Residential Electrical Load Model based on Mixture Model Clustering and Markov Models

Wouter Labeeuw, *Graduate Student Member, IEEE*, Geert Deconinck, *Senior Member, IEEE*

***Abstract*—Detailed large scale simulations require a lot of data. Residential electrical load profiles are well protected by privacy laws. Representative residential electrical load generators get around the privacy problem and allow for Monte Carlo simulations. A top-down model of the residential electrical load, based on a dataset of over 1300 load profiles, is presented in this paper. The load profiles are clustered by a Mixed Model to group similar ones. Within the group, a behaviour model is constructed with a Markov model. The states of the Markov models are based on the probability distribution of the electrical power. A second Markov model is created to randomize the behaviour. A load profile is created by first performing a random-walking the Markov models to get a sequence of states. The inverse of the probability distribution of the electrical power is used to translate the resulting states into electrical power.**

***Index Terms*—Data analysis, Markov Models, Clustering, Statistical distributions**

I. INTRODUCTION

The knowledge of consumer electricity consumption is essential for the development of smart grid integration strategies such as integration of electrical vehicles and distributed generation. Studies tend to fall back on the aggregated data when no detailed data is available. The use of aggregated data is not a problem if the focus is on aggregated results. Examples of this are total electricity demand [1], discrete load bands [2] and average load profiles [3]. However, when detail is important, data can not be aggregated. Valid electricity profiles of households are required when simulating voltage problems due to electric vehicle charging at distribution level [4], when managing a micro-grid with photovoltaic (PV) systems [5] or when estimating the potential for battery storage in a distribution grid [6].

Residential electricity consumer data is well protected in Europe due to privacy concerns [7], [8]. Only a few companies monitor electricity consumption at residential level and they are not keen on sharing these load profiles. In stead, they provide synthetic load profiles that aggregate the consumption of a wide range of electricity consumers. Individual peaks and information about the load factors, amongst others, are lost due to the aggregation. Load profile generators are a solution to the privacy issues and allow for Monte Carlo simulations.

Various strategies have been developed to model residential low voltage electricity demand. These attempts can be categorized in two approaches: bottom-up and top-down. The bottom-up approach has been implemented in various studies. The top-down approach is much less encountered in research.

Bottom-up approaches start from the behaviour of the individuals in the households. Capasso et al. [9] couple the probability of being at home with the probability of appliance load and sum the resulting loads to get the household load. Yao et al. [10] follow a similar approach, but put more effort in the 'being at home'-patterns. Widén et al. [11] define specific activities, e.g. vacuum cleaning, and generate load profiles based on the activities. Richardson et al. [12] work with activity profiles. Stokes [13] uses the behaviour indirectly and combines the load signature of an appliance with the yearly and daily patterns of demand for an appliance. Palensky et al. [14] work with on/off states and drain, store and flush events, but mainly to reproduce synthetic load profiles.

The downsides of bottom-up approaches are the intensity of modelling and the risk of missing appliances to model. However, bottom-up approaches allow for simulation of demand side management: appliances can be shifted or curtailed.

McLoughlin et al. [15] present a top-down approach. By using a homogeneous Markov chain, an attempt is made to re-generate electricity load profiles of five households. Autocorrelations are not reproduced because the behaviour of the individuals in the household are not taken into account.

Aggregated load has already been successfully modelled using top-down methods. Singh et al. [16] model distribution system load and Valverde et al. [17] model load for load flow analysis with Gaussian mixture models to capture the probability density functions. However, autocorrelation found in electricity demand of households was never incorporated.

Bottom-up approaches have in general good results because of the incorporation of a behaviour model. Top-down approaches have a lot of potential because of the lower modelling intensity: there is no need to model every appliance individually, which lowers the intensity of modelling significantly.

The detection of behaviour is in general done by pattern analysis. Techniques have been developed to find similarities within load profiles [18], [19] as between profiles [20]–[28], within different domains such as clustering or classification of profiles [20]–[25], forecasting [18], [26], selecting scenarios for load-wind combinations [19] and selecting demand response policies [28].

Bruckner et al. use hidden Markov models to capture behaviour in households. Sensor values are modelled with Markov models to extract semantic concepts. The goal is to learn daily routines. [29] Wind power output now is also correlated with previous measurements. Papaefthymiou et al. created a Markov chain Monte Carlo method to create synthetic series of wind power. [30] The probability density functions and the autocorrelation was regenerated in their approach.

Copyright (c) 2013 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

This paper presents a model that is able to regenerate residential electrical load. The method is intended for larger sets of data, where each item has autocorrelation. Similar items are grouped together to cluster resembling behaviour. A Markov model is created for each group to capture the behaviour. Other Markov models are developed to add variation to the behaviour. The variation is needed to make the regenerated data more realistic.

II. APPROACH

The research presented in this paper uses the top-down approach to generate load profiles. The model of the load profile generator is built in six steps:

- At first the load profile of each residential customer is transformed into a load curve, which represents the electrical demand during an average day.
- Clustering is applied to the transformed dataset. The cluster centres represent the artificial types of customers.
- For every cluster, a cumulative probability density function of the average fifteen minute power is calculated.
- The cumulative probability density function is made discrete via states with each an equal probability.
- A second order non-homogeneous Markov chain for each cluster is calculated to create a model that represents the behaviour of a household.
- A non-homogeneous Markov decision process for each weekday is put in place to spread the behaviour across a week.

A large dataset of load profiles of residential customers is the basis for modelling. Each load profile is a sequence of measured data, with a resolution of fifteen minutes, over a duration of a year. The data was provided, after the approval of the privacy commission, by the Flemish Regulator for the Electricity and Gas market[1]. The regulator keeps records of the monitored consumption of over thousand households.

Load curves are an aggregated form of load profiles. They are built by aggregating load profiles on a time basis, e.g. the average day of the week. Aggregation removes peaks in consumption, but keeps the underlying trend. Load curves are hence able to capture the consumption patterns and, as a consequence, the life patterns of a household. The life pattern of most people is in general relatively constant.

A fuzzy clustering technique is applied to the load curves to group together similar ones. A similar load curve implies resembling behaviour. Households are part of multiple clusters, depending on how likely the load curve belongs to the cluster. The clustering results, i.e. the cluster centres, are load curves that represent the cluster members. The load curves are also called ‘typical consumption patterns’.

A distribution of the electrical power is calculated for each household. The probability of belonging to a cluster for a household is used to weight the distribution for that particular cluster. The electrical power distribution of a cluster is determined by calculating the normalised sum of those weighted distributions. The cluster’s power distributions are parametrised by fitting a curve through them. The distributions are split into ten equal parts. These parts are the states for the Markov models. The interval edges of each part are the limits of the states for the Markov models.

A ‘behaviour’ Markov chain and seven ‘variation on behaviour’ Markov decision processes are built for each cluster. Autocorrelation in load profiles indicates that the power demand of any of the preceding days is correlated with current power demand. The behaviour Markov chain is constructed on the data of Mondays and is used to keep the power demand over the different days consistent. The preceding day is hence always the preceding Monday. Variation on the general behaviour for the different days of the week is generated with a Markov decision process that takes the general behaviour as input and gives specific day behaviour as output. Such a Markov decision process is trained on the data of the specific day and the corresponding previous Monday.

III. TYPICAL CONSUMPTION PATTERNS

Typical consumption patterns describe the way a group of customers uses electricity. There are multiple ways to aggregate customers in typical ones: family size, housing type, etc. However, it is also possible to define customers in an artificial way. By applying cluster analysis, similar consumption patterns are grouped together. The centre of the cluster represents the consumption pattern of all similar ones.

Amongst the different clustering techniques for load profiles and load curves are k-means clustering [21]–[24], [26], fuzzy c-means (FCM) clustering [21]–[23], hierarchical clustering [21], [23], modified follow the leader [21], self organising maps (SOM) [21], [22], [25], [28] and Expectation Maximization (EM) clustering [20].

Chicco et al. [21] compared k-means, FCM, hierarchical, modified follow-the-leader and SOM clustering. The best results were obtained by modified follow-the-leader and hierarchical clustering, k-means and fuzzy c-means performed slightly worse and SOM gave the worst results. Zhang et al. [22] compared k-means, FCM and SOM and found that in general, k-means performed slightly better than FCM which performed better than SOM. Zhang et al. applied clustering to load curves, Chicco et al. normalized the load curves before clustering. Coke et al. [20] pointed out that mixture models are better in smoothing out random effects and used a modified Expectation Maximisation clustering to group electrical load series. Clustering itself is done on the correlation structures and trends in the load profiles.

A clustering technique is called soft clustering or fuzzy clustering, if every customer belongs to more than one cluster [31]. The likelihood of belonging to a cluster is then the weight the customer has in that cluster. To scale up the data for the Markov models, clustering techniques with weights are preferred. The load curve clustering techniques with weights are FCM, EM clustering and SOM. SOM was not considered, because it had the worst performance of those three in other work. EM clustering is, in contrast with FCM (which is a centroid based technique), a density based technique. Density based techniques are better at handling randomness in data. Therefore, EM clustering is selected as clustering algorithm.

---

[1] Flemish Regulator of the Electricity and Gas market, http://www.vreg.be/

A. Data transformation

A clustering algorithm regards each measurement of a load profile as a dimension. Given the duration of a profile (a year) and the frequency of measuring (every fifteen minutes) this results in 35040 dimensions. The more dimensions, the harder it gets for a clustering algorithm to reject an item from a cluster, which results in a higher computational cost [32]. Another problem is that the clustering algorithm could find more clusters than there actually are by finding some similarities in the less relevant dimensions.

Chicco et al. [33] introduced representative load patterns (RLP) to cope with this problem. A representative load pattern is a normalised load curve. The approach has a big disadvantage: the magnitude of consumption is lost, only variations in electricity consumption are considered. Zhang et al. [22] used the same approach, but didn't normalise the load curves and named them typical load profiles (TLPs).

In the proposed method of this paper, the dimensionality is lowered by using load curves without normalisation. Load curves are an intuitive way of representing the electrical demand of customers and hence there behaviour. The load curves are constructed by using the average power every fifteen minutes of the day, for every day of the week, for a joint week every quarter, resulting in 2688 dimensions. Other combinations of time dimensions have been tried to construct load curves, but the results were less satisfying. The use of power every hour, for example, gave similar results, but the load curve is less smooth than in the fifteen minute case.

B. Clustering

The purpose of clustering is the formation of representatives of a whole group. The cluster centres in EM clustering are defined by the normal distribution over each dimension of the members of the cluster. Tenfold cross validation ensures the generalisation properties of the model.

The EM clustering algorithm [34] requires knowledge about the number of clusters to group the data into. To find the number of clusters, the algorithm initiates the k-means clustering algorithm for different values of k. k-means clustering divides the instances, i.e. the training examples, in k clusters by trying to reduce the sum of the square distances between the instance and the closest cluster centre. The value of k with the lowest overall square error is picked as the amount of clusters for EM clustering.

After initialisation, the EM clustering algorithm iterates between the expectation and the maximization steps. The iteration stops when the difference over the overall log-likelihoods, i.e. the sum of the log-likelihoods of all instances, of successive iterations is lower than a predefined threshold.

In the expectation step, the normal probability density value (pdf) of an instance (k) with corresponding load curve vector (x_k) in each dimension (d) for each cluster (c) is calculated as shown in Equation 1. The normal distribution requires the mean (μ) and the standard deviation (σ) of the dimension of the cluster to find the probability density value. To limit the amount of multiplications inside the algorithm, the log probability density ($\log pdf$) is calculated, as presented in Equation 2.

$$pdf_{k,c,d} = \frac{1}{\sqrt{2\pi{\sigma_{c,d}}^2}} e^{-\frac{(x_{k,d}-\mu_{c,d})^2}{2{\sigma_{c,d}}^2}} \tag{1}$$

$$\log pdf_{k,c,d} = -\log\sqrt{2\pi} - \log\sigma_{c,d} - \frac{x_{k,d}-\mu_{c,d}}{2{\sigma_{c,d}}^2} \tag{2}$$

In the maximization step, the joint log probability density value of an instance for a cluster ($\log jpdf_{k,c}$) is calculated. The value is the sum of the log probability density values over each dimension divided by the number of dimensions (n_d).

$$\log jpdf_{k,c} = \frac{\sum_{d=1}^{n_d} \log pdf_{k,c,d}}{n_d} \tag{3}$$

The joint probability density function of an instance for a cluster is converted into a weight ($w_{k,c}$). The weight is used to determine the mean (μ) and the standard deviation (σ) of the cluster centre in a dimension. The new mean is the normalized weighted sum of the instances. The standard deviation is recalculated using the new mean.

$$w_{k,c} = \frac{jpdf_{k,c}}{\sum_{c=1}^{n_c} jpdf_{k,c}} \tag{4}$$

$$\mu_{c,d} = \frac{\sum_{k=1}^{n_k} w_{k,c} \cdot x_{k,d}}{\sum_{k=1}^{n_k} w_{k,c}} \tag{5}$$

$$\sigma_{c,d} = \frac{\sum_{k=1}^{n_k} w_{k,c} \cdot (x_{k,d} - \mu_{c,d})}{\sum_{k=1}^{n_k} w_{k,c}} \tag{6}$$

The EM clustering algorithm iterates between determining the weights and recalculating the cluster centres until the weights are stabilised, i.e. the difference in the sum of all log joint probability values between the current step and the previous step is below a threshold. Once the algorithm finishes, the mean and the standard deviation of each dimension of each cluster is known.

C. Results

All profiles of the data-set have been converted into load curves. EM clustering with tenfold cross validation is applied to the load curves. The algorithm found ten clusters in the load curve dataset. The clusters and their probability can be found in Table I. Two of them have a very low probability and are considered outliers. However, the focus of load profile generation is on the clusters with a high probability. An example of the load curve of a cluster centre for week and weekend of the first quarter of the year (q1) is shown in Figure 1.

TABLE I
CLUSTERS WITH THEIR PROBABILITY

cluster	% of customers	outlier	focus
1	27.81		✓
2	25.90		✓
3	15.41		✓
4	14.09		✓
5	7.70		
6	3.23		
7	2.93		
8	2.27		
9	0.51	✓	
10	0.15	✓	

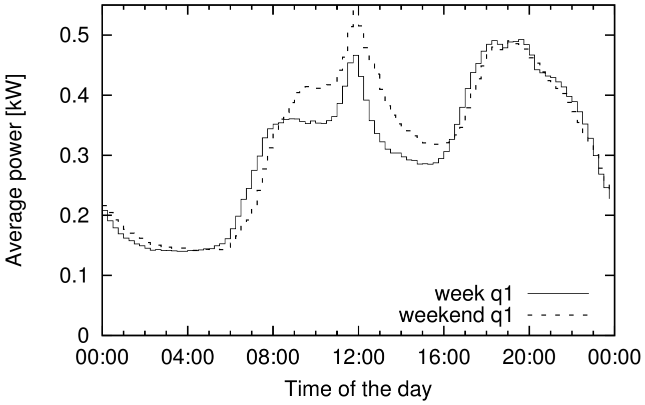


Fig. 1. Plot of the centre of cluster 2 for week and weekend of the first quarter.

IV. CREATING CUSTOMER MODELS

The electricity demand of a customer can be derived from historical information. An autocorrelation plot shows in-signal cross correlations. The signal for the autocorrelation plot in Figure 2 is from a randomly picked load profile. The plot indicates that the most useful information for the prediction of electricity demand is the demand of the last fifteen minutes and the demand of any day of the preceding week. The correlation between successive measurement points in the given load profile is 0.780. Peaks in the autocorrelation plot are noticeable when points are one day apart, resulting in a autocorrelation for the selected load profile of 0.167 for a distance of one day between two points. The behaviour of one day is similar to the behaviour on other days, i.e. the behaviour during a week can be modelled based on the behaviour of one day.

Behaviour and autocorrelations have been captured successfully with Markov models [29], [30] and are hence the modelling choice. A Markov model is a mathematical system defined by a set of states. States are interconnected with transitions and each transition has a certain probability which doesn't depend on the past. The event that causes a transition is, in this case, a new time step.

A. States

The electricity demand in load profiles is discrete in the time domain but continuous in power. The power has to be discrete to create states. Each state represents an interval in power. For an optimal use of the Markov models, each state should in general be equally probable.

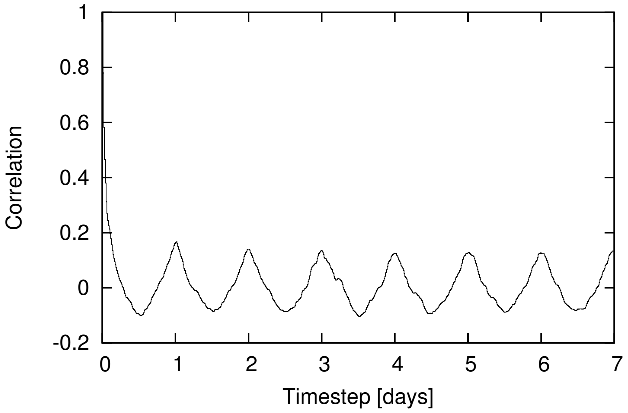


Fig. 2. Load profile autocorrelation

The discrete states for the Markov models are created by first defining the power distribution for each cluster. Each load curve belongs to multiple clusters. The weight with which a load curve belongs to a cluster is calculated using Equation 4.

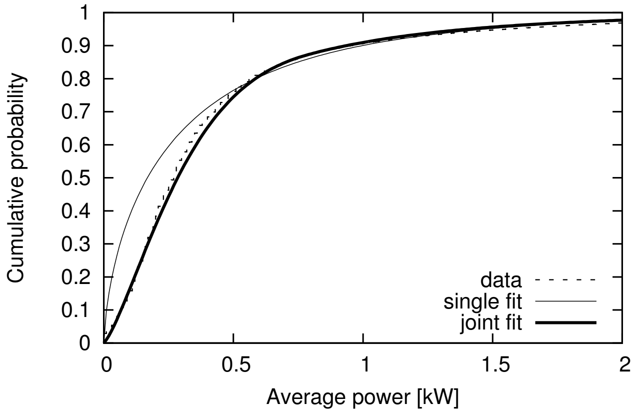
Defining the power probability distribution of a group of customers is done by fitting a curve through the histogram of the power distribution. The most common distribution over power or total consumption is Weibull [35], [36]. Equation 8 and Equation 7 show respectively the probability and the cumulative probability density functions of the Weibull distribution. k is the shape parameter, λ is the scale parameter and x is the random variable (in this case average power) in both equations.

The more observations available to create the histogram, the better the power distribution will be estimated. A way to scale the quality of the histograms up is by using fuzzy ones [37]. The softening of the histograms is done by using the cluster weight of an instance in order to build up the power histogram of that cluster. A customer with a large weight in a cluster, will have a large impact on the histogram of that cluster, while a customer with a very small weight will still have a small influence on the histogram.

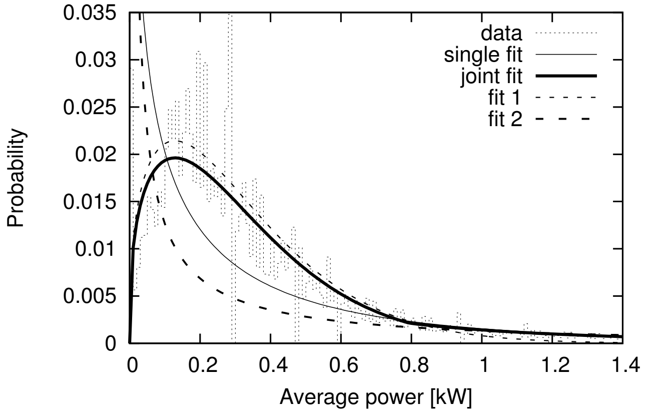
$$cdf_{weibull} = 1 - e^{-\left(\frac{x}{\lambda}\right)^k} \tag{7}$$

$$pdf_{weibull} = \frac{k}{\lambda}\left(\frac{x}{\lambda}\right)^{k-1} e^{-\left(\frac{x}{\lambda}\right)^k} \tag{8}$$

Figure 3(a) represents the cumulative probability fit 'single fit' (Equation 7) and the histogram of the 'data' with a bin width of 0.01 kW. As shown in Figure 3(a), the Weibull curve does not fit the data perfectly, which is confirmed by the plot of the probability density fit (Equation 8) in Figure 3(b). The fit got improved by using a joining of two Weibull curve fits. Each of the two curve fits are specialized in one part of the data. Fit 1 targets the first 65 % of the data, while fit 2 targets the tail which contains the remaining 35 % of the data. Both are fitted on the cumulative distribution. In Figure 3(b), both fits are shown as probability density functions by using the resulting λ



(a) Single and joint curve fit of cumulative probability



(b) Single and joint curve fit of probability

Fig. 3. Cumulative (a) and normal probability (b) fit of average power per fifteen minutes for cluster 8

and k values of the cumulative fits, i.e. the parameters found by fitting to Equation 7 are used to plot according to Equation 8. The joint fit in Figure 3(b) is the result of taking the first fit for the first 65 % of the data and the rest for the remaining values and normalizing the data so that the sum of all values equals one.

Table II shows the comparison of the single and the joint fit of the density functions for cluster 8. Because the first value of the single fit is infinity, all values are calculated from the second value (0.01 kW); the last data point is at 22 kW. The average power in the data set is 0.45 kW. The average power in the joint fit is closer to the average power of the data than the single fit. The difference in median compared to the data is almost the same for the single and the joint fit. The median of the joint fit is higher than the median in the data, while the single fit has a smaller median. The root mean square error ($RMSE$) is found to be much smaller for the joint fit case, compared to the single fit.

States of a Markov model of a cluster are chosen to have the same probability. For the Markov models in this paper, 10 states are considered. The boundaries for each state are set at the power of a multiple of 10% of the cumulative probability of the fit. Given the probability density function, this means a lot of detail for the frequently occurring low powers and lesser detailed high powers which aren't frequent.

TABLE II
EVALUATION OF SINGLE AND JOINT FIT FOR CLUSTER 8

	data	single fit	joint fit
average	0.45 kW	0.41 kW	0.44 kW
median	0.26 kW	0.22 kW	0.29 kW
RMSE	0	0.0021	0.00086

Markov models consist of transition matrices. The size of the matrices scales with the order of the Markov models. The behaviour is modelled with second order chains, the detail is captured with decision processes. In both cases, this results in a transition matrix of S^3, where S represents the number of states. The more states, the larger and the sparser the matrices become. Moreover, a higher number of states means less data to calculate the probability of each transition between states and more risk of over-fitting the data. On the other hand, few states results in lower detail. The number of transitions is chosen as a trade-off between over-fitting and having enough detail.

B. Transitions

The probability of a certain fifteen minute average power varies during a day. In cluster 2 for example, the average fifteen minute power at 12h00 is more likely to be higher than at 04h00, as shown in Figure 1. This means that the fit of the cumulative probability cannot be used to determine the state to start at. The probability of each state at the start needs to be calculated separately, again taking into account the cluster weight of that instance.

The transition probabilities need to change at each time step because of the variation in probability of the power between different time steps. The change in transition probabilities make the Markov models non-homogeneous ones. For the models themselves, best results were obtained by using a second order Markov chain and Markov decision processes. A second order Markov chain involves using data of current and previous step to predict the next step. Markov decision processes take the current state and a given state to predict the next step.

For each time step, a transition matrix is constructed. A transition matrix contains the probabilities to go from one state to another state at the next time step. The main difference of the Markov models used in this paper and regular Markov models is the use of the cluster weights of the instances to calculate the probabilities. An instance with a large cluster weight for the given cluster, will have a larger impact on the transition matrices for that cluster. It also ensures that instances with a low cluster weight are still able to impact transitions in contrast to using regular frequency counting.

The general behaviour in a cluster is determined by the behaviour on Mondays. As mentioned before, consumption is made discrete in 10 states, each representing 10 % of the data of that cluster. Let be $X(t)$ a stochastic process with 96 time steps with one time step t for each fifteen minutes of a day.

The behaviour is modelled by (Pr^b), presented in Equation 9, which represents the probability of going to state i at current time step t from state j at previous time step ($t-1$) and state k at the state before that ($t-2$). Each time step has its own transition matrix $P^b(t, t-1, t-2)$. Each transition matrix has a 10x10x10 dimension, given that there are 10 states. Each cluster has its own behaviour model.

The models of the different days of the week are built in the same way as the behaviour model and are called detailed models. The purpose of the detailed models is to include day specific information in the sequence of states, e.g. the time of electricity demand is different during weekdays compared to weekends. The state at current time step ($X^d(t) = i$) depends on the state at the same time step ($X^b(t) = j$) of the behaviour model and previous step ($X^d(t-1)$) in the current sequence, as expressed by Pr^d in Equation 10. Again, each time step has its own transition matrix $P^d(t, t, t-1)$ of dimension 10x10x10. Each day of the week has its own range of transition matrices. Also, each cluster has its own range of detailed models.

$$Pr^b\{X^b(t) = i|X^b(t-1) = j, X^b(t-2) = k\} = P^b_{i,j,k}(t, t-1, t-2) \tag{9}$$

$$Pr^d\{X^d(t) = i|X^b(t) = j, X^d(t-1) = k\} = P^d_{i,j,k}(t, t, t-1) \tag{10}$$

V. PROFILE GENERATION

The basis for a generated load profile of a cluster is constructed by a random walk through a behaviour chain and by using that sequence to random walk the detailed model. The result is a sequence of a week, or the sequence of multiple weeks if the detailed models are walked multiple times. The state are made continuous again by taking a random sample in the interval of the state and passing it to the inverse of the joint distribution fit. The outcome of the inverse distribution is the actual power in that state. When applied to the sequence of states, the load profile is generated. The process is described more formally in following 4 steps:

- Select a cluster to generate a profile from.
- Create the general behaviour of the customer.
  - Randomly select two start states ($X^b(t_0 - 1)$ and $X^b(t_0)$) according to the probability of both states.
  - Random-walk the behaviour Markov chain until all behaviour states ($X^b(t)$) are generated.
- Create the behaviour of the customer on different days.
  - Randomly select a start state for the day ($X^d(t_0)$), according to the probability of the state.
  - Use the start state of the day ($X^d(t_0)$) and the start state of the model ($X^b(t_0)$) to create the next state ($X^d(t_1)$).
  - Random-walk the detail Markov model of that day until the detailed behaviour of the customer during that day is generated.
  - Repeat for all days of the week and repeat multiple times if more than one week is needed.
- Convert the states into power.
  - Randomly sample a value from the power distribution within the limits of the state.
  - Repeat for each state until the load profile is generated.

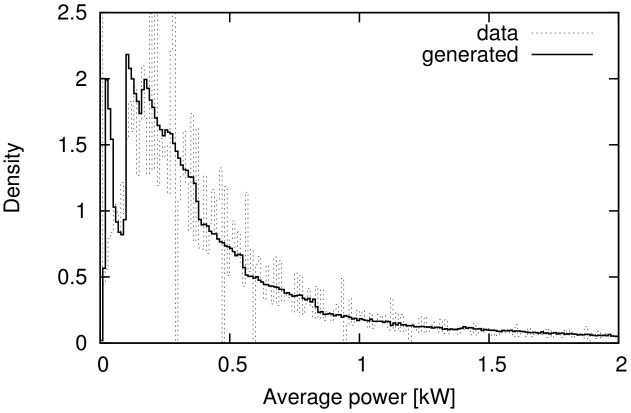


Fig. 4. Power distribution of generated profiles of cluster 4.

To make a representative set of generated profiles, the amount of profiles for each cluster needs to be in proportion to the probability being part of the cluster, as defined in Table I.

VI. VALIDATION

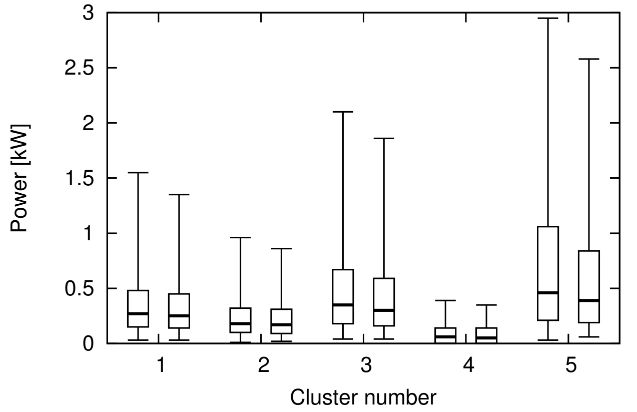
The evaluation of the presented method is done by comparing the measured profiles with the generated profiles. Four aspects of the profile are checked:

- The average power of the cluster centre should be the same as the average power of all generated profiles of that centre.
- The power distribution of a all measured and generated profiles of a cluster should be the same.
- The shape of the cluster centre from the EM-clustering algorithm and the aggregation (average) of the generated profiles of that cluster should be the same.
- Load profiles with a similar autocorrelation should be found in the measured and the generated profiles.

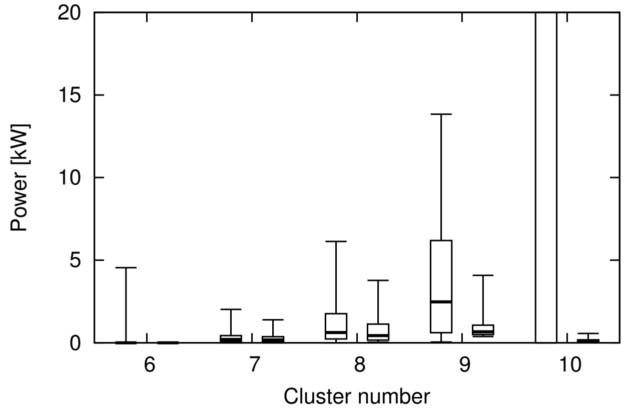
The data of only very few customers are used to build up the Markov models for the outlier clusters (clusters 9 and 10). The models for these clusters are bad, given the limited amount of data. The lower the probability of a cluster, the less data is available to train the Markov models, hence the worse the Markov model of that cluster will perform.

The power distribution of the real data and the generated profiles of cluster 4 are shown in Figure 4. The difference in the power distribution is marginal in this case, but Table III shows that the average power is in general a bit lower in the case of the generated profiles.

An overview of the power distributions is presented in Figure 5 by depicting a box-plot of the measured (left) and the generated (right) data for all clusters. The minimum value in the plot represents the fifth percentile of the data, the ninety-fifth percentile is the maximum value. The box itself is bound to the lower quartile (first 25 % of the data) and the higher



(a) Box-plots representing the power distribution of cluster 1 to 5 for measured (left) and generated (right) data



(b) Box-plots representing the power distribution of cluster 6 to 10 for measured (left) and generated (right) data

Fig. 5. Box-plots of clusters 1 up to 5 (a) and clusters 6 up to 10 (b)

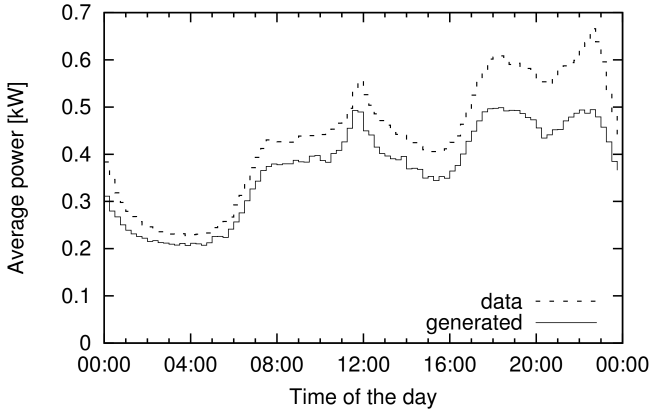


Fig. 6. Cluster 1 load curve for the actual and the generated data (from the joint fit model) for an average week in the second quarter.

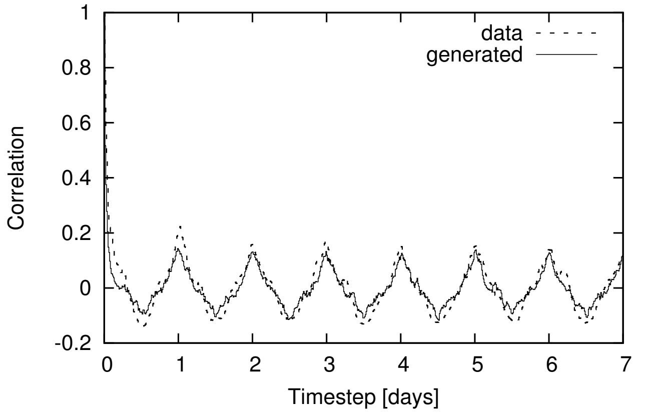


Fig. 7. 13 week autocorrelation of a measured and a similar generated load profile.

quartile (75 % of the data). The median is indicated by a thick line inside the box.

The box-plot confirms the power distribution comparison of cluster 4 in Figure 6: the distributions clearly resemble each other. The power distribution of measured and generated data of clusters with focus on (i.e. clusters 1, 2, 3 and 4, see Table I) are very similar. For cluster 5 and 7, median, higher quartile and ninety-fith percentile are lower in the generated data case in comparison to the measured data. Cluster 6 is a special case because power is 86.7 % of the time zero. The generated data in cluster 6 is 95 % of the time zero, which is reflected in the low maximum value in the box-plot. The power distribution of the generated data of cluster 8 is bad. For clusters 9 and 10, the distributions do not resemble the original data.

The average power over a year gives an idea about the total electricity consumption during that year. It can be seen as a proxy of the total electricity consumption. The average power in watt of the cluster centres and the generated data is shown in Table III. The average power of the clusters with a high probability, i.e. the clusters with focus on as indicated in Table I, are a bit lower but close to the real average power. This is a very good result: given the skewness of the probability distribution, it is hard to get the average correct without overfitting the data.

However, there are some problems. Cluster 5 represents 7.7 % of the data and the average power is only half of what it should be. Another problem is the average power of cluster 7 and 8. Combined, they represent 5.2 % of the data. The average powers of both are only a third of the average power they should be. The results are, unsurprisingly, bad for the outlier clusters. For cluster 10, the generation of load profiles even failed. Since the outlier clusters only represent 0.66 % of the data jointly, this is not really a problem.

Figure 6 shows, for cluster 1, the average power during the week plotted against the time of the day. The joint power distribution fit has about the same shape as the data. A large group of generated load profiles of a cluster has a similar load curve as the given cluster.

The autocorrelation of a 13 week (3 months) load profile can be found in Figure 7. A generated profile that looks like the selected load profile is chosen. The autocorrelation is very similar to the autocorrelation of the real profile.

TABLE III
AVERAGE POWER [W] OF THE CLUSTER CENTRES AND DATA GENERATED FORM THE MARKOV CHAIN

cluster	Q1		Q2		Q3		Q4	
	data	generated	data	generated	data	generated	data	generated
1	*534*	*434*	*444*	*374*	*433*	*372*	*538*	*457*
2	*314*	*289*	*265*	*253*	*257*	*253*	*317*	*303*
3	*848*	*583*	*684*	*491*	*647*	*478*	*848*	*608*
4	*101*	*109*	*82*	*100*	*81*	*104*	*104*	*121*
5	1550	814	1204	639	1111	635	1440	842
6	1332	1421	389	483	122	345	987	1436
7	1546	494	648	376	412	376	1386	499
8	3781	1177	2965	809	2675	782	3523	1240
(9)	(7871)	(1845)	(8120)	(1299)	(7549)	(1152)	(8430)	(1711)
(10)	(20748)	(435)	(21538)	(286)	(22315)	(379)	(21964)	(220)

The overall electricity demand will be underestimated by using the generated profiles instead of the real data. Also, the use of the generated data in detailed simulations will have an impact. When the same cluster probabilities of the data are chosen, 86.44 % of the generated profiles will jointly have a slightly lower average power and a power distribution that fits the original data well. 12.9 % will have an joint average power that is only half or a third of what it should be. 0.66 % will have a joint average power that is a fifth or lower. The problems are mainly situated in less frequently occurring situations, like very high electricity demand at residential level. It is possible to test standard neighbourhoods, but more extreme situations will be underestimated.

VII. CONCLUSIONS

Detailed information about the electricity demand of residential customers is needed for the development of smart grid integration strategies. Finding voltage problems at distribution level through simulations requires full customer load profiles and can not be based on aggregated data. Load profiles from the residential sector are hard to get due to privacy concerns. The proposed method bypasses the problem of the availability of data provided by electrical companies by transforming a large dataset of residential load profiles into a model that is able to create a set of synthetic non-aggregated profiles. The model is based on statistical information, which ensures that the original customer load profiles cannot be traced back using the synthetic profiles. With this method, companies that monitor electricity consumption have a way to provide data, just as they do with aggregated data, without having to worry about privacy issues. An extra advantage of the method is the possibility to use the models for Monte Carlo simulations.

Current best practice load profile generation techniques work bottom-up: apparatuses and home activity are modelled and used to create load profiles. The disadvantage of these techniques is the intensity of modelling. A top-down technique has been proposed before, but the results were poor because behaviour was not incorporated into the method. The method in this paper copes with this problem by clustering and modelling similar consumption behaviour.

Behaviour is expressed in load curves. Grouping of similar behaviour is done with a fuzzy clustering algorithm. In this way, each customer has a weight in every cluster. The cluster weights are used to construct the probability density functions of the electrical power and to build the Markov models that capture the behaviour and the variation of the behaviour of customers.

A load profile of a cluster is generated by randomly walking the Markov models of the cluster. The found states are translated into electrical power with the use of the inverse probability density function.

The load profile generation works properly for simulations of standard neighbourhoods. 86.44 % of the data has a power distribution that fits the data well. More extreme cases, like very high electricity demand (e.g. a small enterprise in a residential area), are underestimated in the load profile generation. 12.9 % of the generated data has an average power that is half or a third of the expected average power. Only 0.66 % of the data has an average power that is a fifth or lower than what it should be.

ACKNOWLEDGEMENTS

This work is supported by the Flemish Minister for Innovation (Minister I. Lieten) via the project Linear [38] organized by the Institute for Science and Technology (IWT).

REFERENCES

[1] C. De Jonghe, B. Hobbs, and R. Belmans, "Optimal generation mix with short-term demand response and wind penetration," *IEEE Transactions on Power Systems*, vol. 27, no. 2, pp. 830–839, 2012.

[2] C. Cecati, C. Citro, A. Piccolo, and P. Siano, "Smart operation of wind turbines and diesel generators according to economic criteria," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 10, pp. 4514–4525, 2011.

[3] K. Dyke, N. Schofield, and M. Barnes, "The impact of transport electrification on electrical networks," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 12, pp. 3917–3926, 2010.

[4] K. Clement, E. Haesen, and J. Driesen, "The impact of vehicle-to-grid on the distribution grid," *Electric Power Systems Research Journal*, vol. 81, no. 1, pp. 185–192, 2011.

[5] H. Kanchev, D. Lu, F. Colas, V. Lazarov, and B. Francois, "Energy management and operational planning of a microgrid with a PV-based active generator for smart grid applications," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 10, pp. 4583–4592, 2011.

[6] J. Tant, F. Geth, D. Six, P. Tant, and J. Driesen, "Multiobjective battery storage to improve pv integration in residential distribution grids," *IEEE Transactions on Sustainable Energy*, vol. 4, no. 1, pp. 182–191, 2013.

[7] P. McDaniel and S. McLaughlin, "Security and privacy challenges in the smart grid," *Security Privacy, IEEE*, vol. 7, no. 3, pp. 75–77, 2009.

[8] W. Labeeuw and G. Deconinck, "Non-intrusive detection of high power appliances in metered data and privacy issues," in *6th International Conference on Energy Efficiency in Domestic Appliances and Lighting (EEDAL 11)*, Copenhagen, Denmark, 2011.

[9] A. Capasso, W. Grattieri, R. Lamedica, and A. Prudenzi, "A bottom-up approach to residential load modeling," *IEEE Transactions on Power Systems*, vol. 9, no. 2, pp. 957–964, 1994.

[10] R. Yao and K. Steemers, "A method of formulating energy load profile for domestic buildings in the uk," *Energy and Buildings*, vol. 37, no. 6, pp. 663–671, 2005.

[11] J. Widén, M. Lundh, I. Vassileva, E. Dahlquist, K. Ellegård, and E. Wäckelgård, "Constructing load profiles for household electricity and hot water from time-use data-modelling approach and validation," *Energy and Buildings*, vol. 41, no. 7, pp. 753–768, 2009.

[12] I. Richardson, M. Thomson, D. Infield, and C. Clifford, "Domestic electricity use: A high-resolution energy demand model," *Energy and Buildings*, vol. 42, no. 10, pp. 1878–1887, 2010.

[13] M. Stokes, "Removing barriers to embedded generation: a fine-grained load model to support low voltage network performance analysis," Ph.D. dissertation, Institute of Energy and Sustainable Development, De Montfort University, Leicester, 2005.

[14] P. Palensky, F. Kupzog, A. Zaidi, and K. Zhou, "Modeling domestic housing loads for demand response," in *34th Annual Conference of IEEE Industrial Electronics. IECON 2008*, 2008, pp. 2742–2747.

[15] F. McLoughlin, A. Duffy, and M. Conlon, "The generation of domestic electricity load profiles through markov chain modelling," in *3rd International Scientific Conference on Energy and Climate Change*, Athens, Greece, 2010, pp. 18–27.

[16] R. Singh, B. Pal, and R. Jabr, "Statistical representation of distribution system loads using gaussian mixture model," *IEEE Transactions on Power Systems*, vol. 25, no. 1, pp. 29–37, 2010.

[17] G. Valverde, A. Saric, and V. Terzija, "Probabilistic load flow with non-gaussian correlated random variables using gaussian mixture models," *Generation, Transmission Distribution, IET*, vol. 6, no. 7, pp. 701–709, 2012.

[18] X. Li, C. Bowers, and T. Schnier, "Classification of energy consumption in buildings with outlier detection," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 11, pp. 3639–3644, 2010.

[19] P. Siano, C. Cecati, H. Yu, and J. Kolbusz, "Real time operation of smart grids via fcn networks and optimal power flow," *IEEE Transactions on Industrial Informatics*, vol. 8, no. 4, pp. 944–952, 2012.

[20] G. Coke and M. Tsao, "Random effects mixture models for clustering electrical load series," *Journal of Time Series Analysis*, vol. 31, no. 6, pp. 451–464, 2010.

[21] G. Chicco, R. Napoli, and F. Piglione, "Comparisons among clustering techniques for electricity customer classification," *IEEE Transactions on Power Systems*, vol. 21, no. 2, pp. 933–940, 2006.

[22] T. Zhang, G. Zhang, J. Lu, X. Feng, and W. Yang, "A new index and classification approach for load pattern analysis of large electricity customers," *IEEE Transactions on Power Systems*, vol. 27, no. 1, pp. 153–160, 2012.

[23] G. Tsekouras, N. Hatziargyriou, and E. Dialynas, "Two-stage pattern recognition of load curves for classification of electricity customers," *IEEE Transactions on Power Systems*, vol. 22, no. 3, pp. 1120–1128, 2007.

[24] V. Figueiredo, F. Rodrigues, Z. Vale, and J. Gouveia, "An electric energy consumer characterization framework based on data mining techniques," *IEEE Transactions on Power Systems*, vol. 20, no. 2, pp. 596–602, 2005.

[25] S. Verdu, M. Garcia, C. Senabre, A. Marin, and F. Franco, "Classification, filtering, and identification of electrical customer load patterns through the use of self-organizing maps," *IEEE Transactions on Power Systems*, vol. 21, no. 4, pp. 1672–1682, 2006.

[26] M. Espinoza, C. Joye, R. Belmans, and B. De Moor, "Short-term load forecasting, profile identification, and customer segmentation: A methodology based on periodic time series," *IEEE Transactions on Power Systems*, vol. 20, no. 3, pp. 1622–1630, 2005.

[27] D. De Silva, X. Yu, D. Alahakoon, and G. Holmes, "A data mining framework for electricity consumption analysis from meter data," *IEEE Transactions on Industrial Informatics*, vol. 7, no. 3, pp. 399–407, 2011.

[28] S. Valero, M. Ortiz, C. Senabre, C. Alvarez, F. Franco, and A. Gabaldon, "Methods for customer and demand response policies selection in new electricity markets," *Generation, Transmission Distribution, IET*, vol. 1, no. 1, pp. 104–110, 2007.

[29] D. Bruckner and R. Velik, "Behavior learning in dwelling environments with hidden markov models," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 11, pp. 3653–3660, 2010.

[30] G. Papaefthymiou and B. Klockl, "Mcmc for wind power simulation," *IEEE Transactions on Energy Conversion*, vol. 23, no. 1, pp. 234–240, 2008.

[31] R. Nock and F. Nielsen, "On weighting clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 8, pp. 1223–1235, 2006.

[32] R. Bellman, *Adaptive Control Processes*. Princeton University Press, 1961.

[33] G. Chicco, R. Napoli, F. Piglione, P. Postolache, M. Scutariu, and C. Toader, "Load pattern-based classification of electricity customers," *IEEE Transactions on Power Systems*, vol. 19, no. 2, pp. 1232–1239, 2004.

[34] I. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd ed. Morgan Kaufmann, 2005.

[35] A. Seppälä, "Load research and load estimation in electricity distribution," Ph.D. dissertation, Helsinki University of Technology, Espoo, Finland, 1996.

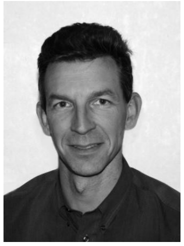
[36] W. Labeeuw and G. Deconinck, "Customer sampling in a smart grid pilot," in *2012 IEEE PES General Meeting*, San Diego, CA, USA, 2012.

[37] K. Loquin and O. Strauss, "Fuzzy histograms and density estimation," in *Soft Methods for Integrated Uncertainty Modelling*, ser. Advances in Soft Computing. Springer Berlin / Heidelberg, 2006, vol. 37, pp. 45–52.

[38] B. Dupont, P. Vingerhoets, P. Tant, K. Vanthournout, W. Cardinaels, T. De Rybel, E. Peeters, and R. Belmans, "Linear breakthrough project: Large-scale implementation of smart grid technologies in distribution grids," in *IEEE PES International Conference and Exhibition on Innovative Smart Grid Technologies (ISGT Europe)*, Berlin, Germany, 2012.

BIOGRAPHIES

Wouter Labeeuw received the M.Eng. degree in electronics-ict from Provinciale Industriële Hogeschool, Kortrijk, Belgium, in 2007 and the M.Sc. degree in computer science from the Katholieke Universiteit Leuven (KU Leuven), Leuven, Belgium, in 2009. He is currently a Research Assistant at KU Leuven ESAT/ELECTA, where he is working toward the Ph.D. degree. His research interests include data analysis and demand side management.



Geert Deconinck received the M.Sc. degree in electrical engineering and the Ph.D. degree in applied sciences from the Katholieke Universiteit Leuven (KU Leuven), Leuven, Belgium, in 1991 and 1996, respectively. Since 1997, he has been with the staff of the Department of Electrical Engineering, KU Leuven, first as a Postdoctoral Fellow and now as full Professor. His research interests include the design, analysis, and assessment of software-based fault-tolerant solutions to meet real-time, dependability, and cost constraints for embedded applications on parallel and distributed systems.