

Smart Building's Elevator with Intelligent Control Algorithm based on Bayesian Networks

Yerzhigit Bapin¹, Vasilios Zarikas^{1,2}

School of Engineering, Nazarbayev University, 53 Kabanbay Batyr ave., Astana, Kazakhstan¹
Theory Division, General Department, University of Thessaly, Lamia, Greece²

Abstract—Implementation of the intelligent elevator control systems based on machine-learning algorithms should play an important role in our effort to improve the sustainability and convenience of multi-floor buildings. Traditional elevator control algorithms are not capable of operating efficiently in the presence of uncertainty caused by random flow of people. As opposed to conventional elevator control approach, the proposed algorithm utilizes the information about passenger group sizes and their waiting time, provided by the image acquisition and processing system. Next, this information is used by the probabilistic decision-making model to conduct Bayesian inference and update the variable parameters. The proposed algorithm utilizes the variable elimination technique to reduce the computational complexity associated with calculation of marginal and conditional probabilities, and Expectation-Maximization algorithm to ensure the completeness of the data sets. The proposed algorithm was evaluated by assessing the correspondence level of the resulting decisions with expected ones. Significant improvement in correspondence level was obtained by adjusting the probability distributions of the variables affecting the decision-making process. The aim was to construct a decision engine capable to control the elevators actions, in way that improves user's satisfaction. Both sensitivity analysis and evaluation study of the implemented model, according to several scenarios, are presented. The overall algorithm proved to exhibit the desired behavior, in 94% case of the scenarios tested.

Keywords—Bayesian network; smart city; smart building; elevator control algorithm; intelligent elevator system; decision theory; decision support systems

I. INTRODUCTION

Environmental degradation and depletion of natural resource force us to pursue sustainable and not greedy way of living. Utilization of smart technologies, such as, Internet-of-Things, smart grid and smart buildings may bolster our advance toward preserving the natural environment. According to the United Nations Organization, 68% of the world population will live in urban areas by 2050 [1]. This suggests that improving sustainability of the multi-floor buildings may have a positive impact on environmental issues. The notion of smart building had been introduced in the early 1980's, and ever since it has been gaining wide popularity among academia and many other fields [2]. Since its introduction, many different definitions, of what a smart building is, have been proposed [3], [4], [5]. Nevertheless, most of these definitions share common idea - a smart building should provide sustainable, secure, effective and flexible environment for its occupants through utilization of integrated technological

systems. Today, a typical smart building solution enables automated control of building's heating, ventilation, air-conditioning, lighting, fire alarm, security and elevator systems. The latter attracts particular interest of the research community, since an effective operation of elevator system is a challenging yet rewarding task.

The Elevator technology has undergone dramatic progress since introduction of an electric elevator by Werner von Siemens in 1880 [6]. Modern elevators are more comfortable, more reliable, faster and spend less energy as compared to their pioneer counterparts. Nevertheless, most of the conventional passenger elevators are not capable of adequately handling heavy traffic of people due to ineffective control system operation. A study by IBM Corporation, conducted in 16 US cities, suggests that office workers spend substantial amount of time waiting for or stuck in elevators [7]. It is evident that the conventional elevator control approaches must be reshaped in order to cope with increasing population density in large megalopolises.

It is quite rare to see a single elevator car serving whole building. Most of the modern buildings are designed to have multiple elevators working back-to-back in order move the continuous traffic of people in a timely manner. When multiple elevators are placed in a group, the elevator group control (EGC) algorithm is used to control their operation. EGC controls each elevator with an objective to minimize a certain cost-function; most commonly, energy consumption and the passenger wait or travel time [8]. Conventional EGC algorithms are based on conditional logic, such that, the elevator dispatching is performed based on the location of elevator cars and passenger calls. More advanced conventional EGC algorithms are capable of changing the elevator dispatch strategy based on the traffic patterns. For instance, in an office building, a weekday morning passenger traffic is often intense because most of the office workers get to work at the same time. The dispatching of the elevator cars, in this case, may be performed with more emphasis on moving people from lobby to their office floors as opposed to inter-floor movements. Another intelligent EGC system, the so-called destination control (DC) system, groups passengers according to their destination. The passengers register their destination floors in the lobby using a specially dedicated electronic system, once the floor is registered, the system will display the elevator car number assigned to the passenger [8]. Modern commercial elevators with DC system can reduce destination time by an average of 30% [9].

Although, existing state-of-the-art EGC systems include features that significantly improve operational efficiency of an elevator system, their major weakness lies in inability to handle the uncertainties caused by unpredictable nature of passenger traffic. The negative impact of these uncertainties on operational efficiency of the elevator system can be mitigated through utilization of Artificial Intelligence (AI) algorithms.

II. AI TECHNIQUES FOR ELEVATORS

One of the earliest works related to implementation of AI into EGC [10] proposed an EGC governed by the Fuzzy logic. The algorithm determines traffic patterns based on the statistical information recorded during its daily operation. The proposed AI-based algorithm was compared to the conventional EGC, the results show 35-40% improvement in the mean landing call time. Somewhat similar, but more recent work is presented by [11]. The study proposes an elevator pattern traffic recognition based on a fuzzy BP neural network with self-optimizing map algorithm. The algorithm detects the traffic patterns by analyzing the existing traffic flows using fuzzy BP neural network. The authors conclude that the traffic pattern recognition greatly increase the effectiveness of EGC strategies.

Other recent works mostly focus on improvement of elevator group control algorithms in terms of electricity consumption or passenger satisfaction and elevator dispatch optimization. In [12] the authors propose EGC algorithm based on the passenger detection and tracking using optical cameras. The main objective of the algorithm is to minimize the passenger wait time and consumption of electrical power by the elevators. The algorithm employs the Haar-like feature-based passenger detection, while the passenger motion tracking is achieved through utilization of the Unscented Kalman Filter. In [13] the authors propose a decision-making model focusing on energy efficiency of elevator systems. The model uses Bayesian networks to dispatch elevators effectively. According to the test results, the proposed framework show reduction in energy consumption as compared to conventional EGC system. In [14], the authors present a mixed integer linear programming (MILP) formulation of the elevator dispatch problem (EDP) with explicitly formulated of operational constraints. In [15], the authors extend their work onto the destination control (DC) elevator systems operating under the collective control (CC) rule. The study focuses on evaluation of the quality of EDP with CC using proposed MILP formulation of EDP. In [16] the authors propose an energy-saving oriented regenerative elevator dispatching optimization strategy that takes into account the stochastic nature of the traffic flow. The proposed model implements a single-objective optimization considering the traffic flow patterns. The study utilizes robust convex optimization method, proposed in [17], to handle the traffic flow uncertainty. The authors consider the number of passengers waiting for an elevator on each floor as the main source of uncertainty. In [18] the authors attempted to develop a model that unifies immediate and delayed call allocation systems to improve elevator dispatching. The former allocates the call immediately after the call was made by a passenger, the later allocates the call just before an elevator is ready to serve the passengers. Based on this model, the authors present an EGC algorithm

which employs a set partitioning model solved by the Branch & Price and Branch & Bound methods. In [19] the authors propose an EGC method for a multi-car elevator system in which the information on floor stoppage time is not known. The method utilizes an optimization-based collision and reversal avoidance technique for simultaneous operation of elevator cars in a single shaft. Similarly, as in [18], the passenger call assignment is done under immediate or delayed call allocation control policy. In [20] the authors attempt to improve the energy-efficiency of an elevator group without compromising the passenger satisfaction. The proposed algorithm takes into consideration dynamically changing electricity price and controls the operation of an elevator group with the objective to minimize total electricity consumption. The optimization problem is formulated as a single-objective minimization problem with predefined passenger wait time constraint. In [21] the authors propose an elevator dispatch optimization method based on Genetic Algorithm. A single-objective cost function aims at reduction of the passenger wait time. The reported results show better performance compared to a conventional EDP algorithm not just in terms of the passenger wait time, but also in terms of computational intensity.

Among existing intelligent elevator solutions, visual-aided systems are one of the most promising research directions. In [22] the authors present an elevator security monitoring method that uses video surveillance cameras to detect hostile behavior of the passengers. A three-level procedure is used by the method to determine violent actions inside the elevator, these are: extraction of foreground blobs, determination of number of passengers and image based motion analysis. In [23] the authors propose a camera-based EGC algorithm to improve energy-saving in elevators. In addition to general information (position of an elevator car, movement direction etc.), the proposed EGC algorithm takes into account the number of passengers waiting for an elevator on each floor to perform energy efficient dispatching of elevator cars. According to the reported results, the proposed algorithm can save up to 20% of energy in down-peak traffic. Somewhat similar approach is proposed in [24]. However, in this paper the main objective is to minimize the passenger wait time through utilization of information from hallway cameras. The gathered data is analyzed by the Region Based Convolutional Neural Network and transferred to conventional elevator control system to perform the elevator dispatch.

This study is an extension of [25] focusing on the intelligent elevator control algorithm based on the visual object recognition and Bayesian network theory.

The rest of this paper is organized as follows. Section III presents general information about Bayesian networks and modeling techniques used by the proposed algorithm. Section IV describes the methodology used to model the elevator control logic. Section V of this study discusses the evaluation of the proposed algorithm. Finally, Section VI summarizes the results of this study.

III. BAYESIAN NETWORKS

Nowadays we see a massive upsurge in Machine Learning (ML), and Deep Learning (DL) algorithms being applied to

solve some real-world problems. These algorithms have found application in many different areas including medical research [26], [27], [28], [29], power system operation [30], image recognition [31], [32], [33] and indoor object tracking [34].

The core of the proposed algorithm, the Bayes' rule, determines the probability of an event, in light of precedent information of conditions that have certain relation to the event. Bayes' rule is built on top of conditional probability and serves as the foundation of Bayesian Inference. Mathematically, Bayes' rule is expressed as follows:

$$p(X | Y) = \frac{p(Y | X)p(X)}{p(Y)} \quad (1)$$

where $p(X)$ and $p(Y)$ are the marginal probabilities of events X and Y respectively. The former term is also called the *prior* probability and represents one's initial belief before any information about event Y is taken into account, whereas the later terms can be considered as a normalizing constant. The conditional probability $p(X|Y)$ represents the probability of an event X occurring given that event Y has already occurred. It is also called the *posterior* probability because it is determined after the information about event Y is taken into account. Similarly, the term $p(Y|X)$, also called the *likelihood*, is the conditional probability of an event Y occurring given that event X has occurred. BNs are defined by their structures, and the probability distribution functions of variables, also called the *node parameters*. Due to the specifics of this study, the further discussion will solely focus on BNs consisting of discrete random variables.

An important part of Bayesian inference, Bayesian network (BN), is a directed acyclic graphical model in which random variables are represented by nodes and causal relationship between the nodes is represented by arcs.

A unidirectional relationship between the nodes imply hierarchical or family-like structure of BNs. The kinship relation of nodes is presented in Fig. 1, where W and Y , for instance, are the *parent* and the *child* nodes respectively. A parent node has some influence on a child node, but not the other way around. All nodes that are hierarchically higher relative to a node of interest are called *ancestor* nodes, whereas hierarchically lower nodes are called *descendant* nodes. Finally, a node with no parents is called a *root* node, and a node with no children is called a *sink* node. The power of graphical representation of probabilistic model lies in the ability to depict the joint probability functions in a compact and coherent way [35]. More detailed description of BN structures and its components can be found in [35], [36] and [37].

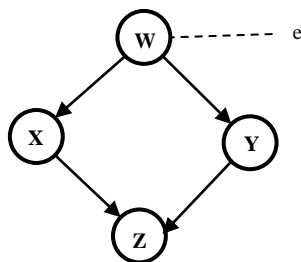


Fig. 1. Example of Bayesian Network.

Another important part of Bayesian inference, parameters of the model, specifies the Conditional Probability Distribution (CPD) at each node. In case of discrete random variables, the conditional relationship between the nodes can be represented in terms of Conditional Probability Table (CPT). The construction of CPT is conducted in the way that [38]:

- Each row represents the conditional probability of a random variable with respect to the values of the parent nodes.
- Each row must sum up to 1.
- The root nodes must have one row.

The computational complexity of BN-based models depends on their structure, number of nodes and the number of states per variable. Several studies show that doing probabilistic inference using BNs is an NP-hard problem [39], [40]. For example, consider nodes X and Y represented in Fig. 1, assuming that both X and Y are dichotomous random variables the resulting CPT will consist of 2^2 possible states.

It would be useful to introduce some general concepts related to BNs in order to proceed further. According to the Local Markov property - a variable is conditionally independent of other variables given its neighbors [41]. The Local Markov property can be generalized to BNs as follows:

$$X_v \perp X_{ND(v)} | X_{PA(v)} \quad (2)$$

where X_v is a random variable represented by a BN node, $X_{ND(v)}$ is a non-descendant node and $X_{PA(v)}$ is a parent node. Consider a simple BN presented in Fig. 1, where X is conditionally independent of non-descendant (W/Y), this yields:

$$p(X | W, Y) = p(X | W) \quad (3)$$

Decomposition of a joint distribution of variables in BN is done using chain rule presented by the following equation:

$$p(X_1, \dots, X_n) = p(X_n | X_1, \dots, X_{n-1}) \times p(X_{n-1} | X_1, \dots, X_{n-2}) \times p(X_2 | X_1)P(X_1) \quad (4)$$

Next, a general form of the chain rule for BN can be derived using equation 4.

$$p(X_1, \dots, X_n) = \prod_{i=1}^N p(X_i | PA(X_i)) \quad (5)$$

A. Variable Elimination Algorithm

Application of BNs in practice bring some difficulties because most of the time we have to deal with large number of random variables each having many different states. A straightforward way to do inference in BNs is to use entire joint distribution and sum out all latent variables [42]. However, for large BNs this task can be very cumbersome, since the full joint probability table for n binary variables will consist of 2^n entries [43]. A simple yet powerful technique called *Variable Elimination* (VA) can be used in order to reduce the computational burden while conducting inference.

A case of calculating a subset of queried variables X given evidence E and latent variables Y is generalized bellow. The conditional probability of X given evidence E is equal to the ratio of the joint probability distribution of X and E to the marginal probability distribution of E :

$$p(X | E = e) = \frac{p(X, E = e)}{p(E = e)} \quad (6)$$

The calculation of the numerator of equation (6) requires marginalization over all latent variables Y_1, \dots, Y_n :

$$\begin{aligned} p(X = x_i, E = e) \\ = \sum_{Y_1} \dots \sum_{Y_n} p(Y_1, \dots, Y_n, X = x_i, E = e) \end{aligned} \quad (7)$$

we introduce *factors* serving as the multi-dimensional tables that we use to avoid duplicate calculations. The joint probability of all variables can be expressed in terms of factors i.e., $f(X, E_1, \dots, E_n, Y_1, \dots, Y_n)$. The joint probability of X and E can be calculated by assigning $E_1=e_1, \dots, E_k=e_k$ and marginalizing out the latent variables Y_1, \dots, Y_n one by one as follows:

$$\begin{aligned} p(X, E_1 = e_1, \dots, E_k = e_k) \\ = \sum_{Y_1} \dots \sum_{Y_n} f(X, E_1, \dots, E_k, Y_1, \dots, Y_n)_{E_1=e_1, \dots, E_k=e_k} \end{aligned} \quad (8)$$

Next, the joint factors can be expressed as a product of factors, by applying the chain rule for BNs (equation (5)), as follows:

$$\begin{aligned} p(X_i | PA(X_i)) &= f(X_i, PA(X_i)) \\ &= f_i p(X, E_1 = e_1, \dots, E_k = e_k) \\ &= \sum_{Y_n} \dots \sum_{Y_1} f(X, E_1, \dots, E_k, Y_1, \dots, Y_n)_{E_1=e_1, \dots, E_k=e_k} \\ &= \sum_{Y_n} \dots \sum_{Y_1} \prod_{i=1}^N (f_i)_{E_1=e_1, \dots, E_k=e_k} \end{aligned} \quad (9)$$

Thus, inference in BNs reduces to computing the sums of products of the last term of equation (8). In order to compute the last term of equation (8) efficiently the terms that do not involve the latent variables must be factored out.

B. Expectation-Maximization Algorithm

Practical implementation of BNs show that we often have to deal with the problem of incomplete data. Sometimes data can be missing due to technical issues in data acquisition system, other times the presence of data can be dependent on values of observed variables [36]. When the probability that the data is absent does not depend on observed values the data is called *missing at random completely* (MARC), whereas when the absence of the data is dependent on observed values, the data is called *missing at random* (MAR). Incomplete data sets can significantly bias the parameter estimates, thus resulting in highly inaccurate probabilistic model. The problem of missing data can be mitigated by implementation of the data generation algorithms. In this study we use the Expectation-Maximization Algorithm to generate randomly missing data.

Given a BN model structure with variables X_1, \dots, X_n we introduce θ_{ijk} - the parameter corresponding to the conditional probability of X_i in state k , at j^{th} configuration of its parent nodes i.e., $p(X_i=k|PA(X_i)=j)$. According to this notation, for a data set $D = \{d_1, \dots, d_m\}$, the likelihood estimate θ'_{ijk} can be found as follows [36]:

- Let $\theta^0 = \{\theta_{ijk}\}$, where $1 \leq i \leq n$, $1 \leq k \leq |SP(X_i)| - 1$, and $1 \leq j \leq |SP(PA(X_i))|$ are the arbitrary initial estimates of the parameters, and $SP(X_i)$ is the state space of X_i .
- Set $t := 0$;
- E-step: For each $1 \leq i \leq n$ calculate the expected counts:

$$\begin{aligned} E_{\theta'}[N(X_i, PA(X_i)) | D] \\ = \sum_{d \in D} P(X_i, PA(X_i) | d, \theta^t) \end{aligned} \quad (10)$$

where N represents the number of counts. This step finds the conditional expectation of the complete-data loglikelihood, given the observed component of the data and the current values of the parameters.

- M-step: Use the expected counts to calculate a new likelihood estimate for all θ_{ijk}

$$\theta'_{ijk} = \frac{E_{\theta'}[N(X_i = k, PA(X_i) = j) | D]}{\sum_{h=1}^{|SP(X_i)|} E_{\theta'}[N(X_i = h, PA(X_i) = j) | D]} \quad (11)$$

Set $\theta^{t+1} := \theta'$ and $t := t+1$.

This step consists of simply performing a maximum likelihood estimation of θ , assuming that the data is complete.

Repeat steps 3 and 4 until convergence or until other stopping criteria are met.

IV. MODELING OF ELEVATOR CONTROL LOGIC

The proposed algorithm is applied on top of the *collective control strategy*, where an elevator control algorithm dispatches an elevator such that it travels in one direction and stops only to pick up people who travel in the same direction. When all requests in that direction have been exhausted the elevator will run in another direction or stays in an idle state in case there are no more elevator calls. The elevator control algorithm, proposed by this study, sends commands to an elevator system based on the information about the size of the group of people waiting for the elevator. This information is acquired by digital cameras installed in the lobby, hallways and in front of the elevator doors, and processed by an image processing system on a real-time basis. Discussion related to the data acquisition and image-processing system falls beyond the scope of this study; therefore, this section focuses merely on description of the structure and parameters of BN used to control elevator cars.

As discussed in the previous section, the Bayesian inference requires updating the probability distributions of the variables based on new evidence. In this study, we assume that the group size measurements are conducted every 30 seconds and this information is sent to the control system with random interruptions. The Expectation-Maximization algorithm,

described in the previous section, is used to ensure that the control algorithm receives complete data sets. The proposed algorithm optimizes elevator dispatching based on the variables representing the passenger group size, their waiting time and the location of an elevator car during the call. The parameters of the dispatching priority can be adjusted according to the user preference; i.e. a user can assign higher priority to the waiting time variable, thus reducing overall passenger waiting time but at the cost of higher consumption of electricity. Fig. 2 depicts the graphical model of the proposed algorithm for an upward direction. The downward direction model is similar to the upward with some difference in the BN structure.

The group size variable determines the number of people waiting for an elevator. Categorization of the group size data must be done based on the size of an elevator car. For instance, for an average-size elevator car the group size data categorization should be done as follows: 0 passengers – *none* (N), 1-2 passengers – *medium* (M) and 3 or more passengers – *high* (H). These categories can be changed based on the user preference and the size of the elevator car, however it is

important to keep in mind that a very excessive number of group categories may cause an increase in computation time while have little or no effect on overall performance of elevator dispatching. The group size may vary due to random movements of the group members. In some instances, people may just be passing by an elevator and caught by the camera, or decided to use stairs after waiting for several minutes. The update of marginal probabilities of the group size node must be done taking into account such instances. For this reason, it is important to represent the group size node in terms of probability of this node being in certain states. For example, for 3th floor: 3 persons with certainty 70% or 2 persons with certainty 20%. The uncertainty is due to occasionally poor lighting or to walking persons etc.

Next, the algorithm proceeds with calculation of CPT of each node by applying the fuzzy Rules. Note that throughout this paper, F_i refers to a building floor where subscript i represent the floor number and n the total number of floors in a building.

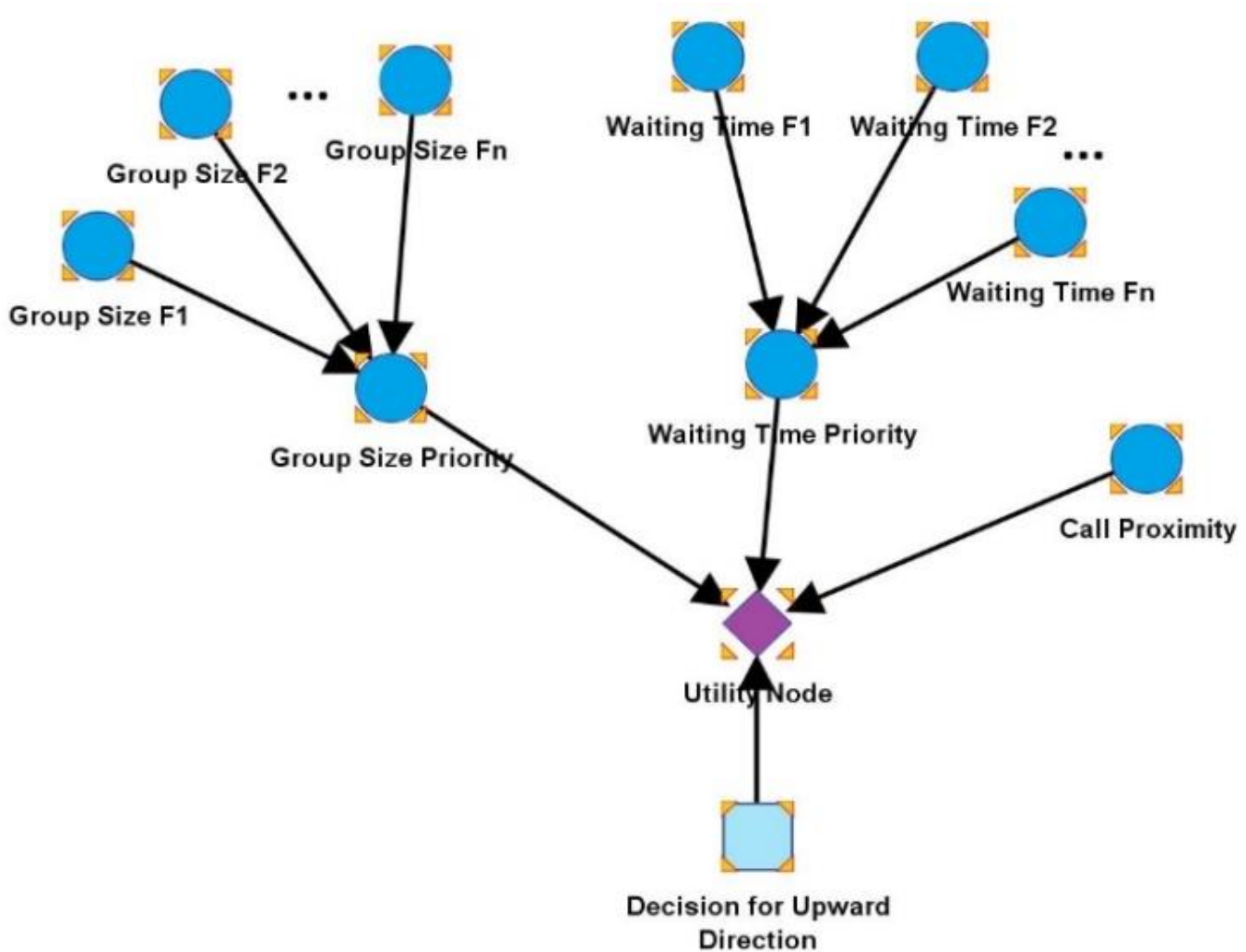


Fig. 2. Graphical Model of the Proposed Algorithm for an Upward Direction.

The fuzzy Rules for upward direction at are:

- If F_i is in state H neglect other floors and go to F_j .
- If F_i is in state M go to the floor with state H except for F_{n-1} . If there are several floors in state H assign equal priority to each one of them.
- If F_i is in state M go to F_j if all other floors are in state M or N .
- If F_i is in state N then go to a floor in state M or H . If there are several floors in state M or in state H assign equal priority to each of them.
- If all floors are in state N go to F_j .
- If F_i is in state H neglect the group sizes of other floors and go to F_i , except if F_j is in state H . If there are several floors in state H assign the same priority to each of them.
- If F_i is in state M go to the floor with state H . If there are several floors in state H assign the same priority to each of them.
- If F_i is in state M go to F_j if all other floors except F_j are in state M or N . If there are several floors in state M assign the same priority to each of them.
- If F_i is in state N go to the floor in state H or M . If there are several floors in state M or independently in state H assign equal priority to each of them.

There are no rules for F_n because this study analyzes only upward direction. Similar rules describe the downward direction.

Besides the floor states, the proposed algorithm considers other factors such as proximity of an elevator car to a caller and how long the caller has been waiting for an elevator. Final decision on where to send an elevator car first significantly depend on these variables. The video cameras installed in the lobby, hallways and in front of the elevator doors send images to the image processing algorithm every 30 seconds. The image processing algorithm determines the number of people and their waiting time and reports this information to the control unit. Similar to the group size variable the waiting time variable is set in terms of probabilities to account for random movements of people in front of the elevator doors.

To account for waiting time a set of fuzzy Rules is exercised by the algorithm. The states of this variable are 1-30 seconds-*short* (S), 31-60 seconds-*average* (A) and 61 seconds or more-*long* (L).

The fuzzy Rules for upward direction at are:

- If F_i is in state L neglect other floors and go to F_j .
- If F_i is in state A give priority to F_j except if there are floors in state L . If there are several floors in state L assign equal priorities to each of them.
- If F_i is in state S give priority to F_j except if there are floors in state A or L . If there are several floors in state

A or independently in state L assign equal priorities to each of them.

- If F_i is in state L then assign priority to F_j with the exception of F_j . If there are many floors in state L assign the same priority.
- If F_i is in state A give priority to F_j except for F_j in with waiting time in state A and except case where there are floors in state L . If there are several floors in state A assign equal priorities to each of them.
- If F_i is in state S give priority to F_j except for F_j with waiting time in state S and except there are other floors in state L or A . If there are several floors in state S assign equal priorities to each of them.

Finally, the third critical information that will be utilized in the present model is the factor of proximity. This third factor can be represented with just a determining informational node without parents. The reason is that there is always availability of the piece of information, about the floor that the cabinet is. There are no fuzzy rules for this issue and there is no uncertainty. The update evidence process for the BN assigns the probability one to one of the five floors which are states of this proximity node. Thus in the proposed algorithm the elevator car location variable is represented as an evidence.

The structure of the proposed model includes utility and decision nodes. In general, the utility node represents a variable accountable for aims and objectives of the controlled action. Often, these nodes determine the decision maker's choice over the outcome of the parent nodes. The decision node represents a variable that can be controlled by the decision maker and thus is utilized to predict decision maker's choices [44]. It is important to note that implementation of this kind of decision-making framework will require adjustment of the functions determining the parameters of multi-objective elevator dispatch strategy.

A mutually exclusive variable A_i where $i = 1, \dots, n$, representing action commands along with three variables H^a with possible states H_j where $j = 2, \dots, m$ representing hypothesis influencing the decision. Another important feature of the proposed algorithm is that the action commands do not have any correlation with $P(H)$.

Finally, a utility node $U(A_i, H_j)$ determining action commands A_i and hypothesis states H_j must be determined. The expected utility responsible for action commands is defined as follows:

$$EU(A_i) = \sum_{a=1}^3 \sum_{j=1}^N (A_i, H_j^a) P(H_j^a) \quad (12)$$

The action commands that have maximum expected utility (MEU) value are sent to the control unit.

$$MEU(A_i) = \max EU(A_i) \quad (13)$$

In our influence diagram there are three determining variables that influence the utility node. This Utility node attributes utility values in cardinal scale to the states of the decision node. The decision node has as states: GoFloor 1,

GoFloor 2, ..., GoFloor 4. The BN designer of this elevator decision making is now responsible to develop a strategy for the overall utility in order to assign the correct weight/utility to the various combinations of states of the three determining nodes. The whole procedure needs a two or three stages evaluation scheme in order to correct wrong weights that lead to unreasonable decision i.e. we want to avoid the elevator going more often to some floors without any particular reason but due to wrong weights.

The form of all fuzzy rules utilized by the proposed BN are:

- If F_1 is in state A and F_2 in state B and F_3 in state C ...then more (much less, less, more, much more) priority must be given to F_x .
- The waiting time priority is assigned as follows: "If F_1 is in state A and F_2 in state B and F_3 in state C ... then the waiting time priority of F_x is (much less, less, equally, more, much more) strong".

Finally, the conversion of the string type fuzzy sets into numerical values is required in order to calculate the CPTs of each node. Conversion of the fuzzy sets is conducted through defuzzification of these sets given their membership functions. For the purpose of this study, the triangular membership function is used to represent the fuzzy sets.

V. EVALUATION

Evaluation of the proposed algorithm was conducted taking into account 35 dispatching scenarios for upward and 35 scenarios for downward direction. Each scenario is characterized by a different set of evidence and derivation of these scenarios was done based on a random set of all possible combinations of evidence nodes states. The final set of scenarios was selected such that the trivial or repeating scenarios were not considered. The main goal is to analyze the scenarios and come up with the list of elevator control decisions. Next, this list was compared to the so-called "golden" decisions reported by our experts. The flow chart of the algorithm evaluation procedure is presented in Fig. 3.

Three rounds of experiments were conducted in order to evaluate the proposed algorithm. The first round resulted in 68% similarity of the exercised decisions with the golden rules. To tune the algorithm performance the variables affecting unexpected decisions were assigned with adjusted probabilities in CPT and weights in the utility table. The second round of experiments showed improvement of the overall algorithm performance. The similarity with the set of golden decisions was 85%. Another adjustment of the variable parameters resulted in 94% of similarity with the golden decisions.

The proposed BN model was implemented in BayesiaLab software. The sensitivity analysis as a part of evaluation procedure was conducted on 30 random experiments. Various combinations of evidence data was updated for each randomly selected case. This was done using BayesiaLab built-in feature. To conduct the sensitivity analysis an additional variable indexing the values of probabilities in question was utilized. BayesiaLab calculates the strength of influence of each value on the result, and determines the sensitivity of the final decision to changes in prior or posterior probability

distribution. The sensitivity analysis showed that the final decision is sensitive to certain floors. This can be explained by the fact that the bottom floors have more influence on the overall dispatch strategy due to fuzzy rules. Nevertheless, the final control actions exercised by the elevator control system based on the overall system state were quite accurate.

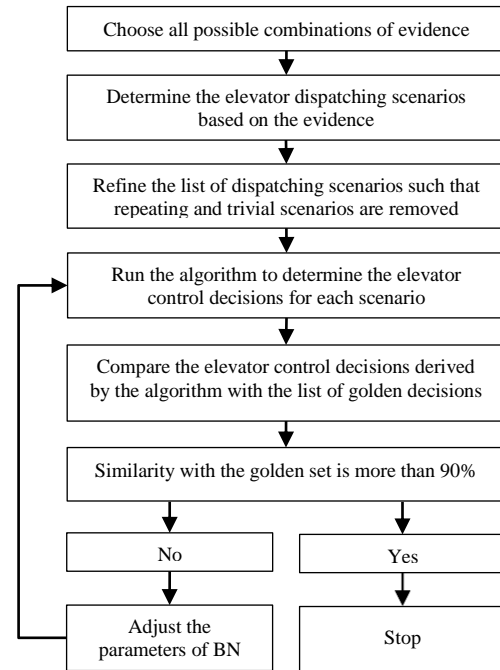


Fig. 3. Flow Chart of Algorithm Evaluation.

VI. CONCLUSION

Firstly, it is important to note, successful implementation of an elevator control strategy, such as an algorithm proposed by this study, in practice will require considering existing elevator control practice. This information is crucial in order to properly tune the elevator control algorithm. The control system was tuned to result in control actions based on the set of fuzzy rules and data provided by image acquisition and processing system.

To implement the proposed algorithm a BN model was constructed using BayesiaLab. Randomly chosen 35+35 scenarios were analyzed in order to update the network with evidence data. Next, decisions made by the algorithm were evaluated and the probability distributions of BN variables were adjusted to result in better decision making. After couple adjustment, the algorithm showed 94% similarity with golden decisions.

The advantages of the proposed algorithm are:

- Clear and simple graphical data processing model.
- Information with high level of uncertainty can also be included and fully investigated.
- The decision-making strategies can be adjusted according to user preference.
- The decision-making rules are not hard-coded into the algorithm, thus could be adjusted or modified.

- Implementation of new elevator control rules will simply require reassignment of conditional probabilities of various variables or changing the topology of the model.

The disadvantages of the proposed algorithm are:

- A sensitivity analysis must be conducted in order to determine variables that have high influence on final decisions.
- The algorithm implementers must have thorough understanding of not only elevator control and dispatching but also BNs and probabilistic inference in general.

Important aspects related to implementation of the proposed algorithm are:

- Derivation of fuzzy rules was conducted in coordination with the field experts.
- Conversion of fuzzy rules to numerical values have been conducted using variable defuzzification with three-stage algorithm tuning.
- The number of nodes affecting the utility node is kept at very low level.

Future work will focus on extending this algorithm with the development of a BN based EGC algorithm for large office buildings with multiple elevators.

ACKNOWLEDGMENT

This work was supported by NUIG Grant funded by Nazarbayev University.

REFERENCES

- [1] O. L. Frank, "INTELLIGENT BUILDING CONCEPT: the challenges for building practitioners in the 21st century," *AARCHES*, vol. 6, no. 3, pp. 107–113, 2007.
- [2] J. Sinopoli, *Smart building systems for architects, owners, and builders*. Amsterdam: Elsevier/Butterworth-Heinemann, 2010.
- [3] O. Y. Ercoskun, *Green and ecological technologies for urban planning: creating smart cities*. Hershey, PA: Information Science Reference, 2012.
- [4] D. Clements-Croome, *Intelligent buildings: an introduction*. New York: Routledge, 2014.
- [5] "68% of the world population projected to live in urban areas by 2050, says UN," United Nations, 16-May-2018.
- [6] A. Bernard, *Lifted: a cultural history of the elevator*. New York: New York Univ., 2014.
- [7] "Smarter Buildings Survey," IBM Corporation, Apr. 2010.
- [8] L. Al-Sharifi, "Introduction to elevator group control," *Lift Report*, vol. 42, no. 2, pp. 59–68, 2016.
- [9] "Driving Urban mobility," <https://www.schindler.com>. [Online]. Available: <https://www.schindler.com/za/internet/en/mobility-solutions/products/destination-technology/destination-control-technology.html>. [Accessed: 13-Feb-2019].
- [10] M.-L. Siikonen, "Elevator Group Control with Artificial Intelligence," KONE Corporation, Helsinki, Finland, tech., 1997.
- [11] Z. Yang and W. Yue, "Elevator Traffic Pattern Recognition Based on Fuzzy BP Neural Network with SOM Algorithm," *Advances in Modelling and Analysis B*, vol. 60, no. 4, pp. 630–645, 2017.
- [12] Z. Zhang, Y. Zheng, H. Xu, and H. Li, "A novel elevator group control algorithm based on binocular-cameras corridor passenger detection and tracking," *Multimedia Tools and Applications*, vol. 74, no. 6, pp. 1761–1775, 2013.
- [13] V. Zarikas, N. Papanikolaou, M. Loupis, and N. Spyropoulos, "Intelligent Decisions Modeling for Energy Saving in Lifts: An Application for Kleemann Hellas Elevators," *Energy and Power Engineering*, vol. 05, no. 03, pp. 236–244, 2013.
- [14] M. Ruokokoski, H. Ehtamo, and P. M. Pardalos, "Elevator dispatching problem: a mixed integer linear programming formulation and polyhedral results," *Journal of Combinatorial Optimization*, vol. 29, no. 4, pp. 750–780, 2013.
- [15] M. Ruokokoski, J. Sorsa, M.-L. Siikonen, and H. Ehtamo, "Assignment formulation for the Elevator Dispatching Problem with destination control and its performance analysis," *European Journal of Operational Research*, vol. 252, no. 2, pp. 397–406, 2016.
- [16] J. Zhang and Q. Zong, "Energy-saving-oriented group-elevator dispatching strategy for multi-traffic patterns," *Building Services Engineering Research and Technology*, vol. 35, no. 5, pp. 543–568, 2014.
- [17] A. Ben-Tal and Nemirovski A. Robust convex optimization. *Math Oper Res* 1998; 23: 769–805.
- [18] . B. Hiller, T. Klug, and A. Tuchscherer, "An exact reoptimization algorithm for the scheduling of elevator groups," *Flexible Services and Manufacturing Journal*, vol. 26, no. 4, pp. 585–608, 2013.
- [19] S. Tanaka, D. Hoshino, and M. Watanabe, "Group control of multi-car elevator systems without accurate information of floor stoppage time," *Flexible Services and Manufacturing Journal*, vol. 28, no. 3, pp. 461–494, 2016.
- [20] S. Ahn, S. Lee, and H. Bahn, "A smart elevator scheduler that considers dynamic changes of energy cost and user traffic," *Integrated Computer-Aided Engineering*, vol. 24, no. 2, pp. 187–202, 2017.
- [21] E. O. Tartan and C. Ciftlikli, "A Genetic Algorithm Based Elevator Dispatching Method For Waiting Time Optimization," *IFAC-PapersOnLine*, vol. 49, no. 3, pp. 424–429, 2016.
- [22] Y. Lee, T. Song, H. Kim, D. Han, and H. Ko, "Hostile intent and behaviour detection in elevators," 4th International Conference on Imaging for Crime Detection and Prevention 2011 (ICDP 2011), 2011.
- [23] J. Wang, Y. Shen, S. Wang, Q. Zhao, K. Nakamura, H. Yamada, and T. Tanaka, "Energy-saving algorithm for elevator group control system with cameras," *Proceeding of the 11th World Congress on Intelligent Control and Automation*, 2014.
- [24] S.-Y. Chou, D. A. Budhi, A. Dewabharata, and F. E. Zulvia, "Improving elevator dynamic control policies based on energy and demand visibility," 2018 3rd International Conference on Intelligent Green Building and Smart Grid (IGBSG), 2018.
- [25] V. Zarikas, N. Tursynbek, *Intelligent Elevators in a Smart Building*, FTC 2017 - Future Technologies Conference 2017. 29-30 November 2017. Vancouver, BC, Canada.
- [26] A. Eleftheriadou, S. Deftereos, V. Zarikas, G. Panagopoulos, S. Korres, S. Sfetsos, C. Karageorgiou, E. Ferekidou, S. Kandiloros. 2009. Test - retest Reliability. VEMP eliciting in normal subjects. Normative data of Vestibular Evoked Myogenic Potential Stimulation (VEMPS) in a large healthy population, *Otolaryngol Head Neck Surg.*, vol. 38, no .4, pp. 462-473
- [27] V. Zarikas, E. Papageorgiou, and P. Regner, "Bayesian network construction using a fuzzy rule based approach for medical decision support," *Expert Systems*, vol. 32, no. 3, pp. 344–369, 2014.
- [28] J. Xia, J. Gateno, J. Teichgraber, P. Yuan, J. Li, K.-C. Chen, A. Jajoo, M. Nicol, and D. Alfi, "Algorithm for planning a double-jaw orthognathic surgery using a computer-aided surgical simulation (CASS) protocol. Part 2: three-dimensional cephalometry," *International Journal of Oral and Maxillofacial Surgery*, vol. 44, no. 12, pp. 1441–1450, 2015.
- [29] K. Lykeridou, M. Lambadiari, V. Raftopoulos, M. Noula, E. Papageorgiou, E. Kapreli, D. Bourdas, D. Mastroggiannis, V. Zarikas, and A. Deltsidou, "Reliability analysis of Finometer and AGE-Reader devices in a clinical research trial," *International Journal of Reliability and Safety*, vol. 11, no. 1/2, p. 78, 2017.
- [30] Y. Bapin. V. Zarikas. Probabilistic Method for Estimation of Spinning Reserves in Multi-Connected Power Systems with Bayesian Network-

- Based Rescheduling Algorithm. International Conference on Agents and Artificial Intelligence. 2019.
- [31] T. Ashfaq and K. Khurshid, "Classification of Hand Gestures Using Gabor Filter with Bayesian and Naïve Bayes Classifier," International Journal of Advanced Computer Science and Applications, vol. 7, no. 3, 2016.
- [32] T. K. Soon, A. Samad, N. Khilwani, B. Hussin, A. Idris, N. Mohd, M. Almahdi, and N. A., "The Utilization of Feature based Viola-Jones Method for Face Detection in Invariant Rotation," International Journal of Advanced Computer Science and Applications, vol. 9, no. 12, 2018.
- [33] Y. M. Ahmad, S. Sahran, A. Adam, and Syazarina, "Linear Intensity-Based Image Registration," International Journal of Advanced Computer Science and Applications, vol. 9, no. 12, 2018.
- [34] A. H. Mahafzah and H. Abusaimeh, "Optimizing Power-Based Indoor Tracking System for Wireless Sensor Networks using ZigBee," International Journal of Advanced Computer Science and Applications, vol. 9, no. 12, 2018.
- [35] J. Pearl, *Casuality: models, reasoning, and inference*. Cambridge: Cambridge University Press, 2005.
- [36] F. V. Jensen and T. D. Nielsen, *Bayesian networks and decision graphs*. New York, NY: Springer, 2010.
- [37] T. Koski and J. M. Noble, *Bayesian networks: an introduction*. Chichester: Wiley, 2009.
- [38] A. Amrin, V. Zarikas, and C. Spitas, "Reliability analysis and functional design using Bayesian networks generated automatically by an 'Idea Algebra' framework," Reliability Engineering & System Safety, vol. 180, pp. 211–225, 2018.
- [39] G. F. Cooper, "The computational complexity of probabilistic inference using bayesian belief networks," Artificial Intelligence, vol. 42, no. 2-3, pp. 393–405, 1990.
- [40] S. L. Lauritzen, *Graphical models*. Oxford: Clarendon Press, 2004.
- [41] C. P. de Campos. New complexity results for MAP in Bayesian networks. International Joint Conference on Artificial Intelligence (IJCAD), pp 2100–2106. AAAI Press, 2011.
- [42] M. Richards, "Introduction to Artificial Intelligence," 2008.
- [43] V. Zarikas, E. Papageorgiou, D. Pernebayeva, and N. Tursynbek, "Medical Decision Support Tool from a Fuzzy-Rules Driven Bayesian Network," Proceedings of the 10th International Conference on Agents and Artificial Intelligence, 2018.
- [44] V. Zarikas, "Modeling decisions under uncertainty in adaptive user interfaces," Universal Access in the Information Society, vol. 6, no. 1, pp. 87–101, 2007.