Using Reservoir Computing for Wind Ramp Events Classification and Prediction

Tatyana M. P. Santos, Mêuser J. S. Valença
Polytechnic School of Pernambuco, POLI-UPE
Recife, Brazil
{tmps, meuser}@ecomp.poli.br

***Abstract*—The increasing use of wind power as source of electricity motivates a continuous improvement of the accuracy of wind power forecasts. There is a considerable value in optimizing forecasts systems to provide the best performance in an environment where the wind power production increases and/or decreases by a large amount over a short period of time. This paper presents a model that uses Reservoir Computing to classify energy production variations in wind farms, known as ramp events. This method is compared with two other approaches: one that uses a MLP network and the other is based in Persistence. The tests were performed and the results are given for real cases, reaching up over 90% of success rate.**

Keywords-ramp events; wind power forecast; reservoir computing; neural networks; mlp.

I. INTRODUCTION

In recent years, with the large-scale expansion of wind farms, the percentage of energy derived from wind sources is increasing rapidly. Thus, the demand for more reliable wind energy is driving the need for detection and prediction of ramp events [1].

There are wind power predictive models that are based on physical characteristics, of the weather, of the terrain, and thus dependent on the physical aspects. Recent research works in wind power forecasting, however, have focused on associating uncertainty estimates to these point forecasts, using historical measurements and machine learning algorithms to induce a predictive model [2][3]. In the following section, there are some examples of works done in this area, but no model was found, even among those which use machine learning algorithms, with the same techniques compared in this work for the same purpose (classify and predict ramp rates).

The series representing power generation in wind farms are very dynamic, predisposed to many variations. These series oscillate a lot in short periods of time, since it suffers various influences of physical and meteorological factors, which requires the use of a technique that handles very well with this volatility.

If properly applied, these works have much to add in wind power generation, increasing significantly the value of this modality in our energy matrix.

Today, a major difficulty when it comes to the prediction accuracy of wind power is to provide a forecast able to handle extreme situations, these situations that still rely heavily on the activities of end users, who need to develop procedures that meet the electricity demand, as well as maximizing the economic and environmental benefits. These extreme events are associated with large deviations on power generation compared with what was expected. The severity of these events depends on the speed with which they occur and when they occur, because the demand for electricity is also highly uncertain. The sooner these events are planned, the most effective are the procedures [2][3].

One solution to be considered is to try to determine in advance, and with the best possible accuracy, the timing, the amplitude and the width of the variations of the power generated. In this work, we try to optimize this type of solution using this *Reservoir Computing* model.

The remaining of this paper is organized as follow. Section 2 presents some related work, Section 3 addresses the proposed model based on *Reservoir Computing*, a recurrent neural network approach, whose structure is discussed, in addition with how it was applied and why it was chosen. Still talking about the proposed model, it was explained a little about ramp events, bringing the concept and defining the parameters considered in this work. In Section 4, the experiments are presented, showing the improvements compared to the results from a model using a *Multi-Layer Perceptron* (MLP) neural network and a second model based on the Persistence concept. Section 4 also explains a little about the common use of Persistence models as a reference predictor. The final remarks and future works are discussed in the conclusion section.

II. RELATED WORK

As discussed in introduction, physical-based models are still most common in this area, such as weather and terrain-based. These models use to determine relationships between the physical aspects and wind farms output power [4][5][6].

The other approach is the mathematical modelling, in which statistical and/or artificial intelligence methods are used to find the relationship between historical data sets and wind farms output power [7][8][9][10].

In the last decade, there was strong research effort on the improvement of wind power predictions using meteorological forecast data from Numerical Weather Prediction (NWP) systems. NWP systems uses mathematical models of the atmosphere and oceans to predict the weather based on current weather conditions data.

In the European project ANEMOS, several prediction models (multi-model approach) were applied and compared for the prediction of selected wind farms located in areas

 ISBN: 978-1-61208-340-7

with different characteristics. The ANEMOS project develops intelligent management tools for addressing the variability of wind power.

In other studies, strong improvements, up to about 20%, were obtained by using the data of different NWP models or ensemble models as input data for the wind power prediction models [11][12][13].

Within the short-term context, time series based models have shown a better performance than NWP models for horizons up to few hours [14][15][16]. These models, as the model brought in this paper, try to learn and replicate the dynamic shown by certain time series, for instance the power output time series of a wind farm.

For wind ramps forecasting, there are studies which showed improvements of predictions by using statistical and/or artificial methods too [17][18][19] [20].

Any work was found related to our case study regarding with *Reservoir Computing*.

III. Proposed model

A. Theoretical background

There are certain complex situations and problems which are part of our reality. These facts have stimulated and continue boosting research aimed at bringing computing solutions to what could not be solved in a most trivial way. Many interesting and challenging problems in engineering also do not have direct solution using heuristic methods or algorithms explicitly programmed. These problems are prime candidates for the application of machine learning methods [21], which share the common property of learning by example and being able to generalize these examples in a "smart" way to new entries yet unseen.

A large subclass of machine learning methods is formed by Artificial Neural Networks, which are very abstract connectionist models of how the brain makes computing. They consist of networks of simple and non-linear computational nodes that communicate values via weighted connections, i.e., having their respective weights. Mainly, these weights are computed through features extracted from examples in such way that the desired behavior of the network is reached. If the network has a recurrent structure, i.e., with feedback loops, then it will have a memory of past inputs, which allows it to make the processing of temporal signals making them powerful computational nonlinear methods [21].

In Fig.1, there are two examples of neural networks topologies. At left, a feedfoward network, without feedback, where the signal travels through the network in a single direction. At right, a recurrent network, with feedback loops, that provide memory of past input, as explained above.

These recurrent neural networks are, however, notoriously difficult to train. A new learning paradigm called *Reservoir Computing* (RC) was introduced, allowing the use of recurrent neural networks alleviating the consuming and difficult phase of training. This idea was emerged simultaneously from the Echo State Networks [22] and Liquid State Machines [23] approaches, proposed independently and in periods very close (2001 and 2002 respectively). In both networks, the architecture consists of a recurrent network of neurons, we call this reservoir, which is built randomly and not trained initially, and a separate linear output layer, trained by simple one-shot methods [21], i.e., do not require large data sets or multiple training rounds, leaving to the discretion of the modeler doing training in batches, if desired.

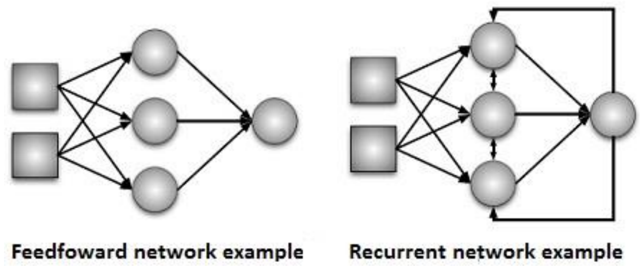


Figure 1. Differences between network topologies.

Fig. 2 below shows a schematic representation of a network with *Reservoir Computing* approach. The fixed connections and randomly formed are indicated with a solid line, the trained connections are indicated with a dashed line.

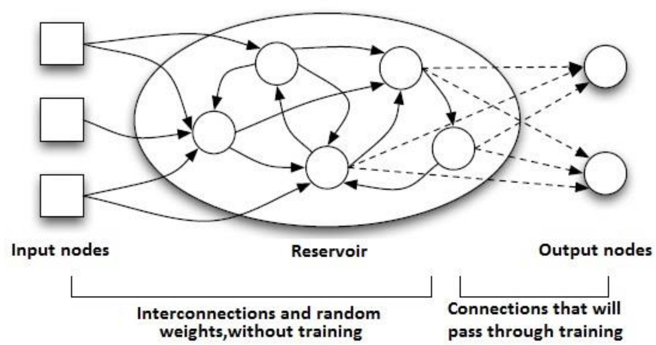


Figure 2. Schematic representation of a network with *Reservoir Computing* approach.

Since its introduction, *Reservoir Computing* has attracted much attention in the community of neural networks, due to the combination of simplicity of use and its good performance in a variety of difficult benchmark tasks [21]. Therefore, *Reservoir Computing* is used in our proposed model in the classification of ramp events task. Basically, ramp events can be understood as a variation on the nominal power greater than a threshold that lasts for a certain period of time. A more concrete definition about ramp events is presented in the following section.

B. Databases preparation for ramp events classification

Recently, the wind power industry began to evaluate the nature of ramp events. There is still no universal acceptance threshold to detect them [22], the most commonly used concept is the one that defines the ramp event as a variation that exceeds a minimum percentage of the nominal power (V_{min}) in a wind farm within a time period less than or equal to a maximum (T_{max}) [2][22], that is, when there is an alteration in the output power that has an amplitude sufficiently large for a relatively short period of time [3]. It is difficult to find a consensus between the values for V_{min} and T_{max}, because they usually depend on geographical situation, climate and complexity of the terrain and end up being set "arbitrarily" by the solution modeler [2].

Fig. 3 below shows an example based on Ferreira [3] where the ramps are defined as a change in power of at least 50% of the capacity over a maximum duration of 4 hours.

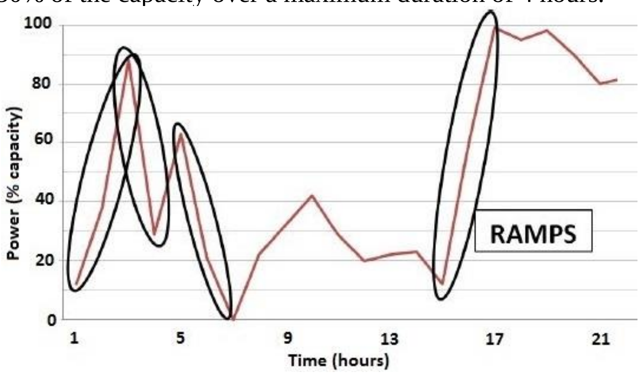


Figure 3. Ramp event definition: for this image, a change in power of at least 50% of the capacity over a maximum duration of 4 hours (Ferreira, 2010).

In this work, the behavior of the databases fairly reflected what was previously said, the values for V_{min} and T_{max} vary considerably according to local conditions of the wind farm. To define the parameters and continue favoring the individual behavior of generation in each wind farm, we chose not to set a universal value for V_{min} and T_{max}, but to do it according to the number of events observed in accordance with the variation of these parameters. The ramp should be a sparse event, not repeated many times, so the event does not become widespread throughout the generation. To process the data, we assume that the number of values found above the threshold V_{min} should be about 10% of total. To find these quantities of values observed at each threshold, the following process was made:

1) Apply filters that make it possible to view the percentage of variation of the series;

2) Count occurrences for each variation percentage;

3) For each variation percentage, check how much the ramps of this threshold represent in relation to the whole.

For the first step of the process, we based on an approach that does not work directly with the sign of the power generated in the farm, but turns the signal into a more appropriate representation. This strategy is used by Bossavy et al. (2010), who consider n differences in the amplitude of the power generated. In this procedure, let (p_t)t as the time series of wind power, and (p_t^f)t as the filtered signal associated, obtained by the following equation:

$$p_t^f = \text{mean}\ \{p_{t+h} - p_{t+h-n_{am}}\ ;\ h = 1, ..., n_{am}\} \qquad (1)$$

In (1), the n_{am} comes from the amount of differences in power measurements to be considered in the average (n_{am} = number of averaged differences of measures). The filtered signal (p_t^f)t measures variations of the wind power series (p_t)t. The ramp event then corresponds to a time interval where the absolute value of the filtered signal (p_t^f)t exceeds a threshold t>0. The ramp time is the point where the filtered signal (p_t^f)t reaches a local maximum. Figures 4, 5, and 6 demonstrate part of the analysis done until we could choose the appropriate n_{am}.

In Fig. 4 below, there is a part of the power generation time series on Farm A (p_t)t and absolute values of the filtered signal (p_t^f)t with parameter n_{am} = 2. Ramps in power series coincide with the local peaks of the filtered signal. Considering the threshold of 20% of the nominal power, for example, we observe 6 ramp events on this stretch.

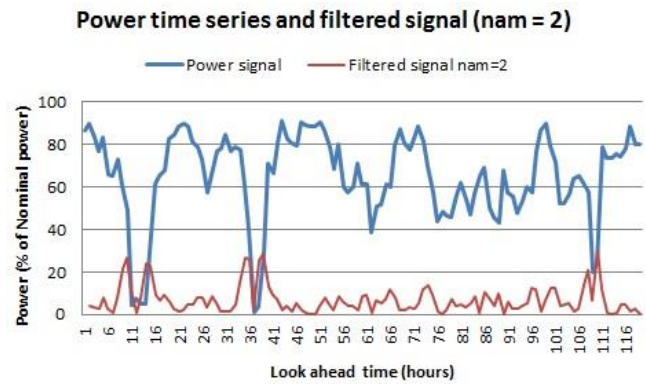
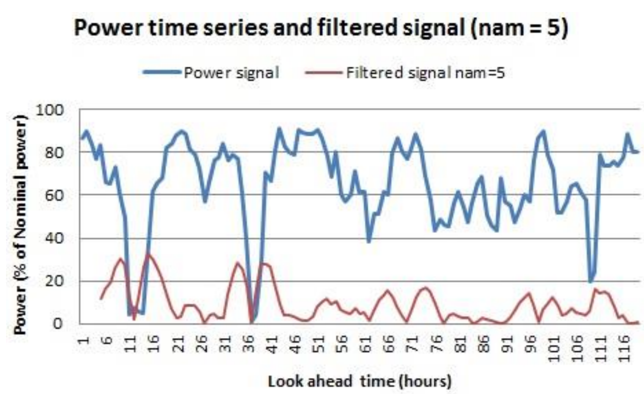


Figure 4. Part of the power generation time series on Farm A (p_t)t and absolute values of the filtered signal (p_t^f)t with parameter n_{am} = 2.

In Fig. 5 below, the parameter n_{am} was set as 5. Considering the threshold of 20% of the nominal power too, 4 ramp events can be observed on this stretch. The identification of the ramps observed at t = 108 and 110 in Fig. 4 is lost, this happens because lower values of n_{am} do the filtered signal (p_t^f)t be more sensitive to variations in power series occurred in shorter periods of time.



$\alpha + \beta = \chi$. (1) (1)

Figure 5. Part of the power generation time series on Farm A (p_t)t and absolute values of the filtered signal (p_t^f)t with parameter n_{am} = 5.

In Fig. 6 below, the parameter n_{am} was set as 10. It is possible to observe that bigger values of n_{am} will result in a filtered signal increasingly less sensitive to variations in the generated power.

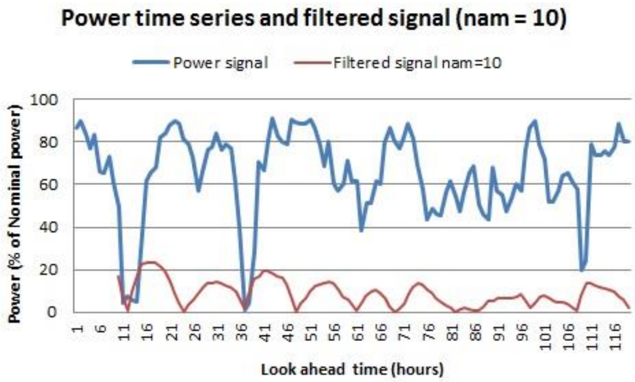


Figure 6. Part of the power generation time series on Farm A (p_t)t and absolute values of the filtered signal (p_t^f)t with parameter n_{am} = 10.

This work was done using power generation time series of three wind farms in Brazil, here we call them Farm A, Farm B and Farm C. These wind farms range in capacity 54.6, 70.56 and 126 MW respectively. This dataset is available for research purpose under request. The following figures were extracted from the Farm A series.

With the three power series used as case study in this paper (Farm A, Farm B and Farm C), using the previously detailed definition, the value of n_{am} remained 5 for the three cases and the thresholds were defined as: 15% to the Farm A, 20% to the Farm B and 40% for the Farm C.

890 values were found exceeding the threshold of 15% over a period of 5,855 hours with measurement occurring 30 to 30 minutes in the Farm A, 1236 values exceeding the threshold of 20% over a period of 8,784 hours (one year) with measurement occurring 30 to 30 minutes in the Farm B and 1,243 values exceeding the threshold of 40% over a period of 8,784 hours (one year) with measurement occurring 30 to 30 minutes in the Farm C.

The series were transformed into an hourly measurement before setting the n_{am}s, then n_{am} = 5 means 5 hours and n_{am} = 2 means 2 hours, i.e., for the Farm A, for example, the definition of the ramp is a variation of 15% in a period of 5 hours or less.

Figure 7 portrays these situations more easily and also shows that we would have more examples of ramps opting to use n_{am} = 5 instead of n_{am} = 2. This figure also presents that choosing a threshold below 15%, 10% for example, more ramps could be observed, however, as said previously, would increase greatly the sensitivity of the filter and the whole generation would be filled with ramps, generalizing too much the event.

In the following section, the experiments and results of ramp events classification are discussed. Then, the objective was to train the model to signalize in which periods the ramp events may occur.

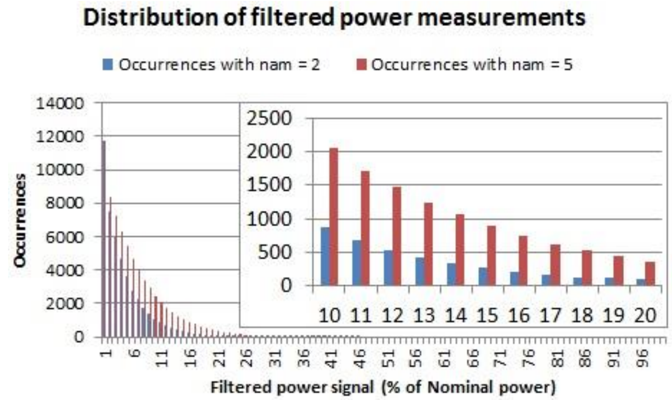


Figure 7. Filtered signal (p_t^f)t of Farm A considering n_{am} = 2 and n_{am} = 5 to verify the amount of occurrences exceeding all thresholds.

In addition with the proposed system based on *Reservoir Computing*, the experimental results of systems based on the both methods are presented: MLP network and a Persistence model.

IV. EXPERIMENTS AND RESULTS COMPARISONS FOR WIND RAMPS CLASSIFICATON

A. *Used settings*

The experiments were made using the last 24 hours to predict 24 hours ahead, but it can be parameterized for other options.

For neural networks, both of the *Reservoir* and for the MLP, we used 48 input neurons and 48 output neurons. The 48 entries relate to a full day of measurement, occurring in 30 to 30 minutes, the 48 output neurons correspond to 5-hour intervals (each interval has 10 values, since the measurement are arranged in 30 to 30 minutes) from the first hour of the day until the 9th measument of the subsequent month, since this was the maximum period set for detecting ramp events using 100% of power from one day. To find the amount of neurons in the hidden layer and other settings specifics to each type of network (interconnectivity rate and warm up cycles for RC and the learning rate and moment for MLP), we performed tests between possible configurations, explained in the following section, comparing the percentage of correct classifications (*Success Rate*) and checking if the choice had generalization capability to perform a good classification over all wind farms in study. These values will be showed and discussed in results comparisons section.

B. *Results comparisons*

To compare the classifications performed by the models employed in this work, and choose the best one, we used Success Rate (SR), as said before.

1) *Reservoir Computing*

With RC, as shown in Table I, we have achieved a success rate of 76.85% with standard deviation of 4.49 in Farm A, 80.48% with standard deviation of 3.93 in Farm B and 91.52% with standard deviation of 2,71 in Farm C.

TABLE I. RESULTS FOR CLASSIFICATION USING A RC MODEL

	Success Rate	Deviation
Farm A	76.85%	±4.49
Farm B	80.48%	±3.93
Farm C	91.52%	±2.71

The tests were conducted always incrementing the number of neurons in the hidden layer, starting with 5 neurons until 100, the rate of interconnectivity between the neurons of the hidden layer, from 50% to 100%, and the amount of warm up cycles from 1 to 100, but after 4 warm up cycles, any positive difference detected is very low.

Figures 8, 9 and 10 below show how the success rate changes due to the number of neurons for Farms A, B and C. The x-axis and y-axis correspond to executions and the number of neurons, respectively.

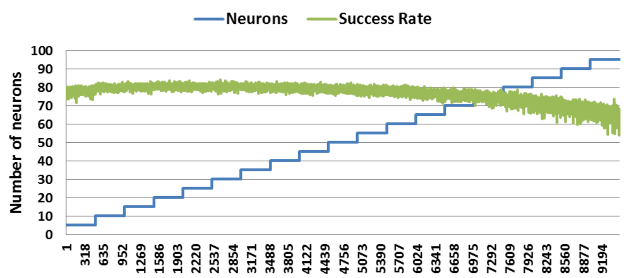


Figure 8. Changes in Success Rate according to the number of neurons in the hidden layer for Farm A data.

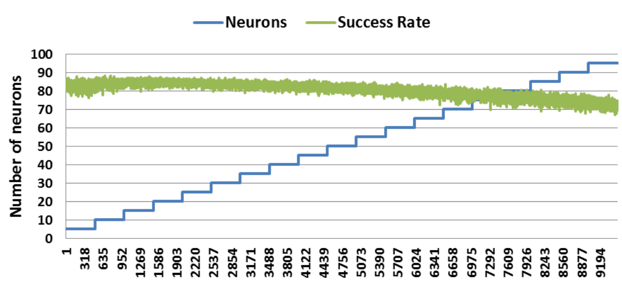


Figure 9. Changes in Success Rate according to the number of neurons in the hidden layer for Farm B data.

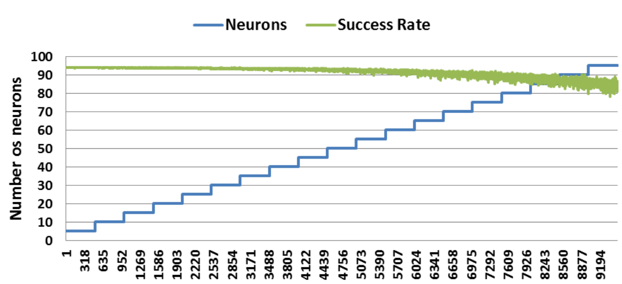


Figure 10. Changes in Success Rate according to the number of neurons in the hidden layer for Farm C data.

The blue line, referring to the number of neurons, remains at the same level during a sequence of executions because other parameters are changed (warm-up cycles and rate of interconnectivity).

Despite the small deviation, it's important to mention that each wind farm presented a different configuration for the best result. This information is on the Table II below:

TABLE II. BEST RESULTS FOR CLASSIFICATION USING RC MODEL

	Number of neurons in hidden layer	Interconnectivity rate	Warm up cycles	Success Rate
Farm A	30	64%	36	84.02%
Farm B	10	76%	11	88.13%
Farm C	10	94%	66	93.93%

2) *MLP*

With the MLP tool, no significant difference was detected in results between the tested configurations. This scenario occurred for the three wind farms used as case study.

According to Table III below, in Farm A, the success rate ranged an average of 71.57% with standard deviation of 0.49 in Farm B, ranged an average of 83.07% with standard deviation of 0.52 and in Farm C, ranged an average of 90.28% with standard deviation of 0.30.

TABLE III. RESULTS FOR CLASSIFICATION USING MLP MODEL

	Success Rate	Deviation
Farm A	71.57%	±0.49
Farm B	83.07%	±0.52
Farm C	90.28%	±0.30

Despite the small deviations, it's important to mention that each wind farm presented a different configuration for the best result. This information is on the Table IV below:

TABLE IV. BEST RESULTS FOR CLASSIFICATION USING MLP MODEL

	α	β	Number of neurons in hidden layer	Success Rate
Farm A	0.7	0.4	10	72.35%
Farm B	0.6	0.3	120	83.76%
Farm C	0.4	0.7	45	91.87%

3) *Persistence*

As Kariniotakis [24] reports, it is worthwhile to use operationally an advanced tool for wind forecasting only if this is able to outperform naive techniques resulting from simple considerations without special modeling effort. Such simple techniques are used as reference to evaluate advanced ones. The most commonly used reference predictor is Persistence. This approach states that the future wind generation will be the same as the last measured power value.

Despite its apparent simplicity, this model might be hard to beat for the first look-ahead times (up to 4-6 hours). This is due to the scale of changes in the atmosphere, which are relatively slow, in order of days [24].

Here, it was considered that the occurrence of ramps the next day is the same as the present day, at the same moments. For example, if we had ramp events at 12 o'clock today, we will consider the occurrence of ramps at 12 o'clock tomorrow. Assuming that, the moments that this model hits the occurrence of ramps will be counted in the success rate. The Table V below shows the results found for wind farms A, B and C:

TABLE V. RESULTS FOR CLASSIFICATION USING A PERSISTENCE MODEL

	Success Rate
Farm A	41.87%
Farm B	43.92%
Farm C	12.87%

C. Improvement calculation

The improvement between the RC model and the two other reference models was calculated, based, for classification, on the best success rate found.

The improvement is calculated as in (2):

$$Improvement = \frac{SR_{model1} - SR_{model2}}{SR_{model1}} * 100 \qquad (2)$$

The results obtained are shown in the Table VI below:

TABLE VI. IMPROVEMENT CALCULATION BETWEEN COMPARED MODELS FOR CLASSIFICATION

Compared models	**Wind farm**	***Improvement***
Reservoir Computing x MLP	A	13.89%
	B	4.96%
	C	2.19%
Reservoir Computing x Persistence	A	50.17%
	B	50.16%
	C	86.30%
MLP x Persistence	A	42.13%
	B	47.56%
	C	85.99%

With the used metric, the RC model has shown better results than the MLP and the Persistence model, as we can see above.

As the RC model offers recurrence between neurons, it was expected that it would provide better results, as discussed in theoretical background in Section 3. This expectation was met.

Besides the recurrence, *Reservoir Computing* has a simpler way of training, as explained in Section 3 too. This feature helps to create models that tend to use less processing time compared with other recurrent neural networks.

CONCLUSION AND FUTURE WORK

In this work, a model that uses a recurrent neural network with differentiated learning method, called *Reservoir Computing*, was proposed for trying to promote better results in ramp events classifications and in power generation forecast in wind farms. A MLP neural network and a Persistence model were used as reference models.

The results indicate that the proposed model has better performance compared with reference models for classifying ramp events, reaching up over 90% of success rate.

As prospects for future works, is suggested the investigation of new options of input variables and architectures for the RC, as well as further support to indicate the amplitude of wind energy ramp events.

REFERENCES

[1] C. Kamath, "Understanding Wind Ramp Events through Analysis of Historical Data," Proc. IEEE PES Transmission and Distribution Conference and Expo (T&D), Apr. 2010, pp. 1-6.

[2] A. Bossavy, R. Girard, and G. Kariniotakis, "Forecasting Uncertainty Related to Ramps of Wind Power Production," Proc. European Wind Energy Conference & Exhibition (EWEC 2010), Apr. 2010, pp. 20-23.

[3] C. Ferreira, J. Gama, L.Matias, A. Botterrud, and J. Wang, "A Survey on Wind Power Ramp Forecasting", Argonne National Laboratory Technical Report, Chicago, USA, 2010.

[4] H. G. Beyer, D. Heinemann, H. Mellinghoff, K. Mönnich, and H. P. Waldl, "Forecast of Regional Power Output of Wind Turbines," Proc. European Wind Energy Conference & Exihibition (EWEC 1999), Mar. 1999, pp. 1070-1073.

[5] G. Giebel et al., "Short-term Forecasting Using Advanced Physical Modelling – The Results of the Anemos Project, Results from mesoscale, microscale and CFD modeling," Proc. European Wind Energy Conference & Exihibition (EWEC 2006), Feb. 2006.

[6] M. Lange, and U. Focken., "Physical Approach to Short-Term Wind Power Prediction," Berlin: Springer-Verlag, p. 208, 2006.

[7] G. Giebel, L. Landberg, G. Kariniotakis, and R. Brownsword, "The State-Of-The-Art on Methods and Software Tools for Short-Term Prediction of Wind Energy Production," Proc. European Wind Energy Conference & Exihibition (EWEC 2003), Jul. 2006.

[8] T. S. Nielsen et al., "Short-term Wind Power Forecasting Using Advanced Statistical Methods," Proc. European Wind Energy Conference & Exihibition (EWEC 2006), Feb. 2006.

[9] I. Sánchez, "Short-term prediction of wind energy production," International Journal of Forecasting, vol. 22, Jan.-Mar. 2006, pp. 43 – 56, doi:10.1016/j.ijforecast.2005.05.003.

[10] R. Jursa, "Wind power prediction with different artificial intelligence models," European Wind Energy Conference & Exihibition (EWEC 2007), May 2007.

[11] L. von Bremen, "Optimal linkage of different NWP models with neural networks for offshore wind power predictions," Sixth International Workshop on Large-Scale Integration of Wind Power and Transmission Networks for Offshore Wind Farms, Oct. 2006, pp. 291-296.

[12] M. Lange et al., "Optimal combination of different numerical weather models for improved wind power predictions," Sixth International Workshop on Large-Scale Integration of Wind Power and Transmission Networks for Offshore Wind Farms, Oct. 2006, pp. 273-276.

[13] G. Gregor et al., "Wind Power Predictions using Ensembles," Risø National Laboratory, Technical Report, Roskilde, Denmark, Sep. 2005.

[14] A. Costa, "Mathematical/Statistical and Physical/Meteorological Models for Short-term Prediction of Wind Farms Output," Ph. D Thesis, Escuela Técnica Superior de Ingenieros Industriales, Universidad Politécnica de Madrid, 2005.

 ISBN: 978-1-61208-340-7

[15] G.Giebel, R. Brownsword, G. Kariniotakis, M. Denhard, C. Draxl, “The state of the art in short-term prediction of wind power – A literature overview,” ANEMOS EU Project, Technical Report, 2011.

[16] H. Zheng, and A. Kusiak, “Prediction of Wind Farm Power Ramp Rates: A Data-Mining Approach,” Journal of Solar Energy Engineering, vol. 131, Jul. 2009, pp. 031011-1–031011-8, doi:10.1115/1.3142727.

[17] C. Gallego, A. Costa, and A. Cuerva, "Improving short-term forecasting during ramp events by means of Regime-Switching Artificial Neural Networks," Proc. 8th European Conference on Applied Climatology (ECAC 2010), Adv. Sci. Res., Sep. 2010, pp. 55-58, doi:10.5194/asr-6-55-2011.

[18] D. Lee, “Short-Term Wind Power Ensemble Prediction Based on Gaussian Processes and Neural Networks,” IEEE Transactions on Smart Grid, vol. 5, Jan. 2014, pp. 501-510, doi: 10.1109/TSG.2013.2280649.

[19] R. Sevlian and R. Rajagopal, “Detection and Statistics of Wind Power Ramps,” IEEE Transactions On Power Systems, vol.28, Jan. 2013, pp. 3610-3620, doi: 10.1109/TPWRS.2013.2266378.

[20] A. Couto, P. Costa, L. Rodrigues, V. Lopes, and A. Estanqueiro, “Impact of Weather Regimes on the Wind Power Ramp Forecast,” Twelfth International Workshop on Large-Scale Integration of Wind Power and Transmission Networks for Offshore Wind Farms, Oct. 2013.

[21] D. Verstraeten, “Reservoir Computing: computation with dynamical systems,”, Doctoral Thesis, Eletronics and Information Systems, Gent, Ghent University, 2009.

[22] H. Jaeger, “The "echo state" approach to analysing and training recurrent neural networks,” GMD - German National Research Institute for Computer Science, Report 148, 2001.

[23] W. Maass, T. Natschlaeger, and H. Markram, “Real-time computing without stable states: A new framework for neural computation based on perturbations,” Neural Computation, vol. 14, Nov. 2002, pp 2531-2560, doi:10.1162/089976602760407955.

[24] G. Kariniotakis, P. Pinson, N. Siebert, G. Giebel, and R. Barthelmie, “The State of the Art in Short-term Prediction of Wind Power – From an Offshore Perspective,” Proc. Ocean Energy Conference ADEME-IFREMER, Oct. 2004.