

Online Tuned Neural Networks for Fuzzy Supervisory Control of PV-Battery Systems

Lucio Ciabattoni, *Student Member, IEEE*, Gianluca Ippoliti, *Member, IEEE*,
Sauro Longhi, *Senior Member, IEEE*, and Matteo Cavalletti

Abstract—The paper deals with a neural network based fuzzy supervisor control to manage power flows in a Photo-Voltaic (PV) - Battery system. An on-line self-learning prediction algorithm is used to forecast, over a determined time horizon, the power mismatch between PV production and electrical consumptions. The learning algorithm is based on a Radial Basis Function (RBF) network and combines the growing criterion and the pruning strategy of the minimal resource allocating network technique. The power flows are scheduled by a Fuzzy Logic Supervisor (FLS) which controls the charge and discharge of a battery used as an energy buffer. The proposed solution has been experimentally tested on a 14 KWp PV plant and a lithium battery pack.

I. INTRODUCTION

The need to reduce gas emissions, the rising prices of the retail electricity and the decreasing cost of the photovoltaic (PV) technology have led to a large scale development of PV distributed generation in electrical grids. Nowadays less attractive PV feed-in-tariffs and incentives to promote self-consumption suggest that new operation modes for PV should be explored in order to reach grid parity, which has been predicted to become a reality in the next years in the European Union [1]–[4]. By increasing the selfconsumed local generated energy, the grid parity could be achieved earlier and solar power will finally make economic sense becoming cheaper (over the lifetime of the system) than to buy it from electricity [5], [6]. In particular, there are two tasks to integrate Distributed Energy Resources (DER), both locally and globally: integrating them into the electricity network and into the energy market. One solution to decrease the problems caused by the variable output of some distributed generation is to add energy storages into the systems (centralised or distributed energy storages [7]–[9]). The main reasons of storage absence in the grid are its expensive cost and the generation control possibilities of the classical power stations, making apparently not profitable the implementation of new storage technologies on a large scale. However, the current technological and energetic situation encourages the investment in storage systems. In fact a massive use of renewable energy generation implicates a strong hourly mismatch between demand and generation

patterns and the call for new grid infrastructures where existing transmission lines are not able to accept significant amounts of DER. In this paper we study the effects of a smart energy storage system used to reduce variability of the electricity exported to the grid, for power flow management and to use PV electricity indirectly, maximizing self consumption. The whole system has been developed with the help of the company Energy Resources SPA an consists of 14 KW_p of PV plant grid connected, a lithium battery pack, a controllable inverter, a battery charger and a smart meter. We carried out simulations for long-time experiments (yearly studies) and real measures for short and mid-time experiments (daily and weekly studies). From an engineering perspective, supervision and management are necessary to know in real-time the status of the system components, as well as to forecast the expected generation and consumption in a short time scale (for example, on a 24-h basis) [10]–[12]. In particular, in this paper, Radial Basis Function Networks (RBFNs) have been used for the prediction of the energy mismatch between production and consumption. The considered on-line learning algorithm is based on the Minimal Resource Allocating Network (MRAN) technique [13]–[15], that adds hidden neurons to the network based on the innovation of each new RBFN input pattern which arrives sequentially. As stated in [14], [15], to obtain a more parsimonious network topology a pruning strategy is introduced. Pruning is necessary in this forecast because inactive hidden neurons could be present as the dynamics which caused their creation becomes nonexistent. If an observation has no novelty then the existing parameters of the network are adjusted by an Extended Kalman Filter (EKF) [15], [16]. The main advantage of this sequential learning method is that a large data set of measured PV production, consumption, weather forecast, temperature for a specific location is no longer required for the training of the Neural Network, drastically reducing setup time. A Fuzzy Logic Supervisor (FLS) ([17], [18]) and two PID controllers have been designed to manage power flows, using the information provided by NNs forecasts. The principal characteristics of this controller are:

- Manage the battery charge and discharge (the controllable inverter has current limiters, which are used to modify the power flows) in order to maximize selfconsumption
- The battery controller has to limit strong fluctuations of the exported PV energy

This work was supported by the company Energy Resources spa
L. Ciabattoni, G. Ippoliti, S. Longhi are with the Dipartimento di Ingegneria dell'Informazione, Università Politecnica delle Marche, 60131 Ancona, Italy, e-mail: {l.ciabattoni, gianluca.ippoliti, sauro.longhi}@univpm.it
M. Cavalletti is with the company Energy Resources spa, via Ignazio Silone, 60035 Jesi (AN), Italy, e-mail: matteo.cavalletti@energyresources.it

- Preserve the battery against overcharge and overdischarge.
- Minimize the inactivity time of the battery (a well known problem in storage systems due to their self-discharge rate).

The paper is organized as follows. The on-line prediction algorithm is described in Section II and the performance of the considered NNs are discussed in Section III. The proposed supervisory control is described in Section IV.

II. PREDICTION ALGORITHM

The following approach to implement a Minimal Resource Allocating Network (MRAN) is based on a sequential learning algorithm and an Extended Kalman Filter (EKF) [13], [15], [16], [19]. In particular the sequential learning algorithm adds or removes neurons on-line to the network according to a given criterion [13]–[15] and an EKF is used to update the net parameters [16].

A. Radial Basis Function Neural Network

A RBFN with input pattern $\mathbf{x} \in \mathbb{R}^m$ and a scalar output $\hat{y} \in \mathbb{R}$ implements a mapping $f: \mathbb{R}^m \rightarrow \mathbb{R}$ according to

$$\hat{y} = f(\mathbf{x}) = \lambda_0 + \sum_{i=1}^K \lambda_i \phi(\|\mathbf{x} - \mathbf{c}_i\|) \quad (1)$$

where $\phi(\cdot)$ is a given function from \mathbb{R}^+ to \mathbb{R} , $\|\cdot\|$ denotes the Euclidean norm, λ_i , $i = 0, 1, \dots, K$ are the weight parameters, $\mathbf{c}_i \in \mathbb{R}^m$, $i = 1, 2, \dots, K$, are the radial basis function centers (called also units or neurons) and K is the number of centers [20]. The terms:

$$o_i = \lambda_i \phi(\|\mathbf{x} - \mathbf{c}_i\|), \quad i = 1, \dots, K \quad (2)$$

are called the hidden unit outputs.

In this paper the RBFN is used for the prediction of the output of a dynamical system and the system dynamics can be taken into account through the network input pattern \mathbf{x} , that must be composed of a proper set of system input and output samples acquired in a finite set of past time instants [21], i.e. $\mathbf{x} \in \mathbb{R}^{n_y + n_u}$ and it is defined as:

$$\mathbf{x}(n) = [y(n-1), \dots, y(n-n_y), u(n-1), \dots, u(n-n_u)]^T \quad (3)$$

where $n = 1, 2, \dots$ are the time instants, $y(\cdot)$ and $u(\cdot)$ are the system output (the lack or exceeding power; see Section III) and input (whether forecast, the number of day of the year, the hour of the day, the day of the week; see Section III), respectively; n_y , n_u are the lags of the output and input, respectively.

Theoretical investigation and practical results show that the choice of the non-linearity $\phi(\cdot)$, a function of the distance d_i between the current input \mathbf{x} and the centre \mathbf{c}_i , does not significantly influence the performance of the RBFN [20]. Therefore, the following gaussian function is considered:

$$\phi(d_i) = \exp(-d_i^2/\beta_i^2), \quad i = 1, 2, \dots, K \quad (4)$$

where $d_i = \|\mathbf{x} - \mathbf{c}_i\|$ and the real constant β_i is a scaling or “width” parameter [20].

B. Minimal Resource Allocating Network Algorithm

The learning process of MRAN involves allocation of new hidden units and a pruning strategy as well as adaptation of network parameters [13], [15], [16]. The network starts with no hidden units and as input-output data $(\mathbf{x}(\cdot), y(\cdot))$ are received, some of them are used to generate new hidden units based on a suitably defined growth criteria. In particular at each time instant n the following three conditions are evaluated to decide if the input $\mathbf{x}(n)$ should give rise to a new hidden unit:

$$\|e(n)\| = \|y(n) - f(\mathbf{x}(n))\| > E_1 \quad (5)$$

$$e_{rms}(n) = \sqrt{\sum_{j=n-(M-1)}^n \frac{e(j)^2}{M}} > E_2 \quad (6)$$

$$d(n) = \|\mathbf{x}(n) - \mathbf{c}_r(n)\| > E_3 \quad (7)$$

where $\mathbf{c}_r(n)$ is the centre of the hidden unit that is nearest to $\mathbf{x}(n)$ and M represents the number of past network outputs for calculating the output error $e_{rms}(n)$. The terms E_1 , E_2 and E_3 are thresholds to be suitably selected. As stated in [14], [15], these three conditions evaluate the novelty in the data. If all the criteria of relations (5)–(7) are satisfied, a new hidden unit is added and the following parameters are associated with it:

$$\lambda_{K+1} = e(n) \quad (8)$$

$$\mathbf{c}_{K+1} = \mathbf{x}(n) \quad (9)$$

$$\beta_{K+1} = \alpha \|\mathbf{x}(n) - \mathbf{c}_r(n)\| \quad (10)$$

where α determines the overlap of the response of a hidden unit in the input space as specified in [15], [16]. If the observation $(\mathbf{x}(n), y(n))$ does not satisfy the criteria of relations (5)–(7), an EKF is used to update the following parameters of the network:

$$\mathbf{w} = [\lambda_0, \lambda_1, \mathbf{c}_1^T, \beta_1, \dots, \lambda_N, \mathbf{c}_N^T, \beta_N]^T \quad (11)$$

The update equation is given by:

$$\mathbf{w}(n) = \mathbf{w}(n-1) + \mathbf{k}(n)e(n) \quad (12)$$

where the gain vector $\mathbf{k}(n)$ is expressed by:

$$\mathbf{k}(n) = \mathbf{P}(n-1)\mathbf{a}(n) [r(n) + \mathbf{a}^T(n)\mathbf{P}(n-1)\mathbf{a}(n)]^{-1} \quad (13)$$

with $\mathbf{a}(n)$ the gradient vector of the function $f(\mathbf{x}(n))$ (see Eq. 1) with respect to the parameter vector $\mathbf{w}(n-1)$ [15], [16], $r(n)$ is the variance of the measurement noise and $\mathbf{P}(n-1)$ is the error covariance matrix which is updated by:

$$\mathbf{P}(n) = [I - \mathbf{k}(n)\mathbf{a}^T(n)] \mathbf{P}(n-1) + \mathbf{Q}(n-1) \quad (14)$$

where $\mathbf{Q}(n-1)$ is introduced to avoid that the rapid convergence of the EKF algorithm prevents the model from adapting to future data [15], [16]. The $z \times z$ matrix $\mathbf{P}(n)$ is positive definite symmetric and z is the number of parameters

to be adjusted. When a new hidden neuron is allocated, the dimension of $\mathbf{P}(n)$ increases to:

$$\mathbf{P}(n) = \begin{pmatrix} \mathbf{P}(n-1) & 0 \\ 0 & p_0 \mathbf{I}_{z_1 \times z_1} \end{pmatrix} \quad (15)$$

where p_0 is an estimate of the uncertainty in the initial values assigned to the parameters and z_1 is the number of new parameters introduced by adding the new hidden neuron. As stated in [14], [15], to keep the RBF network in a minimal size a pruning strategy removes those hidden units that contribute little to the overall network output over a number of consecutive observations. To carry out this pruning strategy, for every observation $(\mathbf{x}(n), y(n))$ the hidden unit outputs are computed:

$$o_i(n) = \lambda_i \phi(\|\mathbf{x}(n) - \mathbf{c}_i\|), \quad i = 1, \dots, K \quad (16)$$

and normalized with respect to the highest output:

$$\bar{o}_i(n) = \frac{o_i(n)}{\max\{o_i(n)\}}, \quad i = 1, \dots, K. \quad (17)$$

The hidden units for which the normalized output (17) is less than a threshold δ for ξ consecutive observations are removed and the dimensionality of all the related matrices are adjusted to suit the reduced network [14], [15].

The EKF has been implemented with the assumption that $\mathbf{Q}(n) = \mathbf{I}_{z \times z} \sigma_\eta^2$ and $r(n) = \sigma_v^2$.

The MRAN prediction algorithm [14], [15] with the EKF, here called MRANEKF algorithm is briefly summarized as follow:

1. For each observation $(\mathbf{x}(n), y(n))$ do: compute the overall network output: $\hat{y}(n) = f(\mathbf{x}(n)) = \lambda_0 + \sum_{i=1}^K \lambda_i \phi(\|\mathbf{x}(n) - \mathbf{c}_i\|)$ where K is the number of hidden units;
2. Calculate the parameters required by the growth criterion:
 - $\|e(n)\| = \|y(n) - f(\mathbf{x}(n))\|$
 - $e_{rms}(n) = \sqrt{\sum_{j=n-(M-1)}^n \frac{e(j)^2}{M}}$
 - $d(n) = \|\mathbf{x}(n) - \mathbf{c}_r(n)\|$
3. Apply the criterion for adding a new hidden unit:

if

 $\|e(n)\| > E_1$ **and** $e_{rms}(n) > E_2$ **and** $d(n) > E_3$
allocate a new hidden unit $K + 1$ with:
 - $\lambda_{K+1} = e(n)$
 - $\mathbf{c}_{K+1} = \mathbf{x}(n)$
 - $\beta_{K+1} = \alpha \|\mathbf{x}(n) - \mathbf{c}_r(n)\|$

else

 - tune the network parameters:
 $\mathbf{w}(n) = \mathbf{w}(n-1) + \mathbf{k}(n)e(n)$
 - update the error covariance matrix:
 $\mathbf{P}(n) = [\mathbf{I} - \mathbf{k}(n)\mathbf{a}^T(n)] \mathbf{P}(n-1) + \mathbf{Q}(n-1)$

end
4. Check the criterion to prune hidden units:
 - compute the hidden unit outputs:
 $o_i(n) = \lambda_i \phi(\|\mathbf{x}(n) - \mathbf{c}_i\|), \quad i = 1, \dots, K$

- compute the normalized outputs:
 $\bar{o}_i(n) = \frac{o_i(n)}{\max\{o_i(n)\}}, \quad i = 1, \dots, K$
- **if** $\bar{o}_i(\cdot) < \delta$ for ξ consecutive observations **than**
prune the i th hidden unit and reduce the dimensionality of the related matrices

end

5. $n = n + 1$ and **go** to step 1.

III. NEURAL NETWORK BASED FORECASTS

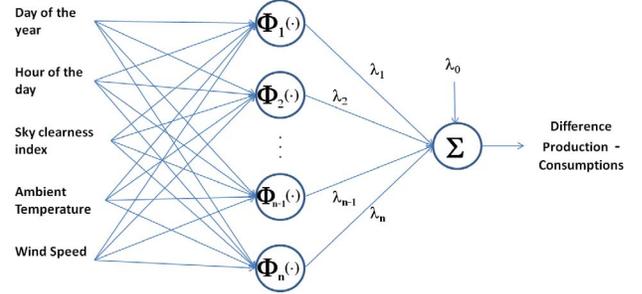


Fig. 1. Input-output structure of the network.

Tests are based on data acquired from January 2012 to July 2012 during PV plant standard working. In particular the considered $14KW_p$ PV plant, equipped with polysilicon solar panels south oriented and tilt angle 27 deg., is located in Jesi (AN), Italy. The MRANEKF learning algorithm starts with a pre-trained net based only on few historical information found on the web such as power production profile of clear sky days and cloudy days for the specified location, panel orientation and tilting and mean consumption of the office. This is a common operating condition, when no sensors and measures are available for the plant before the forecast begins. To measure the performance of the proposed algorithm, the normalized Root Mean Square of the Error $e(\cdot)$ (RMSE), its Standard Deviation (SD) and the percentage RMSE have been calculated. The set of experimental data is composed of 8000 pairs of input and output samples. Sampling time is 1 h and the data have been normalized, between 0 and 1, in order to have the same range. Normalization is a widely used preprocessing method performed on data to distribute them evenly and to scale them into an acceptable range for the input neurons of the NN. This contribute to increase the NN ability to learn the association between inputs and outputs as well as to fasten significantly the calculations [22]–[24]. In particular the set of experimental data is given by the pairs $(\mathbf{x}(n), y(n))$, $n = 1, 2, \dots$ where $\mathbf{x}(n)$ is composed by the whether forecast, the number of day of the year, the hour of the day, the air temperature, the wind speed and $y(\cdot)$ is the mismatch between PV production and office consumptions, as shown in Fig.1.

In Fig. 2 has been reported a sample of the network results. The measured mismatch between PV power production and consumption (dotted red line) is compared with the one predicted by the pre-trained MRANEKF network (continuous

blue line) for the considered PV plant. Results for the data set (relative to the period January - July 2012) have been summarized in Table I.

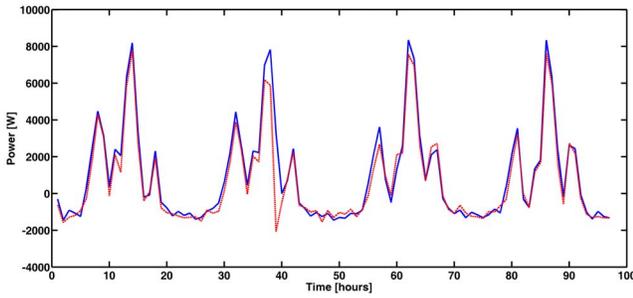


Fig. 2. Jesi (AN), Italy. The dotted red line is the measured mismatch between production and consumptions, the continuous blue line is the MRANEKF network forecast.

TABLE I
DATA SET RELATIVE TO THE PERIOD JANUARY - JULY 2012

DATA	RMSE	SD	RMSE%
pre-trained MRANEKF	0,0971	0,0849	10,9%

The whiteness test on the prediction errors $e(\cdot)$ (residuals) has been also used for network validation [25]. The whiteness of residuals is usually evaluated by computing the sample covariances

$$\hat{R}_e^N(\tau) = \frac{1}{N} \sum_{n=1}^N e(n)e(n+\tau) \quad (18)$$

with $\tau = 1, \dots, P$.

If $e(\cdot)$ is a white-noise sequence, then the quantity

$$\zeta_{N,P} = \frac{N}{(\hat{R}_e^N(0))^2} \sum_{\tau=1}^P (\hat{R}_e^N(\tau))^2 \quad (19)$$

will have, asymptotically, a chi-square distribution $\chi^2(P)$ [25]. The independence between residuals can be verified by testing whether $\zeta_{N,P} < \chi_\alpha^2(P)$, the α level of the $\chi^2(P)$ -distribution, for a significant choice of α . The whiteness test for this network passes with $\alpha = 0.05$.

IV. EXPERIMENTAL IMPLEMENTATION

The experimental setup is shown in Fig. 3 (the meaning of signals and blocks shown in Fig. 3 will be explained in the following). It is composed by 8 strings of Renergies 220P/220 polysilicon panels [26] where each string is connected to a SMA Sunny Boy 1700IT solar inverter [27]. A lithium battery pack is composed by the series of two sub-module with 80 ThunderSky modules 40 Ah, a Battery Management System (BMS) and a battery charger for each module [28]. A solar inverter (model SIAC soleil 10Kw) is connected to this pack [29]. A power meter from Schneider Electric (model PM9P) is used to measure consumptions of the office. All communication is done through the TCP/IP protocol, using serial to TCP/IP converters. All devices are

connected to a server, where it is located the software to manage the whole system.

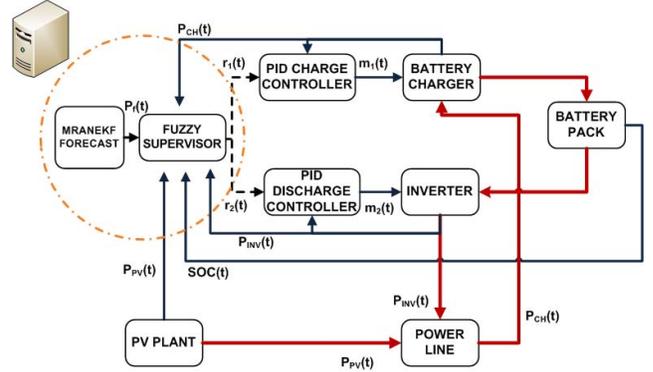


Fig. 3. Scheme of the whole system.

In Fig. 3 red continuous lines represent power fluxes, blue continuous lines are the information fluxes that goes over TCP/IP protocol and black dotted lines are data exchanged between software components. The supervisor, the MRANEKF RBF network and PID controllers have been developed by LabVIEW, the graphical programming environment of National Instruments (see [30]).

The aims of the proposed supervised controller are, in a priority order:

- Preserve the battery against overcharge and overdischarge (the battery state of charge has to remain between 20% and 90%) and regulate the charging rate (using the NNs informations).
- Maximize self consumption of PV production, storing the excess power to re-use it during low production periods.
- Reduce variability of the electricity exported to the grid.

A. Control Strategy

As shown in Fig. 3, the structure of the whole control system is composed by two different low level PID controllers that regulate the inverter power generation and the battery charging and the supervisor. The setpoints $r_1(t)$ and $r_2(t)$ of PID controllers for the inverter and battery charger, respectively, (see Fig. 3) are computed by the supervisor using fuzzy logic inference.

The measures used as feedback by the two inner PID controllers are the absorbed power ($P_{CH}(t)$) and the power fed to the line ($P_{INV}(t)$). The outputs of PID controllers ($m_1(t)$ and $m_2(t)$ in Fig. 3) are the reference signals used by the drive of the inverter and battery charger, respectively. The PID sampling time is 100 ms; the supervisor acts every 10 s.

B. Fuzzy Logic Supervisor

The Fuzzy Logic Supervisor (FLS) determines the power references for the two inner PID controllers, in order to obtain the performance described above. The input set for fuzzy inference is composed by the difference between the measured PV production and consumption ($e(t) = P_{PV}(t) -$

$C(t)$), the 2 hour ahead forecast of energy mismatch between production and consumption ($e_2(t)$ provided by NNs) and the measure of battery State Of Charge (SOC(t)). The membership functions of the input variables are shown in Fig. 4 and consist of triangular asymmetric and trapezoidal functions. The fuzzy sets considered are: Negative Very Big (NB), Negative Big (NB), Negative Medium (NM), Negative Small (NS), Zero (Z), Positive Small (PS), Positive Medium (PM), Positive Big (PB) and Positive Very Big (PB) for $e(t)$; NB, NS, Z, PS, PB for $e_2(t)$; Very Low (VL), Low (L), Medium (M), High (H), Very High (VH) for SOC(t). A sample of the fuzzy control rule base of the supervisor is shown in Table II; the Max-Min fuzzy inference algorithm is considered, while the defuzzification is performed according to the centroid method ([18])

The aim of these rules is to regulate the charge and discharge of the battery pack in order to maximize selfconsumption of PV production and on the same time prevent damages to the battery pack and minimize its inactivity time. In particular can be noticed that, to prevent damages on battery, when SOC(t) is lower than 20% or higher than 90% (first three rows on Table II), for each input set, the action of the controller is to keep the battery out of the critical situation. On the same time, when $e(t)$ is positive (medium, big or very big) and $e_2(t)$ is positive big, the controller reduces the instantaneous charging rate of the battery (predicting at least other 2 hours of charge) in relation to the battery SOC (last 4 rows on Table II). When an input set arrives, from 1 to 8 of the 140 rules are activated to compute the value of $r(t)$. This value, if positive, becomes the reference value $r_1(t)$ for the PID of the battery charger ($r_2(t) = 0$, see Fig. 3); otherwise it is the reference $|r_2(t)|$ for the PID of the inverter with $r_1(t) = 0$.

TABLE II
SAMPLE OF THE FUZZY RULES USED FOR INFERENCE.

$e(t)$	$e_2(t)$	$SOC(t)$	$r(t)$
PS	NS	VH	NS
PM	Z	VH	NS
NM	PS	VL	PS
PB	PB	L	PS
PB	PB	M	PM
PS	PB	M	PS

Nine fuzzy sets are considered for the output variable: NVB, NB, NM, NS, Z, PS, PM, PB and PVB, the membership functions are shown in Fig. 5

C. Experimental Results

This supervisory control system has been tested from July to August 2012. In Table III is reported the percentage of selfconsumption of the considered period with respect the same period of the previous year and the percentage of battery inactivity time.

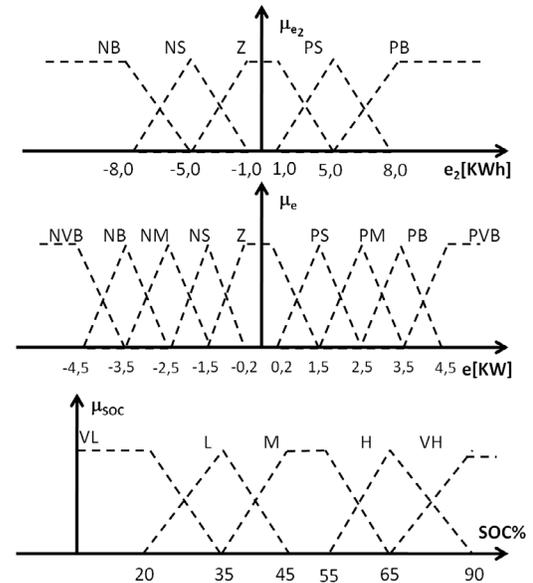


Fig. 4. Fuzzyfication of the input sets.

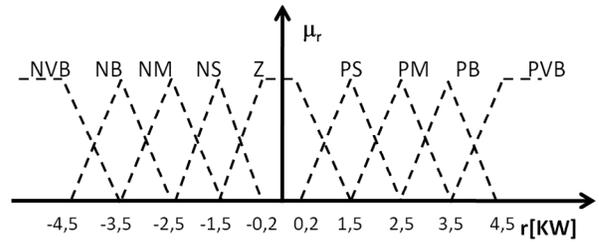


Fig. 5. Fuzzyfication of the output variable.

TABLE III
SELFCONSUMPTION RATE OF THE CONSIDERED PERIOD AND PERCENTAGE OF INACTIVITY TIME OF THE BATTERY

DATA	Selfconsumption	Battery inactive
Jul-Aug 2012	83.12%	7.22%
Jul-Aug 2011	65.91%	No Battery

These results show that the experimental setup (14 KWp of PV plant and 20 KWh of battery pack), equipped with the NN-based forecasting algorithm and the fuzzy logic supervisor, is able to increase the selfconsumption by 18% , also protecting the battery without high rate of charge. In Fig. 7 is shown the produced and consumed power of the PV plant and the corresponding trend of the battery state of charge (12 – 15 July 2012).

V. CONCLUDING REMARKS

In this paper a fuzzy logic based supervisory control system is proposed to manage a PV-Battery System. On-line tuned neural network have been considered to forecast of the power mismatch between PV production and consumption.

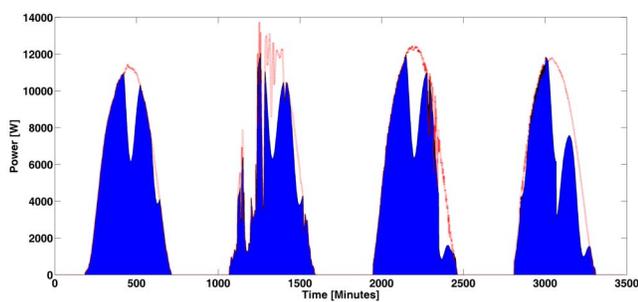


Fig. 6. 12 – 15 July 2012. PV produced power is the red dotted line line; the self consumed power is the blue area.

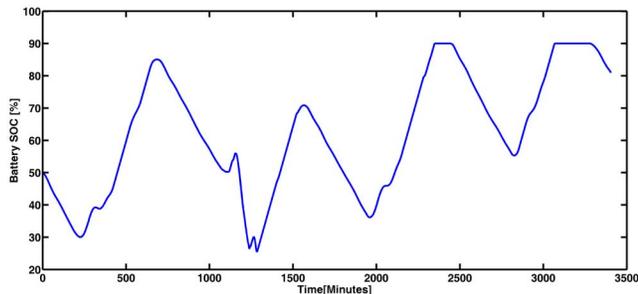


Fig. 7. 12 – 15 July 2012. Battery state of charge.

This forecast is used by the supervisor to manage the power flows, increasing selfconsumption and prevent damages to the battery.

REFERENCES

- [1] H. Kanchev, D. Lu, F. Colas, V. Lazarov, and B. Francois, "Energy management and operational planning of a microgrid with a pv-based active generator for smart grid applications," *IEEE Trans. on industrial electronics*, vol. 58, no. 10, pp. 4583–4592, 2011.
- [2] P. Palensky and D. Dietrich, "Demand side management: Demand response, intelligent energy systems, and smart loads," *IEEE Transactions on Industrial Informatics*, vol. 7, p. 381:388, 2011.
- [3] A. Fazeli, E. Christopher, C. Johnson, M. Gillott, and M. Sumner, "Investigating the effects of dynamic demand side management within intelligent smart energy communities of future decentralized power system," in *Innovative Smart Grid Technologies (ISGT Europe), 2011 2nd IEEE PES International Conference and Exhibition on*, dec. 2011, pp. 1 –8.
- [4] Y. Zong, D. Kullmann, A. Thavlov, O. Gehrke, and H. Bindner, "Application of model predictive control for active load management in a distributed power system with high wind penetration," *Smart Grid, IEEE Transactions on*, vol. 3, no. 2, pp. 1055 –1062, june 2012.
- [5] A. Lopez-Polo, R. Haas, C. Panzer, and H. Auer, "Prospects for grid-parity of photovoltaics due to effective promotion schemes in major countries," in *Power and Energy Engineering Conference (APPEEC), 2012 Asia-Pacific*, march 2012, pp. 1 –4.
- [6] D. Lewis, "Solar grid parity - [power solar]," *Engineering Technology*, vol. 4, no. 9, pp. 50 –53, may - 5 june 23 2009.
- [7] E. Veldman, M. Gibescu, J. Slootweg, and W. Kling, "Technical benefits of distributed storage and load management in distribution grids," in *PowerTech, 2009 IEEE Bucharest*, 28 2009-july 2 2009, pp. 1 –8.
- [8] X. Vallve, A. Graillot, S. Gual, and H. Colin, "Micro storage and demand side management in distributed pv grid-connected installations," in *Electrical Power Quality and Utilisation, 2007. EPQU 2007. 9th International Conference on*, oct. 2007, pp. 1 –6.
- [9] L. Ciabattani, G. Ippoliti, S. Longhi, M. Cavalletti, and M. Rocchetti, "Solar irradiation forecasting using rbf networks for pv systems with storage," in *IEEE ICIT Conference 2012*, 2012.

- [10] Z. Osman, M. Awad, and T. Mahmoud, "Neural network based approach for short-term load forecasting," in *Power Systems Conference and Exposition, 2009. PSCE '09. IEEE/PES*, march 2009, pp. 1 –8.
- [11] A. Dorvlo, J. Jervase, and A. Ali, "Solar radiation estimation using artificial neural networks," *Applied Energy*, vol. 71, pp. 307–319, 2002.
- [12] M. Lange, "A mature market? the history of short term prediction services," 2006.
- [13] J. Platt, "A resource allocating network for function interpolation," *Neural Computation*, vol. 3, pp. 213–225, 1991.
- [14] L. Yingwei, N. Sundararajan, and P. Saratchandran, "Performance evaluation of a sequential minimal radial basis function (RBF) neural network learning algorithm," *IEEE Trans. Neural Networks*, vol. 9, no. 2, pp. 308–318, 1998.
- [15] N. Sundararajan, P. Saratchandran, and Y. Li, *Fully tuned radial basis function neural networks for flight control*. London, Great Britain: Kluwer Academic, 2002.
- [16] V. Kadiramanathan and M. Niranjan, "A function estimation approach to sequential learning with neural network," *Neural Computation*, vol. 5, pp. 954–975, 1993.
- [17] S.G.Li, S.M.Sharkh, F.C.Walsh, and C.N.Zhang, "Energy and battery management of a plug-in series hybrid electric vehicle using fuzzy logic," *IEEE Transactions on Vehicular Technology*, vol. 60, p. 3571:3585, 2011.
- [18] B. Bose, "Fuzzy logic and neural networks in power electronics and drives," *IEEE Industry Applications Magazine*, vol. 6, no. 3, p. 57:63, 2011.
- [19] A.Giantomassi, G.Ippoliti, S.Longhi, I.Bertini, and S.Pizzuti, "On-line steam production prediction for a municipal solid waste incinerator by fully tuned minimal rbf neural networks," *Int. J. of Contr.*, vol. 21, p. 164:172, 2011.
- [20] S. Chen, C. F. N. Cowan, and P. M. Grant, "Orthogonal least squares learning algorithm for radial basis function networks," *IEEE Trans. Neural Networks*, vol. 2, no. 2, pp. 302–309, 1991.
- [21] K. J. Hunt, D. Sbarbaro, R. Zbikowski, and P. J. Gawthrop, "Neural networks for control systems - a survey," *Automatica*, vol. 28, no. 6, pp. 1083–1112, 1992.
- [22] J. Sola and J. Sevilla, "Importance of input data normalization for the application of neural networks to complex industrial problems," *IEEE Transactions on Nuclear Science*, vol. 44, no. 3, pp. 1464–1468, June 1997.
- [23] G. Chakraborty and B. Chakraborty, "A novel normalization technique for unsupervised learning in ann," *IEEE Transactions on Neural Networks*, vol. 11, no. 1, pp. 253–257, Jan. 2000.
- [24] F. Aminian, M. Aminian, and H. W. Collins, "Analog fault diagnosis of actual circuits using neural networks," *IEEE Transactions on Instrumentation and Measurement*, vol. 51, no. 3, pp. 544–550, June 2002.
- [25] L. Ljung, *System identification, theory for the user*, ser. Information and System Sciences Series. New Jersey, USA: Prentice Hall PTR, 1999.
- [26] "Renergies italia website," 2011. [Online]. Available: <http://www.renergiesitalia.it/>
- [27] "Sma solar technology website," 2011. [Online]. Available: <http://www.sma-italia.com/it/ prodotti/ inverter-solari/sunny-boy.html>
- [28] "Winston battery website," 2011. [Online]. Available: <http://en.winston-battery.com/>
- [29] "Siel energy and safety website," 2011. [Online]. Available: <http://www.sielups.com/INVERTER/Fotovoltaico/SIAC-Soleil-10-TL.aspx>
- [30] "National instruments labview website," 2011. [Online]. Available: <http://www.ni.com/labview/>