ISSN 1392–124X (print), ISSN 2335–884X (online) INFORMATION TECHNOLOGY AND CONTROL, 2015, T. 44, Nr. 2

Solution of the Augmenting Sequence of Linear Programming Problems as a Tool for the Intellectual Home Environment's Self-Training

Andrius Dmuchovskis, Raimundas Jasinevicius, Vaidas Jukavicius, Egidijus Kazanavicius, Laura Kizauskiene, Agnius Liutkevicius

Centre of Real Time Computer Systems, Kaunas University of Technology,
K. Barsausko st. 59-A313, LT-51423 Kaunas, Lithuania
e-mail: andrius.dmuchovskis@ktu.lt, raimundas.jasinevicius@ktu.lt, vaidas.jukavicius@ktu.lt, egidijus.kazanavicius@ktu.lt, laura.kizauskiene@ktu.lt, agnius.liutkevicius@ktu.lt

crossref http://dx.doi.org/10.5755/j01.itc.44.2.10235

Abstract. This work presents the solution based on the augmenting sequence of linear programming problems (LPP) as a tool for intellectualizing home environment. The proposed solution empowers the intelligent decision making procedure which can be applied to various intelligent control applications. The augmenting self-training procedure based on LPP approach is presented as well, which allows making reasonable decisions having only limited data about the controlled environment. The method permits retraining the decision making system when new data is available. As a proof of concept, this solution is applied to intelligent light control application. The obtained simulation results show the method's capability in making reasonable decisions according to users preferences.

Keywords: Linear programming; augmenting self-training; intelligent control; intelligent decision making; smart home environment.

1. Introduction

The distinction between intellectualized and non-intellectualized home environments is usually grounded in certain system properties, like autonomy, self-awareness, proactivity and others. The user is undoubtedly one of the most important elements of this environment and the real presence of intellectics usually resides in modelling his or her activities. Three typical features of intellectual human activities are proposed for implementation and simulation in an agent/multi-agent system as the basic paradigms for agent and multi-agent system intellectics in [21] and [17]. As underlined in the paper [21], operation according to those paradigms (recognition and classification, behaviour according to a set of fuzzy rules, and operation according to some prescribed tendency) is solidly mathematically based (correspondingly: mathematical programing, fuzzy logic and stochastic approximation). This paper is dedicated to the elaboration of mathematical programming and, more precisely, a set of multiple linear programming problem (LPP) solutions as a tool for the training and especially self-training of intelligent home environments.

Linear programming is an optimization technique based on mathematical models that can be used to represent real life situations in the form of a linear objective function and its constraints. Usually, linear programming deals with various problems in business, economy and technological domain, solving various tasks in the field of production planning, staff scheduling or investment management, as well as solving problems in various engineering domains and applications [1, 4-8, 10, 11, 14, 19, 20, 22, 23, 25, 27-31, 33, 35-38, 40-42]. Therefore, the reviewed works do not focus on the decision-making process that concerns user's feedback to the actions performed by the system. The novelty of this paper is, therefore, in that it proposes the usage of linear programming methods for intelligent decision-making and system training based on user control actions in the smart home environment. The paper focuses on the training of the system based on user-defined wishes and desires. The main idea is demonstrated in an example case with simplified home illumination settings that involve not only the desired luminance level, but the preferred lighting devices as well.

This is the first time that LPP is used for the training and adaptation of a smart home control system to the preferences of multiple smart home users. The proposed approach aims to exploit the benefits of linear programming optimally in intelligent decision-making by choosing an optimum decision from a list of alternatives automatically. Adaptation is achieved via the self-training procedure of the system.

The rest of this paper is structured as follows: in Section 2, the proposed method is described; in Section 3, the augmenting self-training procedure based on proposed LPP method is outlined. Results of experiments are presented and discussed in Section 4. In Section 5, the conclusions of the paper are given.

2. Preliminaries

The approach to the problem of environment intellectualization explicated in this paper is based on the theoretical considerations and practical experience delivered in [13].

It is quite natural that users who live in a smart home environment are constantly interacting with it by performing different actions when specific situations that cause the changes in the environment occur. For example, a home user switches the light source on (an action) or increases the illumination in his or her work area (another action) if the illumination in the room is too dim (a situation). A situation is characterized by a set of specific environmental parameters, e.g. illumination level, temperature, humidity level, user's location in the environment and his or her physiological readings. These parameters are further denoted as features of situations. User actions that cause environmental changes are usually related to the control of various household devices, e.g. increasing the light intensity, turning the lights off, turning the heating on and etc., which, in turn, requires a certain type of actuators to be triggered. Similar environmental situations usually cause the same user actions.

According to the above mentioned considerations, an intellectualized environment is also supposed to perform a certain action *p*, which is adequate to the situation that has occurred in the environment. Similar situations must trigger the same action. Therefore, all situations that trigger the same action *p* form a *p*-th class of situations. In a case of multiple actions occurring in the environment, multiple classes of situations are formed and defined as $p = 1,2,\ldots,r,\ldots,S$. It is quite natural that the decisions made by the intellectualized environment should be made according to already known situations and their dependency to particular classes. In a case of a new specific situation arising in the environment, in order to perform an adequate action, the system must first decide to which class of situations it belongs. To determine if a new specific situation belongs to the *p*-th class of situations, the characteristics of the *p*-th class should be extracted and a certain generalized pattern of the situation class under investigation that triggers corresponding action p must be determined and described. In this case, a pattern is defined as a model that describes consistent and recurring characteristics of a particular class. Patterns should be determined for each class of situations respectively. Each situation is then described by N features, numbered as $i = 1,2,\ldots,j,\ldots,N$. The degree of illumination, temperature, humidity, blood pressure and other parameters can play a role of the feature of a situation. In case of a certain feature extraction, measurement and normalization procedures are performed as denoted in [21] Chapter 5 and [13]. The *i*-th feature of a situation that belongs to the *p*-th class (and corresponds to the *p*-th pattern) can be represented by a real number α_{pi} that expresses a degree of intensity of this particular feature. It is convenient to use a vector-row notation to describe the whole situation $\vec{\alpha}_p = (\alpha_{p1}, \alpha_{p2}, \cdots, \alpha_{pj}, \cdots, \alpha_{pN})$.

If we have several situations of the same class *p* (numbered as $l = 1,2,\ldots,k,\ldots,L$) and know in advance that they are similar according to certain criteria (for example, user's wishes and/or user's reactions to the situation) as they are originated by the *p*-th pattern, then it can be concluded that the situations of class *p* are represented by a set of vectors $\vec{\alpha}^l_p$ ($l = 1,2,\ldots,k,\ldots,L$).

Better reasoning results are usually achieved when features of situations are not only normalized but centred as well [13, 16, 18]. The normalization permits to have dimensionless values of the degrees of intensities for further processing that varies in a certain interval, for example [0, 1]. This allows us to use the same verbal evaluation for the same numerical values of variables under consideration. The meaning of centralization according to the formula (1) facilitates a human type reasoning. It becomes easier to understand which values exceed the average of the degrees of intensities and which are lower.

The whole situation is represented as a vector $\vec{\alpha}^{o\,l}_p = \left(\alpha^{o\,l}_{p1}, \alpha^{o\,l}_{p2}, \cdots, \alpha^{o\,l}_{pj}, \cdots, \alpha^{o\,l}_{pN}\right)$ with components calculated according to the following formula [13]:

$$\alpha^{o\,l}_{pi} = \alpha^l_{pi} - \frac{1}{N}\sum_{j=1}^{N} \alpha^l_{pj} \tag{1}$$

So, all available information about the pattern of a situation is hidden in the set of $\vec{\alpha}^{o\,l}_p$, where $p = 1,2,\ldots,r,\ldots,S$ and $l = 1,2,\ldots,k,\ldots,L$. Now the main task is to determine or extract the significance of pattern's feature of each situation class (let's say, class p) and to present them in a vector form denoted as the generalized situation pattern (GSP) for class p: $\vec{K}_p = (K_{p1}, K_{p2}, \ldots, K_{pj}, \ldots, K_{pN})$. The problem can be easily solved once the corresponding linear programming problem (LPP) is formulated in the following way.

Consider selecting a random representative of the class *p,* of a situation, for example, $\vec{\alpha}^{o\,k}_p$. In this case, the requirement is to find such $\vec{K}_p$ so that the measure of a certainty degree $\Phi_p\left(\vec{K}_p, \vec{a}^{o^k}_p\right)$ that the situation $\vec{\alpha}^{o^k}_p$ belongs to the pattern *p* of the selected situation *k* would be maximum:

$$Max \quad \Phi_p\left(\vec{K}_p, \vec{\alpha}^{o^k}_p\right) = \sum_{i=1}^{N} \alpha^{o^k}_{pi} \cdot K_{pi} \tag{2}$$

This must be reached under the following constraints:

$$\sum_{i=1}^{N} \alpha^{o^l}_{pi} \cdot K_{pi} \geq \gamma \sum_{i=1}^{N} \alpha^{o^k}_{pi} \cdot K_{pi}, \forall l \tag{3}$$

and

$$\sum_{i=1}^{N} \alpha^{o^l}_{ri} \cdot K_{pi} \leq \kappa \sum_{i=1}^{N} \alpha^{o^k}_{pi} \cdot K_{pi}, \forall r \neq p, \forall l \tag{4}$$

As recommended in [16], optimal values of the real numbers γ and κ should be selected from the interval [0-1], and $\gamma > \kappa$. Concrete values of these coefficients depend on expert knowledge or choice in terms of the structure of a pattern (internal connections and dispersion of the features of a pattern). Constraints (3) denote, that all situations that belong to class *p* should have bigger (by factor γ) certainty degrees of belonging to the pattern p as a selected random representative from this particular class of a situation. However, constraints (4) denote that all situations that belong to other classes should have smaller (by factor κ) certainty degrees of belonging to the pattern p as a selected random representative from class *p*. In other words, these constraints define the similarities of situations inside the class *p* for *p*-th pattern of situations and dissimilarities between all other situations that belong to different classes. Even a quick investigation of the problem described above shows that the problem actually belongs to the class of linear programing problems (LPP) where inequalities (3) and (4) need additional constraints:

$$0 \leq \vec{K}_p \leq A, \tag{5}$$

where A is any practically convenient real number. Naturally, a solution of the LPP (2)-(5) for the pattern of class *p* of a situation consists of the obtained value for $Max \quad \Phi_p\left(\vec{K}_p, \vec{\alpha}^k_p\right) = \Phi_{p\max}$ and the generalized pattern of situations for class *p* $\vec{K}_p = \left(K_{p1}, \ldots, K_{pj}, \ldots, K_{pN}\right)$.

The procedure must be repeated for all classes $p = 1, 2, \ldots, r, \ldots, S$ (for all corresponding patterns of situations). In this way, a set of S solutions will be obtained. The procedure of situation recognition must be performed considering the necessity to assure the proportionality condition. It means that for each class the same quantitative (numerical) evaluation *B* must correspond to the qualitative (verbal) evaluation, like "very similar". This proportionality condition is guaranteed by fulfilling the following requirement:

$$c_1 \Phi_{1\max} = \cdots = c_r \Phi_{r\max} = \cdots = c_S \Phi_{S\max} = B \tag{6}$$

here *B* and c_r are real numbers.

A block diagram that represents the final decision-making act in the case of a newly appeared unknown situation $\vec{x}^o$ for recognition is shown in Fig. 1. Here, an output in the form of $\Phi_p / \max_{\forall p} \Phi_p$ denotes a normalized proposed degree of certainty of the requirement of the action *p,* as a reaction to the situation $\vec{x}^o$ under consideration. A decision is usually made according to the maximum value of the degree, but different reasonable decisions can be made as well.

A complex of such actions enables to construct a situation recognition instrument capable of assigning any unknown but properly described situation $\vec{x}$ to one of the possible patterns (or classes) and performing the corresponding environmental action $p = 1, 2, \ldots, r, \ldots, S$.

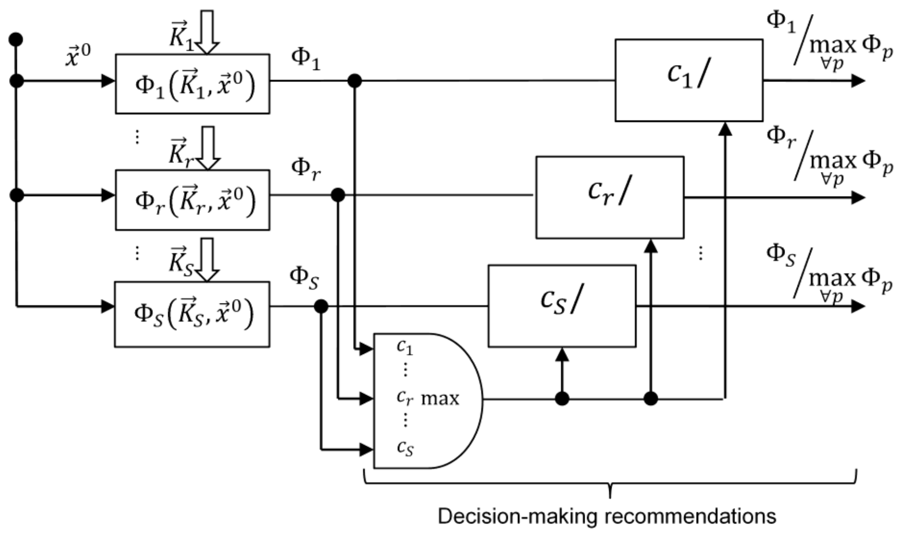

Figure 1. The decision-making procedure

3. The augmenting self-training procedure based on the LPP approach

3.1. The augmenting self-training procedure

At the moment of initiation of the decision-making procedure, there is usually no data available about situations that have already occurred and actions that have been performed. Only the possible actions and features that describe those situations are defined. Whenever an unknown situation $\vec{x}$ arises, the decision-making procedure requires data about similar situations that occurred previously and subsequent actions taken by the user. It is quite obvious that in the beginning, the decision-making procedure acquires only limited data about the situations that occur in the environment. Nevertheless, the decision-making procedure should be able to produce some decisions (appropriate or not) even with limited data \available. It should also retrain itself when additional data about the situations and taken actions appear. New data about the new situations and actions taken can be extracted by observing the environment: constantly reading sensor values and actuator states. When the user changes some actuator state, he gives feedback to the decision-making system, and in doing so, initiates the retraining process. It is important to note that data is gathered even in those situations when the user takes no corrective actions. The system takes it as a sign that the user is satisfied with certain actions produced by the decision-making procedure. Otherwise, he corrects some decisions in order to establish a comfortable environment for himself or herself. In both cases, new data is made available about the situations and actions that should be taken next time when a similar situation occurs. It means that the decision-making system should retrain itself every time new data is available.

A block diagram that represents the self-training procedure is shown in Fig. 2. Let us assume that the sequence of known situations is numbered as $t = 1, 2, \ldots, \tau, \ldots, T$ according to the time moment of when a situation occurred. All possible actions are numbered as $p = 1, 2, \ldots, r, \ldots, S$ and a set of actions already taken at the time moment $t-1$ is defined as P^{t-1}.

When a new situation, described as $\vec{\alpha}_r^t$, occurs at a time moment *t* when an action *r* is taken, the decision-making system checks if this situation is known or not. If the situation is known, then the decision-making system checks if action *r* belongs to P^{t-1}. If action *r* belongs to P^{t-1}, then the decision-making system expands the constraints for all LPPs formed at the time moment $t-1$ according to (3) and (4) by using $\vec{\alpha}_r^t$. Then, the system retrains itself by resolving each modified LPP and acquiring new coefficients K_p^t for $\forall p \in P^{t-1}$. In another case, the decision-making system not only retrains itself by expanding constrains for the mentioned LPP, but also by forming and solving a new LPP for action *r*, according to (2), (3), (4) and (5). By solving the modified and newly formed LPP, the system acquires new coefficients K_p^t

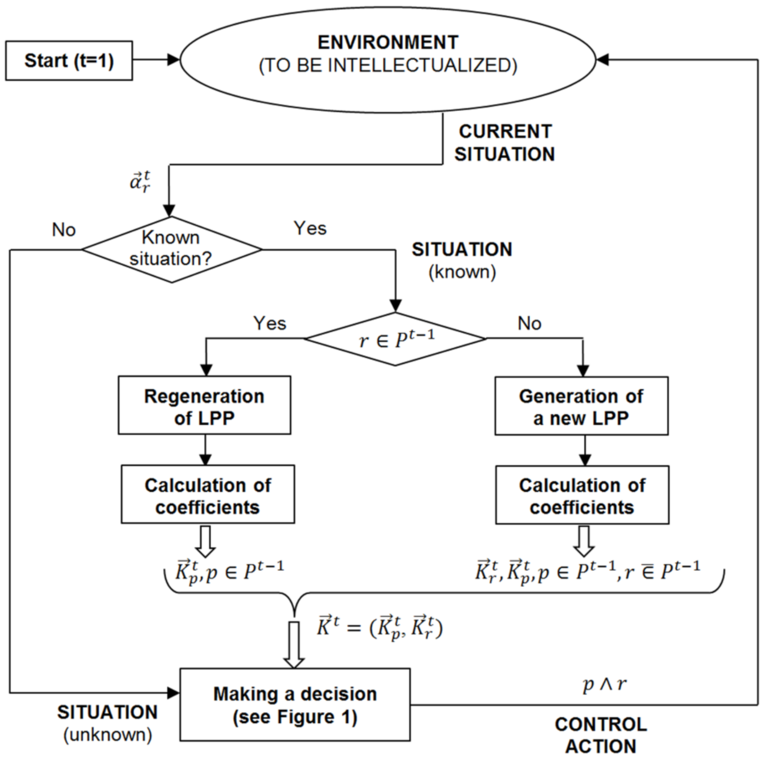

Figure 2. Intellectualization of the environment with the augmenting self-training procedure

for $\forall p \in P^{t-1}$ and K_r^t for $r \notin P^{t-1}$. When an unknown situation occurs, these newly acquired coefficients are used for making decisions, as depicted in Fig. 1.

It is important to note that this self-training procedure is an event-based procedure. The retraining process is initiated only when the user gives feedback to the decision-making system. A new known situation can be one of the two types: completely new or already known but with different action taken, meaning that the user changes his preferences. In both cases the retraining process is practically the same. Still, in the first case, all LPPs become larger or even new LPPs are formed. Also, this process of augmentation will be limited only if user's wishes are limited. In the second case, information about an already known situation is updated and information about old actions is discarded. In this case, the size for all LPPs is the same but new LPPs can be formed as well.

3.2. Intelligent light control as an example of application of the augmenting self-training procedure

The augmenting self-training procedure can be applied to various intelligent control applications. Here we discuss an application of the augmenting self-training procedure as an example of the intelligent light control. The task of this section is to demonstrate the principal viability of the approach while excluding any engineering particularities, such as accuracy, speed and so on.

In general, smart home environments are equipped with various light sources (often controlled in different manners), and particular users with different habits control these light sources according to their preferences. The decision-making system should train itself according to the way each user acts in a particular environment and make a decision that meets user preferences. An important fact is that the decision-making system at the moment of initiation has only limited information about the environment. It only knows that there are some light sources and sensors in a particular environment. Light sources can be controlled by existing actuators, while sensors give light intensity and both emotional and physical measures of the user.

The development of the intelligent light control application based on the augmenting self-training procedure begins with defining possible actions and features that describe the situations. By controlling light sources, each user generally performs one of the three actions: does nothing (the environment is comfortable for him or her), increases or decreases (or switches on or off) the light intensity of a particular light source. The listed actions p are denoted as *DN* (do nothing), *I* (increase) and *D* (decrease).

The features that describe the situation can be generally divided into two groups. One group consists of critical features for decision-making. It is composed of light intensity and the location of a particular user (coordinates in the environment) as given by sensor measures. It is natural that the decision-making system should know how the users are controlling light sources when they are in a certain location. Location of the user can be monitored by using a hybrid ultrasound and radio frequency (RF) technology [34]. In this case, user positioning system is composed of several beacons and listeners. Each user carries one personal beacon that emits synchronized (using RF technology) ultrasound signals to the environment. Listeners are located in the specific areas of the room and listen to the incoming ultrasound and RF signals. The positioning system calculates the exact position of each user by measuring ultrasound propagation times transferred from each personal beacon to all listeners.

Another group of features could consist of emotional [26] (happy, normal, angry, etc.), physical (standing, sitting, laying down, etc.) and biophysical [12] state (blood pressure, heart rate, etc.) measurements as long as the behaviour of the user changes according to these mentioned features and affects their light control habits.

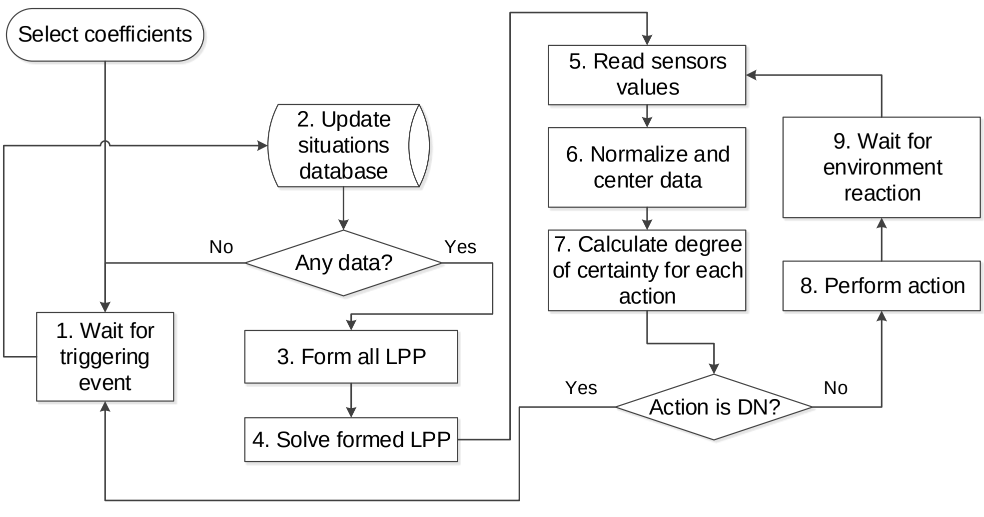

Figure 3. The intelligent light control algorithm based on augmenting self-training procedure (Fig. 2)

A block diagram that represents an intelligent light control algorithm for one light source based on augmenting self-training procedure (as depicted in Fig. 2) is presented in Fig. 3.

As mentioned earlier, decision-making is an event based procedure, thus the algorithm first and foremost waits for the triggering action to occur in the environment (step 1). It can be user interaction (user performs actions) or a significant change of the environment state. Once the triggering event occurs, the main steps for intelligent light control involve the update of situation database, formation of the objective function and constrains according to (2) - (6) for each possible action using all previously known situations and performed actions that are saved in a database and that solve the formed LPP (steps 2-4). For this task, the needed coefficients (γ, κ , A and B) are usually selected by an expert at the initial state of the algorithm. Then the formed LPPs are solved, and the obtained coefficients are used to calculate the degree of certainty for each action (step 7) by multiplying them by the normalized and centred values of the features (step 6). According to the calculated degrees, the control system performs an action (step 8) with the largest degree of certainty adjusting the light source intensity by the chosen actuator step. When the action with the largest degree of certainty is *DN*, then the control system waits for the next triggering event. Otherwise, the control system increases or decreases the intensity of the light source by modifying the position of an actuator and then waits for the reaction of the environment (step 9). Then, the reading of sensors, calculation of the degrees of certainty for each action and light control process is repeated until the action with the largest degree of certainty is *DN*.

An actuator step describes the value denoting how much the light intensity is increased or decreased. The choice of an actuator step is limited to actuator specifications. In cases when the on/off switch is used, it can be chosen only to completely turn on or turn off the light source. When dimmers are used to control the light sources, the minimal available step should be chosen to achieve the best performance. In this case, the control system can adjust the light intensity with the same accuracy as the users do. Bigger step (include several levels of light intensity) can also be chosen to speed up the reaction of the control system. Still, considerations should be made because in the latter case the accuracy of the control system will be decreased since larger actuator steps prevent the control system from fine-tuning its decisions.

When this intelligent light control algorithm is applied to more than one light source, LPPs are formed and all calculations and decisions are made for each light source separately. Still, because each light source (its light intensity) in the environment can influence the decision-making of other light sources, the control system repeats the light control process until the action with the largest degree of certainty is *DN* for all light sources.

4. Results of the experimental simulation and further discussions

Let us assume that the light sources should be controlled according to user preferences in a simple environment, such as depicted in Fig. 4. The two existing light sources can be controlled differently. The first light source (one on the left) is a halogen lamp and can be controlled by an actuator that has 100 positions (0 – the light source is off, 100 – the light is at maximum intensity). The second light source (one on the right) is a fluorescent lamp and can be controlled by an actuator only by switching the lamp on or off. Two illumination sensors are placed on each table below the light sources.

Only three features that describe the situation are taken into consideration: the illumination of each light source, each user's location measures as given by the sensors and an additional feature that indicates the presence of each user in the room.

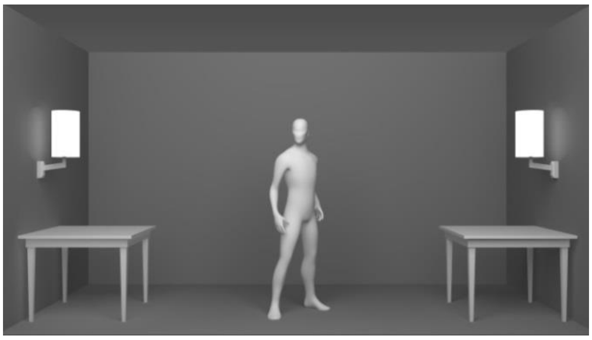

Figure 4. The testing environment

4.1. The formation of LPP and experimental results when only one user is interacting in the environment

In order to demonstrate how the LPPs are formed, we take a case of only one user interacting in the testing environment. His preferences are:

1. all light sources should be turned off when he is in the middle of the room;
2. halogen lamp intensity should be adjusted by selecting the thirtieth actuator position when he is near the table on the left;
3. fluorescent lamp should be switched on when he is near the table on the right.

The testing results are obtained by using the BIAsim simulation tool ([11]) depicted in Fig. 5.

BIAsim tool is used for simulation of the testing environment and for integrating the intelligent light control algorithm. All user positions, when certain learning (denoted as L) or testing situations (denoted as T) occur, are depicted in Fig. 5 as well.

During the first experiment, the user stands in the middle of the room (ambient light illumination is 50 lx) and then he goes to the table on the left and selects the 30th actuator position for the halogen lamp (the measure of the first illumination sensor is 245 lx, and the measure of the second one – 74 lx). Now, the decision-making system obtains for the first time the important information on how to control the halogen lamp when a similar situation occurs. Then he goes back to the middle of the room and switches the halogen lamp off. At this moment the decision-making system registers four situations that are described in Table 1.

As mentioned earlier, situations are described by several properties: the user's presence, which, in this case, is chosen to be 1 when the user is in the room, *X* and *Y* coordinates, and light illumination measures of the halogen and fluorescent lamps illumination sensors *L1* and *L2*. The feature that denotes the user's presence could also have other values, indicating when the user is or is not in the room. But they should considerably differ from each other. The modelled situations simply denote when the light intensity of the halogen lamp should be increased, decreased or left unchanged according to the values of certain features obtained when the user took corresponding actions. Situation data obtained for the fluorescent lamp is described in Table 2. The *DN* action with the fluorescent lamp was taken in all situations, so there is no need to formulate and solve any LPPs in that particular moment.

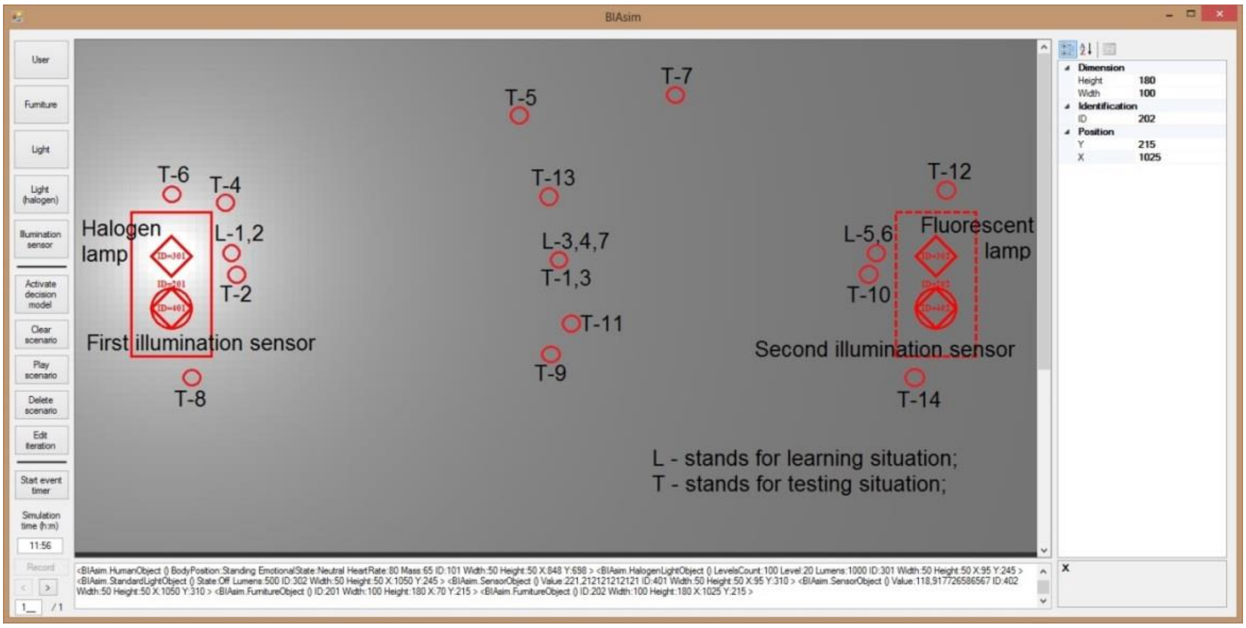

Figure 5. The BIAsim simulation tool used for simulation and testing of various light control systems and environments

Table 1. The data from the situations that occurred near the halogen lamp

Situation number	Presence	*X*	*Y*	L_1	L_2	Action
1	1	163	243	50	50	
1 normalized	0.1	0.105	0.255	0.066	0.066	*I*
1 centred	-0.018	0.013	0.136	-0.052	-0.052	
2	1	163	243	245	74	
2 normalized	0.1	0.105	0.255	0.326	0.098	*DN*
2 centred	-0.077	-0.072	0.078	0.149	-0.078	
3	1	675	263	245	74	
3 normalized	0.1	0.435	0.276	0.326	0.098	*D*
3 centred	-0.147	0.187	0.029	-0.079	-0.148	
4	1	700	251	50	50	
4 normalized	0.1	0.451	0.264	0.066	0.066	*DN*
4 centred	-0.089	0.261	0.074	-0.123	-0.123	

Table 2. The data from the situations that occurred near the fluorescent lamp

Situation number	Presence	*X*	*Y*	L_1	L_2	Action
1	1	163	243	50	50	
2	1	163	243	245	74	*DN*
3	1	675	263	245	74	
4	1	700	251	50	50	

Having data on these four situations, three LPPs can be formed for each action associated with the halogen lamp, and decisions can be made according to the obtained solutions. The selected coefficients are: $\gamma = 0.7, \kappa = 0.35$, $A = 4$ and B=10. For the action *DN*, LPP objective function can be formed using the second or the third centred situation data. Let us assume that the objective function (7) is formed using the second centred situation data, then constraints (8) are defined according to (3) using the third centred situation data, and (4) using the first and the second centred situation data. Note that all data is multiplied by 1000 in order to simplify the expressions. Additional constraints according to (5) must be defined for each LPP as well.

$$-77K_{DN1} - 72K_{DN2} + 78K_{DN3} + 149K_{DN4} - 78K_{DN5}, \quad (7)$$

$$\begin{aligned} &8K_{DN1} + 11K_{DN2} + 109K_{DN3} - 104K_{DN4} - 24K_{DN5} \le 0 \\ &-120K_{DN1} + 213K_{DN2} + 1K_{DN3} + 26K_{DN4} - 121K_{DN5} \le 0, \\ &35K_{DN1} - 312K_{DN2} - 19K_{DN3} + 227K_{DN4} + 68K_{DN5} \le 0 \end{aligned} \quad (8)$$

The second LPP objective function (9) for the action *D* can be formed using only the third centred situation data and constrains (10) defined only according to (4) using first, second and fourth centred situations data because there are no similar situations with the same action.

$$-147K_{D1} + 187K_{D2} + 29K_{D3} - 79K_{D4} - 148K_{D5}, \quad (9)$$

$$\begin{aligned} &32K_{D1} - 79K_{D2} + 126K_{D3} - 79K_{D4} + 0K_{D5} \le 0 \\ &-25K_{D1} - 137K_{D2} + 68K_{D3} + 121K_{D4} - 26K_{D5} \le 0, \\ &-38K_{D1} + 195K_{D2} + 64K_{D3} - 150K_{D4} - 71K_{D5} \le 0 \end{aligned} \quad (10)$$

The third LPP objective function (11) for action *I* can also be formed using only the first centred situation data, and constrains (12) can be defined according only to (4) using second, third and fourth centred situation data.

$$-18K_{I1} + 13K_{I2} + 136K_{I3} - 52K_{I4} - 52K_{I5}, \quad (11)$$

$$\begin{aligned} &-70K_{I1} - 67K_{I2} + 30K_{I3} + 167K_{I4} - 60K_{I5} \le 0 \\ &-140K_{I1} + 192K_{I2} - 18K_{I3} + 97K_{I4} + 130K_{I5} \le 0, \\ &-83K_{I1} + 266K_{I2} + 26K_{I3} - 104K_{I4} - 104K_{I5} \le 0 \end{aligned} \quad (12)$$

Corresponding coefficients $\vec{K}_p$, maximum values of objective functions Φ_p and c_p coefficients are obtained by solving each formed LPP, as denoted in Table 3. These obtained coefficients can now be used for making the control decisions for the halogen lamp when new situations occur.

Table 3. The calculated coefficients

Action *p*	K_{p1}	K_{p2}	K_{p3}	K_{p4}	K_{p5}	Φ_p	Coefficients c_p
DN	4	1.96	1.6	2.2	0	0.0046	2173
D	0	3.37	0	4	0.81	0.83	12
I	1.62	0.11	4	0	0	0.51	19.6

The decision system was tested in 14 different situations (Fig. 5). During the experiments, the user went to each table from different direction and then went back to the middle of the room. The testing results of the halogen lamp control are depicted in Fig. 6. The acquired results confirm that the decision-making system can make reasonable decisions. However, the accuracy of the decisions associated with switching the halogen lamp on in order to attain the desired illumination is not always good because the system lacks information about the new occurring situations (they are not very similar to the known ones). It can be seen that the decision-making system can be sufficiently accurate in making decisions as the second situation occurs because it is similar to the situation used in the retraining process of the system. The halogen lamp is switched off correctly in most cases as well.

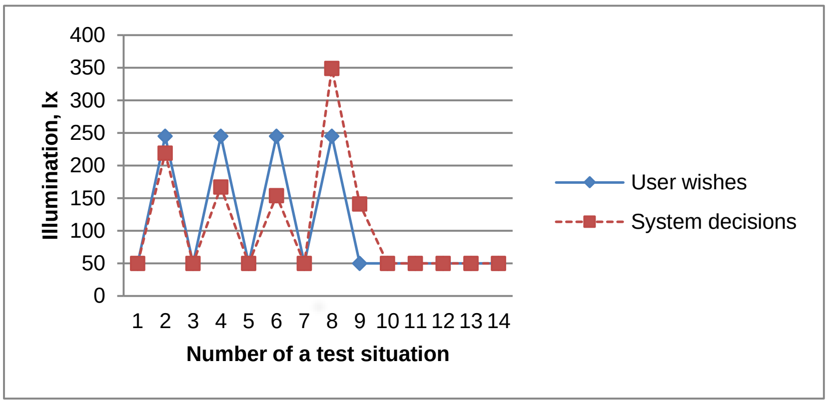

Figure 6. The halogen lamp control results (using the data from the first four known situations)

In a situation when the user goes to the table on the right and turns the fluorescent lamp on, additional data about the occurred situation is made available and so the decision-making system can be retrained. But let us assume that later he goes back to the middle of the room and switches it off. In such case, an additional fifth, sixth and seventh situation data shown in Table 4 was available for the halogen lamp when action *DN* was taken (only fluorescent lamp was switched on and off).

Additional fifth, sixth and seventh situation data described in Table 5 was available not only with actions *DN*, but also with *D* and *I* actions for the fluorescent lamp as well. The decision-making system had to retrain itself taking into account the newly available situations.

In the case of the halogen lamp, the procedure is quite simple as it requires only for additional constrains to be added for each previously formed LPP using new additional situation data described in Table 4. For the fluorescent lamp, three LPPs should be formed using all available situation data that is described in Table 5, similarly as it has been done previously for the halogen lamp.

The control results from the same 14 testing situations (Fig. 5) are depicted in Fig. 7. It should be taken into account that each light source influences both illumination sensor measures. So these two a) and b) parts as seen in Fig. 7 should be interpreted in conjunction with each other. For example, in a test situation seven, where the fluorescent lamp is turned on and the halogen lamp is turned off, the fluorescent lamp still has influence on the measures of the first illumination sensor.

Table 4. Additional data from the occurred situations for the halogen lamp

Situation number	Presence	*X*	*Y*	L_1	L_2	Action
1	1	163	243	50	50	*I*
2	1	163	243	245	74	*DN*
3	1	675	263	245	74	*D*
4	1	700	251	50	50	*DN*
5	1	1136	259	50	50	*DN*
6	1	1136	259	91	368	*DN*
7	1	677	255	91	368	*DN*

Table 5. Additional data from the situations that occurred near the fluorescent lamp

Situation number	Presence	*X*	*Y*	L_1	L_2	Action
1	1	163	243	50	50	*DN*
2	1	163	243	245	74	*DN*
3	1	675	263	245	74	*DN*
4	1	700	251	50	50	*DN*
5	1	1136	259	50	50	*I*
6	1	1136	259	91	368	*DN*
7	1	677	255	91	368	*D*

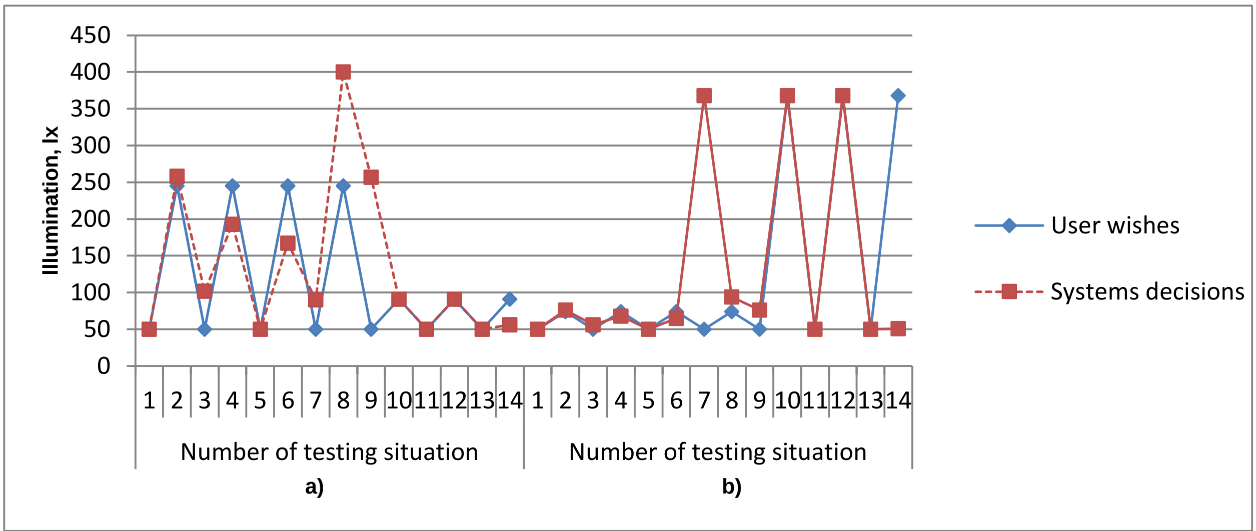

Figure 7. The first (a) and the second (b) illumination sensor values (with additional known situation data)

The test results obtained after the retraining of the decision-making system with additional known data show that the system is able to control both light sources reasonably. In some cases, the decision-making system makes exactly the same decisions as the user wishes. However, the test situation seven can be considered as an opposite example, where the system decides to turn the fluorescent lamp on, or the test situation fourteen, where the system decides not to turn the fluorescent lamp on. To be clear, this happens because there is no sufficient data about these situations and the decision-making system makes decisions according to limited data available.

4.2. Experimental results when two users are interacting in the environment

Now let us consider a more complex scenario: two users interacting in the environment. The first user has the same preferences as in the previous example, while the second user has different preferences: 1) all light sources should be turned on at maximum when he is in the middle of the room; 2) the halogen lamp intensity should be adjusted by selecting the sixtieth actuator position when he is near the table on the left; 3) all lamps should be switched off when he is near the table on the right. When they are both at the same place, their preferences are: 1) the halogen lamp should be turned off and the fluorescent lamp should be turned on when they are in the middle of the room; 2) the halogen lamp intensity should be adjusted by selecting the fourteenth actuator position when they are both near the table on the left; 3) the fluorescent lamp should be switched off when they are near the table on the right.

The experimental results for the control of the lamps when the first user is interacting in the environment are depicted in Fig. 8. The same test situations (Fig. 5) are used as previously. In order to achieve the desired accuracy, the first user has given his feedback to the control system in three situations near the table on the left, in two situations near the table on the right, and in two situations when he was in the middle of the room. The second user also gave feedback to the control system the same number of times. The experimental results of the control of the lamps are depicted in Fig. 9 where the second user is interacting in the environment respectively. Fig. 10 depicts the experimental results when both users are interacting in the environment. The number of learning situations in this case was also the same.

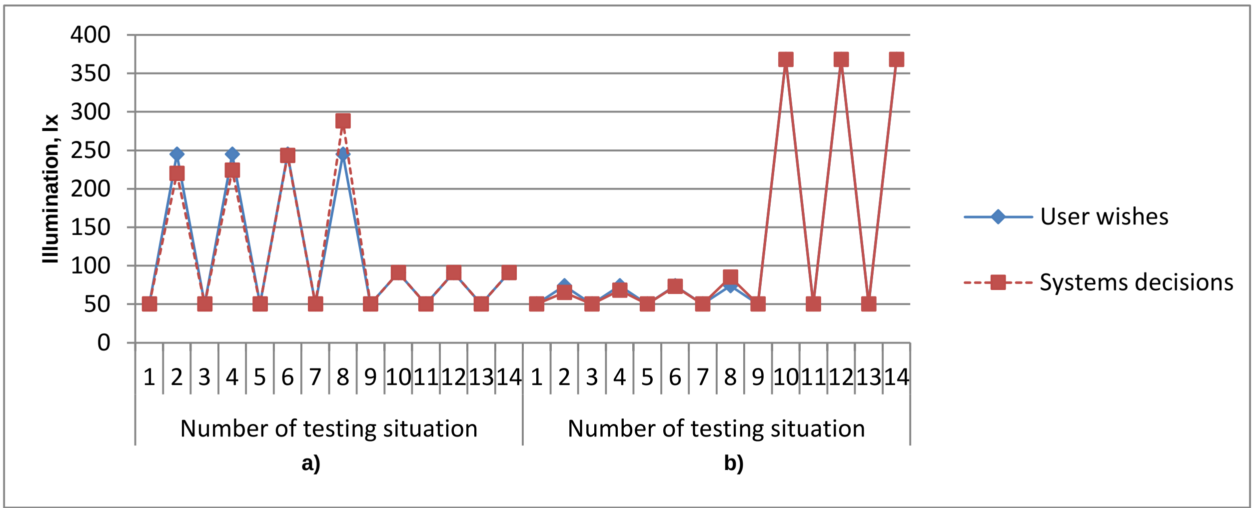

Figure 8. The first (a) and second (b) illumination sensor values (wishes of the first user)

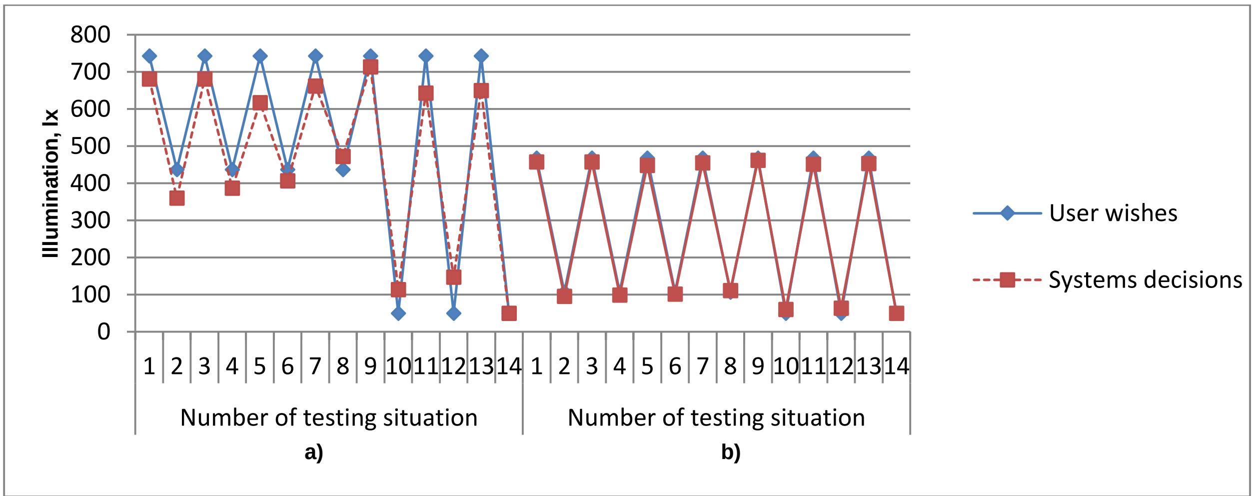

Figure 9. The first (a) and second (b) illumination sensor values (wishes of the second user)

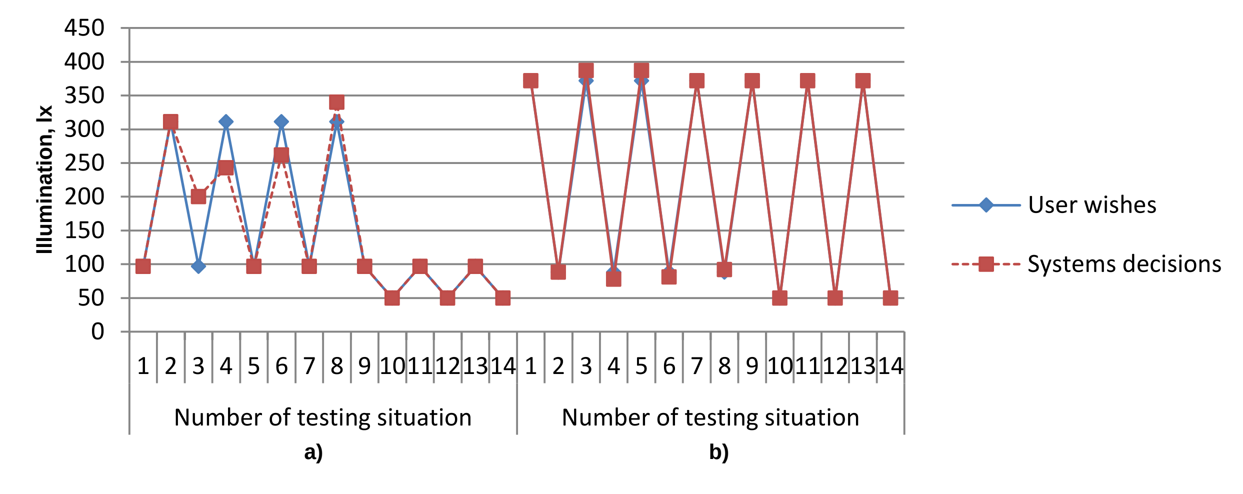

Figure 10. The first (a) and second (b) illumination sensor values (wishes of both users)

These experiments show that the decision-making system is capable of making reasonable decisions in more complex scenarios as well. As the results show, the system achieves different performance levels when using different types of actuators for the control of light sources. According to the experimental results, once the decision-making system learns user's preferences, it is easier to decide whether to switch the fluorescent lamp on or off than to set the correct illumination level for the halogen lamp. In this case, the fluorescent lamp is controlled correctly in all test situations, while the halogen lamp illumination is not always set accurately, though with an acceptable error margin. On the other hand, to achieve reasonable performance levels, the users can give feedback to the control system once, when a particular situation arises. There is no need to teach the control system when very similar situations arise and the same control actions are taken because, as our experiments show, the control system will not achieve a significantly better performance.

5. Concluding remarks

This paper presents an intelligent decision-making procedure based on the augmenting linear programming approach. The viability of the idea is demonstrated by application of the procedure to the intelligent light control system model. The presented self-training procedure enables to retrain the decision-making system every time new data about the newly occurred situation is available, i.e. whenever the user gives feedback to the system. Thus, the self-training procedure ensures the continuous learning of the user habits.

In our simulation, the decision-making procedure was based on the analysis of the user's feedback to the system considering dual aspects/characteristics: the target illumination level and the preferred lighting devices. The latter issue makes the proposed solution distinguishable from the reviewed research in this field, as other linear programming methods that deal with power consumption minimization take into account only the illumination levels as desired by the user.

Testing results show that the presented intelligent decision-making procedure is capable of making reasonable decisions according to defined user preferences when similar situations occur, even in situations when data is very limited. In this case, the system requires no initial data for learning and learning is achieved by using the presented self-training procedure.

It must be emphasized that situations in which the user suddenly changes his habits and preferences require further investigations. This task requires for certain modifications to be made in some previously analysed situations. It is a difficult task because it must be determined which situations should be modified and how.

Acknowledgments

This research was supported by the EU Structural Funds project VP1-3.1-ŠMM-10-V-02-020 "Research on Smart Home Environment and Development of Intelligent Technologies (BIATech)".

References

[1] **M. Akita, M. Miki, T. Hiroyasu, M. Mitsunori, M. Yoshimi.** Optimization of the Height of Height-Adjustable Luminaire for Intelligent Lighting System. *Artificial Intelligence and Soft Computing, Lecture Notes in Computer Science*, 2010, Vol. 6114, 355–362.

[2] **M. R. Alam, M. B. I. Reaz, F. H. Hashim, M. A. M. Ali.** Stochastic analysis of smart home user activities. In: *International Joint Conference on Neural Networks*, 2011, pp. 21–23.

[3] **F. Amirjavid, A. Bouzouane, B. Bouchard, K. Bouchard.** Recognition of fuzzy contexts from

temporal data under uncertainty case study: Activity recognition in smart homes. In: *13th International Conference on Information Reuse and Integration* 2012, pp. 231–238.

[4] **D. J. Caine, B. J. Parker.** Linear programming comes of age: a decision-support tool for every manager. *Management Decision*, 1996, Vol. 34, 46–53.

[5] **P. Chen, K. Hong, N. Naikal.** A low-bandwidth camera sensor platform with applications in smart camera networks. *ACM Transactions on Sensor Networks*, 2013, Vol. 9, No. 2, 21–23.

[6] **M. S. Choi, L. Xiang, S. J. Lee, T. W. Kim.** An Innovative Application Method of Monthly Load Forecasting for Smart IEDs. *Journal of electrical engineering & technology*, 2013, Vol. 8, No.5, 984–990.

[7] **D. De, W. Z. Song, S. Tang, D. Cook.** ActSee: Activity-Aware Radio Duty-Cycling for Sensor Networks in Smart Environments. *Journal of electrical engineering & technology*, 2013, Vol. 8, No. 5, 984–990.

[8] **E. Denny, A. Keane.** A Smart Integrated Network for an Offshore Island. *Proceedings of the IEEE*, 2013, Vol. 101, No. 4.

[9] **M. A. Dragan, I. Mocanu.** Human Activity Recognition in Smart Environments. In: *19th International Conference on Control Systems and Computer Science*, 2013, pp. 495–502.

[10] **Y. R. Fan, G. H. Huang, Y. P. Li.** Robust interval linear programming for environmental decision making under uncertainty. *Engineering Optimization*, 2012, Vol. 44, No. 11, 1321–1336.

[11] **Y. R. Fan, G. H. Huang, A. L. Yang.** Generalized fuzzy linear programming for decision making under uncertainty: Feasibility of fuzzy solutions and solving approach. *An International Journal of Information Sciences*, 2013, Vol. 241, 12–27.

[12] **R. Gircys, E. Kazanavicius, A. Vrubliauskas, A. Liutkevicius.** Movement artefact resistant photoplethysmographic probe. *Electronics and Electrical Engineering*, 2014, Vol. 20, No. 3, 73–76.

[13] **M. Guarracino, R. Jasinevicius, R. Krusinskiene, V. Petrauskas.** Fuzzy Hyperinference-Based Pattern Recognition. Towards Advanced Data Analysis by Combining Soft Computing and Statistics. *Editors C. Borgelt, M. Á. Gil, J.M.C. Sousa, M. Verleysen. Springer*, 2013, 223–240.

[14] **Y. He, M. S. Yazdi.** Differentiated Service for Smart Grid Neighbourhood Area Networks via Optimal Resource Allocation. *International Journal of Sensors and Sensor Networks*, 2013, Vol. 1, No. 5, 55–60.

[15] **S. Hua, K. Yoshihiro, M. Hiroyuki, A. Tomonori.** A Stochastic Approach for Creating Context-aware Services based on Context Histories in Smart Home. In: *3rd International Conference on Pervasive Computing, Exploiting Context Histories in Smart Environments*, 2005, 37–41.

[16] **R. Jasinevicius.** Parallel Space-Time Computing Structures. *Mokslas*, 1988.

[17] **R. Jasinevicius, E. Kazanavicius, V. Petrauskas, A. Sanayei.** Intellectualized home environment as a complex system. *ISCS 2014: Interdisciplinary Symposium on Complex Systems. Emergence. Complexity and Computation 14. Springer*, 2014, 87–97.

[18] **R. Jasinevicius, V. Petrauskas.** Fuzzy expert maps: the new approach. WCCI 2008 Proceedings: 2008 IEEE World Congress on Computational Intelligence, 2008, Hong Kong: 2008 IEEE International Conference on Fuzzy Systems. In: 2008 IEEE International Joint Conference on Neural Networks. 2008 IEEE Congress on Evolutionary Computation. Piscataway: IEEE, 2008, 1511–1517.

[19] **F. Javed, N. Arshad, F. Wallin, I. Vassileva, E. Dahlquist.** An adaptive optimization model for power conservation in the smart grid. In: *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, 2010, pp. 1563–1570.

[20] **F. Kaku, M. Miki, T. Hiroyasu, M. Yoshimi, T. Nishida, N. Kida.** Construction of Intelligent Lighting System Providing Desired Illuminance Distributions in Actual Office Environment. *Artifical Intelligence and Soft Computing, Lecture Notes in Computer Science*, 2010, Vol. 6114, 451–460.

[21] **A. Konar.** Computational Intelligence (Principles, Techniques and Applications). *Springer*, 2005.

[22] **J. M. L. Lesaca, E. Z. Guerrero, F. E. Ubis, J. A. Bernardo.** Decision-making Tools and Memetic Algorithms in Management and Linear Programming Problems. *Review of international comparative management*, 2011, Vol. 12, No. 2, 350–365.

[23] **D. F. Li, S. P. Wan.** Fuzzy linear programming approach to multiattribute decision making with multiple types of attribute values and incomplete weight information. *Applied Soft Computing*, 2013, Vol. 11, 4333–4348.

[24] **H. Medjahed, D. Istrate, J. Boudy, J. L. Baldinger, B. Dorizzi.** A pervasive multi-sensor data fusion for smart home healthcare monitoring. In: *IEEE International Conference on Fuzzy Systems*, 2011, pp. 1466–1473.

[25] **M. Miki, T. Hiroyasu, K. Imazato.** Proposal for an intelligent lighting system, and verification of control method effectiveness. In: *IEEE Conference on Cybernetics and Intelligent Systems*, 2004, Vol. 1, pp. 520–525.

[26] **A. Mikuckas, I. Mikuckiene, A. Venckauskas, E. Kazanavicius, R. Lukas, I. Plauska.** Emotion Recognition in Human Computer Interaction Systems. *Electronics and Electrical Engineering*, 2014, Vol. 20, No. 10, 51–56.

[27] **Y. MingYan Yuan.** A Multi-attribute Reverse Auction Decision Making Model Based on Linear Programming. *Systems Engineering Procedia*, 2012, Vol. 4, 372–378.

[28] **A. T. Naser, A. Husam.** A linear programming model to optimize the decision-making to managing cogeneration system. *Clean Technologies and Environmental Policy*, 2007, Vol. 9, No. 3, 235–240.

[29] **M. S. Pan, L. W. Yeh, Y. A. Chen, Y. H. Lin, Y. C. Tseng.** A WSN-Based Intelligent Light Control System Considering User Activities and Profiles. *Sensors Journal*, IEEE, 2008, Vol. 8, No. 10, 1710–1721.

[30] **H. Park, J. Burke, M. B. Srivastava.** Design and Implementation of a Wireless Sensor Network for Intelligent Light Control. *Information Processing in Sensor Networks*, 2007, 370–379.

[31] **T. V. Prabhakar, S. N. Akshay Uttama Nambi SN.** A Distributed Smart Application for Solar Powered WSNs. *Lecture Notes in Computer Science*, 2012, Vol. 7290, 291–303.

[32] **C. Rong, T. Yu.** A Two-stage Method for Solving Multi-resident Activity Recognition in Smart Environments. *Entropy*, 2014, Vol. 16, No. 4, 2184–2203.

[33] **N. Sahebjamnia, S. A. Torabi.** A fuzzy stochastic programming approach to solve the capacitated lot size problem under uncertainty. In: *International Conference on Fuzzy Systems*, 2011, pp. 2327–2334.

[34] **G. Salcius, E. Povilaitis, A. Tacilauskas.** Indoor Positioning Using Ultrasound and Radio Combination. In: *The Third International Conference on Informatics Engineering and Information Science ICIEIS*, 2014, pp. 110–120.

[35] **V. Singhve, A. Krause, C. Guestrin, J. H. Garrett, H. S. Matthews.** Intelligent light control using sensor networks. In: *Proceedings of the 3rd international conference on Embedded networked sensor systems*, 2005, pp. 218–229.

[36] **S. Tanaka, M. Miki, M. Hiroyasu Tand Yoshikata.** An evolutional optimization algorithm to provide individual illuminance in workplaces. In: *IEEE International Conference on Systems, Man and Cybernetics*, 2009, 941–947.

[37] **E. Tanfani, A. Testi.** Planning surgical activities: a societal point of view. In: press on *Proceedings of the 33rd International Conference on Operational Research Applied to Health Services*, 2008.

[38] **C. Tomishima, M. Miki, M. Ashibe, T. Hiroyasu, M. Yoshimi.** Distributed Control of Illuminance and Color Temperature in Intelligent Lighting System. *Artifical Intelligence and Soft Computing, Lecture Notes in Computer Science*, 2010, Vol. 6114, 411–419.

[39] **A. M. Vainio, M. Valtonen, J. Vanhala.** Proactive Fuzzy Control and Adaptation Methods for Smart Homes. *Intelligent Systems*, 2008, Vol. 23, No. 2, 42–49.

[40] **Y. J. Wen, J. Granderson, A. M. Agogino.** Towards Embedded Wireless-Networked Intelligent Daylighting Systems for Commercial Buildings. In: *IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing*, 2006, Vol. 1, 326–331.

[41] **Y. Xu, G. H. Huang, X. S. Qin, Y. Huang**. SRFILP: A Stochastic Robust Fuzzy Interval Linear Programming Model for Municipal Solid Waste Management under Uncertainty. *Journal of Environmental Informatics*, 2009, Vol. 14, No. 2, 74–82.

[42] **L. W. Yeh, C. Y. Lu, C. W. Kou, Y. C. Tseng, C. W. Yi.** Autonomous Light Control by Wireless Sensor and Actuator Networks. *IEEE Sensors Journal*, 2010, Vol. 10, No.6, 1029–1041.

Received March 2015.