Short-term load forecasting by separating daily profiles and using a single fuzzy model across the entire domain

Gregor Černe, Dejan Dovžan, and Igor Škrjanc, *Member, IEEE*

***Abstract*—The problem of energy load forecasting has emerged as an essential area of research for electrical distributors seeking to minimize costs. This problem has a high degree of complexity; therefore, this paper solves the problem of short-term load forecasting for a day ahead using an adaptive fuzzy model, defined across the entire input space in order to share information between different areas. The proposed solution first separates the forecasting of daily load profiles into smaller, simpler subproblems, which are solved separably using a Takagi-Sugeno fuzzy model. This is done in order to solve smaller subproblems better, which brings improved forecasting accuracy after combining the subproblem results. The identification of the model is based on a recursive Gustafson-Kessel clustering and recursive weighted least mean squares, to which a combined membership function is proposed in order to improve domain partitioning. The model was tested on the real data obtained from a large Slovenian energy distribution company, at which the developed model forecast outperformed other methods, especially in the start of the week and the winter.**

***Index Terms*—fuzzy adaptive model, combined membership function, profile separation**

I. Introduction

Short-term load forecasting (STLF) is essential in the electricity market for the participants to manage the market efficiently and stably. In a simplified model of electric retail chain (power plants → distributers → consumers), the demand (of the consumers) dictates the amount of produced energy, and distributers need to meet the demand by purchasing electric energy from power plants. Distributers want to minimize the cost of the needed energy, which depends greatly on the purchased price set by power plants. Power plants are selling energy at lower prices, if it is bought in advance. Therefore, electrical distributors attempt to purchase as much energy as soon as possible, but any error in the forecast brings additional costs. According to Bunn and Farmer [1], an increase of forecasting error of 1% caused an increase of 10 million pounds in operating costs per year for one electric utility in the UK in 1984. In order to buy the appropriate amount of electric energy, they have to correctly predict the electrical load of their customers and buy energy according to that forecast, meaning most of the STLF is the responsibility of electrical distributers.

The STLF problem has been tackled with many different techniques. The most frequently used method in literature is the autoregressive integrated moving average (ARIMA) [2]. ARIMA models have been used in STLF in many variations: seasonal ARIMA (SARIMA) [3], ARIMA transfer function model [4], an expert system that incorporated the ARIMA method [5] and an ARMA model including non-Gaussian process consideration [6]. After a paper reviewing the usage of artificial neural networks (ANN) in STLF was published in 2001 [7], that points out the problems of the ANN model, more research was done on this topic to overcome the mentioned problems, using bagged ANN [8], cascaded ANN [9], radial basis functions networks [10], combining a multilayer perceptron layer with hybrid Levenberg-Marquardt differential evolutionary learning algorithms and using an adaptive hybrid genetic optimization back-propagation neural network algorithm [11]. Furthermore, ANN learned by genetic algorithm was proposed in [12] [13].

Other methods have also been used for the STLF problem, but to a lesser extent: using general exponential smoothing [14], weighted Gaussian process regression [15], linear regression (LR) and robust linear regression (RLR) [16], the similar-day method [17] (it searches historical data for days with similar characteristics in comparison to the forecasting day) and others. The method of similar days was also combined with ANN and wavelet transform in [18]. Another widespread method in STLF are support vector machines (SVM) [19], which was combined with genetic algorithms [20], the similar-day method [21], and neural networks [22].

The problem of STLF can be viewed as a time-series prediction problem (e.g. [23]) in which the next-day load is predicted based on the current-day load. Not considering the date, temperature and other weather influences, such models can produce poor predictions. In other papers (e.g. [24]), the hourly load is treated independently (model for every hour), which is not necessarily the case if we look at Fig. 1.

Therefore, in this paper, we propose splitting the load forecasting problem into three subproblems: forecasting the average daily load, the amplitude of the load, and its shape. Each subproblem is solved separately and then merged back together.

Manuscript received July 4, 2017; revised August 31, 2017, November 1, 2017 and accepted January 8, 2018. (Corresponding author: Gregor Černe).

G. Černe, D. Dovžan and I. Škrjanc are with the Faculty of Electrical Engineering, University of Ljubljana, Ljubljana, Slovenia (e-mail: gregor.cerne@fe.uni-lj.si; dejan.dovzan@fe.uni-lj.si; igor.skrjanc@fe.uni-lj.si)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIE.2018.2795555, IEEE Transactions on Industrial Electronics

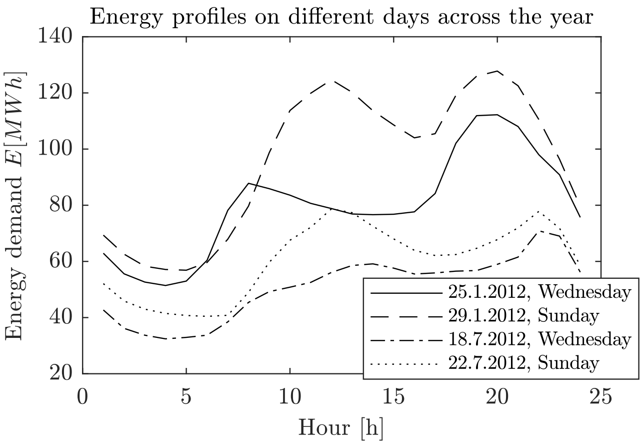


Fig. 1. Daily load profiles different days: summer weekday, summer weekend, winter weekday, winter weekend

Each subproblem is solved using an adaptive Takagi-Sugeno (TS) model because of its ability to model non-linearities in the process (seasons, weekends, etc.). The TS model has been successfully used in many applications [25]–[27] including STLF [28]–[30]. The online TS model learning methods were used to cope with changing relations across the seasons between the measured input variables and load profile. They can be divided into groups: *Adaptive methods* (e.g., ANFIS [31], recursive fuzzy c-means (rFCM), and used recursive Gustafson-Kessel (rGK) clustering [32]), where the number of clusters (or other structure model) is predefined, and only the parameters of local models and membership functions are adapted; *Incremental methods* (e.g. DENFIS [33], eTS [34], FLEXFIS [35], where only adding new complexity to existing structure is implemented; *Evolving methods* (e.g. SAFIS [36], EFuNN [37], eTS+ [38], eFuMo [39], TSCIT2FNN [40] (also used in STLF) where, in addition to adding the complexity to the structure, simplifying the model structure is also implemented. In this paper, the adaptive model was used, as the energy load structure is similar through the years, and there is no need for incremental or evolving models.

This paper contributes in three points: for improving STLF, splitting the daily profile into average, shape, and amplitude is proposed (1), as is using a single model valid across the whole input domain (2). To improve fuzzy clustering, a combined membership function is proposed (3), which joins information about input and output distance from the cluster prototype.

The organization of the paper is as follows: first, a problem statement is made with the description of the methods used. The proposed improvements are presented and demonstrated. Next, the modeling procedure is described along with regressor selection and model explanation. Then, the results are presented and compared to results obtained with conventional methodology. Finally, concluding remarks are given as well as some guidelines for future work and research.

II. Problem statement

The goal in short-term load forecasting (STLF) is to predict the load profile $\mathbf{E}(k)$ of the next day at once for the purpose of the day-ahead market. Load profile is defined with $\mathbf{E}(k) = [\mathrm{E}_1(k), \ldots, \mathrm{E}_{24}(k)]^T$, where $\mathrm{E}_l(k)$ is energy load on k-th day in l-th hour. Load can be forecast using measurements for previous days and weather forecast for the k-th day. In Fig. 1 and 2 examples of load profiles are shown. It can be seen that weekend profiles are much different than weekday profiles; the same can be said for winter and summer profiles.

Electrical load prediction is a non-linear time-variant problem, which has many non-measurable factors that increase the uncertainty of the prediction model, such as short-term migrations (holidays, business trips, sport events), long-term migrations, big public events in the area, the daily routines of people [41], factory shutdowns, etc. However, some factors do not change very rapidly and can be compensated to some extent with the adaptation of the prediction model, such as long-term migrations.

The STLF problem presented in this paper was defined with a real-world dataset. The data was provided for the SW region of Slovenia by one of the largest electrical distributers in the country. The data include three main groups of measured variables: *weather data* (temperature, precipitation, wind speed and sun radiation), *time categorical data* (hour, month, day etc.) and *electrical load* (electric energy). The provided data included hourly measurements for three years (2010 to 2012). In Table I the measured variables are represented with their notations.

TABLE I
Hourly sampled variables in dataset

Ime	**Symbol**	**Unit or value set**
Electric energy	E	MWh
Temperature	T	C
Sun radiation	Γ	W/m^2
Precipitation	R	mm
Wind speed	W	m/s
Hour (6 = 6:00 to 7:00)	H	$1 - 24$
Day in a week (1 = Monday)	F	$1 - 7$
Day in a month	G	$1 - 31$
Month	M	$1 - 12$
Year	Y	$2010 - 2012$
Holiday (1 = holiday, 0 = rest)	P	0 or 1

Another degree of difficulty when dealing with the database is that solar radiation, temperature, and wind speed are sampled only once per hour and are not averages in the hour. If the radiation is sampled at the moment a cloud passes by in an otherwise clear sky, the measurement is corrupted. This is also one of the reasons that separation was done (section IV-A). The aforementioned problem is referenced as *single measurements* in this paper.

A. Data preparation

The initial data analysis showed quite distinct differences among daily load profiles between weekends, weekdays and border days (days before or after the weekend or holiday). The latter has similarities to both - weekdays and weekends. To include this finding into the model, firstly special date variable

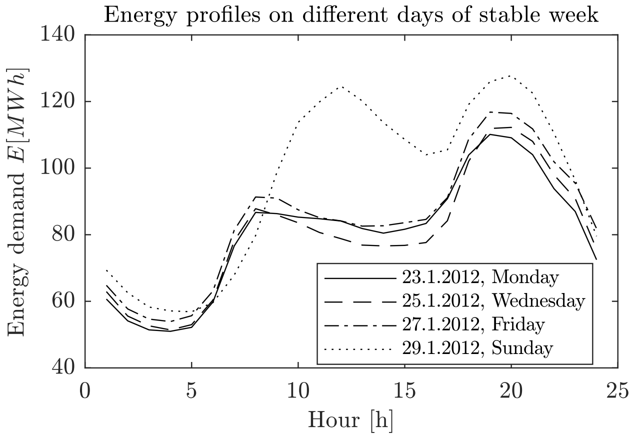


Fig. 2. Daily load profiles different days of a week with a stable weather

D^* is introduced. Variable D^* has value 0 at weekdays (F) and 1 at weekends (F) or holidays (P) (Eq. 1):

$$\mathrm{D}^*(k) = \begin{cases} 1, & \mathrm{F}(k) \in \{6,7\} \vee \mathrm{P}(k) = 1 \\ 0, & otherwise \end{cases} \tag{1}$$

To model border days, the influence of neighboring special days, presented by symbol $\mathrm{D}^{\pm}$ need to be taken into account. The value of $\mathrm{D}^{\pm}$ is calculated with Eq. 2, where $\mathrm{D}^*(k-1)$ and $\mathrm{D}^*(k+1)$ are weighted with parameters g_-, g_+, respectively and their product with $g_{\pm}$. The product $(g_{\pm}\mathrm{D}^*(k-1)\mathrm{D}^*(k+1))$ was added to the equation due to the following observation: if a weekday falls between two special days, it has very similar characteristics to the special days; thus, the joined influence $\mathrm{D}^{\pm}$ is bigger than the sum of next and previous days' influence.

Variables D^* and $\mathrm{D}^{\pm}$ are merged into the *date specialty* D with Eq. 3, in which larger one is used; therefore, it follows: if k-th day is a holiday or weekend $\mathrm{D}(k)$ is set to 1, otherwise it is defined with the influence of neighboring days $\mathrm{D}(k) = \mathrm{D}^{\pm}(k)$. In order to insure that $\mathrm{D}^{\pm}$ is always lower than 1 as the day cannot be more special than the special day itself, the condition under Eq. 4 must apply for the parameters g.

$$\begin{aligned} \mathrm{D}^{\pm}(k) =& g_+\mathrm{D}^*(k+1) + g_-\mathrm{D}^*(k-1) + \\ &+ g_{\pm}\mathrm{D}^*(k-1)\mathrm{D}^*(k+1) \end{aligned} \tag{2}$$

$$\mathrm{D}(k) = \max\Big[\mathrm{D}^{\pm}(k),\ \mathrm{D}^*(k)\Big] \tag{3}$$

$$g_+ + g_- + g_{\pm} \leq 1 \tag{4}$$

Another observation made when analyzing the data is that load is not linearly correlated with amount of precipitation: it mostly depends on whether it was raining or not. The reason for that might be that precipitation does not significantly influence heating, but it does influence whether people stay at home or go for a trip. This consequentially has an impact on electrical load. Therefore, the hourly precipitation is limited upwards to $\mathrm{R}_{\max} = \frac{10}{24}\frac{mm}{h}$ (Eq. 5). Value $\frac{10}{24}\frac{mm}{h}$ is defined in order to have the maximal sum of a precipitation in a day equal to 10 mm.

$$\mathrm{R}_l(k) = \begin{cases} \mathrm{R}_l(k), & \mathrm{R}_l(k) \leq \mathrm{R}_{max} \\ \mathrm{R}_{max}, & \mathrm{R}_l(k) > \mathrm{R}_{max} \end{cases} \quad ,\forall(l,k) \ , \tag{5}$$

where $\mathrm{R}_l(k)$ is precipitation in k-th day on l-th hour.

III. Methodology

In this section, the identification of the TS fuzzy model is described. In subsection III-A, a basic TS model is presented, subsection III-B explains the used recursive Gustafson-Kessel (GK) clustering and subsection III-C propose improvements to GK clustering. The section is concluded with subsections about the identification of local models (subsection III-D).

A. Takagi-Sugeno model

The TS model was first introduced in [42]. It is used in many different fields, most notably in the area of modeling nonlinear systems. The main idea of the TS model is to separate the input-output problem space into smaller local areas, where the system is simpler. For each local area, a local model is designed that approximates nonlinear system dynamics in that area. With the introduction of membership degrees, a smooth transition between the local models is achieved. The output of the TS model is defined as the weighted sum of local model outputs Eq. 6.

$$\hat{y} = \sum_{i=1}^{m} \mu_i(\mathbf{x}_p)\hat{y}_i(\mathbf{x}) \ , \quad \sum_{i=1}^{m} \mu_i(\mathbf{x}_p) = 1 \ , \tag{6}$$

where $\hat{y}$ represents the output of the TS model, $\mathbf{x} = [x_1, \ldots, x_{r-1}, 1]^T$ represents the regression vector, $\mathbf{x}_p = [x_{p_1}, \ldots, x_{p_q}]^T$ represents the partitioning vector, $\mu_i(\mathbf{x}_p)$ the membership degree function of i-th cluster, $\hat{y}_i(\mathbf{x})$ is output of i-th local model, and m is number of clusters.

To uniquely define the TS model, the designer needs to set:

- Membership functions $\mu_i(\mathbf{x}_p)$ for space partitioning
- Local models and their parameters.

In this paper, membership functions are defined using a GK clustering, and the local models' parameters are defined using a weighted least squares algorithm (WLSE). Both methods will be explained in the next sections.

B. Recursive GK (rGK) clustering

The membership functions can be defined using the GK clustering [43]. The GK algorithm defines local areas of the partition space with *clusters*. Each cluster is described with a prototype $\boldsymbol{v}_i$ that represents a point in the input space and with a fuzzy covariance matrix $\mathbf{F}_i$ that describes the size and shape of a cluster-data dispersion around the cluster prototype. These parameters can be, for example, directly used as parameters for Gaussian membership functions.

To adapt to changes in the system, a recursive GK clustering was proposed in [32]. The method is derived from the off-line version of the GK algorithm. Prototypes $\boldsymbol{v}_i$ and covariance matrices $\mathbf{F}_i$ of the i-th cluster are updated in every sample by equations found in [32].

Initial values $\boldsymbol{v}_i(0)$, $\mathbf{F}_i(0)$ and $s_i(0)$ are obtained on a training set of with off-line GK clustering described in [43].

C. Improved cluster membership

One of the problems observed using the original GK clustering for the STLF problem was due to its membership definition, which *does not consider the quality/outputs* of local models; therefore, only data density is considered. One possible consequence is clusters that include large non-linearities, which contradicts the purpose of TS models separating domains into simpler, preferably linear areas. In other papers (e.g. [34]), output is taken into account using GK clustering using input-output space for partitioning, but this method encounters the same problems in the case of the high dimension of input space, as output can become insignificant to clustering (only one of $N+1$ dimensions). Another cluster method is fuzzy c-regression (FCR) [44] in which clustering is done in the output space. However, different problem with similar consequences is encountered here: because only the output comparison is important, samples in the same cluster can be located very far from each other in the partition space. This results in overlapping clusters and, as a consequence, corrupted parameters are estimated as they are gained across multiple regions with different non-linearities (shown in Fig. 3 as FCR (bad), around $x = 0$). Therefore, a combined membership function is proposed, which combines the advantages of both the abovementioned clusterings.

In *GK clustering*, membership functions are described with (7), where $d_i(k)$ is Mahalanobis distance between cluster with center $\boldsymbol{v}_i$ and covariance matrix $\mathbf{F}_i$ and the sample $\mathbf{x}_p$.

$$\mu_i(d_i) = \frac{{d_i}^{\frac{-2}{\eta-1}}}{\sum_{j=1}^{m} {d_j}^{\frac{-2}{\eta-1}}} \tag{7}$$

In *fuzzy c-regression (FCR) algorithm*, membership functions are calculated based on an output distance ζ_i (8) (can be also interpreted as local output error e_i), which is the distance between model output $\hat{\mathbf{y}}_i(k)$ of i-th cluster and measured output of the system $\mathbf{y}(k)$ as seen in [45].

$$\zeta_i(k) = e_i(k) = f_y(\mathbf{y}(k), \hat{\mathbf{y}}_i(k)) \ , \tag{8}$$

where $f_y(\boldsymbol{a}, \boldsymbol{b})$ is a output distance function between two output vectors $\boldsymbol{a}$ and $\boldsymbol{b}$. In [45], output distance used was Euclidean and in this paper *Manhattan distance* was used, but others can also be used.

In order to combine the benefits of both methods (input space separability using d_i and taking into account model quality with ζ_i), this paper proposes a combined membership function using (Eq. 9), in which $\mu(d_i)$ is multiplied with the exponential part using relative output distance $\frac{\zeta_i}{\sigma_i}$ (σ_i is weighted standard deviation of ζ_i).

$$\mu_i^* = e^{-0.5\frac{\zeta_i}{\sigma_i \gamma_\sigma}} \cdot \mu_i(d_i) \ , \tag{9}$$

where $\mu_i(d_i)$ is the membership degree calculated using the Mahalanobis distance (Eq. 7) and γ_σ is the trust multiplier. The proposed method lowers the μ_i of samples, at which the local model produces a significant error (big ζ_i). This eliminates the outlier's influence, the result of which is that only samples with small model error are considered in building or adapting the model. Multiplier γ_σ is added for model designer to decide how dissimilar samples are included (bigger it is set, more samples are included). For the offline GK algorithm, the multiplier is set to $\gamma_\sigma = 1$ as it is desirable that the model be fitted according output similarity as much as possible. In rGK clustering, small γ_σ can identify system change as outlier, which can result in non-adapting model as the data is too far from local model. Therefore, the multiplier at rGK clustering is set higher; in this paper, it was empirically set to $\gamma_\sigma = 4$.

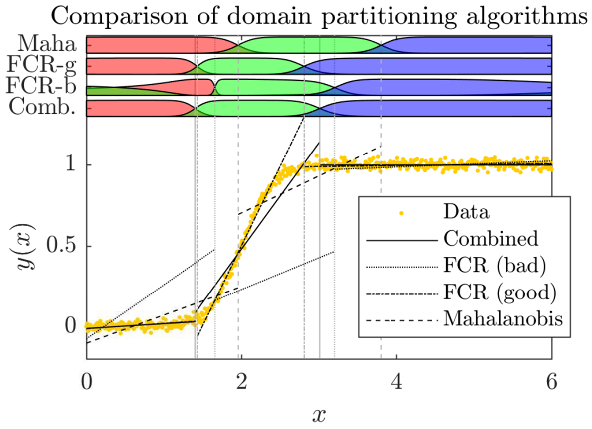


Fig. 3. Comparison of domain separation using different membership functions in sigmoid approximation problem. The domain separation is shown on top, and the resulting local models in the bottom (lines). Vertical lines (with the same type as local models) represents points, where neighbor clusters have same value of membership degree.

The proposed combined membership degree μ_i^* can be used in steps of GK, where output is given (clustering and adaptation). In the forecasting step, where only the input is given, μ_i^* cannot be used. Therefore, $\mu_i(d_i)$ is used as a membership function.

The combined membership degree was compared with original GK and FCR clustering on two datasets: on the noisy sigmoid and on $\overline{\text{E}}$ forecasting. The results from the $\overline{\text{E}}$ forecast are gathered in table III, the results for the noisy sigmoid problem in table II, and detailed results for noisy sigmoid are shown in Fig. 3. The best domain partitioning results are obtained with FCR partitioning, but the algorithm has a problem finding the global minimum and needs many runs to find it even with the simple problems, such as the sigmoid approximation (standard deviation or std of results is 13x bigger in comparison to combined membership); in bigger problems ($\overline{\text{E}}$ forecasting), the minimum is even harder to reach. If comparing original GK clustering to the proposed one based only on numerical results in table II, the combined GK clustering is better overall (minimum root mean square or RMS improved by 12.7%). If the qualitative comparison is done on results in Fig. 3, it can be seen that resulted domain separation of combined membership is clearer than the one from original GK clustering. Therefore, the combined membership function showed the best combination of accuracy and convergence to the global minimum in the given example problems.

TABLE II
NOISY SIGMOID - RMS ANALYSIS IN 100 RUNS

	min	*mean*	*std*	*max*
Mahalanobis	0.055	0.057	0.013	0.181
FCR	**0.043**	0.212	0.064	0.272
Combined	0.048	**0.049**	**0.0005**	**0.050**

TABLE III
LOAD AVERAGE $\overline{\mathrm{E}}$ FORECAST - MAPE ANALYSIS

	Mahalanobis	FCR	Combined
MAPE[%]	2.78	3.20	2.42

D. Defining local models

In this paper, linear models were chosen for the local models. The output of i-th local model is defined with (10).

$$\hat{y}_i = \mathbf{x}^T \boldsymbol{\theta}_i \quad , \tag{10}$$

where $\boldsymbol{\theta}_i = [\theta_{i1}, \ldots, \theta_{ir}]^T$ is the vector of linear model parameters of the i-th cluster. In this paper, vector $\boldsymbol{\theta}_i$ is estimated with weighted least mean squares (WLMS) method.

In this paper, sample weights of i-th model β_i are set the same as cluster membership function $\mu_i(k)$ for that sample (11), so only samples in a cluster will influence local model parameters $\boldsymbol{\theta}_i$. One drawback of the mentioned proposal is in outliers' influence on the local model, which can lower the quality of model prediction.

$$\beta_i(k) = \mu_i(k) \tag{11}$$

The WLMS method is an offline method to estimate initial parameters $\boldsymbol{\theta}_i^0$, which are adapted using a recursive WLMS (rWLMS) method [44].

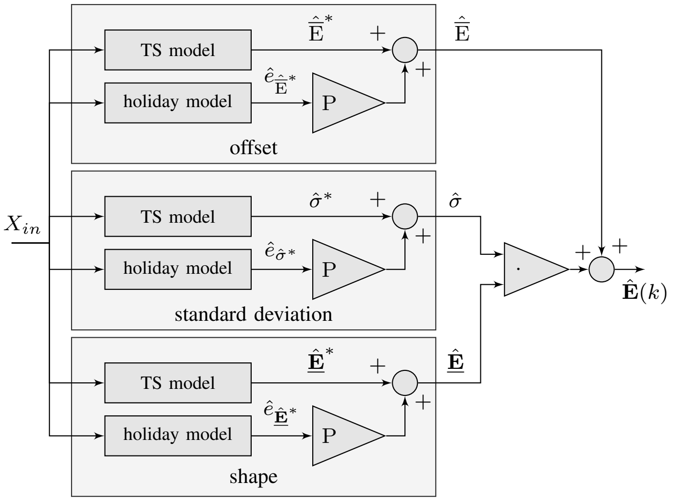
IV. MODEL

To obtain the daily profile forecast, the first load profile components are forecast separately. The load profile separation is explained in SectionIV-A and the methods for component forecast in SectionIV-B. To improve forecasts on holidays, the holiday model is introduced in SectionIV-C. The final model structure is shown in Fig. 4, and the complete forecasting algorithm is explained in section IV-D

A. Problem division

The energy load is dependent hour to hour; therefore, hourly dependency should be also included into the model. Consequently, the first main contribution of this paper is proposed in the division of the daily profile forecasting problem into three subproblems: forecasting of the average $\overline{\mathrm{E}}$ (12), the shape $\underline{\mathbf{E}}$ (16), and the standard deviation σ (13) separately (as seen on Fig. 5). The shape and amplitude subproblem focuses on modeling hourly dependencies, while the offset subproblem focuses on modeling daily changes.

Hourly weather forecasts are not reliable because of *single measurements* (section II); therefore, daily weather averages were used as input for all subproblems.



X_{in}
TS model
holiday model
P
$\hat{\overline{\mathrm{E}}}^*$
$\hat{e}_{\hat{\overline{\mathrm{E}}}^*}$
$\hat{\overline{\mathrm{E}}}$
offset
$\hat{\sigma}^*$
$\hat{e}_{\hat{\sigma}^*}$
$\hat{\sigma}$
standard deviation
$\hat{\underline{\mathbf{E}}}^*$
$\hat{e}_{\hat{\underline{\mathbf{E}}}^*}$
$\hat{\underline{\mathbf{E}}}$
shape
+
·
$\hat{\mathbf{E}}(k)$

Fig. 4. Block diagram of constructed model. Dark-gray represents submodels and mathematical operations, and light-gray represents area of forecasting subproblems $\overline{\mathrm{E}}, \underline{\mathbf{E}}$ and σ.

Because each component is globally valid, some local influences can be overlooked. In this paper, it is shown that it is possible to model local influences in addition to the existing component models. Therefore, the holiday error models $\hat{e}_{\hat{\overline{\mathrm{E}}}^*}$, $\hat{e}_{\hat{\sigma}^*}$ and $\hat{e}_{\hat{\underline{\mathbf{E}}}^*}$ are presented.

B. General component forecasting with TS models

The second main contribution of this paper is designing TS model to work accurately across whole year and all days of the week, using the rGK algorithm. Having a universal model can improve the forecast in two aspects: because of smooth fuzzy transitions that can model non-linearities, fewer clusters/areas are needed to model the whole domain. Consequently, there are more samples per area, therefore estimated parameters in area are more accurate. Second aspect is in *parameter sharing*, as temperature dependency derived from workdays can be shared to weekend days, where there are fewer samples with more noise. To obtain a general forecast of every component, TS model for each component was created separately using described upgraded rGK clustering and rWLMS algorithm. For every component, TS model was adjusted to it's specific behavior, which will be presented in the following subsections.

1) Offset and standard deviation general forecast: Average of daily load profile is calculated with (12) and standard deviation with (13). In this section forecasting of offset and standard deviation is presented together as they have similar properties.

$$\overline{\mathrm{E}}^*(k) = \overline{\mathrm{E}}(k) = \frac{1}{24}\sum_{l=1}^{24} \mathrm{E}_l(k) \tag{12}$$

$$\sigma^{*2}(k) = \sigma^2(k) = \frac{1}{24}\sum_{l=1}^{24} (\mathrm{E}_l(k) - \overline{\mathrm{E}}(k))^2 \tag{13}$$

After initial data analysis, strong correlation of average $\overline{\mathrm{E}}(k)$ and average of previous day $\overline{\mathrm{E}}(k-1)$ was found, as shown in Fig. 6. Therefore, $\overline{\mathrm{E}}(k-1)$ was set as a foundation for

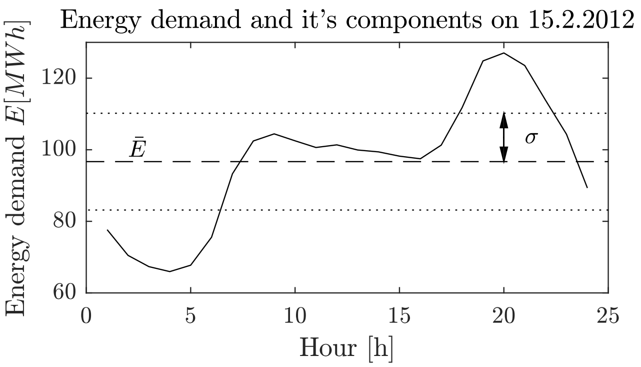


Fig. 5. Problem division on offset $\overline{\mathrm{E}}$, shape $\underline{\mathbf{E}}$ and standard deviation σ

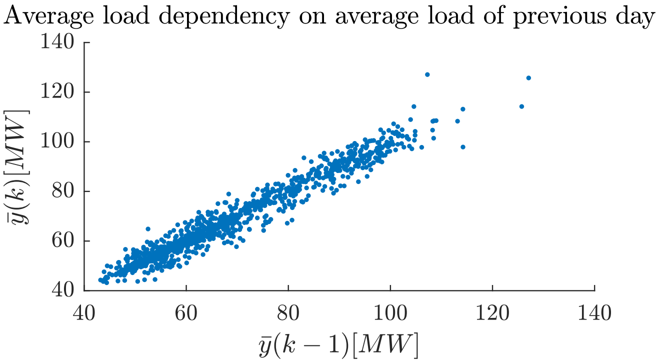


Fig. 6. Graph of load average $\overline{\mathrm{E}}(k)$ versus load average of previous day $\overline{\mathrm{E}}(k-1)$

the forecast. In order to include this finding, the average was forecast using (14), where the average forecast is based on the measured average of the previous day $\overline{\mathrm{E}}(k-1)$. The same applies for standard deviation (15).

$$\hat{\overline{\mathrm{E}}}^*(k) = \overline{\mathrm{E}}(k-1) + \Delta\hat{\overline{\mathrm{E}}}^*(k) \ , \tag{14}$$

$$\hat{\sigma}^*(k) = \sigma(k-1) + \Delta\hat{\sigma}^*(k) \ , \tag{15}$$

where $\hat{\overline{\mathrm{E}}}^*(k)$ stands for the general forecasting average, $\overline{\mathrm{E}}(k-1)$ for the measured average and $\Delta\hat{\overline{\mathrm{E}}}^*(k)$ for the general forecast change from previous day. The mentioned equation reduces average forecasting to forecasting the difference between the average of the previous and forecasting day. The same applies for standard deviation.

Because the difference $\Delta\hat{\overline{\mathrm{E}}}^*(k)$ / $\Delta\hat{\sigma}^*(k)$ is being forecast, only the variable differences are used in $\mathbf{x}(k)$ (table IV). Algorithm parameters are gathered in table V.

2) Shape general forecast: The shape vector of the daily load profile $\underline{\mathbf{E}}$ is calculated with (16). In this section, the forecasting of $\underline{\mathbf{E}}$ is presented

$$\underline{\mathbf{E}}^*(k) = \underline{\mathbf{E}}(k) = \frac{\mathbf{E}(k) - \overline{\mathrm{E}}(k)}{\sigma(k)} \tag{16}$$

Because correlation with previous value $\underline{\mathbf{E}}(k-1)$ was not prominent, $\underline{\mathbf{E}}(k)$ is forecast directly with daily variables used in the regression vector, as presented in TableIV. Algorithm parameters are gathered in TableV.

TABLE IV
PARTITIONING / REGRESSION VECTOR

Step	Regression vector $\mathbf{x}(k)$
$\overline{\mathrm{E}}$	$[\Delta\overline{\mathrm{T}}(k) \quad \Delta\overline{\Gamma}(k) \quad \Delta\overline{\mathrm{R}}(k) \quad \Delta\mathrm{D}(k) \quad \Delta\overline{\mathrm{W}}(k)]$
σ	$[\Delta\overline{\mathrm{T}}(k) \quad \Delta\overline{\Gamma}(k) \quad \Delta\overline{\mathrm{R}}(k) \quad \Delta\mathrm{D}(k)]$
$\underline{\mathbf{E}}$	$[\overline{\mathrm{T}}(k) \quad \overline{\Gamma}(k) \quad \overline{\mathrm{R}}(k) \quad \overline{\mathrm{W}}(k) \quad \mathrm{D}(k) \quad \overline{\mathrm{E}}(k-1)]$
Step	**Partitioning vector $\mathbf{x}_p(k)$**
$\overline{\mathrm{E}}$	$[\overline{\mathrm{T}}(k) \quad \overline{\mathrm{E}}(k-1) \quad \Delta\mathrm{D}(k) \quad \overline{\mathrm{W}}(k)]$
σ	$[\overline{\mathrm{T}}(k) \quad \hat{\overline{\mathrm{E}}}(k) \quad \Delta\mathrm{D}(k) \quad \sigma(k-1)]$
$\underline{\mathbf{E}}$	$[\overline{\mathrm{T}}(k) \quad \hat{\overline{\mathrm{E}}}(k) \quad \Delta\mathrm{D}(k) \quad \overline{\mathrm{W}}(k)]$

TABLE V
ALGORITHM PARAMETERS

	rWLSE	rGK & GK		
	γ_θ	γ_c	η	m
$\overline{\mathrm{E}}$	0.99	0.98	1.6	2
σ	0.99	0.99	1.6	3
$\underline{\mathbf{E}}$	0.97	1	2	3

C. Modeling holidays

The developed TS model yields good results for the majority of the days, but the model failed when observing results for holidays (like 1st January as shown in Fig. 7), and the forecasting error was greater than 10%. This error was also transferred to model parameters, because they were adapting to every sample including holidays.

Firstly, to improve model accuracy, *model adaptation was skipped on holidays* to prevent the corruption of model parameters.

Secondly, in order to model differences between a holiday profile and a normal one, a model was created to *forecast the holiday error* of each component of TS models ($\hat{e}_{\hat{\overline{\mathrm{E}}}^*}, \hat{e}_{\hat{\underline{\mathbf{E}}}^*}$ and $\hat{e}_{\hat{\sigma}^*}$). In this paper, 16 models were created, one for each holiday. The forecast holiday error is added to forecast components and joined with a final forecasting model equation

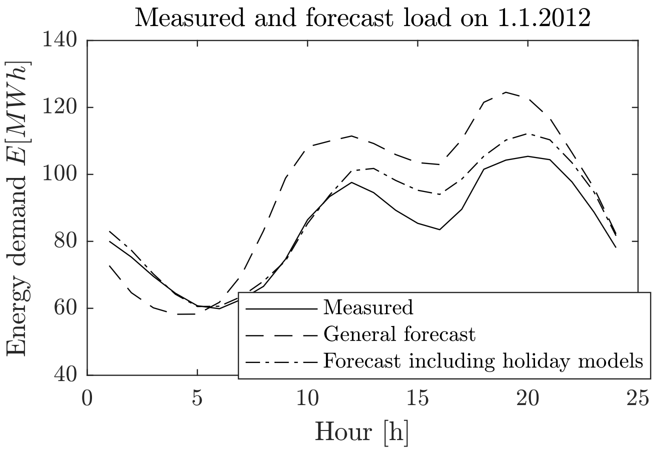


Fig. 7. Bad forecast on the first day of the year 1.1.2012

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIE.2018.2795555, IEEE Transactions on Industrial Electronics

(17).

$$\hat{\mathbf{E}} = ((\overbrace{\underline{\hat{\mathbf{E}}}^* + \mathrm{P}\hat{e}_{\underline{\hat{\mathbf{E}}}^*}}^{\underline{\hat{\mathbf{E}}}}) \cdot (\overbrace{\hat{\sigma}^* + \mathrm{P}\hat{e}_{\hat{\sigma}^*}}^{\hat{\sigma}})) + (\overbrace{\hat{\overline{\mathrm{E}}}^* + \mathrm{P}\hat{e}_{\hat{\overline{\mathrm{E}}}^*}}^{\hat{\overline{\mathrm{E}}}}) \ , \quad (17)$$

where $\hat{e}_{\hat{\overline{\mathrm{E}}}^*}$, $\hat{e}_{\underline{\hat{\mathbf{E}}}^*}$ and $\hat{e}_{\hat{\sigma}^*}$ are holiday error forecasts of TS models, $\hat{\overline{\mathrm{E}}}$, $\underline{\hat{\mathbf{E}}}$ and $\hat{\sigma}$ are corrected component forecasts, P is holiday variable and $\hat{\mathbf{E}}$ is model forecast.

We only have two samples per holiday in the training set; thus, the only model that can be created with satisfactory accuracy is an average model. Therefore, a model's output is calculated as an average of all TS model errors on that holiday in the historical data (18) (19).

$$S_k = \{i \in \{1, \ldots, k-1\} | \mathrm{M}(k) = \mathrm{M}(i) \wedge \mathrm{G}(k) = \mathrm{G}(i)\} \quad (18)$$

$$\hat{e}(k) = \frac{\sum_{i \in S_k} \left(x(i) - \hat{x}(i)\right)}{|S_k|} \ , \quad (19)$$

where S_k is set containing day indexes i of previous days in the dataset with same month and day as the k-th day, $|S_k|$ is the cardinality of S_k, $\hat{e}$ is $\hat{e}_{\hat{\overline{\mathrm{E}}}^*}, \hat{e}_{\underline{\hat{\mathbf{E}}}^*}$ or $\hat{e}_{\hat{\sigma}^*}$ and x is $\overline{\mathrm{E}}, \underline{\mathbf{E}}$ or σ depending on which component is being modeled. Symbol M represents numerical representation of a month and G represents a day in the month.

D. Forecasting algorithm

The proposed forecasting method is presented in algorithm 1. First, the component models are identified with the *model identification* procedure, which can be then used in the online load forecasting (forecasting step). To assure adaptability of the forecasting model, the models are adapted after every sample (adapting step). The forecasting and adapting steps are part of the online procedure, which is run for every new sample.

Algorithm 1 STLF by separating daily profile

```
Model identification
    Use data preparation methods
    Separate daily load profiles $\mathbf{E}$ into $\overline{\mathrm{E}}$, $\sigma$ and $\underline{\mathbf{E}}$
    General component model ($\hat{\overline{\mathrm{E}}}^*$, $\hat{\sigma}^*$ and $\underline{\hat{\mathbf{E}}}^*$) identification using offline GK clustering with combined membership function
Online procedures
    Load forecast for $k$-th sample
        Use data preparation methods
        Forecast general load components($\hat{\overline{\mathrm{E}}}^*$, $\hat{\sigma}^*$ and $\underline{\hat{\mathbf{E}}}^*$)
        Forecast holiday error ($\hat{e}_{\hat{\overline{\mathrm{E}}}^*}, \hat{e}_{\underline{\hat{\mathbf{E}}}^*}$ and $\hat{e}_{\hat{\sigma}^*}$)
        Join forecasts with (17)
    Model adaptation for $k$-th sample
        Adapt cluster parameters ($\boldsymbol{v}_i$, $\mathbf{F}_i$) using rGK
        Adapt local model parameters ($\boldsymbol{\theta}_i$) using rWLMS
```

TABLE VI
MAPE OF DIFFERENT PARTS OF THE MODEL

	Symbol	MAPE[%]
Component models	$\hat{\overline{\mathrm{E}}}$	2.27
	$\underline{\hat{\mathbf{E}}}$	13.34
	$\hat{\sigma}$	6.02
Developed model	$\hat{\mathbf{E}}$	3.68

V. Results and discussion

The quality of the model is measured by mean absolute percentage error (MAPE), calculated with (20).

$$\mathrm{MAPE}[\%] = \frac{100}{N} * \sum_{k=1}^{N} \sum_{l=1}^{24} \left| \frac{\mathrm{E}_l(k) - \hat{\mathrm{E}}_l(k)}{\mathrm{E}_l(k)} \right| \quad , \quad (20)$$

where N stands for the number of samples, $\mathrm{E}_l(k)$ and $\hat{\mathrm{E}}_l(k)$ for the measured load and forecast load on l-th hour of k-th day, respectively.

The developed model was trained on data from 2010/11 and tested on 2012. MAPE errors are gathered in TableVI, for the developed model and its components separately. The MAPE error of the developed model is 3.68%.

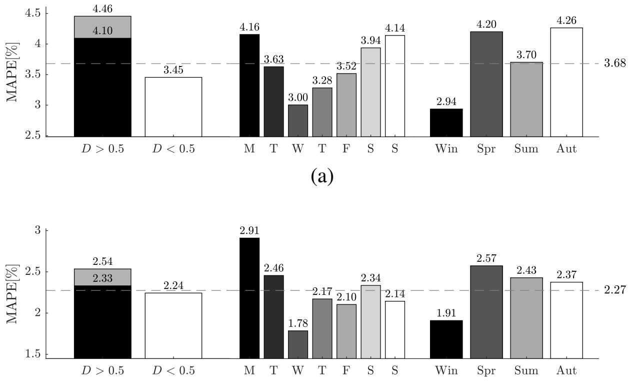
To obtain better insight into the results of the forecast, the MAPE of every component (developed model in Fig.8a and offset in 8b. Shape and standard deviation are omitted because of lack of space) was compared depending on the input variables - date specialty, day of the week and season (winter is defined with months 12, 1 and 2, spring with months 3-5, summer with 6-8, and autumn with 9-11). All comparisons are shown in Fig.8a-b.

As expected, forecasting *on weekends and holidays is worse* than on working days (4.10% versus 3.45%), but the difference is smaller because of holiday error models (Fig. 8).

When comparing MAPE based on the day of the week, *the best results were obtained on Wednesday*, followed by Thursday and Tuesday. This fact is not surprising, because in the middle of the week energy load stabilizes the most as the weekend influence fades out. A more *surprising fact is that the largest error is obtained on Mondays* and not only across the weekend. The explanation for Monday's error is suggested as a consequence of week changes (vacations, work trips, etc.), as this change is expressed on Monday. This change is impossible to predict without more information about migrations; therefore, large errors occur.

Shape forecast analysis showed similar characteristics as the offset, with the expectation that the largest error is obtained on Friday, followed by Monday and Sunday. The reason for this can be found in the fact that loads on Monday and Friday are influenced by the weekend shape and that this influence is hard to model. Errors on Sundays can be the consequence of the nature of the weekend load, which fluctuates from week to week.

Forecast accuracy also varies across different seasons, which can also be seen in Fig.8a. Model accuracy is similar across the seasons except in winter, when the accuracy is the highest.



(a)

(b)

Fig. 8. Above figures show overall forecast MAPE (fig. 8a) and offset MAPE (fig. 8b) of depending on the sample variables. Left cluster of columns compares MAPE of the forecast depending on data variable D (gray bar represent result without holiday error model), middle one on the day of the week F and last cluster on the season. Horizontal line represents the MAPE of component forecast across all validation set.

The reason is not directly a result of the model, but is a combination of energy load composition and the MAPE metric: the substantial part of the load consists of *heating load* (load dependent on temperature), which is relatively much bigger in the winter. Heating load has smaller variance than other load components do; therefore, it can be forecast more accurately by including temperature as an input to a model. Because the metric MAPE is also relative to the absolute load, the relatively larger heating load, in comparison to the total load in the winter, has the consequence of a smaller MAPE error.

The most interesting seasonal results can be observed while finding the sources of errors in other seasons. The primary reason for *error in autumn and spring is bad shape forecasting, while in the summer bad average forecasting had greater influence on the final error*. The reason for bad shape forecasting across spring and autumn can be found in fast shape changing (heating is replaced by air conditioning in spring and vice-versa in autumn). Because of the low number of clusters, the model cannot adapt fast enough to shape changes, meaning the shape model output is always lagging behind the measured shape. Improvements for spring and autumn load forecasting should exist with great certainty, so this paper suggests more research of methods that include more historic data in the shape forecast.

A. Comparison to existing methods

The developed model was compared to LR, RLR [16], original GK clustering (with problem division and holiday error forecast) and SARIMA [3]. Both the LR and RLR methods created separate models for every day and every hour, and all LR, RLR, and SARIMA assume a quadratic dependency on the temperature. The original GK clustering was included to test proposed upgrades to clustering and to test a single model across whole input domain. In Fig. 9, MAPE values are gathered and shown together and for each component separately. Note: with RLR, LR and SARIMA, the first load profile is forecast, and then via equations (12, 13 and 16) components are extracted from the forecast. This is done for the purpose of detailed analysis and pinpointing weak spots of the developed model.

The developed model *improves overall forecast* and the forecast of offset and shape, but falls behind SARIMA in amplitude forecast. This confirms the contribution of load profile division, as division enabled more accurate forecasting of the offset, which is the component with the biggest impact on the final error. The proposed combined membership function also improved forecasting, as it can be seen that they improve the TS model forecast in all subproblems, in comparison to original GK (0.16% MAPE for average, 0.87% MAPE for shape 0.27% MAPE for amplitude and 0.23% MAPE overall). Th

A comparison across seasons is shown in Fig. 10. The developed model provides better results, especially in winter; in summer and spring, the developed method is on par with SARIMA, but for autumn SARIMA has the best results. The reason for this could be found in the already mentioned fact that in autumn the developed model always lags behind the actual profile (especially shape), while SARIMA incorporates those changes better. However, for winter, the capability of the TS model to describe non-linear dependency on temperature exceeds the SARIMA model results.

A comparison of different days of the week is shown in Fig. 11; the results of LR and RLR were omitted for clarity of the figure as they were already found to be considerably worse than those of other methods. The developed forecast model is the most accurate at the start of the week, while the SARIMA model is more accurate on the weekend, from Friday to Sunday (the original GK forecast is the least accurate of the three in most of the days of the week). The reason for the results could be that the weekend load does not considerably depend on the load of the previous day as the weekly change is taking place; therefore, the forecast based on the load of the previous week is more accurate (SARIMA). However, the load on Sunday already incorporates weekly load changes; therefore, the forecast for Monday is more accurate using the load on Sunday as an input (developed model).

Lastly, the holiday forecasts were compared between SARIMA, the original GK, the original GK with holiday error forecast, and the developed model. The results are gathered in TableVII. The fuzzy methods yield better forecast in comparison to SARIMA; the holiday error forecast improved the holiday forecast by 1.33% MAPE. The developed methods yield an additional $0.96\%\mathrm{MAPE}$ improvement over the GK with holiday error forecast. The results shows significant improvements using holiday error forecast.

TABLE VII
COMPARING MAPE ERROR ON HOLIDAYS

	SARIMA	GK	GK holiday	Developed
MAPE[%]	7.60	6.87	5.54	4.58

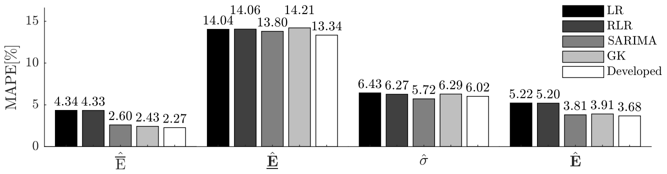


Fig. 9. Error MAPE comparison of different components between developed and existing methods

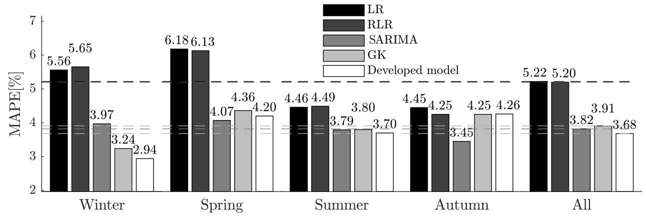


Fig. 10. Error MAPE comparison between different methods across different seasons and whole year.

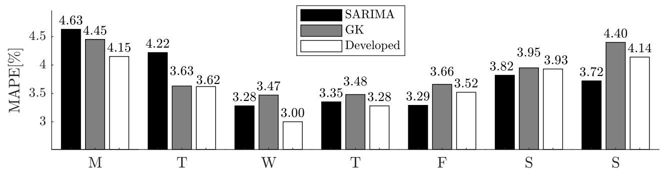


Fig. 11. Error MAPE comparison between SARIMA, GK and developed model on different day of the week.

VI. Conclusion

This paper presents solutions of the problem of STLF by separating it into three subproblems (offset, standard deviation, and shape), each of which was solved using an adaptive TS model defined accurately across the whole input space (rGK clustering and rWLMS algorithms were used with additional developed upgrades). For the average and amplitude subproblems, the paper proposes forecasting daily changes instead of forecasting values directly for better results. To overcome large errors on holidays, a model for each holiday was also created.

The findings of this paper are the following:

- The developed model improved load profile forecasting in comparisons to other methods by at least 0.13% MAPE.
- Splitting daily profiles mostly improved offset forecasting, which is the component with the most influence on errors.
- The TS model used for general forecasting successfully described non-linear dependencies.
- Holiday error forecasting was significantly improved; therefore, it is shown that other local phenomenon can be modeled in addition to the developed model.
- Using a single model valid across the whole input domain enabled parameter sharing across different areas, which resulted in good adaptability.
- The combined membership function improved the domain separation for the TS model, which also improves the forecast.
- The developed model does not use enough historical data; therefore, larger errors were in the autumn and spring.

Future work: A method for including more historical data in the model will be considered in order to improve spring and autumn forecasting, such as including elements of the similar-day method. Another algorithm worth including is automatic feature selection, such as minimal-redundancy-maximal-relevance (mRMR) [46]. Another area for research is to include more factors into our model (announced major events in the area, for example), in which big data approaches will be researched. Furthermore, a separate model for weekends can be introduced, which would be based on the previous weekend and not on the previous day.

References

[1] D. W. Bunn and E. D. Farmer, *Comparative models for electrical load forecasting*, 1st ed. New York: John Wiley and Sons Inc, 1985.

[2] G. Gross and F. Galiana, "Short-term load forecasting," *Proc. IEEE*, vol. 75, DOI 10.1109/PROC.1987.13927, no. 12, pp. 1558–1573, 1987.

[3] G. Box and G. Jenkins, *Time Series Analysis: Forecasting and Control*, rev. ed. Oakland: Holden-Day, cop., 1976.

[4] M. Cho, J. Hwang, and C. Chen, "Customer short term load forecasting by using ARIMA transfer function model," *International Conference on Energy Management and Power Delivery EMPD '95*, vol. 1, DOI 10.1109/EMPD.1995.500746, no. 95, pp. 317–322, 1995.

[5] N. Amjady, "Short-term hourly load forecasting using time-series modeling with peak load estimation capability," *IEEE Transactions on Power Systems*, vol. 16, DOI 10.1109/59.962429, no. 4, pp. 798–805, 2001.

[6] S.-j. Huang, S. Member, and K.-r. Shih, "Short-Term Load Forecasting Via ARMA Model Identification Including Non-Gaussian," *IEEE Transactions on Power Systems*, vol. 18, DOI 10.1109/TPWRS.2003.811010, no. 2, pp. 673–679, 2003.

[7] H. Hippert, C. Pedreira, and R. Souza, "Neural networks for short-term load forecasting: a review and evaluation," *IEEE Transactions on Power Systems*, vol. 16, DOI 10.1109/59.910780, no. 1, pp. 44–55, 2001.

[8] A. S. Khwaja, M. Naeem, A. Anpalagan, A. Venetsanopoulos, and B. Venkatesh, "Improved short-term load forecasting using bagged neural networks," *Electric Power Systems Research*, vol. 125, DOI 10.1016/j.epsr.2015.03.027, pp. 109–115, 2015.

[9] S. Kouhi and F. Keynia, "A new cascade NN based method to short-term load forecast in deregulated electricity market," *Energy Conversion and Management*, vol. 71, DOI 10.1016/j.enconman.2013.03.014, pp. 76–83, 2013.

[10] C. Cecati, J. Kolbusz, P. Siano, and B. M. Wilamowski, "A Novel RBF Training Algorithm for Short-Term Electric Load Forecasting and Comparative Studies," *IEEE Transactions on Industrial Electronics*, vol. 62, DOI 10.1109/TIE.2015.2424399, no. 10, pp. 6519–6529, 2015.

[11] N.-s. Pang and Y.-l. Shi, "Research on short-term load forecasting based on adaptive hybrid genetic optimization BP neural network algorithm," in *2008 International Conference on Management Science and Engineering 15th Annual Conference Proceedings*, DOI 10.1109/ICMSE.2008.4669113, no. 70572090, pp. 1563–1568, Long Beach, CA, 2008.

[12] S. H. Ling, F. H. F. Leung, H. K. Lam, Y. S. Lee, and P. K. S. Tam, "A novel genetic-algorithm-based neural network for short-term load forecasting," *IEEE Transactions on Industrial Electronics*, vol. 50, DOI 10.1109/TIE.2003.814869, no. 4, pp. 793–799, 2003.

[13] S. H. Ling, F. H. F. Leung, H. K. Lam, and P. K. S. Tam, "Short-term electric load forecasting based on a neural fuzzy network," *IEEE Transactions on Industrial Electronics*, vol. 50, DOI 10.1109/TIE.2003.819572, no. 6, pp. 1305–1316, 2003.

[14] W. Christiaanse, "Short-Term Load Forecasting Using General Exponential Smoothing," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-90, DOI 10.1109/TPAS.1971.293123, no. 2, pp. 900–911, 1971.

[15] H. Sheng, S. Member, J. Xiao, Y. Cheng, S. Member, and Q. Ni, "Short-Term Solar Power Forecasting Based on Weighted Gaussian Process Regression," *IEEE Transactions on Industrial Electronics*, DOI 10.1109/TIE.2017.2714127, 2017.

[16] Š. Kunstelj, M. Rejc, and M. Pantoš, "Kratkoročno napovedovanje porabe električne energije po regijah za območje Slovenije," *Elektrotehniški vestnik*, vol. 81, no. 4, pp. 222–228, 2014.

[17] Q. Mu, Y. Wu, X. Pan, L. Huang, and X. Li, "Short-term Load Forecasting Using Improved Similar Days Method," *APPEEC Conference 2010*, DOI 10.1109/APPEEC.2010.5448655, pp. 1–4, 2010.

[18] Y. Chen, P. Luh, C. Guan, Y. Zhao, L. Michel, M. Coolbeth, P. Friedland, and S. Rourke, "Short-Term Load Forecasting: Similar Day-Based Wavelet Neural Networks," *IEEE Transactions on Power Systems*, vol. Vol. 25, no. No. 1, pp. 3353–3358, 2010.

[19] M. Mohandes, "Support vector machines for short-term electrical load forecasting," *International Journal of Energy Research*, vol. 26, DOI 10.1002/er.787, no. 4, pp. 335–345, Mar. 2002.

[20] P. F. Pai and W. C. Hong, "Forecasting regional electricity load based on recurrent support vector machines with genetic algorithms," *Electric Power Systems Research*, vol. 74, DOI 10.1016/j.epsr.2005.01.006, no. 3, pp. 417–425, 2005.

[21] C. C. Cai and M. Wu, "Support vector machines with similar day's training sample application in short-term load forecasting," *3rd International Conference on Deregulation and Restructuring and Power Technologies, DRPT 2008*, DOI 10.1109/DRPT.2008.4523593, no. April, pp. 1221–1225, 2008.

[22] D.-X. Niu, Q. Wang, and J.-C. Li, "Short Term Load Forecasting Model Based on Support Vector Machine," in *Advances in Machine Learning and Cybernetics: 4th International Conference, ICMLC 2005*, D. S. Yeung, Z.-Q. Liu, X.-Z. Wang, and H. Yan, Eds., DOI 10.1007/11739685_92, pp. 880–888. Berlin, Heidelberg: Springer, 2006.

[23] A. M. Al-Kandari, S. A. Soliman, and M. E. El-Hawary, "Fuzzy short-term electric load forecasting," *International Journal of Electrical Power and Energy System*, vol. 26, DOI 10.1016/S0142-0615(03)00069-3, no. 2, pp. 111–122, 2004.

[24] A. Bakirtzis, J. Theocharis, S. Kiartzis, and K. Satsios, "Short term load forecasting using fuzzy neural networks," *IEEE Transactions on Power Systems*, vol. 10, DOI 10.1109/59.466494, no. 3, pp. 1518–1524, 1995.

[25] Y. Wei, J. Qiu, H.-K. Lam, and L. Wu, "Approaches to T-S Fuzzy-Affine-Model-Based Reliable Output Feedback Control for Nonlinear Ito Stochastic Systems," *IEEE Transactions on Fuzzy Systems*, vol. 6706, DOI 10.1109/TFUZZ.2016.2566810, no. c, pp. 1–1, 2016.

[26] Y. Wei, J. Qiu, and H. R. Karimi, "Fuzzy-Affine-Model-Based Memory Filter Design of Nonlinear Systems with Time-Varying Delay," *IEEE Transactions on Fuzzy Systems*, vol. 6706, DOI 10.1109/TFUZZ.2017.2686352, no. c, pp. 1–1, 2017.

[27] Y. Wei, J. Qiu, S. Member, H. R. Karimi, and S. Member, "Reliable Output Feedback Control of Discrete-Time Fuzzy Affine Systems With Actuator Faults," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 64, DOI 10.1109/TCSI.2016.2605685, no. 1, pp. 170–181, 2017.

[28] A. Kazemi, Amir Foroughi, and M. Hosseinzadeh, "A Multi-Level Fuzzy Linear Regression Model for Forecasting Industry Energy Demand of Iran," *Procedia - Social and Behavioral Sciences*, vol. 41, DOI 10.1016/j.sbspro.2012.04.039, pp. 342–348, 2012.

[29] T. Hong and P. Wang, "Fuzzy interaction regression for short term load forecasting," *Fuzzy Optimization and Decision Making*, vol. 13, DOI 10.1007/s10700-013-9166-9, no. 1, pp. 91–103, Mar. 2014.

[30] K.-B. Song, Y.-S. Baek, D. Hong, and G. Jang, "Short-Term Load Forecasting for the Holidays Using Fuzzy Linear Regression Method," *IEEE Transactions on Power Systems*, vol. 20, DOI 10.1109/TPWRS.2004.835632, no. 1, pp. 96–101, 2005.

[31] J. S. R. Jang, "ANFIS: Adaptive-Network-Based Fuzzy Inference System," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 23, DOI 10.1109/21.256541, no. 3, pp. 665–685, 1993.

[32] D. Dovžan and I. Škrjanc, "Recursive clustering based on a Gustafson-Kessel algorithm," *Evolving Systems*, vol. 2, DOI 10.1007/s12530-010-9025-7, no. 1, pp. 15–24, 2011.

[33] N. Kasabov and Qun Song, "DENFIS: dynamic evolving neural-fuzzy inference system and its application for time-series prediction," *IEEE Transactions on Fuzzy Systems*, vol. 10, DOI 10.1109/91.995117, no. 2, pp. 144–154, Apr. 2002.

[34] P. P. Angelov and D. P. Filev, "An approach to online identification of Takagi-Sugeno fuzzy models," *IEEE Trans. Syst. Man Cybern. Part B Cybern.*, vol. 34, no. 1, pp. 484–498, 2004.

[35] E. Lughofer and E. P. Klement, "FLEXFIS: A Variant for Incremental Learning of Takagi-Sugeno Fuzzy Systems," *Proceedings of The 2005 IEEE International Conference on Fuzzy Systems*, DOI 10.1109/FUZZY.2005.1452516, pp. 915–920, 2005.

[36] H. J. Rong, N. Sundararajan, G. B. Huang, and P. Saratchandran, "Sequential Adaptive Fuzzy Inference System (SAFIS) for nonlinear system identification and prediction," *Fuzzy Sets and Systems*, vol. 157, DOI 10.1016/j.fss.2005.12.011, no. 9, pp. 1260–1275, 2006.

[37] N. Kasabov, "Evolving fuzzy neural networks for supervised/unsupervised online knowledge-based learning," *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, vol. 31, DOI 10.1109/3477.969494, no. 6, pp. 902–918, 2001.

[38] P. Angelov, "Evolving Takagi-Sugeno Fuzzy Systems from Streaming Data (eTS+)," in *Evolving Intelligent Systems: Methodology and Applications*, pp. 21–50. Hoboken, NJ, USA: John Wiley & Sons, Inc., Apr. 2010.

[39] D. Dovžan, V. Logar, and I. Škrjanc, "Implementation of an Evolving Fuzzy Model (eFuMo) in a Monitoring System for a Waste-Water Treatment Process," *IEEE Transactions on Fuzzy Systems*, vol. 23, DOI 10.1109/TFUZZ.2014.2379252, no. 5, pp. 1761–1776, 2015.

[40] Y.-Y. Lin, J.-Y. Chang, and C.-T. Lin, "A TSK-Type-Based Self-Evolving Compensatory Interval Type-2 Fuzzy Neural Network (TSCIT2FNN) and Its Applications," *IEEE Transactions on Industrial Electronics*, vol. 61, DOI 10.1109/TIE.2013.2248332, no. 1, pp. 447–459, 2014.

[41] M. Manjunath, P. Singh, A. Mandal, and G. S. Parihar, "Consumer behaviour towards electricity- A field study," *Energy Procedia*, vol. 54, DOI 10.1016/j.egypro.2014.07.295, no. i, pp. 541–548, 2014.

[42] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-15, DOI 10.1109/TSMC.1985.6313399, no. 1, pp. 116–132, 1985.

[43] D. Gustafson and W. Kessel, "Fuzzy clustering with a fuzzy covariance matrix," in *1978 IEEE Conference on Decision and Control including the 17th Symposium on Adaptive Processes*, vol. 2, DOI 10.1109/CDC.1978.268028, no. 2, pp. 761–766. IEEE, Jan. 1978.

[44] R. J. Hathaway and J. C. Bezdek, "Switching Regression Models and Fuzzy Clustering," *IEEE Transactions on Fuzzy Systems*, vol. 1, DOI 10.1109/91.236552, no. 3, pp. 195–204, 1993.

[45] D. Dovžan and I. Škrjanc, "Possible use of evolving c-regression clustering for energy consumption profiles classification," in *2015 IEEE International Conference on Evolving and Adaptive Intelligent Systems (EAIS)*, DOI 10.1109/EAIS.2015.7368792, pp. 1–6. IEEE, Dec. 2015.

[46] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information: Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 27, DOI 10.1109/TPAMI.2005.159, no. 8, pp. 1226–1238, 2005.

Gregor Černe received the B.S. and M.S. degree in Control Systems and Computer Engineering from University of Ljubljana, Slovenia, in 2012 and 2016 respectively. He is currently working toward the Ph.D. degree as a young researcher in the same institution. His research interests include dynamic model identification, fuzzy modeling and deep neural architectures. Mr. Černe received multiple awards for his exceptional master thesis in 2016 (Prešeren Award and Bedjanič Award).

Dejan Dovžan received B.S., M.S. and Ph.D. degree in 2008, 2010 and 2013 respectively from the Faculty of Electrical Engineering, University of Ljubljana, Slovenia, where he is currently employed at the Laboratory for Autonomous and Mobile systems as an assistant professor. His main research interest are fuzzy model learning methods, recursive and evolving fuzzy model identification and model predictive control. For the exceptional Ph.D. thesis he received the Vodovnik award and Jožef Stefan Golden Emblem.

Igor Škrjanc received B.S., M.S. and Ph.D. degrees in electrical engineering, in 1988, 1991 and 1996, respectively, at the Faculty of Electrical and Computer Engineering, University of Ljubljana, Slovenia. He is currently a Full Professor with the same faculty and Head of Laboratory for Autonomous and Mobile Systems. His main research areas are adaptive, predictive, neuro-fuzzy and fuzzy adaptive control systems. He is Humboldt research fellow, research fellow of JSPS and Chair of Excellence at University Carlos III of Madrid. He also serves as an Associated Editor for multiple SCIE journals.