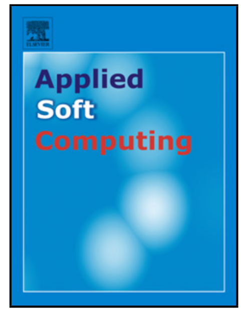



Title: An advanced neural network based solution to enforce dispatch continuity in smart grids

Author: Giacomo Capizzi Grazia Lo Sciuto Christian Napoli Emiliano Tramontana

PII: S1568-4946(17)30540-9
DOI: http://dx.doi.org/doi:10.1016/j.asoc.2017.08.057
Reference: ASOC 4448

To appear in: *Applied Soft Computing*

Received date: 10-2-2017
Revised date: 12-8-2017
Accepted date: 30-8-2017

# An advanced neural network based solution to enforce dispatch continuity in smart grids

Giacomo Capizzi[1], Grazia Lo Sciuto[1], Christian Napoli[2], Emiliano Tramontana[2]

[1]*Dipartimento di Ingegneria Elettrica Elettronica e Informatica, University of Catania, Italy*
[2]*Dipartimento di Matematica e Informatica, University of Catania, Italy*

**Abstract**

In energy generation systems including a photovoltaic park, fluctuations are the norm: both production and demand levels can vary on hourly basis. Hence, energy management and dispatching systems have to cope with the possibility of inadequate production while satisfying as much as possible user demands. We put forward a management solution that models the behaviour of each production plant and consumption device, and determines energy allocation. For this, gathered data are wavelet transformed to let us retain only the useful characteristics of data on both large and small scales of the signal. Models are handled by several neural networks which perform predictions in advance of 48-hour, with a granularity of half an hour. Moreover, according to realtime user demands, the management solution determines energy flows between production plants and consumption devices. Therefore, while in some cases it might be necessary to postpone the activation of some consumption devices, in others we can take advantage of a production surplus. Thanks to the proposed solution proper actuators can be programmed beforehand to improve the fairness to users, and use peaks of energy production, thus reducing green energy shortage, and extra costs.

*Keywords:* Integrated Generation System, Photovoltaic, Renewable energy, Cloud computing, Wavelet analysis, Neural networks, Parallel processing.

## 1. Introduction

Renewable energy plants, including photovoltaic (PV), wind, and bio-diesel generator, are widely used for energy production. However, such energy sources cannot provide a fixed amount of energy continuously, due to their seasonal and intermittent nature [1]. A combination of energy sources can then be more effective, if properly managed, in order to reduce energy production fluctuations.

Intelligent power management systems are paramount when dealing with renewable energy sources [2]. Basically, the management system of a plant has to include the monitoring of production and demand levels and record data with granularity, e.g. on hourly basis.

Due to the intermittent nature of the solar- and wind-derived energy, an open issue is how to minimise the effects of fluctuating green power productions, and how to satisfy the requests of consumption devices over time [3, 4, 5]. A fundamental support allowing proper allocation is reliable load and production forecasting.

Previously proposed management solutions can be examined according to whether they consider the following features: (i) energy fluctuations and adaptation, (ii) data segregation, (iii) planning of energy allocation, and (iv) distributed generation.

Firstly, several approaches build a neural network model by feeding it with *unfiltered* data gathered from PV plants and consumption devices, and resulting in a model that depicts an average behaviour, while missing fine grained details, and for a large time-frame, which results in considerable errors for more fine intervals [6, 7, 8, 9, 10, 11, 12].

Some approaches adopt data *filtering*, however the error is generally high [13, 14]. In [13], the custom network topology used counterbalance in a negative way data filtering, hence missing most of the dynamic of the signal. In [14] the neural network topology used (having no feedback and no delayed lines) is not able to handle time changing finer phenomena with enough accuracy, hence their resulting forecast suffers of a very high error.

Moreover, many existing approaches use a model that fits stationary phenomena, as autoregressive-moving average [15], or adopt mathematical tools, such as Fourier transform still capturing the stationary phenomena only [16]. Some neural network topologies have been used which are adeguate only for a stationary scenario [17, 18, 19]. on

Secondly, several approaches use aggregated data coming from power plants or consumption devices, then the corresponding model provides an average trend, while cannot perform accurate forecast for each plant, device, or place [6, 8, 20].

Thirdly, many approaches simply react to the fluctuations of energy production and demand, without planning in advance [21, 22]. Often their main goal has been an economical balance [21, 23], and in some cases an energy balance has been considered [22, 24].

Fourthly, all the above approaches are unaware of distributed generation and cannot gain advantage of the plant position with respect to the demand. In [25, 26] distributed generation has been considered and for this a certain size level is set, while beyond that level power losses increase.

Our solution overcomes all the said problems, by using gathered data to plan energy dispatching in advance, hence avoiding sudden fluctuations of energy flows. Our aim to reduce energy fluctuations towards each consumer is novel, since other approaches simply consider an overall energy balance. Moreover, the novel contribution of this paper comprises the following two strategies. The proposed combination of wavelet transforms and the several neural networks adopted, for segregated data referring to different plants and devices, manage to accurately estimate power production and consumption for each plant and consumption device, in advance of 48-hour. Differently from other approaches, wavelet transforms let us retain significant details at several scales for the observed behaviour and gathered data; whereas the adopted neural network topology can embed new data and adapt to the changes observed over time.

Moreover, unlike other approaches, we keep data coming from different observations separate. For this we need a considerable amount of neural networks and processing capability. We have tackled such issues by tapping into the cloud for proper storage and processing of each data flow. We propose a computationally capable solution, based on Graphic-Processing-Units (GPUs), to compute in advance and for timesteps of half an hour the suitable energy dispatch between PV plants and load buildings in such a way to satisfy demands as much as possible. The latter is a computationally intensive task when considering a real-world scenario with hundreds of devices.

Thanks to the proposed solution taking advantage of the statistical availability of production and demand data, we reduce as much as possible the energy bought from a tradition provider. The proposed solution ensures continuous availability of green power for selected consumption devices, while turning down requests that cannot be honoured according to previous statistics. Therefore, this strategy avoids that consumption devices witness sudden energy fluctuations. In order to satisfy energy demand when having a limited green energy production, we consider a possible additional energy flow from the commercial energy provider, only when needed, or postponing the activation of some consumption devices. Moreover, we compute how to satisfy over time the demands of the commercial energy provider, with whom some agreements have been taken beforehand, and any variation of energy sold would amount to an economic loss. For the proposed advanced energy management system, the needed computation is performed on cloud computing hosts equipped with GPUs. Since the workload for the data analysis can vary according to the characteristics of the production plant and the constrains coming from load buildings, the proposed solution includes the ability to allocate and release cloud computing resources over time.

The plant investigated in this paper, as a proof of concept, consists of PV arrays and load buildings in the campus of the University of Catania called 'Cittadella'.

This paper is structured as follows. Section 2 discusses the related literature.

Section 3 introduces the distributed infrastructure catering for the needs related to data collection, analysis, and government decisions.

Section 4 describes the proposed solution for predicting power production and load.

Section 5 details the fast computing support that examines the flows between energy sources and sinks. Section 6 reports the results on our proof of concept.

Finally, the last section draws our conclusions and summarises future work.

## 2. Related Works

Our proposed approach deals with two main issues: (i) modelling data coming from energy production and consumption, and estimating future behaviour, (ii) planning how green energy should be used to avoid sudden energy fluctuations. According to the methods of data processing, the proposed models can be roughly summarised into three categories: statistical, neural network-based, and hybrid. Statistical models attempt to find trends and are based on mathematical models to perform forecasting. Generally, forecasting errors are low when the input variables are under normal conditions. Among all the statistical models, the auto-regressive moving average

is a typical method that is widely and commonly applied in time series forecasting [15].

Unlike statistical models, neural networks are data-driven and fault-tolerant, and those robust characteristics make them very suitable for PV energy forecasting. Neural networks have a high ability to address complicated relationships, find prediction patterns and perform forecasts under uncertainty [6, 8, 18, 20]. While several previous approaches attempting forecasting using neural networks have a high error, and merge data coming from different plants [6, 7, 8, 9, 10, 11, 12], our approach affords a lower error for each separated plant.

A hybrid model combines a mathematical model and a neural network. Our approach falls into this category thanks to the use of wavelet transforms on gathered raw data, and then neural networks on such transformed data.

In some previous approaches, the recorded data of solar produced energy are decomposed into several components of various timefrequency domains according to wavelet analysis [13, 14, 27]. In our approach we manage to have a lower forecasting error, due to a more accurate use of wavelet decomposition, recurrent neural network topology, and data segregation of records from different plants.

The literature presents only a few approaches that deal with intelligent decision-making to manage a power plant and the energy distribution [21].

While economical or energy balance have been proposed by previous approaches [24, 23], they proposed to balance energy offer and demand for a given scenario, hence their solution remains valid only till no new conditions emerge.

We use our predictions for energy produced and for consumption requests to find how to match the two over time for future intervals. Hence, adjusting energy flow without letting the consumers know about the fluctuating green power generation.

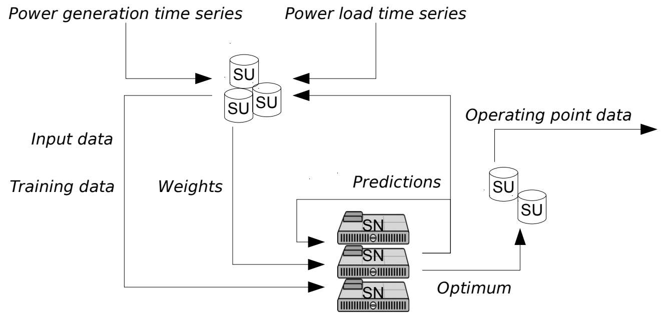



Figure 1: Main cloud resources and data flows among them.

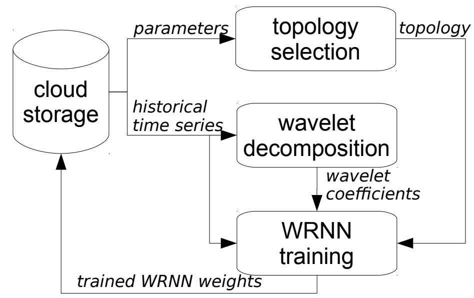



Figure 2: Data flow used during the training of a neural network.

Moreover, no solution has been proposed to make best use of green energy when only partial fulfilment among loading devices can be achieved.

Unlike allo other approaches, we compute every possible combination of match between offer and demand over time thanks to GPU resources over a cloud. Additionally, differently from all other approaches that aggregate collected usage and production data, we store the characterised profile of each consumption device and production plant, on a cloud.

## 3. Distributed Computing for Smart Grid

The computational load deriving from the execution of predicting components, which consist of wavelet transforms and neural networks (see Section 4), and for determining energy flows over time (see Section 5), is handled by a cloud-based infrastructure providing storage and computing resources.

Therefore, raw data gathered from power production plants and load buildings are stored on the cloud and then used to: (i) perform wavelet transforms, this allows a proper data characterisation, (ii) train neural networks, (iii) perform predictions, and (iv) compute the dispatching strategy, hence selecting which devices can consume green energy and when.

Each power production plant or consumption building (or loading device) is associated with a corresponding behavioural model, consisting of a neural network dubbed WRNN, that executes on a cloud resource. Since the characteristics of power production and load are different, several modelling solutions are available, hence the needed WRNN topology and training has to be selected, among available ones that have been previously designed, according to the qualifying parameters of the plant (see *similarity* in the following).

Fig. 1 shows the proposed distributed architecture consisting of the following main resources.

- Data storage units (SU) holding raw data collected from the observations of signals from production plants and from demands of consumption devices.

- Skimming nodes (SN), each building and executing a WRNN taking as input historical time series; then, analysing demands from loading devices and green energy availability, to find possible energy flows.

Virtual machines (VMs) on a cloud are requested by a *resource queue manager* (RQM), which then starts an available VM image to perform the tasks of a SN or a SU (such roles can be added or removed to reflect the dynamic of the real environment). RQM is responsible to check the amount of VMs used and trigger the growth (or decrease) as necessary, hence installing on each VM the required software packages. Once a SN is started, it is connected to a raw data flux reflecting the run time observed significant values for a power plant, or a consumption device. Each SN holds a dedicated WRNN corresponding to the plant to be modelled, hence has to be trained separately.

While PV plants are seldom added, hence the number of needed SNs rarely changes, consumption devices can vary, e.g. when adding/updating air conditioning devices, appliances, server computers, or when significantly varying their usage, e.g. by increasing the number of employees. When consumption changes, the training phase has to be triggered on the new ensemble, or a device has to be represented by a corresponding further SN.

For loading devices, several scenarios and settings can be handled by our proposed solution, according to the installation of power meters, and then the availability of consumption data. I.e., a building could be considered as a whole, when only a power meter has been installed for it, or the granularity could be the floor or a room, if power meters have been installed with high granularity. Without loss of generality, the following considers that power meters have been scattered on a building to gauge different loading subsystems in their own. Analogously, the appropriate number of power outlets have been modified to be controlled by a centralised decision maker for the loading devices to be activated/deactivated.

Fig. 2 shows the main steps and data flow needed for training a modelling WRNN, i.e. a newly available SN is given a replica of the most similar stored WRNN, which has been previously used, and starts training the WRNN by feeding it with the historical time series for the plant. In order to choose a previously stored WRNN, *similarity* is computed according to several parameters, including e.g.: (i) the objective (power production or consumption), (ii) the size of the plant or building, (iii) the location, (iv) the type of device. For the sake of performing WRNN replicas, after training the internal characteristics (topology and weights) of a WRNN are stored, hence such data are transferred

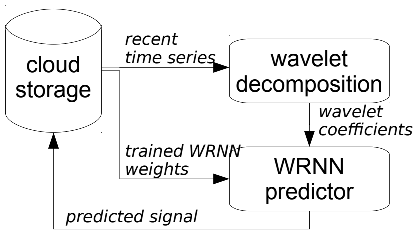



Figure 3: Data flow to WRNN predictors during the real time phase.

to a SU and appropriately tagged with the said parameters.

Fig. 3 shows the data flow and the steps executed to perform predictions using the trained WRNN, i.e. once the WRNN has been validated, then it is possible for it to receive requests of predictions, as well as to continue embedding newly gathered data.

Besides the said WRNN data modeling and prediction, a SN can be used to perform the analysis of different energy dispatch configurations. Such a task is organised as a massively parallel process on a GPU that compares dispatch configurations (see Section 5). Once a SN has computed the viable configuration, resulting data will be sent to a SU and to the actuators (programmable devices or power outlets), hence realising the given dispatching strategy.

Once deployed, the said distributed software architecture continuously receives and analyses data and determines operating conditions for the connected consumption devices, without needing any human intervention, hence it suits for electrical dispatch networks.

## 4. Predicting renewable energy and power load

Several topologies for a WRNN model can predict the generated power of a PV plant, or the power load offered by a building, when properly trained [27, 28]. Therefore, our solution is based on a WRNN model. We will now give the essential concepts underpinning the WRNN model, while referring to [28] for further details; whereas the way predictors are used for the problem at hand is in Section 4.2.

### *4.1. Models as WRNNs*

The main advantage of the WRNN model is the ability to predict the future trends of a data time series while also performing a wavelet inverse transform from wavelet coefficients. This feature enables us to feed the WRNN with data consisting of wavelet coefficients and take as output data in the time domain. Of course, such a procedure requires us to first transform a time series into the wavelet domain.

Wavelet decomposition is employed for physical and dynamic phenomena to reduce data redundancies, therefore giving as a result a compact representation expressing the intrinsic structure of a phenomenon.

When deriving wavelet coefficients from raw data, the main advantage gained is the ability to pack the energy signature and express it in a few relevant non-zero coefficients [29, 30].

The wavelet-based representation has a much lower noise incidence and therefore has the ability to properly model a time series perturbed by several factors, e.g. the weather variability that affects power generation, or the human behaviour that modifies a building power load.

The used WRNNs take as input the wavelet transform of gathered data, and give as output predictions of signals in the time domain. Therefore, WRNNs embed a wavelet inverse transform as in a recursive lifting procedure [28, 29]. This WRNN behaviour is attained by using a composition of Radial Basis Functions (RBFs) as a transfer function that closely approximates a mother wavelet, in fact a RBF is a good enough transfer function while it partially approximates half of a mother wavelet. It is indeed possible to properly scale and shift a couple of RBFs to obtain a mother wavelet. Therefore, the proposed WRNNs embed two hidden layers with RBF transfer functions. The adopted transfer function composition approximates a mother wavelet, whereas it is inappropriate to use a mother wavelet as a transfer function since it lacks needed elementary properties, e.g. the absence of local minima and a sufficient graded and scaled response. Therefore, the WRNN uses RBFs as an approximation for the mother wavelet without harming the response expected by the network [31].

### *4.2. Proposed WRNN predictors*

In our proposed solution, each power plant is associated to its own WRNN, since the observed phenomena are affected by specific factors such as size of the plant, local weather in the area, orientation of panels. Moreover, since each consumption device, or set of devices, e.g. air conditioning, has its own profile of energy consumption, it has an associated WRNN. Then, a WRNN set forecasts the green-generated power and another set of WRNNs estimates the power load.

The proposed WRNN is a recurrent neural network (RNN) taking as input wavelet transformed data. Now, for a RNN, the transient response is a critical issue when realtime signal processing is desired. Conventional RNN training algorithms, such as backpropagation through time and realtime recurrent learning (RTRL) exhibit low convergence speed. In this work, we have achieved both speed and stability, thanks to an improved version of RTRL, dubbed robust adaptive gradient-descent training algorithm [32]. Therefore, the training pattern consists of both realtime online backpropagation and RTRL, according to the derived convergence and stability conditions. The measured generated power coming from a power plant, or the power load measured for each consumption device are generalised as time series $u(\tau)$, where $\tau$ is the discrete time step of the sampled data, which in our experiments has been set to half an hour. A biorthogonal wavelet decomposition of the time series is then computed to obtain the correct input set for the WRNN, as required by the devised architecture.

This decomposition is obtained by applying the wavelet transform: in such a way that the $i$-esime recursion $\hat{W}_i$ produces, for any time step of the series, a set of coefficients $d_i$ and

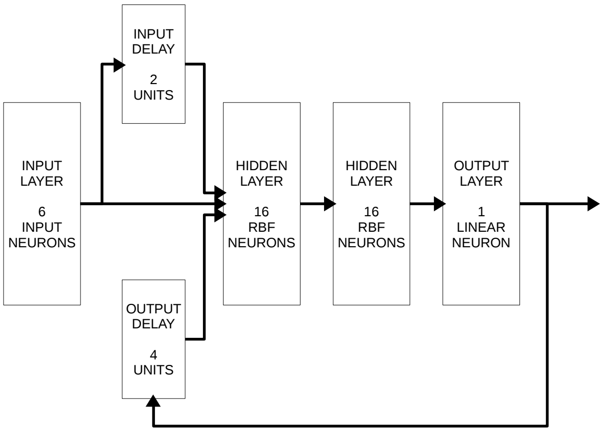



Figure 4: The devised architecture for WRNN predictors.

residuals $a_i$, and so that

$$\hat{W}_i[a_{i-1}(\tau)] = [d_i(\tau), a_i(\tau)] \quad \forall\, i \in [1, M] \cap \mathbb{N} \tag{1}$$

where we intend $a_0(\tau) = u(\tau)$. The input set can then be represented as an $N \times (M + 1)$ matrix of $N$ time steps of a $M$ level wavelet decomposition where the $\tau$-esime row represents the $\tau$-esime time step as the decomposition

$$\mathbf{u}(\tau) = [d_1(\tau), d_2(\tau), \ldots, d_M(\tau), a_M(\tau)] \tag{2}$$

Each row of this dataset is given as input value to the $M$ input neurons of a WRNN. Fig. 4 shows the architecture of the proposed WRNN consisting of the several layers: the input layer having 6 neurons, two hidden layers having 16 neurons each, delay stages, and an output layer having 1 neuron. For each time step $\tau_n$ then the WRNN predictor gives the estimated future value at a time step $\tau_{n+r}$. The WRNN can be considered a functional where $r$ is the number of time steps in the future.

$$\hat{N}[\mathbf{u}(\tau_n)] = x(\tau_{n+r}) \tag{3}$$

Fig. 5 shows the decomposition that we have adopted in our proof of concept, which is identified by the numbers 3.9.

## 5. Energy Dispatch Management

As stated in the previous sections, our cloud-based analyses estimate the power production and consumption at several plants and buildings. The proposed intelligent management solution aims at avoiding fluctuating power flows towards consumption devices. A relevant result of such a management is that the green generated power can be sold to the traditional power provider, e.g. when the request is low from consumption devices, or when it is more convenient economically. However, a major concern in dealing with commercial providers is to maintain a stable energy production for a long period of time or, at least, to provide a precise prediction of the energy sold. Our system copes with both the problems of power fluctuations and predictions, by selecting an appropriate destination for the dispatching of the green energy, and by using forecasting techniques.

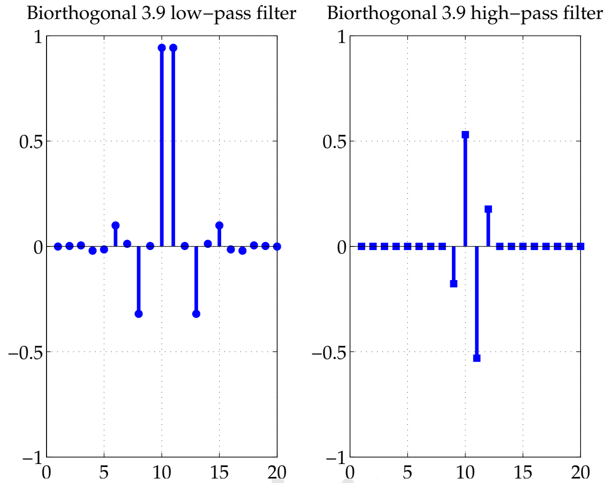



Figure 5: Adopted decomposition using biorthogonal 3.9 wavelet filters, left low-pass and right high-pass filter.

### 5.1. Computing dispatching solutions

The proposed solution aims at avoiding sudden changes in the energy dispatched, and even sold to the traditional power provider, by maintaining a stable power value with smooth variations during time. For this we consider a certain number of consumption devices and a generation plant. Consumption is the use of energy by any load device or set of devices, e.g. an entire building, the air conditioning subsystem, or other subsystems. Let consumption devices be enumerated with an index $k \in [1, N] \cap \mathbb{N}$, and then add a special consumer ($k = 0$) representing an energy provider to whom we want to sell a portion of the generated power.

Each consumer $k$ will be characterised by a power load $P_L^k(\tau)$ for the discrete time step $\tau$. Therefore, the power balance must be kept as

$$P_L^k(\tau) = P_D^k(\tau) + P_E^k(\tau) \tag{4}$$

where $P_D^k(\tau)$ represents the power dispatched to the consumer by the generation plant, and $P_E^k(\tau)$ the power coming from the grid of the energy provider. It is possible to negotiate with each consumer a tolerance $\delta^k(\tau)$ for each time step so that

$$P_L^k(\tau) - \delta^k(\tau) \leq P_D^k(\tau) \leq P_L^k(\tau) \tag{5}$$

and granted that

$$P_G(\tau) = \sum_{k=0}^{N} P_D^k(\tau) \tag{6}$$

where $P_G(\tau)$ is the total generated power. On a real scenario, we have to take into account a constraint regarding the fairness of the energy distribution among consumers, therefore if a certain total energy distribution share ratio $\rho_k$ is granted by contract, then we must impose that

$$\int_{\Delta t} P_D^k(t)dt = \rho_k \int_{\Delta t} P_G(t)dt \quad \forall\, k \in [0, N] \cap \mathbb{N}_0 \tag{7}$$

where $\Delta t$ is an amount of time specified by a contract and considering time $t$ as a continuous variable in order to give a mathematical meaning to the integrals.

Figure 6: An overview the PV plant at the campus.

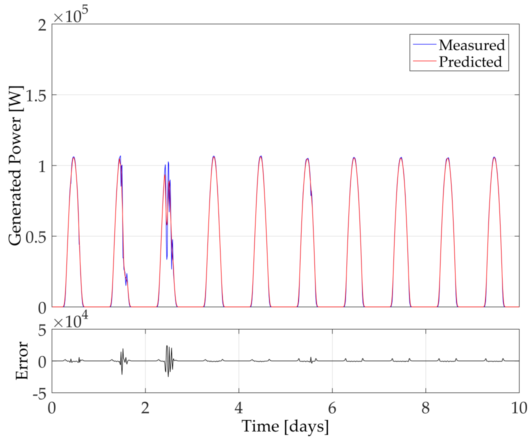



Figure 7: Renewable power generation prediction and measured values.

Within the given definitions and constraints, it is possible to imagine a large number of different distribution options, and corresponding to one of the dispatching setup

$$S(\tau) = \left\{ P_D^k(\tau) \;/\; k \in [0, N] \cap \mathbb{N}_0 \right\} \tag{8}$$

which determines the quantity of power distributed to the different consumers. In this work we have chosen to focus on granting smooth variations of the power sold to the provider that is represented in the mathematical formalism as the consumer number 0. I.e., the managed plant sells a portion of the generated power $P_G$ by dispatching it to the external energy provider as $P_D^0$; moreover due to the characteristics of the energy contracts we give the maximum priority to the stability of $P_D^0$, therefore, at each time step $\tau$, we search for an optimal setup $S_*(\tau)$ so that

$$\left|\dot{P}_D^0(\tau)\right|_{S_*} = \min_{\{S\}} \left\{ \left|\dot{P}_D^0(\tau)\right|_S \right\} \tag{9}$$

where $|\dot{P}_D^0(\tau)|_S$ represents the module of the first time derivate of the distributed power to the energy provider grid at a time step $\tau$ for a setup $S$.

### 5.2. GPU-based dispatch selection

In order to solve the optimal configuration as in (9) under the constraints given by (5), (6), (7), the possible scenarios of (8) have to be simulated, and for this we use a GPU device.

The input data for the developed GPU parallel solver consist of the predicted time series for the generated power $P_G(\tau)$. A solution satisfying all constraints is required for each time step, hence the GPU threads are organised into blocks and specialised for each time step. Therefore, each thread in each block proposes a different solution consisting of a setup for the consumption devices. Of course, if a possible setup does not satisfy the given constraints then such a setup is removed. At the end of this computing step the optimal setup $S_*$ will be selected, as it satisfies the constraints in (9). By collecting all the results, we obtain a time series of predicted optimum setups $S_*(\tau)$.

The GPU-based algorithmic solution is intended to perform very fast computations of different dispatch scenarios, since the computed setup would determine the timely activation or deactivation of some devices. E.g. a computed setup would let us deactivate a device for the following timestep, in order to be activated later on.

## 6. Observations and Results

### 6.1. Overview of the Plant

The observed PV plant has been installed on the roofs of some buildings of the Campus called '*Cittadella*' of the University of Catania as in Fig. 6. The PV system has a nominal peak power of 244.4 kWp and is composed of fields connected in parallel. The fields are: (i) a field having 223 modules arranged in 12 arrays, (ii) a field having 336 modules arranged in 18 arrays, (iii) a field having 223 modules arranged in 12 arrays. The modules are made up of polycrystalline silicon, which guarantees a conversion efficiency of about 13%, and the overall surface is about 1888 $m^2$.

Each array si connected to an inverter that has an open circuit output voltage of 213 VDC and a short circuit output current of 4.4 A. The inverters are connected to the 400 V tree-phase circuit in parallel with the grid of the national energy provider, therefore the PV energy output can be used by any building in the campus. The output is monitored at the PV inverter to gather raw data.

### 6.2. Performed Simulations and Discussion of Results

In the depicted simulation example, a component implementing a WRNN and executing in a SN was used to receive the data characterising a PV field, in order to gain knowledge for predicting, later on, the power production of such a field. Analogously, the other PV fields and load buildings were associated each with a WRNN, receiving the corresponding data. Then, a WRNN was employed to predict the energy consumption of one of the three buildings (the number of plants and buildings is not influent on the functionalities of the component).

We have adopted the above WRNN predictors (see Section 4) for data measuring the power produced by each PV generator

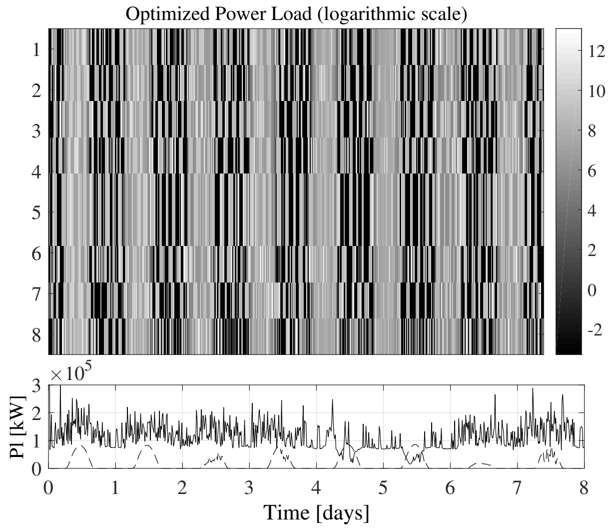



Figure 8: Power load over time adjusted by the proposed management system.

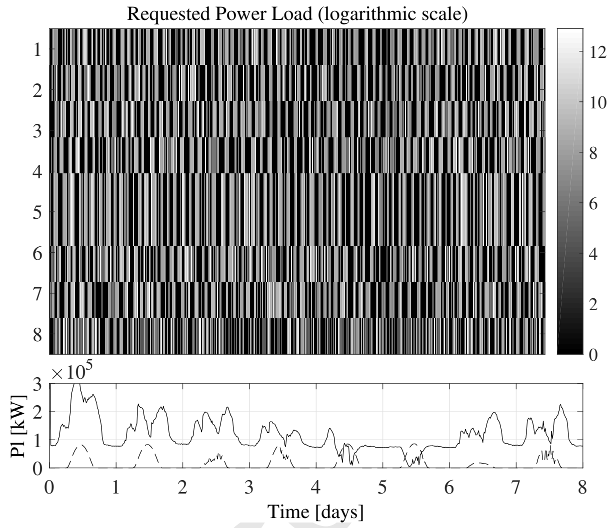



Figure 9: Required power for each loading device over time, total power load (solid line), and generated power (dashed line).

installed in the University of Catania campus. Fig. 7 shows the curves of the predicted power and the measured generated power. The generated power was effectively predicted in advance with a relative root mean square error (RMSE) less than 2.5%.

Fig. 9 shows a simulation consisting of eight consumption devices that request energy, and are served according to their demand. On the top panel, the requested power load is represented as the cells $p_{i,j}$ of a matrix, where $p_{i,j} = \log P_l^i(t_j)$ if $P_l^i(t_j, t_j + \Delta t)$ is the power load of the $i^{th}$ device during a discrete time interval $[t_j, t_j + \Delta t]$ (the color shows the required power, as a logarithm of the actual value, where black shows a lower amount of load, over time on the x-axis). In our simulation the time step $\Delta t$ is 15 minutes. The lower panel shows the overall power load $P_l(t)$, represented as a solid line, and the generated power $P_G(t)$ represented as a dashed line.

Fig. 8 shows the energy dispatch according to the results of the developed solution, which determines the energy flow (graphic notation is the same as Fig. 9). In the resulting dispatching, device requests have been processed, and while some have been honoured, others, when possible, have been rescheduled (for the processing of requests and how the power load and energy consumption forecasts were compared see Section 5). As a result, some loading requests are postponed to the time-intervals when solar panels can produce energy more generated power $P_G$ is available. The alternating vertical light gray stripes in Fig. 8 show high values of energy consumption and indicate that many loading devices requests were honoured in the time-frame corresponding to the width of the light gray stripe. Conversely, mostly-black vertical stripes indicate low energy consumption, hence requests that were postponed. By comparing Fig. 8 and Fig. 9 we can see that power distribution is achieved by ordering requests. Only in the latter diagram there is a correspondence between high volumes of green production (see peaks in the lower panel) and high volumes of consumption (see gray stripes in the higher panel).

Finally, Fig. 10 shows a detail of the proposed dispatching. We can observe that the load of several devices has been rearranged over time, by reallocating some requests, and closely follows the production levels. The introduced optimisation better suits energy availability and reduces the cost due to the energy bought from the national provider.

### 6.3. Performance Measures

In order to evaluate the execution time of the software components that determine energy dispatch solutions, we greatly increased the amount of data to be analysed, simulating more power production plants and consumption devices. When increasing the number of consumption devices, dispatch configurations increase dramatically.

We have called *dispatch setups* the possible configurations that let consumption devices absorb energy (including the energy provider, which buys energy from the PV plant).

When considering $n$ consumption devices, finding which $k$ device requests, with $k < n$, have to be satisfied at a given time step, requires checking a large number of setups where not all $n$ can be given energy.

The *combinations* of $k$ devices to satisfy over the $n$ available devices grows very quickly: there are more than 10000 combinations when taking 8 devices out of 16 whose constraints have to be satisfied. We refer to *combinations number* as the size of such dispatch setups to be checked.

We evaluated the performance of the developed cloud-GPU system by measuring the execution time when seeking dispatch solutions for a varying number of requesting devices, and using several VMs equipped with GPUs. The low measures of execution times let our solution satisfy a big number of consumption devices, and react in real-time to requests, given that 4 GPUs can handle up to 10000 combinations in less than 0.55s. Fig. 11 shows the performance of the parallel GPU version when using multiple GPU-equipped VMs.

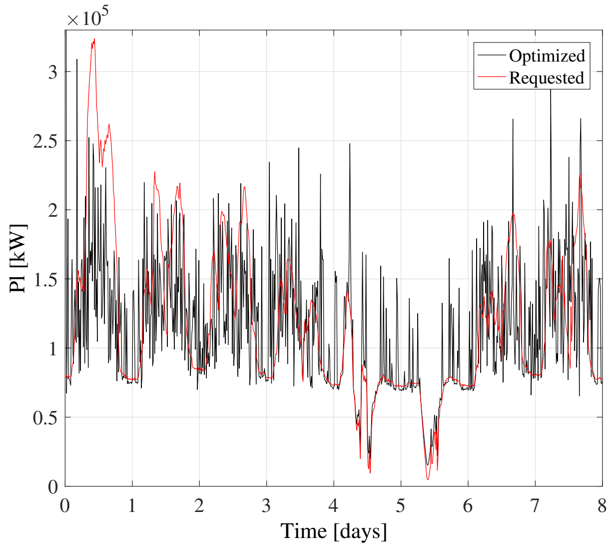



Figure 10: Overall requested power load (red line) and optimised power load (black line).

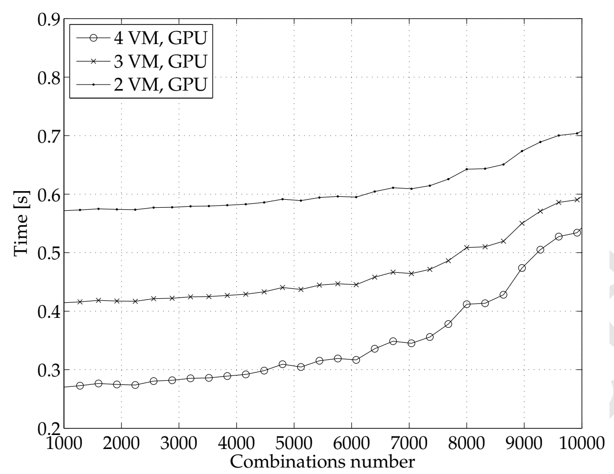



Figure 11: Running time measured for parallel GPU computation while increasing the number of VMs.

## 7. Concluding Remarks

We have presented an advanced management system handling the energy dispatch in a smart grid in order to reduce sudden loss of power from green generation plants, hence fluctuations of energy flows. We have combined several essential components, i.e. a cloud-based software architecture, power monitoring, wavelet transforms, neural networks, and outlet actuators for advancing autonomous and intelligent handling and optimisation of resources. Cloud computing provides us resources on demand, and gives means to properly configure and run the proposed WRNN components. Each of these is associated to an actual device and processes its characteristic consumption data, gathered from actual real time observations. Unlike all other approaches we have seen in the literature (see Section 2) we kept data segregated to improve precision and sensitivity in forecasting the demand levels of each device or the delivered energy of each plant. To achieve proper storage and processing resources we resorted to a cloud-based infrastructure. Wavelet transforms let us process the measured signal in order to retain only the essential characteristics, at both small and large scale. Other previous approaches using wavelet transform and neural networks for forecasting have a much higher error, and cannot adapt to changing conditions of the monitored plant. Neural networks have been used to make a reliable, robust and adaptable estimation of energy production and consumption. Other approaches in the literature using neural network cannot properly tackle not stationary data and have a much higher error, due to the different neural network topology used, or the unfiltered input data.

When compared ours with traditional prediction approaches, the other approaches (i) need much more data, (ii) are less prone to adapt to temporary phenomena, (iii) are less suitable to capture small and large scale characteristics of the signal, due to their lack of filtering on raw data. We tackle filtering with wavelet transforms, and adaptation with neural networks.

Moreover, by means of GPUs and estimated data it was possible to simulate all the possible energy dispatch configurations for loading devices and select the configuration minimising disruption and cost. We believe that this is the first computationally viable solution for a conspicuous number of loading devices, when only partial fulfilment using green energy production is to take into account.

Our solution is the first to tackle the said dispatching problem given the highly complex depicted scenario in such a way that real time processing can be employed. We have tested the solution in a real scenario and it was possible to automatically select the best configuration in order to minimise the unmet load energy demand and then increase the reliability for the smart grid and reduce the amount of energy bought.

## References


[1] A. Fetanat, E. Khorasaninejad, Size optimization for hybrid photovoltaicwind energy system using ant colony optimization for continuous domains based integer programming, Applied Soft Computing 31 (2015) 196 – 209.

[2] K. Christakou, A unified control strategy for active distribution networks via demand response and distributed energy storage systems, Sustainable Energy, Grids and Networks 6 (2016) 1–6.

[3] H. Hooshyar, F. Mahmood, L. Vanfretti, M. Baudette, Specification, implementation, and hardware-in-the-loop real-time simulation of an active distribution grid, Sustainable Energy, Grids and Networks 3 (2015) 36–51.

[4] Y. Zhou, P. Mancarella, J. Mutale, Modelling and assessment of the contribution of demand response and electrical energy storage to adequacy of supply, Sustainable Energy, Grids and Networks 3 (2015) 12–23.

[5] Y. Wang, Q. Chen, C. Kang, Q. Xia, Clustering of electricity consumption behavior dynamics toward big data applications, IEEE Transactions on Smart Grid 7 (5) (2016) 2437–2447.

[6] A. Azadeh, S. Ghaderi, S. Sohrabkhani, Forecasting electrical consumption by integration of neural network, time series and anova, Applied Mathematics and Computation 186 (2) (2007) 1753–1761.

[7] L. Ekonomou, Greek long-term energy consumption prediction using artificial neural networks, Energy 35 (2) (2010) 512 – 517.

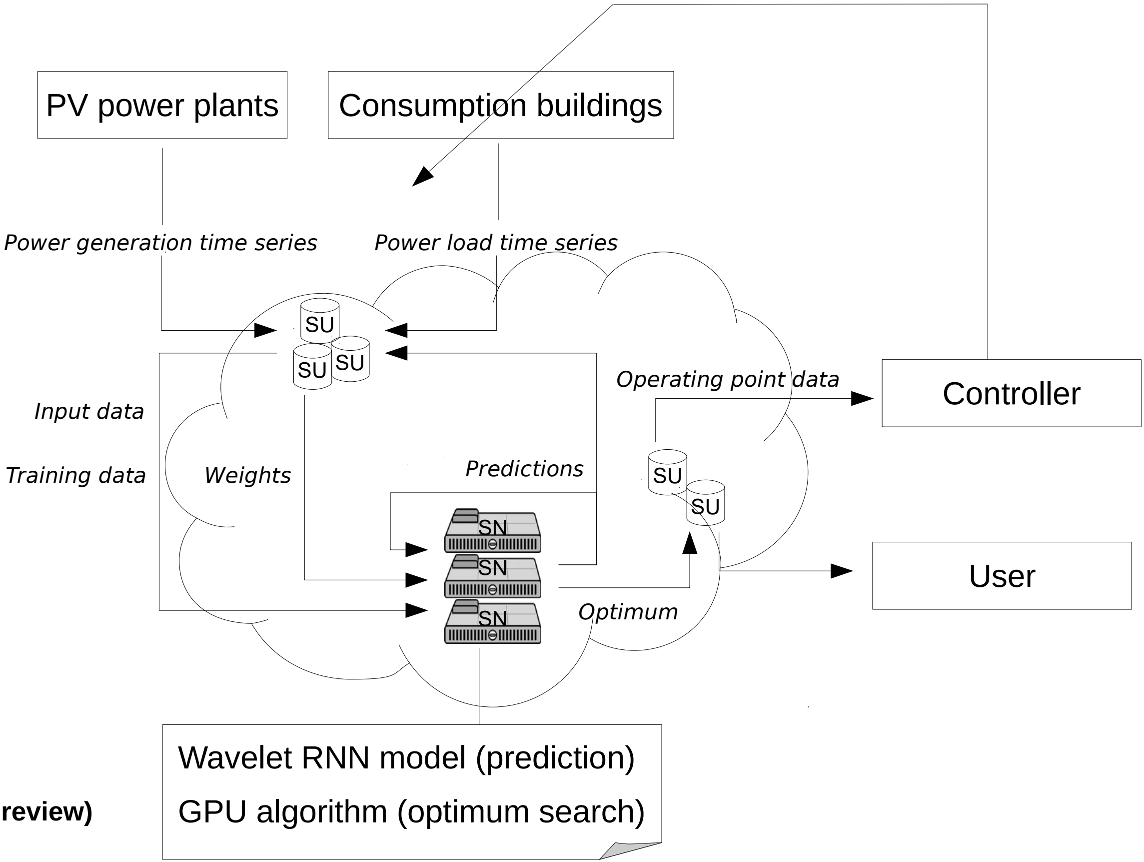
[8] M. Ghofrani, M. Ghayekhloo, R. Azimi, A novel soft computing framework for solar radiation forecasting, Applied Soft Computing 48 (2016) 207–216.

[9] P. A. Gonzlez, J. M. Zamarreo, Prediction of hourly energy consumption in buildings based on a feedback artificial neural network, Energy and Buildings 37 (6) (2005) 595 – 601.

[10] S. Karatasou, M. Santamouris, V. Geros, Modeling and predicting building's energy use with artificial neural networks: Methods and results, Energy and Buildings 38 (8) (2006) 949 – 958.
[11] X. Ma, Y. Jin, Q. Dong, A generalized dynamic fuzzy neural network based on singular spectrum analysis optimized by brain storm optimization for short-term wind speed forecasting, Applied Soft Computing 54 (2017) 296–312.
[12] R. Yokoyama, T. Wakui, R. Satake, Prediction of energy demands using neural network with model identification by global optimization, Energy Conversion and Management 50 (2) (2009) 319 – 327.
[13] H. Chitsaz, H. Shaker, H. Zareipour, D. Wood, N. Amjady, Short-term electricity load forecasting of buildings in microgrids, Energy and Buildings 99 (2015) 50 – 60.
[14] H. Zhu, X. Li, Q. Sun, L. Nie, J. Yao, G. Zhao, A power prediction method for photovoltaic power plant based on wavelet decomposition and artificial neural networks, Energies 9 (1) (2015) 11.
[15] S. Rajagopalan, S. Santoso, Wind power forecasting and error analysis using the autoregressive moving average modeling, in: Proceedings of IEEE Power & Energy Society General Meeting, 2009, pp. 1–6.
[16] M. L. Di Silvestre, E. R. Sanseverino, Modelling energy storage systems using fourier analysis: An application for smart grids optimal management, Applied Soft Computing 14 (2014) 469–481.
[17] M. Meng, D. Niu, W. Sun, Forecasting monthly electric energy consumption using feature extraction, Energies 4 (10) (2011) 1495–1507.
[18] F. Javed, N. Arshad, F. Wallin, I. Vassileva, E. Dahlquist, Forecasting for demand response in smart grids: An analysis on use of anthropologic and structural data and short term multiple loads forecasting, Applied Energy 96 (2012) 150 – 160.
[19] P. Jiang, X. Ma, A hybrid forecasting approach applied in the electrical power system based on data preprocessing, optimization and artificial intelligence algorithms, Applied Mathematical Modelling 40 (23) (2016) 10631–10649.
[20] T. Yalcinoz, U. Eminoglu, Short term and medium term power distribution load forecasting by neural networks, Energy Conversion and Management 46 (9) (2005) 1393–1405.
[21] E. De Santis, A. Rizzi, A. Sadeghian, Hierarchical genetic optimization of a fuzzy logic system for energy flows management in microgrids, Applied Soft Computing 60 (2017) 135–149.
[22] Z. Qi, Z. Li, Energy management based on multi-agent for independent dc micro-grid with wind solar power, in: Proceedings of International Power Electronics and Application Conference and Exposition, 2014, pp. 1226–1229.
[23] J. Warrington, S. Mariéthoz, C. N. Jones, M. Morari, Predictive power dispatch through negotiated locational pricing, in: Proceedings of IEEE Innovative Smart Grid Technologies Conference Europe (ISGT Europe), IEEE, Gothenburg, Sweden, 2010, pp. 1–8.
[24] F. Bonanno, G. Capizzi, A. Gagliano, C. Napoli, Optimal management of various renewable energy sources by a new forecasting method, in: Proceedings of IEEE International Symposium on Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM), Sorrento, Italy, 2012, pp. 934–940.
[25] F. Ugranlı, E. Karatepe, Convergence of rule-of-thumb sizing and allocating rules of distributed generation in meshed power networks, Renewable and Sustainable Energy Reviews 16 (1) (2012) 582–590.
[26] V. Murty, A. Kumar, Optimal placement of dg in radial distribution systems based on new voltage stability index under load growth, International Journal of Electrical Power & Energy Systems 69 (2015) 246–256.
[27] S. Kelo, S. Dudul, A wavelet elman neural network for short-term electrical load prediction under the influence of temperature, International Journal of Electrical Power & Energy Systems 43 (1) (2012) 1063 – 1071.
[28] G. Capizzi, C. Napoli, F. Bonanno, Innovative second-generation wavelets construction with recurrent neural networks for solar radiation forecasting, IEEE Transactions on Neural Networks and Learning Systems 23 (11) (2012) 1805–1815.
[29] W. Sweldens, The lifting scheme: A construction of second generation wavelets, SIAM Journal on Mathematical Analysis 29 (2) (1998) 511–546.
[30] S. Mallat, A wavelet tour of signal processing: the sparse way, Academic press, Cambridge, Massachusetts, USA, 2008.
[31] D. P. Mandic, J. Chambers, Recurrent neural networks for prediction: learning algorithms, architectures and stability, John Wiley & Sons, Inc., Hoboken, New Jersey, USA, 2001.
[32] Q. Song, Y. Wu, Y. C. Soh, Robust adaptive gradient-descent training algorithm for recurrent neural networks in discrete time domain, IEEE Transactions on Neural Networks 19 (11) (2008) 1841–1853.

Highlights:

1) We want to reschedule user devices' energy consuption to overcome the insufficient IGS production
3) We use WRNNs to predict future energy production in order to forecast energy availability
4) We use GPU processing to simulate different energy consuption scenarios
5) Simulations show which activation of consumption devices can be postponed
6) Therefore the proposed overall solution regulates energy flows between production plants and consumption buildings


PV power plants
Consumption buildings
Power generation time series
Power load time series
SU
SU
SU
Input data
Training data
Weights
Predictions
SN
SN
SN
Optimum
Operating point data
SU
SU
Controller
User
Wavelet RNN model (prediction)
GPU algorithm (optimum search)

**r review)**