BULETINUL INSTITUTULUI POLITEHNIC DIN IAŞI
Publicat de
Universitatea Tehnică „Gheorghe Asachi” din Iaşi
Tomul LX (LXIV), Fasc. 4, 2014
Secţia
ELECTROTEHNICĂ. ENERGETICĂ. ELECTRONICĂ

IMPROVEMENT OF A NEURAL NETWORK BASED STATE ESTIMATION ALGORITHM BY USING PMU MEASUREMENTS

BY

OVIDIU IVANOV* **and MIHAI GAVRILAŞ**

“Gheorghe Asachi” Technical University of Iaşi
Faculty of Electrical Engineering

Received: October 14, 2014
Accepted for publication: November 20, 2014

Abstract. By providing system-wide synchronized voltage measurements, phasor measurement units improve the precision of classic state estimation algorithms used in real time power systems management and control. Worldwide scientific research studies proved that adding synchronized phasor measurements into the state estimation’s input measurement set results in a better estimation of the state variables. This paper investigates the opportunity of using these synchrophasor measurements for improving the estimation precision of a Multilayer Perceptron Artificial Neural Network previously developed by the authors. Results of a case study show a better estimation when using voltage magnitude and angle measurements in the input data set.

Key words: artificial neural networks; electrical networks; state estimation; phasor measurement units.

1. Introduction

The advance of electricity markets in Europe, with the objective of creating a unified European market, has led to the Price Coupling of Regions (PCR) initiative, in which seven major electricity exchanges serving 75% of the whole European demand adhered to the use of a common algorithm to compute

* Corresponding author: *e-mail*: ovidiuivanov@tuiasi.ro

electricity prices and manage cross-border capacity on a day-ahead basis (APX, 2015). Since Q4 2014, Romania, Hungary, Slovakia and Czech Republic are part of the 4M Market Coupling (4M-MC) project, which operates under the PCR rules, coordinated by OTE (Czech Republic) and EPEX SPOT (France and Germany), but separately from the Western European market (OPCOM, 2014).

The joint operation of markets requires the monitoring and management of Wide Area electrical Networks (WAN). The preferred tools used by system operators are the State Estimation (SE) algorithms, with measurements provided by high performance Intelligent Electronic Devices (IEDs) such as Phasor Measurement Units (PMUs). Using the GPS clock as common reference, PMUs can provide voltage and current phasor measurements synchronized across wide geographical areas in transmission systems (Chakhchoukh *et.al*, 2014), but the development of small scale Distributed Generation raises the need of deploying such equipment also at the distribution level (Sanchez-Ayala *et al.*, 2013), (Tlusty *et al.*, 2013). While the classic approaches use system-wide estimation, distributed state estimation (DSE) is also possible, where local SE is made in network sections, then an aggregator unifies the results (Shuangshuang *et.al*, 2012), (Qinghua *et.al*, 2009). Attempts were made to replace the classical SE algorithms with Artificial Neural Networks (Kumar, 2012), (Luitel, 2009). The authors proposed such an algorithm in (Ivanov, 2014), which tests the possibility of using Multilayer Percepron (MLP) ANNs as a replacement for load flow or SE algorithms, and study cases performed on a 110 kV distribution network showed promising results. The aim of this paper is to investigate the behavior of the ANN estimator when adding simulated PMU measurements in the input data set. The expected outcome is a better estimation precision.

2. State Estimation

State estimation (SE) algorithms are used by power grid operations to replace load flow algorithms. The most known implementation of a SE algorithm uses the Weighted Least-Squares (WLS) approach (Schweppe, 1970), which computes the state variables (bus voltage magnitudes and angles) based on measurements taken in real time from the monitored system and pseudo-measurements (measurements obtained previously, but considered accurate).

If $[\underline{z}]$ denotes a measurement vector, which can include bus voltages, branch power flows or bus powers, $[\underline{x}]$ is the state vector that needs to be computed, and for each known measurement z_i, a h_i function dependent on $[\underline{x}]$ can be written, then the WLS objective is to minimize the square difference between the measured and the computed value for all measurements

$$J([x]) = \left[[z] - h([x])\right]^T W\left[[z] - h([x])\right], \qquad (1)$$

where W is a diagonal matrix with weights or confidence degrees for each measurement.

This equation is solved in an iterative process, using Newton linearization:

$$G(x)[\Delta x]^k = H^T([x]^k)R^{-1}[\Delta z]^k. \quad (2)$$

In eq. (1), W is a diagonal matrix of weights describing the confidence degree of measurements. In eq. (2), H is the Jacobian matrix, $G = H^T([x]^k) \times \times R^{-1}H([x]^k)$ is the gain matrix and $[z]-h([x]^k)\overset{\text{not}}{=}[\Delta z]^k$ is the vector of errors between measured and computed values.

From eq. (2), the corrections for the voltage approximation are computed, which are used to update the state variables and bring them closer to the optimal solution. The algorithm stops when $[\Delta z]^k$ falls under a given threshold, which indicates that the voltage approximations are close within the desired precision to true voltage values.

The WLS SE algorithm, while being simple and with proven reliability, has some shortcomings. First of all, it requires the knowledge of electrical parameters for all relevant elements (branches, transformers) from the monitored system, in order to compute the Jacobian matrix. Also, while the main source of measurements for SE algorithms remains the SCADA/EMS system, voltage angle measurements cannot be used, because the 1…2 min. sampling rate would induce heavy errors, compromising the estimation results. This shortcoming is lately addressed by using GPS synchronized measurements provided by PMUs Finally, while in transmission systems there are sufficient measurements available and the network configuration is well known, this is not the case in distribution systems, where load measurements are often limited to values metered in substations, and cable routes and parameters are frequently unknown, especially in old networks.

3. Artificial Neural Networks and the Proposed ANN SE Algorithm

For addressing the inconveniences described above, the authors proposed in (Ivanov, 2014) an ANN based state estimation algorithm.

Artificial neural networks are computer algorithms that mimic the architecture and operation of living brains. ANNs are formed by interconnecting neurons which form synapses using weights and process information according to specific activation functions. The unit neuron receives a fraction w of the input value x, which is amplified by a bias b and processed through the activation function F, obtaining the output y. For a neuron with multiple inputs, its equation can be written as:

$$y = F(n) = F\left(\sum_j w_j x_j + b\right). \quad (3)$$

The neurons from intermediate layers can have multiple outputs as well. The interconnection layouts, number of layers and neuron activation functions define several types of ANNs, all requiring two stages of use: training and generalization. In the training stage, the ANN needs to process a data set in order to learn the problem that it needs to solve. The learning process consists in updating the neurons' weight and biases, until a stopping criterion is met.

In the generalization stage, the ANN is able to compute reasonably accurate solutions for new input data, which were not used for training.

The Multilayer Perceptron is well known for its approximation capabilities, even with data of uncertain precision. Trained with discrete input-output pairs *x – f(x),* it is able to discover the relation, or tendency, or mathematical dependence which exists between the inputs and the outputs, provided that two conditions are met: (1) the training data set is sufficient in size, and (2) the training data set is relevant and covers the entire range for which the problem needs to be solved in the generalization stage.

Various training algorithms werte developed. For instance, the Matlab Neural Network Toolbox provides 12 (Matlab, 2012). In this paper, the Resilient Backpropagation algorithm RProp (Riedmiller, 1993) was used. If the standard backpropagation algorithm (Rumelhart *et.al.*, 1986) updates the neuron biases and weights by computing the error E between the desired and obtained outputs and computing the derivative of the error affected by a learning rate η:

$$E = \frac{1}{2}\sum_{m=1}^{M}\left\| d^{(m)} - o^{(m)} \right\|^2, \tag{4}$$

$$w_{ij}^{t+1} = w_{ij}^{t} - \eta \left.\frac{\partial E}{\partial w_{ij}}\right|_{w_{ij}^{t}}, \tag{5}$$

RProp uses instead the error sign adapting the weigts at each $t+1$ cycle

$$w_{ij}^{t+1} = w_{ij}^{t} + \Delta w_{ij}^{t+1} \tag{6}$$

using δ_{ij} coefficients which change their values based on the sign of the error derivative:

$$\delta_{ij}^{t+1} = \begin{cases} \eta^{+}\delta_{ij}^{t} & \text{if } \dfrac{\partial E^{t+1}}{\partial w_{ij}} \cdot \dfrac{\partial E^{t}}{\partial w_{ij}} > 0, \\ \eta^{-}\delta_{ij}^{t} & \text{if } \dfrac{\partial E^{t+1}}{\partial w_{ij}} \cdot \dfrac{\partial E^{t}}{\partial w_{ij}} < 0, \\ \delta_{ij}^{t} & \text{if } \dfrac{\partial E^{t+1}}{\partial w_{ij}} \cdot \dfrac{\partial E^{t}}{\partial w_{ij}} = 0, \end{cases} \tag{7}$$

giving the weight updates:

$$\Delta w_{ij}^{t+1}=\begin{cases}-\operatorname{sign}\left[\dfrac{\partial E^{t+1}}{\partial w_{ij}}\right]\delta_{ij}^{t+1} & \text{if } \dfrac{\partial E^{t+1}}{\partial w_{ij}}\cdot\dfrac{\partial E^{t}}{\partial w_{ij}}\geq 0,\\ -\operatorname{sign}\left[\Delta w_{ij}^{t}\right]\delta_{ij}^{t+1} & \text{if } \dfrac{\partial E^{t+1}}{\partial w_{ij}}\cdot\dfrac{\partial E^{t}}{\partial w_{ij}}<0.\end{cases} \tag{8}$$

This approach ensures better convergence than backpropagation, with less memory usage than other training algorithms such as Levenberg-Marquardt.

The proposed ANN estimator exploits the approximation capabilities of the Multilayer Perceptron and eliminates the need of knowing the electrical parameters and physical configuration of the analyzed electrical network. Also, it replaces the iterative computation with a single forward propagation through the MLP and, as it will be seen in the case study section, can work with less data than required by a classic WLS SE algorithm. A comparison between the two approaches is given in Fig. 1.

In an electrical system, when its operating configuration does not change, the bus voltages are directly related to the load level:

$$\text{voltages} = f(\text{bus loads}). \tag{9}$$

Since the WLS algorithm uses as state variables bus voltages and as inputs measurements which themselves are input to load flow calculations or auxiliary variables computed by load flow algorithms, then eq. (9) can be rewritten:

$$\text{state variables} = g\ (\text{input measurements}), \tag{10}$$

and a MLP ANN can be trained to approximate g using input-output pairs which can be measurements archived by SCADA or simulated with load flow calculations (Fig. 2). For the proposed SE algorithm, selected branch active and

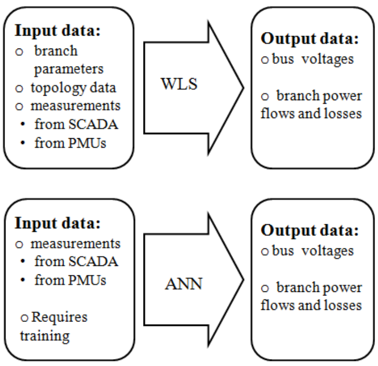


Fig. 1 – A comparison between the WLS and the ANN SE algorithms.

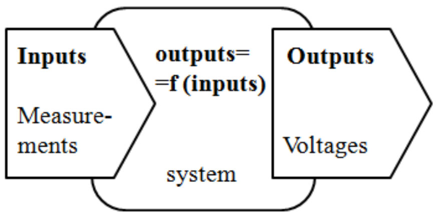


Fig. 2 – The MLP learning principle for the SE problem.

reactive power flows were used as inputs, while the bus voltages in magnitude and angle were used as outputs. In order to simulate PMU measurements, the voltage measurements corresponding to three buses were removed from the output data and inserted into the input data set. MLPs were trained in both scenarios, and their results were compared to see if using the PMU measurements improves the estimation. The data needed for training was obtained by generating randomly a number of load scenarios for a test network, for which the corresponding voltages and branch power flows were computed with a load flow algorithm and then used to build input-output pairs for the MLP. After training, the ANN estimator was tested with another set of randomly generated inputs, and then with a real loading scenario known for the test electrical system. The following performance indices were used, for voltage magnitude and angle:

a) the average percentage bus estimation error:

$$APE = \underset{\text{no.of buses}}{\text{average}}\left[\underset{\text{no.of test scenarios}}{\text{average}}\left(\frac{\text{voltage}_{\text{loadflow}} - \text{voltage}_{\text{MLPSE}}}{\text{voltage}_{\text{loadflow}}}\cdot 100\right)\right], \quad (11)$$

b) the maximum average bus estimation error:

$$AMPE = \underset{\text{no.of buses}}{\max}\left[\underset{\text{no.of test scenarios}}{\text{average}}\left(\frac{\text{voltage}_{\text{loadflow}} - \text{voltage}_{\text{MLPSE}}}{\text{voltage}_{\text{loadflow}}}\cdot 100\right)\right], \quad (12)$$

c) the maximum percentage estimation error:

$$MPE = \underset{\text{no.of buses}}{\max}\left[\underset{\text{no.of test scenarios}}{\max}\left(\frac{\text{voltage}_{\text{loadflow}} - \text{voltage}_{\text{MLPSE}}}{\text{voltage}_{\text{loadflow}}}\cdot 100\right)\right]. \quad (13)$$

4. Case Study and Results

The method described above was tested on a simplified version of a real 110 kV meshed distribution system with 33 buses and 49 branches, represented in Fig. 3. Real loading data was available only for a peak hour in a typical summer day. Thus, 24,000 random loading scenarios were generated, with loads from 0% up to 80% of the HV/MV transformers rated power. The Newton-Raphson load flow algorithm was used to compute the bus voltages and branch power flows pairs for training and testing the ANN. 20,000 scenarios were used for training, while 4,000 were reserved for testing. As measurements, both the active and reactive power flows from all branches connected to selected buses in the system were used. Two reference measurement placement options were considered, the first with "low" measurements density, which resulted in a

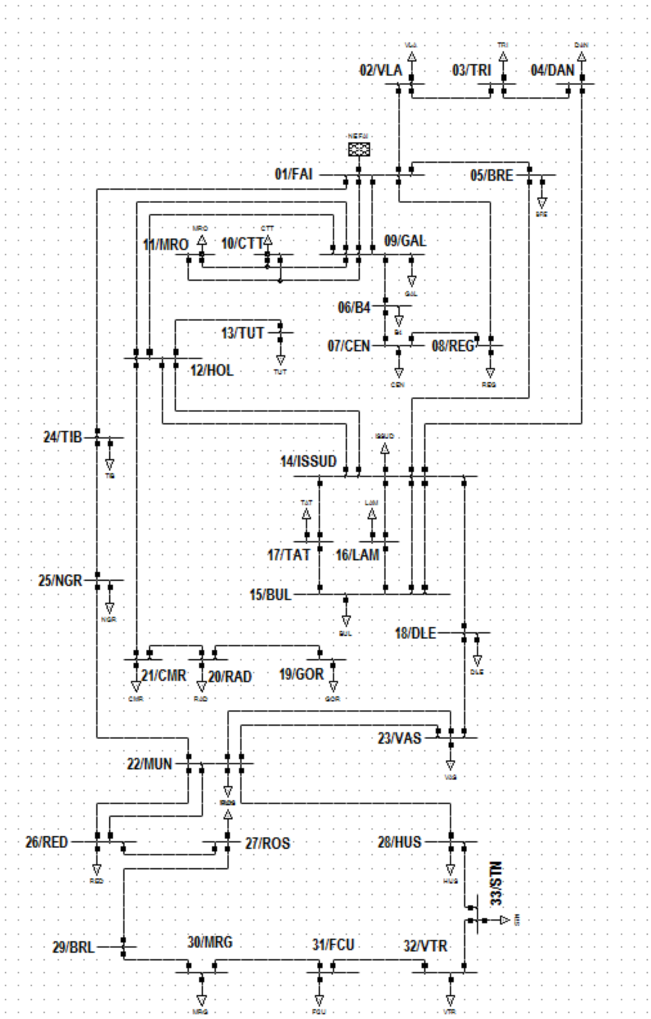
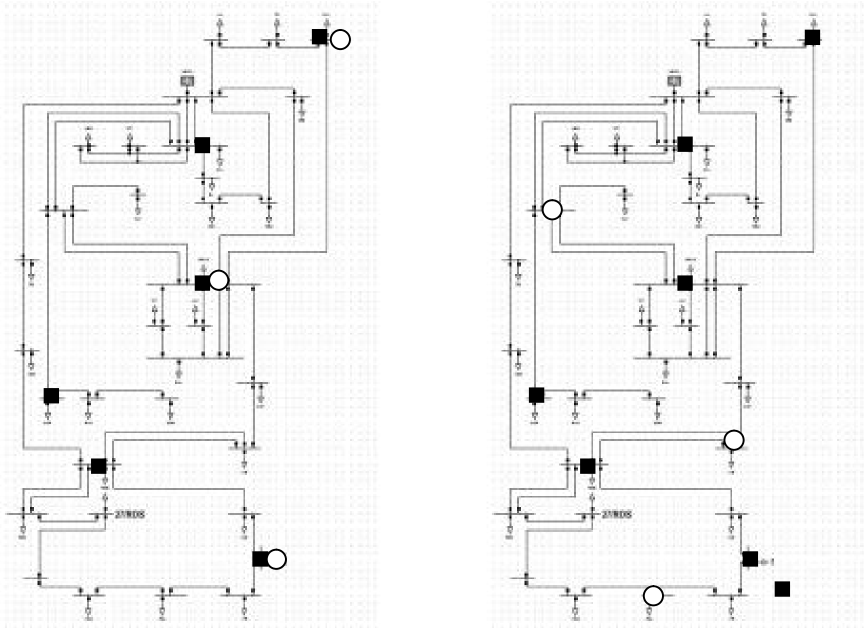


Fig. 3 –The electrical system used in the case study.

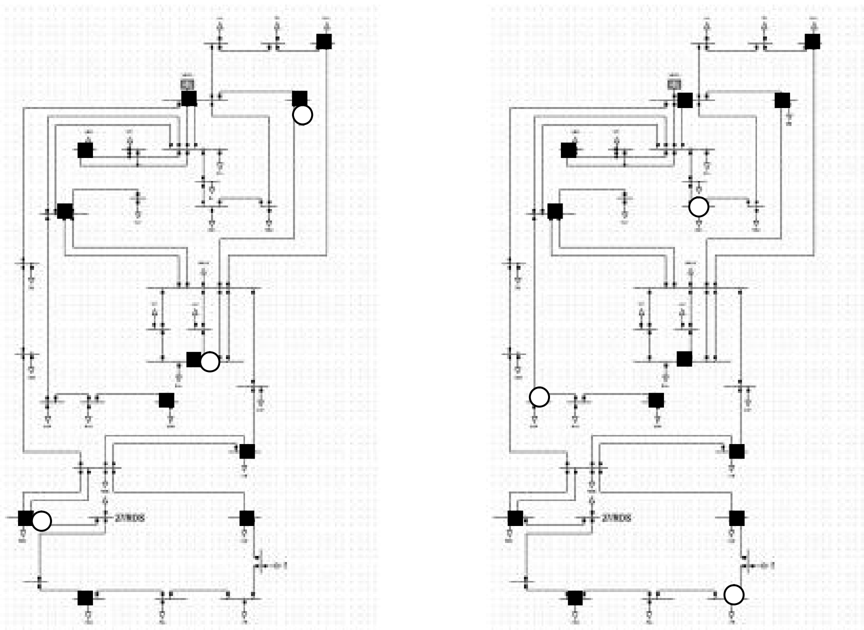
number of 56 measurements from 28 monitored branches found in 7 buses, the other with "high" number of measurements, 66, from 33 monitored branches found in 11 buses. For each reference scenario, two PMU placement scenarios were considered, (a) when three PMUs are placed in buses where branch measurements are known, and (b) when three PMUs are placed in three new buses, thus increasing the number of monitored buses (Figs. 4 and 5). In order to avoid the high angle estimation error seen in (Ivanov, 2014), a voltage reference of 30 degrees was used in the slack bus.

In Figs. 4 and 5, black fill square dots denote buses with classical measurements, and white fill round dots denote PMU measurements.



case (a) case (b)

Fig. 4 – PMU placement for the "low" measurements density scenario.



case (a) case (b)

Fig. 5 – PMU placement for the "high" measurements density scenario

Table 1
Estimation Results, "Low" Measurements Density

	APE		AMPE		MPE	
reference	mag.	angle	mag.	angle	mag.	angle
33 hidden neurons						
min	0.0237	0.0239	0.0326	0.0408	0.1375	0.2071
avg	0.0243	0.0256	0.0335	0.0458	0.1473	0.3088
max	0.0248	0.0275	0.0349	0.0552	0.1585	0.4303
66 hidden neurons						
min	0.0194	0.0250	0.0298	0.0475	0.1339	0.2486
avg	0.0223	0.0263	0.0328	0.0505	0.1507	0.2862
max	0.0235	0.0279	0.0345	0.0545	0.1806	0.3459
132 hidden neurons						
min	0.0138	0.0369	0.0264	0.0638	0.1320	0.3477
avg	0.0157	0.0394	0.0296	0.0705	0.1615	0.4455
max	0.0169	0.0415	0.0356	0.0776	0.2114	0.5058
case (a)	mag.	angle	mag.	angle	mag.	angle
31 hidden neurons						
min	**0.0228**	**0.0225**	**0.0312**	**0.0405**	**0.1258**	**0.2064**
avg	**0.0233**	**0.0242**	**0.0330**	**0.0433**	**0.1418**	**0.2505**
max	**0.0236**	**0.0263**	0.0354	**0.0469**	0.1677	**0.3116**
62 hidden neurons						
min	**0.0190**	**0.0247**	**0.0284**	**0.0459**	0.1358	0.2575
avg	**0.0212**	0.0271	**0.0325**	**0.0500**	0.1553	0.2982
max	**0.0228**	0.0286	0.0373	0.0552	0.1971	0.3491
124 hidden neurons						
min	0.0157	0.0426	0.0303	0.0719	0.1811	0.4167
avg	0.0173	0.0451	0.0332	0.0825	0.2221	0.5206
max	0.0191	0.0467	0.0383	0.0917	0.3383	0.6523
case (b)	mag.	angle	mag.	angle	mag.	angle
31 hidden neurons						
min	**0.0228**	**0.0213**	**0.0315**	**0.0365**	**0.1349**	0.2224
avg	**0.0238**	**0.0230**	0.0341	**0.0400**	**0.1468**	**0.2705**
max	**0.0247**	**0.0245**	0.0364	**0.0423**	0.1621	**0.3832**
62 hidden neurons						
min	**0.0179**	**0.0248**	**0.0275**	**0.0436**	**0.1270**	**0.2147**
avg	**0.0197**	0.0270	**0.0304**	**0.0480**	**0.1431**	**0.2851**
max	**0.0221**	0.0298	**0.0338**	0.0564	**0.1552**	**0.3260**
124 hidden neurons						
min	0.0149	0.0402	**0.0258**	0.0656	0.1728	0.4108
avg	0.0167	0.0441	0.0324	0.0764	0.2024	0.5313
max	0.0200	0.0501	0.0405	0.0887	0.2366	0.7631

Table 2
Estimation Results, "High" Measurements Density

	APE		AMPE		MPE	
reference	mag.	angle	mag.	angle	mag.	angle
33 hidden neurons						
min	0.0236	0.0381	0.0327	0.0767	0.1498	0.3210
avg	0.0247	0.0395	0.0346	0.0780	0.1684	0.3544
max	0.0255	0.0413	0.0370	0.0791	0.1914	0.3903
66 hidden neurons						
min	0.0205	0.0401	0.0298	0.0795	0.1363	0.3590
avg	0.0219	0.0410	0.0351	0.0825	0.1656	0.4077
max	0.0238	0.0423	0.0384	0.0851	0.1766	0.4815
132 hidden neurons						
min	0.0163	0.0508	0.0304	0.0972	0.1672	0.5007
avg	0.0175	0.0535	0.0340	0.1015	0.1946	0.5764
max	0.0191	0.0575	0.0371	0.1106	0.2285	0.7446
case (a)	mag.	angle	mag.	angle	mag.	angle
36 hidden neurons						
min	**0.0231**	**0.0218**	**0.0310**	**0.0333**	**0.1235**	**0.1805**
avg	**0.0240**	**0.0231**	**0.0336**	**0.0352**	**0.1542**	**0.2722**
max	**0.0246**	**0.0254**	0.0393	**0.0386**	**0.1774**	0.4248
72 hidden neurons						
min	**0.0171**	**0.0260**	**0.0284**	**0.0410**	**0.1344**	**0.2527**
avg	**0.0203**	**0.0289**	**0.0316**	**0.0461**	**0.1529**	**0.3197**
max	**0.0229**	**0.0313**	**0.0344**	**0.0540**	0.1855	0.4843
144 hidden neurons						
min	0.0176	**0.0457**	0.0339	**0.0700**	0.1960	**0.4452**
avg	0.0191	**0.0496**	0.0374	**0.0809**	0.2207	**0.5349**
max	0.0204	**0.0552**	0.0419	0.0931	0.2727	**0.6320**
case (b)	mag.	angle	mag.	angle	mag.	angle
36 hidden neurons						
min	**0.0230**	**0.0208**	**0.0321**	**0.0273**	**0.1416**	**0.2217**
avg	**0.0240**	**0.0230**	**0.0339**	**0.0347**	**0.1464**	**0.2722**
max	**0.0245**	**0.0248**	**0.0357**	**0.0395**	**0.1514**	**0.3390**
72 hidden neurons						
min	**0.0170**	**0.0257**	**0.0265**	**0.0404**	**0.1340**	**0.2484**
avg	**0.0203**	**0.0268**	**0.0326**	**0.0434**	**0.1563**	**0.2905**
max	**0.0236**	**0.0296**	**0.0368**	**0.0491**	0.1904	**0.3380**
144 hidden neurons						
min	0.0180	**0.0438**	0.0333	**0.0698**	0.2082	0.4140
avg	0.0195	**0.0489**	0.0382	**0.0800**	0.2296	0.5456
max	0.0209	**0.0524**	0.0496	**0.0906**	0.2652	0.6934

The estimation comparison is given in Tables 1 and 2, when training the MLP with one hidden layer with a size of half, same and double the number of inputs. The results show minimum, average and maximum APEs, AMPEs and MPEs for 10 runs with each MLP configuration. The estimation without PMU measurements is already very accurate for the 4000 test scenarios, below 0.1% average and 0.75% maximum errors, but supplemental PMU measurements have a mixed effect. While estimations with lower number of hidden neurons are usually improved, high hidden neuron count configurations show an increased estimation error. The improvement effect is better in the scenarios with "high" density of measurements, but the differences are minor, which suggests a better behaviour of the MLP estimator compared to the WLS estimator when a limited number of measurements is available.

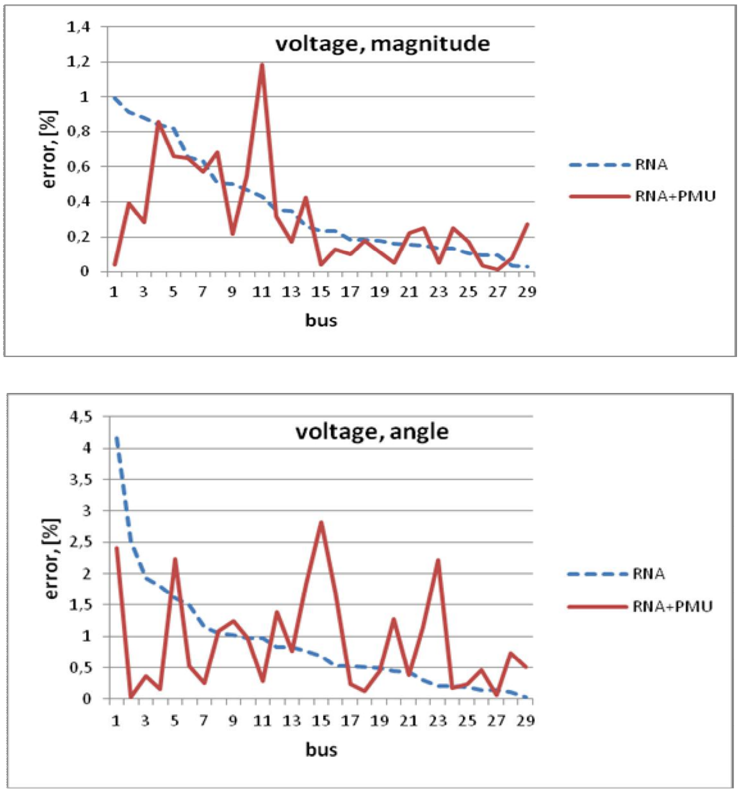


Fig. 6 – Comparison between the voltage estimation error obtained using a 33/36 neurons hidden layer architecture ANN, with classical and classical + PMU measurements (“low density scenario”).

The second test used the real load scenario known for the system, for which input measurements were also computed using Newton-Raphson load flow. Using the best performance configurations found in the first stage, with

33/36 and 66/72 hidden neurons, (b) case, the bus voltages were estimated choosing from the ten tries the one with the lowest APE. The results are presented in Figs. 6 and 7, for voltage and angle and "low" and "high" measurements density. The buses are sorted in the descending APE value for the reference RNA estimation without PMU.

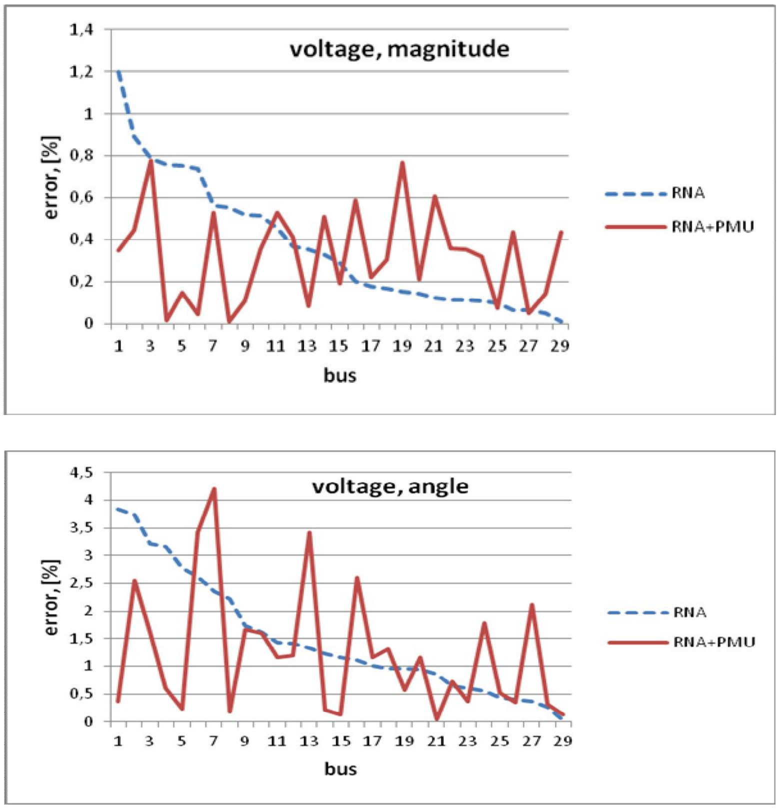


Fig. 7 – Comparison between the voltage estimation error obtained using a 36/72 neurons hidden layer architecture PMU with classical and classical + PMU measurements ("high density scenario").

The estimation precision is improved for some buses and worsened for others, but a tendency to leverage the errors is observed, high errors being improved and low errors being worsened. The estimation precision is lower than in the ransom scenarios test, but with acceptable errors, which do not exceed 1.2% for magnitude and 4.3% for angle in the the ANN+PMU scenario, usually being much lower. However, it should be noted that the measurements were not chosen based on an optimization process, but placed in such a manner that would ensure maximum observability.

4. Conclusions

The study case results suggest that using PMU measurements in the proposed ANN estimator has the potential to improve the estimation results. However, obtaining the best results requires the optimization of the ANN architecture and of the measurements' placement, which demands for additional research. The ANN estimator can be successfully used in small systems which have a known operation configuration, which does not change over time, and it can deliver accurate results even with a limited number of measurements.

REFERENCES

Chakhchoukh Z., Vittal V., Heydt G.T., *PMU Based State Estimation by Integrating Correlation*. IEEE Trans. on Pow. Syst., **29**, *2*, 17-626 (2014).

Ivanov O., Gavrilaş M., *Intelligent Monitoring and Control in Transmission and Distribution Networks*. Proc. of 14th Intl. Conf. OPTIM, May 22-24, 2014, 185-191.

Kumar A., Chakrabarti S., *ANN-based Hybrid State Estimation and Enhanced Visualization of Power Systems*. ISGT IEEE PES, Dec. 1-3, 2011, 78-83.

Luitel B., Venayagamoorthy G.K., *Wide area monitoring in power systems using cellular neural networks*. IEEE CIASG, April 11-15, 2011, 1-8.

Qinghua H., Schulz N.N., Srivastava A.K., Haupt T., *Distributed State Estimation with PMU Using Grid Computing*. IEEE PES 09, July 26-30, 2009, 1-7.

Riedmiller M., Braun H., *A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm*. Proc. IEEE Intenat. Conf. Neur. Netw., **1**, 586-591 (1993).

Rumelhart D., McClelland J.L and the PDP research group (Eds.), *Parallel Distributed Processing*: *Explorations in the Microstructure of Cognition*. Volume **1**. *Foundations*. MIT Press, 1986.

Sanchez-Ayala G., Aguero J.R. , Elizondo D., Lelic M, *Current Trends on Applications of PMUs in distribution systems*, Proc. ISGT IEEE PES 2013, 1-6.

Schweppe F.C., Wildes J., *Power System Static-State Estimation, Part I: Exact Model*, IEEE Trans Power App. and Syst., **PAS89**, 1, 125-130 (1970).

Shuangshuang J,. Chen Y., Rice M., Yan L., *A testbed for deploying distributed state Estimation in Power Grid*. IEEE PES, July 22-26, 2012, 1-7.

Tlusty J. *et al.*, *The Monitoring of Power System Events on Transmission and Distribution Level by the Use of Phasor Measurements Units (PMU).* Proc. 20th Internat. Conf. Exhib. Elect. Distrib. (CIRED), Jun. 8–11 (2009).

* * * OPCOM, 4M Market Coupling, Prezentare generală a proiectului, Aprilie 29, 2014, http://www.opcom.ro/, accessed on March 2015.

* * * MATLAB Neural Networks Toolbox Guide, Mathworks, Inc., 2012.

* * * *Successful Operation of PCR Solution: Power Exchanges Focus on Pan-European Power Market*. APX, http://www.apxgroup.com/press-releases/, accessed on March 2015.

ÎMBUNĂTĂŢIREA PRECIZIEI UNUI ESTIMATOR DE STARE BAZAT PE REŢELE NEURONALE ARTIFICIALE, FOLOSIND MĂSURĂTORI FAZORIALE

(Rezumat)

Fiind capabile să furnizeze măsurători de fazori de tensiune sincronizate la nivelul unui întreg sistem electroenergetic, dispozitivele pentru măsurări fazoriale au îmbunătăţit precizia algoritmelor clasice de estimare a stării utilizate la monitorizarea şi controlul în timp real al reţelelor electrice. Această lucrare testează oportunitatea utilizării măsurărilor fazoriale ca date de intrare într-un estimator de stare bazat pe o reţea neuronală artificială de tip perceptron multistrat, dezvoltat de către autori într-o lucrare anterioară. Rezultatele obţinute într-un studiu de caz arată că precizia estimării se îmbunătăţeşte atunci când în setul de măsurători folosit ca date de intrare sunt incluse şi măsurări de tensiune, modul şi fază, provenite de la dispozitive pentru măsurări fazoriale.