Office Building’s Occupancy Prediction Using Extreme Learning Machine Model with Different Optimization Algorithms

Violeta MOTUZIENĖ1*, Jonas BIELSKUS2, Vilūnė LAPINSKIENĖ3, Genrika RYNKUN4

1–4 Vilnius Gediminas Technical University, Saulėtekio av. 11, Vilnius, Lithuania

Abstract – Increasing energy efficiency requirements lead to lower energy consumption in buildings, but at the same time occupants’ influence on the energy balance of the building during the use phase becomes more crucial. The randomness of the building’s occupancy often leads to the mismatch of the predicted and measured energy demand, also called Energy Performance Gap. Therefore, prediction of occupancy is important both in the design and use phases of the building. The goal of the study is to apply Extreme Learning Machine (ELM) models with different optimisation algorithms – Genetic (GA–ELM) and Simulated Annealing (SA–ELM) for occupancy prediction in an office building based on measured CO₂ concentrations. Both models show similar and high accuracy of prediction: $R^2$ – 0.73–0.74 and RMSE – 1.8–1.9 for the whole measured period. Influence of population size, number of neurons, and number of iterations on results accuracy was also analysed and recommendations are given. It was concluded that both methods are suitable for occupancy prediction, but because of different simulation times, SA–ELM is recommended for the Building Management Systems (BMS), where higher speed is required.

Keywords – CO₂ (carbon dioxide); Genetic Algorithm (GA); office; Simulated Annealing (SA)

ELM Nomenclature

| Symbol | Description |
|--------|-------------|
| $g(x)$ | Activation function |
| $L$ | Number of hidden layer nodes |
| $N$ | Number neurons |
| $T$ | One of the least-squares solutions of a general linear system |
| $H$ | Hidden layer output matrix |
| $β$ | Output weights |
| $x_i$, $t_i$ | Input neurons |
| $w_i$ | Input weight vector |
| $b_i$ | Bias weight vector |
| $R$ | Real numbers |
| $H^+$ | Moore-Penrose generalized inverse of matrix |

* Corresponding author.
E-mail address: violeta.motuziene@vilniustech.lt
1. **INTRODUCTION**

Increasing energy efficiency requirements lead to the lower energy consumption in buildings, but at the same time it was noticed, that buildings in reality do not perform as well as it was predicted. The mismatch of the expected and measured energy demand is called Energy Performance Gap (EPG) [1], [2]. RIBA CIBSE platform Carbon Buzz and other sources report, that many buildings in the European Union consume 1.5 to 3.0 times more energy than predicted during the design stage [3], [4]. Geraldi and Ghisi [5] identified EPG as one of the four key building energy performance-related research topics. The causes of the EPG can be found in different life cycle stages of the building – design, construction and operation stages and many scientists emphasized the importance occupants’ behaviour – both passive and active [4], [6]–[9]. According to Peper & Feist [10], occupants behaviour influence on actual energy consumption is ±50 %, therefore it should not be underestimated. Definition, estimation and intervention of occupancy behaviour become essential research challenges during building energy policy making, implementation and evaluation processes [11].

Actual occupant’s presence (passive behaviour) is rarely taken into account in buildings operation, thus causing unnecessary energy consumption. Analysing office buildings it is noticed that rooms are not always fully occupied during daytimes and some rooms are routinely unoccupied [12]. Some monitoring studies showed that average occupancy in offices is just around 60 % [13], [14] and for multi-person offices it is higher and can reach
90 % [14]. Meanwhile Bielskus et al. [15] compared measured occupancy with the ones that are used in prediction and found that actual occupancy in an open-office is much lower compared to the DesignBuilder’s default values and to the ones provided by EN 16798-1 [16]. These differences are accordingly 3.5 and 2.7 times [15]. It can be summarized that realistic assumptions concerning the occupancy of the building may significantly decrease the EPG as well as contribute to increase of post-occupancy energy efficiency of the building through predictive-control.

Comprehensive and reliable occupancy prediction models are still under development. Occupancy is a periodic time series, and different time-independent and non-linear occupancy prediction models are proposed [17]. These models can be categorized as deterministic, stochastic and machine learning. Authors in previously published paper [15] discussed different prediction models; also they are compared in a review paper [18].

As it was already concluded by the authors, that Extreme Learning Machine (ELM) method is promising and reliable in occupancy prediction [15]. The goal of the study is to apply two ELM models with different optimisation algorithms – Genetic (GA–ELM) and Simulated Annealing (SA–ELM) for occupancy prediction in an open-office building based on measured CO₂ concentrations.

2. METHODS AND METHODOLOGY

The study uses data from open-office measurements in Vilnius. As it was already found by the authors in previously published study [15], best correlations in predicting occupancy behaviour are gained based on measured CO₂ concentrations. Therefore, presented models are using as input data just PIR (occupancy sensors) and CO₂ data.

For the prediction of occupancy, two ELM (Extreme Learning Machine) models with different optimisation algorithms - Genetic (GA) and Simulated Annealing (SA) are created and simulated in MATLAB. For both models (GA–ELM and SA–ELM), to define the influence of population size, number of neurons and number of maximum iterations on simulation accuracy, 900 simulations were performed. For each value of the parameter (number of neurons, etc.), simulations were repeated 5 times to assess average reliability.

The reliability of results and methods is compared in terms of $R^2$, RMSE and calculation speed. General methodological scheme is presented in Fig. 1.

![Fig. 1. General methodological scheme.](image)

2.1. Extreme learning machine method

ELM is a learning algorithm linked with Artificial Neural Networks (ANNs) and it was first
introduced by Huang in 2004 [19]. The algorithm is simple, runs extremely fast and has a good generalization performance. It attempts to reach both the smallest training error and the smallest norm of weights. Because of the mentioned advantages, ELM has gained increasing attention and wide application in various fields [20]–[25], such as medicine, marine environment, mechanics of materials, etc. Different improvements of the algorithm have been proposed, thus enabling its applications also for big data analysis [26]. Hence, the potential of the ELM until now is not sufficiently exploited in the field of the building’s occupancy prediction, where not just prediction accuracy is important, but also speed of the prediction as it is related to the control of the building services systems.

The most popular architecture of the ANNs is Single-layer Feedforward Networks (SLFNNs), which is also used in further presented ELM models. Network is constructed for $N$ arbitrary distinct samples $(x_i, t_i)$, with the $L$ hidden nodes and sigmoidal activation function $g(x)$ (Fig. 2). It has 3 layers of neurons (information-processing units): input layer, hidden layer and output layer. Using ELM, parameters between the input and hidden layer are generated randomly (random feature mapping stage) and output weights are calculated analytically (linear parameters solving stage). For better solution search and faster learning, least squares (LS) method [21] is applied and outputs are simulated according to Eq. (1).

![Fig. 2. Structure of the SLFNN](image)

The concepts of the ELM procedure with the related equations is presented in Fig. 3 together with the equations used in the solution.

For a set number of $L$ hidden nodes, for randomly generated input and bias weights and activation function $g_i(x)$, $L$ samples are approximated according to Eq. (3) (Fig. 3). For a given $N$, these equations can be written as $H\beta = T$ and expressed as Eq. (4), where $H$ is a hidden layer output matrix. Activation and transfer functions are infinitely differentiated and the number of samples is significantly larger than the number of hidden neurons $L < N$. 

$$g(x) = \frac{1}{1+\exp(-x)} \quad (1)$$

$$o_j = g_i(x_j) = \sum_{i=1}^{L} \beta_i g_i(x_j + b_i) \quad (2)$$
The output weight (\(\beta\)) matrix is solved by minimizing the approximation error in the squared error sense \(\left(\min \| H\beta - T \|^2 \rightarrow 0\right)\) and the smallest norm least-squares solution is reached by the Moore-Penrose generalization of the inverse matrix – Eq. (5).

As traditional ELM methods fail to train big data efficiently, to solve this problem different optimization algorithms are employed. Two algorithms of different nature are applied in the study and presented below.

2.2. Genetic algorithm

Genetic algorithms are randomized global search algorithms that imitate the natural selection and natural genetics. GA operates on a population of potential solutions and searches a result equal to or close to the answer of a given problem. New generation of solutions is generated from solutions in previous generation by performing crossover (recombination), mutation and repeating the process. This process leads to the evolution of populations of solutions that are better than those that they were created from.

Process of optimisation using Genetic algorithm is presented in Fig. 4. and each of the steps is described below.

---

**Fig. 3** Steps of the ELM learning procedure.

**Fig. 4.** Process of optimisation based on GA.
Population initialization. In this step the population size $M$ is set and the chromosomes of the randomly generated population are determined as the initial $Y(0)$.

Fitness function calculation. The suitability of each chromosome in the population $Y(t)$ is determined at this step. Population is assumed as function $f(y)$ (Fig. 4, Eq. (6)) which depends on population size $(y \in M)$. To optimise ANN fitness, function $fit(y)$ is expressed as Eq. (7).

Selection. At this step, using the selection operator, $L$ pairs of parents from $Y(t)$ are selected. When $L \geq M$, calculation continues. Otherwise, if $L < M$ iterations are terminated (calculation stops).

Crossover. It is one of the three basic operators of the algorithm. Crossover (also called recombination) is the process when genetic information of two randomly selected chromosome parents is combined to create an offspring (child). The crossover is performed with a high probability $P_c$, which is calculated according to Eq. (8). If this probability of one of parent chromosome is less than $P_c$, then one or more points are developed to perform the crossover and get new child chromosomes and form $L$ intermediate individuals.

The result of crossover is a new population $Y(t+1)$.

Mutation. This step is required to maintain diversity within the population and prevent premature convergence. To avoid primitive random search, mutation probability must be set to low.

Terminate. When the evolutionary algebra $N$ is reached, the evolution is stopped and the chromosome with the highest suitability in $N$ generations is selected as an optimal solution.

GA-ELM algorithm was proposed and presented in details in [27].

2.3. Simulated Annealing algorithm

Simulated Annealing (SA) is a low complexity, powerful stochastic algorithm, which is used to solve various optimisation problems [28]. It is based on the search of the global minimum of the specific objective function attempting to avoid local minimums by performing the random search (movements in the searched neighbourhood). In other words, the algorithm is based on random traveling upwards instead of choosing a straight path. If a randomly selected move improves the solution, it is accepted. Otherwise, SA makes the move anyway with probability less than 1. The probability decreases exponentially and it depends on the motion.

The overall flow chart of the SA, which has two iterative loops, is presented in the Fig. 5. Before starting the SA – initial temperature ($t_0$), which is the main control parameter and is reduced during the optimisation process, must be set [29]. Annealing starts when an initial solution ($x_0$) is generated. It is considered as the best present solution. The next step is creation of the neighbourhood structure. At this stage two positions $i$ and $j$ are selected and reverse, insert, and swap operators are used to produce neighbouring solutions. The list of input data is created and used to calculate the probability $p$ of a candidate’s decision. If the decision function $f(x)$ is worse than the candidate decision function $f(y)$ ($f(y) > f(x)$), then the probability of acceptance can be calculated according to Eq. (9). In each iteration, the maximum data are used from the list. Each time a bad candidate solution is met, a random number $r$ is generated. If $r < p$, then the bad candidate solution will be accepted [30]. Therefore, for each pair of $d_i$ and $p_i$, $r_i$ is selected randomly and then $r_i < p_i$. For the new data, $t_i$ is calculated according to the Eq. (10). When the condition is met, the inner loop is stopped. The average of all data $t_{av}$ is used to update the data list, i.e. $t_{max}$ is removed from the list and the average of the data is loaded because $t_{av} < t_{max}$, and every time $t_i < t_{max}$. This way, the data shrinks when a search is performed. The outer loop is stopped when the lowest limit of the temperature is reached or predefined number of iterations is reached.

Simulating annealing flow chart shown in Fig. 5.
In this research, one open-office space with 32 work stations was analysed. Maximum occupancy during the measured period did not exceed 16. Observed occupancy and CO$_2$ concentrations data were used to train neural networks and to forecast occupancy. Validation is performed based on real measured data.

Already in the initial stage, SA and GA determine the structure of the neural network by selecting or evolving data, respectively. SA or GA is applied in the training stage. Algorithms require to enter measured values of CO$_2$ and occupancy. Occupancy is measured with PIR sensors mounted under the tables. They are called input parameters $x$. After SA and GA optimisations from input data $x$ resulting values are called $y$.

3. RESULTS

Simulation with MATLAB ELM models enabled to predict occupancy profiles based on CO$_2$ concentration data from 8 weeks measurement period. Measurements were recorded every 5 minutes and resulted in 16,279 points of unfiltered data. In the model 70% of these data were used for training, 10% – for testing and 20% for validation of the results.

Assumptions for the initial simulations with SA-ELM and GA-ELM models:
- Number of hidden neurons – 20;
- Population size (PopSize) – 100;
- Maximums number of iterations (MaxIter) – 100.

Figures 6 and 7 present simulation results for both models, where desired output is measured occupancy and estimated output – predicted occupancy. The reliability of results was estimated using $R^2$ and RMSE. For the whole analysed period reliability is very similar: for SA-ELM – $R^2 = 0.74$ and RMSE = 1.802 with the calculation time up to few seconds, accordingly for GA-ELM – $R^2 = 0.73$ and RMSE = 1.903 with a calculation time approx. 1 min.
Fig. 6. SA-ELM model occupancy prediction results based on CO₂.

Fig. 7. GA-ELM model occupancy prediction results based on CO₂.

From the Fig. 8 it is seen that SA-ELM gives slightly better accuracy. It is also important to note, that accuracy of the results strongly depends on the measurement period – the longer the period, the better predictability and this proves that short term measurements results in low reliability, as data set is not sufficient. Reliability or prediction strongly increases starting already from the 4th week. Authors in previous research gained same results concerning the measurement period [13].
Additional simulations are required to make the comparison of models more accurate. For this purpose, 900 simulations were performed varying the number of neurons, population size and maximum number of iterations. RMSE is expressed as average of 5 simulations for each value.

For both optimisation models 200 simulations (40 values × 5 simulations for each) were performed seeking to assess the influence of number of neurons on the accuracy of results (Fig. 9). It is obvious that higher number of neurons increases the accuracy of prediction and significant influence is noticed up to 25 neurons. Higher number of neurons seems to be
unreasonable. RMSE values from all the 200 simulations variate as follows: for SA-ELM from 1.773 to 1.994 and for GA-ELM – from 1.758 to 1.934.

In ELM methodology optimisation is applied in training stage and results are influenced by population size (data set used for training). To estimate that influence, for each optimisation model 125 additional simulations changing the population size were performed. The results are presented in Fig. 10.

![Fig. 10. RMSE depending on population size (PopSize).](image)

It can be seen (Fig. 10), that for the GA-ELM 20 is reasonable number of population size, as higher population makes almost no influence on accuracy and increases the speed (decreases the duration) of simulation. Meanwhile for SA-ELM model population size makes no influence on accuracy of results as well as on calculation duration which takes up to few seconds, therefore is not shown on the graph. RMSE values from all the 125 simulations variate as follows: for SA-ELM from 1.788 to 1.950 and for GA-ELM – from 1.776 to 3.622.

Figure 11 presents comparison of the accuracy of two models depending on maximum number of iterations and shows how it influences calculation duration. 125 simulations were performed for each of the alternative optimisation model.

![Fig. 11. RMSE depending on Max Iterations.](image)
It is seen (Fig. 11) that for GA-ELM higher number of iterations results in increase of the calculation time, which may take more than 1 min. For SA-ELM simulation time fluctuates from 1 to 5 s. From the results it is obvious that for both models 10 iterations can be assumed as reasonable number. RMSE values from all the 125 simulations variate as follows: for SA-ELM from 1.777 to 3.602 and for GA-ELM – from 1.766 to 3.602.

All the simulations show that both optimisation algorithms show very similar accuracies and main difference is simulation speed, which is the main advantage of the SA-ELM method.

4. CONCLUSION

In the study, using MATLAB software two Extreme Learning Machine (ELM) models with different optimisation algorithms – Genetic (GA-ELM) and Simulated Annealing (SA-ELM) were created for occupancy prediction in an office building based on measured CO₂ concentrations. Both models showed similar and high accuracies of prediction for the whole measured period: $R^2$ – 0.73–0.74 and RMSE – 1.8–1.9 (at PopSize – 100, MaxIter – 100 and neurons number 20). Simulation duration for SA-ELM is up to 5 s, and GA-ELM up to 1 min.

Additionally 900 simulations were performed to check sensitivity of the accuracy of results on: number of neurons, population size and number of iterations. For both models, all of the analysed parameters may have impact on prediction accuracy. Meanwhile calculation speed is influenced significantly just in GA-ELM model. It was concluded that sufficient prediction accuracy is reached for both models when: number of neurons is no less than 25, population size no less than 20 and number of iterations no less than 10 resulting simulation duration up to 5 s.

It can be concluded that both ELM optimisation algorithms are suitable for occupancy prediction, but because of different simulation time, SA-ELM is recommended for the BMS systems, where higher speed is required. Occupancy prediction model integration into BMS system would enable control of HVAC and other building service systems according to realistic demand thus increasing energy efficiency of the building and would help to decrease the EPG.

ACKNOWLEDGEMENT

This research was funded by a grant (No. S-MIP-20-62) from the Research Council of Lithuania (LMTLT).

REFERENCES

[1] Yan D., Hong T. Definition and Simulation of Occupant Behavior in Buildings. Annex 66 Final (2018) Report. [Online]. [Accessed 10.03.2021]. Available: https://annex66.org/?q=Publication%0Ahttps://www.iea-ebc.org/projects/project?AnnexID=66
[2] Delzendeh E., Wu S., Lee A., Zhou Y. The impact of occupants’ behaviours on building energy analysis: A research review. Renewable and Sustainable Energy Reviews 2017:80:1061–1071. https://doi.org/10.1016/j.rser.2017.05.264
[3] RIBA CIBSE platform CarbonBuzz [Online]. [Accessed 12.06.2020]. Available: https://www.carbonbuzz.org/index.jsp
[4] Zou P. X. W., Alam M. Closing the building energy performance gap through component level analysis and stakeholder collaborations. Energy and Buildings 2020:224:110276. https://doi.org/10.1016/j.enbuild.2020.110276
[5] Geraldi M. S., Ghisi E. Building-level and stock-level in contrast: A literature review of the energy performance of buildings during the operational stage. Energy and Buildings 2020:211:109810. https://doi.org/10.1016/j.enbuild.2020.109810
[6] Simanic B., Nordquist B., Bagge H. & Johansson D. Influence of User-Related Parameters on Calculated Energy Use in Low-Energy School Buildings. Energies 2020:13(11):2985. https://doi.org/10.3390/en13112985
[7] Delzendeh E., Wu S., Lee A., & Zhou Y. The impact of occupants’ behaviours on building energy analysis: A research review. Renewable and Sustainable Energy Reviews 2017:80:1061–1071. https://doi.org/10.1016/j.rser.2017.05.264

[8] Ding Y., Chen W., Wei S., Yang F. An occupancy prediction model for campus buildings based on the diversity of occupancy patterns. Sustainable Cities and Society 2021:64:102533. https://doi.org/10.1016/j.scs.2020.102533

[9] Liang J., QuY., Hu M. Mind the energy performance gap: Evidence from green commercial buildings. Resources, Conservation and Recycling 2019:141:364–377 https://doi.org/10.1016/j.resconrec.2018.10.021

[10] Peper S., Feist W., Monitoring und Bilanzerzung: Ganz ohne Performance GAP. In Proceedings of the CESB Central European Symposium on Building Physics and BauSIM 2016, Fraunhofer IRB Verlag: Dresden, Germany.

[11] Hu S., Yan D., Azar E., Guo F. A systematic review of occupant behavior in building energy policy. Building and Environment 2020:175:106807. https://doi.org/10.1016/j.buildenv.2020.106807

[12] Peng Y., Rysanek A., Nagy Z., Schlüter A. Using machine learning techniques for occupancy-prediction-based cooling control in office buildings. Applied Energy 2018:211:1343–1358. https://doi.org/10.1016/j.apenergy.2017.12.002

[13] Mahdavi A., Mohammadi A., Kabir E., Lambeva L. Occupants’ operation of lighting and shading systems in office buildings. Journal of Building Performance Simulation 2008:1:57–65. https://doi.org/10.1080/1940149080196502

[14] Peng Y., Rysanek A., Nagy Z., Schlüter A. Occupancy learning-based demand-driven cooling control for office spaces. Building and Environment 2017:122:145–160. https://doi.org/10.1016/j.buildenv.2017.06.010

[15] Bielskus J., Motuziene V., Vilutienė T., Indriulionis A. Occupancy Prediction Using Differential Evolution Online Sequential Extreme Learning Machine Model. Energies 2020:13(15):4033. https://doi.org/10.3390/en13154033

[16] EN 16798-1:2019 Energy performance of buildings – Ventilation for buildings – Part 1: Indoor environmental input parameters for design and assessment of energy performance of buildings addressing indoor air quality, thermal environment, lighting and acoustics – Module M1-6.

[17] Candanedo L. M., Feldheim V. Accurate occupancy detection of an office room from light, temperature, humidity and CO2 measurements using statistical learning models. Energy and Buildings 2016:112:28–39. https://doi.org/10.1016/j.enbuild.2015.11.071

[18] Han M., Zhao J., Zhang X., Shen J., Li Y. The reinforcement learning method for occupant behaviour in building control: A review. Energy and Built Environment 2021:2(2):137–148. https://doi.org/10.1016/j.enbenv.2020.08.005

[19] Huang G.-B., Zhu Q.-Y., Siew C.-K. Extreme learning machine: A new learning scheme of feedforward neural networks. Proc. International Joint Conference on Neural Networks. 2004:1098–7576. https://doi.org/10.1109/IJCNN.2004.1380068.

[20] Storm R., Price K. Differential Evolution—A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces. International Computer Science Institute, CA, Technical Report TR-95-012, Berkeley, USA, 1995.

[21] Huang G. B., Babri H. A. Upper bounds on the number of hidden neurons in feedforward networks with arbitrary bounded nonlinear activation functions. IEEE Transactions on Neural Networks. 1998:9(1):224–229. https://doi.org/10.1109/72.655045

[22] Huang G. B. Learning capability and storage capacity of two-hidden-layer feedforward networks IEEE Transactions on Neural Networks 2003:14:274–281. https://doi.org/10.1109/TNN.2003.809401

[23] Huang G. B., Liang N. Y., Rong H. J., Saratchandran R., Sundararajan N. On-line sequential extreme learning machine. The IASTED International Conference on Computational Intelligence. Calgary, Canada, 2005.

[24] Huang G. B., Zhu Q. Y., Siew C. K. Extreme learning machine: Theory and applications. Neurocomputing 2006:70(1–3):489–501. https://doi.org/10.1016/j.neucom.2005.12.126

[25] Huang G. B. An Insight into Extreme Learning Machines: Random Neurons, Random Features and Kernels. Cognitive Computation 2014:6:376–390. https://doi.org/10.1007/s12559-014-9255-2

[26] Chen C., Li K., Duan M., Li K. Chapter 6 - Extreme Learning Machine and Its Applications in Big Data Processing, In Intelligent Data-Centric Systems. Cambridge: Academic Press, 2017:117–150. https://doi.org/10.1016/B978-0-12-809393-1.00006-4

[27] Song S., Wang Y., Lin X., Qingbao H. Study on GA-based Training Algorithm for Extreme Learning Machine. 7th International Conference on Intelligent Human-Machine Systems and Cybernetics, Hangzhou, China, 2015. https://doi.org/10.1109/IHMSC.2015.156

[28] Roshani A., Roshani A., Roshani A., Salehi M., Esfandiyari A. A simulated annealing algorithm for multi-manned assembly line balancing problem. Journal of Manufacturing Systems 2013:32(1):238–247. https://doi.org/10.1016/j.jmsy.2012.11.003

[29] Sieniutycz S., Jeżowski J. Brief review of static optimization methods. Energy Optimization in Process Systems and Fuel Cells (Third Edition) 2018:1–41. https://doi.org/10.1016/B978-0-08-102557-4.00001-3

[30] Zhan S., Lin J., Zhang Z., Zhong Y. List-Based Simulated Annealing Algorithm for Traveling Salesman Problem. Corporation Computational Intelligence and Neuroscience 2016:1712630. https://doi.org/10.1155/2016/1712630