# The Usage of Artificial Neural Networks for Intelligent Lighting Control Based on Resident's Behavioural Pattern

Agne Paulauskaite-Taraseviciene[1], Nerijus Morkevicius[1], Audrone Janaviciute[1], Agnius Liutkevicius[1], Arunas Vrubliauskas[1], Egidijus Kazanavicius[1]
[1]*Centre of Real Time Computer Systems, Kaunas University of Technology, Studentu St. 50–214, LT-51424 Kaunas, Lithuania*
*agne.paulauskaite-taraseviciene@ktu.lt*

***Abstract*—Learning from the behavior of the resident is essential in order to adapt the lightning system and to provide intelligent lighting control based on behavior patterns. Different homes have different conditions and habits which have to be taken into account for the intelligent system to be useful. However, with passage of time even deeply ingrained habits are subject to change. Therefore, a truly intelligent system has to respond to the changing and diverse environment. An intelligent lighting control system that employs artificial neural networks for on-line learning and adaptation and is based on resident's behavioural patterns is presented in this paper. In order to manage problems related to constant increase of data and dynamic environments, a group of algorithms have been improved by implementing a similarity threshold based data replacement algorithm that has been experimentally tested and compared with alternative algorithms.**

***Index Terms*—artificial neural network, behavioural pattern, lighting control, resilient backpropagation, on-line learning.**

## I. INTRODUCTION

In recent years, many different ideas and approaches have been proposed for the development of smart homes. Research on intelligent lighting control has become very important in this field. Most of academic or commercial systems aim to modernize lighting control by focusing on minimization of energy consumption [1], [2]. Therefore, major attention is paid to energy-efficient lighting systems that are able to estimate a sufficient illumination level and control lighting sources based on different features. A strong motivation for such systems emerged from business area where illumination constitutes a large portion of energy consumption costs. In general, such control systems aim to maintain the balance between energy saving and users' visual and/or psychological comfort provision [3]–[5]. However, for residential homes individual comfort plays a more important role than energy costs. Smart home systems that are able to meet the goals for comfort and efficiency could improve the quality of living, particularly for older people or people with disabilities. It is obvious that a truly intelligent and user-friendly lighting system should control illumination according to the behavioural patterns formed by lighting adjustments made by the residents. Another important aspect that must be taken into account is the individual habits that change over time. Resident's visual comfort level must be considered as well. Therefore, a lighting system has to be flexible and dynamic: it has to be able to respond to a changing and diverse environment. Some provided solutions are based on a new training process that must be undertaken every time the resident confirms his wishes for a change [6]. To make a decision after the resident was questioned is not the best solution for two reasons. First of all, the resident himself cannot always identify his own behavioural habits because sometimes habits change slowly, inconsistently or separately for each specific situation. Besides, recognition of habits is one of the intelligent control areas that is under intensive *investigations* [7]. Secondly, an intelligent and unobtrusive lighting control system should be able to adapt and make right decisions in a changing environment without having to make constant requests to approve resident's behavior. To meet all these objectives, methods that are capable to learn and make correct decisions quickly should be developed.

Artificial neural networks (ANNs) have a high potential in terms of creating intelligent systems. The usage of ANNs provide expected functionality for intelligent systems through the learning, adaptation or prediction capabilities. In the context of a smart home, ANNs are mostly used for image recognition and classification, e.g., human face [8], posture [9], or even emotion [10]. ANNs are also applied in environmental control systems [11], [12] that include lighting systems, ensure efficient energy consumption and users satisfaction [13], [14]. Depending on the problem, different methods of ANNs' adaptation can be used. Some researchers aim to control lighting systems by learning statistical data, others try to predict its tendencies. Prediction algorithms for human behaviour do not always provide expected accuracy [15]–[17]. For real-time applications, ANNs are capable of solving problems based on on-line learning strategies by adaption or network retraining when new data is received [18], [19]. To solve this issue, many researchers have proposed various methods for ANN on-line learning optimization [20], [21].

The aim of this paper is to present an intelligent lighting control method that operates according to resident's behavioural patterns and to investigate the artificial neural network learning capabilities in various environmental conditions. In the experimental study, several scenarios,

Manuscript received on November 26, 2014; accepted on January 25, 2015.

including constant and changing resident's habits, were created and tested using different ANN data replacement algorithms. An algorithm based on data similarity threshold is proposed in this paper. The algorithm was created in order to produce decisions for a more accurate and adaptive lighting control.

## II. Problem Statement

Back-propagation neural network (BPNN) is the most popular technique for supervised learning of a neural network [22]. The iRprop+ algorithm with improved weight- Resilient backtracking scheme is the fastest algorithm of all Rprop modifications. However, network training becomes more challenging when data is collected constantly and the neural network must engage in adaptive learning in an on-line mode. The network has to work in this mode in order to provide intelligent lighting control for residential buildings. *On-line* learning means that data is received and collected constantly for further network learning. Sample data sets are stored in a database. This causes a continual increase of data in storage. To maintain quick and accurate learning, some optimization methods or incremental learning techniques [20] could be applied to avoid the rapid growth of the data sets. Several theories have been proposed to validate the learning optimization decision methods created to find the best network solutions and eliminate low-quality solutions [21]. Because of the complexity of algorithms, *achieving high accuracy* results is time consuming and may not be suitable for solving the problem investigated in this paper. To improve learning process efficiency and to avoid unlimited increase of data sets, limitations on data storage may be useful. The decision is to use a finite space for data storage, i.e. to define a natural number $F$, and set a constraint on the size of data storage, $i = 1,2,...,F$ . If the data array $[e_i]$ is full and a new data entry $e_{new} = e_{F+1}$ arrives, one of the data entries $e_i$ can be selected and replaced with the new one according to predefined rules. The old entry can be selected and replaced by the ANN that has the smallest learning error (SLE algorithm), or the old entries can be replaced randomly (RD algorithm). However, inappropriate data replacement can increase inaccuracy of the network if resident's habits of lighting control change over time. This decision is even more complex when those habits change drastically. For example, a resident is standing at the same position and the settings of a certain luminaire change from *Off* to *On*. Each resident wants that the intelligent system to be able to adapt quickly to his changing habits. However, it should be taken into account that some of the actions may be accidental and those changes are temporary. If the resident is strongly convinced that the changes he made are significant and the system must react immediately, it is possible to replace all conflicting data and to get decisions that are strongly influenced by the last actions. Replacing data one by one allows the network to adapt gradually and leads to less rigorous decision-making. However, if the old data entries are replaced with the new ones randomly or by the smallest error, it may be necessary to go over many replacement and retraining iterations until a proper decision is made.

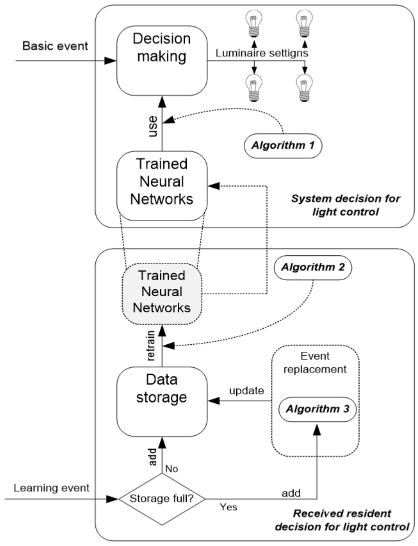



Fig. 1. Decision-making scheme for intelligent lighting control

A decision-making scheme for light control is depicted in Fig. 1. Firstly, data is collected into the storage. When evaluating the algorithm, not only its accuracy, but also execution time and data storage capacity is critical. Therefore, two types of events that influence system decisions are excluded. *Learning events* – actions of the resident related to lighting adjustments. These data entries denote what settings of luminaires are made under what conditions. Learning events are received from a resident in two cases: the system is in initial learning state (data storage is not full); the resident sends his feedback to the system because system's decision does not meet his wishes. In both instances it provokes a retraining process for updating decisions. Each control action is carried out whenever a resident invokes a *basic event*, which indicates that some actions must take place in order to change current lighting control settings. Basic event occurs when the conditions of the environment are changed (resident's location, ambient lighting or time).

## III. ANN-based Lighting Control Decision

In the lighting control system we propose, separate neural networks for each luminaire are created and trained using the data collected from the observations of the resident's behaviour. The pseudo code of a lighting decision made by implementing a trained neural network is presented in Algorithm 1.

**Algorithm 1: control decision using trained *ANN***

**Input:** Data entry $e_{new} = \left\langle e^i_{new}, e^o_{new} \right\rangle$ corresponding to a changed situation in the environment.

**begin**

$r := runANN\left(e^i_{new}\right);$

**if** $\left\| e^o_{new} - r \right\| > 0$ **then**

set luminaire according to $r$;

**endif**

**end**

**Output:** new state of luminaire $r$.

Algorithm 1 uses a new data entry $e_{new}$ that contains data on the situation in the controlled environment after a *basic event* took place. Each data entry $e_{new}$ consists of two components: input $e^{i}_{new}$ − current values of all the sensors, coordinates of the resident, etc.; and output $e^{o}_{new}$ − current state of the controlled luminaire. The corresponding ANN is executed using the values from $e^{i}_{new}$ as inputs. The results that provide new lighting parameters predicted by ANN are assigned to the variable $r$. If the predicted lighting settings differ from the current situation $r$), then the controlled luminaire is set to new settings that correspond to *r*.

After the lighting is set, two possible actions can take place: the resident accepts the decision – he/she leaves everything as it is; the resident disagrees with the decision – he/she generates a *learning event* and sends his/her feedback indicating his/her requests. In the latter case, the system must respond. If data storage is not full, the new data entry $e_{new}$ is added into the data set associated with a particular ANN. In another case, a particular old entry $e_i$ is replaced with the new one $e_{new}$. So the neural network is retrained due to an incorrect decision. In this stage adjustments are made according to the values of the weights and biases of the network. When the next learning event is received, all actions are repeated. The pseudo code of the online training algorithm is presented in Algorithm 2, where the index $idx$ identifies the data entry as the one to be replaced with $e_{new}$.

**Algorithm 2: on-line ANN training**
Input: Data set $e_{new}$ that represents the learning event data;
An array $[e_i]$ of collected data entries from previous actions made by the resident, $i = 1,2,...,n$.
**begin**
  **if** $n < F$ **then**
    $n := n+1$;
    $[e_n] := e_{new}$;
  **else**
    $idx := getIndex()$;
    $[e_{idx}] := e_{new}$;
  **endif**
  $trainANN([e_i])$;
**end**
Output: updated ANN structure.

### A. Algorithm Based on Data Similarity Threshold

Algorithm based on data similarity threshold, named the TB algorithm, is proposed in this research. To get correct decisions faster, the similarity rule based selection of old entries to be replaced with new ones is employed in this algorithm. The point of such selection is that the new entry $e_{new}$, $new > F$ replaces the old entry $e_i$, which is the most similar according to the input variables but has a different output. For example, each data entry consists of a set of inputs (resident's coordinates, values of sensors, time, and settings of luminaires) and one output that corresponds to the preferred settings of the lighting system. A different output condition allows us to avoid a vicious cycle of the same entry selection and replacement. Here we are dealing with another question - how to determine the difference between outputs? Everything is clear when we have luminaires with *On/Off* control. But if luminaires are with dimmers, specific deviation limits that describe possible variations for each lighting level should be specified. For example, outputs of dimmable luminaires are distinct if the difference between their brightness is more than a predefined threshold value $Oth$. By analogy, similarity of inputs can be defined by using a threshold value $Ith$ between components of particular inputs. For example, if the coordinates of the resident in two data entries $e_i$ and $e_{new}$ differ less than $Ith$, then these two inputs are considered as similar. If there are no similar data entries to the new entry $e_{new}$, the data set remains the same. In such case, a second replacement algorithm should be executed. The replacement can be carried out randomly or by taking an entry with the smallest ANN learning error. Our experiments proved that the smallest learning error (SLE) algorithm is more accurate than the random algorithm (RD). Therefore, SLE algorithm is chosen for data selection and replacement in the proposed algorithm. The pseudo code of TB is presented as Algorithm 3.

**Algorithm 3: selection of data for replacement (getIndex)**
Input: An array $[e_i]$ of collected data entries of actions made by the resident, $i = 1,2,...,F$ ;New data entry $e_{new}$.
**begin**
***sim* :=maxfloat;**
**for** $i = 1,2,...,F$
  **if** $\left\| e^{o}_{i} - e^{o}_{new} \right\| > Oth$ **then**
    **if** $\left\| e^{i}_{i} - e^{i}_{new} \right\| < Ith$ **and** $\left\| e^{i}_{i} - e^{i}_{new} \right\| < sim$ **then**
      **j:=i;** $sim := \left\| e^{i}_{i} - e^{i}_{new} \right\|$;
    **endif**
  **endif**
**end for**
**if *sim*=maxfloat then**
  $r_{\min} := \left\| e^{o}_{1} - runANN\left(e^{i}_{1}\right) \right\|$; **j:=1;**
  **for** $i = 2,...,F$
    $r := runANN\left(e^{i}_{i}\right)$;
    **if** $\left\| e^{o}_{i} - r \right\| < r_{\min}$ **then**
      $r_{\min} := \left\| e^{o}_{i} - r \right\|$; ***j*:=*i*;**
    **endif**
  **end for**
**endif**
**end**
Output: index of entry to replace *j*

Algorithm 3 finds a data entry in the array $[e_i]$ with an output that is different to the output of the new data entry $e_{new}$ based on conditions $\left\| e^{o}_{i} - e^{o}_{new} \right\| > Oth$, but both have similar inputs that satisfy $\left\| e^{i}_{i} - e^{i}_{new} \right\| < Ith$. If such data entry is not found, then the data entry with minimal ANN learning error is selected for replacement.

### B. Accuracy Evaluation

Three different algorithms (RD, SLE, TB) used for data replacement are investigated in this research in order to evaluate and compare the obtained results. Since ANNs outputs acquired for the lighting control task have different formats, it is impossible to compare them directly. Simple luminaire output is *On* or *Off*, while the output for dimmable luminaire is a positive number. To evaluate the accuracy of these algorithms used for lighting control, a formal quantitative measure of accuracy is needed. Ideally, this measure should not be dependent on particular algorithms. Proximity and distance measures that are widely applied in clustering algorithms [23] have been used in this paper with some adaptations to meet our research objectives.

All data that is used to make a control decision can be defined as an input vector $\mathbf{x}=(x_1,x_2,\ldots,x_n)^T$, $\mathbf{x}\in I^n$, where $I^n$ is *n*-dimensional space of input components of mixed types (resident's coordinates, sensor values, settings of luminaires, time of day, etc.). These components include all parameters used for decision-making. In other words, input vector $\mathbf{x}$ has the same information as the input part of a data record $e$.

All parameters that can be controlled by the lighting control system are considered as output vector $\mathbf{y}$, $\mathbf{y}=(y_1,y_2,\ldots,y_m)^T$, $\mathbf{y}\in O^m$, where $O^m$ is *m*-dimensional space of mixed type output components.

Distance between two output vectors $\mathbf{y}_i=(y_{i1},y_{i2},\ldots,y_{im})^T$ and $\mathbf{y}_j=(y_{j1},y_{j2},\ldots,y_{jm})^T$ is defined [23] as:

$$D(\mathbf{y}_i,\mathbf{y}_j)=\frac{1}{m}\sum_{k=1}^{m}d_k(y_{ik},y_{jk}) \tag{1}$$

where $d_k(y_{ik},y_{jk})=¢_{ijk}$ indicates the distance of $k^{th}$ feature of two output vectors $\mathbf{y}_i$ and $\mathbf{y}_j$. All distances $¢_{ijk}$ must satisfy the inequality $0\le ¢_{ijk}\le 1$, then the distance between two vectors is bounded by $0\le D(\mathbf{y}_i,\mathbf{y}_j)\le 1$.

Distances $¢_{ijk}$ must be defined for all different types of output components. In this study, there are two different types of output objects – dimmable luminaire and a simple *On*/*Off* luminaire.

If $k$-th object is a simple luminaire, then the corresponding output vector component is $y_k\in\{On,Off\}$, and the distance $¢_{ijk}$ between $k$-th components of two output vectors $\mathbf{y}_i$ and $\mathbf{y}_j$, is defined as

$$¢_{ijk}=d_k(y_{ik},y_{jk})=\begin{cases}0, & if\ y_{ik}=y_{jk}\\ 1, & otherwise\end{cases}. \tag{2}$$

For a dimmable luminaire object, the output vector component is a positive number $y_k$, $0\le y_k\le mlevel_k$, where $mlevel_k$ is the maximum brightness level for that particular dimmable luminaire. Distance $¢_{ijk}$ between $k$-th components of two output vectors $\mathbf{y}_i$ and $\mathbf{y}_j$, which correspond to dimmable luminaire, is defined [23] as

$$¢_{ijk}=d_k(y_{ik},y_{jk})=\frac{|y_{ik}-y_{jk}|}{R_k}, \tag{3}$$

where $R_k$ is the range of $k$-th component of output vectors over all recorded observations, $R_k=\max_l y_{lk}-\min_l y_{lk}$, $l=1,2,...,T$, where $T$ is the total number of registered observations.

In order to evaluate the accuracy of a lighting control method by using distance measure, all observation data is recorded for a certain period of time as a collection of input and output vector pairs $\mathbf{x}_i,\mathbf{y}_i$, $i=1,2,...,T$, where $i$ is the index of observation. Additionally, output vectors $\mathbf{y}^*{}_i$, $i=1,2,...T$ are calculated in each step using ANNs that predict resident's actions using the information obtained from the input vector $\mathbf{x}_i$. In this way, we have two output vectors for each step: $\mathbf{y}_i$ is the $i$-th output vector, which is a recorded observation of resident's actions; $\mathbf{y}^*{}_i$ is the output vector predicted by ANNs for corresponding values of input vector $\mathbf{x}_i$. If it is presumed that all the data collected from resident's actions is absolutely correct, then the distance $D$ between these vectors $\mathbf{y}_i$ and $\mathbf{y}^*{}_i$, calculated using the formula (1), could be used as the measure of inaccuracy of the lighting control algorithm.

In some cases, distance $D$ has a very clear value. For example, if only one simple *On/Off* luminaire is used in a scenario, then the average distance *D*=0.05 means that the prediction algorithm made mistakes in 5% of all the decisions made.

## IV. EXPERIMENTS

*BiaSim* simulation software for intelligent light control was created in order to perform our experimental study. *BiaSim* is a graphical environment that allows to insert four types of objects: residents (humans); passive dimensional objects (e.g., tables); active dimensional objects, such as luminaires with desirable dimensions, power and control type, and light sensors. In our experiments, different scenarios were created to explore the capabilities of ANN-based algorithms for lighting control with both constant and changing habits of the resident (Fig. 2). A 6.5 m × 4.5 m room was with two tables *T1*, *T2* ( sized 1 m × 2 m and 1 m × 1 m), four luminaires *L1*, *L2*, *L3*, *L4* and three light sensors *S1*, *S2*, *S3* was used for each scenario. Two of these luminaires *L1* and *L2* are dimmable, and *L3*, *L4* are simple *On/Off* desk and night luminaires respectively. Control of the luminaires depends only on the habits of resident *ID1*, who is the only user in the experimental research. A scenario titled "Constant" defines unchanging lighting control habits, while the scenario "Changing" represents a situation where the resident's habits change. It is the *L3* zone that changed in reference to new resident's habits near the *T2* table. Fig. 2 depicts the situation when the habits of the resident have changed near the *T2* table. The *L3* zone of the lighting is significantly smaller.

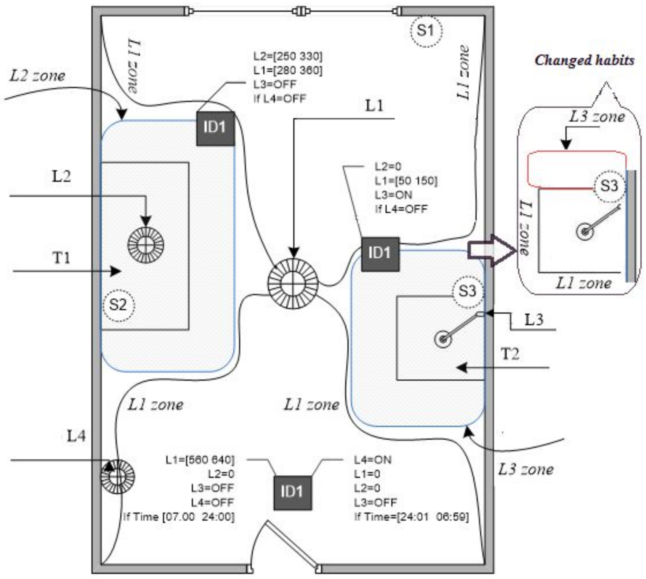



Fig. 2. Predefined lighting control rules and zones with constant and changing habits of the resident near table T2.

Statistical data was collected in the beginning stages of our experiment. Once a sufficient amount was gathered, we were able to divide the room area into separate lighting zones based on lighting settings made by the dweller in different locations of that room. Desirable level of brightness is captured through the sensors values. Actually, this is a target output that determines the settings of luminaires depending on resident's coordinates, time, brightness of location area and settings of luminaires of this area. The aim of this research is to create a more user friendly rather than energy efficient lighting control system. The decision for setting luminaires to a desirable level of brightness is defined by resident's behaviour patterns only.

TABLE 1. PREDEFINED SETTINGS OF LUMINAIRES IN EXPERIMENTS.

| Controllable luminaires | ID location | | |
|---|---|---|---|
| | L1 zone | L2 zone | L3 zone |
| L1 (lm), if L4=Off | 560-640 | 280-360 | 50-150 |
| L2 (lm), if L4=Off | 0 | 250-330 | 0 |
| L3, if L4=Off | Off | Off | On |
| L4 | *Time [ 07:00 24:00]* | | |
| | Off | Off | Off |
| L4 | *Time [ 24:01 06:59]* | | |
| | On, if L1=0,L2=0, L3=Off | | |

The lighting zones specified in experiments define resident's position and settings of luminaires for preferred brightness. All three sensors show < 30 lux when all luminaires are switched *Off* (i.e., it is almost dark outside). Predefined settings of luminaires in each zone are provided in Table 1.

### A. *ANN Structure*

In order to learn resident's habits, a multilayer (one hidden layer) artificial neural network for each controlled luminaire is used in this research. Each ANN has the following structure: 10 inputs (user's $X$ and $Y$ coordinates, values of controllable luminaires *L1-L4*, values of sensors *S1-S3*, and time of day when the recorded event occurred), one hidden layer with 6 neurons, and one output neuron. Before ANNs training, collected data have to be normalized. All numeric values are transformed into real numbers from interval $[-1, \ 1]$. ANN training by employing sample data (i.e. when a previously collected data set is used for ANN training) was performed using the iRprop [22] algorithm.

### B. *Experiments with Unchanging Habits of the Resident*

In order to verify the validity of the proposed algorithm for lighting control, several experiments were carried out. In order to evaluate the obtained results, RA, SLE, TBA algorithms embedded in separate neural networks were tested and compared in a *BiaSim* simulation environment. In the first group of experiments, the analysis was focused on the lighting control accuracy in the scenario "Constant", where resident's habits do not change over time. Experiments were performed in the following fashion. A few sequences of resident's actions for scenario "Constant" have been recorded using *BiaSim* software, which involves all of resident's actions according to the rules described in Table 1, from 21:00 to 00:20. Time after midnight was chosen in order to check system decisions under night lighting rules.

All recorded actions were collected, but the amount of data was several times bigger than the data storage limit used for ANN training. In other experiments, three different algorithms were used to limit the amount of stored data to a predefined level. Lastly, neural networks were trained with a particular sequence of actions filling up the data storage. Using the remaining data ANNs were run simultaneously to provide their control decision for luminaires. In such way, at each step $i$ ($i = 1,2,...,T$, where $T$ is the total number of registered observations or steps) all the required data has been collected to calculate the distance $D$ between vectors $\mathbf{y}_i$ and $\mathbf{y}^*_i$ by using the formula (1). This distance shows the accuracy of each algorithm in predicting resident's actions at each step in the "Constant" scenario.

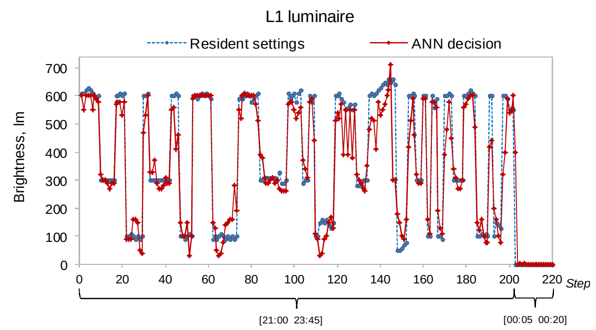



Fig. 3. TB algorithm for dimmable luminaire *L1* with unchanging habits of the resident

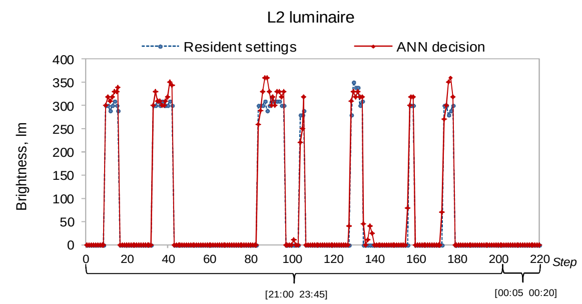



Fig. 4. TB algorithm for dimmable luminaire *L2* with unchanging habits of the resident

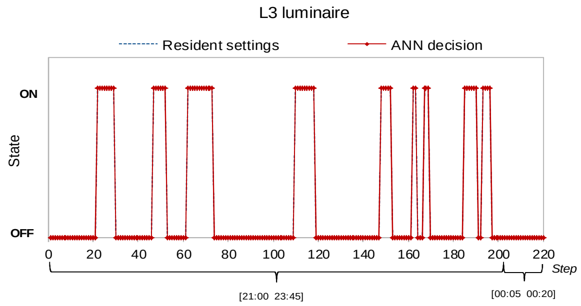



Fig. 5. TB algorithm for simple luminaire L3 with unchanging habits of the resident

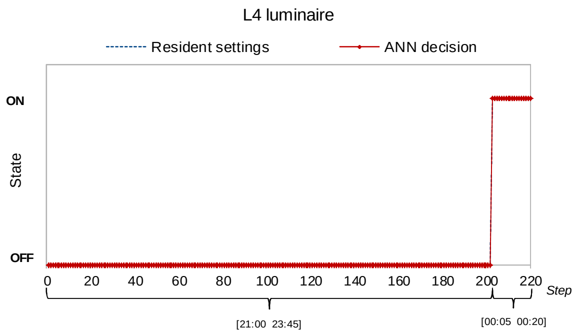



Fig. 6. TB algorithm for simple luminaire L4 with unchanging habits of the resident

The experimental results of the proposed TB algorithm for dimmable luminaires *L1*, *L2*, and simple luminaires *L3*, *L4* are presented in Fig. 3−Fig. 6. Output and input threshold values are defined for data replacement. In the case of dimmable luminaires, threshold value *Oth* corresponds to the change of output by 10%. A threshold value for inputs *Ith*=0.0125 corresponds to the average change of resident's coordinates by 13 cm and to the average change of light sensor values by 10 lux. The data amount limit *F* was 66. The dotted lines represent resident's setting of luminaires and the solid line represents the results obtained from ANNs' control of respective luminaires. Average distance *D* of all luminaires in this case is 0.035. For both simple luminaires *L3* and *L4*, the proposed algorithm TB produced results with the distance value *D*=0, meaning that a 100% accuracy is achieved (Fig. 5 and Fig. 6). For dimmable luminaires *L1* and *L2*, distance value *D* is 0.078 and 0.027, respectively.

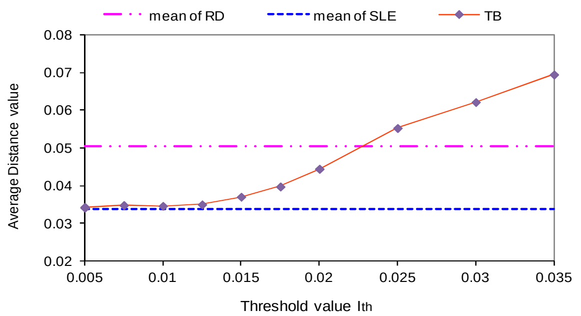



Fig. 7. Influence of Ith value on the accuracy of the TB algorithm with stable habits of the resident.

As has been noted earlier, the proposed TB algorithm is based on threshold values, hence the selection of these values plays an important role here. Experimental results have shown that the threshold value *Ith* = 0.005 should be used when applying the TB algorithm because it produces the smallest average distance *D* (Fig. 7). Theoretically, if the threshold value *Ith* is 0, then TB acts exactly the same as the SLE algorithm, providing average *D* = 0.034. Otherwise, this approach may fail if the threshold value *Ith* is relatively high. TB algorithm with $Ith = 0.025$ becomes worst of all three algorithms with *D* = 0.055 (when coordinates $Ith = 0.025$ correspond to ~30 cm, and to ~22 lux in case of light sensors).

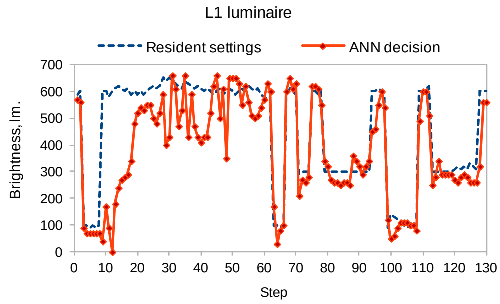
Other experimental results have shown that the RA algorithm performed worst: the obtained average distance was *D* = 0.050. However, distance *D* when applying the RA algorithm may be significantly different, varying from 0.087 to 0.033) over the series of tests performed. It also depends on which data entries have been taken for replacement.

### C. *Experiments with Changing Habits of the Resident*

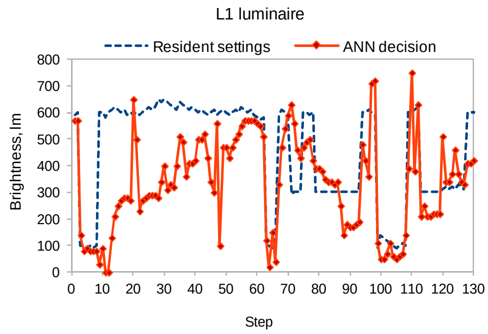
To explore aforementioned lighting control algorithms in situations when resident's habits change over time, another series of experiments were carried out. In *BiaSim* environment, scenario "Changing" was recorded, and it involves several sequences of resident's actions, including changing habits. The aim of these experiments is to find out how quickly ANNs can adapt to the new habits of the resident. Additionally, evaluation was carried out of how data replacement relate to the change of resident's habits and how that influences the decisions made at the zones where lighting settings remain the same (in this scenario, the habits of the resident near the table *T1* are unchanged).

A series of experiments were performed using the following pattern. Two sequences of resident's control actions were recorded. The first part of actions that indicate a change in resident's habits were recorded – resident is walking near the table *T2* but he does so in the *L1 zone* (Fig. 2). In the second part, the same unchanged actions were recorded (e.g., the resident is walking near the table *T2*, but only in *L3 zone*). In such way, we can evaluate how quickly ANNs adapt to changes, and how accurately they are able to produce a decision in an unchanged situation after the last adaptation. The same data sample and trained ANNs were used for all three algorithms. Training was performed with data representing unchanged habits of the resident.

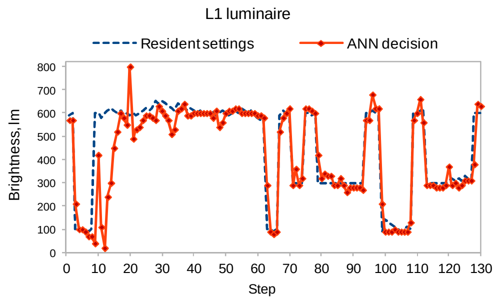
Experimental results are presented below in Fig. 8−Fig. 10. Recorded scenario with 130 resident's control actions (i.e. *T*=130) under day lighting rules was analysed. Whereas all actions were performed until 24:00, the results of *L4* luminaire were not included in this scenario. The first 70 actions (steps) correspond to the changed habits of the resident near the table *T2* in *L1 zone* (Fig. 2). The remaining 60 actions were performed near the table *T1* or in the middle of the room, where habits remained the same. The amount limit for data (*F)* was 66 in this experiment, so only about a half of the whole data set could be used to train ANNs. Output threshold value *Oth* used for this experiment corresponds to the change in output by 10% if luminaire is dimmable, and a different state (*On/Off*) if the luminaire is simple. An additional test was performed to select the most suitable input threshold *Ith*. Value *Ith*=0.02 provides the smallest distance *D*=0.038, which corresponds to the average change of coordinates by 26 cm, and to the average change of 18 lx of light sensors (if a data entry differs less than 18 lx, it is considered similar and thus - suitable for replacement).




a)

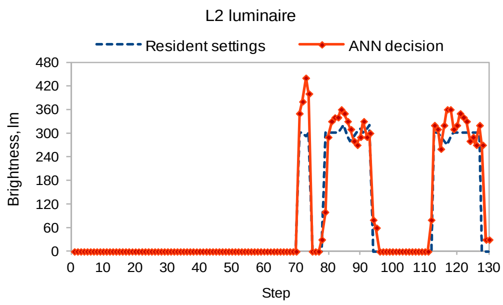



b)

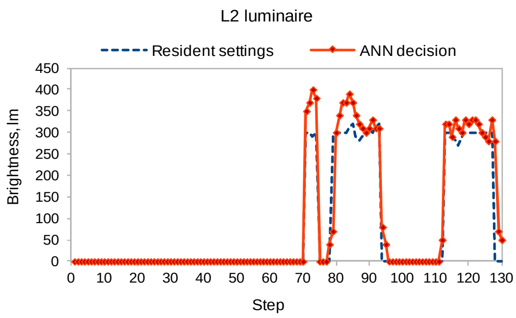



c)

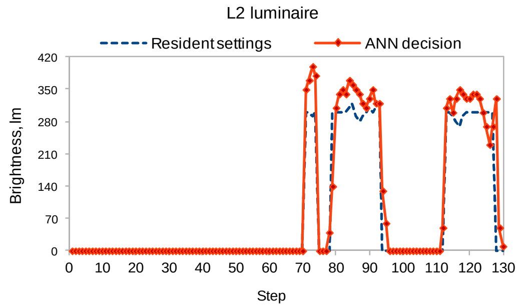
Fig. 8. Lighting control using algorithms a) RD, b) SLE and c) TB for dimmable luminaire *L1*.




a)

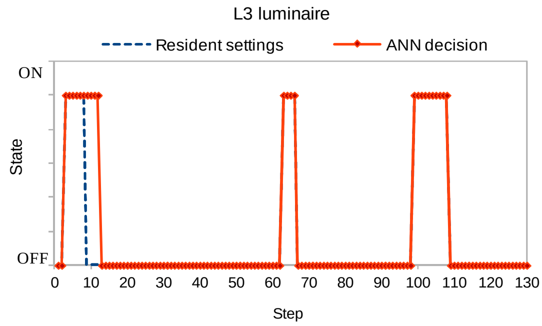



b)

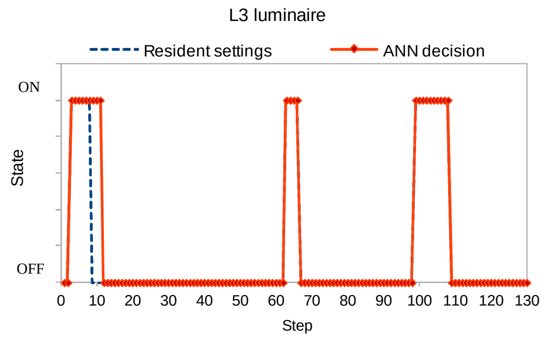



c)

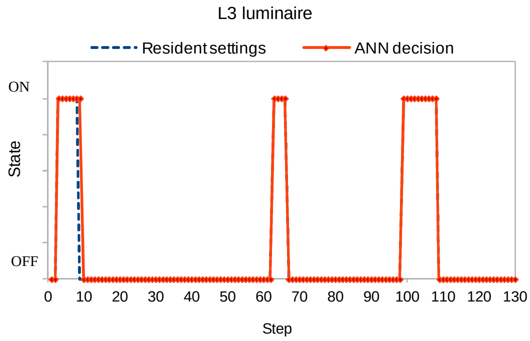
Fig. 9. Lighting control using algorithms a) RD, b) SLE and c) TB for dimmable luminaire *L2*.




a)




b)




c)

Fig. 10. Lighting control using algorithms a) RD, b) SLE and c) TB for On/Off luminaire *L3*.

The experimental study for lighting control showed that SLE algorithm performs significantly worse ($D$=0.087) in a changing environment than in a constant one ($D$=0.034). The average accuracy rate of random data selection $D$=0.074 is better than the SLE, when adaptation to new conditions is required. The major drawback of the SLE algorithm is incorrect elimination of old data that may actually be very important, and that results in decreased overall accuracy.

This may be seen from the results obtained from the first luminaire *L1* control, where steps starting from 70 represent network's decision for the zones with constant lighting settings (Fig. 9 b). TB algorithm adapts to the changes pretty quickly and at the same time provides correct decisions in unchanged lighting zones (Fig. 8 c, Fig. 9 c, Fig. 10 c). Summing up the results, it can be concluded that an algorithm based on data similarity threshold is the most suitable for lighting control in a changing environment.

## V. CONCLUSIONS

The capabilities of artificial neural network usage for on-line lighting control according to resident's behaviour patterns were explored in this research. Finite data storage and different approaches for data replacement were included, allowing to shorten the time for a control decision to be made and to constrain the size of database storage. An algorithm based on data similarity threshold (TB) for data replacement in a neural networks was proposed in this paper. The aforementioned algorithm was tested and compared with two other algorithms focusing on entry replacement based on smallest error (SLE) or at random (RD). Evaluation of the proposed algorithm was performed in two scenarios, where stable and changing lighting control habits are specified. Obtained results show that the superiority of the proposed algorithm TB can be clearly seen in situations with changing habits of the resident. If resident's habits are constant, the TB algorithm acts similarly or identically to the SLE algorithm, if the threshold values are selected properly. The results also show that the RD algorithm for data replacement is not stable and its accuracy may be significantly different each time because it depends on a selected data set for replacement.

The research challenge of finding the best solution for lighting control in situations with more than one resident with conflicting habits is planned in future works. We are also planning to give research directions for the incorporation of prior knowledge and incremental learning techniques that can provide a constructive way to identify the influence of new training data on decisions made by the system.

## REFERENCES

[1] Z. Wang, Y. K. Tan, "Illumination control of LED systems based on neural network model and energy optimization algorithm", *Energy and Buildings*, vol. 62, pp. 514–521, 2013. [Online]. Available: http://dx.doi.org/10.1016/j.enbuild.2013.03.029

[2] C. Carrillo et al., "Lighting control system based on digital camera for energy saving in shop windows", Energy and Buildings, vol. 59, pp. 143–151, 2013. [Online]. Available: http://dx.doi.org/10.1016/j.enbuild.2012.12.012

[3] A. M. Agogino, "Intelligent Commercial Lighting: Demand-Responsive Conditioning and Increased User Satisfaction", *Development and Technology, University of California Energy Institute, UC Berkeley*, 2005. [Online]. Available: http://best.berkeley.edu/~aagogino/papers/Leukos_prepub2006.pdf

[4] A. E. Mady, M. Boubekeur, G. Provan, C. Ryan and K. N. Brown, "Intelligent Hybrid Control Model for Lighting Systems Using Constraint-Based Optimisation", in *5th International Workshop Advances in Intelligent and Soft Computing (SOCO 2010)*, 2010, pp. 249–259.

[5] A. A. Bielskis, E. Guseinoviene, D. Dzemydienė, D. Drungilas, G. Gricius, "Ambient Lighting Controller Based on Reinforcement Learning Components of Multi-Agents", *Elektronika ir Elektrotechnika*, no. 121, pp. 79–84, 2012 [Online]. Available: http://dx.doi.org/10.5755/j01.eee.121.5.1656

[6] Z.Y. Chen, C.L. Wu, L.C. Fu, "Using Semi-Supervised Learning to Build Bayesian Network for Personal Preference Modeling in Home Environment", in *IEEE International Conference on Systems, Man and Cybernetics (SMC 06)*, 2006, pp. 3816–3821. [Online]. Available: http://dx.doi.org/10.1109/ICSMC.2006.384725

[7] A. Aztiria, J. C. Augusto, R. Basagoiti, A. Izaguirre, D. J. Cook, "Learning Frequent Behaviors of the Users in Intelligent Environments", *IEEE Transactions on Systems,* Man, and Cybernetics: systems, vol. 43, no. 6, pp. 1265–1278, 2013. [Online]. Available: http://dx.doi.org/10.1109/TSMC.2013.2252892

[8] M. Balasubramanianr, S. Palanivel and V. Ramalingam, "Real time face and mouth recognition using radial basis function neural networks", *Expert Systems with Applications*, vol. 36, no. 3, part 2, pp. 6879–6888, 2009.

[9] A. Iosifidis, A. Tefas and I. Pitas, "View-Invariant Action Recognition Based on Artificial Neural Networks", *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 3, pp. 412–423, 2012. [Online]. Available: http://dx.doi.org/10.1109/TNNLS. 2011.2181865

[10] G. U. Kharat, S. V. Dudul, "Emotion Recognition from Facial Expression Using Neural Networks", in *Human-Computer Systems Interaction Conference*, 2008, pp. 422–427.

[11] A. Badlani, S. Bhanot, "Smart Home System design based on Artificial Neural Networks", in *Proc. of the Word Congress on Engineering and Computer Science*, 2011.

[12] P. M. Ferreira, A.E. Ruano, S. Silva and E. Z. E. Conceicao, "Neural Networks based predictive control for thermal comfort and energy savings in public buildings", *Energy and Buildings,* vol. 55, pp. 238–251, 2012.

[13] S. Hernandez, C. A. Romero R, D. Giral, "Optimization of the Use of Residential Lighting with Neural Network", in *International Conference on Computational Intelligence and Software Engineering (CiSE)*, 2010, pp. 1–5.

[14] G. H. Stefan, "Neural daylight control system", in *International Conference ILUMINAT*, 2009.

[15] C. Chen, B. Das, D. Cook, "Prediction Using Resident's Activity", in *Proc. of the International Workshop on Knowledge Discovery from Sensor Data (KDD workshop)*, 2010.

[16] R. P. Singh, P. X. Gao, D. J. Lizotte, "On Hourly Home Peak Load Prediction", in *Smart Grid Communications Conf.*, 2012, pp. 163–168. [Online]. Available: http://dx.doi.org/10.1109/SmartGridComm.2012.6485977

[17] A. Shelestov, A. Skakun, O. Kussul, "Complex neural network model of user behavior in distributed systems", *International Conference Knowledge-Dialogue-Solutions*, 2007.

[18] T. Mikluscak, M. Gregor, A. Janota, "Using Neural Networks for Route and Destination Prediction in Intelligent Transport Systems", *Communications in Computer and Information Science*, vol. 329, pp. 380–387, 2012. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-34050-5_43

[19] A. D. Doulamis, D.S. Doulamis and S. D. Kollias, "On-Line Retrainable Neural Networks: Improving the Performance of Neural Networks in Image Analysis Problems", *IEEE Transactions on Neural Networks*, vol. 11, no. 1, pp. 137–155, 2000. [Online]. Available: http://dx.doi.org/10.1109/72.822517

[20] C. Cifarelli, M. R. Guarracino, O. Seref, S. Cuciniello, P. M. Pardalos, "Incremental Classification with Generalized Eigenvalues", in *Journal of Classification*, vol. 24, no. 2, pp. 205–219, 2007. [Online]. Available: http://dx.doi.org/10.1007/s00357-007-0012-z

[21] R. Marti, A. E. Fallahi, "Multilayer Neural Networks: An Experimental Evaluation of On-Line Training Methods", *Computers & Operations Research,* vol. 31, no. 9, pp. 1491–1513, 2004. [Online]. Available: http://dx.doi.org/10.1016/S0305-0548(03)00104-7

[22] C. Igel, M. Husken, "Improving the Rprop Learning Algorithm", in *H. Bothe and R. Rojas, eds.: Second International Symposium on Neural Computation*, ICSC Academic Press, 2000, pp. 115–121.

[23] B. S. Everitt, *Cluster analysis. Chichester,* 5th ed., West Sussex, U. K.: Wiley, 2011. [Online]. Available: http://dx.doi.org/10.1002/9780470977811