



Deep Learning-Based Multivariate Probabilistic Forecasting for Short-Term Scheduling in Power Markets

Jean-François Toubeau , *Student Member, IEEE*, Jérémie Bottieau , *Student Member, IEEE*, François Vallée, *Member, IEEE*, and Zacharie De Grève, *Member, IEEE*

Abstract—In the current competition framework governing the electricity sector, complex dependencies exist between electrical and market data, which complicates the decision-making procedure of energy actors. These must indeed operate within a complex, uncertain environment, and consequently need to rely on accurate multivariate, multi-step ahead probabilistic predictions. This paper aims to take advantage of recent breakthroughs in deep learning, while exploiting the structure of the problem to design prediction tools with tailored architectural alterations that improve their performance. The method can provide prediction intervals and densities, but is here extended with the objective to generate predictive scenarios. It is achieved by sampling the predicted multivariate distribution with a copula-based strategy so as to embody both temporal information and cross-variable dependencies. The effectiveness of the proposed methodology is emphasized and compared with several other architectures in terms of both statistical performance and impact on the quality of decisions optimized within a dedicated stochastic optimization tool of an electricity retailer participating in short-term electricity markets.

Index Terms—Bidirectional LSTM, copula, multi-step ahead prediction, probabilistic forecasting, scenario generation.

I. INTRODUCTION

THE deregulation of power markets, which has introduced competition at both generation and retail levels, associated with a strong will to move towards decarbonisation of the overall energy system, has facilitated the development of renewable-based generation, typically from wind and photovoltaic (PV) sources. The intermittent and uncertain nature of these resources is challenging the traditional operation of electricity networks, which gives rise to complex stochastic optimization problems, not only for system operators but also for all the other players in the electricity sector. In this context, the success of their operational planning strategies, and as a corollary of the energy transition, strongly relies on the knowledge of the system state

(through adequate prediction tools) at time horizons which go from quasi real-time to day-ahead [1], [2].

However, predictions over such time horizons are ineluctably vitiated by errors, originating from noise in the explanatory variables (e.g., due to the chaotic nature of weather conditions) as well as model misspecifications. Hence, traditional point (deterministic) forecasts that only predict the conditional mean of the signal are providing very limited information to decision-makers. Indeed, in order to ensure decisions that are robust with regard to forecast errors and unexpected events, it is also necessary to quantify the level of uncertainty associated with predictions.

Specifically, techniques such as robust, interval and chance-constrained optimization frameworks were developed to hedge against these uncertainties by relying on probabilistic forecasts in the form of intervals. In this way, different approaches for obtaining such uncertainty regions (or by extrapolation densities) can be found in the literature for respectively wind power [3]–[7], PV generation [8], load [7], [9] and electricity prices [10], [11]. However, these optimization techniques have two main drawbacks. Firstly, robust and interval techniques are known to yield conservative (and thus sub-optimal) solutions since these are intrinsically designed to be optimal with regard to extreme scenarios [12]. Chance-constrained offers a less conservative and more practical approach by considering a probability for satisfying each constraint, but such a formulation is very difficult to solve in practice (due to the non-convexity of the resulting problem). Secondly, since probabilistic forecasts provide no representation of the interdependencies between consecutive time steps (i.e., no information regarding the correlation of the variables of interest at different time points is available), the quality of subsequent decisions may be affected.

Consequently, for time-dependent decision problems that have to be carried out on a regular basis (e.g., daily participation to electricity markets), scenario-based stochastic optimization provides a practical framework that yields efficient outcomes (less conservatives) in general [13]. But this technique, which optimizes the expectation of some loss function (e.g., profit of an electricity retailer) under the forecast distribution, can be associated with tractability issues, depending on the number of scenarios (time trajectories) used to represent uncertainties. In this respect, implementing a methodology able to provide a limited set of representative scenarios is highly valuable.

Manuscript received February 12, 2018; revised June 2, 2018 and July 18, 2018; accepted August 19, 2018. Date of publication September 13, 2018; date of current version February 18, 2019. This work is being supported by Public Service of Wallonia (Belgium), within the framework of the Smartwater project. Paper no. TPWRS-00208-2018. (*Corresponding author: Jean-François Toubeau.*)

The authors are with the Electrical Power Engineering Unit, University of Mons, Mons 7000, Belgium (e-mail: Jean-Francois.TOUBEAU@umons.ac.be; Jeremie.bottieau@umons.ac.be; Francois.VALLEE@umons.ac.be; zacharie.DEGREVE@umons.ac.be).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TPWRS.2018.2870041

This problem is tackled in [14], where an autoregressive moving average (ARMA) model is developed for individual wind sites, and a stationary variance-covariance matrix is thereafter used for integrating the spatial correlation among series. However, this approach does not allow to properly take into consideration the predictive densities, i.e., uncertainty associated with particular conditions (e.g., higher uncertainty during strong winds). In [15]–[17], the scenarios are constructed by computing the complex covariance matrix based on a multivariate Gaussian distribution assumption. In [18], a nonparametric neural network dedicated to quantify prediction intervals (PIs) is firstly implemented. Then, these PIs are used to estimate an empirical cumulative distribution function, from which scenarios are generated. But, due to the independent nature of the sampling methodology, these scenarios do not account for the time-varying structure of forecasts errors. In [19], [20], the conditional error distributions are constructed based on point forecasts, and a copula-based sampling method. This methodology allows to capture the spatial dependencies among the different variables but do not exploit the information contained in the explanatory variables, which can lead to an erroneous estimation of the uncertainty space. In this way, these techniques are unable to differentiate different regimes such as, for instance, highest forecast errors during windy days.

In this work, however, accurate probabilistic forecasting tools (whose predictions properly process the variance contained in the input vector) are combined with a copula sampling so as to generate scenarios that comply with both the predictive distribution of variables (i.e., variability of the uncertainty) and their interdependence structure. The methodology is thus composed of two consecutive steps:

- Determination of probabilistic forecasts for each variable of the whole prediction horizon. It should be noted that these intervals can directly be used as inputs for robust optimization.
- Sampling of the resulting multivariate distribution using a copula-based approach to obtain a set of stochastic scenarios that can then be exploited in scenario-based optimization frameworks.

Regarding probabilistic forecasts, the objective of the paper is to exploit recent breakthrough in the field of data science by using advanced deep learning structures, i.e., the Long Short Term Memory (LSTM) neural networks, which is a particular type of recurrent architecture with rich dynamics, designed to automatically select and propagate through time the most relevant contextual information. Such models have proved their superiority in many tasks such as load forecasting [21], [22], and are extensively used by major technology companies for products such as Google Translate [23] or speech recognition applications in smartphones [24].

Overall, the contributions of the paper can be summarized as follows.

Firstly, the objective is to take advantage of the specificities of the day-ahead operational planning to design prediction tools with tailored architectural variations that improve their performance. Practically, since the predictions are needed

simultaneously for each time step of the scheduling horizon, LSTM networks are combined with a bidirectional processing of data. The results demonstrate that this approach infers lower forecast errors than traditional techniques, which allows reducing the uncertainty space of the subsequent optimization.

Secondly, two different models for characterizing the uncertainty are compared. In this way, the BLSTM network is trained to either generate a Gaussian [25] or a non-parametric predictive distribution of the dependent variables [26]. It enables to confront the Gaussian assumption of prediction errors with an empirical approach (that makes no assumption on the underlying probability distribution of variables).

Thirdly, although the method can provide prediction intervals and densities, it is here extended with the aim to provide predictive scenarios. Practically, the tool relies on a copula-based sampling of the multivariate forecasted distribution so as to generate time trajectories (sample paths) that mimic actual time and cross-variable dependencies. In this way, whereas most of the literature focuses on individual variables, the proposed approach attempts to exploit information in a multi-dimensional context with heterogeneous data from different natures. Indeed, in the competitive framework governing the current electricity sector, complex dependencies between electrical and market data are taking shape, and it is thus important to implement a strategy that is able to capture this information.

Fourthly, the value of the methodology is compared with other approaches not only in terms of statistical performance, but also regarding the practical impact of the quality of scenarios on the decisions optimized within a scenario-based stochastic optimization tool. Here, the day-ahead multi-market scheduling of electricity aggregators (such as energy retailers or generation companies) is used as a case study.

Moreover, thanks to the self-learning nature of the proposed methodology, minimal manual engineering or data pre-processing is needed. Then, within the objective of quickly and efficiently integrating the new information that is revealed each day, the method is developed such that the models can be dynamically adapted using exclusively the new data. This circumvents the need of retraining the global architecture from scratch with the whole set of historical data. However, an important aspect related to this re-training is to find the extent to which it is optimal to modify the previous optimal architecture, but this research topic is outside the scope of this work.

The rest of the paper is structured as follows. Section II explains and motivates the global architecture of the tool implemented for constructing adequate predictive day-ahead scenarios. In particular, the strategies used for the probabilistic predictions and the subsequent sampling policy to generate predictive multivariate scenarios are thoroughly described. Then, the different tasks that have to be carried out in order to maximize the predictive potential of the tool are presented in Section III. The results illustrating the benefits of the proposed approach with regard to traditional methods such as the multilayer perceptron are presented and discussed in Section IV. Finally, in Section V, appropriate conclusions are reached.

II. METHODOLOGY

The multivariate predictive scenarios are obtained by means of a two-step procedure. Probabilistic forecasts (that consider both the noise in explanatory variables and misspecifications of the prediction model) are first obtained using an efficient architecture of deep recurrent neural networks (Section II.A-B). It should be noted that the distributions pertaining to data of different nature can be either jointly or individually predicted (without any impact on the rest of the methodology). Then, the resulting multivariate distribution (encompassing at this stage both temporal and cross-variable information) is adequately sampled using a statistical copula model that captures the interdependence structure of variables (Section II.C).

A. Probabilistic Forecasting Tool

It has been observed in the past decade that, due to improvement in computer capabilities, data-driven methods outperform the best physical models for prediction tasks, especially for complex nonlinear time sequences when a sufficient amount of historical data is available [27]. Besides, recurrent neural networks (RNNs), i.e., advanced deep learning-based structures that are characterized by architectural features specifically designed to hold relevant information from past inputs, have shown high potential in processing sequentially dependent data. In this regard, they constitute the natural modeling framework for time series prediction. These models do not rely on a mathematical model of predefined complexity but attempt instead to find a natural dependence between inputs and outputs through a self-learning procedure.

However, traditional RNNs are characterized by two limitations. The first problem, widely known as the vanishing gradient problem, is that the back-propagated errors during training either fades or blows up over time due to the multiple gradient calculations associated with the steepest descent algorithm, preventing the model to reliably access time dependencies more than a few time steps long [28]. Secondly, standard RNNs process inputs in temporal order and ignore the information contained in the future context (resulting in an inadequate modeling of backwards dependencies).

The first problem is here tackled by using an alternative (more complex) neural architecture, referred to as Long Short-Term Memory (LSTM), where the information can be propagated through time among consecutive time step within the internal state of the network [29]. In this way, an LSTM layer $l \in [1, N_L]$ is made up of N_h recurrently connected blocks, referred to as memory blocks (or neurons). As represented in Fig. 1, each block has three multiplicative units, known as input, output and forget gates, which can be seen as modules for respectively writing, reading and resetting information. The inputs of each layer l at time t are composed of the outputs of the same layer at the previous time step $s_{t-1}^{(l)}$ as well as the outputs of the layer below $s_t^{(l-1)}$. For the first hidden layer $l = 1$, the bottom layer $s_t^{(l-1)}$ corresponds to the network input variables \mathbf{x}_t .

At each time step, the input gate can be used to memorize new information from either the explanatory variables or the

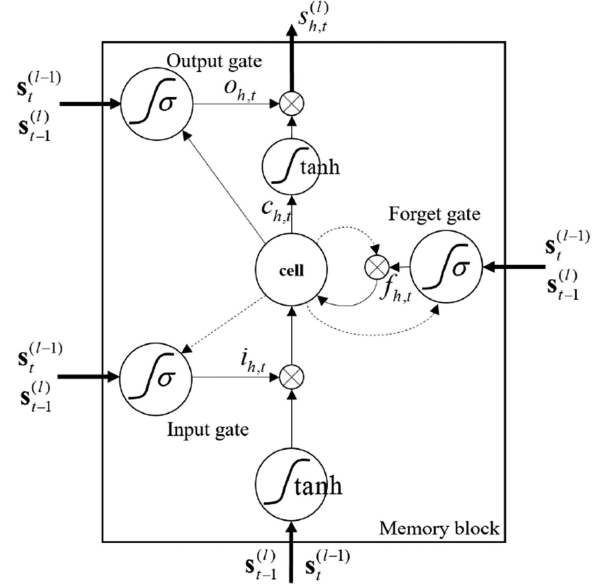


Fig. 1. Single-cell LSTM memory block (cell h of hidden layer l at time t) used in this work.

previous state of the network. The forget gate can, for its part, discard irrelevant information from the past, whereas the output gate is used to exploit useful information from the memory content. Since the information can be either propagated or eliminated simultaneously among the different LSTM blocks of each hidden layer, the neural network is potentially able to model any complex nonlinear signals with multiple time scales, resulting in performance enhancement.

The composite equations associated with the LSTM architecture are the following:

$$\mathbf{i}_t = \sigma \left(\mathbf{W}_{it} \mathbf{s}_t^{(l-1)} + \mathbf{W}_{hi} \mathbf{s}_{t-1} + \mathbf{W}_{ci} \mathbf{c}_{t-1} \right) \quad (1)$$

$$\mathbf{f}_t = \sigma \left(\mathbf{W}_{if} \mathbf{s}_t^{(l-1)} + \mathbf{W}_{hf} \mathbf{s}_{t-1} + \mathbf{W}_{cf} \mathbf{c}_{t-1} \right) \quad (2)$$

$$\mathbf{c}_t = \mathbf{f}_t \mathbf{c}_{t-1} + \mathbf{i}_t \tanh \left(\mathbf{W}_{i\gamma} \mathbf{s}_t^{(l-1)} + \mathbf{W}_{h\gamma} \mathbf{s}_{t-1} \right) \quad (3)$$

$$\mathbf{o}_t = \sigma \left(\mathbf{W}_{io} \mathbf{s}_t^{(l-1)} + \mathbf{W}_{ho} \mathbf{s}_{t-1} + \mathbf{W}_{co} \mathbf{c}_t \right) \quad (4)$$

$$\mathbf{s}_t = \mathbf{o}_t \tanh(\mathbf{c}_t) \quad (5)$$

where σ is the logistic sigmoid function, and \mathbf{i}_t , \mathbf{f}_t and \mathbf{o}_t are the activation vectors of the input gate, forget gate and output gate respectively, whereas \mathbf{c}_t stands for the cell activation vector. All these vectors are of similar size, equal to the one of the hidden vector $\mathbf{s}_{t-1}^{(l)}$ (i.e., output of the hidden layer l). The weight matrices \mathbf{W}_\bullet (connections between LSTM memory blocks) constitute the inner parameters of the neural network that need to be optimized during the learning procedure.

An efficient solution to the second issue is provided by the bidirectional topology, which exploits at each time step t the complete information about the whole temporal horizon (before and after t). As illustrated in Fig. 2, the principle of such bidirectional RNN is to process the training sequence forwards and backwards by two different recurrent networks, both of which

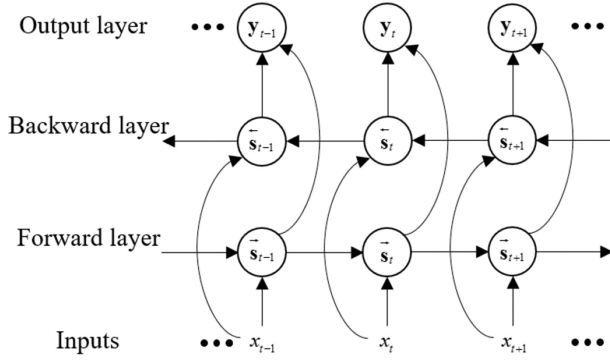


Fig. 2. Bidirectional recurrent neural network.

being connected to the same output vector [30]. The concept has been developed in speech recognition (another very noisy one-dimensional time series) where it was proved that a word can be better recognized with the knowledge of the whole sentence rather than by using only previous words [31]. The same principle is applied here where the information contained in the explanatory variables at each time step is fully exploited for each point of the prediction horizon as it is likely to generate improved performance. This approach is moreover perfectly suited for our task of offline multi-step ahead predictions for which outputs are needed simultaneously at the end of the input segment. Furthermore, bidirectional networks are faster to train, and are more robust to model uncertainties and biased inputs. Indeed, in contrast with unidirectional RNNs, they do not rely on a recursive strategy that iteratively fed back previous predictions as inputs for the next time step, which is shown to lead to error accumulation [32], [33].

Combining bidirectional RNNs with LSTM gives Bidirectional LSTM (BLSTM), which has the benefits of both long-range memory and bidirectional processing. Furthermore, it is possible to take advantage of deep architectures, which are able to build up progressively higher level representations of data, by piling up RNN layers on top of each other (the output sequence of one layer forming the inputs for the next).

B. Network Training

The objective of the BLSTM training is to use the historical datasets (of both explanatory and dependent variables) within a supervised learning strategy that adjusts the model parameters in order to maximize the predictive capability of the tool, while quantifying the uncertainty associated with the predictions. Practically, this consists in finding the optimal weights between neurons (LSTM blocks) so as to determine the full conditional distribution $p(y_{i,t+k} | y_{i,\rightarrow t})$ of outputs $y_{i,t}$ for each k of the prediction horizon (for each variable i), knowing the values from previous time steps.

In contrast with point forecasts, probabilistic predictions yield a representation of the probability distribution of the dependent variables. This distribution can be either obtained using a fully parametrized model or via an empirical function.

1) *Parametric Model of Prediction Errors:* In order to obtain this predictive probability distribution of outputs, the first

investigated procedure is to make an assumption on the distribution of the uncertainty by defining a statistical (parametric) model of forecast errors, and to use the neural network for predicting the parameters of the specified distribution (e.g., mean, variance, skewness, kurtosis, etc.).

The neural network is then trained to optimize a specific error function (with the objective of enhancing the statistical properties of the output distribution of the probabilistic forecast). Practically, the parameters of the network are adjusted to maximize the likelihood $L(\theta)$ that, given its outputs θ , the model generates the historical observations. The maximum likelihood estimation is equivalent to minimizing the negative log-likelihood, and this loss function E_L can be expressed as follows:

$$E_L = - \sum_{t=t_0}^T \ln L(y_{i,t} | \theta(s_t)) \quad (6)$$

Here, the Gaussian likelihood L_G is employed, which is parametrized using the mean and standard deviation of past observations $\theta = (\mu, \sigma)$:

$$L_G = (2\pi\sigma^2)^{-1/2} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right) \quad (7)$$

where x is the actual measurement of the dependent variable. The mean μ of the distribution is given by an affine function of the BLSTM output (8), whereas the standard deviation σ is determined by applying sequentially a softplus activation after the affine transformation, in order to ensure that its value remains strictly positive (9).

$$\mu(s_t) = \mathbf{w}_\mu s_t \quad (8)$$

$$\sigma(s_t) = \ln(1 + \exp(\mathbf{w}_\sigma s_t)) \quad (9)$$

where \mathbf{w}_μ and \mathbf{w}_σ represent the output weight vectors associated respectively with the mean and standard deviation.

It should be emphasized that other likelihood models can be employed, provided that the function derivatives with respect to their parameters θ can be obtained.

2) *Non-Parametric Model of Prediction Errors:* In real-life applications, it may be difficult to know the exact distribution of the uncertainty at hand. In this context, methods that do not rely on a pre-defined distributional assumption are likely to be more robust compared to other parametric methods. A solution consists therefore in using quantile regression [34], for which the objective is to directly predict the specified quantiles $q \in Q$ of the target distribution:

$$q = P\left(y_{t+k} \leq y_{t+k}^{(q)} | y_{\rightarrow t}\right) \quad (10)$$

In this framework, models are trained to minimize the quantile loss (or pinball loss) since it has been proved in [35] that minimizing this pinball loss E_q yields the optimal quantiles. The total loss is therefore the result of the sum over all specified quantiles of interest:

$$E_Q = \sum_{q \in Q} q \max(0, d - y^{(q)}) + (1 - q) \max(0, y^{(q)} - d) \quad (11)$$

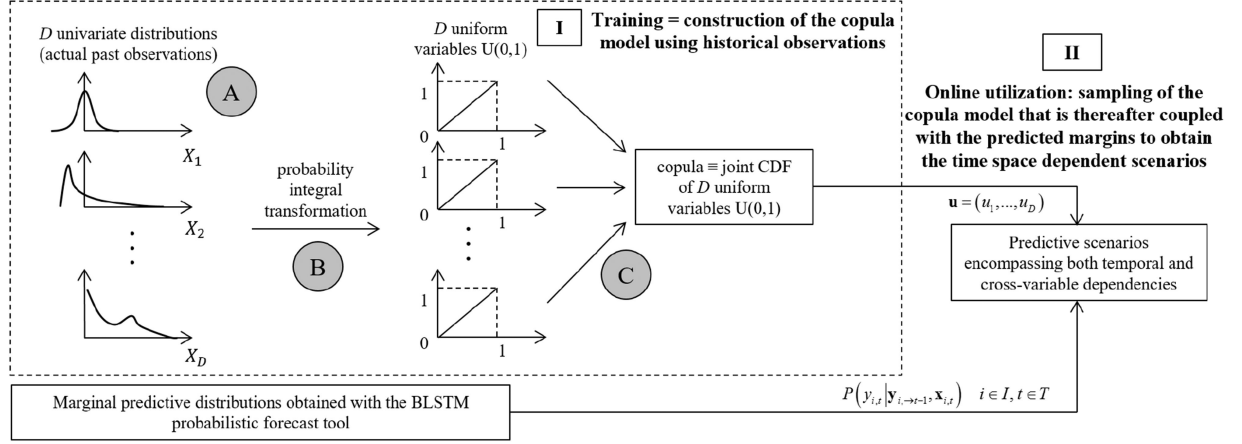


Fig. 3. Generation of predictive scenarios from multivariate distributions.

where the quantiles $y^{(q)}$ are given by an affine function of the BLSTM outputs. It is interesting to notice that, when $q = 0.5$, we get an estimate of the conditional median of the output distribution.

In contrast with deterministic predictions where the neural network has only one output (i.e., the conditional mean of the studied variable), the prediction model is here characterized by a number of outputs equal to the number of quantiles of interest. These quantiles are thus simultaneously predicted by a single neural network architecture.

It should be noted that in case of perfect prediction, the quantile loss cannot be differentiated because of the kink (sharp change of trajectory) of the function at this particular point. Hence, in accordance with the philosophy of the training strategy, it is considered (by forcing numerically in the coded procedure) that the gradient is equal to zero when this case is encountered.

Moreover, similarly to the parametric model previously described, a great asset of the methodology is that the loss function is differentiable, so that the neural network can be trained using gradient descent. This learning procedure (contrary to meta-heuristics such as genetic algorithm or particle swarm optimization) allows the network to be systematically retrained each day using only the new information that has been revealed, so that the computational burden of this retraining task is very limited.

C. Generation of Predictive Scenarios

Once the predictive distributions (either Gaussian or under the form of empirical quantiles) at each time step $t \in T$ for each variable $i \in I$ are obtained (outputs of the BLSTM), the objective is to obtain samples $\mathbf{y}_{i,t_0:T}^s$ from the D -dimensional distribution ($D = \#I$), where $\#$ stands for the cardinality of the associated set. The generated scenarios therefore contains the global dependence structure of variables (12):

$$\mathbf{y}_{i,t_0:T}^s \sim P(\mathbf{y}_{i,t_0:T} | \mathbf{y}_{i,-t_0-1}, \mathbf{x}_{i,t_0:T}) \quad (12)$$

where t_0 stands for the start of the prediction horizon of interest (data before t_0 are therefore assumed to be known for the prediction phase).

However, the task of generating random vectors from a high dimensional distribution is really complex, even when the marginal distributions of each dimension are known [36]–[38].

In this work, the resulting D -dimensional space is therefore statistically represented using a copula model. Such models allow substituting the difficult task of identifying a multivariate distribution by performing two simpler tasks. The first one consists in appropriately modeling the marginal distributions of each variable (i.e., dimension) and the second is to estimate the copula, which summarizes the whole dependence structure [39]. Hence, copulas enable generating vectors with any specific dependence structure, and not only the linear correlation such as traditional methods.

As represented in Fig. 3, sampling the multivariate distribution using a copula-based approach is composed of two sub-steps:

- *Phase I*: in pre-processing (before the probabilistic predictions), a copula model that encompasses the dependence structure of the multivariate distribution is constructed. The model is trained with all the relevant historical information.
- *Phase II*: after obtaining the multivariate distribution, the copula model constructed in phase I can be used to generate numbers $\mathbf{u} = (u_1, \dots, u_D) \in [0, 1]^D$ with the dependence structure of the original data.

The construction of the copula model (Phase I) based on historical data can be described by the following three steps:

- A) The relevant historical data are collected.
- B) These data points are transformed into points u_i of the unit D -cube $[0, 1]^D$ by using the probability integral transformation on univariate marginal distributions.
- C) The density of the copula is estimated.

The copula structure (Step C) is empirically represented as a frequency distribution of historical observations based on the methodology presented in [40], [41]. This nonparametric estimation is parameter-free, thereby bypassing the need to fit samples to a family of copulas and offering a greater generality by allowing any type of dependence. Practically, the unit D -cube is partitioned into a given number of sub-cubes, whose granularity is fixed by the parameter K . In each sub-cube, the density is evaluated by counting the number of historical points

in the sub-cube, divided by the total number of observations n and the volume of the sub-cube.

The empirical copula density, which is denoted by a small letter $c(\mathbf{u})$, is thus determined as follows:

$$c(\mathbf{u}) = \frac{N_{\mathbf{j}}}{n \left(\frac{1}{K}\right)^D}, \quad \mathbf{j} = (j_1, \dots, j_D) \quad (13)$$

where $N_{\mathbf{j}}$ is the number of points in subcube $S_{\mathbf{j}}$, $\mathbf{j} \in \{1, \dots, K\}^D$. Hence, in the original unit D -cube, the cumulative density is equal to one. The marginal densities $c_d(\mathbf{u})$, $d = 1, \dots, D-1$ are determined as follows:

$$c_d(u_1, \dots, u_d) = \int_{u_{d+1}=0}^1 \dots \int_{u_D=0}^1 c(\mathbf{u}) du_{d+1} \dots du_D \quad (14)$$

In this way, the needed conditional distribution functions are expressed as:

$$\begin{aligned} C_d(u_d | u_1, \dots, u_{d-1}) &= \text{Prob}(U_d \leq u_d | U_1 = u_1, \dots, U_{d-1} = u_{d-1}) \\ &= \frac{\int_{u=0}^{u_d} c_d(u_1, \dots, u_{d-1}, u) du}{c_{d-1}(u_1, \dots, u_{d-1})}, d = 2, \dots, D \end{aligned} \quad (15)$$

where $\mathbf{U} = (U_1, \dots, U_D)$ is a random vector with univariate uniform margins restricted to the unit D -cube.

The segmentation of the D -cube into sub-cubes influences both the accuracy of the model and the computational requirements associated with the copula structure. A trade-off between those two considerations has thus to be determined in the choice of the granularity K .

Here, the curse of dimensionality is alleviated, not by focusing on architectural innovations (such as vine copulas for which the high-dimensionality is decomposed into a cascade of bivariate copulas, where each pair-copula can be chosen independently from the others), but rather by relying on an intricate data structure for storing and accessing the needed densities of the empirical copula. Practically, these elements are stored as sparse arrays. Moreover, a hash-based data structure is elaborated so as to realize the more favorable space and time complexity of the data structure, thereby increasing the precision of the copula model by enabling greater values of the parameter K .

Finally, the multivariate random vectors can be generated (Phase II) using copulas in two steps:

- 1) Generate depend random numbers. To that end, u_1^{gen} is firstly sampled from the uniform distribution $U(0,1)$. Then, u_d^{gen} , $d = 2, \dots, D$, are generated in turn using the conditional distribution function $C_d(u_d | u_1^{gen}, \dots, u_{d-1}^{gen})$ given by (15).
- 2) Transform the generated variables of the unit D -cube into the original variables dimension using the inverse transform sampling, based on the marginal distributions coming from probabilistic forecasts.

To summarize, the copula model is trained only once, and can thereafter be used in real-time to generate uniform dependent numbers \mathbf{u} . Using the marginal predictive distributions obtained with the probabilistic forecasting tool, these uniform numbers

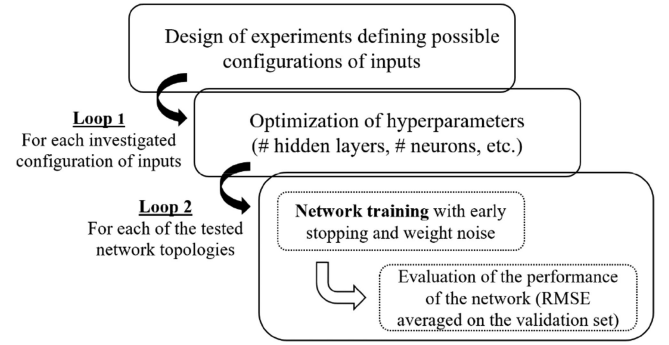


Fig. 4. Two-nested loops procedure designed for the optimal model selection.

can be converted into the original dimensions thanks to the inverse transform sampling so as to obtain the scenarios $\mathbf{y}_{i,t_0:T}^s$. The sampled scenarios thus encompass both time and inter-variable dependencies.

III. CONFIGURATION OF THE PREDICTION TOOL

The predictive capacity of the BLSTM tool depends on two important conditions.

The first one is the selection of the appropriate set of explanatory variables. This task is essential since any missing information will inevitably deteriorate the model ability to provide accurate outcomes, whereas irrelevant input data will, for their part, lead to improper temporal dependencies (due to the strong relationship between the dynamics of the recurrent network and its inputs).

Secondly, the model has to be sufficiently sophisticated for capturing all hidden characteristics of historical data, but not too complex such as to avoid overfitting (i.e., modeling error that arises when a model has insufficient training data and so many parameters that it begins to memorize the data instead of learning the underlying trend, considerably reducing its predictive capacity). It should be mentioned that the complexity of the BLSTM network can be tuned along two dimensions (also referred to as hyperparameters): the number of hidden layers within the network architecture and the number of neurons (LSTM blocks) within each hidden layer.

Finding the optimal architecture (in terms of both inputs selection and model complexity) is task-dependent [42], and is here achieved using to the two-nested loops approach presented in Fig. 4.

First, different relevant configurations of the input vectors are evaluated.

All studied variables (load, renewable generation and electricity prices) are characterized by both daily and yearly cycles (i.e., temporal profiles hold essentially the same shape from one day to the next and from year to year). Additionally, load and electricity prices are also strongly related to human activity, which results in a weekly periodicity. However, in order to optimally exploit this temporal information, the input selection has also to identify the best way to represent the variables. Indeed, the relative importance of these time data is not easily quantified by a numerical value. For instance, the second hour of the day

is not 2 times more important than the first one. In this context, a binary representation may provide a more natural way of expressing such data, but at the expense of an increased dimensionality of the network input space. Here, different inputs combinations were therefore tested. For instance, for describing the hourly variation within the day, the different options were:

- Incremental indexing: a single input in the form of a continuous value within the range $[0.1, 2.4]$.
- Incremental binary representation: 5 inputs representing a binary Gray coding (from '00001' to '10100'). In contrast with the traditional binary representation, the Gray coding is a binary numerical system in which two successive values differ in only one bit, which allows smoother transitions between time steps.
- Mutually exclusive binary representation: 24 binary inputs, one for each hour of the day. With such an input representation, when one input is equal to 1, all others are set to 0.

Then, weather conditions can also have a significant influence on the studied variables. Here, the day-ahead predicted features (temperature, wind speed, cloud cover and solar radiation) provided by numerical weather predictions (NWP) at a single location in Belgium are thus integrated as potential network inputs.

Finally, past information is also provided so that the neural network can exploit the recent dynamics of the variable.

For each of the tested input configurations, the architecture of the network is tailored by optimizing its hyperparameters, and the BLSTM model is trained (using online Back Propagation Through Time [43]).

In order to increase the network robustness regarding unseen data, two regularization techniques are used during the learning phase. First, early stopping is implemented. It consists in dividing the historical set of data into a training set and a validation set in order to stop the learning phase at the optimal time (before the network begins to be too closely adapted to the training dataset). The second technique is the addition of weight noise during training so as to ensure that the network ignores the irrelevant information (noise in the data). It should thus be noted that other hyperparameters (such as the variance of weight noise, or the learning rate of the gradient descent learning procedure) have to be optimized together with the complexity of the network architecture during the optimal model selection. Thereafter, once the optimal model is determined, the statistical quality of its forecasts computed on the validation set can be evaluated.

At the end of the two-nested loop procedure, the different models (differentiated regarding both their inputs and hyperparameters) can be ranked with respect to their statistical score on the validation set (unseen data), and the best model is then used for practical application. It should be noted that it is also common to rely upon an ensemble of models for actual predictions. Typically, the results of models that tend to either under- and overestimate the solution are averaged to give a more stable outcome.

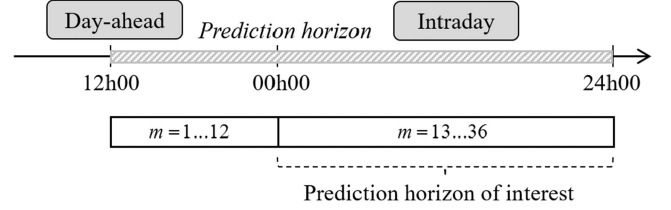


Fig. 5. Representation of the prediction horizon.

IV. CASE STUDY

In the current context of liberalized electricity markets, energy aggregators, also referred to as Virtual Power Plants (VPPs), need to define each day (typically at 12 h00) their optimal bidding strategy in the day-ahead wholesale market. Practically, they have to determine the energy exchanges (sold or purchased) in the market, which is cleared through an auction mechanism at the end of which both clearing price and volume are obtained for the 24 hours of the next day. The market clearing is carried out by the market operator that aggregates demand bids and generation offers into the so-called merit order curve. Market players must thus have accurate prediction of the stochastic variables influencing the decision process. Predictions are performed simultaneously at 12 h in day-ahead for the 24 hours of the next day. In this way, as represented in Fig. 5, the prediction horizon of interest (for the day-ahead stochastic decision-making problem) spans thus from $m = 12$ to 36 hours in the future.

In this work, the predictions focus on the aggregated load and renewable generation (wind and PV) in Belgium in order to confront such predictions with the day-ahead forecasts published by the system operator. Indeed, the latter publishes each day its forecasts for the purpose of promoting a transparent and more competitive market. Although, for confidentiality reasons, its underlying prediction tool cannot be disclosed. Within the objective to compare results on a fair basis, it should be noted that our predictions are therefore realized in the same conditions (at 12 h in day-ahead).

Regarding renewable generation, the goal is to quantify the maximum amount of energy that can be extracted at each time step of interest. The actual generation level (after curtailment or control strategies, e.g., to provide ancillary services) is outside the scope of this work.

Similarly, the work focuses on the non-shiftable load. Indeed, it is important that this part of the total consumption can be reliably predicted (for maintaining the energy balance).

These variables (load and renewable generation) are therefore predicted as forcing terms for a power system (exogenous variables that are not influenced by the system), and are then integrated as explanatory variables for predicting electricity prices. The data-driven forecasting tools are trained to automatically learn the hidden market mechanism, and do not explicitly take physical constraints into account. However, in Europe, it has little impact since the day-ahead market is cleared without accounting for grid constraints (congestions or voltage levels) within the constitutive market zones [44].

A. Benchmark

Practically, the following neural network architectures are compared (thereafter referred by their abbreviations in brackets):

- Multilayer perceptron (MLP), i.e., traditional static feed-forward network, in which outputs at every time steps are simultaneously predicted so as to avoid accumulation of errors.
- Unidirectional LSTM (LSTM).
- Bidirectional LSTM (BLSTM).

In order to compare the different variants on a fair basis, the same amount of effort was given in the determination of the optimal topology (same number of investigated configurations in the two-nested loops procedure). Moreover, all architectures are implemented and tested using the same simulation environment (Matlab).

Then, for obtaining a more representative benchmark study, other state-of-the-art forecasting approaches are analyzed. These methods encompass autoregressive integrated moving average (ARIMA) models that assume a constant variance of the series while time correlation are linearly represented, support vector machine (SVR) that performs a nonlinear mapping of the input data into a high-dimensional space where linear functions are used for regression, and random forests (RF) where different models with low bias and high variance are combined to obtain a forecaster with a lower variance that still maintain a low bias.

The prediction models were trained using hourly historical data from 2012 until 2017. The performance of the three compared neural networks (final architectures at the end of the optimal model selection) is evaluated on a month of winter 2017 (test set composed of out-of-sample data that are not included within the learning phase).

B. Performance of Point Forecasts

Firstly, the statistical quality of point forecast, which focuses on the degree of correspondence between the predictions and the actual observations, is estimated. For these deterministic forecasts, the root mean square error (RMSE) is used as error metric:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{t=1}^n (y_t - d_t)^2} \quad (16)$$

where n is the number of sample data (number of time-steps T predicted each day multiplied by the number of simulated days), y_t the output of the prediction model and d_t the actual measured value. The results are presented in Table I.

The results show that machine learning models perform better than statistical models. This can be explained by the fact that ARIMA models are linear forecasters, which makes them poorly suited for modeling the nonlinear behavior with quick variations of the electrical variables of interest. The outcomes are also consistent with previous studies [45] that suggest that SVR and RF constitute reliable alternatives to deep learning methods, although their performance are slightly inferior.

Moreover, it can be seen that recurrent neural networks show higher accuracy than all other models, including the multilayer

TABLE I
COMPARISON OF TESTED ARCHITECTURES IN TERMS OF RMSE

Network	Wind	PV	Load	DA prices
BLSTM	101 MW	53 MW	236 MW	17 €
LSTM	108 MW	72 MW	242 MW	20 €
MLP	113 MW	78 MW	282 MW	22 €
SVR	114 MW	80 MW	283 MW	22 €
RF	118 MW	81 MW	287 MW	23 €
ARIMA	122 MW	84 MW	311 MW	24 €
System Operator	109 MW	67 MW	391 MW	NA

perceptron and the tool used by the system operator. Specifically, the best performance is given by the bidirectional architecture, which emphasizes the importance of accurately accounting for intricate time dependencies in the context of multi-step ahead forecasting.

In general, it is also interesting to emphasize that the optimal results are obtained with deep architectures (with several hidden layers), and that deep learning forecasters outperforms all other models of the benchmark. In this way, the best topology for predicting the wind generation, the total load and the day-ahead prices is obtained with a 3-layers BLSTM, whereas the best prediction model for the aggregated photovoltaic generation is a BLSTM network with 4 hidden layers (Section IV.C). Furthermore, the optimal size (capacity) of the neural networks is relatively low (10 to 20 neurons within each hidden layers, which can be explained by the necessity to avoid overfitting with the small number of historical data available. In this way, over the years, the accuracy of the proposed self-learning approach is expected to grow thanks to the amount of data that will allow to progressively increase the network capacity (complexity).

C. Inputs and Hyperparameters Selection

The optimal model selection (Fig. 4) requires to train a large number of different models with different architecture in order to properly cover the design space, which is time-consuming (around 1 minute for MLP models and 5 minutes for LSTM-based architectures). However, the models can be trained in parallel and this training task needs to be carried only once in pre-processing. The resulting optimal model can then be used in real-time for predicting the uncertain variables of interest, which then takes less than 1 second.

Overall, the selected inputs (explanatory variables) of the neural networks are listed below.

For all variables, the last measured values (typically the previous 6 to 24 hours) were highly important to capture the dynamics of the time series. The accuracy of forecasts for load and electricity prices is also significantly improved with the knowledge of the values of the same day of the previous week.

Then, for all temporal information (hours of the day, day of the week and month of the year), the best performance was obtained with the mutually exclusive binary representation, provided that the network complexity was sufficiently important (with a sufficient number of hidden units with each layer). Furthermore, an additional day index is introduced to efficiently represent occasional events such as public holidays.

TABLE II
COMPARISON OF BLSTM ARCHITECTURES IN TERM OF RMSE

Predicted variable	# layers - #neurons	#epochs	RMSE
wind	1 - 30	59	108 MW
	2 - 20	67	109 MW
	3 - 15	65	101 MW
	4 - 8	59	104 MW
PV	1 - 30	125	65 MW
	2 - 20	206	61 MW
	3 - 16	196	55 MW
	4 - 13	199	53 MW

The quality of the models is also increased with the introduction of weather data (such as temperature, cloud cover, etc.) coming from meteorological models.

Finally, it should also be noted that the electricity prices are forecasted after the aggregated renewable generation and total load. Indeed, these two variables are used as additional meaningful explanatory variables.

Once the inputs are selected, a sensitivity analysis is conducted to investigate the effect of the model complexity on the forecasting performance of renewable generation. To that end, different topologies of BLSTM networks are evaluated, varying along both main dimensions: the number of hidden levels and the number of neurons within each hidden layer. For each topology, the number of neurons is around 10 000. The results are illustrated in Table II and encompass the number of epochs (i.e., number of iterations of the gradient descent algorithm through the training set before convergence) as well as the final error on the test set.

The sensitivity analysis on network depth indicates that increasing the number of hidden layers can enhance the accuracy, but only up to a limit number of layers. Indeed, by increasing the model complexity, we are facing overfitting issues due to the lack of data diversity and network parameter redundancy. In general, in order to improve the model performance, it is preferable to increase the complexity with additional hidden layers than with more neurons within the same recurrent layer.

It should also be noted that the hyperparameter solution is closely related to the size of the historical database. In this way, the models that require a large number of parameters (such a deep learning networks with a larger amount of hidden layers) necessitate large amount of data to accurately estimate the parameters. Consequently, if the size of current dataset is limited, the hyperparameter optimization is likely to select a smaller model that performs better with the available information, although it is not the best model in general.

D. Performance of Gaussian and Non-Parametric Forecasts

The second objective is to compare the Gaussian assumption of prediction errors with a non-parametric approach, and the BLSTM architecture is used as a reference to evaluate these (parametric and empirical) methods. The statistical accuracy of both methods is computed using the total quantile loss, such as defined in (11) with $q = 1, 5, 10, 25, 50, 75, 90, 95$ and

TABLE III
COMPARISON OF PARAMETRIC AND NON-PARAMETRIC QUANTILES

Topology	Wind	PV	Load	DA prices
BLSTM + Gaussian	171 MW	42 MW	422 MW	34 €
BLSTM + quantile	147 MW	41 MW	389 MW	28 €

99%, averaged over the 24 hourly time steps of the prediction horizon. This error metric is evaluated on the same test set (winter month of 2017), and the outcomes are presented in Table III. Practically, this function allows to measure if the intervals properly encapsulate the actual realization of uncertain variables, while quantifying the tightness of these intervals. The smaller is the quantile loss, the better is the performance of the forecasting method.

Overall, the non-parametric model slightly outperforms the outcomes obtained with the Gaussian error assumption (i.e., the quantiles enclose more accurately the actual observations, and are characterized by tighter intervals). These results tend thus to support that the empirical model should be privileged, but, for the studied variables, assuming a Gaussian distribution of prediction errors does not lead to significant modeling errors (especially for PV production).

For illustrating the quality of results obtained using the BLSTM network with the (non-parametric) quantile loss function, the probabilistic forecasts associated with the four studied variables are shown in Fig. 6. Specifically, the concatenation of day-ahead predictions (at 12 h in day-ahead for the 24 hours of the next day) carried out during 7 consecutive days (from Monday to Sunday) are presented.

Generally, one can see that the predicted intervals properly encompass the actual realizations of uncertainties (the volatility of the studied variables is well captured). However, we observe that the quantiles are more tightened for the aggregated load, which indicates that the amount of uncertainty associated with this variable is much lower than, for instance, wind generation. Moreover, it should be mentioned that the simulated month was characterized by a high demand (and very low renewable generation) throughout Western Europe, which has considerably increased the price uncertainty (volatility) during this period.

E. Importance of the Copula Model

In order to evaluate the dependencies among electrical and market data (to validate the need of the copula model), the dependence between variables is computed using two different metrics. First, the linear correlation between pairs of variables is measured with the Pearson coefficient. Then, the Spearman coefficient, which is able to capture more than linear dependencies, is used to assess the relationship between rankings of two variables. The results of the correlation study are summarized in Table IV.

As intuitively expected, the electricity prices are driven down when the share of renewable generation increases. Indeed, due to their very low operating costs, these technologies can bid at low prices in the wholesale market. Likewise, the prices tend to increase when the total demand rises. However, we can see

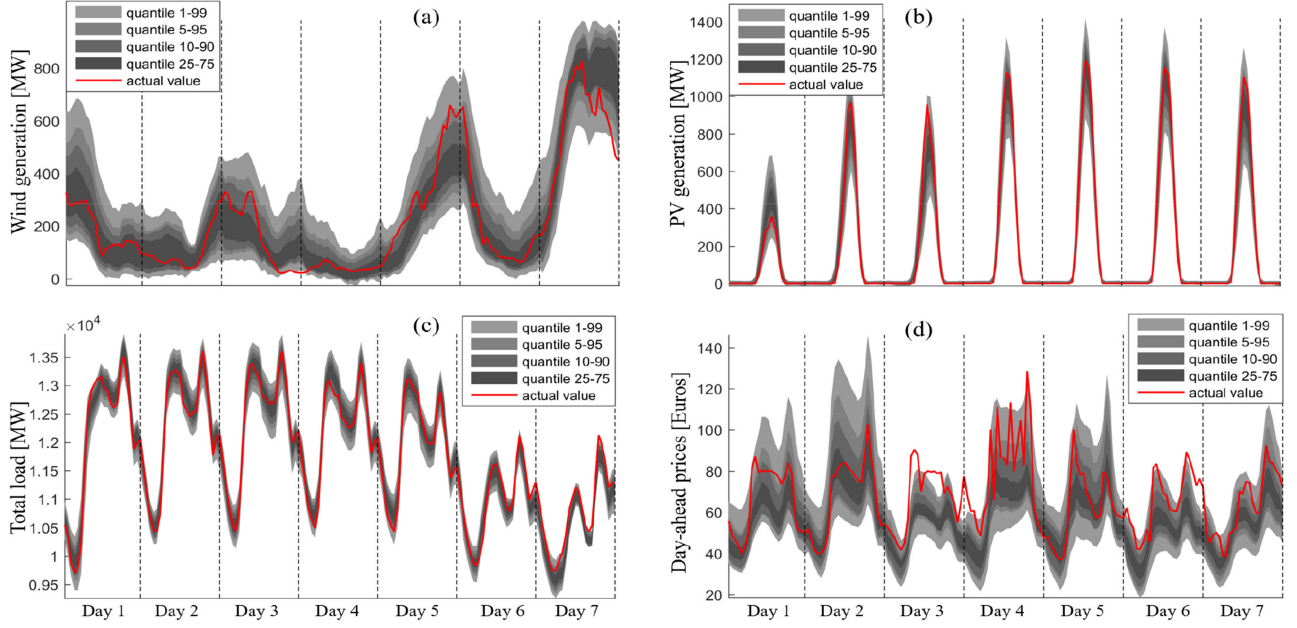


Fig. 6. Probabilistic forecasts performed during 7 consecutive days for wind. (a), PV generation (b), total load (c), and day-ahead electricity prices (d).

TABLE IV
CORRELATION BETWEEN VARIABLES

Var. 1	Var. 2	Pearson	Spearman
Wind	Price	-0.17	0.10
Load	Price	0.43	0.65
Load	Wind	0.11	0.10

that the correlation between the wind generation and the load is quite limited.

F. Quality of Scenarios

Once these probabilistic forecasts are obtained, the distributions are sampled to obtain the time trajectories that can thereafter be used in the stochastic optimization. The 10 scenarios generated with the independent [18] and copula-based sampling methods for the wind generation (for a typical day) are shown in Fig. 7 (along with the 1-99% quantiles).

It can be seen that the proposed copula-based sampling strategy allows to better capture the statistical information of the multivariate time-varying distribution of interest. Indeed, the independent sampling leads to scenarios with numerous sharp ramps that do not represent the smoother time profile of the aggregated wind power.

In order to quantify the statistical accuracy of the generated scenarios of electrical variables that are inherently correlated, the interdependence structure of forecast errors is studied. To that end, the autocorrelation function (ACF) of scenarios is compared with the one associated with the original variables. The ACF yields indeed the (linear) correlation between two values of the same variable at times different (lagged) times.

The results are summarized in Table V, where the mean ACF deviation (i.e., deviations between the ACF of the scenarios and

TABLE V
TEMPORAL PROPERTIES OF GENERATED SCENARIOS (DEVIATION OF THE AUTOCORRELATION FUNCTION ON REPRESENTATIVE LAGS)

	Copula-based sampling	Independent sampling
Wind	0.24	0.75
PV	0.17	0.92
Load	0.19	0.62
DA prices	0.09	0.63

the one of the actual data averaged on the first representative lags of the serial correlation) are presented.

The results show that the studied variables (renewable generation, load and electricity prices in the day-ahead market) do not come from a random processes (high values of autocorrelation between consecutive time steps), and that the copula-based sampling, contrary to the independent policy, appropriately captures this time-dependent information.

G. Value of Probabilistic Forecasts

Finally, we analyze the practical value of generating more accurate scenarios, by studying the benefits (e.g., economical) resulting from the use of these scenarios in the subsequent decision-making procedure. Here, the day-ahead optimization faced each day by an electricity retailer having its own renewable generation capacity is used as a case study. The portfolio is composed of one percent of the Belgian load as well as twenty percent of the installed (onshore) wind and PV capacity. Basically, the retailer aims to balance its portfolio on a quarter-hourly basis (so as to avoid financial penalties in case of imbalance) by exchanging (the surplus or deficit of energy) in the day-ahead electricity market [46].

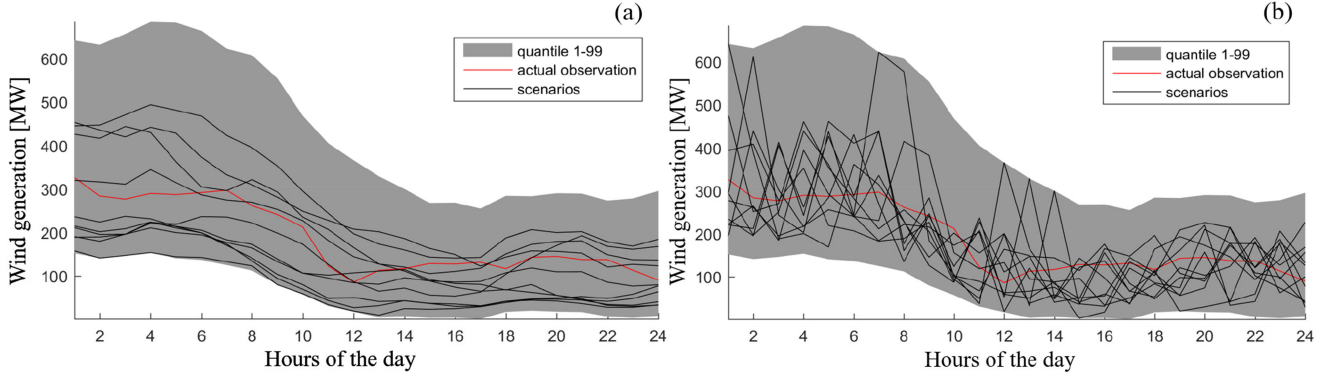


Fig. 7. Scenarios of wind generation obtained using the copula-based sampling (a), and independent sampling (b) methods.

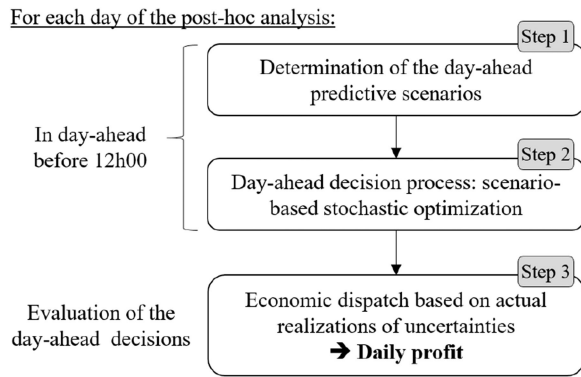


Fig. 8. Procedure used to compare the quality of day-ahead decisions based on the different techniques to characterize the forecast uncertainty.

In this context of time-dependent decisions under uncertainty, it is interesting to estimate the value of the different techniques to generate day-ahead scenarios, which is here realized through the procedure depicted in Fig. 8.

The methodology is carried out for three different variants, which differ by the way scenarios are generated (Step 1):

- 1) MLP + copula-based sampling
- 2) Probabilistic BLSTM + independent sampling
- 3) Probabilistic BLSTM + copula-based sampling

The practical quality of scenarios is then analyzed through a post-hoc analysis, which consists in confronting the day-ahead decisions (obtained at the end of the day-ahead stochastic optimization of Step 2) with respect to the actual realizations (observations) of uncertainties. To that end, an economic dispatch of the VPP (Step 3) has to be performed. The objective is to compute the profit actually generated based on the actual trajectories of uncertain variables as well as the day-ahead decisions, i.e., energy exchanged in the day-ahead market for each of the 24 hours. This procedure is performed for each day of the studied month, and the results (daily profit for the three investigated variants) are represented in Fig. 9.

By comparing scenarios #2 and #3, it can be concluded that using representative scenarios in the stochastic optimization process of step 2 (scenarios that account for the complex dependence structure among variables) is an highly important factor to take reliable decisions, which is here associated with an increase of profit of around 4×10^5 Euros (i.e., relative increase of more

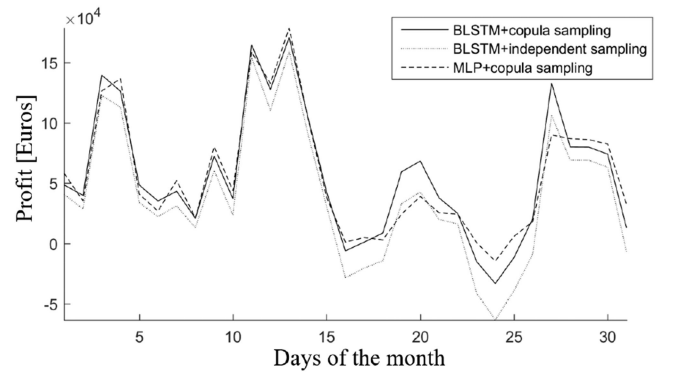


Fig. 9. Daily profit generated by the electricity aggregator with respect to the stochastic scenarios used to model uncertainties.

than 10%) over the simulated month. Moreover, the quality of predictions (in our case the fact of using the BLSTM neural networks instead of traditional feedforward networks) plays also an important role to improve decisions in an uncertain environment. In this way, better predictions enable decision makers to avoid taking overly conservative policies (so as to guarantee their robustness towards extreme scenarios). Here, improved predictions led the portfolio to rise its total profit throughout the considered month by 0.5×10^5 Euros.

V. CONCLUSION AND PERSPECTIVES

In this work, a new approach to generate short-term multivariate predictive scenarios is presented. The methodology attempts to address the main challenges associated with such a task, i.e., obtaining accurate forecasts that efficiently catch the contextual information contained in the explanatory variables by exploiting the structure of the problem, while capturing both temporal and cross-variable dependencies when generating scenarios. The results demonstrate that the proposed methodology yields accurate, calibrated forecast distributions learned from the historical dataset, and that the generated scenarios enable to increase the economic profit of energy aggregators participating in electricity markets.

An interesting perspective of this work is to combine data-driven models with structural constraints arising from the knowledge of the underlying environment (market rules and constraints). Indeed, with the development of physically-based

appliance and equipment at both generation and consumption levels, it may become interesting to predict such new power usage behaviors.

REFERENCES

- [1] P. Pinson, C. Chevallier, and G. N. Kariniotakis, "Trading wind generation from short-term probabilistic forecasts of wind power," *IEEE Trans. Power Syst.*, vol. 22, no. 3, pp. 1148–1156, Aug. 2007.
- [2] A. A. Thatte, L. Xie, D. E. Viassolo, and S. Singh, "Risk measure based robust bidding strategy for arbitrage using a wind farm and energy storage," *IEEE Trans. Smart Grid*, vol. 4, no. 4, pp. 2191–2199, Dec. 2013.
- [3] Y. Zhang, J. Wang, and X. Wang, "Review on probabilistic forecasting of wind power generation," *Renewable Sustain. Energy Rev.*, vol. 32, pp. 255–270, Apr. 2014.
- [4] P. Pinson and G. Kariniotakis, "Conditional prediction intervals of wind power generation," *IEEE Trans. Power Syst.*, vol. 25, no. 4, pp. 1845–1856, Nov. 2010.
- [5] C. Wan, Z. Xu, P. Pinson, Z. Y. Dong, and K. P. Wong, "Probabilistic forecasting of wind power generation using extreme learning machine," *IEEE Trans. Power Syst.*, vol. 29, no. 3, pp. 1033–1044, May 2014.
- [6] A. Kavousi-Fard, A. Khosravi, and S. Nahavandi, "A new fuzzy-based combined prediction interval for wind power forecasting," *IEEE Trans. Power Syst.*, vol. 31, no. 1, pp. 18–26, Jan. 2016.
- [7] H. Quan, D. Srinivasan, and A. Khosravi, "Short-term load and wind power forecasting using neural network-based prediction intervals," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 2, pp. 303–315, Feb. 2014.
- [8] C. Wan, J. Lin, Y. Song, Z. Xu, and G. Yang, "Probabilistic forecasting of photovoltaic generation: An efficient statistical approach," *IEEE Trans. Power Syst.*, vol. 32, no. 3, pp. 2471–2472, May 2017.
- [9] A. Khosravi, S. Nahavandi, and D. Creighton, "Construction of optimal prediction intervals for load forecasting problem," *IEEE Trans. Power Syst.*, vol. 25, no. 3, pp. 1496–1503, Aug. 2010.
- [10] J. H. Zhao, Z. Y. Dong, Z. Xu, and K. P. Wong, "A statistical approach for interval forecasting of the electricity price," *IEEE Trans. Power Syst.*, vol. 23, no. 2, pp. 267–276, May 2008.
- [11] N. A. Shrivastava, A. Khosravi, and B. K. Panigrahi, "Prediction interval estimation of electricity prices using PSO-tuned support vector machines," *IEEE Trans. Ind. Inform.*, vol. 11, no. 2, pp. 322–331, Apr. 2015.
- [12] K. Bruninx, "Improved modeling of unit commitment decisions under uncertainty," Ph.D. dissertation, Appl. Mech. Energy Conver., KU Leuven, Leuven, Belgium, 2016.
- [13] G. A. Morales-Espana, "Unit commitment computational performance, system representation and wind uncertainty management," Ph.D. dissertation, Elect. Power Syst., Universidad Pontificia Comillas, Madrid, Spain, 2014.
- [14] J. M. Morales, R. Minguez, and A. J. Conejo, "A methodology to generate statistically dependent wind speed scenarios," *Appl. Energy*, vol. 87, pp. 843–855, 2010.
- [15] P. Pinson, H. Madsen, H. A. Nielsen, G. Papaefthymiou, and B. Klöckl, "From probabilistic forecasts to statistical scenarios of short-term wind power production," *Wind Energy*, vol. 12, no. 1, pp. 51–62, 2008.
- [16] G. Papaefthymiou and P. Pinson, "Modeling of spatial dependence in wind power uncertainty," in *Proc. 10th Int. Conf. Probabilistic Methods Appl. Power Syst.*, Rincon (Puerto Rico), 2008, pp. 1–8.
- [17] X.-Y. Ma, Y.-Z. Sun, and H.-L. Fang, "Scenario generation of wind power based on statistical uncertainty and variability," *IEEE Trans. Sustain. Energy*, vol. 4, no. 4, pp. 894–904, Oct. 2013.
- [18] H. Quan, D. Srinivasan, and A. Khosravi, "Incorporating wind power forecast uncertainties into stochastic unit commitment using neural network-based prediction intervals," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 9, pp. 2123–2135, Sep. 2015.
- [19] Z. Wang, W. Wang, C. Liu, Z. Wang, and Y. Hou, "Probabilistic forecast for multiple wind farms based on regular vine copulas," *IEEE Trans. Power Syst.*, vol. 33, no. 1, pp. 578–589, Jan. 2018.
- [20] Y. Wang, N. Zhang, Q. Chen, D. S. Kirschen, P. Li, and Q. Xia, "Data-driven probabilistic net load forecasting with high penetration of behind-the-meter PV," *IEEE Trans. Power Syst.*, vol. 33, no. 3, pp. 3255–3264, May 2018.
- [21] H. Shi, M. Xu, and R. Li, "Deep learning for household load forecasting—A novel pooling deep RNN," *IEEE Trans. Smart Grid*, vol. 9, no. 5, pp. 5271–5280, Sep. 2018.
- [22] D. Gan, Y. Wang, N. Zhang, and W. Zhu, "Enhancing short-term probabilistic residential load forecasting with quantile long-short-term memory," *J. Eng.*, vol. 2017, no. 14, pp. 2622–2627, 2017.
- [23] Y. Wu *et al.*, "Google's neural machine translation system: Bridging the gap between human and machine translation," unpublished paper, 2016. [Online]. Available: <https://arxiv.org/abs/1609.08144>
- [24] M. Schuster, "Speech recognition for mobile devices at Google," in *PRICAI 2010: Trends in Artificial Intelligence*. Berlin, Germany: Springer, 2010, pp. 8–10.
- [25] V. Flunkert, D. Salinas, and J. Gasthaus, "DeepAR: Probabilistic forecasting with autoregressive recurrent networks," unpublished paper, 2017. [Online]. Available: <https://arxiv.org/abs/1704.04110>
- [26] R. Wen, K. Torkkola, and B. Narayanaswamy, "A multi-horizon quantile recurrent forecaster," unpublished paper, 2017. [Online]. Available: <https://arxiv.org/abs/1711.11053v1>
- [27] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015.
- [28] S. Hochreiter, Y. Bengio, P. Frasconi, and J. Schmidhuber, "Gradient flow in recurrent nets: The difficulty of learning long-term dependencies," In S. C. Kremer and J. F. Kolen, eds., in *A Field Guide to Dynamical Recurrent Neural Networks*, IEEE Press, 2001.
- [29] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [30] M. Schuster and K. K. Paliwal, "Bidirectional Recurrent Neural Networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.
- [31] A. Graves, N. Jaitly, and A. Mohamed, "Hybrid speech recognition with deep bidirectional LSTM," in *Proc. IEEE Workshop Autom Speech Recognit. Understanding*, Olomouc, Czech Republic, 2013, pp. 273–278.
- [32] S. Bengio, V. Oriol, N. Jaitly, and N. Shazeer, "Scheduled sampling for sequence prediction with recurrent neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 1171–1179.
- [33] A. M. Lamb, A. Goyal, Y. Zhang, S. Zhang, A. C. Courville, and Y. Bengio, "Professor forcing: A new algorithm for training recurrent networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 4601–4609.
- [34] R. Koenker and B. Jr Gilbert, "Regression quantiles," *Econometrica, J. Econometric Soc.*, vol. 46, pp. 33–50, 1978.
- [35] I. Takeuchi, Q. V. Le, T. Sears, and A. J. Smola, "Nonparametric quantile estimation," *J. Mach. Learn. Res.*, vol. 7, pp. 1231–1264, 2006.
- [36] A. M. Law and W. D. Kelton, *Simulation Modeling And Analysis*, 3rd ed. New York, NY, USA: McGraw-Hill, 2000.
- [37] M. Sun, I. Konstantelos, S. Tindemans, and G. Strbac, "Evaluating composite approaches to modelling high-dimensional stochastic variables in power systems," in *Proc. 2016 Power Syst. Comput. Conf.*, Genoa, 2016, pp. 1–8.
- [38] W. Wu, K. Wang, B. Han, G. Li, X. Jiang, and M. L. Crow, "A versatile probability model of photovoltaic generation using pair copula construction," *IEEE Trans. Sustain. Energy*, vol. 6, no. 4, pp. 1337–1345, Oct. 2015.
- [39] A. Sklar, "Fonctions de repartition à n dimensions et leurs marges," *Publ. Inst. Statist. Paris*, vol. 8, pp. 229–231, 1959.
- [40] J. C. Strelén and F. Nassaj, "Analysis and generation of random vectors with copulas," in *Proc. 2007 Winter Simul. Conf.*, 2007, pp. 488–496.
- [41] J. C. Strelén, "Tools for dependent simulation input with copulas," Proceedings of the 2nd International Conference on Simulation Tools and Techniques for Communications, Networks and Systems, SimuTools 2009, Rome, Italy, pp. 30–36, Mar. 2–6, 2009.
- [42] J.-F. Toubeau, J. Bottieau, F. Vallée, and Z. De Grève, "Improved day-ahead predictions of load and renewable generation by optimally exploiting multi-scale dependencies," in *Proc. 7th IEEE Conf. Innovative Smart Grid Technol.*, Dec. 2017, p. 5.
- [43] R. J. Williams and D. Zipser, "Gradient-based learning algorithms for recurrent networks and their computational complexity," in *Back-Propagation: Theory, Architectures and Applications*, Y. Chauvin and D. E. Rumelhart eds., Hillsdale, NJ: Erlbaum, 1995, pp. 433–486.
- [44] K. Van den Bergh, D. Couckuyt, E. Delarue, and W. D'haeseleer, "Redispatching in an interconnected electricity system with high renewables penetration," *Elect. Power Syst. Res.*, vol. 127, no. 10, pp. 64–72, 2015.
- [45] J. Lago, F. De Ridder, and B. De Schutter, "Forecasting spot electricity prices: Deep learning approaches and empirical comparison of traditional algorithms," *Appl. Energy*, vol. 221, pp. 386–405, 2018.
- [46] J. F. Toubeau, Z. De Grève, and F. Vallée, "Medium-term multi-market optimization for virtual power plants: A stochastic-based decision environment," *IEEE Trans. Power Syst.*, vol. 33, no. 2, pp. 1399–1410, Mar. 2018.

Jean-François Toubau (S'13) was born in Mons, Belgium, in 1990. He received the degree in civil electrical engineering and the Ph.D. degree in electrical engineering from the University of Mons, Mons, Belgium, in 2013 and 2018, respectively. He is currently a Research Assistant with Electrical Power Engineering Unit of the same faculty. His research interests include decision-making in the context of power markets as well as data analytics and processing.

Jérémie Bottieau (S'17) received the diploma in electrical engineering from the University of Mons, Mons, Belgium. Since 2017, he is working toward the Ph.D. degree with the Department of Electrical Power Engineering, University of Mons. His research interests include short-term forecasting and optimization in electricity markets.

François Vallée (M'09) received the degree in civil electrical engineering and the Ph.D. degree in electrical engineering from the Faculty of Engineering, University of Mons, Mons, Belgium, in 2003 and 2009, respectively. He is currently an Associate Professor with the Electrical Power Engineering Unit of the same faculty. He has authored or coauthored of several publications in that field and his Ph.D. work has been awarded by the SRBE/KBVE Robert Sinave Award in 2010. His research interests include PV and wind generation modeling for electrical system reliability studies in presence of dispersed generation. He is currently a member of the governing board from the 'Société Royale Belge des Electriciens - SRBE/KBVE' (2017).

Zacharie De Grève (M'12) received the electrical and electronics engineering degree from the Faculty of Engineering, University of Mons, Mons, Belgium, in 2007, and the Ph.D. degree in electrical engineering from the University of Mons. He has been a Research Fellow of the Belgian Fund for Research (F.R.S/FNRS) until 2012. He is now a Research and Teaching Assistant with the Electrical Power Department of the same university. His research interests include the numerical modeling of electromagnetic fields, as well as the integration of renewable energies in power electrical networks.