Residential Power Forecasting Using Load Identification and Graph Spectral Clustering

Chinthaka Dinesh, *Student Member, IEEE,* Stephen Makonin, *Senior Member, IEEE,* and Ivan V. Bajić, *Senior Member, IEEE*

***Abstract*—Forecasting energy or power usage is an important part of providing a stable supply of power to all customers on a power grid. We present a novel method that aims to forecast the power consumption of a single house, or a set of houses, based on non-intrusive load monitoring (NILM) and graph spectral clustering. In the proposed method, the aggregate power signal is decomposed into individual appliance signals and each appliance's power is forecasted separately. Then the total power forecast is formed by aggregating forecasted power levels of individual appliances. We use four publicly available datasets (REDD, RAE, AMPds2, *tracebase*) to test our forecasting method and report its accuracy. The results show that our method is more accurate compared to popular existing approaches such as autoregressive integrated moving average (ARIMA), similar profile load forecast (SPLE), artificial neural network (ANN), and recent NILM-based forecasting.**

***Index Terms*—power forecasting, load disaggregation, non-intrusive load monitoring (NILM), spectral clustering, smart grid**

I. Introduction

ACCURATE power demand forecasting is important for maintaining a stable power grid. With advance warning of demand surges, energy providers would be able to better plan their power generation and/or perform other measures such as peak shaving or load shifting [1], [2]. Various forecasting methods have been proposed based on extrapolation [3], [4], Kalman filtering [5]–[7], fuzzy logic [6], [8]–[10], autoregressive integrated moving average (ARIMA) models [11]–[14], artificial neural networks (ANN) [6], [11], [15], and similar profiles load forecast (SPLF) [16], [17]. All these methods attempt to forecast the aggregate power (all loads combined) directly by relying on the temporal dependence of the aggregate power signal. However, we note that stronger temporal dependence may exist in power signals of individual appliances. This is easily seen in the case of cyclical appliances such as refrigerators, which turn ON and OFF roughly periodically. When the power signals of different appliances are aggregated, such temporal dependence may be disrupted, hence forecasting the aggregate power may be more difficult than forecasting the power usage of individual appliances.

Recently, a few residential power forecasting methods have been proposed based on the principle of disaggregating individual appliances first [18], [19], using non-intrusive load monitoring (NILM)[1]. Here, the aggregate power signal is first decomposed into individual appliance signals, then each appliance's power is forecasted separately, and finally the total power forecast is formed by aggregating forecasted power levels of individual appliances. These NILM-based methods give better forecasting accuracy than the traditional methods that operate on the aggregate signal directly [18]. However, both [18], [19] assume that the behaviour of an individual appliance is uncorrelated with the behaviours of other appliances, which is not the case in practice [26]. Therefore, in this paper, we propose a NILM-based residential power forecasting method that incorporates correlations of appliances behaviours in terms of their state (ON/OFF) durations. Moreover, since we have individual appliance level forecasting, the proposed forecasting method is more suitable for *real-time demand side management* that relies on appliance-level information to evaluate the amount of energy that can be saved. A discussion of state duration probabilities is presented in Section II, followed by the proposed method in Section III. In Section IV, we show through extensive evaluation that the proposed forecasting method offers higher accuracy compared to several existing methods based on NILM [19], ANN [11], ARIMA [11], and SPLF [17]. Finally, conclusions are presented in Section V.

II. State Duration Probabilities

Some appliances, such as traditional refrigerators, have only two states: ON and OFF. Others, such as washing machines, may have multiple states (with continuous varying power levels), one of which is an OFF state, and the others represent different power levels used in different modes of operation. To simplify our model, we decompose continuous-state and/or multi-state appliances into multiple two-state (ON/OFF) appliances as follows. Continuously-varying power profiles are quantized into multiple states following [27]. Then, if there are N states in the resulting multi-state profile (with one of the states being the OFF state), we represent such an appliance by N-1 virtual two-state appliances, where each of these two-state appliances consists of an OFF state and one of the N-1 remaining states. In the remainder of the text, the term "appliance" refers to a two-state (ON/OFF) appliance, whether it really is a two-state appliance, or a virtual two-state appliance obtained from a multi-state appliance.

From experience we know that some appliances have certain expected ON durations. For example, a washing machine would often be ON for 30-60 minutes, while a water kettle would often be ON for 2-5 minutes. In this work, we utilize such ON/OFF duration patterns to forecast future power demand. For a given appliance a_i, the duration of the ON state

The authors are with the School of Engineering Science, Simon Fraser University, Burnaby, BC, Canada, e-mail: hchintha@sfu.ca, smakonin@sfu.ca, ibajic@ensc.sfu.ca. This work was supported by IC-IMPACTS.

[1]NILM is a technique to determine how much power each appliance is using from an aggregate signal [20]. In the literature, there are number of NILM methods for residential appliance identification [21]–[25].

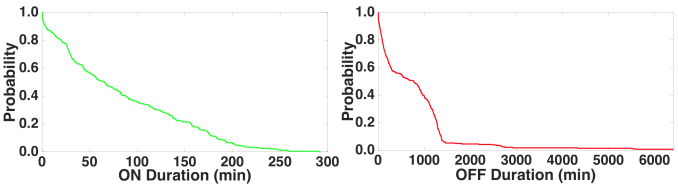

Fig. 1. Examples of ON/OFF-state duration probabilities.

is denoted by $T_{\{a_i\}}$ and represents the time elapsed since the appliance was turned ON. A set of appliances S is said to be ON if all the appliance in the set are ON. The duration of the ON state of set S is denoted by T_S. If different appliances in S turn ON at different times, then T_S is measured from the time instant when the last appliance in the set is turned ON.

State durations are random quantities, so we describe them with a probability distribution. Specifically, we compute the following probabilities:

$$P[T_S \geq t] = \frac{n_S^t}{n_S}, \tag{1}$$

where n_S is the number of times within the training set that all appliances in the set S were ON, and n_S^t is the number of times that all of them were ON for at least t time units.

Set S of appliances is said to be OFF if at least one appliance in the set is OFF. The OFF-state duration is characterized analogously to the ON state:

$$P[\overline{T}_S \geq t] = \frac{\overline{n}_S^t}{\overline{n}_S}, \tag{2}$$

where $\overline{T}_S$ represents the time elapsed since the set S entered the OFF state, $\overline{n}_S$ is the number of times within the training set that the set S was OFF, and $\overline{n}_S^t$ is the number of times that S was OFF for at least t time units. If different appliances in S turn OFF at different times, then $\overline{T}_S$ is measured from the time instant when the first appliance in the set is turned OFF.

Fig. 1 shows the ON and OFF duration probability distributions for the (LCD TV, lamp) pair. Here, fifty-day profiles in *tracebase* [28] were used to compute these probabilities.

III. Proposed Method

Let t denote the current time, t_p denote a previous time ($t_p < t$), and t_f denote a future time ($t_f > t$) for which we want to make a prediction. Also, let $d_c = t - t_p$ be the time elapsed between t_p, and current time t, and let $d_f = t_f - t_p$ be the time that will have elapsed between t_p and t_f. The proposed residential power forecasting method has three main stages: appliance state identification, ON-set prediction, and aggregation. Each stage is described below.

Appliance State Identification: First, ON/OFF appliances are identified at the current time t from the given aggregated power signal of a house. Then for each appliance, the most recent time instant when the current state of the appliance started is identified. NILM methods can identify appliances that are/were ON/OFF at any given time up to the current time t. Therefore, our recent NILM method [24] is used to identify the most recent time instant when the current state of the appliance started. The identified appliance states, together with state duration probabilities (Section II), allow us to make predictions about which appliances will be ON at time t_f.

ON-Set Prediction: The ON-set is the set of all appliances that are ON at a particular time. We predict the ON-set at time t_f with the help of spectral clustering [29], which has three main steps: graph construction, spectral representation, and clustering. First, we construct a graph of appliances and choose distances between pairs of appliances as functions of their state duration probabilities.

Suppose a pair of appliances $\{a_i, a_j\}$ is currently ON. Based on the previously identified appliance states, we can determine the most recent time instant at which the pair entered the ON state. Let t_p be that time instant. Now consider the probability of these two appliances remaining in the ON state until the future time t_f or beyond. The conditional probability of this event, given that these two appliances are currently ON, is $P[(T_{\{a_i,a_j\}} \geq d_f)|(T_{\{a_i,a_j\}} \geq d_c)]$. Using Bayes' theorem,

$$P[(T_{\{a_i,a_j\}} \geq d_f)|(T_{\{a_i,a_j\}} \geq d_c)] = \frac{P[(T_{\{a_i,a_j\}} \geq d_c)|(T_{\{a_i,a_j\}} \geq d_f)] \cdot P[(T_{\{a_i,a_j\}} \geq d_f)]}{P[(T_{\{a_i,a_j\}} \geq d_c)]}. \tag{3}$$

From the definition of d_c and d_f, we have $d_f \geq d_c$. Hence

$$P[(T_{\{a_i,a_j\}} \geq d_c)|(T_{\{a_i,a_j\}} \geq d_f)] = 1,$$

so the required conditional probability becomes

$$P[(T_{\{a_i,a_j\}} \geq d_f)|(T_{\{a_i,a_j\}} \geq d_c)] = \frac{P[(T_{\{a_i,a_j\}} \geq d_f)]}{P[(T_{\{a_i,a_j\}} \geq d_c)]}. \tag{4}$$

The two terms on the right hand side of (4) are found from the state duration probabilities (Section II). Finally, the distance $\mathbf{D}[i,j]$ between appliances a_i and a_j at time t_f is defined as

$$\mathbf{D}[i,j] = 1 - P[(T_{\{a_i,a_j\}} \geq d_f)|(T_{\{a_i,a_j\}} \geq d_c)], \tag{5}$$

so that large conditional probability means small distance.

Now consider a pair of appliances $\{a_i, a_j\}$ that is currently OFF, which, according to our definition in Section II, means that either a_i or a_j or both are currently OFF. Based on the previously identified appliance states, we can also determine the most recent time instant (t_p) at which the pair entered the OFF state. Using the same reasoning as above, we can compute the conditional probability of the pair staying in the OFF state until a future time t_f given that it is currently in the OFF state:

$$P[(\overline{T}_{\{a_i,a_j\}} \geq d_f)|(\overline{T}_{\{a_i,a_j\}} \geq d_c)] = \frac{P[(\overline{T}_{\{a_i,a_j\}} \geq d_f)]}{P[(\overline{T}_{\{a_i,a_j\}} \geq d_c)]}. \tag{6}$$

The probabilities on the right hand side of (6) are found from the state duration probabilities (Section II). Then the distance $\mathbf{D}[i,j]$ between appliances a_i and a_j at time t_f is defined as

$$\mathbf{D}[i,j] = P[(\overline{T}_{\{a_i,a_j\}} \geq d_f)|(\overline{T}_{\{a_i,a_j\}} \geq d_c)]. \tag{7}$$

After obtaining $\mathbf{D}[i,j]$ for each appliance pair, a fully connected undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, called an *appliance graph* is formed, where $\mathcal{V}$ is the set of all appliances (vertices of the graph) and $\mathcal{E}$ is the set of edges. The weights of edges

are represented by the *affinity matrix* $\mathbf{A}$, where the weight of the edge connecting a_i and a_j is given by [29]:

$$\mathbf{A}[i,j] = \mathbf{A}[j,i] = \begin{cases} \exp\left\{-\frac{\mathbf{D}[i,j]}{2\sigma^2}\right\}, & \text{if } i \neq j \\ 0, & \text{otherwise,} \end{cases} \tag{8}$$

where σ is an affinity scaling factor. The intuition between affinities is that they reflect correlation of appliance behaviours. Similarly to [30], we select σ as the standard deviation of the values in $\mathbf{D}$. An example of an appliance graph is shown in Fig. 2, with affinities between appliances as edge weights, which can be computed from (8).

After constructing the appliance graph at time t_f, spectral representation of each appliance is obtained from the Laplacian matrix $\mathbf{L}$. First, $\mathbf{L}$ is computed as [29]:

$$\mathbf{L} = \mathbf{W}^{-1/2}\mathbf{A}\mathbf{W}^{-1/2}, \tag{9}$$

where $\mathbf{W}$ is a diagonal matrix whose diagonal entries are summations of the corresponding columns of $\mathbf{A}$: $\mathbf{W}[i,i] = \sum_{j=1}^{n} \mathbf{A}[j,i]$. Then, eigenvectors $\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_m$ corresponding to the m largest eigenvalues of $\mathbf{L}$ are used to construct a matrix $\mathbf{X}$ as $\mathbf{X}[:,i] = \mathbf{v}_i$, where $\mathbf{X}[:,i]$ is the i-th column of the matrix $\mathbf{X}$. Finally, matrix $\mathbf{Y}$ is defined as

$$\mathbf{Y}[i,:] = \frac{\mathbf{X}[i,:]}{\|\mathbf{X}[i,:]\|_2}, \tag{10}$$

where $\mathbf{Y}[i,:]$ is the i-th row of the matrix $\mathbf{Y}$ and $\|\cdot\|_2$ is the Euclidean norm. Matrix $\mathbf{Y}$ is a spectral representation of the appliance graph, where appliance a_i is represented as a point in $\mathbb{R}^m$ by the i-th row of $\mathbf{Y}$.

The rows of $\mathbf{Y}$ are then clustered using K-means clustering [31]. The number of clusters (K) is determined based on eigenvalue differences of the matrix $\mathbf{L}$ as follows [32]:

$$K = \arg\max_k(|\lambda_k - \lambda_{k+1}|), \tag{11}$$

where λ_k denotes the k-th largest eigenvalue of $\mathbf{L}$. Upon clustering, the j-th cluster represents appliance set S_j. For each of the K clusters, the average Euclidean distance AED_j of the vectors associated with S_j is

$$AED_j = \frac{1}{N_j}\sum_{a_i \in S_j} \|\mathbf{Y}[i,:] - \mathbf{c}_j\|_2. \tag{12}$$

where N_j be the number of appliances in S_j and $\mathbf{c}_j$ is the centroid of the vectors representing those appliances: $\mathbf{c}_j = \frac{1}{N_j}\sum_{a_i \in S_j} \mathbf{Y}[i,:]$. As discussed in [24], the value of AED_j is inversely related to the joint probability of appliances in S_j being ON at time t_f. However, for clusters that contain a single appliance, $AED_j = 0$. Hence, if there are any singleton clusters after clustering (say $S_j = \{a_k\}$) , we extend S_j with a dummy appliance $a_{k'}$ (so that $S_j = \{a_k, a_{k'}\}$) with $\mathbf{D}[k,k']$ at time t_f computed as

$$\mathbf{D}[k,k'] = \begin{cases} 1 - P[(T_{\{a_k\}} \geq d_f)|(T_{\{a_k\}} \geq d_c)], & \text{if } a_k \text{ is ON at } t \\ P[(\overline{T}_{\{a_k\}} \geq d_f)|(\overline{T}_{\{a_k\}} \geq d_c)], & \text{if } a_k \text{ is OFF at } t. \end{cases} \tag{13}$$

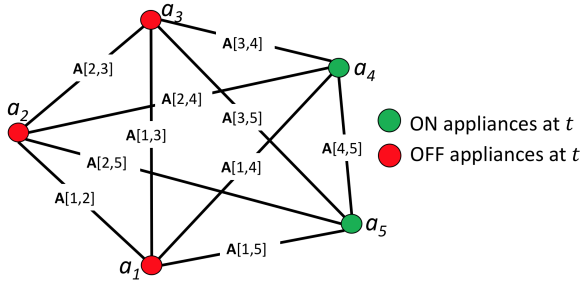

Fig. 2. An example of an appliance graph.

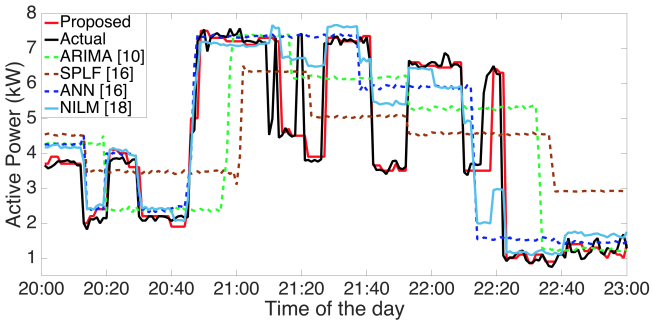

Fig. 3. A sample forecasted profile for house 1 in the REDD dataset.

Moreover, $\mathbf{D}[i,k'] = \mathbf{D}[i,k]$ and $\mathbf{D}[k',i] = \mathbf{D}[k,i]$ for $i \neq k$. After adding dummy appliances to singleton clusters, a new graph is constructed, its spectral representation is obtained, and new AED_j's are computed. The appliance set with the smallest AED_j is predicted to be ON at t_f.

Aggregation: The average power level of each appliance in this set is used as its forecasted power level at time t_f. Here, the average power level of each appliance is obtained similar to [23]. The sum of the forecasted individual appliance power levels is considered as the total forecasted power at time t_f. For a group of houses, the sum of the forecasted powers for each house is the total forecasted power demand of that group.

IV. Experimental Results

We compare the proposed method with four other forecasting methods based on NILM [19], ANN [11], ARIMA [11], and SPLF [17]. All methods were implemented in MATLAB R2015b on a 2.2 GHz MacBook Pro with Intel core i7 processor and 16GB memory. No special optimization was performed for any of the methods. Following [17], the Mean Absolute Percentage Error (MAPE) and the Root Mean Square Error (RMSE) are used as performance metrics.

We conducted three case studies using the data from four public datasets: the Reference Energy Disaggregation Data Set (REDD) [33], the Rainforest Automation Energy Dataset (RAE) [34], the Almanac of Minutely Power dataset version 2 (AMPds2) [35], and *tracebase* [28]. REDD, RAE, and *tracebase* data was converted to 1-minute intervals to match the data in AMPds2.

Case Study 1: In this case study, all six houses from the REDD dataset and one house from the RAE dataset were used. From the REDD dataset, the first 26 days of active power profiles were used for the training and the next 30 days were used for testing. Here, "training" means computing ON/OFF-state duration probabilities (1)-(2) in the proposed method, determining ON/OFF duration patterns of individual appliance

TABLE I
FORECASTING ACCURACY AND AVERAGE EXECUTION TIME OF EACH HOUSE IN REDD AND REA DATASETS FOR A 180-MINUTE AHEAD FORECAST

House Name	Forecasting accuracy in terms of MAPE (%) and RMSE (kW) respectively										Average execution time (s)				
	Proposed		NILM [19]		ANN [11]		ARIMA [11]		SPLF [17]		Proposed	NILM [19]	ANN [11]	ARIMA [11]	SPLF [17]
REDD1	**4.69**	**0.57**	8.14	1.08	9.21	1.17	9.63	1.29	10.14	1.48	1.02	0.91	0.84	3.72	0.93
REDD2	**4.28**	**0.51**	7.87	1.10	9.51	1.24	9.39	1.27	9.42	1.21	1.03	0.93	0.78	3.11	1.08
REDD3	**6.34**	**0.71**	14.72	1.41	16.48	1.65	17.71	1.69	18.48	1.52	1.17	1.01	0.86	2.99	1.07
REDD4	**6.16**	**0.68**	12.98	1.31	15.92	1.62	16.32	1.63	18.89	1.66	1.06	0.95	0.79	3.01	0.92
REDD5	**8.04**	**0.72**	15.48	1.43	19.53	1.79	19.46	1.82	20.64	2.01	1.12	0.97	0.91	3.04	1.13
REDD6	**7.24**	**0.69**	13.18	1.46	19.98	1.85	21.38	1.92	20.57	1.98	1.09	0.91	0.86	2.97	1.08
REA	**4.32**	**0.55**	7.92	1.02	9.08	1.12	8.99	1.21	9.36	1.17	1.03	0.87	0.81	3.18	1.05

TABLE II
THE MAPE(%)/RMSE(KW) FOR A 180-MINUTE AHEAD POWER FORECASTING OF THE HOUSE IN AMPDS2 DATASET

Season	Proposed		NILM [19]		ANN [11]		ARIMA [11]		SPLF [17]	
	Context-free	Seasonal	Context-free	Seasonal	Context-free	Seasonal	Context-free	Seasonal	Context-free	Seasonal
Winter	4.21/0.62	**3.62/0.51**	8.56/1.21	8.01/1.17	10.5/1.32	9.97/1.24	11.4/1.43	10.9/1.34	11.3/1.39	10.7/1.34
Spring	3.97/0.56	**3.58/0.44**	7.97/1.15	7.21/1.07	9.51/1.23	9.04/1.15	9.48/1.37	9.02/1.23	10.9/1.35	10.4/1.27
Summer	3.69/0.45	**3.01/0.37**	9.15/1.21	8.13/1.12	11.5/1.38	10.8/1.24	11.8/1.33	11.1/1.29	12.5/1.37	12.0/1.31
Fall	3.93/0.54	**3.41/0.42**	8.75/1.16	7.91/1.04	10.1/1.37	9.7/1.21	12.1/1.48	11.5/1.33	12.2/1.34	11.7/1.29

TABLE III
FORECASTING ACCURACY AND AVERAGE EXECUTION TIME FOR THE AGGREGATED POWER FORECASTING OF 400 HOUSES IN THE CASE STUDY 3

	Proposed	NILM [19]	Aggregating forecast			Forecasting aggregate		
			ANN [11]	ARIMA [11]	SPLF [17]	ANN [11]	ARIMA [11]	SPLF [17]
MAPE (%)	**1.87**	4.52	5.27	6.08	5.35	5.82	6.85	5.78
RMSE (kW)	**37**	81	99	127	117	124	138	123
Average execution time (s)	1.53	1.37	**0.85**	3.47	1.68	0.83	3.31	1.63

for the method in [19], and obtaining model parameters for the two methods in [11]. In [17], the training set is the search space of similar profiles. From the RAE dataset, the first 25 days were used for training, and the next 38 days for testing.

Power consumption for each house in REDD was predicted 180 minutes ahead in 1-minute steps using the proposed method as well as the methods in [11], [17], [19]. The MAPE and RMSE results are shown in Table I, with the best results indicated in bold. As seen in the table, the proposed method consistently outperforms others by a large margin, with MAPE reduced by 42-67% and RMSE reduced by 46-65%. A sample forecasted power profile for house 1 is shown in Fig. 3, where we can see that the proposed method tends to perform fairly accurate predictions. NILM [19], ANN [11], ARIMA [11], and SPLF [17] are generally able to predict upward and downward swings in power consumption, but with much less accurate predictions of power levels and with some phase offset. The average execution time per sample to predict 180 minutes ahead is also shown in Table I. As seen in the results, ANN-based method is the fastest and NILM-based method is the second fastest, followed closely by SPLF and proposed, while ARIMA-based method is about three times slower.

Case Study 2: In the second case study, we evaluate the effectiveness of the forecasting methods on the AMPds2 dataset. This dataset includes two years of data for a single house. We use the first year of data for the training and the second year for testing. In this case, we have several choices: we can create one model trained over the whole year, which we call *context-free* model, or produce four models by training over each season (winter, spring, summer, fall) separately, which we call *seasonal context-based* models.

The 180-minute ahead (in one minute steps) forecasting accuracy of the power consumptions in each season from April 2013 to March 2014 in the AMPds2 house are presented in Table II in terms of MAPE and RMSE. Again, we compare against the methods in [11], [17], [19]. As shown in Table II, the MAPE and RMSE of the proposed method are significantly better than those of other methods, with MAPE and RMSE reduced by 50-71% and 48-63%, respectively. Further, the seasonal context helps improve the forecasting accuracy of all five methods, but the seasonal-context version of the proposed method is the most accurate one, with MAPE and RMSE reduced by 50-75% and 56-72%, respectively, compared to other methods.

Case Study 3: Finally, we evaluate the performance of aggregated power forecasting for a large set of houses. Since REDD, RAE and AMPds2 datasets contain the data for relatively few houses, we created a large set of "virtual" houses from *tracebase* manually. Most of the appliances selected from *tracebase* have more than 120 days worth of power profiles. The first 50 days were used for the training and the rest were used for testing on virtual houses. For each virtual house, 12 appliances were chosen randomly from the 20 appliances[2]. Then, from each chosen appliance, a power profile window of 10 consecutive days was selected randomly to represent 10-day's usage of the given house. This way, 10-days usage profiles of 400 virtual houses were generated.

The 180-minute ahead (in one minute steps) forecasting accuracy of the aggregated power consumptions of all 400 houses is presented in Table III in terms of MAPE and RMSE. We show two sets of results for the methods in [11], [17]: forecasting each house and aggregating the forecast ("aggregating forecast") and forecasting the aggregate directly ("forecasting aggregate"). As seen in the table, the proposed method significantly outperforms the other methods in each

[2]Single state: 60 W and 100 W lamps, 1800 W and 2000 W water kettles, microwave oven, toaster, iron, video projector; continuous varying: LCD television, CRT television, CRT monitor, TFT monitor, remote desktop, desktop computer; multiple states: cooking stove, refrigerator, washing machine, dish washer, laundry dryer, washer dryer combo.

case, with MAPE and RMSE reduced by 59% and 54%, respectively, compared to the best alternative, which is [19]. The average execution time[3] per sample to predict 180 minutes ahead is also shown in Table III. As seen in the results, although ANN-based method [11] is the fastest, the proposed method has a practically viable average execution time.

V. Conclusions

A novel power forecasting method that uses appliance state identification and graph spectral clustering was presented. Once the current and previous appliance states are identified, they are used to perform the power forecast for each appliance, and the results are aggregated to provide the total power forecast. The proposed method was compared against four forecasting methods from the literature, showing superior performance in each case. The advantage of the proposed method comes from the following two aspects: (1) disaggregating the total power signal into individual appliances, which may be easier to forecast, and (2) modelling joint appliance behaviour via an appliance graph. In the future, we plan to incorporate time-of-day context in the proposed method to improve the forecasting performance.

References

[1] V. Zdraveski, M. Todorovski, D. Trajanov, and L. Kocarev, "Dynamic load balancing and reactive power compensation switch embedded in power meters," *IEEE Trans. Circuits Syst. II: Exp. Briefs*, vol. 64, no. 4, pp. 422–426, 2017.

[2] D. Karimipour and F. R. Salmasi, "Stability analysis of ac microgrids with constant power loads based on popov's absolute stability criterion," *IEEE Trans. Circuits Syst. II: Exp. Briefs*, vol. 62, no. 7, pp. 696–700, July 2015.

[3] D. Luo and H. He, "A shape similarity criterion based curve fitting algorithm and its application in ultra-short-term load forecasting," *Power System Technology*, vol. 31, no. 21, pp. 81–84, 2007.

[4] Z. Yang, G. Tang, Y. Song, and R. Cao, "Improved cluster analysis based ultra-short term load forecasting method," *Automation of Electric Power Systems*, vol. 24, p. 023, 2005.

[5] D. J. Trudnowski, W. L. McReynolds, and J. M. Johnson, "Real-time very short-term load prediction for power-system automatic generation control," *IEEE Trans. Control Syst. Technol.*, vol. 9, no. 2, pp. 254–260, 2001.

[6] C. Guan, P. B. Luh, L. D. Michel, and Z. Chi, "Hybrid kalman filters for very short-term load forecasting and prediction interval estimation," *IEEE Trans. Power Syst.*, vol. 28, no. 4, pp. 3806–3817, 2013.

[7] C. Lynch, M. J. O'Mahony, and R. A. Guinee, "Electrical load forecasting using an expanded kalman filter bank methodology," *IFAC-PapersOnLine*, vol. 49, no. 25, pp. 358–365, 2016.

[8] L. C. M. de Andrade and I. N. da Silva, "Very short-term load forecasting using a hybrid neuro-fuzzy approach," in *Eleventh Brazilian Symposium on Neural Networks (SBRN)*, 2010, pp. 115–120.

[9] A. Tale, A. S. Gusain, J. Baguli, R. Sheikh, and A. Badar, "Study of load forecasting techniques using fuzzy logic," *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, vol. 6, no. 2, 2017.

[10] H. H. Çevik and M. Çunkas, "A fuzzy logic based short term load forecast for the holidays," *International Journal of Machine Learning and Computing*, vol. 6, no. 1, p. 57, 2016.

[11] A. Marinescu, C. Harris, I. Dusparic, S. Clarke, and V. Cahill, "Residential electrical demand forecasting in very small scale: An evaluation of forecasting methods," in *International Workshop on Software Engineering Challenges for the Smart Grid*, 2013, pp. 25–32.

[3]Since there are 400 houses in case study 3, for the proposed method, [19], and "aggregating forecast" version of [11], [17], we need to calculate the forcasted power level for each house separately and obtain their sum for the total forcasted power. The forecast for each house can be performed in parallel. Therefore, we measure the maximum execution time for a given prediction (180 minutes ahead) across all 400 houses and calculate the average of these maximuma over all predictions. This average is reported in the table.

[12] B. Stephen, X. Tang, P. Harvey, S. Galloway, and K. Jennett, "Incorporating practice theory in sub-profile models for short term aggregated residential load forecasting," *IEEE Trans. Smart Grid*, vol. 8, no. 4, pp. 1591–1598, 2017.

[13] D. Alberg and M. Last, "Short-term load forecasting in smart meters with sliding window-based arima algorithms," *Vietnam Journal of Computer Science*, Jun 2018.

[14] T. D. Ha, F. M. Bianchi, and R. Olsson, "Local short term electricity load forecasting: Automatic approaches," in *International Joint Conference on Neural Networks*, May 2017, pp. 4267–4274.

[15] W. Kong, Z. Y. Dong, D. J. Hill, F. Luo, and Y. Xu, "Short-term residential load forecasting based on resident behaviour learning," *IEEE Trans. Power Syst.*, vol. 33, no. 1, pp. 1087–1088, 2018.

[16] E. Paparoditis and T. Sapatinas, "Short-term load forecasting: the similar shape functional time-series predictor," *IEEE Trans. Power Syst.*, vol. 28, no. 4, pp. 3818–3825, 2013.

[17] M. Tucci, E. Crisostomi, G. Giunta, and M. Raugi, "A multi-objective method for short-term load forecasting in european countries," *IEEE Trans. Power Syst.*, vol. 31, no. 5, pp. 3537–3547, 2016.

[18] M. Wurm and V. C. Coroamă, "Grid-level short-term load forecasting based on disaggregated smart meter data," *Computer Science-Research and Development*, vol. 33, no. 1-2, pp. 265–266, 2018.

[19] S. Welikala, C. Dinesh, M. P. B. Ekanayake, R. I. Godaliyadda, and J. Ekanayake, "Incorporating appliance usage patterns for non-intrusive load monitoring and load forecasting," *IEEE Trans. Smart Grid*, 2018, to appear.

[20] G. W. Hart, "Nonintrusive appliance load monitoring," *Proc. IEEE*, vol. 80, no. 12, pp. 1870–1891, Dec. 1992.

[21] M. Z. A. Bhotto, S. Makonin, and I. V. Bajić, "Load disaggregation based on aided linear integer programming," *IEEE Trans. Circuits Syst. II: Exp. Briefs*, vol. 64, no. 7, pp. 792–796, 2017.

[22] K. Basu, V. Debusschere, S. Bacha, U. Maulik, and S. Bondyopadhyay, "Non intrusive load monitoring: A temporal multi-label classification approach," *IEEE Trans. Ind. Informat.*, vol. 11, no. 4, pp. 689–698, 2015.

[23] C. Dinesh, B. W. Nettasinghe, R. I. Godaliyadda, M. P. B. Ekanayake, J. Ekanayake, and J. V. Wijayakulasooriya, "Residential appliance identification based on spectral information of low frequency smart meter measurements," *IEEE Trans. Smart Grid*, vol. 7, no. 6, pp. 2781–2792, Nov. 2016.

[24] C. Dinesh, S. Makonin, and I. V. Bajić, "Incorporating time-of-day usage patterns into non-intrusive load monitoring," in *Proc. IEEE GlobalSIP'17*, 2017, pp. 1–5.

[25] K. He, L. Stankovic, J. Liao, and V. Stankovic, "Non-intrusive load disaggregation using graph signal processing," *IEEE Trans. Smart Grid*, vol. 9, no. 3, pp. 1739–1747, May 2018.

[26] K. Basu, V. Debusschere, and S. Bacha, "Appliance usage prediction using a time series based classification approach," in *IECON*, 2012, pp. 1217–1222.

[27] S. Makonin, F. Popowich, I. V. Bajić, B. Gill, and L. Bartram, "Exploiting HMM sparsity to perform online real-time nonintrusive load monitoring," *IEEE Trans. Smart Grid*, vol. 7, no. 6, pp. 2575–2585, 2016.

[28] A. Reinhardt, P. Baumann, D. Burgstahler, M. Hollick, H. Chonov, M. Werner, and R. Steinmetz, "On the accuracy of appliance identification based on distributed load metering data," in *Proc. IEEE SustainIT*, Oct. 2012, pp. 1–9.

[29] A. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Proc. NIPS'02*, 2002, pp. 849–856.

[30] W. M. Song, T. D. Matteo, and T. Aste, "Hierarchical information clustering by means of topologically embedded graphs," *PLoS One*, vol. 7, no. 3, p. e31929, 2012.

[31] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means clustering algorithm: analysis and implementation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 881–892, 2002.

[32] U. V. Luxburg, "A tutorial on spectral clustering," *Journal of Statistics and Computing*, vol. 17, pp. 395–416, 2007.

[33] J. Kolter and M. Johnson, "REDD: A public data set for energy disaggregation research," in *Proc. ACM SustKDD*, 2011.

[34] S. Makonin, Z. J. Wang, and C. Tumpach, "RAE: The rainforest automation energy dataset for smart grid meter data analysis," *Data*, vol. 3, no. 1, 2018.

[35] S. Makonin, B. Ellert, I. Bajić, and F. Popowich, "Electricity, water, and natural gas consumption of a residential house in canada from 2012 to 2014," *Scientific Data*, vol. 3, pp. 1–12, 2016.