Recalibration of Recurrent Neural Networks for Short-Term Wind Power Forecasting

Jean-François Toubeau, Pierre-David Dapoz, Jérémie Bottieau, Aurélien Wautier, Zacharie De Grève, François Vallée
Power Systems & Markets Research Group, University of Mons, Belgium

***Abstract*—This paper is focused on the day-ahead prediction of the onshore wind generation. This information is indeed published each day, ahead of the market clearing, by European Transmission System Operators (TSOs) to help market actors in their scheduling strategy. In that regard, our first objective is to improve the forecast performance by efficiently capturing the complex temporal dynamics of the wind power using recurrent neural networks. Practically, advanced architectures of Long Short Term Memory (LSTM) networks are implemented and compared. Secondly, in order to continuously refine the prediction tool, different techniques for recalibrating the model during its practical utilization are analyzed. This procedure consists in adjusting the parameters of the neural networks by taking advantage of the new information revealed over time, without the (time-consuming) need to retrain the model from scratch using the whole available dataset. Finally, the financial savings from the improvement of the forecast accuracy are estimated. Outcomes from the Belgian case study show that an optimal model recalibration can significantly improve forecast reliability, thereby decreasing the balancing costs of the system.**

***Index Terms*—Bidirectional LSTM, Deep Learning, Electricity Markets, Recalibration Forecast, Wind Power Prediction.**

I. Introduction

The liberalization of the electricity sector has introduced new prerogatives for Transmission System Operators (TSOs), among which the task of facilitating the access to the market for all actors. In that regard, TSOs must provide various information to market participants such as the anticipated wind generation. With the increased contribution of such weather-dependent (and thus, uncertain and intermittent) renewable generation, this forecasting task has recently become essential for ensuring a reliable and cost-effective system operation.

Researchers have thus studied a variety of techniques for wind prediction. Firstly, statistical approaches based on the inference (from observed data) of basic statistics such as the mean, variance and autocorrelation have emerged [1], [2]. However, the underlying assumptions often involve that such forecasters rely on simple linear models which are not able to capture the nonlinear characteristics (such as the different ramp rates) of the wind. In parallel, physical models were also developed, but they necessitate a complex mathematical description of the environment, which is computationally intensive, and often based on arbitrary simplifying assumptions [3]. Such models are thus often employed for longer term forecasts. To address these issues, machine learning approaches have recently been tested by the prediction community, and have progressively exhibited better performances than classical methods [4]-[7]. This trend is mainly driven by the ability of such techniques to accurately capture and represent hidden characteristics of complex variables, without the need to arbitrarily define the model complexity. It should however be noted that outputs of physics-based forecasts can be treated as inputs of purely data-driven approaches in order to enrich their input feature space with physical considerations. In addition, the flexible nature of data-driven tools, mainly neural networks, allows to adapt their architecture to the characteristics of the forecasting problem, thereby improving their accuracy. This property has led to the advent of recurrent neural networks (RNNs), deep learning structures that are able to build an internal representation of past events, thus propagating relevant information through time. Their success has been fostered by the Long Short Term Memory (LSTM) architecture, which has shown a high potential in processing time series such as wind power [8]-[9]. However, different LSTM-based networks can be developed, depending on how the data are fed into the model. Our objective is thus to implement the most relevant networks, and to compare their accuracy on a fair benchmark.

In parallel, one of the main challenges that still needs to be properly studied relates to the recalibration of the models. Indeed, once the forecaster is trained (using historical observations), it is then used for actual field operation (on new data). But, at that stage, the literature is very sparse on how the model should be updated with the new information that is revealed at each time step. In [10], the models are re-trained from scratch (using all the historical database) on a daily basis, but at the expense of a continuous utilization of large computational resources. In this work, we aim at improving this naïve approach by retraining the existing forecaster at optimal time intervals (e.g. every day, week, season, etc.) with a sliding window that includes the relevant set of past observations. This interest is strongly driven by long-term weather forecasting tools, which have demonstrated the interest of such recalibration strategies by periodically retraining their models using only the most recent years of data [11].

Practically, we want to quantify to which extent it may be beneficial to locally increase the variance of the model (by dynamically over-fitting to recent conditions) rather than to rely on a single static model that performs well in average along the year but that is suboptimal for each of its constituting subperiods. The underlying objective is to regularly adapt/rescale the model to any changes in long-term trends, or to the time-varying predictability (since some time periods may be intrinsically less variable than others). It should be noted that an alternative approach to alleviate such issues consists in combining forecasts from multiple models simultaneously (e.g. through ensemble learning) [12]-[15]. Overall, the three main contributions of the work can be summarized as follows.

Firstly, we exploit the flexible nature of neural networks by implementing three different recurrent architectures, based on Long Short Term Memory (LSTM) cells [16]. The objective is to predict (at 11:00 a.m. in day-ahead) the expected wind generation for the 24 hours of the next day. The three models, i.e. (i) the encoder, (ii) the decoder, and (iii) the bidirectional decoder differ in the way they capture space-time dependencies, which affects their predictive capabilities. In that regard, their accuracy is not only compared to state-of-the-art techniques (such as *gradient boosting* where new models are created to correct the errors of prior models and then added together to make the desired prediction), but also with the predictions performed and published by the TSO.

Secondly, the development of a recalibration procedure is proposed. This process allows to adjust the parameters of the neural networks by taking advantage of the new information continuously revealed over time (during the actual daily utilization of the forecaster), without the time-consuming need to retrain the model over all the historical data set.

Thirdly, the financial impact of prediction errors (on both the TSO and wind producers) is estimated. This allows to evaluate the financial gain of improving forecasting models, in particular by relying on efficient recalibration strategies, due to the saving of balancing costs (which are needed to compensate the wind imbalances).

The paper is organized as follows. In Section II, we develop different LSTM architectures to capture the dynamical behavior of wind generation, and we discuss several strategies for recalibrating the model over time. Section III focuses on the prediction accuracy of the models, which are compared with outcomes from TSO and state-of-the-art methods. The best model is then optimally recalibrated over time, which allows improving the prediction quality. Section IV finally evaluates the costs incurred by prediction errors, using actual market data. Finally, in Section V, conclusions are exposed.

II. Methodology

This section is divided into two parts. Firstly, different LSTM-based architectures of recurrent neural networks are presented (Section II-A). Secondly, the methodology to identify the best recalibration policy is discussed (Section II-B).

A. Development of LSTM-based forecasting tools

This work focuses on neural networks, which are flexible tools (theoretically able to learn any complex nonlinear functions) that combine multiple advantages. In that respect, the complexity of the model can be tailored to the complexity of the task (thereby avoiding both under- and over-fitting issues), and the architecture can be adapted to the specificities of the problem [17]. Given that wind generation is an inherently dynamic process, we consider recurrent networks (Fig. 1), which are purposely designed to process temporal dependencies.

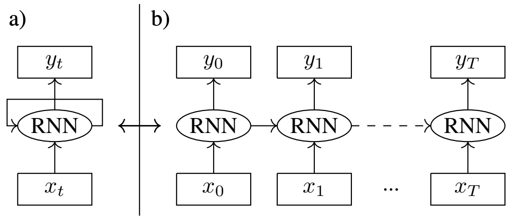

Fig. 1. General representation of recurrent neural networks (RNN) with cyclical connections that act as a dynamical memory (a), i.e. the network is unrolled though time to seamlessly represent time dependencies (b).

The general principle of recurrent neural networks (RNN) is to generate the prediction y_t based on the input information x_t, for each time step $t \in T$ of the prediction horizon of interest. Based on historical data, the RNN is trained to minimize the error between its output y_t and the actual observation d_t.

The RNN is made up of different stacked layers, each one composed of multiple neurons, which overall define the model complexity. The recurrent architecture, which is llustrated in Fig. 1, is also characterized by cyclical links, connecting the state of the neurons among consecutive time steps t, thereby propagating information through time.

In recent years, RNN applications have been very successful for a variety of problems such as speech recognition or language modeling and translation [18]. However, RNNs are known to struggle in capturing long-term dependencies, such that relevant information arising from longer term periodicities (such as seasonal effects) can be lost. To address this issue, LSTM neurons were developed, and rely on gating units that regulate the flow of information that is propagated through time. The principle of LSTM cells is depicted in Fig. 2.

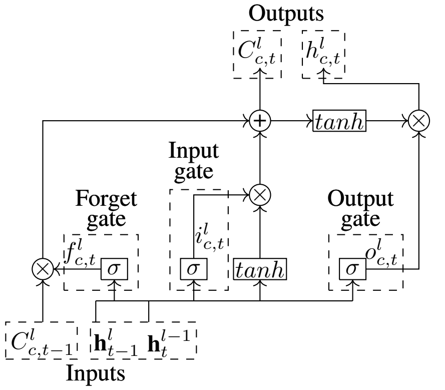

Fig. 2. Single-cell LSTM memory block c (pertaining to layer l at time t).

In Fig. 2, we observe that the LSTM cell c at layer l at time step t is fed by three different contributions, i.e. $\mathbf{h}^{l-1}_t$ the output vector (of all LSTM cells) of the layer below at the same time, $\mathbf{h}^l_{t-1}$ the output vector (of all LSTM cells) of the same layer at the previous time step, and $C^l_{c,t-1}$ the state of the cell c at the previous time step (which acts as a dynamical memory). Overall, the LSTM neuron is composed of 3 gated units (input, output and forget gates) and the LSTM layer l is thus characterized by the following composite function:

$$\mathbf{f}_t^l = \sigma\left(\mathbf{W}_f\ \mathbf{h}_t^{l-1} + \mathbf{W}_f\ \mathbf{h}_{t-1}^l + \mathbf{b}_f\right) \tag{1}$$

$$\mathbf{i}_t^l = \sigma\left(\mathbf{W}_i\ \mathbf{h}_t^{l-1} + \mathbf{W}_i\ \mathbf{h}_{t-1}^l + \mathbf{b}_i\right) \tag{2}$$

$$\mathbf{C}_t^l = \mathbf{f}_t^l\ \mathbf{C}_{t-1}^l + \mathbf{i}_t\ \tanh\left(\mathbf{W}_c\ \mathbf{h}_t^{l-1} + \mathbf{W}_c\ \mathbf{h}_{t-1}^l + \mathbf{b}_c\right) \tag{3}$$

$$\mathbf{o}_t^l = \sigma\left(\mathbf{W}_o\ \mathbf{h}_t^{l-1} + \mathbf{W}_o\ \mathbf{h}_{t-1}^l + \mathbf{b}_o\right) \tag{4}$$

$$\mathbf{h}_t^l = \mathbf{o}_t\ \tanh(\mathbf{C}_t^l) \tag{5}$$

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, forget and output gates respectively, whereas $\mathbf{C}_t$ stands for the cell activation vector. The weight matrices $\mathbf{W}_\bullet$ (i.e. links between LSTM neurons) and the bias vectors $\mathbf{b}_\bullet$ are the parameters of the network that need to be optimized during the learning procedure.

In this work, three different LSTM-based architectures, which differ by the way they process temporal information, are developed and compared, i.e. (i) the encoder, (ii) the decoder, and (iii) the bidirectional decoder.

The encoder, which is shown in Fig. 3, is a topology that sequentially process the past information $x_{-k:0}$, and that generate the predictions $y_{0:T}$ at the end of the $k+1$ steps of the sequence. The issue consists thus in feeding the tool with the available (known or estimated) information about the future $x_{1:T}$. Such information typically comes from numerical weather forecasts, which provide estimation on future temperatures, cloud covers or wind characteristics. It is thus essential to include these features as input data for the prediction model. In the encoder, it is done by providing those data at the last time step of the input sequence, which may not be optimal.

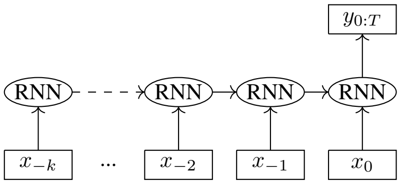

Fig. 3. General representation of the encoder architecture, where the available future information $x_{1:T}$ is fully provided in x_0.

Another option for incorporating the temporal information is to rely on a decoder, which generates a prediction at each time step of the horizon. This design, which is represented in Fig. 4, is traditionally used for on-line tasks (such as sequence generation), and is thus not well suited to take advantage of past information. Indeed, these data need to be incorporated at the first time step of the decoder (i.e. into x_0), which may thus struggle to properly extract the relevant information from both short- and long-range past features.

To improve on the decoder architecture, a third topology, i.e. the bidirectional decoder, is investigated. This design aims at optimally exploiting (at each time step) the complete contextual information. For the prediction at time t, the network is not only fed by the past information (by exploiting the traditional recurrent connections) but also by the available future data (such as the estimation of weather variables at next time steps). The underlying idea is that the available information at time $t+j$ with $j>0$ (e.g. through weather forecasts) can help explaining what will happen at time t. As we can see in Fig. 5, the bidirectional decoder is composed of two separate hidden layers, both of which connected to the same output layer (providing the predictions of interest). The resulting topology treats (simultaneously) the input sequence forwards and backwards, thereby leveraging all surrounding context in the input sequence.

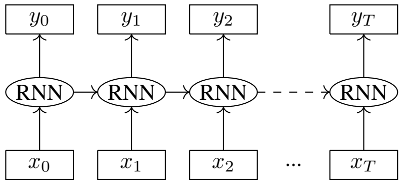

Fig. 4. General representation of the decoder architecture, where the past information $x_{-k:0}$ is fully provided in x_0.

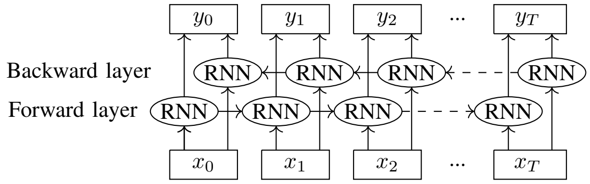

Fig. 5. General representation of the bidirectional decoder.

B. Recalibration strategy

In general, prediction errors arises from (i) incomplete or noisy explanatory variables (e.g. due to the chaotic nature of weather conditions), and (ii) model inaccuracies (commonly referred to as functional form misspecifications). Here, we ensure that all models rely on the same information (using all available inputs), and we try to determine the best parameters for each individual model (to minimize its misspecifications).

However, when the same prediction model is used each day (with the same fixed parameters each time), two problems inevitably arise. Firstly, the model does not take advantage of the new information that continuously becomes available over time (and that can be used to improve the accuracy of the data-driven model). Secondly, the model may be good in average, but not optimal for each sub-period of the year. To address both these issues, a recalibration of the model is investigated, where the model can be slightly over-fitted to most recent data (e.g. the inner dynamics of the model will differ between winter and summer months).

As represented in Fig. 6, when identifying the best recalibration strategy, two questions need to be answered :

- what is the frequency at which the model needs to be recalibrated, i.e. the optimal number of days p between two recalibrations ?
- what is the size of the sliding window, i.e. the number of days r whose information is exploited to adjust the parameters of the forecaster ?

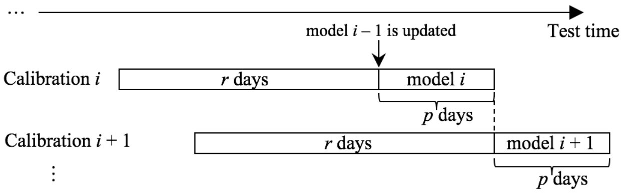

Fig. 6. Recalibration strategy: the model $i-1$ is updated with the information from the r previous days to obtain model i, which is then used for the actual day-ahead predictions for each of the next p days. Then, the model i is updated using data from the r past days (to obtain model $i+1$), and the same procedure is carried out over time.

To determine the best values of r and p, a design of experiments is carried out, and the outcomes are fully discussed in Section III-C. In particular, we show that too frequently rescaling the model is irrelevant and counter-productive. In that regard, for identifying the extent to which the model needs to be modified, three strategies are investigated. Firstly, an ideal (non-realistic) benchmark is considered, which yields the best outcome that can be expected from the recalibration. To that end, the model is trained on the r past days, but the p days to predict are used as validation set. In reality, these days cannot be used as validation (since they are not yet realized). By doing so, we ensure that the model is recalibrated in such way that it will provide the best outcomes for the days to predict. A second method selects the validation set in a classical way (using 10% of the historical information), so that the model is trained on the remaining 90% data, until convergence is achieved on the validation set. The third model is trained with a fixed number of epochs (i.e. we impose the number of iterations of the gradient descent algorithm through the training sequence of r days), so that no data are discarded for the validation set.

III. Case study

In this work, we focus on the deterministic prediction of the Belgian onshore wind generation. Our results can thus be compared with those of the system operator (i.e. Elia), which publishes each day (at 11.00 a.m., 1 hour before the closure of the day-ahead market) its hourly forecasts in order to promote a more competitive and transparent market. Indeed, a better prediction will result in better information for market players, hence increasing the reliability of their bidding policy. To compare models on a fair basis, our predictions are also carried out at 11.00 a.m. for the 24 hours of the following day. Thus, the prediction horizon of interest ranges from m = 13 to 37 hours into the future. The prediction tool used by the TSO is not disclosed for confidentiality reasons.

A. Data pre-processing

The available dataset includes the onshore wind power (aggregated at the Belgian level) for four years, starting from 2014 to the end of 2017. These four years are separated into a training, a validation and a test set. The training set starts on January 1, 2014, and ends on September 30, 2016, the validation set is composed of the next three months, and the year 2017 is used as test set.

The prediction tools are fed by input (explanatory) variables of different types. Firstly, we use weather data (such as temperature, cloud cover, etc.) that are expected for each hours of the next day. This information typically comes from advanced meteorological models. For this work, we had only access to the data from a single station (located at the center of the country). It is worth noting that the performance of the models could be increased by leveraging space-time information [19]-[20] Secondly, the last measured values (typically the previous 6 to 48 hours) of wind generation are highly important to capture the dynamics of the variable, and are thus provided to the models. In particular, different time intervals are compared (during the inputs and hyperparameters selection at the end of which the best model is selected). Thirdly, temporal information (hours of the day, day of the week and month of the year) is also used to better capture multi-scale time characteristics [21]. Finally, the installed capacity of wind generation is also used as input (to capture the increase in the wind power capacity). As a reminder, all models used in the paper are trained using the same available information, and the differences between their individual performance is thereby only driven by their intrinsic ability to capture the complexity of the forecasting task.

Before training the model, it is necessary to standardize the data for two main reasons. First, different variables are typically associated with different ranges, e.g. the scale of temperature values (in °C) is naturally lower than the historical wind generation (in MW) by several orders of magnitude. However, it does not mean that the latter variable is that much more important than the first one. Such differences will lead to more difficulty in correctly adjusting the weights of the neural network, resulting in poor outcomes and longer simulation times. Secondly, the range of variables must be adapted to the activation function of the LSTM. For instance, the hyperbolic tangent in (3) and (5) reaches saturation when the input is higher than 2. Feeding the network with higher values thereby wipes off the processing power of the network. The scaled variables $X_{scaled} \in [0, 1]$ are computed as:

$$X_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{6}$$

where X_{min} and X_{max} are the minimum and maximum values of the database for each variable X.

B. Comparison with state-of-art approaches

In this part, we calculate the prediction accuracy (over the test year 2017) for the three developed LSTM-based architectures, the encoder (Enc.), decoder (Dec.) and bidirectional decoder (B.Dec.). The models are trained using the "Adam" optimization algorithm [22]. These models are compared to the predictions published by the Belgian TSO, as well as to other classical methods, i.e.:

- Multi-Layer Perceptron (MLP) [23], the basic architecture of feedforward neural networks, containing neurons with rectifier linear units (ReLUs) as activation function.

- eXtreme Gradient Boosting (XGBoost), a (multi-stage) ensemble method in which new models are sequentially created to forecast the residuals of the global model obtained at the previous stage. At each stage, models are trained (updated) together (using a gradient descent algorithm) to make the final prediction [24].

In practice, Python 3.6.0 and the Keras library (with the TensorFlow backend) have been used for implementing neural networks, whereas the scikit-learn library has been employed for XGBoost. The complexity of each technique is optimized within an (hyperparameters optimization) procedure that compares the performance of a large number of different architectural variations of the model. This procedure is time-consuming since it takes around 1 minute to train MLP models and 5 minutes for LSTM-based networks. The resulting optimal models can then be used for predicting the wind power, which takes less than 1 second.

The results are represented in Table I. The root mean square error (RMSE) is used as error metric :

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - d_t)^2} \tag{7}$$

with $n = 8760$ the number of predicted values (i.e. hourly data over the 2017 test set), y_t the output of the prediction model and d_t the actual measured value.

TABLE I
COMPARISON OF LSTM-BASED MODELS WITH OTHER METHODOLOGIES

Methodology	MLP	XGBoost	Enc.	Dec.	B.Dec.	TSO
RMSE (MW)	128	140	127	125	115	111

Interestingly, the bidirectional decoder (B.Dec) outperforms other LSTM-based tools, which can be explained by its tailored architecture that empowers traditional RNN by better capturing temporal dependencies. Overall, all recurrent models are more accurate than classical methods (MLP and XGBoost). The optimal complexity of the bidirectional network is given by a single hidden layer with 32 LSTM neurons in its two constitutive forward and backward layers (Fig. 5). Moreover, the best results were obtained by feeding the models with 2 days of historical wind generation.

Overall, those results are very promising since they are closely challenging the performances of the TSO, which has potentially access to more input features (such as several meteorological stations in Belgium). Indeed, our best model (i.e. bidirectional decoder B.Dec.) has an error of 115 MW while the TSO has an error of 111 MW (over the year 2017). In the next Section III-C, we will investigate (for the B.Dec.) whether adjusting the model at regular intervals throughout the test year can improve the prediction accuracy.

C. Performance of the recalibration

Firstly, we define the ideal benchmark for the (B.Dec) model calibration. The results are shown in Table II, where the calibration is performed in different conditions, i.e. for a calibration performed every p days, using the information from a number r of past days.

TABLE II
PERFORMANCE OF DIFFERENT RECALIBRATION STRATEGIES FOR THE IDEAL BENCHMARK.

RMSE (MW)		r			
		1 day	7 days	30 days	90 days
p	1 day	102.76	102.06	101.64	104.22
	7 days	X	101.96	100.3	102.91
	30 days	X	X	101.95	102.94
	90 days	X	X	X	105.5

From Table II, we see that recalibrating the initial model (RMSE of 115 MW) in an optimal fashion can significantly improve its accuracy (to reach a RMSE of 100 MW, i.e. improvement of 13%), thereby surpassing the performance of the TSO model. Outcomes show that the ideal frequency for recalibrating the bidirectional decoder is p = 7 days, with an historical database composed of the past r = 30 days. These parameters will thus be used in the rest of the paper (for other recalibration methods). The value of these parameters can be explained by the nature of the learning procedure. Indeed, training the model on a lower number of days (or, on a more extreme fashion, after every hour) results in over-fitting the recalibrated model to these new observations (thereby loosing the generalization capabilities of the prediction tool). On the other hand, when the model is too rarely updated, we do not take advantage of the beneficial effect of slightly adapting the model parameters to the current conditions.

As a reminder, the stopping criterion of the ideal benchmark is triggered by the performance on the days to predict, allowing the model to perfectly over-fit on these days. It thereby yields an upper bound of the gain that can be expected by the recalibration. In actual field operation, these outcomes cannot be achieved. Different practical methods are thus studied to try reaching comparable performances.

In that regard, the most straightforward strategy consists in relying on a conventional validation set (in a similar fashion as the one used to train the original model). This allows to dynamically regularize the model, by avoiding that the parameters are overly adapted to the training data. Unfortunately, this validation set decreases the amount of data that can be used during the training phase. Here, we choose a validation test containing 10% of the dataset. As a preliminary study, we assess the impact of the position of the validation set within the historical database. Practically, four cases are studied : (i) the validation set is chosen at the beginning of the dataset (older data), (ii) in the middle, (iii) at the end (more recent data), and (iv) randomly within the whole training sequence. However, this sensitivity analysis shows that modifying the position of the validation set does not influence the accuracy of the prediction (with a difference of at most 0.1 MW).

Another approach for calibrating the model is to bypass the use of a validation set (that decreases the number of data for updating the model) by considering a fixed number of epochs (i.e. number of iterations of the gradient descent algorithm). Finding the optimal number of epochs is a challenging task since smaller values do not allow to fully exploit the new revealed information, while large values result in over-fitting issues. In both cases, we do not learn optimally. Fig. 7 shows the prediction error for different number of epochs. A number between 100 and 500 epochs is relatively stable and introduce a RMSE close to 102 MW (i.e. improvement of 11%), which is very close to the ideal benchmark. Evidently, it should be kept in mind that retraining the model on a higher number of epochs will inevitably increase the simulation time.

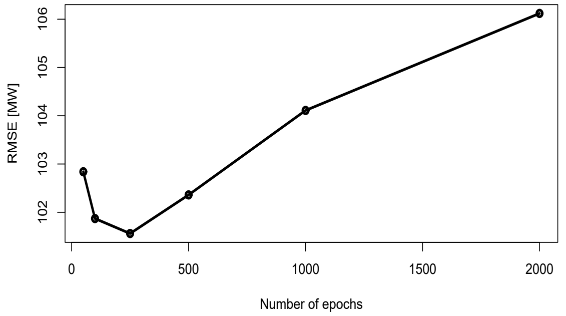

Fig. 7. Evolution of the error regarding the number of epochs.

Finally, these methods are compared with a more simple (but time-consuming) methodology where the model is retrained from scratch every $p = 7$ days.

Monthly errors from all recalibrated models are summarized in Fig. 8, where we observe that the ideal (non realistic) way for recalibrating the model systematically improves the results (for all months of the year). Then, we see that using a fixed number of epochs (i.e. 250 in accordance with Fig. 7) seems to be the best strategy (outperforming all other approaches), and leads to results close to the ideal benchmark. In this way, retraining from scratch is less efficient than our proposed recalibration method (that slightly over-fit to recent conditions). Interestingly, after recalibration, our optimal model (Epoch fixed) shows higher accuracy than the model of the TSO.

In general, we can also note than the prediction error (quantified through the RMSE) is slightly lower during summer months. However, the winter period is the more critical in terms of generation adequacy, and it is thus important to have reliable information during that time. In that regard, it is interesting to notice that our models are significantly better than the tool of the TSO for these important months.

IV. Financial costs arising from forecast errors

In this part, we evaluate the costs that can be saved by recalibrating the wind generation forecaster. Indeed, in case of real-time imbalance, the TSO restores the system frequency by relying on (costly) operating reserves. Both downward and upward reserves are needed to respectively compensate excesses and shortages of wind power [25].

The costs associated with this balancing mechanism result from two contributions, (i) the capacity allowance (€/MW/h) that remunerates the procurement of power margins (that can be activated by the TSO in case of need), and (ii) the actual deployment of the requested energy (€/MWh). However, these costs are supported by different actors. The TSO is responsible to size and build the reserve capacity, and the resulting costs (i) are transferred to the electricity bill of end-users [26]. The reserve activation costs (ii), on the other hand, are supported by market actors who are responsible for creating the imbalance [27]. In this way, by enhancing the forecast reliability, we decrease the (costly) reserve capacity to be contracted by the TSO, while decreasing the penalties incurred to wind producers, thus boosting their profitability.

In this work, we assume that the real-time system imbalance originates only from the wind forecast error (i.e. the dispatch of other resources strictly follows their committed day-ahead schedule, and the failures of network components are neglected). In accordance with the current European legislation, i.e. the System Operation Guidelines, we consider that the TSO defines the minimum reserve capacity (required to maintain the balance in the control zone) with the goal of covering the imbalances for at least 99% of the time, taking into account historic imbalance observations [26]. Hence, based on wind forecast errors computed at each of the 8760 hourly time step of the year 2017, we infer the resulting need of upward R^+ and downward R^- reserve capacity (as depicted in Fig. 9). Once the sizing is determined, we consider the average annual price (from the Belgian market) of 10 €/MWh, such that the annual costs C_r (in €) can be simply computed according to $(R^+ + |R^-|)$*10*8760.

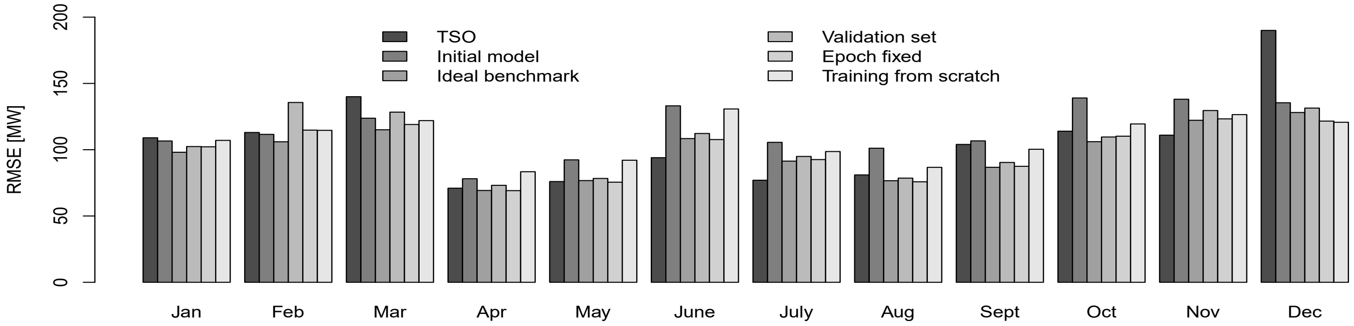

Fig. 8. Comparison of monthly evolution of each recalibration model.

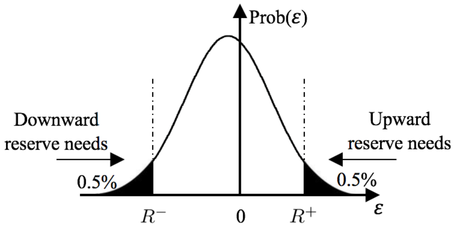

Fig. 9. Representation of the method for sizing the reserve capacity.

In Fig. 9, the prediction error ε_t is defined as the difference between the prediction y_t and the actual value d_t:

$$\varepsilon_t = y_t - d_t \tag{8}$$

A positive error corresponds thus to overestimating the wind production (such that upward reserves R^+ are needed), while a negative error underestimates the generation (resulting in the activation of downward reserves R^-). Table III provides the results of the different forecasting models, i.e. the TSO model (TSO), the static bidirectional decoder (Static), and its recalibrated version with a validation set (Val.), from scratch (Scratch), and with a fixed number of epochs (Epoch). Specifically, we represent the need of upward R^+ and downward R^- reserve capacity, and their associated costs C_r^+ and C_r^-. The total system costs are thus $C_r = C_r^+ + C_r^-$.

TABLE III
ANNUAL BALANCING COSTS ASSOCIATED WITH EACH METHODOLOGY.

	R^+[MW]	R^-[MW]	C_r^+[M€]	C_r^-[M€]	C_r[M€]
TSO	153.35	-374.57	13.43	32.81	46.24
Static	265.49	-318.72	23.26	27.92	51.18
Val.	326.16	-262.84	28.57	23.02	51.83
Scratch	283.81	-299.18	24.86	26.21	51.07
Epoch	291.88	-255.11	25.57	22.35	47.92

We see that prediction errors can strongly differ between tools. For instance, the TSO tends to underestimate the wind generation, leading to high costs C_r^- for downward capacity. For most of our models, the prediction errors tend to be symmetrical (around zero), which is logical since positive and negative errors are equally penalized in the learning procedure. However, we also observe that our LSTM-based model (and its subsequent recalibrations) lead to heavy-tailed distributions of prediction errors in which extreme inaccuracies are more frequently encountered. In that regard, even though our models are more effective in general, they necessitate to rely on higher balancing needs to cover 99% of the imbalances. However, we observe that recalibrating the static model decrease these costs by 3.26 M€ (see last colum of Table III), i.e. a reduction of 6.3%, which stresses again the added value of this re-training phase. From these observations, an interesting perspective is to modify the model training to further penalize large errors.

Then, the financial penalties incurred to wind producers are computed. In general, these balancing costs increase with the severity of the imbalance position, and vary with respect to the direction of the error. In particular, the costs curves (Fig. 10) are constructed (and made publicly available) in day-ahead by the TSO based on the market offers of the service providers. There is thus no correlation between the imbalance prices and the real-time conditions (arising, e.g., from forecast errors).

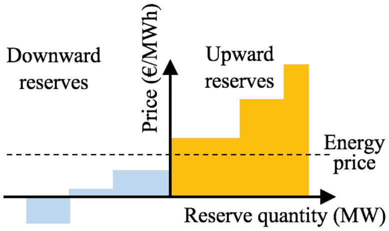

Fig. 10. Merit-order activation of reserves.

Two cases can occur. On the one hand, if the wind producer generates less than expected (i.e. positive error ε_t), upward reserve will be activated, and he will pay the resulting activation price (which is higher than the price he has received in the energy market). This penalty cost Λ^+ is calculated by (9). On the other hand, if the generation exceeds the forecasted value (i.e. negative error ε_t), the producer will sell the surplus energy at the downward activation price (which is lower than the price that he would have received in the energy market). The resulting opportunity loss Λ^- is calculated by (10).

$$\Lambda^+ = \sum_{t=1}^{n} (\lambda_t^{res+} - \lambda_t^{\mathrm{DA}}) \cdot \varepsilon_t \quad \text{(only when } \varepsilon_t > 0) \tag{9}$$

$$\Lambda^- = \sum_{t=1}^{n} (\lambda_t^{\mathrm{DA}} - \lambda_t^{res-}) \cdot |\varepsilon_t| \quad \text{(only when } \varepsilon_t < 0) \tag{10}$$

with λ_t^{res+} and λ_t^{res-} the upward and downward reserve prices, and λ_t^{DA} the electricity price on the day-ahead market.

The financial shortfall over the year 2017 (for each prediction tool) is computed using the actual price-quantity offers in the Belgian reserve market [28], and the results are given in Table IV. We see that all recalibrated models reduce the shortfall of the static forecaster, up to a factor 2 for the model relying on an optimal number of epochs. This impressive gain is explained by the merit order effect (Fig. 10), in which large deviations are more heavily penalized. Hence, even slight improvements can significantly reduce the balancing fees. In addition, we also observe that opportunity losses Λ^- are much higher than penalty costs Λ^+, which arises from the fact that the price spread between the energy price λ_t^{DA} and the price for the generation surplus λ_t^{res-} is usually much higher than the difference between λ_t^{DA} and λ_t^{res+}. Wind producers are thus incentivized to overestimate their future generation (and thus to pay the moderate penalty λ_t^{res+}) rather than to receive the very low λ_t^{res-} when they generate more than expected.

We conclude that relying on an (optimally-calibrated) model allows to save 3.3 M€ (for the reserve capacity) and 45 M€ (for the reserve activation) compared to a static model.

TABLE IV
ANNUAL ENERGY COSTS FOR WINDS PRODUCERS.

	Λ^+ [M€]	Λ^- [M€]	Shortfall [M€]
TSO	5.78	66.54	72.32
Static	13.13	78.28	91.41
Val.	19.09	30.05	49.14
Scratch	14.6	57.42	72.02
Epoch	17.82	27.29	45.11

V. CONCLUSION

This paper was devoted to the day-ahead prediction of the onshore wind power generation. Firstly, we exploited the flexible nature of recurrent neural networks to implement different LSTM-based topologies, which all provided accurate results in regards to other state-of-the-art approaches. Secondly, we observed that recalibrating the model during its actual utilization can strongly improve the accuracy of predictions. In that regard, it appears that a recalibration with a fixed (optimally-chosen) number of epochs is a very effective solution compared to the traditional use of a validation set. Finally, we quantified the financial impact of prediction errors on both the TSO and wind producers. It was observed that, due to the structure of the balancing costs, even small prediction improvements can lead to substantial costs savings [29]. Such results are expected to be further exacerbated if one consider smarter operations of wind turbines [30]-[31], thereby paving the way to further research in wind forecasting.

VI. ACKNOWLEDGEMENTS

The work is supported via the energy transition funds project ”BEOWIND” organized by the FPS economy, S.M.E.s, Self-employed and Energy.

REFERENCES

[1] S. Rajagopalan and S. Santoso, ”Wind power forecasting and error analysis using the autoregressive moving average modeling,” *IEEE Power & Energy Society General Meeting*, Calgary, pp. 1-6., 2009.

[2] J. W. Taylor, P. E. McSharry and R. Buizza, ”Wind Power Density Forecasting Using Ensemble Predictions and Time Series Models,” *IEEE Trans. Energy Conver.*, vol. 24, no. 3, pp. 775-782, Sept. 2009.

[3] J. Figa-Saldaña, J. J. W. Wilson, E. Attema, R. Gelsthorpe, M. R. Drinkwater and A. Stoffelen, ”The advanced scatterometer (ASCAT) on the meteorological operational (MetOp) platform: A follow on for European wind scatterometers,” *Canadian Journal of Remote Sensing*, vol. 28, no. 3, pp. 404-412, 2002.

[4] G. N. Kariniotakis, G. S. Stavrakakis and E. F. Nogaret, ”Wind power forecasting using advanced neural networks models,” *IEEE Trans. Energy Conver.*, vol. 11, no. 4, pp. 762-767, Dec. 1996.

[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] J.-F Toubeau, et al., ”Deep Learning-Based Multivariate Probabilistic Forecasting for Short-Term Scheduling in Power Markets,” *IEEE Trans. Power Syst.*, vol. 34, no. 2, pp. 1203-1215, March 2019.

[7] Y. Li, H. Shi, F. Han, Z. Duan, H. Liu, ”Smart wind speed forecasting approach using various boosting algorithms, big multi-step forecasting strategy,” *Renewable Energy*, vol. 135, May 2019.

[8] J. Wang, T. Niu, H. Lu, W. Yang and P. Du, ”A Novel Framework of Reservoir Computing for Deterministic and Probabilistic Wind Power Forecasting,” *IEEE Trans. Sust. Energy*, vol. 11, no. 1, pp. 337-349, Jan. 2020.

[9] Z. Peng et al., ”A novel deep learning ensemble model with data denoising for short-term wind speed forecasting,” *Energy Conversion and Management*, vol. 207, March 2020.

[10] J. Lago, F. De Ridder, B. De Schutter,”Forecasting spot electricity prices: Deep learning approaches and empirical comparison of traditional algorithms”, *Applied Energy*, vol. 221, pp. 386-405, 2018.

[11] P.G. Sansom, C.A.T. Ferro, D.B. Stephenson, L. Goddard, S.J. Mason, ”Best practices for post-processing ensemble climate forecasts, part I: selecting appropriate recalibration methods,” *Journal of Climate*, no. 29, pp. 7247-7264, 2016.

[12] Y. Wang, N. Zhang, Y. Tan, T. Hong, D. S. Kirschen and C. Kang, ”Combining Probabilistic Load Forecasts,” *IEEE Trans. Smart Grid*, vol. 10, no. 4, pp. 3664-3674, July 2019.

[13] T. Li, Y. Wang and N. Zhang, ”Combining Probability Density Forecasts for Power Electrical Loads,” *EEE Trans. Smart Grid*, in press.

[14] Y. Lin, M. Yang, C. Wan, J. Wang and Y. Song, ”A Multi-Model Combination Approach for Probabilistic Wind Power Forecasting,” *IEEE Trans. Sust. Energy*, vol. 10, no. 1, pp. 226-237, Jan. 2019.

[15] M. Sun, C. Feng, J. Zhang, ”Multi-distribution ensemble probabilistic wind power forecasting,” *Renewable Energy*, vol. 148, April 2020.

[16] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[17] A. Mashlakov, L. Lensu, A. Kaarna, V. Tikka and S. Honkapuro, ”Probabilistic Forecasting of Battery Energy Storage State-of-Charge under Primary Frequency Control,” *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 1, pp. 96-109, Jan. 2020.

[18] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, pp. 436–444, 2015.

[19] Y. Zhao, L. Ye, P. Pinson, Y. Tang and P. Lu, ”Correlation-Constrained and Sparsity-Controlled Vector Autoregressive Model for Spatio-Temporal Wind Power Forecasting,” *IEEE Trans. Power Syst.*, vol. 33, no. 5, pp. 5029-5040, Sept. 2018.

[20] M. Sun, C. Feng, J. Zhang, ”Conditional aggregated probabilistic wind power forecasting based on spatio-temporal correlation,” *Applied Energy*, vol. 256, Dec. 2019.

[21] 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,” *Proc. 7th IEEE Conf. Innovative SmartGrid Technol.*, Dec. 2017.

[22] D. P. Kingma and J. L. Ba, ”Adam : A method for stochastic optimization,” 2014. arXiv:1412.6980v9.

[23] N. Kayedpour, A. E. Samani, J. De Kooning, L. Vandevelde and G. Crevecoeur, ”A data-driven approach using deep learning time series prediction for forecasting power system variables,” *IEEE 2nd International Conference on Renewable Energy and Power Engineering*, 2019.

[24] X. Liao, N. Cao, M. Li and X. Kang, ”Research on Short-Term Load Forecasting Using XGBoost Based on Similar Days,” *International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS)*, Changsha, China, 2019, pp. 675-678.

[25] J. Bottieau, L. Hubert, Z. De Grève, F. Vallée, J.-F. Toubeau, ”Very Short-Term Probabilistic Forecasting for a Risk-Aware Participation in the Single Price Imbalance Settlement” *IEEE Trans. Power Syst.*, in press.

[26] K. De Vos, N. Stevens, O. Devolder, A. Papavasiliou, B. Hebb and M.-D. Johan, ”Dynamic Dimensioning Approach for Operating Reserves: Proof of Concept in Belgium,” *Energy Policy*, vol. 124, pp. 272-285, 2019.

[27] J.-F Toubeau, Z. De Grève and F. Vallée, ”Medium-Term Multimarket Optimization for Virtual Power Plants: A Stochastic-Based Decision Environment,” *IEEE Trans. Power Syst.*, vol. 33, no. 2, pp. 1399-1410, March 2018.

[28] Elia website, ”Available regulation capacity,” https://www.elia.be/en/grid-data/balancing/available-regulation-capacity.

[29] L. Exizidis, J. Kazempour, P. Pinson, Z. De Grève and F. Vallée, ”Impact of Public Aggregate Wind Forecasts on Electricity Market Outcomes,” *IEEE Trans. Sust. Energy*, vol. 8, no. 4, pp. 1394-1405, Oct. 2017.

[30] J. Van de Vyver, J. De Kooning, B. Meersman, L. Vandevelde, and T. Vandoorn, ”Droop control as an alternative inertial response strategy for the synthetic inertia on wind turbines,” *IEEE Trans. Power Syst.*, vol. 31, no. 2, pp. 1129–1138, 2016.

[31] J. Bottieau, F. Vallée, Z. De Grève and J.-F. Toubeau, ”Leveraging provision of frequency regulation services from wind generation by improving day-ahead predictions using LSTM neural networks,” *IEEE International Energy Conference (ENERGYCON)*, pp. 1-6, 2018.