World Academy of Science, Engineering and Technology
International Journal of Electrical and Computer Engineering
Vol:17, No:10, 2023

Fast Short-Term Electrical Load Forecasting under High Meteorological Variability with a Multiple Equation Time Series Approach

Charline David, Alexandre Blondin Massé, Arnaud Zinflou

Abstract—We present a multiple equation time series approach for the short-term load forecasting applied to the electrical power load consumption for the whole Quebec province, in Canada. More precisely, we take into account three meteorological variables — temperature, cloudiness and wind speed —, and we use meteorological measurements taken at different locations on the territory. Our final model shows an average MAPE score of 1.79% over an 8-years dataset.

Keywords—Short-term load forecasting, special days, time series, multiple equations, parallelization, clustering.

I. Introduction

THE sustainable energy transition is underway throughout the world, and electricity produced from so-called "green" sources will play a major role to enable it. In Quebec, Canada, where more than 99% of electricity is generated from renewable energy — mostly from hydroplants and wind turbines — fossil fuel is still used for power transportation, building space heating and heat generation in industrial processes. The rapid electrification of these sectors of activities is part of the government's plan to meet its Paris Agreement commitment to reduce the province's emissions by 25 megatons of CO_2 by 2030 with respect to 1990 emissions. Hence, as more and more intermittent renewable energy is added to the grids, accurate load forecasting is of critical importance to balance production and consumption for the modern power grids. Any improvement on short-term load forecasting is very beneficial for both the consumers and utility companies to optimize resources and costs.

Load forecasting is often classified according to the horizon it focuses on: very short-term load forecasting (VSTLF), short-term load forecasting (STLF), medium-term load forecasting (MTLF), or long-term load forecasting (LTLF). The forecasting horizon can span seconds, minutes, days, weeks or even years, depending on the planning or operational function it supports. Among them, STLF, which is useful in real-time energy management, is our main focus.

STLF has always been a challenging task that triggers the interest of both academia and the industrial sector [1]–[5]. The overall electricity consumption represented by all aggregated electric power loads is often influenced by meteorological factors, such as temperature, wind speed, cloud cover, type and intensity of precipitations [1]. And the effect can be quite significant: For example, an increase of only 1 °C in ambient temperature or urban warming can impact the peak of electricity consumption or global consumption by an overhead of more than 4% [6]. In the particular case of Quebec, the climate is characterized by a wide thermal range: extreme temperatures both during the winter and the summer, high wind variability in some parts of the territory and major continental air masses sweeping down from the northwest. Fig. 1 shows the correlation between temperature and power load consumption observed on a 8-year dataset ranging from 2012-01-01 to 2019-12-31. Under 15 °C, the lower the temperature is, the higher the power load is, which is explained by the fact that more than 80% of the households use electricity for heating. Similarly, above 20 °C, we observe that the power load starts increasing due to the increase in use of cooling devices. Hence, the forecast models must deal with a temperature-sensitive and season-dependent load profile. On top of that, power consumption is not only related to weather and may be influenced by several other factors, such as the decline of heavy industry, new social behavior

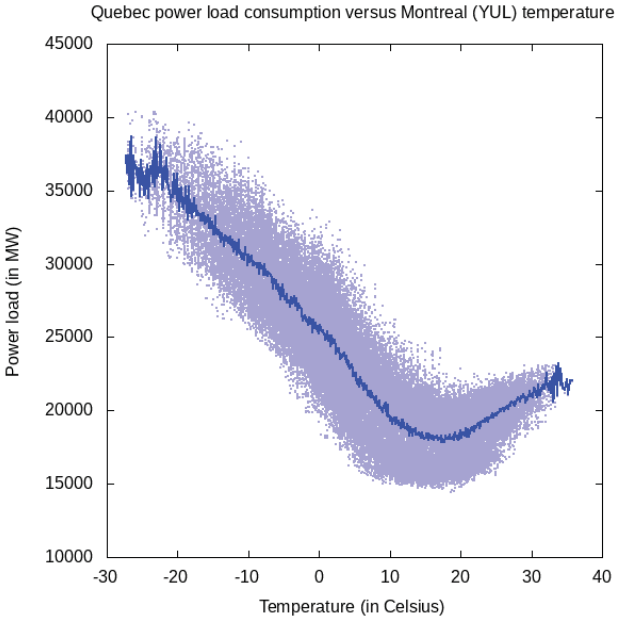

Fig. 1 Comparison of Quebec hourly power load against temperature (measured in Montreal) between 2012-01-01 and 2019-12-31

Charline David, Alexandre Blondin Massé, and Arnaud Zinflou are affiliated with the Institut de recherche d'Hydro-Québec (IREQ).

C. David and A. Blondin Massé* are also affiliated with the Department of Computer Science, Université du Québec à Montréal, Montréal (*e-mail: blondinmasse.alexandre@hydroquebec.com).

of the consumer (teleworking, behind-the-meter production, active role of the consumer), calendar information (day of the week, special days), transportation electrification, process automation, building insulation, and eventually electricity price [1], [2].

Various approaches have been used to build STLF models — see [7] for a recent survey. Quantitative models, such as seasonal auto-regressive integrated moving average (SARIMA) with support vector regression (SVR) [8], non parametric regression (NR) [9] or multiple equation time series [3] have proved to be reliable, very accurate and interpretable. Several works, especially in the recent years, have also successfully used machine learning approaches, ranging from artificial neural networks [10]–[12] and reinforcement learning [13], [14] to meta-learning [15] or graph neural networks [16]. Deep learning techniques are very promising, but their black-box nature and the complexity of scheduling their training and maintaining their lifecycle remain challenging [17].

In this paper, we concentrate on a multiple equation time series approach. Initiated by Ramanathan et al. in [18], significant improvements have been proposed, in particular by Cancelo et al. [2] and, more recently, by Clements et al. [3]. Although multiple equation time series models seem less popular in the literature, we believe that they present advantages that make them interesting in a high performance computing setting. Indeed, as detailed in the next sections, (1) the model we describe is easily interpretable, (2) its training can be parallelized with only slight modifications of the implementation, (3) it can be trained very fast (a few seconds on a standard laptop) and (4) it has competitive forecast accuracy (we obtained an average MAPE of 1.79% on our dataset).

The remaining of the paper is divided as follows: In Section II, we recall the main ideas behind Clements et al.'s [3] mathematical model. Then, in Section III, we present various extensions of the reference model. Section IV is devoted to the experimental setting, while Section V reports the main observations and results obtained. Section VI concludes with a brief discussion about special days.

II. A Multiple Equation Time-Series Approach

In this section, we briefly summarize the model equations from [3]. Let $\bar{t}$ be the number of time steps per day ($\bar{t} = 24$ in our case, but Clements et al. [3] consider half-hours, so that $\bar{t} = 48$). Also, for any time step t of day d, let

- $L_{d,t}$ be the logarithm of the load (in MW) at time step t of day d;
- $T_{d,t}$ be the temperature (in Celsius) at time step t of day d;
- $W_{d,p}$ indicate whether d is the p-th day of the week;
- $S_{d,k}$ indicate whether d is a special day of kind $k \in \{1, 2, \ldots, k'\}$, where k' is the number of special day types.

Moreover, for any day d and time step t of the day, let

$$\begin{aligned} Y_{d,t,q} &= 2q\pi(d\bar{t} + t)/(365.2425\bar{t}) \\ H_{d,t,1} &= \textsc{ClRamp}^{-}_{-23,13}(T_{d,t}) \\ H_{d,t,2} &= \textsc{ClRamp}^{-}_{-23,1}(T_{d,t}) \\ C_{d,t,1} &= \textsc{ClRamp}^{+}_{21,33}(T_{d,t}) \\ C_{d,t,2} &= \textsc{ClRamp}^{+}_{28,33}(T_{d,t}), \end{aligned}$$

where

$$\textsc{ClRamp}^{-}_{x_0,x_1}(x) = \begin{cases} x_1 - x_0, & \text{if } x \le x_0; \\ x_1 - x, & \text{if } x_0 < x \le x_1; \\ 0, & \text{if } x_1 < x. \end{cases}$$

$$\textsc{ClRamp}^{+}_{x_0,x_1}(x) = \begin{cases} 0, & \text{if } x \le x_0; \\ x - x_0, & \text{if } x_0 < x \le x_1; \\ x_1 - x_0, & \text{if } x_1 < x. \end{cases}$$

are respectively the descending and ascending clipped ramp functions. It is worth mentioning that the intervals $[-23, 13]$, $[-23, 1]$, $[21, 33]$ and $[28, 33]$ are different from those selected in [3] and were deduced by examining the correlation between power load consumption and temperature on Quebec territory (see Fig. 1). Finally, for any real number α, β and any random variable X, let

$$\textsc{SC}(\alpha, \beta, X) = \alpha \sin X + \beta \cos X.$$

and let $\mathcal{O} = \{0, -1\}$ be the set of days offsets when considering temperature and special days (0 is for the current day, -1 for the previous day, -7 would be for the previous week). Then the complete and preferred model adapted from [3] is given by

$$\begin{aligned} L_{d,t} = \; & \lambda_{t,0} + \left(\sum_{p=1}^{7} \omega_{t,p} W_p \right) L_{d-1,t} \\ & + \left[\lambda_{t,1} + \sum_{q=1}^{4} \textsc{SC}(\gamma_{t,q,1}, \gamma_{t,q,2}, Y_{d,t,q}) \right] L_{d-7,t} \\ & + \lambda_{t,2} L_{d-1,\bar{t}-1} + \lambda_{t,3} L_{d,t-1} \\ & + \phi_{t,1} \varepsilon_{d-1,t} + \phi_{t,2} \varepsilon_{d-7,t} + \varepsilon_{d,t} \\ & + \sum_{o \in \mathcal{O}} \sum_{k=1}^{2} (\tau_{t,o,k,1} H_{d+o,t,k} + \tau_{t,o,k,2} C_{d+o,t,k}) \\ & + \sum_{o \in \mathcal{O}} \sum_{k=1}^{k'} \sigma_{t,o,k} S_{d+o,k}, \end{aligned} \tag{1}$$

where, for each $t = 1, 2, \ldots, \bar{t}$ and each $o \in \mathcal{O}$, $\lambda_{t,i}$ (for $i = 0, 1, 2, 3$), $\phi_{t,i}$ (for $i = 1, 2,$), $\omega_{t,p}$ (for $p = 1, 2, \ldots, 7$), $\gamma_{t,q,i}$ (for $q = 1, 2, 3, 4$ and $i = 1, 2$), $\tau_{t,o,k,i}$ (for $k = 1, 2$ and $i = 1, 2, 3, 4$) and $\sigma_{t,o,k,i}$ (for $k = 1, 2, \ldots, k'$ and $i = 1, 2$) are the parameters to learn, $\varepsilon_{d,t}$ and $\varepsilon_{d-1,t}$ are the unexpected changes in load on the previous day and week, and $\varepsilon_{d,t}$ is the expected error of the forecast model.

The model described in (1) presents several advantages. First, the parameters to learn are linearly combined to form an estimate of the forecased load, making it easy to interpret. More precisely:

- $\lambda_{t,0}$ explains the basic expected load for the given time step, $\lambda_{t,1}$ the contribution of the previous week load, $\lambda_{t,2}$ that of the last observed load and $\lambda_{t,3}$ the contribution of the previous time step load;
- $\phi_{t,1}$ and $\phi_{t,2}$ are used to take into account the previous day and previous week errors;
- $\omega_{t,p}$, for $p = 1, 2, \ldots, 7$, addresses week seasonality, while $\gamma_{t,q,1}$ and $\gamma_{t,q,2}$, for $q = 1, 2, 3, 4$, capture the annual, semi-annual, quadrimester and quaterly seasonality;
- $\tau_{t,o,k,1}$ and $\tau_{t,k,3}$ explain the effect of heating of the current day ($o = 0$) and the previous day ($o = -1$), while $\tau_{t,o,k,2}$ and $\tau_{t,o,k,4}$ explain the effect of cooling;
- $\sigma_{t,o,k,1}$ and $\sigma_{t,o,k,2}$ explain the effect on the electrical power load on special days or on the days following a special day.

Secondly, it can be efficiently trained using an algorithm called *iterated ordinary least square* [19], which proceeds as follows:

1) First, we estimate, with a regular linear regression, each equation by ignoring the moving-average error terms and by storing the residuals;
2) Next, we extend the input with the residuals and lagged residuals as estimates of the error;
3) Then we estimate the complete model;
4) We repeat the two previous steps until numerical convergence is attained on the learned parameters.

It is important to notice that the algorithm is not guaranteed to converge, for instance, if the number of parameters to estimate is too large, the time series is too short or if the time series exhibits nonstationary types of variations [19]. Therefore, for the sake of robustness, we set a threshold of 10^{-20} for numerical convergence and a maximal number of iterations of 100. In practice, in our experiments, the models converged quickly in almost every case, and even when it was not the case, the forecasted loads did not appear to be of low quality in comparison with the other ones. We also provide more detail in Section IV about how one can parallelize the training to fully take advantage of high performance computing clusters.

Finally, the accuracy of the forecast model is quite good. Clements et al. [3] report an impressive 1.36% mean absolute percentage error (MAPE) over a period of 11 years for the Queensland region of Australia. Although we do not succeed in reproducing the same score, we obtain a respectable final MAPE score of about 1.79% for the residential and commercial power loads of the complete Quebec territory.

III. Extending the Reference Model

In this section, we introduce small extensions, or variations, of the model of Clements et al. [3]. The simple nature of 1 makes the process quite easy. But first, we need to define some additional random variables. Let $\mathcal{L}$ be a set of fixed geographical locations. For any time step t of day d and any location $\ell \in \mathcal{L}$, let

- $T_{d,t}^{\ell}$ be the temperature (in Celsius) at location ℓ on time step t of day d;

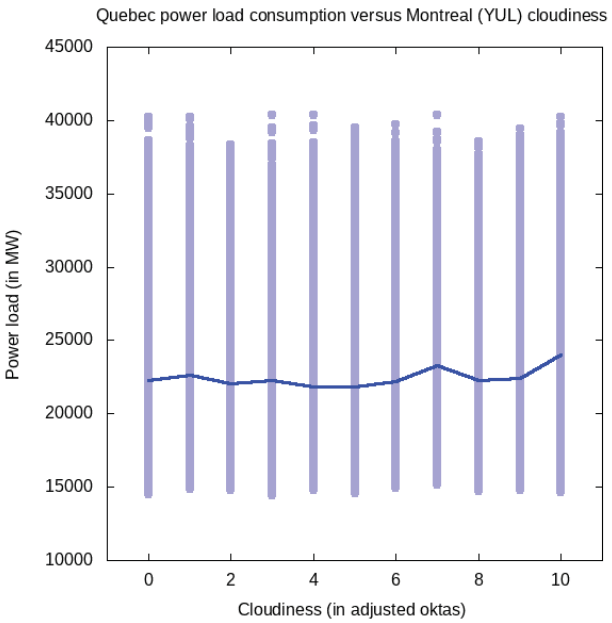

Fig. 2 Comparison of Quebec hourly power load against cloudiness (measured in Montreal) between 2012-01-01 and 2019-12-31

- $N_{d,t}^{\ell}$ be the cloudiness (in oktas) at location ℓ on time step t of day d;
- $W_{d,t}^{\ell}$ be the wind speed (in km/h) at location ℓ on time step t of day d and
- DST_d indicate whether day d is on daylight saving time (DST).

In other words, we introduce four changes: (1) We take into account *cloudiness* (or cloud cover), which measures the fraction of the sky obscured by clouds at a given time, from a particular location [20]. (2) We also consider the *wind speed* measured at a given time and place. (3) We allow the use of multiple meteorological locations in order to take into account geographical variations. (4) We allow the separation between days using the DST and those which are not.

Indeed, as shown in Figs. 2 and 3, both cloudiness and wind speed show some correlation with the power load consumption [1]. More precisely, there does not seem to be any significant variation for cloudiness between values in range $[0, 6]$, but there are two small peaks at 7 and 10 that could explain some variability in the power load consumption: It seems reasonable to assume that if the sky is almost or completely covered, then we might observe a slight increase of load due to the increase in lighting devices use. The effect is not expected to be as high as the one observed for heating and cooling devices, since lighting is not energy-intensive, but should nevertheless be significant. As for the wind speed, we notice a slight increase in power load consumption as the speed grows, which could be explained by the wind chill factor, when combined with cold temperatures. The irregular variations above 50 km/h should be ignored since there are not many data points available in the case of very high winds.

As for daylight saving time, although we do not expect it to bring significant changes, it draws a clean separation between

Summer and Winter periods, and it should also make forecasts more reliable during the transition dates.

Next, we must update the heating and cooling variables accordingly. For any day d and time step t of the day, let

$$\begin{aligned} H^{\ell}_{d,t,1} &= \textsc{ClRamp}^{-}_{-23,13}(T^{\ell}_{d,t}) \\ H^{\ell}_{d,t,2} &= \textsc{ClRamp}^{-}_{-23,1}(T^{\ell}_{d,t}) \\ C^{\ell}_{d,t,1} &= \textsc{ClRamp}^{+}_{21,33}(T^{\ell}_{d,t}) \\ C^{\ell}_{d,t,2} &= \textsc{ClRamp}^{+}_{28,33}(T^{\ell}_{d,t}), \end{aligned}$$

where $\textsc{ClRamp}^{-}$ and $\textsc{ClRamp}^{+}$ have the same meaning as in Section II. We use the same breakpoints than before, whatever the location. Indeed, examination of the correlations between temperatures and power load for different locations dit not reveal any significant differences. To take into account the nonlinear relations between cloudiness/wind speed and load, we define the following variables:

$$\begin{aligned} \overline{N}^{\ell}_{d,t,1} &= \textsc{ClRamp}^{+}_{0,3}(N^{\ell}_{d,t}) \\ \overline{N}^{\ell}_{d,t,2} &= \textsc{ClRamp}^{+}_{3,10}(N^{\ell}_{d,t}) \\ \overline{N}^{\ell}_{d,t,3} &= \textsc{ClRamp}^{+}_{9,10}(N^{\ell}_{d,t}) \\ \overline{W}^{\ell}_{d,t} &= \textsc{ClRamp}^{+}_{12,39}(W^{\ell}_{d,t}) \end{aligned}$$

Finally, let $\mathcal{O} = \{0, -1, -7\}$ the set of allowed days offsets. Then the updated equation becomes

$$\begin{aligned} L_{d,t} = \; & \lambda_{t,0} + \left(\sum_{p=1}^{7} \omega_{t,p} W_p\right) L_{d-1,t} \\ & + \left[\lambda_{t,1} + \sum_{q=1}^{4} \mathrm{SC}(\gamma_{t,q,1}, \gamma_{t,q,2}, Y_{d,t,q})\right] L_{d-7,t} \\ & + \lambda_{t,2} L_{d-1,\bar{t}-1} + \lambda_{t,3} L_{d,t-1} \\ & + \phi_{t,1}\varepsilon_{d-1,t} + \phi_{t,2}\varepsilon_{d-7,t} + \varepsilon_{d,t} \\ & + \sum_{o\in\mathcal{O}} \sum_{\ell\in\mathcal{L}} \sum_{k=1}^{2} \tau^{\ell}_{t,o,k,1} H^{\ell}_{d+o,t,k} \\ & + \sum_{o\in\mathcal{O}} \sum_{\ell\in\mathcal{L}} \sum_{k=1}^{2} \tau^{\ell}_{t,o,k,2} C^{\ell}_{d+o,t,k} \\ & + \sum_{o\in\mathcal{O}} \sum_{\ell\in\mathcal{L}} \sum_{k=1}^{3} \nu^{\ell}_{t,o,k} \overline{N}^{\ell}_{d+o,t,k} \\ & + \sum_{o\in\mathcal{O}} \sum_{\ell\in\mathcal{L}} \omega^{\ell}_{t,o} \overline{W}^{\ell}_{d+o,t} \\ & + \sum_{o\in\mathcal{O}} \sum_{k=1}^{k'} \sigma_{t,o,k} S_{d+o,k} + \sum_{o\in\mathcal{O}} \delta_o \mathrm{DST}_{d+o}, \end{aligned} \tag{2}$$

where the parameters $\lambda_{t,i}$, $\gamma_{t,q,i}$, $\phi_{t,i}$ and $\sigma_{t,o,k}$ have the same meaning as in (1), and, for $t = 1, 2, \ldots, \bar{t}$, $o \in \mathcal{O}$ and $\ell \in \mathcal{L}$, $\tau^{\ell}_{t,o,k,i}$ (for $k = 1, 2$ and $i = 1, 2$), $\nu^{\ell}_{t,o,k}$ (for $k = 1, 2$) and $\omega^{\ell}_{t,o}$ are the new parameters to learn, explaining respectively the effect of temperature, cloudiness and wind speed on the power load, for each location, and, finally, δ_o is the parameter that explains the effect of using daylight saving time.

It is worth mentioning that we allowed ourselves to use the letters W and ω both for the weekly seasonality and the wind speed: Since the number of superscripts and subscripts is distinctive in both cases, there should be no ambiguity.

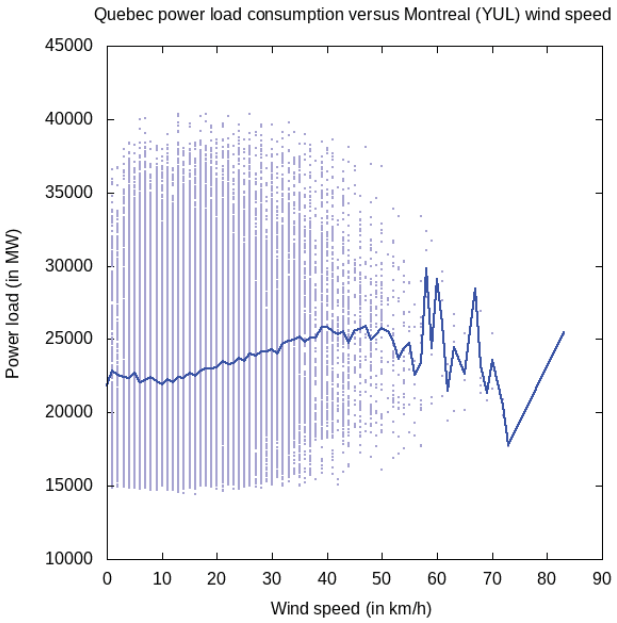

Fig. 3 Comparison of Quebec hourly power load against wind speed (measured in Montral) between 2012-01-01 and 2019-12-31

IV. Experimental Setting

The model and its variations described respectively in Sections II and III were implemented in a C++ prototype, based on the *xtensor* library, an open-source matrix computation framework inspired from NumPy [21]. Also, to facilitate experimentation, a command line interface is provided to allow the addition or the removal of any specific given feature. More precisely, the following parametrization is available to the user, with sensible default values:

- the horizon duration (default: 24 hours);
- the maximum number of iterations allowed during training (default: 100);
- the acceptable threshold for numerical convergence (default: 10^{-8});
- any number of paths to CSV files containing meteorological (temperature, cloudiness and wind speed) data, from one or many different locations;
- the heating breakpoints (default: $[-23, 1]$ and $[-23, 13]$), the cooling breakpoints (default: $[21, 33]$ and $[28, 33]$), the cloudiness breakpoints (default: $[0, 3]$, $[3, 10]$ and $[9, 10]$) and the wind speed break points (default: $[12, 39]$) to consider;
- whether to take into account meteorological data on the previous day and/or on the previous week (default: only the current day);
- whether to take into account the daylight saving time (default: yes);
- the year harmonics to use (default: $1, 2, 3, 4$).

- whether to use the last observed load as input (default: yes);
- whether to chain the models between each time step (default: yes).

To evaluate the performance of the model, we used a 8-years dataset of hourly power load for the complete province of Quebec, ranging from January 1st, 2012 to December, 31st, 2019. We also retrieved the corresponding meteorological data (temperature, cloudiness and wind speed) for the same range of dates and same time steps, for five locations: Montreal (`yul`), Quebec city (`qc`), Ottawa (`ott`), Sherbrooke (`sh`) and Baie-Comeau (`bc`). Those locations seemed strategically sound to ensure wider geographical cover with relatively high population density, except for Baie-Comeau, in the North Coast administrative region, which is mostly included in the hope to capture power consumption behavior in the northern parts of Quebec. In each case, the MAPE score is computed using a classical rolling window strategy, i.e. for each day d between 2013 and 2019,

- We train the model using all data between day 0 and day $d-1$.
- We produce a forecast for the day d.
- We store the forecasted values together with the real power loads observed.

We can then use all stored value to evaluate the global MAPE score, or to study it for each time step or the day, for each days of the week, and so on.

In order to find a set of parameters as best as possible, we proceed as follows. First, we start with the basic model of Clements et al. [3], but with the meteorological and special days components removed. For the next steps, we look at the improvement obtained on the global MAPE score by introducing a simple variation on each available parameter, such as using the daylight saving time, enhancing the number of year harmonics considered, adding meteorological data from a specific station, etc. After having compared the different variations, we pick the one that shows the largest improvement, and we consider it fixed for the next experiment. We repeat the process until no significant improvement is observed.

As mentioned earlier, the training of a model for a specific date is done very quickly, whatever the date and the activated parameters. It takes a few seconds on a standard laptop. However, the evaluation of several variations of the model, over a period of 7 years can take dozens of hours if computed sequentially. Since a high performance computing cluster is available on premises, we introduced the following straightforward parallelizations:

- On a given day, each time step of the day, the model can be trained independently. Only the forecasts must be computed sequentially, but since the training is much longer than the forecast process, significant computation time can be gained.
- The evaluations of the models can also be distributed between the days of the evaluation range.
- Finally, each model variation can also be trained independently.

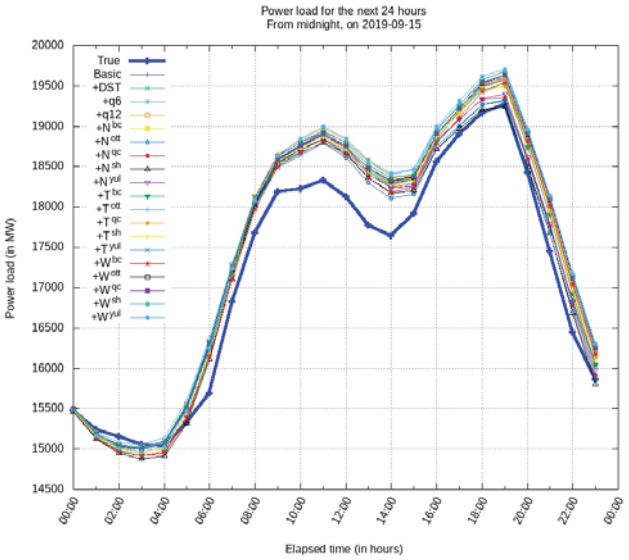

Fig. 4 Comparison of the forecasted values by different variations of the basic reference model on an arbitrary typical day (September 25, 2019)

Exploiting those "embarassingly" parallel workload observations, we were able to reduce the computation time of each step to a time between 5 to 10 minutes.

V. Experimental Results

We are now ready to present the results of the carried out experiment. We first started with a *basic* or *reference model*, i.e. the model described in ((1)), but with the temperature and special days components removed. In other words, no meteorological information is used: Only the week and year seasonality are taken into account, as well as the last observed load and the previous time step forecasted load. This seems like a natural starting point: As it is based only on the power load history, one could apply this model without additional change and to any scale, ranging from single household smart meters and off-grid power systems to single electrical substations and the complete power grid. Then, we introduced 19 atomic variations of the model:

- One for each of the 5 meteorological locations and each of the 3 meteorological variables (temperature, cloudiness and wind speed); Moreover, we only consider those variables on the current day and not for the previous day or week;
- One activating the use of the daylight saving time;
- One activating the 2-months year harmonic ($q = 6$) and
- One activating the 1-month year harmonic ($q = 12$).

Fig. 4 depicts a comparison between the reference model, the 18 variations and the real observed load for an arbitrary typical day (September 25, 2019). For this particular day, the models are quite accurate during the start/end of the day and for the largest peak, but they show over-forecasting errors both for the morning peak and the middle day trough.

Next, we evaluated the average MAPE score of each of the variations for the time period ranging from 2013-01-01 to 2019-12-31. Fig. 5 shows the average MAPE for each

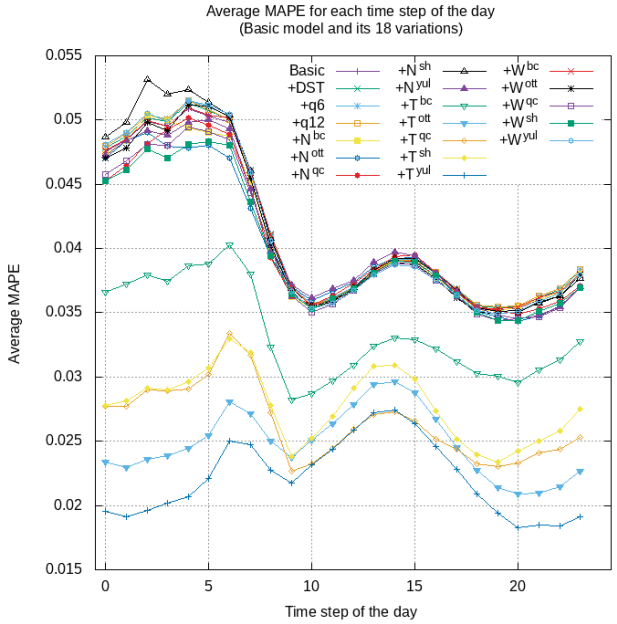

Fig. 5 Comparison of the average MAPE score obtained for the basic model and each of its 18 variations

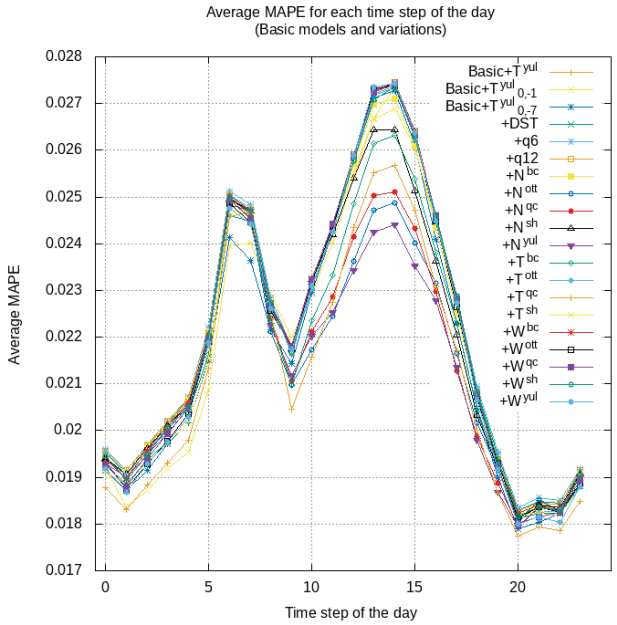

Fig. 6 Comparison of the average MAPE score obtained for the new reference model and each of its variations

time step of the day and each variations. Without much surprise, it is the temperature measured in Montreal — the most populated city of the Quebec province — that shows the greatest improvement, both on the global MAPE score and the average MAPE for each time step of the day. Thus, the new reference model becomes the basic one with the additional $T^{\texttt{yul}}$ meteorological variable (on the current day of forecast).

Based on the new reference model, we repeated the same idea as below, with the additional variation of considering the meteorological variables on the previous day and on the previous week. Fig. 6 shows the obtained comparison. At that stage, there is no clear candidate that stands out: the cloudiness measured in Montreal ($N^{\texttt{yul}}$) significantly lowers the highest peak, but the temperature measured in Quebec city ($T^{\texttt{qc}}$) shows a greater improvement when averaged on all time steps. In any case, it turns out that the inclusion of both variables improved the overall performance.

Using the same process over and over, we added, in that order, the following variations (the global MAPE score improvement in each case is indicated between parentheses):

1) The current day temperature measured in Montreal ($T^{\texttt{yul}}$) ($\approx -1.88\%$);
2) The current day temperature measured in Quebec city ($T^{\texttt{qc}}$) ($\approx -0.10\%$);
3) The current day cloudiness measured in Montreal ($N^{\texttt{yul}}$) ($\approx -0.10\%$);
4) The inclusion of the meteorological variables on the previous day on top of the current day (-0.08%);
5) The cloudiness measured in Quebec city ($N^{\texttt{qc}}$) ($\approx -0.05\%$);
6) The inclusion of the meteorological variables on the previous week on top of the current day and previous day ($\approx -0.05\%$);
7) The wind speed measured in Montreal ($W^{\texttt{yul}}$) ($\approx -0.03\%$).

From there, we tried several variations by trial-and-error, with only slight improvements. In all cases, including the daylight saving time always improved the forecasts, so we decided to keep it activated. It is unclear whether the additional 2-months (`+q6`) and 1-month (`+q12`) year harmonics really improve the accuracy of the model: depending on the values of the other parameters, we either observed small improvements or small deteriorations. Finally, the meteorological variables for the city of Sherbrooke (`sh`) did not seem to bring more information than the four other cities did so that we decided to remove it from our final model. To summarize, the final preferred model uses the following parameters:

- All meteorological variables are used (temperature, cloudiness and wind speed);
- Four locations (Montreal, Quebec city, Ottawa and Baie-Comeau) are used, Sherbrooke is not considered;
- Meteorological variables on the current day, the previous day and the previous week are taken into account;
- The daylight saving time is used and
- The 1-month harmonic ($q = 12$) is used on top of the other four ones ($q = 1, 2, 3, 4$).

The global average MAPE score for our final model is $\approx$ 1.79%. The average MAPE scores for each day of the week are depicted in Fig. 7. The forecasts are typically better on Wednesdays, Thursdays and Fridays. Some accuracy is lost on Tuesdays, Saturdays and Sundays, but are relatively acceptable. Since there is usually over-forecasting on Saturdays compared to Fridays, this could explain the decrease in quality. However, the score for Mondays are quite bad. This result is probably explained by the fact that many special days fall on that specific weekday combined with the effect of moving from the weekend to the week and not taking into

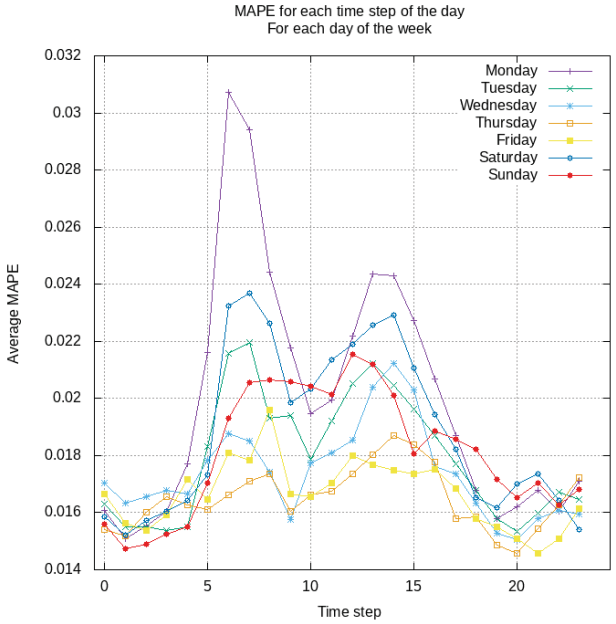

Fig. 7 MAPE score by time step for each day of the week

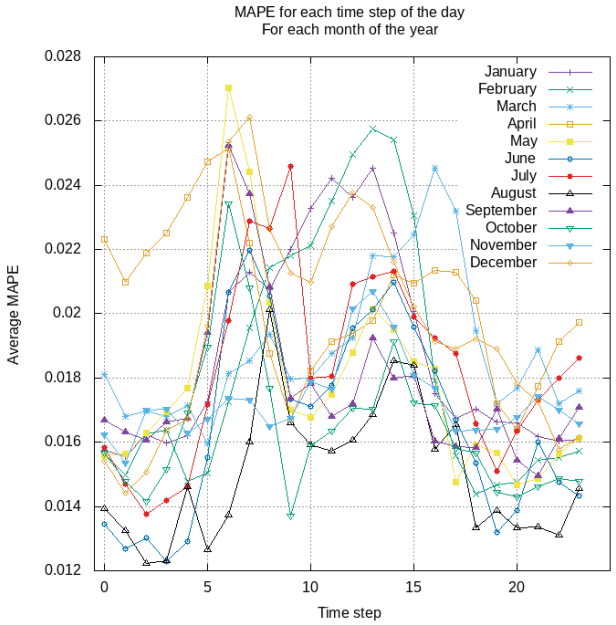

Fig. 8 MAPE score by time step for each month of the year

account the 2-days lagged demand (from Saturday to Monday). Finally, the average MAPE scores for each month of the year are depicted in Fig. 8. The patterns are less clear in that case:

- In general, the forecasts during August, September, October and November seem better than for the other months;
- April performs poorly during the first hours in comparison with the other months. It is unclear why this is the case.
- The forecasts are more error-prone for the Winter months (December, January and February, March), which is probably explained by the higher volatility of the meteorological conditions.
- February is quite accurate for the morning and evening peaks, but shows relatively poor performance during the middle day.

We believe that a better categorization of "day type" could explain part of those remaining variations.

VI. Concluding Remarks

In this paper, we presented an extended multiple equation time series model to address short-time load forecasting. We studied the inclusion of three meteorological variables (temperature, cloudiness and wind speed), as well as multiple meteorological stations. All turned out to improve significantly the accuracy of the forecast models. Moreover, given the fact that training of the model is fast, it should be quite easy to adapt our C++ prototype so that it also provides confidence intervals on the forecasts, for instance by training the model on the previous days and on similar days such as those in the same period of the previous year.

We conclude with a short discussion on how to handle special days. It is well known that the power load is generally difficult to forecast around holidays [2]. The global load during the day, as well as the peaks, are typically lower, but there can also be unexpected higher demand during some specific time steps. Unfortunately, it is not always obvious how to identify what day is really special and how special it is going to turn out.

In the spirit of generalizing the handling of special days, we computed an "error profile" for each day, by taking the absolute percentage error (APE) for each time step of that day, effectively associating a time series $t(d)$ with each day d. Then, using Scikit-learn's cluster library [22], we computed clusters for those day profiles. Fig. 9 illustrates the clusters obtained according to five methods: affinity propagation [22], spectral clustering [23], Ward's agglomerative clustering method [22], BIRCH [24] and MiniBatch k-means [25]. Other methods were tested, but these five yielded the most visually appealing distributions of the day profiles. For instance, taking the affinity propagation method:

- Clusters 4 and 7 seem to regroup typical days;
- Clusters 1, 9, 10 and 12 contain days during which almost every time step presents an under-forecasted error;
- Clusters 5 and 8 contains days during which almost every time step presents an over-forecasted error;
- Cluster 2 shows an increasingly under-forecasting model as the day passes;
- Cluster 3 presents over-forecasting during the morning peak;
- Cluster 6 regroup days during which there seems to be over-forecasting during the middle of the day and
- Cluster 11 has a bad start during the first hours, but becomes better for the rest of the day.

It is unclear whether the identification of the "kind" of day would significantly improve the forecasts, but it seems to be worth further exploration.

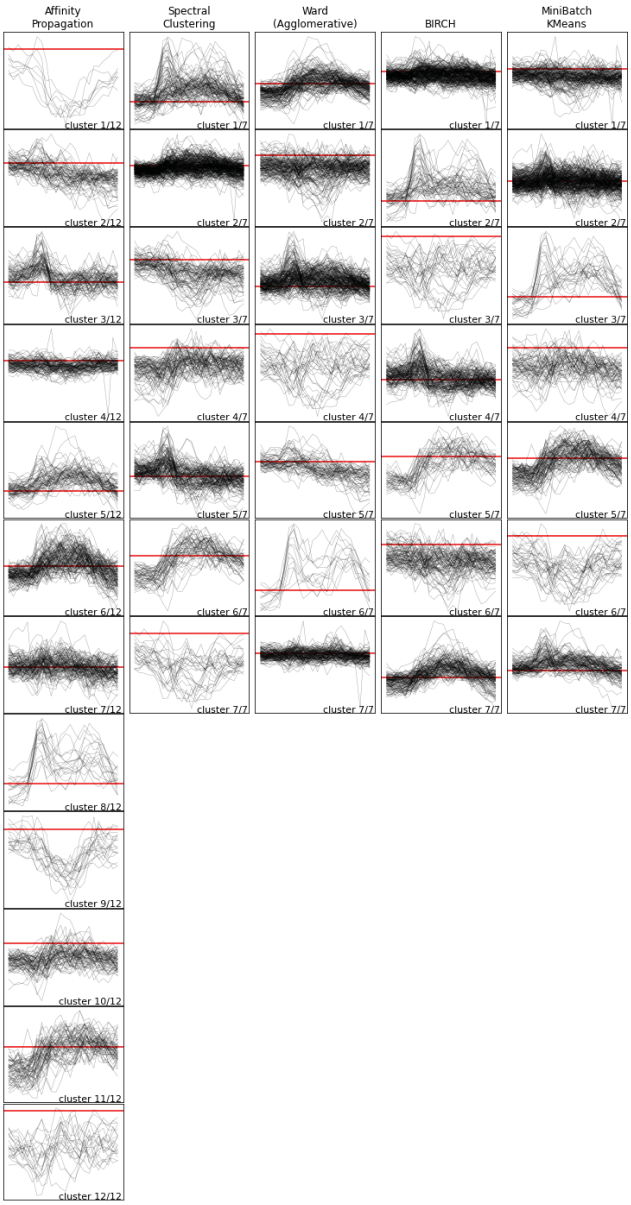

Fig. 9 Day profiles clusters computed by different strategy (the horizontal red line identifies a null error)

ACKNOWLEDGMENT

At the exception of Fig. 9, all figures were produced using the open-source software Gnuplot [26].

REFERENCES

[1] M. Grenier, "Short-term load forecasting at hydro-québec transénergie," in *2006 IEEE Power Engineering Society General Meeting*. IEEE, 2006, pp. 5–pp.

[2] J. R. Cancelo, A. Espasa, and R. Grafe, "Forecasting the electricity load from one day to one week ahead for the spanish system operator," *International Journal of forecasting*, vol. 24, no. 4, pp. 588–602, 2008.

[3] A. E. Clements, A. Hurn, and Z. Li, "Forecasting day-ahead electricity load using a multiple equation time series approach," *European Journal of Operational Research*, vol. 251, no. 2, pp. 522–530, 2016.

[4] M. Rafiei, T. Niknam, J. Aghaei, M. Shafie-Khah, and J. P. Catalão, "Probabilistic load forecasting using an improved wavelet neural network trained by generalized extreme learning machine," *IEEE Transactions on Smart Grid*, vol. 9, no. 6, pp. 6961–6971, 2018.

[5] C. Feng and J. Zhang, "Assessment of aggregation strategies for machine-learning based short-term load forecasting," *Electric Power Systems Research*, vol. 184, p. 106304, 2020.

[6] M. Santamouris, C. Cartalis, A. Synnefa, and D. Kolokotsa, "On the impact of urban heat island and global warming on the power demand and electricity consumption of buildings—a review," *Energy and buildings*, vol. 98, pp. 119–124, 2015.

[7] V. Kondaiah, B. Saravanan, P. Sanjeevikumar, and B. Khan, "A review on short-term load forecasting models for micro-grid application," *The Journal of Engineering*, 2022.

[8] S. Maldonado, A. González, and S. Crone, "Automatic time series analysis for electric load forecasting via support vector regression," *Applied Soft Computing*, vol. 83, p. 105616, 2019.

[9] J. Wang and J. Hu, "A robust combination approach for short-term wind speed forecasting and analysis–combination of the arima (autoregressive integrated moving average), elm (extreme learning machine), svm (support vector machine) and lssvm (least square svm) forecasts using a gpr (gaussian process regression) model," *Energy*, vol. 93, pp. 41–56, 2015.

[10] H. S. Hippert, C. E. Pedreira, and R. C. Souza, "Neural networks for short-term load forecasting: A review and evaluation," *IEEE Transactions on power systems*, vol. 16, no. 1, pp. 44–55, 2001.

[11] X. Zhang and J. Wang, "A novel decomposition-ensemble model for forecasting short-term load-time series with multiple seasonal patterns," *Applied Soft Computing*, vol. 65, pp. 478–494, 2018.

[12] M. Aouad, H. Hajj, K. Shaban, R. A. Jabr, and W. El-Hajj, "A cnn-sequence-to-sequence network with attention for residential short-term load forecasting," *Electric Power Systems Research*, vol. 211, p. 108152, 2022.

[13] C. Feng, M. Sun, and J. Zhang, "Reinforced deterministic and probabilistic load forecasting via q-learning dynamic model selection," *IEEE Transactions on Smart Grid*, vol. 11, no. 2, pp. 1377–1386, 2019.

[14] R.-J. Park, K.-B. Song, and B.-S. Kwon, "Short-term load forecasting algorithm using a similar day selection method based on reinforcement learning," *Energies*, vol. 13, no. 10, p. 2640, 2020.

[15] D. Wu, C. Cui, and B. Boulet, "Residential short-term load forecasting via meta learning and domain augmentation," in *IFIP International Workshop on Artificial Intelligence for Knowledge Management*. Springer, 2021, pp. 184–196.

[16] W. Lin, D. Wu, and B. Boulet, "Spatial-temporal residential short-term load forecasting via graph neural networks," *IEEE Transactions on Smart Grid*, vol. 12, no. 6, pp. 5373–5384, 2021.

[17] G. Joshi, R. Walambe, and K. Kotecha, "A review on explainability in multimodal deep neural nets," *IEEE Access*, vol. 9, pp. 59 800–59 821, 2021.

[18] R. Ramanathan, R. Engle, C. W. Granger, F. Vahid-Araghi, and C. Brace, "Short-run forecasts of electricity loads and peaks," *International journal of forecasting*, vol. 13, no. 2, pp. 161–174, 1997.

[19] H. Spliid, "A fast estimation method for the vector autoregressive moving average model with exogenous variables," *Journal of the American Statistical Association*, vol. 78, no. 384, pp. 843–849, 1983.

[20] R. E. Huschke *et al.*, "Glossary of meteorology," 1959.

[21] J. Mabille, S. Corlay, and W. Vollprecht, "xtensor: A C++ library for multi-dimensional arrays with broadcasting and lazy computing," 2016, https://github.com/xtensor-stack/xtensor.git.

[22] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[23] X. Y. Stella and J. Shi, "Multiclass spectral clustering," in *Computer Vision, IEEE International Conference on*, vol. 2. IEEE Computer Society, 2003, pp. 313–313.

[24] T. Zhang, R. Ramakrishnan, and M. Livny, "Birch: an efficient data clustering method for very large databases," *ACM sigmod record*, vol. 25, no. 2, pp. 103–114, 1996.

[25] D. Sculley, "Web-scale k-means clustering," in *Proceedings of the 19th international conference on World wide web*, 2010, pp. 1177–1178.

[26] T. Williams, C. Kelley, and many others, "Gnuplot 5.2.8: An interactive plotting program," http://gnuplot.sourceforge.net/, December 2019.