2018 IEEE 14th International Conference on Control and Automation (ICCA)
June 12-15, 2018. Anchorage, Alaska, USA

Intelligent Load Forecasting for Building Energy Management Systems

Cristina Nichiforov, Grigore Stamatescu, Iulia Stamatescu, Ioana Făgărăşan and Sergiu Stelian Iliescu

Abstract— **As the built environment currently plays a determining role as driver of energy consumption in the developed world, with economic, social and environmental impact, better understanding of the underlying consumption models can lead to more reliable energy efficiency applications and programs. Within these frameworks, smart buildings generate large amounts of data from networks of embedded sensors and metering devices which have to be efficiently exploited for real-time decision and control. Lately, increasing attention is being given to black-box models which serve as a suitable alternative to more conventional system identification methods. In this context data-driven methods to derive reliable predictive models are needed on benchmark datasets that enable implementation and validation. We present the application and evaluation of feed-forward auto-regressive neural networks to intelligent load forecasting of smart buildings. By using public benchmarking datasets, replicable research is enabled as well as direct performance evaluation in both error metrics complexity. Experimental results are discussed with conclusion and outlook for implementation in energy management.**

I. INTRODUCTION

Reliable and efficient energy generation, transport, consumption and storage, while essential for the modern society, poses significant engineering challenges from the perspective of multiple domains. From an automation perspective many of the challenges are already addressed by conventional methods of system identification, control algorithms and optimisation. However increasing demands for real-time response of such complex heterogeneous systems has presented the limitations of the conventional approaches and spawned new research areas. This has been tackled recently through the development of new paradigms, such as Cyber-physical Energy Systems (CPES) which bind together the control. computing, communication and cognition layers of a complex system for an integrated view. One core idea of this paradigm is that the hierarchical control approach becomes flatter with distributed local computational nodes handling time sensitive, low latency, tasks while intense computational loads are being offloaded to a remote supervisor.

Focusing the current issues with concern to the Smart Buildings domain, the working definition that we adhere to refers to the building automation as a collection of increasingly capable end devices such as sensors, meters, actuators and local controllers, meshed-up together through a communication networks and related IT subsystems. They key objectives in the control of such systems relate to energy awareness and occupant friendliness, both objectives being handled in a competitive manner. In practice these objectives are translated into more or less complex rule-based or optimisation intensive algorithms. The Building Energy Management System (BEMS) represents the information system which handles all aspects related to data acquisition, monitoring and control for the various subsystems that deliver and consume energy in the building and implements at its core the models and control methods of choice.

As producer and consumer of structured and unstructured data the building information systems require intelligent methods and algorithms which can generate accurate yet robust models of the underlying phenomena. Among the various methods for intelligent modelling and control we focus on artificial neural networks (ANN), as universal approximators suitable to a wide class of modelling and optimisation problems. The well studied structure of such systems along with the recent increase in the availability of efficient libraries and computational power has led to increased adoption in both academia and industry. The common view is to put into balance the benefits of approximating virtually any nonlinear function with the drawback of high complexity and necessary high quantities of quality training data. In this context, the end goal of this contribution is to assess the performance of neural networks on raw electrical metering data streams, with an outlook of leveraging the resulting models into demand response schemes for the grid.

The main contributions of the papers are argued to be the following:

- Discussion on black-box modelling of energy consumption patterns through ANNs for medium and large scale commercial buildings;
- Methodology for the application of feed-forward autoregressive neural networks to load forecasting applications, in the future context of data-driven control;
- Experimental validation of the proposed methods, including computational assessment of the design decisions for neural network modelling, complexity and scalability issues.

The rest of the paper is structured as follows. Section II discusses related work in the context of electrical load modelling and forecasting for smart buildings. Section III focuses on neural network background and methodology enabling input-output black box modelling. Section IV discusses the main findings achieved by applying various configurations of feed-forward neural networks to two benchmark datasets stemming from real university buildings. Conclusions and

*This work has been partially funded by University "Politehnica" of Bucharest, through the "Excellence Research Grants" Program, UPB-GEX. Identifiers: UPB-EXCELENTA-2017 "Distributed Control in Cyber-Physical Energy Systems" (CYNERGY), 23/25.09.2017.

The authors are with the Department of Automatic Control and Industrial Informatics, University "Politehnica" of Bucharest, 313 Splaiul Independentei, 060042 Bucharest, Romania grigore.stamatescu@upb.ro

outlook for future work as well as closing the loop are discussed in section V.

II. RELATED WORK

The field of data-driven control (DDC) is emerging as a complementary approach to conventional control in many applications. The authors of [1] carry our a comprehensive survey on this topic by introducing the field by comparison to the challenges face in certain domains by conventional model-based control (MBC). The application of ANNs is considered as one of the key techniques for enabling DDC in situations where classic model uncertainty and/or complexity become unacceptable for control purposes.

Derivation of analytical models from raw data streams in smart buildings is discussed by [2]. The authors saliently conclude that several techniques might be suitable to achieve good results depending on the type of data, particularities of the building and expert knowledge available as well as the metrics being defined for evaluation. Best results are achieved in their smart building case study using bayesian regularized neural networks (BRNN) and random forests (RF). [3] design and validate an neural network adaptive model for increasing energy efficiency in a smart home by temperature control. The specific application is designed to predict the required time for heating the home using six predictive variables: outside temperature, wind speed, room temperature, target temperature, direct sunlight and presence.

Multiple current methods for time series forecasting in energy applications are surveyed by [4]. Nine methods are identified and discussed comparatively. These include: moving average and exponential smoothing (MA & ES), nearest neighbors (NN), ARIMA models, artificial neural networks (ANN), support vector machines (SVM) for regression, case-based reasoning, fuzzy logic, grey models, as well as hybrid approaches which combine elements from the others. The paper [5] provides an in-depth study of neural network models: NAR and NARX implemented for time series energy forecasting in public buildings. The discussion focuses on the requirements that NAR models have on several time delays to produce accurate predictions and a thicker structure: number of layers and neurons compared to NARX. NARX models however require the availability of exogenous information, that might not be readily available, difficult or expensive to collect. In the recent period, multiple types of recurrent neural networks are being studied in time series modelling and forecasting. These have origin in industrial research on deep learning, focused on image and video processing as well as natural language processing (speech, text, translation) and require adaptations in our case.

Building upon previous work, in [6] the comparison between ARIMA and ANN models has been evaluated in depth on locally collected data. [7] presents a decision support system implementation for integration of where intelligent techniques are suitable. We further develop these works by going into a more thorough analysis of ANN models structure for long term load forecasting purposes.

III. METHODS FOR ELECTRICAL LOAD FORECASTING

A. The non-linear autoregressive neural networks

Because of the good results and performance achieved in many documented cases of real applications, time series neural networks are considered one of the most suitable methods of also describing predictive models for electrical load forecasting. One of the considerable benefits of these dynamic neural networks is represented by their capability to model a large variety of non-linear time series problems. In most cases, time series applications are characterized by large variations and transient periods. This makes it really hard to model time series using a linear model, thus, should be recommended a non-linear approach.

A process is called autoregressive of order n, AR(n), if it is represented by the following equation:

$$x_t = F(X_{t-1}) + \epsilon_t \tag{1}$$

where $F : \mathbb{R}^n \to \mathbb{R}$, $X_t = (x_{t-1}, x_{t-2}, ..., x_{t-n})^T$ and ϵ_t is an independent and identically distributed random variable. If $F(X_{t-1})$ is a non-linear function then, the model is a non-linear AR.

The first term of the equation (1) is called predictable part and the second term, stochastic part. The non-linear models try to improve the predictable part in order to explain the process rather than to add some more stochastic components or to introduce some assumptions which are undoubtedly difficult to handle [8], [9].

Considering the graphic architecture and the interconnections between neurons, neural networks can be classified into two types: feed-forward and feedback networks. The first type network represents a non-linear function of its inputs and is represented as a set of interconnected neurons, in which information is transmitted only in the forward direction, from inputs to outputs. Such a model, with a single output neuron and q hidden layers, is defined by the following equation:

$$o_t = \Phi(\beta_0 + \sum_{i=1}^{q} \beta_i \Psi(\alpha_i + \sum_{j=1}^{n} \omega_{ij} x_{j,t})) \equiv f(x_t; \Theta), \tag{2}$$

where:

o_t represents the estimator of the target y_t;

$\Phi(\cdot)$, $\Psi(\cdot)$ are the activation functions of the network;

$x_t = (x_{1,t}, ..., x_{n,t})$ are the input variables at time t;

$\Theta = (\beta_0, \beta_1, ..., \beta_q, \alpha_1, \alpha_2, ..., \alpha_q, \omega_{11}, ..., \omega_{qn})$ represent the parameters vector of the network. Θ is calculated by minimizing the sum of squared differences:

$$\sum_{t=1}^{n} (y_t - \hat{o}_t)^2. \tag{3}$$

The feed-forward neural networks are more studied and applied in practice than the feedback ones because they are more effective in what may concern their implementation and simulation.

A non-linear autoregressive neural networks is a feed-forward network that can learn to predict a time series y given past values of y. The non-linear autoregressive model of order p, defined as:

$$y_t = h(y_{t-1}, y_{t-2}, ..., y_{t-p}) + \epsilon_t \tag{4}$$

represents a direct generalization of a linear autoregressive model (AR), where $h(\cdot)$ is a non-linear well known function and $\epsilon(t)$ is considered to be a sequence of random independent variables and similarly distributed with zero mean and finite variance σ^2.

The autoregressive neural network, is a feed-forward network that represents a non-linear approximation $h(\cdot)$, which is defined as:

$$\hat{y}_t = \hat{h}(y_{t-1}, y_{t-2}, ..., y_{t-p}) \tag{5}$$

$$\hat{y}_t = \beta_0 + \sum_{i=1}^{q} \beta_j f(\alpha_i + \sum_{j=1}^{p} \omega_{ij} y_{t-j}), \tag{6}$$

where $f(\cdot)$ is the non-linear activation function and:

$$\Theta = (\beta_0, \beta_1, ..., \beta_q, \alpha_1, \alpha_2, ..., \alpha_q, \omega_{11}, ..., \omega_{qp}) \tag{7}$$

represent the parameters vector [8], [9], [5].

B. Benchmarking data sets

The development of the current study was done in three stages. First stage: data pre-processing, second stage: non-linear autoregressive neural network (NAR) modelling using different network configurations and third stage: the assessment of performance and analogy between all distinctive forecasting NAR models.

Data sets used for modelling are taken from the Building and Urban Data Science (BUDS) Group at the National University of Singapore - `http://www.budslab.org` and are part of a data collection of several non-residential buildings, proposed for performance analysis and algorithm benchmarking [10].

There were used two sets of data containing the active energy consumed mainly by offices in two educational buildings with an approximate surface area of 9.000 square meters. The buildings are two universities, one from Chicago (USA) and the other one from Zurich (Europe). Choice of the target buildings was done in conjuction to a local campus building at our university to which a data collection study is currently underway. The determining factors were size - medium to large building, mixed usage pattern - office, laboratory space, some classrooms and non-extreme temperate climate with four distinct seasons.

Each data set contains 8.760 values, collected at a sampling time of one hour, over a 1-year period.

IV. EXPERIMENTAL RESULTS

For data pre-processing it was used the median filter technique in order to remove the noise from the data set. This approach is a typical pre-processing step to improve the results of later processing and the main idea of this technique is to go through the data set sample by sample, replacing each value with the median of the neighbouring values.

The two data sets used for models estimation, validation and testing can be seen in Figure 1. The upper plot represents the active energy consumed by the university building from Zurich over 1-year period and the other one by the university from Chicago.

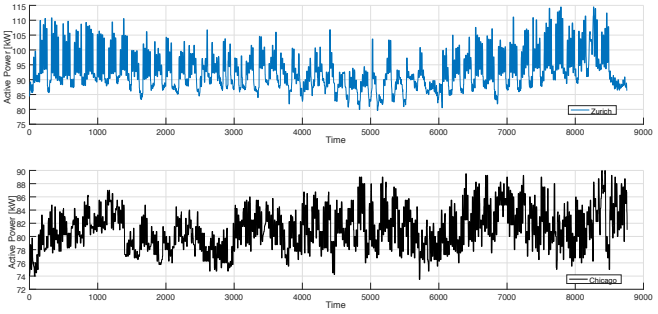

Fig. 1: Data sets used for models estimation and validation

Based on the two data sets, we proposed several configurations for NAR models in order to predict the energy consumption in such buildings. The goal was to have both an evaluation of the performance metrics of the various structure as well as a computational assessment on the target data set. The proposed networks are the following:

Defined networks for Zurich (Z) building data set:

- *Z-N0:* 1 hidden layer with 8 neurons and 1 output layer with 1 neuron;
- *Z-N1:* 2 hidden layers with 8 and 16 neurons, respectively and 1 output layer with 1 neuron;
- *Z-N2:* 3 hidden layers with 8, 8 and 16 neurons, respectively and 1 output layer with 1 neuron;
- *Z-N3:* 4 hidden layers with 8, 8, 16 and 16 neurons, respectively and 1 output layer with 1 neuron;
- *Z-N4:* 5 hidden layers with 8, 8, 8, 16 and 16 neurons, respectively and 1 output layer with 1 neuron;
- *Z-N5:* 6 hidden layers with 8, 8, 8, 16 and 16 neurons, respectively and 1 output layer with 1 neuron.

Defined networks for Chicago (C) building data set:

- *C-N0:* 1 hidden layer with 8 neurons and 1 output layer with 1 neuron;
- *C-N1:* 2 hidden layers with 16 and 8 neurons, respectively and 1 output layer with 1 neuron;
- *C-N2:* 3 hidden layers with 16, 8 and 8 neurons, respectively and 1 output layer with 1 neuron;
- *C-N3:* 4 hidden layers with 16, 16, 8 and 8 neurons, respectively and 1 output layer with 1 neuron;
- *C-N4:* 5 hidden layers with 16, 16, 8, 8 and 8 neurons, respectively and 1 output layer with 1 neuron.

All defined networks are standard feed-forward neural networks with different configurations of hidden layers and one output layer. For training the networks the Levenberg - Marquardt (LMBP) back-propagation learning algorithm was used. The Levenberg-Marquardt algorithm is a technique for solving non-linear least squares problems. Least squares

problems appear in the context of fitting a parametrized function to a set of measured data points by minimizing the sum of the squares of the errors between the data points and the function [11], [12]. This algorithm adaptively varies the parameter updates between the gradient descent update and the Gauss-Newton update:

$$[J^T W J + \lambda I] h_{lm} = J^T W (y - \hat{y}), \tag{8}$$

where small values of the λ result in a Gauss-Newton update and large values result in a gradient descent update. λ is initialized to be large so that first updates are small steps in the steepest-descent direction. If any iteration happens to result in a worse approximation, meaning: $\chi^2(p + h_{lm}) > \chi^2(p)$ then λ is increased, otherwise is decreased.

The Levenberg-Marquardt algorithm approaches the Gauss-Newton method and the solution typically accelerates to the local minimum. The Levenberg Marquardt update relationship is:

$$[J^T W J + \lambda diag(J^T W J)] h_{lm} = J^T W (y - \hat{y}), \tag{9}$$

where the values of λ are normalized to the values of $J^T W J$ [11], [12].

Regarding the training process, it is controlled by a cross-validation technique that consists in randomly dividing the initial set of data into three subsets. The three subsets were defined as follows: the first 70% of the data, was used for training the NAR. The second subset, comprising 15% of the data, was used for validation, and the last 15% was used for testing the model.

The two following plots show that the degree of matching of real measured values with predicted values is very good for one step predicted output and it indicates that each NAR model has a good prediction performance. Figure 2 presents the prediction response by a NAR model with one hidden layer with 8 neurons versus the first real data set (university building from Zurich).

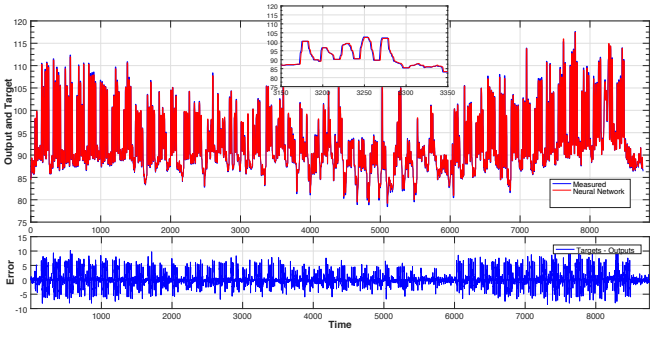

Fig. 2: Prediction result by NAR model with 1 hidden layer with 8 neurons - Zurich

Figure 3 shows the prediction result by a NAR model with 1 hidden layer with 6 neurons versus the second real data set (university building from Chicago).

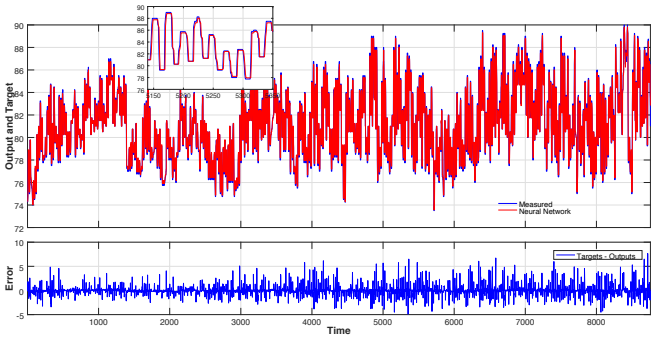

Fig. 3: Prediction result by NAR model with 1 hidden layer with 8 neurons - Chicago

Regarding the methods used for performance analysis of the models, four criteria were used to measure prediction performance. These are: Mean Squared Error (MSE), Root Mean Squared Error (Root Mean Squared Error) RMSE), Normalized Mean Squared Error (NMSE), Correlation Coefficient (R). They are described by the following equations:

$$\begin{aligned} MSE &= \sum_1^n \frac{(Y_t - Yp_t)^2}{n} \\ RMSE &= \sqrt{\sum_1^n \frac{(Y_t - Yp_t)^2}{n}} \\ NMSE &= \frac{\sum_1^n n(Y_t - Yp_t)^2}{\sum_1^n Y_t \sum_1^n Yp_t} \end{aligned} \tag{10}$$

where Y_t and Yp_t represent the vector of the actual values and predicted values, respectively.

The smaller the mean squared error, the better is the model prediction. Although, depending on the data, it may be impossible to get a very small value for the mean squared error. If the correlation coefficient value and the normalized mean squared error are close to 1, it also means that the model has a good approximation.

In the following tables the values of these errors for each defined neural network are shown.

TABLE I: Forecasting performance of each NAR - Zurich

	Z-N0	Z-N1	Z-N2	Z-N3	Z-N4	Z-N5
Time(s)	< 1	1	5	25	112	265
MSE	2.2592	2.2546	2.2525	2.2499	2.2485	**2.2510**
RMSE	1.5031	1.5015	1.5008	1.5	1.4995	**1.5003**
NMSE	0.9489	0.949	0.9492	0.9493	0.9495	0.9492
R	0.9754	0.9754	0.9754	0.9754	0.9754	0.9754

TABLE II: Validation error for each NAR - Zurich

	Z-N0	Z-N1	Z-N2	Z-N3	Z-N4	Z-N5
MSE	2.7752	2.3422	2.0551	1.9838	1.9685	**2.0628**

First of all, it can be noticed that for every proposed neural network the normalized mean squared errors and also the correlation coefficient values are very close to 1 which means, as it was mentioned before, that all the prediction

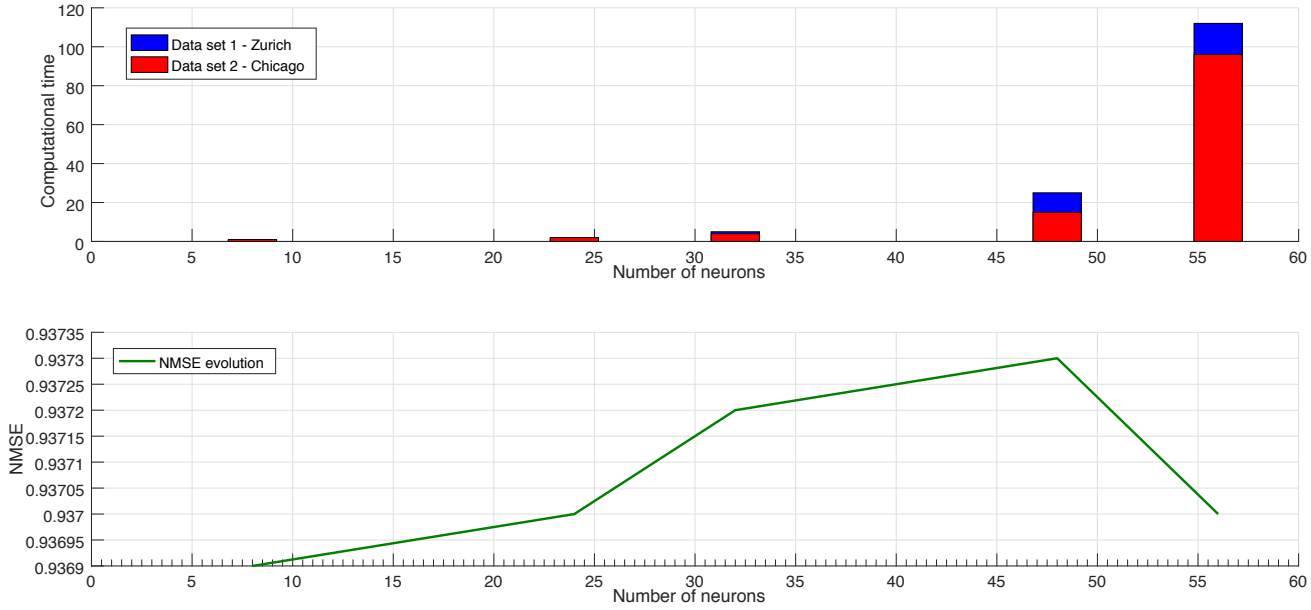

Fig. 4: NAR performances

TABLE III: Forecasting performance of each NAR - Chicago

	C-N0	C-N1	C-N2	C-N3	C-N4
Time(s)	< 1	5	6	27	128
MSE	0.788	0.777	0.757	0.738	**0.766**
RMSE	0.8782	0.8779	0.8774	0.8769	**0.8771**
NMSE	0.9243	0.9246	0.9247	0.9248	0.9246
R	0.9644	0.9644	0.9645	0.9645	0.9644

TABLE IV: Validation error for each NAR - Chicago

	C-N0	C-N1	C-N2	C-N3	C-N4
MSE	0.859	0.842	0.831	0.734	**0.775**

models have a good accuracy. Taking into account that the prediction accuracy is very good for every proposed network, it can be said that the mean squared error and root mean squared errors have acceptable values. There are no imposed limits for MSE except that the lower the MSE, the higher the accuracy of prediction, as there would be good match between the real and predicted data sets.

Figure 4 presents neural networks performances. The first sub-plot shows that increasing the number of layers and implicitly the number of neurons has a reduced impact on performance, but the computational time consumed to train the networks grows exponentially. On the other hand, the second sub-plot shows the fact that increasing the number of neurons from the hidden layers has some influence on networks quality in a positive way until overshooting (the value of the normalized mean squared error is increasing until layer number 5 that has 56 neurons when, it begins to decrease). This means that growing the number of neurons leads to computational time increasing but offers a better performance regarding the prediction.

We also investigate a NARX (nonlinear autoregressive model with exogenous input) model structure in order to observe if the prediction quality will be better than the NAR model. A NARX model is a recurrent dynamic network which relates the current value of a time series with the trained and predicted past values of the driving (exogenous) series. For the exogenous component it was used the outside temperature data set. Specifically, the defining equation for the operation of NARX model is:

$$y_t = h(y_{t-1}, y_{t-2}, ..., y_{t-p}, u_{t-1}, u_{t-2}, ..., u_{t-p}), \quad (11)$$

where u_n and y_n stand for the externally determined training input (temperature) and respectively the target variables to be predicted.

Figure 5 presents the prediction response by a NARX model with one hidden layer with 8 neurons versus real data sets (univeristy building from Chicago). The plot shows that the matching degree of real measured values with predicted values are also very good for one predicted output.

The value of the training mean squared error (MSE) is 0.7768 and the value of the validation MSE is 0.6714. Also, the regression (R) value which measure the correlation between outputs and targerts is 0.9643 for the training.

It can be seen that the values for the MSE and R are better than the one obtained by the NAR model but that is not necessary a significantly better performance.

It is interesting to see that there is no clear performance gain from 1 layer to 6 layers and 5 layers, respectively. In fact, performance decreases when the number of layers exceeds 4 for the first data set and 5 for the second data set. The computational time consumed to train the network does increase as the number of hidden layers grows. For networks with more than one hidden layer, pre-training is used for each layer to decrease the training time. These experiments lead us to the conclusion that more layers is not always a better solution and it depends on the processes and the input data sets used for prediction.

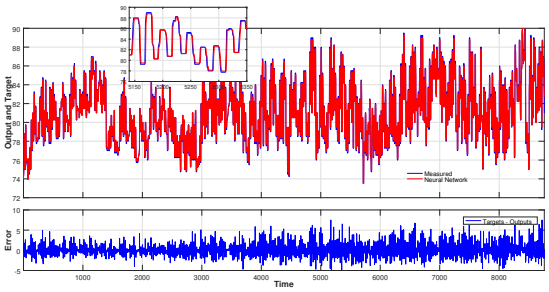

Fig. 5: Prediction result by NARX model with 1 hidden layer with 8 neurons - Chicago

V. CONCLUSIONS

Energy consumption modelling through load forecasting is a subject of high interest especially for large consumers which require rigorous patterns in order to assess future energy demands and also to understand how the current system can be improved. In this paper different NAR models were developed for energy prediction.We presented different non-linear autoregressive neural networks implemented to identify patterns for time series. Real data for modelling problems was obtained from a publicly available database of energy consumption of medium and large buildings. The data sets were used to train, validate and test the NAR forecast models. This study includes the empirical results obtained on the performance of 11 NAR models in predicting energy consumption in two given perimeters over a time period of one year. The comparison was made to highlight which of the two methods is more accurate in the context presented.

Future work is currently focused on including outside temperature information as the exogenous component of a class of NARX models and deploying these for online load forecasting as a component of a building information system. We also aim at exploring the potential of deep learning based approaches using different RNN models for energy time series forecasting.

REFERENCES

[1] Z.-S. Hou and Z. Wang, "From model-based control to data-driven control: Survey, classification and perspective," *Information Sciences*, vol. 235, pp. 3 – 35, 2013, data-based Control, Decision, Scheduling and Fault Diagnostics.

[2] A. González-Vidal, V. Moreno-Cano, F. Terroso-Sáenz, and A. F. Skarmeta, "Towards energy efficiency smart buildings models based on intelligent data analytics," *Procedia Computer Science*, vol. 83, pp. 994 – 999, 2016.

[3] T. Teich, F. Roessler, D. Kretz, and S. Franke, "Design of a prototype neural network for smart homes and energy efficiency," *Procedia Engineering*, vol. 69, pp. 603 – 608, 2014, 24th DAAAM International Symposium on Intelligent Manufacturing and Automation, 2013.

[4] C. Deb, F. Zhang, J. Yang, S. E. Lee, and K. W. Shah, "A review on time series forecasting techniques for building energy consumption," *Renewable and Sustainable Energy Reviews*, vol. 74, pp. 902 – 924, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1364032117303155

[5] L. G. B. Ruiz, M. P. Cuéllar, M. D. Calvo-Flores, and M. D. C. P. Jiménez, "An application of non-linear autoregressive neural networks to predict energy consumption in public buildings," *Energies*, vol. 9, no. 9, 2016.

[6] C. Nichiforov, I. Stamatescu, I. Făgărăşan, and G. Stamatescu, "Energy consumption forecasting using arima and neural network models," in *2017 5th International Symposium on Electrical and Electronics Engineering (ISEEE)*, Oct 2017, pp. 1–4.

[7] I. Stamatescu, N. Arghira, I. Fagarasan, G. Stamatescu, S. S. Iliescu, and V. Calofir, "Decision support system for a low voltage renewable energy system," *Energies*, vol. 10, no. 1, 2017. [Online]. Available: http://www.mdpi.com/1996-1073/10/1/118

[8] A. Tealab, H. Hefny, and A. Badr, "Forecasting of nonlinear time series using ann," *Future Computing and Informatics Journal*, vol. 2, no. 1, pp. 39 – 47, 2017.

[9] A. Reynaldi, S. Lukas, and H. Margaretha, "Backpropagation and levenberg-marquardt algorithm for training finite element neural network," in *2012 Sixth UKSim/AMSS European Symposium on Computer Modeling and Simulation*, Nov 2012, pp. 89–94.

[10] C. Miller and F. Meggers, "The building data genome project: An open, public data set from non-residential building electrical meters," *Energy Procedia*, vol. 122, pp. 439 – 444, 2017.

[11] H. P. Gavin, "The levenberg-marquardt method for nonlinear least squares curve-fitting problems," Department of Civil and Environmental Engineering, Duke University, Tech. Rep., 2017.

[12] K. Madsen, N. Nielsen, and O. Tingleff, "Methods for nonlinear least squares problems. technical report." Informatics and Mathematical Modelling, Technical University of Denmark, Tech. Rep., 2004.