Available online at www.sciencedirect.com
ScienceDirect
Journal of Electrical Systems and Information Technology xxx (2017) xxx–xxx
www.elsevier.com/locate/jesit

Particle Swarm Optimization trained recurrent neural network for voltage instability prediction

Amr M. Ibrahim [a,*], Noha H. El-Amary [b]

[a] Electrical Power and Machines Engineering Dept., Ain Shams University, Cairo, Egypt
[b] Electrical Power and Control Engineering Dept., Arab Academy for Science and Technology, Egypt

Received 9 February 2017; received in revised form 30 April 2017; accepted 1 May 2017

Abstract

Voltage instability is considered as a major problem that faces the power systems during its operation. Voltage instability prediction is necessary for avoiding voltage collapse. This paper investigates the performance of recurrent neural network (RNN) in voltage instability prediction. A recurrent neural network trained with Particle Swarm Optimization (PSO) is proposed in this paper. The proposed method is examined on 14-bus and 30-bus IEEE standard systems. These systems are simulated using MATLAB/Power System Toolbox program. Also, a detailed comparison between PSO algorithm and Backpropagation (BP) algorithm is discussed. The results proved the effectiveness of the proposed method.
© 2017 Electronics Research Institute (ERI). Production and hosting by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

Keywords: Backpropagation algorithm; Particle Swarm Optimization technique; Recurrent neural network; Voltage instability predictor; Voltage stability

1. Introduction

Voltage instability problems play a great role in power systems planning and operation. Nowadays, power systems are operated near to their capacity borders because of the environmental and the economic aspects. Preserving a steady-state process of the power system is a vital matter. Therefore, it is highly suggested to monitor the voltage stability in the electrical networks.

Many researchers study the power system voltage stability from different points of view. The excessive loading of transmission lines is one of the essential factors causing voltage instability.

The research progress is started with defining and identifying the voltage stability circumstances, factors, indices and problems. Some articles are performed to improve and enhance the voltage stability index (Khatua and Yadav, 2015; Mohandas et al., 2015; Murty and Kumar, 2015; Angelim and Affonso, 2016). Also, many techniques are used in voltage instability detection (Nakawiro and Erlich, 2008; El-Amary et al., 2008; Kamalasadan, 1998).

Phasor measurement units (PMUs) are vital devices in monitoring and collecting data from the electrical grid that helps in many applications and problems (Zhang et al., 2009; Lin et al., 2004; Laverty et al., 2009). The correlation

* Corresponding author.
E-mail address: amrmohamedhassan@yahoo.com (A.M. Ibrahim).

http://dx.doi.org/10.1016/j.jesit.2017.05.001
2314-7172/© 2017 Electronics Research Institute (ERI). Production and hosting by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

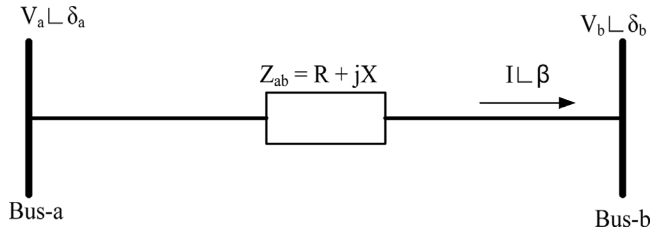

Fig. 1. Two-bus system.

between PMUs and GPS provide high capability to synchronize the measurable data of voltage and current signals from several positions that located at the electrical grid (Phadke, 2002; Depablos et al., 2004). This work employs PMUs to give accurate measurements of voltage phase angle of each bus in the studied networks.

Early prediction of voltage instability is helped in avoiding voltage collapse of the power systems. This is can be achieved by monitoring the changes in phase difference between each two consecutive buses of the electrical networks instead of focusing on the voltage magnitudes. This is because voltage magnitudes, at certain loading conditions, may not detect voltage collapse in an early phase.

In this article, voltage instability prediction is accomplished by recurrent neural network (RNN). Recurrent neural network is a powerful learning algorithm. Several studies on electrical power systems using RNN have been conducted, including applications in induction motor speed estimation (Goedtel et al., 2006), islanding detection for distributed generation (Bayrak, 2009), wind speed and power forecasting (Barbounis and Theocharis et al., 2006), and design of power system stabilizer (Chen and Chen, 2006).

Recurrent neural networks consist of different layers of neurons organized in input, output and hidden layers. The neurons are connected to each other by synaptic weights. During the learning process, the network weights are adapted until a minimum error is achieved. Neural networks training process took the attention of many researchers in the last few years (Janson and Frenzel, 1993; Alpaydin et al., 2002; Mendes and Cortez et al., 2002; Salerno, 1997; Gudise and Venayagamoorthy, 2003).

The application of various population-based search algorithms in the training of neural networks has been enabled due to a recent growth in evolutionary computation mechanisms.

One of the most famous evolutionary computation techniques is Particle Swarm Optimization (PSO) which based on swarm intelligence. The PSO has been found to be fast and robust in solving many nonlinear optimization problems (Angeline, 1998; Clerc and Kennedy, 2002; Trelea, 2003). One of the first implementation of PSO was that training neural networks (Kennedy and Eberhart, 2001). The comparative simplicity of PSO in training neural networks is the main advantage of its usage over other optimization algorithms.

This paper represents a comprehensive study using RNN for voltage instability prediction. PSO algorithm is proposed to train a recurrent neural network for prediction of voltage instability. To prove the effectiveness of the proposed method a comparison has been established with a method using Backpropagation (BP) algorithm in training RNN.

The rest of the paper is organized as follows: Section 2 presents an explanation about the voltage stability. Section 3 describes in details two algorithms used in training recurrent neural network (RNN): Backpropagation (BP) training algorithm and Particle Swarm Optimization (PSO) algorithm. Section 4 evaluates the results of voltage instability predictor when applied on 14-bus and 30-bus IEEE standard systems, also this section investigates the performance of the training algorithms. Finally, Section 5 introduces the extracted conclusion.

2. Voltage stability

Voltage stability is defined as the system ability to supply appropriate reactive power support, in order to preserve voltage magnitudes of the load within particular operating boundaries for both balanced and transient conditions. From the load point of view, voltage stability can be defined as the load ability to give more power as the loading is increased without voltage dips beyond the limits.

This research follows the strategy of voltage instability prediction through monitoring the change in phase difference between each two consecutive buses of the power system. Fig. 1 shows the voltage phasors $V_a \angle \delta_a$ and $V_b \angle \delta_b$ of two

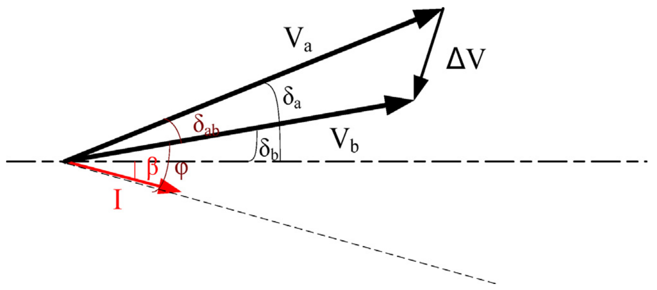

Fig. 2. Two-bus system V–I Phasor diagram.

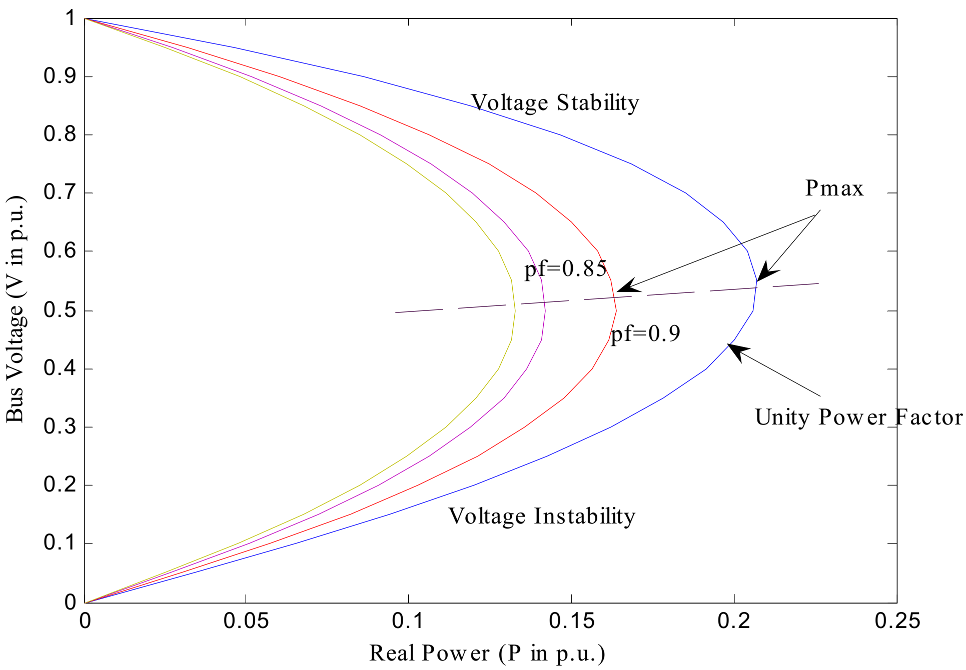

Fig. 3. Two-bus system P–V curves for different power factors.

buses from an electrical network. The transmission line impedance $Z_{ab} = R + jX$ connects the two buses *a* and *b*. The current phasor $I\llcorner\beta$ flows from bus-*a* to bus-*b*. Fig. 2 illustrates the phasor diagram of the two-bus system, in which

$$V_b\llcorner\delta_b = V_a\llcorner\delta_a - \Delta V \tag{1}$$

$$\Delta V = I\llcorner\beta Z_{ab} \tag{2}$$

(Assume no charging current)

The voltage stability mode can be determined using the classical relations and curves of real power versus voltage (P–V) and reactive power versus voltage (Q–V). The curve between the real power and the voltage magnitude (P–V curve) is shown in Fig. 3. The relation between the real power and the voltage magnitude (P–V) is derived from the system transmitted real and reactive power equations, which are illustrated as follows:

$$P_{ba} = V_b V_a Y_{ab} \cos(\delta_a - \delta_b + \theta_{ab}) - {V_b}^2 Y_{ab} \cos(\theta_{ab}) \tag{3}$$

$$Q_{ba} = V_b V_a Y_{ab} \sin(\delta_a - \delta_b + \theta_{ab}) - {V_b}^2 (Y_{ab} \sin(\theta_{ab}) + B_{capab}) \tag{4}$$

where:

Y_{ab}: Line a-b admittance magnitude (between bus-a and bus-b).

θ_{ab}: Line: a-b admittance angle.
δ_a: Bus-a voltage angle.
δ_b: Bus-b voltage angle.
B_{capab}: The total line charging susceptance.

Since the total power ($S = P + jQ$) is transmitted from bus-a to bus-b, so a direct relation between P, Q and V_b for certain V_a, can be expressed in (5). For different power factor and per unit V_a values, P–V and Q–V curves can be determined. In P–V curves V_b varies from zero to $V_b = V_a$. It results in different power values which started from zero increasing to P_{max} (the curve nose) then decreasing again to zero as illustrated in Fig. 3.

$$V_b^{\,4} + (2PR + 2QX - V_a^{\,2})\, V_b^{\,2} = -(R^2 + X^2)\,(P^2 + Q^2) \tag{5}$$

A simulation of real-time voltage instability alarming predictor using RNN depending on PMUs' readings is discussed. A MATLAB program is used to accomplish this prediction, this program is consists of some steps. First, the system data and parameters are used as inputs to the program. Then, a system of P–V curves for different power factors is designed. From P–V curves, the maximum power for voltage magnitude variation of 5% and 15% reduction is determined. Subsequently, the phase difference between each two consecutive buses is calculated. The protection apparatus may operate at 5% voltage reduction while at 15% voltage reduction the voltage is collapsed and the proposed predictor must give "trip" decision. Different loading cases at different positions of the system are investigated. Several loading cases are provided through varying the active power and reactive power. Hence, the voltage angles measured by PMUs, active power, and reactive power are converted into a look-up table which is used to train the RNN. The RNN gives an output of 1, 0, or −1 for each bus in the electrical network where 1 means "stable", 0 means "alarm" where the protection equipment may operate and −1 means "trip". The flow chart of the MATLAB program which is used in this paper is presented in Fig. 4.

Voltage instability predictor employing recurrent neural network (RNN) is a powerful tool in protecting the electrical grid from voltage collapse. In this article, suitable preparations are considered to design an appropriate predictor of voltage instability. The main advantage of the proposed predictor is that, it can predict the voltage instability of the whole power system buses in the same time. The steps of the preparation of the data as inputs and outputs for RNN can be summarized as follows:

(1) Building the suggested study systems (14-bus and 30-bus IEEE standard systems), taking into consideration the placement of PMUs where complete observability is achieved.
(2) Different loading cases at different locations of the system are applied and studied in this stage. The load changing is applied through two strategies; the first strategy is by increasing the active power while keeping the reactive power fixed. The second strategy is by increasing the reactive power while keeping the active power fixed.
(3) Designing the system P–V curves based on different power factors.
(4) From P–V curve, determining the maximum power that corresponds to voltage variation of 5% and 15% below collapse point.
(5) According to the predetermined maximum power and from PMUs readings of the phase difference between each two consecutive buses at different loading conditions, a load-flow analysis, based on the equation illustrated in Section 2, is created. This analysis provides the following data at each bus:
    - Voltage angles.
    - Load active power.
    - Load reactive power.
    - Generation active power.
    - Generation reactive power.

    These parameters are used as inputs to the RNN.
(6) Constructing a look up table based on the load-flow data. The look-up table data, after the data is normalized, is used as inputs to the RNN. Thus, the RNN are contains five inputs.
(7) The output layer consists of one output for each system bus which indicates its status. The output status of the RNN is classified into three categories according to the input loading cases as follows:

"1" means "stable",
"0" means "alarm" where the protection equipment may operate and,

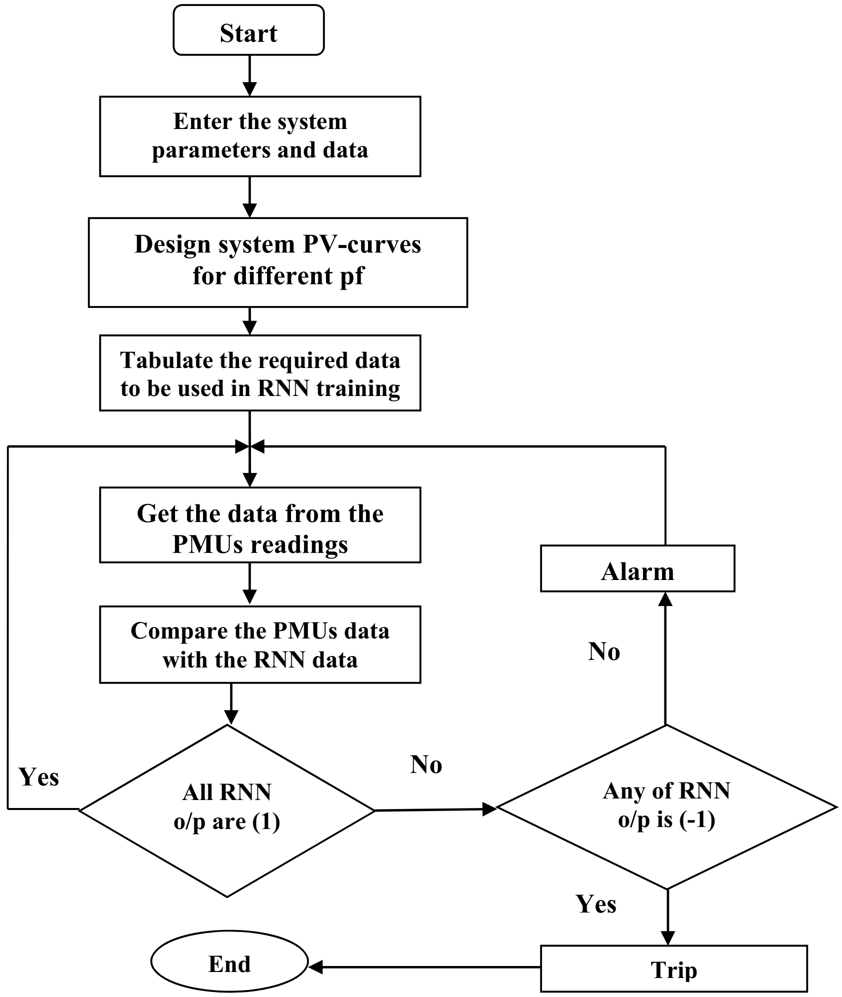

Fig. 4. A flow chart of MATLAB program of a simulation of voltage instability alarming predictor.

"−1" means "trip" where the voltage is collapsed and the proposed predictor must give "trip" decision.

3. Recurrent neural network

Recurrent neural networks (RNNs) have feedback loops from the network output $z_1(t),\ldots, z_{nj}(t)$ to the network inputs $x(t)$. The existence of these loops has a great influence on the learning ability of the network. For a new input, the output is determined and fed back to set the modified input. This operation is continued till the output becomes constant. The steps of this process are as follow:

$$a_i(t) = \Sigma \varphi_{ji} x_i(t) + \Sigma \sigma_{ji} h_i(t-1),\ j = 1, \ldots., n_H \quad (6)$$

$$h_i(t) = F(a_i(t)),\ j = 1, \ldots, n_H \quad (7)$$

$$b_j(t) = \Sigma a_{ji} h_i(t),\ j = 1, \ldots, n_J \quad (8)$$

$$z_j(t) = G(b_j(t)),\ j = 1, \ldots, n_J \quad (9)$$

where φ_{ji}, σ_{ji} and a_{ji} are the weights. The network is composed of interconnected processing elements named neurons. In the learning process, the hidden neurons which receive the input vector $x(t)$ make a computed modifications to the weights of the network. F represents a non- linear transformation that exists between the neurons. The hidden neurons number n_H is established by trial and error.

Please cite this article in press as: Ibrahim, A.M., El-Amary, N.H., Particle Swarm Optimization trained recurrent neural network for voltage instability prediction. J. Electr. Syst. Inform. Technol. (2017), http://dx.doi.org/10.1016/j.jesit.2017.05.001

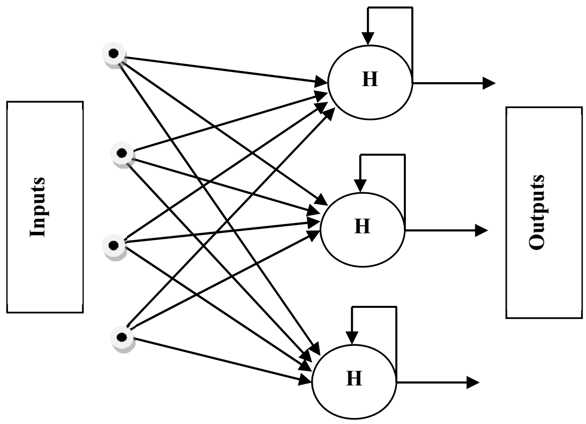

Fig. 5. Recurrent neural network architecture.

A simple construction of an RNN is given in Fig. 5. This figure illustrates that the neurons in the middle layer obtain input values from both the input neurons and the hidden neurons.

In this paper, recurrent neural networks (RNNs) are composed of one input layer, two hidden layers, and one output layer. The input layer of both RNNs consists of five nodes as follows voltage angles, load active power, load reactive power, generation active power and generation reactive power.

3.1. Backpropagation (BP) training algorithm

Backpropagation (BP) algorithm is a supervised learning technique based on error correction learning rule. Training of recurrent neural networks is one of the most important applications of BP. The desirable network's weights that give a minimum error can be modified by a random gradient descent BP algorithm. This algorithm calculates the gradient of the error network with respect to the network newly weights (Xiao et al., 2007).

Slow convergence and being trapped in local minima are the most known disadvantages of the backprobagation algorithm.

3.2. Particle Swarm Optimization (PSO) training algorithm

The creation of a group of networks by choosing random weights is the core of using PSO in training the recurrent neural network. Each network is named particle which is a candidate solution.

Each candidate point (particle) is capable of retaining its velocity and position. This information about the state of each particle is shared within the network. In search space, all the particles fly towards the best solutions known where each particle makes a communication channel with the other particles so each particle recognizes the position of the best solution.

The training operation is initiated by selecting random numbers to the weight parameters and the velocity vectors $v(0)$. Then the training data is supplied to each network. The mean squared error is calculated to each network and compared with the last best value (pbest) error. Update pbest error and retain the existing weights as the pbest weights if the current error is lower than the pbest error.

Locate the minimum calculated error in the swarm network and compare it with the global best (gbest) error. Update the global best error and save the corresponding weights as the gbest weights if the minimum error is lower than gbest error.

Change the velocity particle with

$$v_i(\mathrm{t}+1) = wv(t) + c_1\rho_1(\mathrm{W}_{\mathrm{pbest}} - \mathrm{W}_\mathrm{i}) + \mathrm{c}_2\rho_2(\mathrm{W}_{\mathrm{gbest}} - \mathrm{W}_\mathrm{i}) \quad (10)$$

where

w: the inertia parameter,

c_1 and c_2: the acceleration parameters,

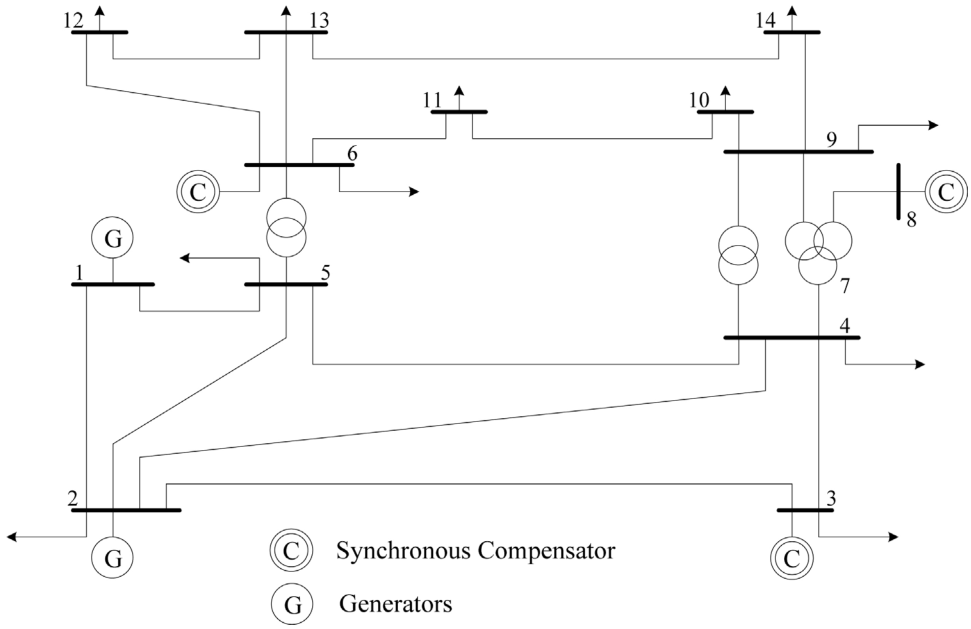

Fig. 6. 14-bus IEEE standard system.

ρ_1 and ρ_2:distributed random numbers [0,1].
W_{pbest}, W_{gbest}, and W_i: the weight vectors of pbest and gbest, respectively.
Wi: the weight vectors of the existing particle
This cycle of iterations is continued till the error of gbest is lower than the required value (Xiao et al., 2007).

4. Simulation results

The voltage instability early predictor using RNN technique is applied to 14-bus and 30-bus IEEE standard systems. Various loading cases at different positions are utilized for the applied systems. The voltage phase angle of each bus is recorded by PMUs. Active power and reactive power of each bus are collected under different loading case studies. The load changing is applied through varying the active power and reactive power with different power factors through two strategies that are previously mentioned in Section 2. Voltage phase angle, active power and reactive power of each bus in the applied systems are used as inputs to the RNN. The RNN's outputs are indicating the status of each bus; stable (1), alarm (0) or trip (−1). The proposed RNN has one input layer, two hidden layers and one output layer.

4.1. The 14-bus IEEE standard system

The proposed detector is utilized to IEEE 14-bus system. The studied 14-bus IEEE standard system is shown in Fig. 6. The load flow results when the load active power is changed to 165 MW at bus 9 is illustrated in Table 1.

1357 study cases are normalized and tabulated. 900 of the study cases are used in the RNN training, the rest of the study cases are used in the RNN testing process. In the training process using BP algorithm, the number of neurons in each layer of the two hidden layers is 10 and 12 neurons respectively, while 15 and 20 neurons are used for each of the two hidden layers in utilizing PSO. The PSO parameters are: $w = 0.8$, $c_1 = c_2 = 1.5$ and number of particles is set to be 35. Samples of the testing cases are illustrated in Table 2; indicating the output results of using BP and PSO in training RNN. Results clarified that the proposed predictor operates effectively with different loading cases.

Based on the results, the overall accuracy of the RNN for the 14-bus IEEE standard system using BP in the testing process is 94.2% while using PSO gives an accuracy of 96.7%.

Table 1
The load-flow results when P_{Load} is changed to 165 MW at bus 9.

Bus number	Voltage angle (degree)	Load MW	Load MVAR	Generation MW	Generation MVAR
1	0	30.38	17.78	297.294	26.826
2	−4.592	0	0	232	67.289
3	−19.186	131.88	26.6	0	89.39
4	−17.232	66.92	10	0	0
5	−14.457	10.64	2.24	0	0
6	−27.11	15.68	10.5	0	67.263
7	−29.959	0	0	0	0
8	−29.962	0	0	0	68.548
9	−36.888	165	23.24	0	0
10	−35.195	12.6	8.12	0	0
11	−30.564	4.9	2.52	0	0
12	−28.427	8.54	2.24	0	0
13	−28.616	18.9	8.12	0	0
14	−40.203	20.86	7	0	0

Table 2
Sample of the testing results for 14-bus IEEE standard system.

Bus number	Voltage angle (Degree)	Load MW	Load MVAR	Generation MW	Generation MVAR	Target	O/P of BP	O/P of PSO
Case 1: P_{Load} is changed to 535 MW at bus 5								
1	0.9954	0.0478	0.6684	0.8587	0.2033	0	0	0
2	0.7783	0	0	0.2305	0.6719	1	1	1
3	0.4966	0.2077	1	0	0.2882	1	1	1
4	0.4594	0.1054	0.3759	0	0.0411	0	1	1
5	0.4174	0.8425	0.0842	0	0.0411	1	1	1
6	0.3106	0.0247	0.3947	0	0.2671	0	0	0
7	0.3760	0	0	0	0.0411	1	1	1
8	0.3759	0	0	0	0.1635	1	1	1
9	0.3347	0.0650	0.8737	0	0.0411	1	1	1
10	0.3236	0.0198	0.3053	0	0.0411	1	1	1
11	0.3122	0.0077	0.0947	0	0.0411	1	1	1
12	0.2987	0.0135	0.0842	0	0.0411	1	1	1
13	0.3014	0.0298	0.3052	0	0.0411	1	0	1
14	0.2926	0.0329	0.2632	0	0.0411	1	1	1
Case 2: Q_{Load} is changed to 15 MVAR at bus 10								
1	0.9954	0.0478	0.6684	0.2032	0.0939	1	1	1
2	0.9630	0	0	0.2305	0.0711	1	1	1
3	0.7967	0.2077	1	0	0.1549	1	1	1
4	0.8323	0.1054	0.3759	0	0.0411	1	1	1
5	0.8586	0.0168	0.0842	0	0.0411	1	−1	−1
6	0.7300	0.0247	0.3947	0	0.1205	1	1	1
7	0.7396	0	0	0	0.0411	1	1	1
8	0.7396	0	0	0	0.1157	1	1	1
9	0.6900	0.0650	0.8737	0	0.0411	0	0	0
10	0.6683	0.0945	0.6224	0	0.0411	1	1	1
11	0.7016	0.0078	0.0947	0	0.0411	−1	−1	−1
12	0.7150	0.0135	0.0842	0	0.0411	1	1	1
13	0.7141	0.0298	0.3053	0	0.0411	1	1	1
14	0.6515	0.0329	0.2632	0	0.0411	1	1	1

Please cite this article in press as: Ibrahim, A.M., El-Amary, N.H., Particle Swarm Optimization trained recurrent neural network for voltage instability prediction. J. Electr. Syst. Inform. Technol. (2017), http://dx.doi.org/10.1016/j.jesit.2017.05.001

4.2. The 30-bus IEEE standard system

The proposed instability predictor is utilized also to IEEE 30-bus standard system. The Studied 30-bus IEEE standard system is shown in Fig. 7. The load flow results when the load active power is changed to 135 MW at bus 15 is illustrated in Table 3.

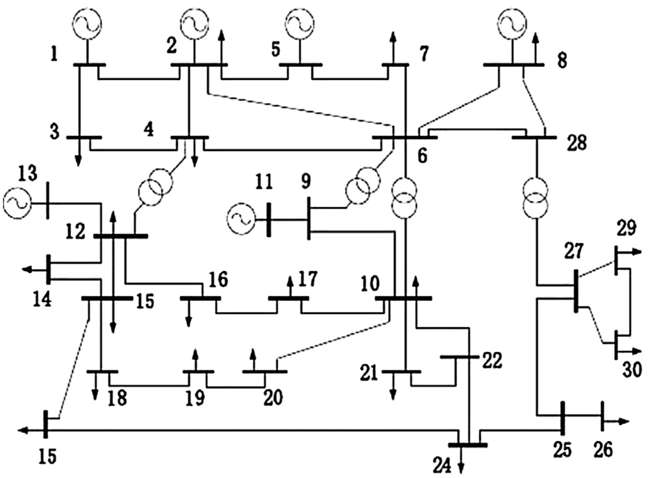

Fig. 7. 30-bus IEEE standard system.

Different load-flow analyses for different loading cases are calculated. 895 study cases are normalized and tabulated. 615 of the study cases are used in the RNN training.

In the training process using BP algorithm, the number of neurons in each layer of the two hidden layers is 14 and 20 neurons respectively, while 15 and 25 neurons are used for each of the two hidden layers in utilizing PSO.

Table 3
The load-flow results when P_{Load} is changed to 135 MW at bus 15.

Bus number	Voltage angle (degree)	Load MW	Load MVAR	Generation MW	Generation MVAR
1	0	0	0	420.822	49.824
2	−8.66	21.7	12.7	40	38.868
3	−13.815	2.4	1.2	0	0
4	−16.834	7.6	1.6	0	0
5	−20.382	94.2	19	0	43.777
6	−19.395	0	0	0	0
7	−20.4	22.8	10.9	0	0
8	−20.423	30	30	0	82.559
9	−26.876	0	0	0	0
10	−30.827	5.8	2	0	0
11	−26.918	0	0	0	38.852
12	−34.225	11.2	7.5	0	0
13	−34.226	0	0	0	35.917
14	−38.165	6.2	1.6	0	0
15	−40.802	135	2.5	0	0
16	−33.139	3.5	1.8	0	0
17	−31.768	9	5.8	0	0
18	−37.966	3.2	0.9	0	0
19	−36.117	9.5	3.4	0	0
20	−34.829	2.2	0.7	0	0
21	−31.645	17.5	11.2	0	0
22	−31.73	0	0	0	0
23	−37.785	3.2	1.6	0	0

Table 3 (*Continued*)

Bus number	Voltage angle (degree)	Load MW	Load MVAR	Generation MW	Generation MVAR
24	−33.43	8.7	6.7	0	0
25	−30.147	0	0	0	0
26	−30.83	3.5	2.3	0	0
27	−27.671	0	0	0	0
28	−20.628	0	0	0	0
29	−29.239	2.4	0.9	0	0
30	−29.968	10.6	1.9	0	0

The PSO parameters are: $w = 0.8$, $c_1 = c_2 = 1.5$ and number of particles is set to be 30. The generated RNN is then tested using 280 testing points. Samples of the testing cases are illustrated in Table 4. Results clarified that the proposed predictor operates effectively with different loading cases.

Based on the results, the overall accuracy of the RNN for the 30-bus IEEE standard system using BP in the testing process is 95.1% while using PSO gives an accuracy of 97.5%.

Table 4
Sample of the testing results for 30-bus IEEE standard system.

Bus number	Voltage angle (degree)	Load MW	Load MVAR	Generation MW	Generation MVAR	Target	O/P of BP	O/P of PSO
Case 1: P_{Load} is changed to 200 MW at bus 3								
1	0.499784	0	0	0.74805	0.248743	0	0	0
2	0.474302	0.108	0.4233	0.060907	0.244092	1	1	1
3	0.445793	1	0.04	0	0.20629	1	1	1
4	0.446654	0.038	0.0533	0	0.20629	−1	1	−1
5	0.44128	0.471	0.6333	0	0.241822	1	1	1
6	0.444935	0	0	0	0.20629	1	1	1
7	0.441799	0.114	0.3633	0	0.20629	1	1	1
8	0.44186	0.15	1	0	0.251209	1	1	1
9	0.434979	0	0	0	0.20629	1	1	1
10	0.42991	0.029	0.0666	0	0.20629	1	1	1
11	0.435268	0	0	0	0.228404	1	1	1
12	0.430853	0.056	0.25	0	0.20629	0	0	0
13	0.430735	0	0	0	0.224122	1	1	1
14	0.428289	0.031	0.0533	0	0.20629	1	1	1
15	0.428292	0.041	0.0833	0	0.20629	1	1	1
16	0.429731	0.017	0.06	0	0.20629	1	1	1
17	0.429271	0.045	0.1933	0	0.20629	1	1	1
18	0.426777	0.016	0.03	0	0.20629	1	1	1
19	0.426502	0.047	0.1133	0	0.20629	1	1	1
20	0.427178	0.011	0.0233	0	0.20629	1	1	1
21	0.428634	0.087	0.3733	0	0.20629	1	1	1
22	0.428628	0	0	0	0.20629	1	1	1
23	0.427534	0.016	0.0533	0	0.20629	1	1	1
24	0.427445	0.043	0.2233	0	0.20629	1	0	0
25	0.429038	0	0	0	0.20629	1	1	1
26	0.428126	0.017	0.0766	0	0.20629	1	1	1
27	0.430499	0	0	0	0.20629	1	1	1
28	0.4431	0	0	0	0.20629	1	1	1
29	0.427094	0.012	0.03	0	0.20629	1	1	1
30	0.424039	0.053	0.0633	0	0.20629	1	1	1
Case 2: Q_{Load} is changed to 5 MVAR at bus 16								
1	0.499784	0	0	0.827171	0.248342	1	1	1
2	0.468268	0.108	0.4233	0.060907	0.299617	1	1	1
3	0.450548	0.012	0.04	0	0.20629	1	1	1

Please cite this article in press as: Ibrahim, A.M., El-Amary, N.H., Particle Swarm Optimization trained recurrent neural network for voltage instability prediction. J. Electr. Syst. Inform. Technol. (2017), http://dx.doi.org/10.1016/j.jesit.2017.05.001

Table 4 (*Continued*)

Bus number	Voltage angle (degree)	Load MW	Load MVAR	Generation MW	Generation MVAR	Target	O/P of BP	O/P of PSO
4	0.438811	0.038	0.0533	0	0.20629	1	1	1
5	0.428839	0.471	0.6333	0	0.270582	1	1	1
6	0.430095	0	0	0	0.20629	1	1	1
7	0.427924	0.114	0.3633	0	0.20629	1	1	1
8	0.424847	0.15	1	0	0.317924	1	1	1
9	0.395687	0	0	0	0.20629	0	0	0
10	0.375182	0.029	0.0666	0	0.20629	1	1	1
11	0.39668	0	0	0	0.261759	1	1	1
12	0.366749	0.056	0.25	0	0.20629	−1	−1	−1
13	0.366185	0	0	0	0.2807	0	0	0
14	0.36434	0.031	0.0533	0	0.20629	1	1	1
15	0.366803	0.041	0.0833	0	0.20629	1	1	1
16	0.31555	0.95	0.1421	0	0.20629	1	1	1
17	0.360053	0.045	0.1933	0	0.20629	1	1	1
18	0.366096	0.016	0.03	0	0.20629	1	1	1
19	0.368654	0.047	0.1133	0	0.20629	1	1	1
20	0.36935	0.011	0.0233	0	0.20629	1	1	1
21	0.375013	0.087	0.3733	0	0.20629	1	1	1
22	0.374853	0	0	0	0.20629	1	1	1
23	0.370626	0.016	0.0533	0	0.20629	1	0	0
24	0.3759	0.043	0.2233	0	0.20629	1	1	1
25	0.390217	0	0	0	0.20629	1	1	1
26	0.389331	0.017	0.0766	0	0.20629	1	1	1
27	0.398846	0	0	0	0.20629	1	1	1
28	0.426247	0	0	0	0.20629	1	1	1
29	0.395499	0.012	0.03	0	0.20629	1	1	1
30	0.391541	0.053	0.0633	0	0.20629	1	1	1

Sample of the testing cases results are combined in a bar chart representation as shown in Figs. 8 and 9. These bar charts clarify the superiority of using PSO than using BP in training RNN. These cases are arranged for both IEEE standard systems as listed in Table 5.

Table 5
Sample of the testing cases.

	14-bus IEEE standard system	30-bus IEEE standard system
Case (i)	P_{Load} is changed to 535 MW at bus 5	P_{Load} is changed to 200 MW at bus 3
Case (ii)	Q_{Load} is changed to 15 MVAR at bus 10	Q_{Load} is changed to 5 MVAR at bus 16
Case (iii)	P_{Load} is changed to 60 MW at bus 10	P_{Load} is changed to 30 MW at bus 18 and to 105 MW at bus 20
Case (iv)	P_{Load} is changed to 165 MW at bus 9 and to 50 MW at bus 12	P_{Load} is changed to 135 MW at bus 15

5. Conclusion

This paper presents an early predictor for voltage instability based recurrent neural network (RNN) mechanism. The predictor is constructed on the basis of voltage phase angle data of each bus in the electrical network recorded by PMUs. The trained RNN with Particle Swarm Optimization (PSO) is proposed in this paper. The proposed approach is examined on 14-bus and 30-bus IEEE standard systems. The performance of PSO training algorithm is compared with

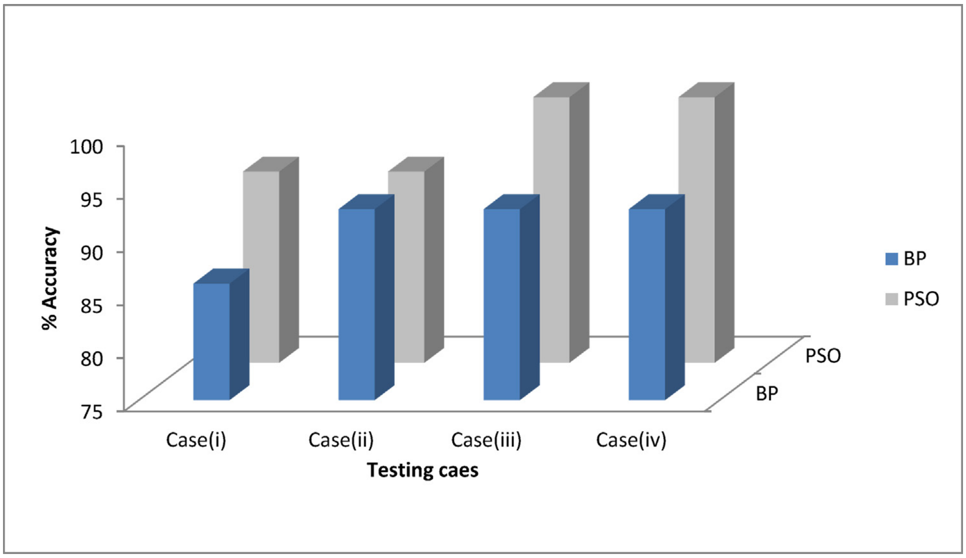

Fig. 8. The percentage accuracy of the proposed method applied for 14-bus IEEE standard system.

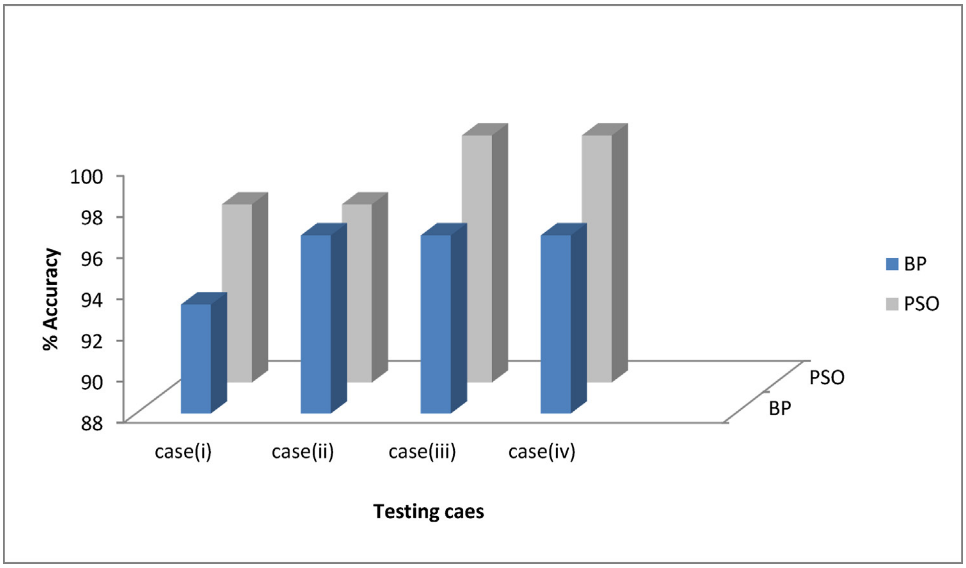

Fig. 9. The percentage accuracy of the proposed method applied for 30-bus IEEE standard system.

Backpropagation (BP) training algorithm. Training recurrent neural network using BP or PSO successfully predicts the voltage instability effectively as demonstrated from results. Based on the results, utilization of PSO in training recurrent neural networks gives higher performance than using BP.

References

Alpaydin, G., Dundar, G., Balkir, S., 2002. Evolution-based design of neural fuzzy networks using self-adapting genetic parameters. IEEE Trans. Fuzzy Syst. 10 (April (2)), 211–221.

Angelim, Jorge Henrrique, Affonso, Carolina Mattos, 2016. Impact of distributed generation technology and location on power system voltage stability. IEEE Latin Am. Trans. 14 (April (4)), 1758–1765.

Angeline, P.J., 1998. Evolutionary optimization versus particle swarm optimization: philosophy and performance differences. In: Proceeding of 7th International Conference, EP98 San Diego, California USA, pp. 601–610.

Barbounis, T.G., Theocharis, J.B., et al., 2006. Long-term wind speed and power forecasting using local recurrent neural network models. IEEE Trans. Energy Conv. 21 (March (1)), 273–284.

Bayrak, Mehmet, 2009. Recurrent artificial neural network based islanding protection by using generator deviation. Scientific Res. Essay 4, 212–216.

Chen, Chun-Jung, Chen Tien-Chi, Design of a Power System Stabilizer Using a new Recurrent Neural Network, Innovative Computing, Information and Control, ICICIC '06., vol. 1, pp. 39–43, August 2006.

Clerc, M., Kennedy, J., 2002. The particle swarm-explosion, stability, and convergence in a multidimensional complex space. IEEE Trans. Evolut. Comput. 6 (February (1)), 58–73.

Depablos, Juancarlo, Centeno, Virgilio, Phadke, Arun G., Ingram, Michael, 2004. Comparative testing of synchronized phasor measurement units proceeding of IEEE. Power Engineering Society General Meeting.

El-Amary, N.H., Mostafa, Y.G., Mansour, M.M., Mekhamer, S.F., Badr, M.A.L., 2008. Phasor measurement units' allocation using discrete particle swarm for voltage stability monitoring. In: Proceeding of IEEE Conference, Electrical Power and Energy Conference 2008, EPEC08, Vancouver, British Columbia, Canada, October.

Goedtel, A., daSilva, I.N., Amaral, P.J., 2006. Semi recurrent neural network for induction motor speed estimation in industry applications. IEEE MELECON2006, 1134–1137.

Gudise, V.G., Venayagamoorthy, G.K., 2003. Comparison of particle swarm optimization and backpropagation as training algorithms for neural networks. IEEE Swarm Intelligence Symposium, 110–117.

Janson, D.J., Frenzel, J.F., 1993. Training product unit neural networks with genetic algorithms. IEEE Intell. Syst. Appl. 8 (October (5)), 26–33.

Kamalasadan, Sukumar, 1998. Application of Artificial Intelligence Techniques in Power Systems, Special Study Report. Asian Institute of Technology, Bangkok, Thailand, November.

Kennedy, J., Eberhart, R., 2001. Swarm Intelligence. Morgan Kaufmann Publishers.

Khatua, Kaushik, Yadav, Neha, 2015. Voltage stability enhancement using VSC-OPF including wind farms based on genetic algorithm. Int. J. Electr. Power Energy Syst. 73, 560–567.

Laverty, D.M., Morrow, D.J., Best, R.J., Crossley, P.A., 2009. Differential ROCOF relay for loss-of-mains protection of renewable generation using phasor measurement over internet protocol, integration of wide-scale renewable resources into the power delivery system. CIGRE/IEEE PES Joint Symposium.

Lin, Ying-Hong, Liu, Chih-Wen, Chen, Ching-Shan, 2004. A new PMU-based fault detection/location technique for transmission lines with consideration of arcing fault discrimination—part I: theory and algorithms. IEEE Trans. Power Deliv. 19 (4), 1587–1593.

Mendes, R., Cortez, P., et al., 2002. Particle swarms for feedforward neural network training. Proceedings of the 2002 International Joint Conference on Neural Networks 2, 1895–1899.

Mohandas, N., Balamurugan, R., Lakshminarasimman, L., 2015. Optimal location and sizing of real power DG units to improve the voltage stability in the distribution system using ABC algorithm united with chaos. Int. J. Electr. Power Energy Syst. 66, 41–52.

Murty, V.V.S.N., Kumar, Ashwani, 2015. Optimal placement of DG in radial distribution systems based on new voltage stability index under load growth. Int. J. Electr. Power Energy Syst. 69, 246–256.

Nakawiro, Worawat, Erlich, István, 2008. Online voltage stability monitoring using artificial neural network. In: Proceeding of 3rd International Conference on Electric Utility Deregulation and Restructuring and Power Technologies, Nanjing China.

Phadke, A.G., 2002. Synchronized phasor measurements – a historical overview. Proceeding of IEEE Transmission and Distribution Conference and Exhibition, 476–479.

Salerno, J., 1997. Using the particle swarm optimization technique to train a recurrent neural model. Ninth IEEE International Conference on Tools with Artificial Intelligence, 45–49.

Trelea, I.C., 2003. The particle swarm optimization algorithm: convergence analysis and parameter selection. Inf. Process. Lett. 85 (March (6)), 317–325.

Xiao, Peng, Venayagamoorthy, Ganesh K., Keith, A., 2007. Corzine and combined training of recurrent neural networks with particle swarm optimization and backpropagation algorithms for impedance identification. IEEE Swarm Intelligence Symposium, Paper 1042.

Zhang, Chao, Bo, Zhiqian, Zhang, Baohui, Klimek, A., Han, Min, Tan, Jiancheng, 2009. An Integrated PMU and Protection Scheme for Power Systems. Universities Power Engineering Conference (UPEC).