A New Design Methodology to Predict Wind Farm Energy Production by Means of a Spiking Neural Network Based-System

S. Brusca[1], G. Capizzi[2], G. Lo Sciuto[2] and G. Susi[3]

[1]*Department of Engineering, University of Messina, Messina, Italy*
[2]*Department of Electrical, Electronics and Informatics Engineering, University of Catania, Catania, Italy*
[3]*Department of Electronic Engineering, University of Rome 'Tor Vergata', Rome, Italy*

SUMMARY

In this paper a spiking neural network-based architecture (SNN) for the prediction of wind farm energy production is proposed. The model is also able to evaluate the wake effects due to interactions between the elements of a wind farm on the energy production of the whole farm. This method has been applied to a large wind power plant, composed of 28 turbines and three anemometric towers, located in the rural area of Vizzini's municipality in province of Catania, Italy, that is characterised by a complex orography and an extension of 30 km^2. For the implementation of this architecture it was used the recent neuromorphic framework 'NeuCube'. The results show that the presented method can be successfully applied for predictions of wind energy generation in real wind farm also in presence of faults.

Received . . .

KEY WORDS: Wind, Wind power forecasting, Wind power plant, Spiking Neural Network, NeuCube

1. INTRODUCTION

The rapid development of the wind power industry requires an advanced computational analysis of the wind farm design and wake interactions. Many factors affect the power generation in wind turbines such as wind speed, altitude, tower height, turbine interactions, number of turbines and complexity of the wind farm [1].

In this paper, we present a model able to evaluate the wake effects due to interactions between the elements of a wind farm on the energy production of the whole farm. The model is useful to accurately calculate the daily energy generated from any of wind turbines (T_n), within the wind farm, on the basis of wind speed and direction measured by three distint anemometric towers. In particular a temporal stream data (TSD) regressor based on spiking neural network (SNN) has been trained in order to predict the energy amount generated by a given wind turbine within the wind farm taking in consideration exclusively the wind intensity and direction time series of the three anemometric towers.

Many authors have made important contributions in the field of machine learning to prove that the TSD regressors based on SNN are effective techniques for both classification and prediction tasks [2, 3, 4]. In fact SNN based machine learning architectures have demonstrated high performances in these kinds of problems because SNNs are capable to represent and integrate different information

features, such as time, space, frequency, phase, and to deal with large volumes of data [4]. In particular, the recent neuromorphic framework NeuCube [5], a multi-modular system based on brain-like information processing principles, has demonstrated its suitability to efficiently operate both classification and regression of TSD [6]. Our approach has been implemented by means of the NeuCube computational framework [5], for evolving Spatio Temporal Data Machines (eSTDM), based on Spiking Neural Network.

In this paper we have adopted the basic configuration of NeuCube, that is composed of three different blocks: *Data Encoding Block*, *SNN Block* and *Prediction Block*.

The model described in this paper has been applied to a large wind power plant, composed of 28 turbines and three anemometric towers, located in the rural area of Vizzini's municipality in province of Catania, Italy, that is characterised by a complex orography and an extension of 30 km^2. The measured dataset are averaged over a 10-minute period, and consists of the averaged active powers (kW, the averaged wind speeds (m/s), the averaged wind directions (degree). The data are recorded during the period starting from January 2011 to September 2013.

The accuracy of model have been determined by means of the Mean Absolute Deviation (*MAD*), the Mean Absolute Percent Error (*MAPE*), and the Mean-Root-Squared Error (*MRSE*).

The tests carried out with experimental data show very low values of *MAD*, *MAPE* and *MRSE*. It demonstrates that the model is suitable for this task and has a good generalization capability, therefore, the proposed model can be proficiently used to predict daily energy produced from a turbine, taking in consideration wind speed and direction measured by other elements of the same wind farm.

2. TURBINE-TURBINE INTERACTION: THE WAKE EFFECT

Wind turbines are operating in the lowest part of the atmospheric boundary layer of the Earth. This severely complicates the calculation of the flow around them. As many wind turbines in wind farms have to operate in the wakes of upwind turbines, they are exposed to incoming wind velocities that are smaller than those under unperturbed (unwaked) conditions. As a result, turbine wakes are responsible for important power losses in wind farms [7, 8, 9].

Extensive analytical, numerical and experimental efforts have been carried out to better understand and predict turbine wake flows. One of the pioneering analytical wake models is the one proposed by Jensen [10], which assumes a top-hat shape for the velocity deficit in the wake (see Figure 1).

The analysis is based on the assumption that the momentum is conserved inside the wake, neglecting the near field behind the wind generator. Figure 2 shows a wake scheme, where v_0 is the velocity of the wind velocity behind the rotor, U_∞ is the incoming wind velocity, v is the wind velocity in the wake at a distance x from the rotor, as well as r_0, the rotor radius and r, the wake radius at x distance.

For a linear wake it is possible to link the wake radius, the rotor radius, and the distance from the rotor in the wake using the entrainment constant α as reported in:

$$r = r_0 + \alpha x \tag{1}$$

The entrainment constant is empirically given by [11]

$$\alpha = \frac{0.5}{ln(z/z_0)} \tag{2}$$

where z is the hub height of the turbine and z_0 is the ground roughness of the site.

Moreover, the axial induction factor a [12] can be introduced to determine the air velocity just behind the wind turbine rotor:

$$v_0 = (1 - a)\, U_\infty \tag{3}$$

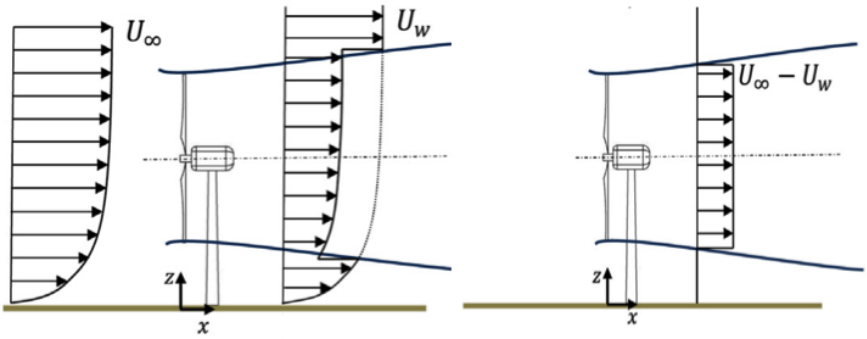

Figure 1. Schematic of the vertical profiles of the mean velocity (top) and velocity deficit (bottom) downwind of a wind turbine, where U_∞ is the incoming wind velocity and U_w the wake velocity in the streamwise direction.

After performing a momentum balance and taking into account the equations (1), (2), and (3) the following expression is derived to describe the wind speed downstream of the turbine:

$$v = U_\infty \left[1 - \frac{a}{(1 + \alpha(x/r_0))^2}\right] \tag{4}$$

In the instance of a wind turbine encountering multiple wakes, the kinetic energy of the mixed wake can be assumed to be equal to the sum of the kinetic energy deficits. This results in the following expression for the velocity downstream of N turbines:

$$v = U_\infty \left[1 - \sqrt{\sum_{i=1}^{N} \left(1 - \frac{v_i}{U_\infty}\right)^2}\right] \tag{5}$$

Power production from wind farms is significantly affected by wind turbine wakes. This power loss can be up to 25% of the total power output. For this reason, the accurate prediction of turbine wakes is imperative to minimize power losses and, thus, increase the overall efficiency of wind farms.

It is found that most related studies on wind farms utilize the wake loss model considered in this section (the Jensen's wake model) which is the simplest available method to date. This model gives an acceptable degree of prediction accuracy for the off-shore as well as for the flat terrain onshore wind farms.

This simple approach ignores all wind directions except the prevailing one, and it is therefore likely to lead suboptimal wind farm layouts. Furthermore, it fails to describe how to compute the power produced when the wind blows from a crosswind direction, in which case the Jensen model is not valid because the distance between the turbines that generate a wake and the turbines affected by it is too short.

To overcome these limitations and obtain a better degree of prediction accuracy we present a model based on SNN in order to predict the energy amount generated by wind turbine within the wind farm.

3. SPIKING NEURAL NETWORKS: SPIKING NEURONS, SYNAPSES AND PLASTICITY

The Spiking Neural Networks (SNNs) are similar to the traditional neural networks [13] but they also incorporate the concept of time into their basic elements (i.e., *spiking neurons*) and *synaptic weights*.

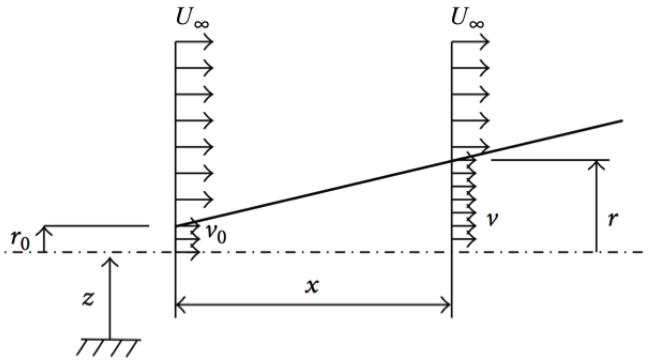

Figure 2. Wind turbine wake model.

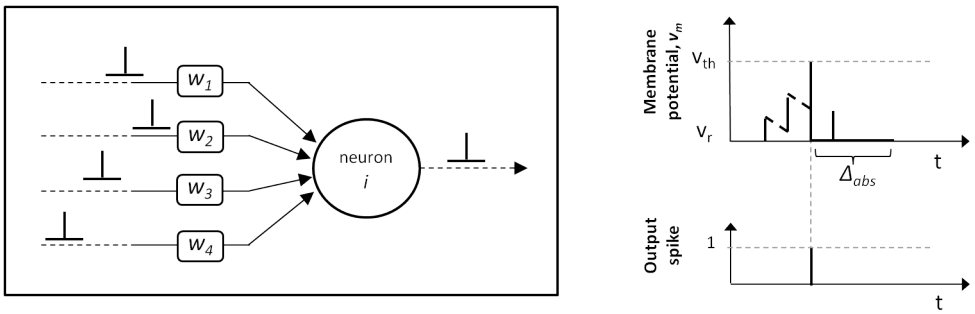

Figure 3. Schematic representation of LIF neuron model operation: input postsynaptic spikes are integrated by neuron *i*, each one with amplitude equal to the related synaptic weight. When the threshold is reached, an output spike is generated. Note that the more the input spikes are synchronized, the more the contributes are effective for the overcoming of v_{th}. After, the membrane potential is instantaneously reset to v_r and remains inactive for a period $\triangle_{abs}$ (that is why the fourth contribution has not been integrated by neuron *i*).

Spiking neurons are basic elements connected by synapses able to emit pulses. The choice of a spiking neuron model is usually a trade-off between biological plausibility and complexity [14, 15]. In NeuCube, the neuron model used is the Leaky Integrate and Fire (LIF) neuron model [5], that is very fast to simulate and particularly attractive for large-scale network simulations. As shown in Figure 3, the neuron behaves as a leaky integrator of its inputs until v_m (the neuron internal state) reaches the *firing threshold* v_{th}. In this case the neuron fires a spike and v_m is instantaneously reset to the *resting potential* v_r. If no input event occurs, the membrane potential (v_m) slowly decreases to zero with a proper behaviour (i.e.: *underthreshold decay*). In such framework the underthreshold decay behavior is approximated as linear, so that the neuron is defined by three values: a linear leak rate (l), a firing threshold (v_{th}), and an absolute refractory period ($\triangle_{abs}$).

In this case, starting from an initial value $v_{m,i}$, the membrane potential of neuron *i* will increase of the quantity $w_{i,j}$ every time that a spike arrives from neuron *j*:

$$v_{m,i}(t) = v_{m,i}(t-1) + w_{i,j} \tag{6}$$

Between spikes, the membrane potential leaks according to the leak rate:

$$v_{m,i}(t) = v_{m,i}(t-1) - l \tag{7}$$

If $v_{m,i}$ overcomes the threshold, a spike is immediately generated on its output:

$$s_i = \begin{cases} 1 & if \quad v_{m,i} \geq v_{th} \\ 0 & otherwise \end{cases} \tag{8}$$

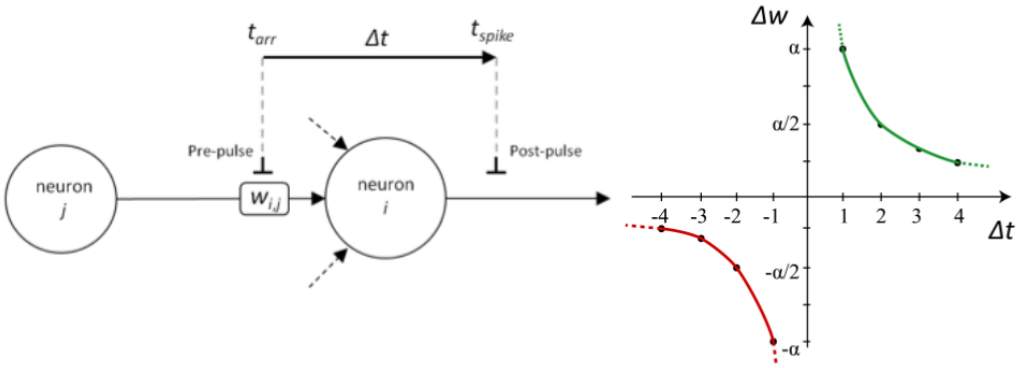

Figure 4. The STDP behaviour: On the left is represented how the $\triangle t$ is computed: it is the time that elapses between the pre-pulse and the spike generation. On the rigth is represented the variation of synaptic weight as a function of the relative timing between presynaptic spike arrival and postsynaptic firing. The right curve represents the change of the synaptic weight when a pre-pulse arrives to the neuron *i* before that the post-pulse is generated; the left curve represents the change of the synaptic weights when a post-pulse is generated from the neuron *i* before that the pre-pulse arrives.

and it is received by all the subsequently connected neurons. In case of spike generation, the membrane potential will keep to zero for an interval equal to $\triangle_{abs}$.

Each couple of connected neurons $j \rightarrow i$ is characterized by its *synaptic weight* (i.e., $w_{i,j}$). Such weight varies according to the activity of the network by means of synaptic plasticity, i.e., the ability of synapses to strengthen or weaken over time, in response to increases or decreases in their activity.

A well-known type of synaptic plasticity, the *spike timing dependent plasticity (STDP)*, is based on the precise timings of pre-pulse and post-pulse referred to a certain neuron, influencing the magnitude and direction of change of the synaptic weight. The original STDP behaviour [16] can be approximated by two exponential functions, but many variants of this algorithm are used in order to capture space-time relationships from the encoded data.In the version of Neucube that we used, the synaptic modification is given by:

$$w_i(t) = \begin{cases} w_j(t-1) \pm \frac{\alpha}{\triangle t} & t_j \neq t_i \\ w_j(t-1) & t_j = t_i \end{cases} \tag{9}$$

where α is the learning rate and $\triangle t$ is the difference between the pre-synaptic pulse arrival instant (i.e., t_{arr}) and the post-synaptic pulse generation instant (i.e., t_{tspike}):

$$\triangle t = t_{tspike} - t_{arr} \tag{10}$$

then, the weight modification can be positive or negative, depending on the arrival order (Figure 4).

4. CASE STUDY: THE WIND FARM INVESTIGATED

The large wind power plant, composed of 28 turbines and three anemometric towers, is located in the rural area of Vizzini's municipality in province of Catania Italy, that is characterised by a complex orography and an extension of 30 km^2.

The wind farm layout of Vizzini in Figure 5 shows the wind turbines marked in green and the anemometer towers marked in red.

All the turbines installed in wind power plant of Vizzini are all of the same type and precisely the Vestas $V52 - 850\,kW$ wind turbines model.

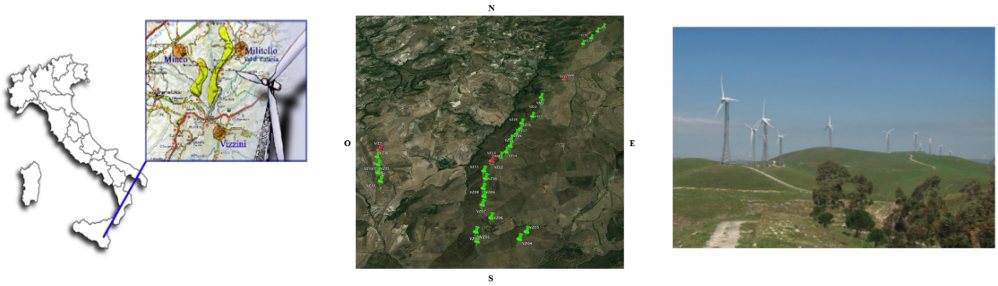

Figure 5. Location of Wind farm in Vizzini.

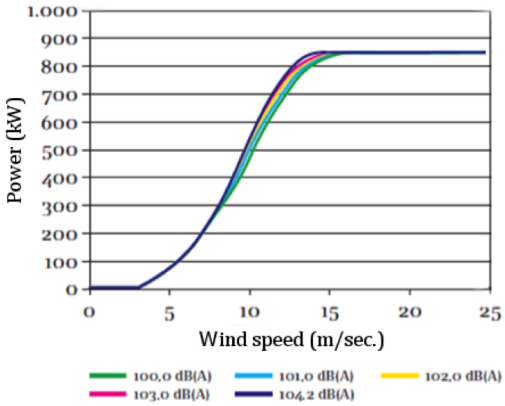

Figure 6. Wind turbine manufacturer's power curves $V52-850\,kW$ with different sound emission levels.

The Figure 6 shows the power curves at different sound levels for the $V52-850\,kW$ turbine. The main feature of this machine can be found in the data-sheets of the manufacturer and summarized in the table I.

All the turbines are equipped with OptiTip®, the special Vestas pitch regulating system. With OptiTip®, the blade angles are constantly regulated so they are always pitched at the optimal angle for current wind conditions. This optimizes both power production and noise levels.

The generator is a special asynchronous 4-pole generator with wound rotor and slip rings with rated power of 850 kW and voltage/grid frequency 690 V/50 Hz Output. At higher wind speeds, the OptiSpeed® and the pitch regulating OptiTip® system, keeps the power at nominal level regardless of the air temperature and air density. At lower wind speeds the OptiTip® system and OptiSpeed® optimises the power output by selecting the optimal RPM and pitch angle.

Each of the three anemometric towers is equipped with two four cups anemometers placed at 10 m and 30 m of height while the 28 turbines are equipped with one four cups anemometer placed at 42 m.

The data has been collected at 10 minutes intervals. In particular: the averaged active power (kW), the averaged wind speeds (m/s), the averaged wind direction (degree). The data is recorded during the period starting from January 2011 to September 2013.

In the Figure 7 is shown the probability distribution of wind speed, relative to one of the towers in the first four months of 2011 at 30 m of height and the statistical Wind Rose at the same height and in the same period.

The wind farm study has been conducted using yhe SCADA power data of each turbine. The Supervisor enables the wind farm operator to see the whole wind farm at once and immediately address any problems, therefore, minimising downtime. It allows to communicate with turbines to

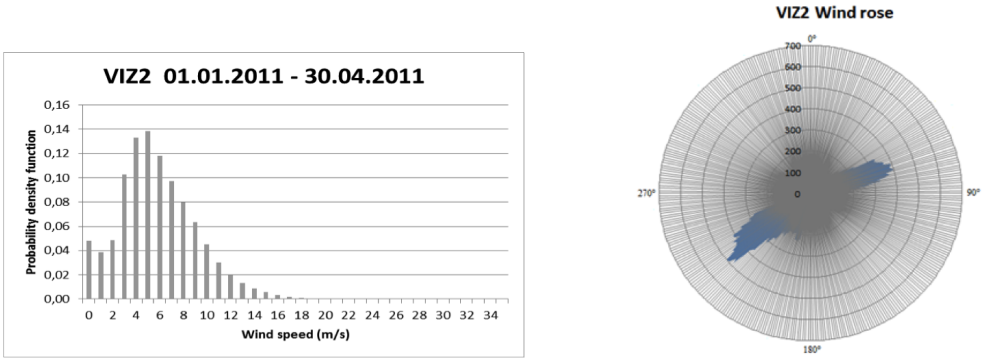

Figure 7. The probability distribution of wind speed, relative to one of the towers in the first four months of 2011 at 30 m of height (left). The statistical Wind Rose at the same height and in the same period(right).

Table I. The power characteristics of Vestas V52-850 kW wind turbine

Cut-in wind speed	4 m/s
Rate output wind speed	16 m/s
Cut-out wind speed	32 m/s

stop, start or reset the turbines, to collect vital data such as wind speed and turbine's status, and download the information to the system's fully relational database.

Wind speed is the most important parameter to be considered in the design and operations of wind turbines since its probability density distribution greatly affects the performance of wind turbines. Therefore a large number of papers have been published on modeling wind speed probability distributions with probability density functions [17, 18]. Weibull distribution model was the most commonly used probability distribution model for wind energy potential assessment. Probability density function (pdf) of Weibull distribution can be expressed as follows

$$f_v(x) = \begin{cases} \frac{k}{c}\left(\frac{x}{c}\right)^{k-1} e^{-\left(\frac{x}{c}\right)^k} & x \geq 0 \\ 0 & x < 0 \end{cases} \tag{11}$$

$$m_v = E\{v\} = \int_0^{+\infty} x \cdot f_v(x)\, dx = c \cdot \Gamma\left(1 + \frac{1}{k}\right) \tag{12}$$

$$\sigma^2 = E\{(v - m_v)^2\} = \int_0^{+\infty} (x - m_v)^2 \cdot f_v(x)\, dx = c^2 \cdot \left[\Gamma\left(1 + \frac{2}{k}\right) - \Gamma^2\left(1 + \frac{1}{k}\right)\right] \tag{13}$$

The estimation of a pdf is often a first step in many data analysis applications. Current density estimation techniques include histogramming, Parzen windows and Gaussian mixture models (GMMs) [19, 20]. Despite their simplicity and popularity, histogram-based methods suffer from the binning problem, due to the absence of a principled method to estimate the "optimal" number of bins in the marginal and joint histograms, or to relate the number of bins to a particular time series size. A smaller than optimal number of bins is known to yield an over-smoothed density estimate, whereas an excess in the same number produces an estimate that is highly sparse and prone to noise. Furthermore, histograms are not differentiable. Continuous histograms (obtained by fitting say, a spline between the values in the chosen bins) do bring in differentiability, but they do not overcome the binning problem, as the shape of the final density will vary depending on how many bins were chosen to start with. Parzen windowing [20] does not suffer from the binning problem, but it requires careful selection of the support of of the kernel, as well as the kernel function itself.

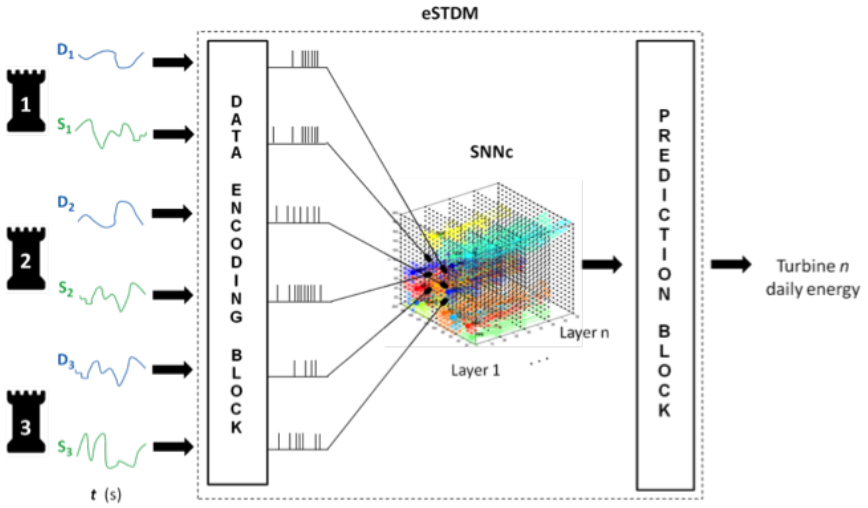

Figure 8. Modular description of the predictive structure (eSTDM). Three main modules are present: Data encoding block, SNN block, Prediction block. Input stream is composed by wind speed and direction continuously detected by each one of the three anemometer towers; output is represented by daily energy produced by the turbine T_n.

We overcome these problems in a new and simple manner. Starting with a Parzen estimate of the pdf of the wind data serie we obtain the expected value and the variance of the data and then the pdf of the Weibull distribution is calculated with a little mathematical calculation. In other terms by solving the system of the two equations (12) and (13) then we have the two numerical values of c and k and the relative Weibull distribution.

5. WIND POWER PLANT PREDICTION BY USING THE PROPOSED SNN BASED MODEL

The goal of the proposed model is the prediction of the daily energy produced (E_d) by a given wind turbine (T_n) (In this study we considered the urbine VIZ14) of the wind farm, on the basis of speed and direction data pertaining to three external anemometric towers in the previous 5 days. To achieve this goal we must solve the following two problems:

- A predictive regression problem, since we want to estimate the energy value of the next day in respect to the data given to the system.
- An inverse problem because the towers are located in different and distant positions with respect to T_n, then input training signals are not exposed to the same wind to the turbine for which the prediction is made.

The training examples are composed of time-series of the three anemometric towers VIZ2, VIZ4, VIZ7 (six 144-samples vectors, wind speeds in km/h and wind directions degrees) as input, and the daily energy produced from turbine T_n (in kWh) as target.

In order to train this neuromorphic system, for each training example first of all we have computed the target value, (i.e. *daily energy* (E_d)), from the *10-minutes-averaged active power* values (P_{10}) of T_n.

For the implementation of the predictive structure, we have used the NeuCube computational framework. The structure consists of three different blocks: data encoding block, SNN block and prediction block (see Figure 8).

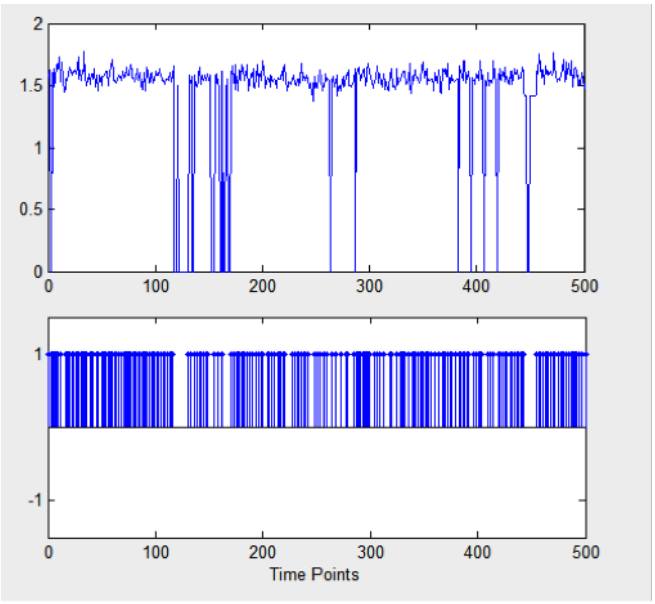

Figure 9. Encoding into trains of spike, using BSA algorithm. Top: signal representing the wind direction during one day. Bottom: corresponding spike train obtained applying BSA algorithm.

5.1. Data Encoding Block

This block is able to encode the information carried out by the tower time series in the form of spike sequences. Each one of the initial 6 time series is described as a 720-samples vector. The data vector is transformed into a spike train applying the Ben spike Algorithm (BSA) [21] with a threshold of 0.3, properly chosen. It allows to generate spike sequences composed of only positive contributes, without the need of normalization preprocessing steps (such as scaling, smoothing, etc..). An example of encoding data into trains of spikes using BSA algorithm is shown in Figure 9.

5.2. SSN block: coding of turbine distance and orography

In the second block spike trains are presented to the interconnected LIF neurons, i.e., the SNN cube (SNNc).

The SNNc is trained in an unsupervised mode, and the connection weights are modified to encode hidden spatio-temporal relationships from the input data. Polychronization effect [22] makes the SNNc able to activate the same groups of spiking neurons with a similar spike-timing pattern when a similar input is presented. Progressively, the spike timing dependent plasticity (STDP) mechanism modulates the connection weight between each couple of connected neurons, in relation to their temporal order of activation. The number of neurons used has been 1000 ($10 \times 10 \times 10$).

In order to obtain good spatio-temporal pattern recognition input stream vectors are injected on the network using one of the following two strategies [23]

1. Using spatial location where the signals are gathered.
2. Operating a proper input mapping.

We used the first strategy. In this way the six input features (i.e., wind direction and speed of the 3 turbines) have been mapped on the basis of the real positions of the turbines, taking into account the tower distance as well as the orography of the territory. We introduced a change to the classic strategy of the first point in order to represent the two features per tower (wind + direction) more specifically: a 3D model of the territory of the Vizzini area has been done and parcellated into $10 \times 10 \times 3$ voxels. The positions of the anemometric towers in the 3D space have been reported on the firsts three layers of the network as shown in the table II. In particular, the plane coordinates of

Table II. Neuron coordinates selected for representing the three anemometric towers. Such coordinates are then used for injecting the spike trains in the classifier.

Anemometric towers	Neuron row	Neuron column	Altitude	Corresponding neuron layer
VIZ2	6	6	590	3
VIZ7	6	1	584	2
VIZ4	3	8	558	1

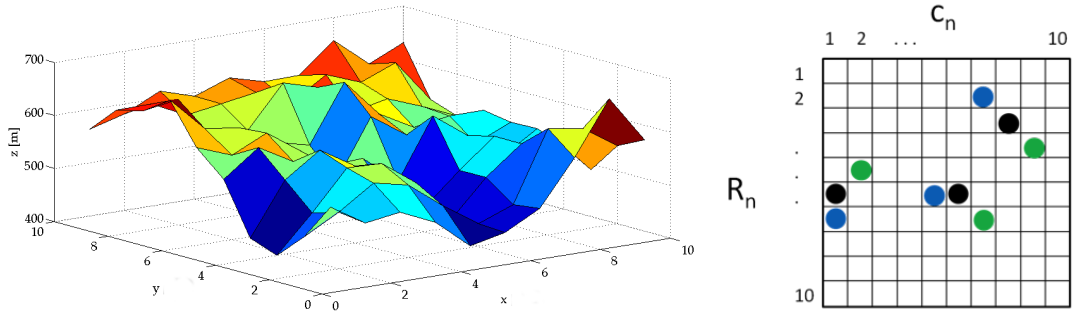

Figure 10. 3D model of the Vizzini wind farm, used for the positioning of the features used in the input coding phase (left). The features positioning in the cube is done in accord to its real position. For example in the right side of the picture is shown how are mapped the geometric coordinates z and y (R_n, C_n) of the Anemometric towers in the cube. Black circles represent the neuron associated to the tower; blue circles represent the Direction associated to the nearest tower; green circles represent the Speed associated to the nearest tower.

each tower have been chosen as row and column of each related input neuron (z and y coordinates in NeuCube, respectively) whereas the altitude range has been normalized in three intervals and chosen as layer (x coordinate in Neucube) of related input neuron (see Figure 10). To represent the altitude has been chosen the coordinate x because in NeuCube spikes travel in such direction (see Figure 11).

This strategy has been chosen in order to confer spatial significance both for direction and speed associated to the towers.

5.3. Prediction block

The second learning stage occurs in the SNN block: the spiking neurons of this block are trained to associate the spiking patterns generated by the SNNc to the target values. The Prediction block is an output neural structure trained via supervised learning method, using the dynamic evolving SNN (deSNN) algorithm, that makes use of both Rank Order Codings scheme (ROC) and spike driven synaptic plasticity (SDSP) learning rules [4]. By using the ROC we set the initial values of weights and SDSP rule adjusting them based on further incoming spikes. The proposed method implies that this block operates as a regressor: the same data used for the training of SNNc is used also to train the prediction block in order to associate input data to target data.

6. EXPERIMENTAL DATA: THE DATASET COMPOSITION

The data available are organized in 365 files, each one containing 144 average values collected at intervals of 10 minutes and also pertaining to one whole day.

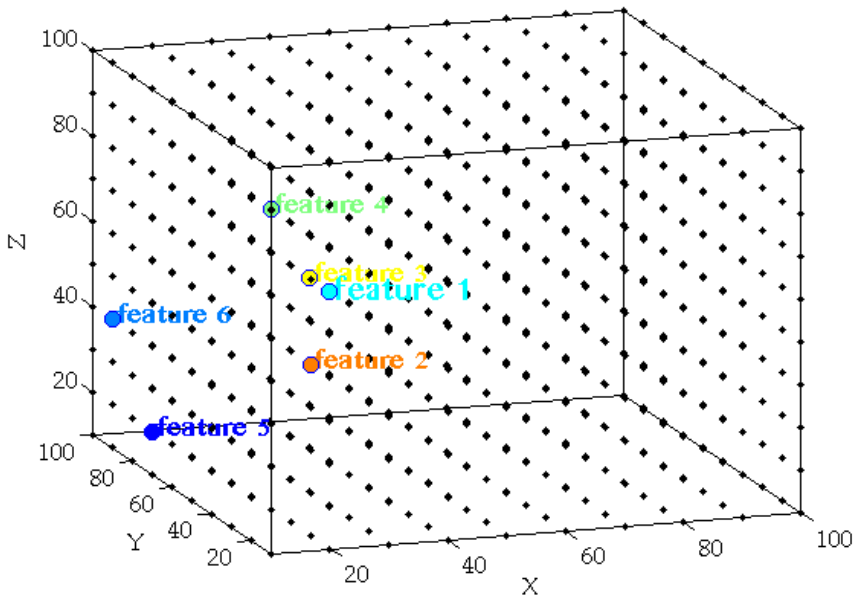

Figure 11. final 3d mapping in the cube considering also the altitude. The information is transmitted from left to right

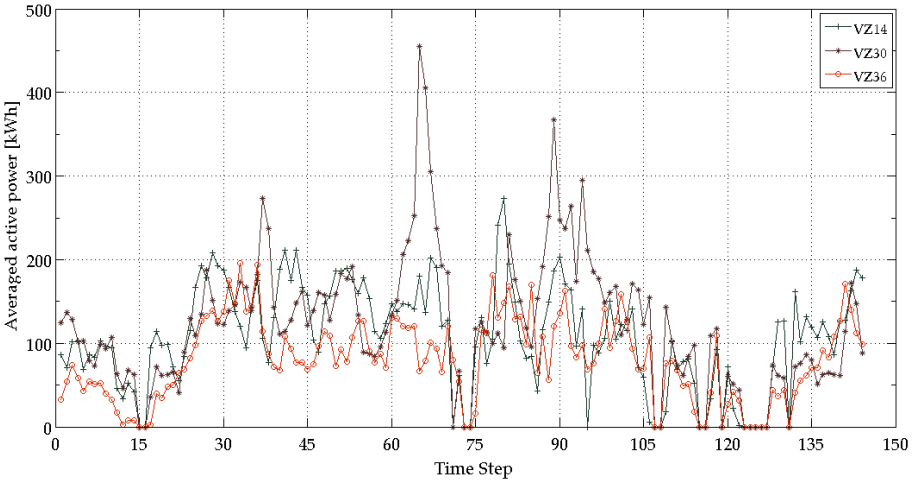

Figure 12. example of P10 sequence of wind turbine VZ14, VZ30,VZ36 during the same day

For the each wind turbine, NaN, zero and negative power values can represent malfunctioning, manteinance or no wind. In particular:

- NaN values means that turbine is under manteinance, or a fault has occurred.
- Zero value means that the average wind speed has not reached the minimum threshold (scarce wind), or the maximum threshold has been reached (protection).

For the anemometer data, NaN can represent malfunctioning or maintenance. Since dataset merges both anemometer and turbine data, different cases are possible:

1. If turbine power is 0 and anemometer speed $<$ Cut-in $\Rightarrow$ Working state.

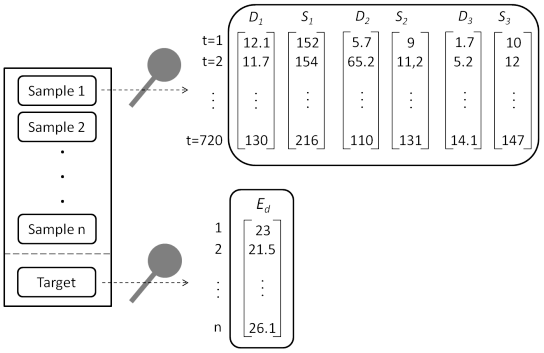

Figure 13. The dataset composition is illustrated on the left. An in-depth view of the structure of Samples and Target is given on the right. Each classifier is trained with a different target vector, each one corresponding to a different turbine considered.

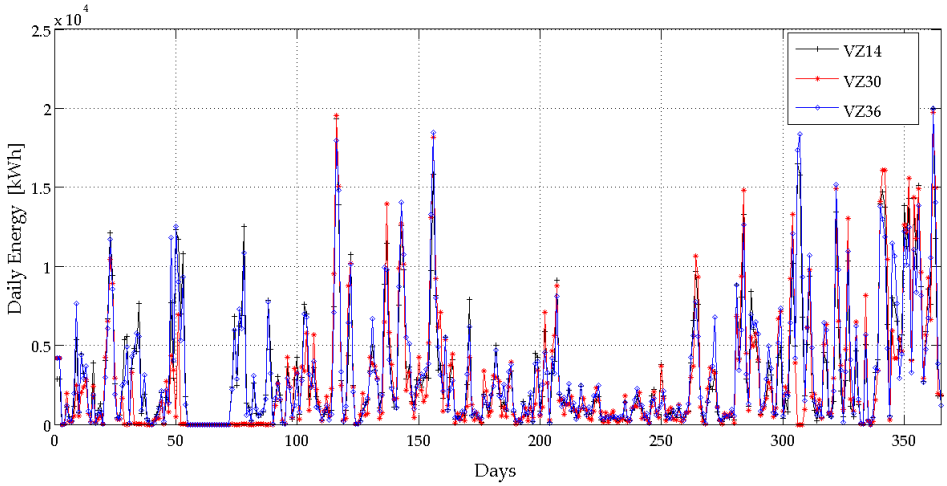

Figure 14. daily energy behaviour during a whole year.

2. If turbine power is 0 and anemometer speed $>$ Cut-out $\Rightarrow$ Protection state.

Remaining combinations have been marked as malfunctioning or manutention.

For this reason a check is done before to collect them in the dataset. Some exclusion criterias have been adopted both for training and test:

- For the training phase daily sets with more than 30% of zero values ($n \approx 44$) have been rejected because statistically unmeaningful or misleading for the predictor;
- For the test phase the predictors have been tested on daily sets that not include malfunctioning or manutention.

The final dataset (see Figure 13) is composed of 6 time, series wind direction and speed of the 3 anemometric towers in 5 consecutive days (days $n-5$ to $n-1$) as input, and the energy produced from a given turbine in the subsequent day (day n). This has been achieved by shifting the data prepared, and collecting blocks of 5 days that not present discontinuity due to rejection.

In order to train the SNN-based system with a statistically significant pattern set, 100 time series, pertaining to 100 different days have been considered for model learning and validation. For testing the SNN based system, other 50 days, different from those used for training, have been considered.

Speed and direction of three different anemometric towers have been considered for the energy prediction. Once the locations of the inputs have been set in the SNN block, the information to the network will be brought from the timings of the encoded input sequences. This is equivalent to say

that the overall system is trained to associate the trajectory of a point in the 6-dimensions feature space (wind direction and speed of the 3 anemometric towers) to the total value of energy produced from a given turbine during the subsequent day.

7. TEST RESULTS

Each training data structure (sample) reflect the activity of 5 days. This structure is constituted by a 720 x 6 matrix, composed of the wind direction and wind speed time series of three anemometric towers in one day, sampled at intervals of 10 minutes. The target is represented by the daily energy values produced by turbine VIZ14 related to the subsequent day.

For the SNN-based ssystem, the SNNc is composed of 1000 neurons ($X = 10, Y = 10, Z = 10$), connected in a "small world" topology. As illustrated in subsection 5.2, the 6 input neurons have been chosen according to the real tower positions, since the temporal data present spatial attributes (see discussion in [4]).

In order to develop the optimum network model, many networks have been trained, but we found similar optimum parameters for the classifier. The optimal values of the parameters for the SNNc are:

- Small world radius = 2.5
- Threshold of firing = 0.5
- Potential leak rate = 0.002
- Refractory time = 6

While the parameters used for the prediction block are:

- Mod = 0.7
- Drift = 0.009
- k=3
- σ=1

The optimisation of the parameters of NeuCube has been achieved by trial and error tests on the SNN models, using an extended grid-search approach [24].

The prediction errors have been calculated by means of the Mean Absolute Deviation (MAD), the Mean Absolute Percent Error (MAPE), and the Mean-Root-Squared Error (MRSE) defined as follows:

$$MAD = \frac{\sum_{i=1}^{n} \mid y_i - \hat{y}_i \mid}{N} \tag{14}$$

$$MAPE = \sum_{i=1}^{n} \frac{\mid y_i - \hat{y}_i \mid}{y_i} \cdot \frac{100\%}{N} \tag{15}$$

$$MRSE = \frac{\sqrt{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}}{N} \tag{16}$$

The obtained values of *MAD*, *MAPE*, *MAPE* are, respectively:

- $MAD = 3.39$
- $MAPE = 19.55$
- $MRSE = 0.9$

In Figure 15 is reported a data sample of the true value of daily energy generated of the turbine VZ14 and the predicted one for the same turbine.

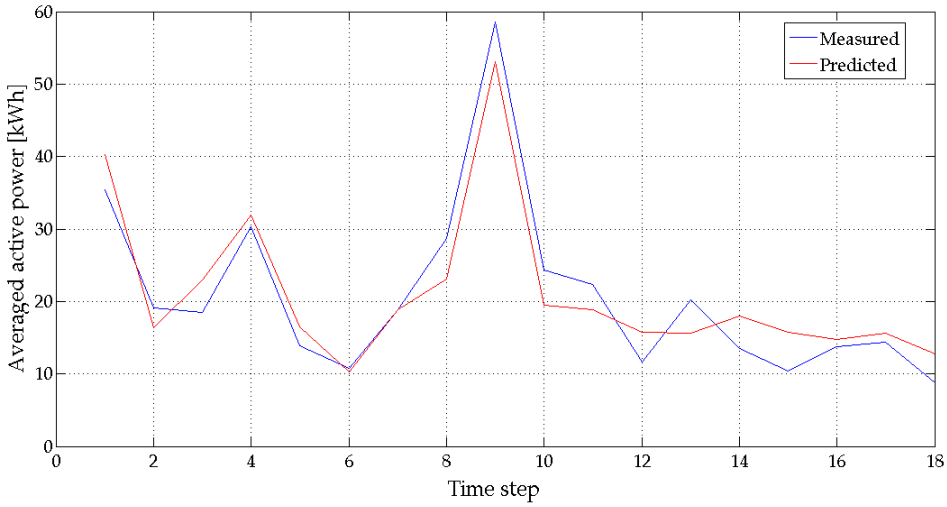

Figure 15. Comparison between the true value of daily energy generated of the turbine VZ14 and the predicted one for the same turbine.

8. CONCLUSIONS

Wind power output unpredictability of the wind speed, unexpected variations of wind farm output may increase operating costs of the electricity grid as well as set potential threats to the reliability of electricity supply. Without the ability to accurately predict wind energy generation, it is very unlikely that wind energy will become an important contributor to the total energy market [25, 26, 27].

In this paper a spiking neural network-based model for the prediction of wind farm energy production is proposed. This model performs a prediction of the daily energy produced by one wind turbine of the wind farm, by using the wind speed and direction data coming from the three anemometric towers, during all the day.

The tests carried out with experimental data show very low values of MAD, MAPE and MRSE. It demonstrates this model could accurately predict the daily energy produced by each wind turbine of the wind farm. Furthermore taking into account that the data supplied to the network (both during training and testing) also contain data that represent malfunctioning, manteinance, turbines mapping as well as the orography of the territory, the model can be successfully applied for predictions of wind energy generation in real wind farm also in presence of faults.

REFERENCES

1. Brusca S, Lanzafame R, and Messina M. Wind Turbine Placement Optimization by means of the Monte Carlo Simulation Method. *Modelling and Simulation in Engineering.* 2014; Article ID 760934: 8 pages.
2. Žliobaitė I, Bifet A, Read J, Pfahringer B, Holmes G., Evaluation methods and decision theory for classification of streaming data with temporal dependence. *Machine Learning.* 2015; 98(3):455-482.
3. Gaber M M, Zaslavsky A, Krishnaswamy S. A Survey of Classification Methods in Data Streams. *Data Streams: Models and Algorithms.* Springer US 2007; 31:39-59.
4. Kasabov N, Dhoble K, Nuntalid N, Indiveri G. Dynamic evolving spiking neural networks for on-line spatio and spectro-temporal pattern recognition. *Neural Networks.* 2013; 41:188-201.
5. Kasabov N. NeuCube: A spiking neural network architecture for mapping, learning and understanding of spatio-temporal brain data. *Neural Networks.* 2014; 52:62-76.
6. Kasabov N et Al. Evolving spatio-temporal data machines based on the NeuCube neuromorphic framework: Design methodology and selected applications. *Neural Networks.* 2016;78:1-14.
7. Vermeer L J, Sørensen J N, Crespo A. Wind turbine wake aerodynamics. *Progress in Aerospace Sciences.* 2003;39(6-7):467-510.
8. Barthelmie R J, Hansen K, Frandsen S T, Rathmann O, Schepers J G, Schlez W, Phillips J, Rados K, Zervos A, Politis E S and Chaviaropoulos P K. Modelling and measuring flow and wind turbine wakes in large wind farms offshore. *Wind Energy.* 2009;12(5):431-444.

9. Porté-Agel F, Wu Y T, Chen C H. A Numerical Study of the Effects of Wind Direction on Turbine Wakes and Power Losses in a Large Wind Farm. *Energies* 2013;6(10):5297-5313.
10. Jensen N O. A Note of Wind Generator Interaction. *RisøNational Laboratory.* 1993, Roskilde, Denmark.
11. rady S A, Hussaini M Y and Abdullah M M. Placement of wind turbines using genetic algorithms. *Renewable Energy.* 2005;30(2):259270.
12. Lanzafame R and Messina M., Fluid dynamics wind turbine design: critical analysis, optimization and application of BEM theory. *Renewable Energy.* 2007;32(14):2291-2305.
13. Maass W. Networks of spiking neurons: The third generation of neural network models. *Neural Networks.* 1997;10(9):1659-1671.
14. Izhikevich E M. Which model to use for cortical spiking neurons. *IEEE Transactions on Neural Networks.* 2004;15(5):1063-1070.
15. Susi G, Cristini A and Salerno M. Path multimodality in Feedforward SNN module, using LIF with latency model. *Neural Network World.* 2016;26(4):363-376.
16. Bi G Q and Poo M M. Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type. *J. Neurosci..* 1998;18:10464-10472.
17. Akdağ S A and Dinler A. A New Method to Estimate Weibull Parameters for Wind Energy Applications. *Energy Conversion and Management.* 2009;50(7):1761-1766.
18. Bonanno F, Capizzi G, Lo Sciuto G and Napoli C. Wavelet recurrent neural network with semi-parametric input data preprocessing for micro-wind power forecasting in integrated generation Systems. *2015 International Conference on Clean Electrical Power (ICCEP), Taormina.* 2015;602-609.
19. Bishop C M. Pattern Recognition and Machine Learning. *Springer.* 2006.
20. Parzen E. On the Estimation of a Probability Density Function and the Mode. *Annals of Math. Stats..* 1962;33(3):1065-1076.
21. Schrauwen B and Van Campenhout J.BSA, a fast and accurate spike train encoding scheme. *Proceedings of the International Joint Conference on Neural Networks.* 2003;4:2825-2830.
22. Izhikevich E M. Polychronization: Computation with Spikes. *Neural Comput..* 2006;18(2):245-282.
23. Tu E, Kasabov N, Jie Yang J. Mapping Temporal Variables Into the NeuCube for Improved Pattern Recognition, Predictive Modeling, and Understanding of Stream Data. *IEEE Transactions on Neural Networks and Learning Systems.* 2016;pp(99):1-13.
24. Sinha D B, Ledbetter N M and Barbour D L. Spike-timing computation properties of a feed-forward neural network model. *Front. Comput. Neurosci..* 2014;8:5:1-17.
25. Lo Sciuto G, Susi G, Cammarata G and Capizzi G. A spiking neural network-based model for anaerobic digestion process. *2016 International Symposium on Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM), Anacapri.* 2016;996-1003.
26. Bonanno F, Capizzi G, Lo Sciuto G, Gotleyb D, Linde S and Shikler R. Extraction parameters and optimization in organic solar cell by solving transcendental equations in circuital models combined with a neuroprocessing-based procedure. *2016 International Symposium on Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM), Anacapri.* 2016;872-877.
27. Lo Sciuto G, Cammarata G, Capizzi G, Coco S and Petrone G. Design optimization of solar chimney power plant by finite elements based numerical model and cascade neural networks. *2016 International Symposium on Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM), Anacapri.* 2016;1016-1022.