorithms. We compared the bi-criteria optimization methods to single-criterion alternatives. The comparison was based on solving time and BODM quality which affects the reliability of the passenger traffic forecasts.

The results show that the proposed bi-criteria optimization methods clearly outperform their single-criterion alternatives and provide a good compromise between solving time and quality. The bi-criteria methods can also make full use of the information contained in the observed boarding and alighting counts as well as a priori information available in the form of, e.g., an old BODM. However, the results show that while perfect a priori information significantly improves the quality of estimation results, the imperfect a priori information used in this study did not provide much improvement. Hence, future research will address how to form better imperfect a priori information. Furthermore, the BODMs estimated using the entropy maximization approach were frequently closer to the true BODMs than those estimated using the information minimization approach. Finally, the proposed bi-criteria methods fulfill real time elevator group control requirements for solving ETODM estimation problems.

References

1. Tyni T and Ylinen J. Genetic algorithms in elevator car routing problem. In Spector L, Goodman ED, Wu A et al. (eds.) *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)*. San Francisco: Morgan Kaufmann, pp. 1413–1422.
2. Ruokokoski M, Ehtamo H and Pardalos PM. Elevator dispatching problem: a mixed integer linear programming formulation and polyhedral results. *Journal of Combinatorial Optimization* 2015; 29(4): 750–780. DOI:10.1007/s10878-013-9620-1.
3. Luh PB, Xiong B and Chang SC. Group elevator scheduling with advance information for normal and emergency modes. *IEEE Transactions on Automation Science and Engineering* 2008; 5(2): 245–258. DOI:10.1109/TASE.2007.895217.
4. Siikonen ML and Kaakinen M. Using artificial intelligence to improve passenger service quality. In Barney GC (ed.) *Elevator Technology 5, Proceedings of Elevcon '93*. Vienna: International Association of Elevator Engineers, pp. 239–246.
5. Siikonen ML. Elevator group control with artificial intelligence. Research report A67, Systems Analysis

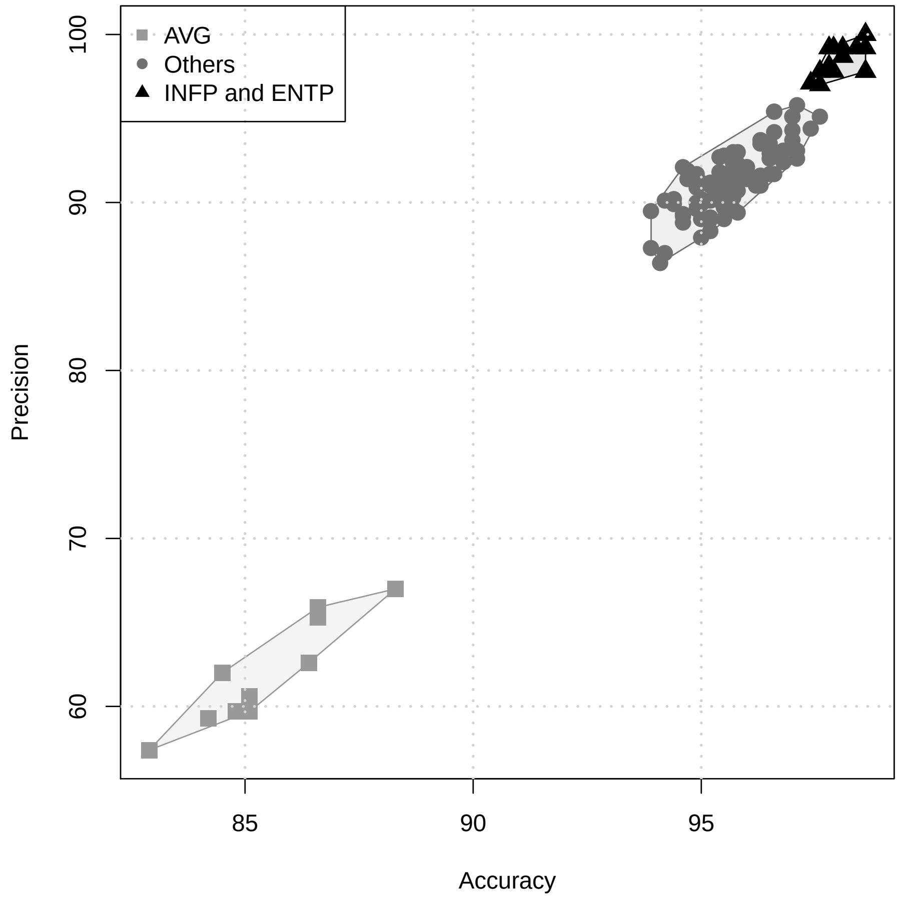


Figure 5. Accuracy and precision of OD pairs.

Laboratory, Helsinki University of Technology, 1997.

6. Utgoff PE and Connell ME. Real-time combinatorial optimization for elevator group dispatching. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans* 2012; 42(1): 130–146. DOI:10.1109/TSMCA.2011.2157134.
7. Yoneda K. Elevator trip distribution for inconsistent passenger input-output data. *Decision Making in Manufacturing and Services* 2007; 1(1-2): 175–190.
8. Nguyen S. Estimating origin-destination matrices from observed flows. In Florian M (ed.) *Transportation Planning Models*. North-Holland, Amsterdam, 1984. pp. 363–380.
9. Tsygalnitsky S. *Simplified Methods for Transportation Planning*. Master's Thesis, Department of Civil Engineering, Massachusetts Institute of Technology, 1977.
10. Van Zuylen HJ and Willumsen LG. The most likely trip matrix estimated from traffic counts. *Transportation Research Part B* 1980; 14(3): 281–293. DOI:10.1016/0191-2615(80)90008-9.
11. Ben-Akiva ME, Macke PP and Hsu PS. Alternative methods to estimate route-level trip tables and expand on-board surveys. *Transportation Research Record* 1985; 1037: 1–11.
12. Maher MJ. Inferences on trip matrices from observations on link volumes: A bayesian statistical approach. *Transportation Research Part B* 1983; 17(6): 435–447. DOI:10.1016/0191-2615(83)90030-9.
13. Cascetta E and Nguyen S. A unified framework for estimating or updating origin/destination matrices from traffic counts. *Transportation Research Part B* 1988; 22(6): 437–455. DOI:10.1016/0191-2615(88)90024-0.
14. Bell MGH. The estimation of origin-destination matrices by constrained generalized least squares. *Transportation Research Part B* 1991; 25(1): 13–22. DOI:10.1016/0191-2615(91)90010-G.
15. Kuusinen JM, Sorsa J and Siikonen ML. The elevator trip origin-destination matrix estimation problem. *Transportation Science* 2015; 49(3): 559–576. DOI:10.1287/trsc.2013.0509.
16. Sorsa J, Ehtamo H, Kuusinen JM et al. Modeling uncertain passenger arrivals in the elevator dispatching problem with destination control. *Optimization Letters* 2017; : 1–15DOI:10.1007/s11590-017-1130-0.
17. Garey MR and Johnson DS. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York, USA: W. H. Freeman and Company, 1979.
18. Kuusinen JM and Malapert A. The effect of randomization on constraint based estimation of elevator trip origin-destination matrices. In Adams J, Cooper D, Evans E et al. (eds.) *Proceedings of the 4th Symposium on Lift and Escalator Technologies*. Northampton: The CIBSE Lifts Group, pp. 115–126.
19. Rossi F, van Peek P and Walsh T. *Handbook of Constraint Programming (Foundations of Artificial Intelligence)*. New York: Elsevier Science Inc., 2006.
20. Brillouin L. *Science and Information Theory*. New York: Academic Press, 1956.
21. Wilson A. *Entropy in Urban and Regional Modelling*. London: Pion, 1970.

22. Siikonen ML, Susi T and Hakonen H. Passenger traffic flow simulation in tall buildings. *Elevator World* 2001; 49(8): 117–123.
23. Backlund J. *Automatic detection of elevator passengers*. Master's Thesis, School of Electrical Engineering, Aalto University, 2014. In Finnish.

List of Figures

List of Tables