

# Comparison of modified teaching–learning-based optimization and extreme learning machine for classification of multiple power signal disturbances

P. K. Nayak<sup>1</sup> · S. Mishra<sup>2</sup> · P. K. Dash<sup>3</sup> · Ranjeeta Bisoi<sup>3</sup>

Received: 21 September 2014 / Accepted: 9 August 2015  
© The Natural Computing Applications Forum 2015

**Abstract** This paper presents a modified TLBO (teaching–learning-based optimization) approach for the local linear radial basis function neural network (LLRBFNN) model to classify multiple power signal disturbances. Cumulative sum average filter has been designed for localization and feature extraction of multiple power signal disturbances. The extracted features are fed as inputs to the modified TLBO-based LLRBFNN for classification. The performance of the proposed modified TLBO-based LLRBFNN model is compared with the conventional model in terms of convergence speed and classification accuracy. Also, an extreme learning machine (ELM) approach is used to optimize the performance of the proposed LLRBFNN and is compared with the TLBO method in classifying the multiple power signal disturbances. The classification results reveal that although the TLBO approach produces slightly better accuracy in comparison with the ELM approach, the latter is much faster in implementation, thus making it suitable for processing large quantum of power signal disturbance data.

**Keywords** LLRBFNN · Power disturbance signals · Feature extraction · Cumulative sum average filter · Pattern recognition · Simultaneous power quality events · Teaching–learning-based optimization · Extreme learning machine

## 1 Introduction

The quality of electric power has become a pressing concern for electric utilities and their customers over the last decade. This is primarily due to the use of solid-state devices in power control resulting in voltage and current waveform distortions; voltage quality disturbances such as sags, swells, and interruptions; oscillatory and impulsive transients; multiple voltage notches due to solid-state converter switching; generation of harmonics; and the use of renewable energy sources. Therefore, power quality event detection and classification assume considerable importance in determining the sources of these disturbances, thereby making it possible for taking appropriate actions in mitigating them. In the past decade, researchers analyzed the power quality issues with the increasing amount of measurement data from power quality monitors. The time of occurrence and the frequency of power quality disturbances [1–4] are unknown, so the monitoring is often required over an extended period. Further, the power quality disturbances can appear simultaneously as in realistic power networks there are multiple sources of different disturbing events. Till date, most of the power quality analysis techniques have only considered a few power quality simultaneous disturbances namely voltage sags with harmonics, voltage swells with harmonics, and harmonics with interruptions. However, it is not enough in a realistic application scenario when there is a possibility of the occurrences of several simultaneous disturbances creating a more adverse impact on the power distribution network. Thus, it will be desirable to evolve a methodology which will be able to analyze other combinations of two or more power signal disturbances.

According to signal processing point of view, the non-stationary power signal disturbance analysis could be

---

✉ P. K. Dash  
pkdash.india@gmail.com

<sup>1</sup> Synergy Institute of Engineering and Technology,  
Dhenkanal, India

<sup>2</sup> Centurion University, Bhubaneswar, India

<sup>3</sup> S.O.A. University, Bhubaneswar, India

divided into groups such as detection, classification, parameter estimation, and localization. Several authors have presented different signal processing techniques based on discrete wavelet transform (DWT) [5–7], S-Transform (ST) [8–10], in order to identify the type of disturbance present in the power signal more effectively. By this method, it is possible to extract important information from a disturbance signal and determine the type of disturbance that has caused a power quality problem to occur. Although ST is a powerful tool for power signal disturbance assessment, it involves high computational cost. This paper, therefore, considers a computationally simpler technique, which is able to detect, localize, and produce features for robust classification of power signal disturbance events. The proposed procedure is based on a cumulative sum average filter that averages input samples and identifies the multiple changes in the nonstationary power quality waveforms. The filter computes the sum of one cycle data of the original signal. Using this filter, several relevant features such as standard deviation, variance, mean, max, min, entropy, and energy are extracted from the nonstationary power signal data. After the features are extracted, a classification technique is used to classify the data, thereby identifying the nature of the power signal disturbance patterns.

In recent years, the radial basis function neural network (RBFNN) is extensively used for classification and pattern recognition [11–13] in a variety of real-world engineering problems including power quality events. The RBFNN is a feedforward neural network which is faster than the multilayer perceptron (MLP) network in terms of speed of convergence and accuracy. The wavelet neural network (WNN) [14, 15] is another powerful tool to accurately identify various types of power signal disturbances by using time- and frequency-localized nonlinear wavelet basis functions. However, it is difficult to find the proper orthogonal or nonorthogonal wavelet basis functions and to determine the optimal number of hidden layer nodes for the WNN. Another alternative to WNN is the local linear wavelet neural network (LLWNN) [16], where the connection weights between the hidden units and output units are replaced by a local linear combination of weighted inputs to remove redundant nodes in the hidden layer. The advantage of this type network is that the number of neurons in the hidden layers is equal to the number of inputs in the input layer. This architecture thus dispenses the arbitrary choice of neurons in the hidden layer as in the cases of traditional MLP and RBFNNs. However, it suffers from slow convergence speed due to the use of wavelet basis functions in the network. Therefore, in this paper, we have considered an alternative type of RBFNN known as LLRBFNN [17] network in order to take the advantage of the local capacity of the radial basis functions for

producing significantly accurate classification. Earlier, LLRBFNN trained with particle swarm optimization (PSO) technique provides good result in financial forecasting in comparison with LLWNN [17]. TLBO algorithm is a new population-based optimization technique proposed in 2011 by Rao et al. [18–24]. Earlier, TLBO algorithm has been applied in constrained mechanical design optimization [21], heat exchangers [22], thermoelectric cooler [23], etc., producing good results.

Therefore, in this paper, the TLBO algorithm is used to train the weights of the LLRBFNN for the classification of power quality events. Also, the optimization of the weights of LLRBFNN model using PSO algorithm is presented in this paper to provide a performance comparison between the TLBO and PSO algorithms. Further to get faster convergence speed and improved classification accuracy, we propose a modified TLBO-based LLRBFNN model for the classification of multiple power signal disturbances. Another powerful learning paradigm is the use of ELM [25–29] for both regression- and classification-type problems. ELM is a much wider type of generalized SLFN (single hidden layer feedforward neural network) whose input weights and offsets of the hidden layer are chosen in a random manner and require no tuning. Architecturally, the ELM is different from traditional learning algorithms for a neural type of SLFN and aims to reach not only the smallest training error but also the smallest norm of output weights. Also, ELM has less computational complexity and provides a unified solution to different practical applications such as regression, binary, and multiclass classifications. It differs from the widely used LS-SVM (least square support vector machines) technique which is computationally far more complex than ELM and necessitates the use of different variants for different problems. On the other hand, ELM provides a unified approach and has only one formulation for different problem types. The ELM completes the whole computation process at once and generates a unique optimal solution without using iterations which provides an ease of parameter selection and fast learning speed. Since practical simultaneous power quality event classification uses a large signal waveform database, it is proposed in this paper to examine the performance of the ELM-trained LLRBFNN classifier in comparison with the modified TLBO-trained LLRBFNN classifier.

This paper has been organized as follows: Sects. 2 and 3 present the cumulative sum average filter (CUSAF) [30–32] for multiple power signal disturbances detection and feature extraction, respectively. Section 4 presents modified TLBO-based LLRBFNN learning, while in Sect. 5, the ELM algorithm is presented. Section 6 outlines the TLBO procedure, and results and discussion are presented in Sect. 7. Laboratory test setup and implementation procedure are described in Sect. 8, while concluding remarks are given in Sect. 9 followed by references.

## 2 Cumulative sum average filter (CUSAF) for multiple power signal disturbances

The Fourier transforms (FT) and its variants such as STFT (short-time Fourier transform) have been used for power quality assessment without much success. In contrast to Fourier transform-based technologies, the wavelet transform uses short windows at high frequencies and long windows at low frequencies. The poor concentration of energy from wavelet transform leads to ST. The computational complexity of ST involves long calculation time even for short data window. To process large volumes of power signal data, it becomes time-consuming and increases cost overhead. The filter designed here is called cumulative sum average filter (CUSAF) [30–32] which is simple to implement and gives better localization of nonstationary multiple power disturbance signals. The CUSAF averages the input samples and localizes high-frequency components of the nonstationary signal. For a given set of voltage signal samples  $s(k)$ , the CUSAF can be represented mathematically as

$$s_{\text{SUM}}(k) = \sum_{l=k-\text{NS}+1}^K s(l) \quad (1)$$

where  $k$  = sample count; NS = number of samples per cycle;  $K$  = total number of samples considered for the study,  $l$  = an index.

Equation (1) can be written in recursive form as

$$s_{\text{SUM}}(k) = s_{\text{SUM}}(k-1) + s(k) - s(k-\text{NS}) \quad (2)$$

For sample variation, Eq. (2) can be written as

$$s_{\text{SUM}}(k_p) = s_{\text{SUM}}(k_p-1) + s(N_s + k_p) - s(k_p) \quad (3)$$

$$\text{Let } s(N_s + k_p) = g_{\text{sum}} \quad (4)$$

where  $N_s$  is the starting sample position,  $k_p$  is the number of sample variations, and  $g_{\text{sum}}$  is an array where the filtered data are to be stored.

The amplitude response of the filter is given by

$$H(e^{j\omega}) = N_s e^{j\omega(N_s+1)/2} \left[ \frac{\sin c(\omega N_s/2)}{\sin c(\omega/2)} \right] \quad (5)$$

For multiple sample variation, the filter equation can be written as

$$\begin{aligned} s_{1\text{SUM}}(k_p) &= s_1(k_{p_1}-1) + s_1((N_{s_1}) + k_{p_1}) - s_1(k_{p_1}) \\ &\vdots \\ s_{1\text{SUM}}(k_{p_n}) &= s_1(k_{p_n}-1) + s_1((N_{s_n}) + k_{p_n}) - s_1(k_{p_n}) \end{aligned} \quad (6)$$

As in Eq. (4), let

$$s_1(N_{s_n} + k_{p_n}) = g_{1\text{sum}} \quad (7)$$

where  $N_{s_n} = N_{s_1}, N_{s_2}, \dots$  sample variations, and  $g_{1\text{sum}}$  is an array where the filtered data are to be stored.

The amplitude response for the multiple sample variation of the filter is given by

$$H_1(e^{j\omega}) = N_{s_n} e^{j\omega(N_{s_n}+1)/2} \left[ \frac{\sin c(\omega N_{s_n}/2)}{\sin c(\omega/2)} \right] \quad (8)$$

### 2.1 Multiple positional variations in nonstationary power signal disturbance

For positional variation [10], the Eq. (2) can be written as

$$s_{\text{SUM}}(k_p) = s_{\text{SUM}}(k_p-1) + s((N_p - N_s) + k_p) - s(k_p) \quad (9)$$

$$s((N_p - N_s) + k_p) = g_{\text{sum}} \quad (10)$$

where  $N_s$  is the starting point of variation, and  $N_p$  is the end point of variation.

Similarly, the amplitude response for the positional variation in the filter is given by

$$H_2(e^{j\omega}) = N_L e^{-j\omega(N_L-1)/2} \left[ \frac{\sin c(\omega N_L/2)}{\sin c(\omega/2)} \right] \quad (11)$$

where  $N_L = N_p - N_s$

For multiple positional variation, the filter equation can be written as

$$\begin{aligned} s_{2\text{SUM}}(k_p) &= s_{2\text{SUM}}(k_{p_1}-1) + s_2((N_{p_1} - N_{s_1}) + k_{p_1}) - s_2(k_{p_1}) \\ &\vdots \end{aligned} \quad (12)$$

$$s_{2\text{SUM}}(k_{p_n}) = s_{2\text{SUM}}(k_{p_n}-1) + s_2((N_{p_n} - N_{s_n}) + k_{p_n}) - s_2(k_{p_n}) \quad (13)$$

$$\text{Let } s_2((N_{p_n} - N_{s_n}) + k_{p_n}) = g_{2\text{sum}} \quad (14)$$

where  $N_{s_n} = N_{s_1}, N_{s_2}, \dots$  sample variations, and  $g_{2\text{sum}}$  is an array where the filtered data are to be stored.

The amplitude response for the multiple positional variations is given by

$$H_3(e^{j\omega}) = N_{L_n} e^{-j\omega(N_{L_n}-1)/2} \left[ \frac{\sin c(\omega N_{L_n}/2)}{\sin c(\omega/2)} \right] \quad (15)$$

where  $N_{L_n} = N_{p_n} - N_{s_n}$  and  $N_{s_n} = N_{s_1}, N_{s_2}, \dots$  starting positions,  $N_{p_n} = N_{p_1}, N_{p_2}, \dots$  end positions, and  $g_{\text{sum}}$  is an array where the filtered data are to be stored.

### 2.2 Combined multiple sample variations with positional variation in nonstationary power signal disturbances

Combining the Eqs. (6) and (13), we get

$$S_{\text{SUM}} = s_{1\text{SUM}}(k_{p_n}) + s_{2\text{SUM}}(k_{p_n}) \quad (16)$$

and

$$G_{\text{sum}} = g_{1\text{sum}} + g_{2\text{sum}} \quad (17)$$

The combined amplitude response is given by

$$H(e^{j\omega}) = N_{s_n} e^{j\omega(N_{s_n}+1)/2} \left[ \frac{\sin c(\omega N_{s_n}/2)}{\sin c(\omega/2)} \right] + N_{L_n} e^{-j\omega(N_{L_n}-1)/2} \left[ \frac{\sin c(\omega N_{L_n}/2)}{\sin c(\omega/2)} \right] \quad (18)$$

The CUSAF averages the number of input samples on a cycle-to-cycle basis in a recursive manner and is capable of detecting changes in the power network disturbance signals.

### 3 Application of CUSAF to nonstationary power signal disturbances

The program for the multiple power disturbance signals such as position and sample variations is written using MATLAB software. Figure 1 shows the localization of different types of multiple nonstationary power signal disturbances using cumulative sum average filter. Referring to Fig. 1a for the case of multiple sample variations, the different notches occur at the 95th, 140th, 170th, and spike occurs at the 340th, 370th, 450th sample points, respectively. After the application of cumulative sum average filter, it is observed that the similar samples in the signal are zero from the beginning of the disturbance and after the completion of the disturbance, and this localizes the fault prominently.

In the case of multiple positional variations, for example in the multiple transient case, the first instance started at 129th sample and second instance started at 250th sample, the transient lasts for 60 samples in both the instances. It is observed that after application of cumulative sum average filter from first sample to the starting of transient is zero and after 190th sample all the sample values are zero up to 249th sample and after 311th sample to end of the signal all the sample values are zero as the filter has been designed in one cycle back fashion. For the case of combined multiple positional and sample variation, the transient for first instance started at 129th sample and lasts for 60 samples, second instance started at 250th sample and lasts for 60 samples, and the spike occurs at 340th, 370th, and 450th sample. After application of cumulative sum average filter, it is observed that the similar samples are zero, and thus, multiple faults are localized prominently.

#### 3.1 Feature extraction

The fundamental sinusoidal signal is taken to be of 50 Hz, and it is sampled at 64 samples per cycle with a sampling frequency of 3.2 kHz ( $50 \times 64$ ). Ten categories of

multiple power signal disturbances making a total of  $(200 \times 10)$  2000 signals are generated using the formulas given in Table 1 from which different statistical features such as standard deviation, mean, max, energy, and entropy have been extracted. Table 2 provides parametric equations for the extraction of the above-mentioned features (shown in Table 3) from the signal samples  $[x_1, x_2, \dots, x_j, \dots, x_N]$  obtained from the change detection filter with the help of the cumulative sum average technique. The total number of positional and sample variation disturbance nonstationary power signals used for classification is given below:

1. Multiple notch
2. Multiple spike + multiple notch
3. Multiple transient
4. Transient + multiple notch
5. Swell + multiple spike
6. Sag + multiple notch
7. Transient + notch + spike
8. Sag + harmonics
9. Swell + harmonics
10. Sag + harmonics

The seven extracted features are given as inputs to the TLBO-based LLRBFNN network for classification. A block diagram for multiclass classification of the power signal disturbance events is shown in Fig. 2.

### 4 Proposed TLBO-based LLRBFNN model

In LLRBFNN architecture, a local linear model replaces the connection of weights between the hidden layer and output layer of conventional RBFNN [19]. In comparison with RBFNN (radial basis function neural network), the LLRBFNN requires less number of neurons than the RBFNN to approximate the same nonlinear system. In LLRBFNN, the input and number of hidden layer nodes are equal which reduces overall node requirements. The LLRBFNN becomes more efficient network because the ability of approximation has been increased by placing a local linear model instead of the hidden layer units. Different optimization algorithms such as PSO with hunter particle [19] and GA [27] have been applied for training the network to have classification accuracy. In this paper, we have proposed a recently developed TLBO-based training algorithm to the LLRBFNN network. Further, the TLBO algorithm has been modified and applied for training to have faster convergence speed and classification accuracy.

#### 4.1 Model description with TLBO-based training

The architecture of the proposed TLBO-based LLRBFNN training is shown in Fig. 3.

**Fig. 1** (Combined multiple positional and sample variation) Localization of power network disturbance signals using cumulative sum average filter. **a** Localization of multiple spikes + notch. **b** (Multiple positional variation) Localization of multiple transient. **c** Localization of transient + notch. **d** Localization of multiple transient + notch. **e** Localization of transient + notch + spike. **f** Localization of transient + sag. **g** Localization of voltage sag + harmonic using moving sum average filter. **h** Localization of voltage swell + harmonic using moving sum average filter

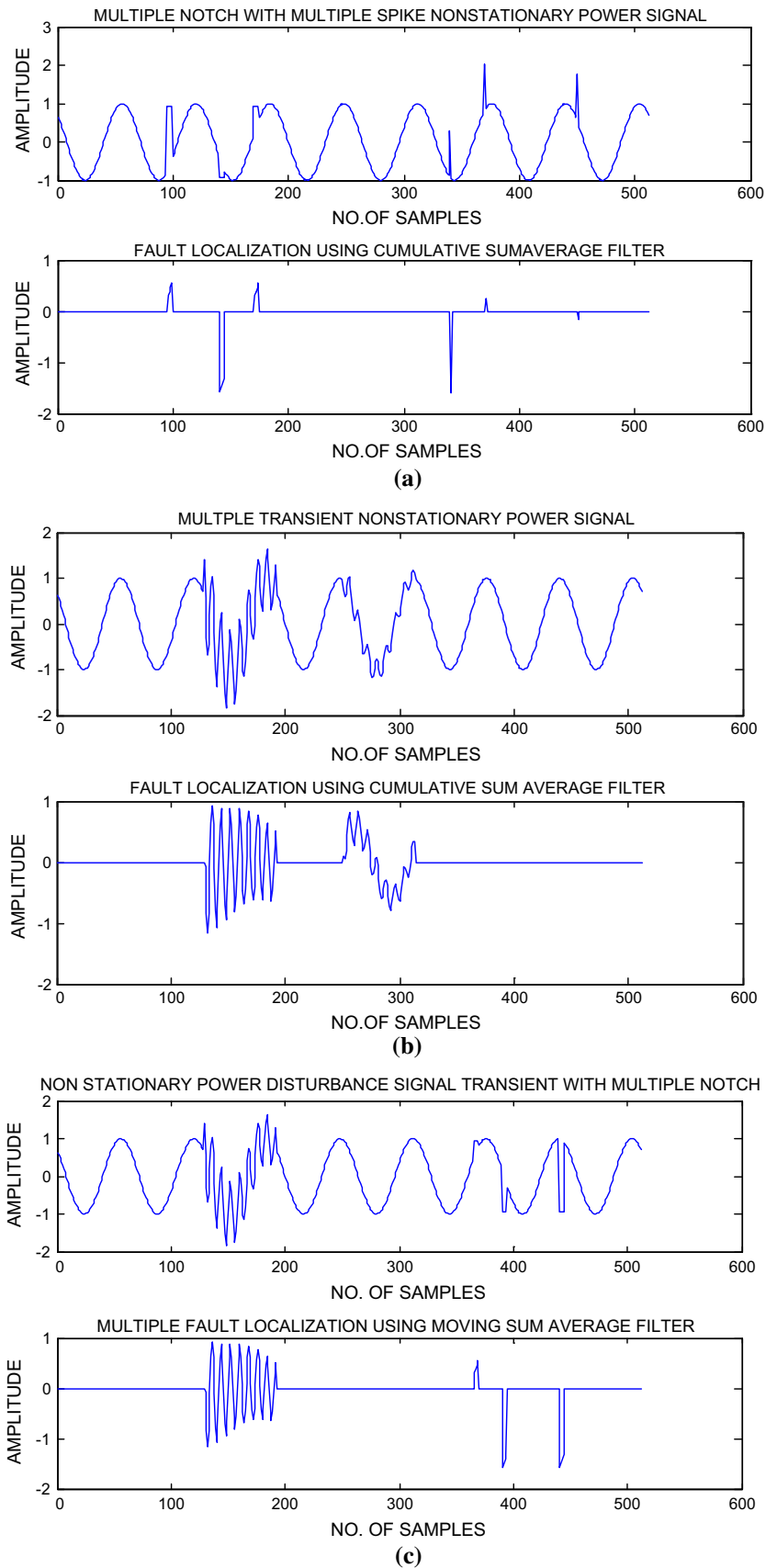
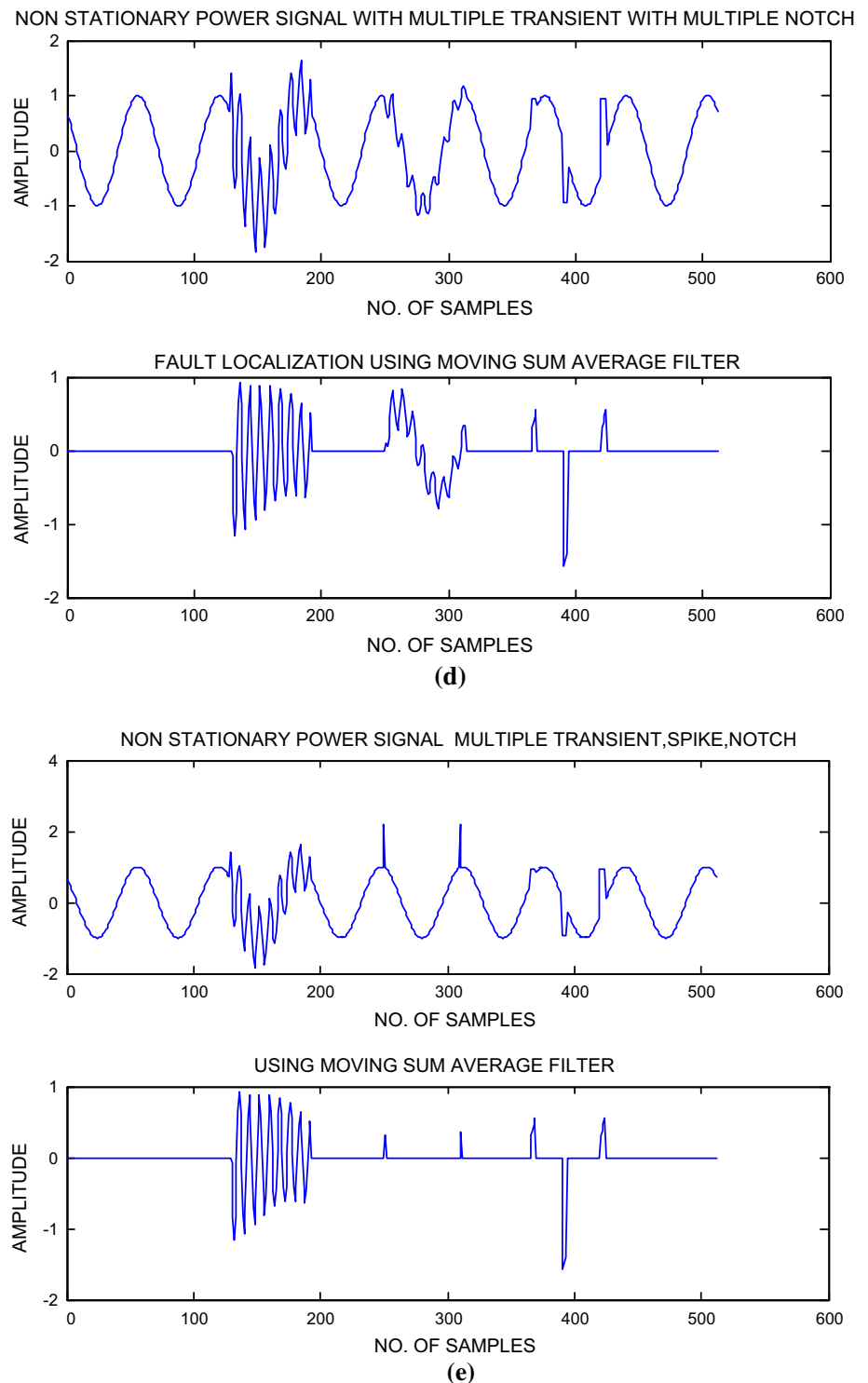


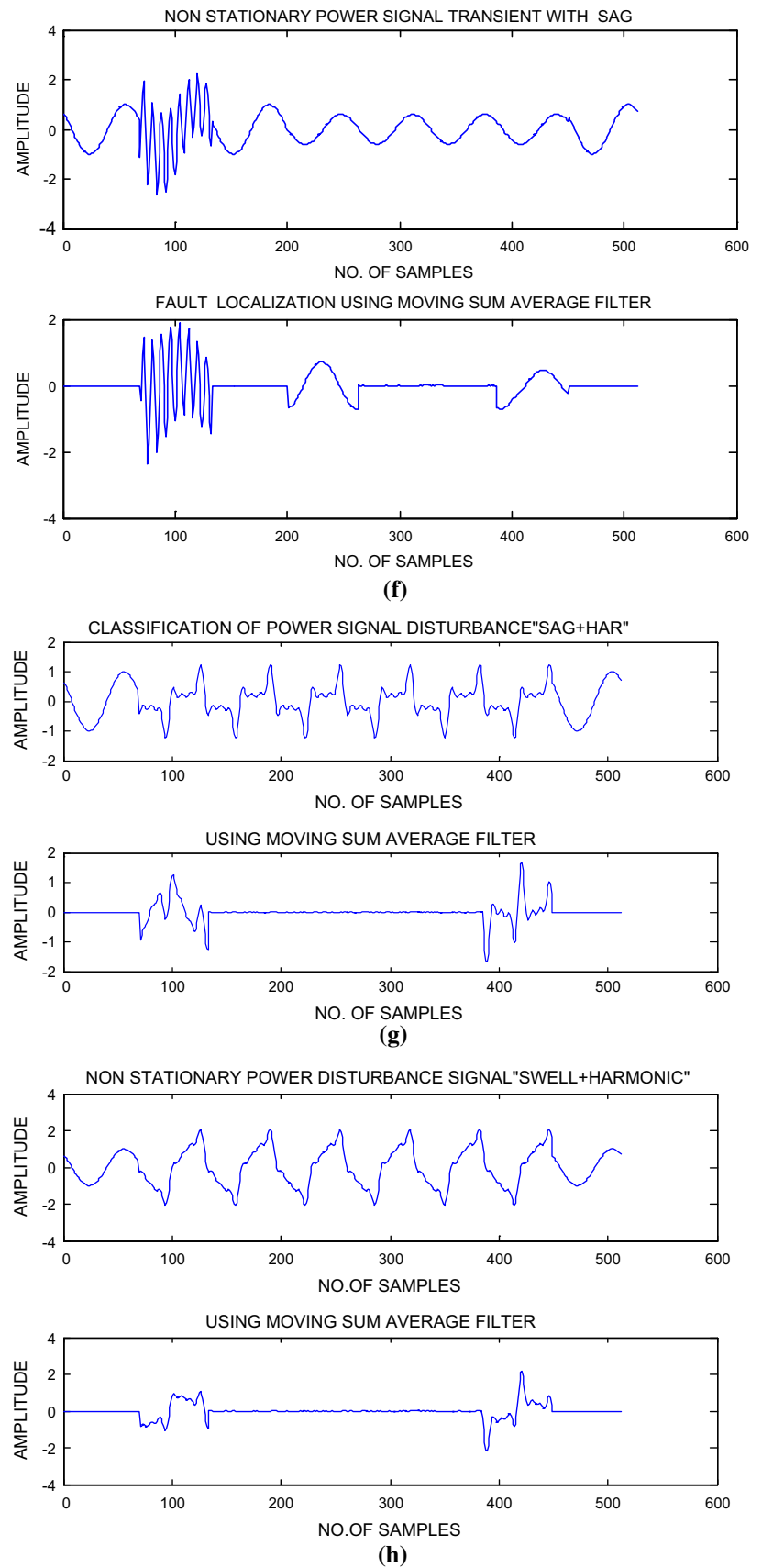
Fig. 1 continued



In Fig. 3,  $x = [x_1, x_2, x_3 \dots x_n]$  are inputs (features);  $g_1(x), g_2(x), \dots, g_n(x)$  are the Gaussian activation functions in the hidden units. The hidden layer of the LLRBFNN model takes the local linear weight and computes the output. The activation function of the  $k$ th hidden neuron is defined by

$$g_k(x) = e^{\left( \frac{-\|x - c_k\|^2}{2\sigma_k^2} \right)}, \quad k = 1, 2, \dots, n \quad (19)$$

where  $h_k(x)$  is the activation function of  $k$ th node,  $\sigma_k^2$  is the parameter which controls the smoothness of the activation function,  $c_k$  is the center of the hidden node, and  $\|x - c_k\|$

**Fig. 1** continued

**Table 1** Power signal disturbance models

Power signal disturbance	Equation
Normal sine	$V(k) = A \cos(k\omega\Delta T + \phi)$ , $A = 1(pu)$ , $f = 50$ Hz
Sag	$V(k) = A(1 + \alpha(u(k - k_1) - u(k - k_2))) \cos(k\omega\Delta T + \phi)$ , $0.3 \leq \alpha \leq 0.8$
Swell	$V(k) = A(1 + \alpha(u(k - k_1) - u(k - k_2))) \cos(k\omega\Delta T + \phi)$ , $1.3 \leq \alpha \leq 1.8$
Flicker	$V(k) = A(1 + \alpha_f) \cos(\beta k\omega\Delta T + \phi)$ , $0.1 \leq \alpha_f \leq 0.2$ , $5 \leq \beta \leq 20$ Hz
Transient	$V(k) = A(\alpha \cos(k\omega\Delta T))e^{-k\gamma}$ , $0.01 \leq \gamma \leq 0.05$ , $2.5 \leq \alpha \leq 4$
Harmonic	$V(k) = A(\alpha_1 \cos(k\omega\Delta T) + \alpha_3 \cos(3 \cos(k\omega\Delta T)) + \alpha_5 \cos(5 \cos(k\omega\Delta T)))$ $0.05 \leq \alpha_1 \leq 0.15$ , $0.05 \leq \alpha_3 \leq 0.15$ , $0.05 \leq \alpha_5 \leq 0.15$ , $\sum \alpha_i^2 = 1$
Oscillatory transient	$V(k) = A(\cos(k\omega\Delta T) + \alpha e^{-(k-k_1)/\tau} \cos \omega_n(k - k_1)(u(k_2) - u(k_1)))$ $0.1 \leq \alpha \leq 0.8$ , $0.5T \leq t_2 - t_1 \leq 3T$ , $8 \text{ ms} \leq \tau \leq 40 \text{ ms}$ , $300 \leq f_n \leq 900 \text{ Hz}$ ,
Sag and harmonic	$V(k) = A(1 + \alpha(u(k - k_1) - u(k - k_2))) \left( \alpha_1 \cos(k\omega\Delta T) + \alpha_3 \cos(3 \cos(k\omega\Delta T)) + \alpha_5 \cos(5 \cos(k\omega\Delta T)) \right)$ $0.3 \leq \alpha \leq 0.8$ , $0.05 \leq \alpha_1 \leq 0.15$ , $0.05 \leq \alpha_3 \leq 0.15$ , $0.05 \leq \alpha_5 \leq 0.15$ , $\sum \alpha_i^2 = 1$
Swell and harmonic	$V(k) = A(1 + \alpha(u(k - k_1) - u(k - k_2))) (\alpha_1 \cos(k\omega\Delta T) + \alpha_3 \cos(3 \cos(k\omega\Delta T)) + \alpha_5 \cos(5 \cos(k\omega\Delta T)))$ $1.3 \leq \alpha \leq 1.8$ , $0.05 \leq \alpha_1 \leq 0.15$ , $0.05 \leq \alpha_3 \leq 0.15$ , $0.05 \leq \alpha_5 \leq 0.15$ , $\sum \alpha_i^2 = 1$
Spike/impulsive transient	$V(k) = V_p(e^{-750(n-n_s)k\Delta T} - e^{-344(n-n_e)k\Delta T}) \times [U(n - n_s) - U(n - n_e)]$
Spike with transient	$V(k) = V_p(e^{-750(n-n_s)k\Delta T} - e^{-344(n-n_e)k\Delta T}) \times [U(n - n_s) - U(n - n_{se})]$ $\times \sin(2\pi f_T n T + \phi) \left[ 1 + V_T e^{-\alpha(n-n_s)T_p} \sin(2\pi f_T(n - n_s)k\Delta T + \phi) \right]$ $0.2 \leq V_p \leq 1$ , $T_p \leq n_s \Delta T \leq 7T_p$ , $n_{se} \Delta T = n_s \Delta T + 0.001s$ $0.3 \leq V_T \leq 1$ , $150 \leq \alpha \leq 1000$ , $300 \leq f_T \leq 900$ , $0 \leq \phi \leq 2\pi$

**Table 2** Parametric equations for feature extraction

Feature name	Feature equation
Standard deviation ( $\sigma$ )	$\sigma_j = \sqrt{\left( \frac{1}{N-1} \sum_{i=0}^{N-1} (x_j(n) - \mu_j)^2 \right)}$
Variance ( $\sigma^2$ )	$\sigma_j^2 = \left( \frac{1}{N-1} \sum_{i=0}^{N-1} (x_j(n) - \mu_j)^2 \right)$
Entropy (S)	$S_j = \sum_{j=1}^N P_j \log_2 \left( \frac{1}{P_j} \right)$
Mean ( $\mu$ )	$\mu_j = \sum_{j=1}^N x_j / N$
Energy (E)	$E_j = \sum_{n=0}^{N-1} (x_j(n))^2$
Max	$\max(x_1, x_2, \dots, x_N)$
Min	$\min(x_1, x_2, \dots, x_N)$

indicates the Euclidean distance between the inputs and the function center. The output from the hidden layer  $k$ th neuron is obtained by multiplying the activation function of the node with a weighted linear combination of the inputs as

$$h_k(x) = W_k g_k(x)$$

In expanded form, the output of the  $k$ th hidden layer node is therefore obtained as

$$h_k(x) = W_k g_k(x) = (w_{k0} + w_{k1}x_1 + w_{k2}x_2 \dots w_{kn}x_n) \cdot e^{\left( \frac{-\|(x - C_k)\|^2}{2\sigma_k^2} \right)} \quad (20)$$

The weight matrix associated with an output node  $m$  is thus written as

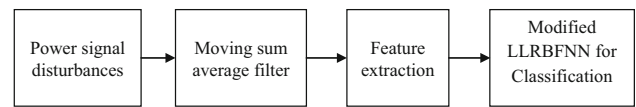
$$W_m = \begin{bmatrix} W_{m1} \\ W_{m2} \\ \dots \\ W_{mn} \end{bmatrix} = \begin{bmatrix} w_{m10} & w_{m11} & \dots & \dots & w_{m1n} \\ w_{m20} & w_{m21} & \dots & \dots & w_{m2n} \\ \dots & \dots & \dots & \dots & \dots \\ w_{mn0} & w_{mn1} & \dots & \dots & w_{mnn} \end{bmatrix} \times \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ \dots \\ x_n \end{bmatrix} \quad (21)$$

The output at the  $m$ th node of the output layer is given by



**Table 3** Extracted features for multiclass classification

Features	Multiple notch	Transient + multiple spike	Multiple spike + multiple notch	Transient + multiple notch	Sag + multiple notch	Swell + multiple spike	Sag + harmonics	Swell + harmonics
Std	0.1398	0.1268	0.1415	0.1716	0.7337	0.3744	0.5134	0.6761
	0.1422	0.1279	0.1656	0.1814	0.7568	0.3769	0.5164	0.6798
Variance	0.0195	0.0161	0.0200	0.0294	0.5383	0.8891	0.2636	0.4571
	0.0197	0.0163	0.0211	0.0299	0.6784	0.8898	0.2667	0.4621
Mean	-0.0046	-0.0030	-0.0033	-0.0080	-0.0012	-0.0383	0.0465	0.0724
	-0.0048	-0.0031	-0.0037	-0.0082	-0.0014	-0.0394	0.0467	0.0726
Max	0.5573	0.9286	0.5573	0.9286	0.7556	0.5112	0.6764	0.7839
	0.5578	0.9296	0.5574	0.9286	0.7566	0.6132	0.6827	0.7877
Min	-0.5744	-0.1573	-0.5744	-0.5644	-0.7556	-0.7112	0.7830	0.7481
	-0.5746	-0.1577	-0.5747	-0.5645	-0.7566	-0.7416	0.7963	0.7364
Entropy	0.1474	0.1426	0.1881	0.2084	0.9143	0.9100	0.6671	0.7307
	0.1564	0.1656	0.1971	0.3214	0.9442	0.9102	0.6715	0.7345
Energy	0.5622	0.5263	0.5971	0.6833	0.7892	0.9817	-0.5443	-0.5705
	0.5562	0.5472	0.5991	0.6831	0.8952	0.9828	-0.5505	-0.5743

**Fig. 2** Schematic diagram for the power disturbance signal classification

$$y_m = \sum_{k=1}^n \beta_{mk} h_k(x), \quad m = 1, 2, \dots, M \quad (22)$$

The next section describes another powerful approach for learning the parameters of the LLRBFNN based on ELM platform for multiclass power signal disturbance pattern recognition.

## 5 Extreme learning machine (ELM)

ELM is a recently introduced learning algorithm for SLFNs in which the input weights and biases are chosen randomly (weights of connections between the input variables and neurons in the hidden layer) and require no tuning. ELM not only has the capability of extremely fast learning and testing speed and exhibits smallest training error and smallest norm of output weights. Also, ELM has a similar cost function like the LS-SVM but with milder constraints and has been used for the feedforward neural networks, RBFNN, and SVMs. Thus, a robust procedure for both regression and classification results by using ELM approach for updating the weight parameters of the LLRBFNN between the hidden layer and the output layer.

The main advantage of ELM is that the hidden layer of SLFNs can work with a wide range of activation functions including piecewise continuous functions [31]. With a given D-dimensional input data set and N-dimensional hidden layer nodes characterized by a feature space H, the output function of ELM for SLFN can be represented as:

$$f_N(x) = \sum_{i=1}^N \beta_i h_i(x) \quad (23)$$

and

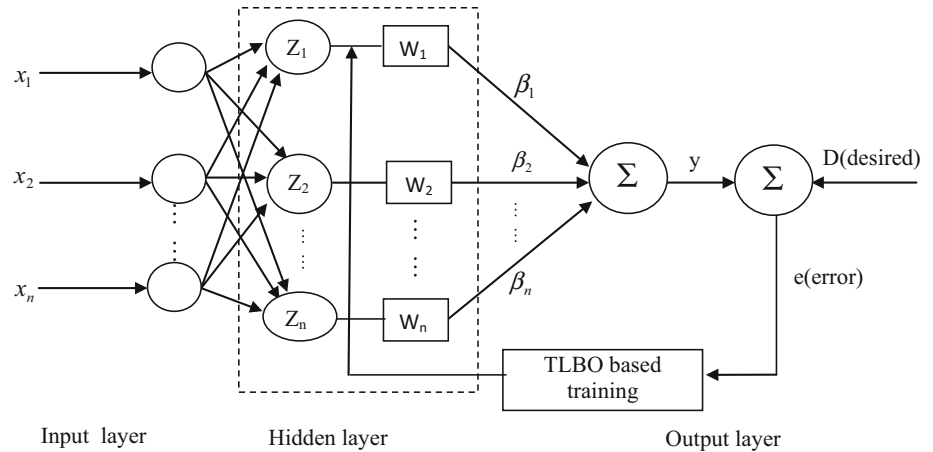
$$\beta = [\beta_1 \beta_2 \dots \beta_N]^T \quad (24)$$

where  $h_i(x)$  is the activation function of hidden layer, and  $\beta$  is the weight vector connecting the hidden layer neurons to output layer neuron. If the hidden layer neurons are of the local radial basis function type, the output row vector of the hidden layer is obtained as a function of the inputs as

$$h(x) = [h_1(x), h_2(x), \dots, h_n(x)] \quad (25)$$

Thus, the hidden layer output matrix is given by the  $H$  matrix is obtained as

**Fig. 3** Modified local linear RBF neural network



$$H = \begin{bmatrix} h_1(x_1) & h_2(x_1) & \dots & h_n(x_1) \\ \vdots & \vdots & \ddots & \vdots \\ h_1(x_n) & h_2(x_n) & \dots & h_n(x_n) \end{bmatrix} \quad (26)$$

Equation (25) can be written as

$$Y = H\beta = T \quad (27)$$

The training error and the norm of the output weights are minimized by using ELMMinimize  $\|H\beta - T\|^2$  and  $\|\beta\|$  to yield

$$\beta = H^\Psi T = (H^T H)^{-1} H^T T \quad (28)$$

where  $H^\Psi$  is the Moore–Penrose generalized inverse of matrix  $H$ , and  $T$  is the target.

The computation of the weight vector of the ELM given in Eq. (28) is done using all the data samples at a time.

### 5.1 Multiclass ELM classifier output function for classification

Since ELM can approximate any target continuous functions, the output of the ELM classifier  $H\beta$  can be as close to the class label of the input data. Thus, in a multiclass classification problem using ELM with a single-output node, the class label closer to the output value is considered to be the predicted class label of the input data. In this case, the classification problem for the proposed constrained-optimization-based ELM (this classification will be termed as ELM1) with a single-output node can be formulated as

$$\text{Minimize } L_P = \frac{1}{2} \|\beta\|^2 + \frac{1}{2} C \sum_{i=1}^N \xi_i^2, \quad (29)$$

subject to  $h(x_i)\beta = t_i - \xi_i, \quad i = 1, 2, \dots, N$

where  $C$  is a parameter to be chosen appropriately,  $\xi_i$  and  $t_i$  are the output and the target values at the  $i$ th instant;  $N$  = number of neurons in the hidden layer. In the case

where the training samples are small in number, the above minimization process yields

$$\beta = H^T \left( \frac{I}{C} + HH^T \right)^{-1} T \quad (30)$$

The output function for the ELM classifier is given by

$$f(x) = h(x)H^T \left( \frac{I}{C} + HH^T \right)^{-1} T \quad (31)$$

where  $h(x)$  is the hidden layer row vector and is obtained as and

$$h(x) = [h_1(x) \ h_2(x) \ \dots \ h_i(x) \ \dots \ h_L(x)] \quad (32)$$

$$h_i(x) = w_{i0} + w_{i1}x_1 + w_{i2}x_2 \dots w_{in}x_n \cdot e^{\left( \frac{-\|(x-C_n)\|^2}{2\sigma_n^2} \right)} \quad (33)$$

Another alternative approach is to consider that there is  $m$  number of output nodes pertaining to  $m$ -class patterns to be classified. If  $p$  signifies the class label of a given set of feature vectors, then the target output vector for the  $m$ -nodes becomes  $t_i = [0, 0, 0, \dots, 1, 0, 0, 0, \dots, 0]^T$ , and only the  $p$ th will have value equal to 1 and all other elements are chosen to be zeroes. In general, the target output for  $m$  classes (Fig. 4) is given by

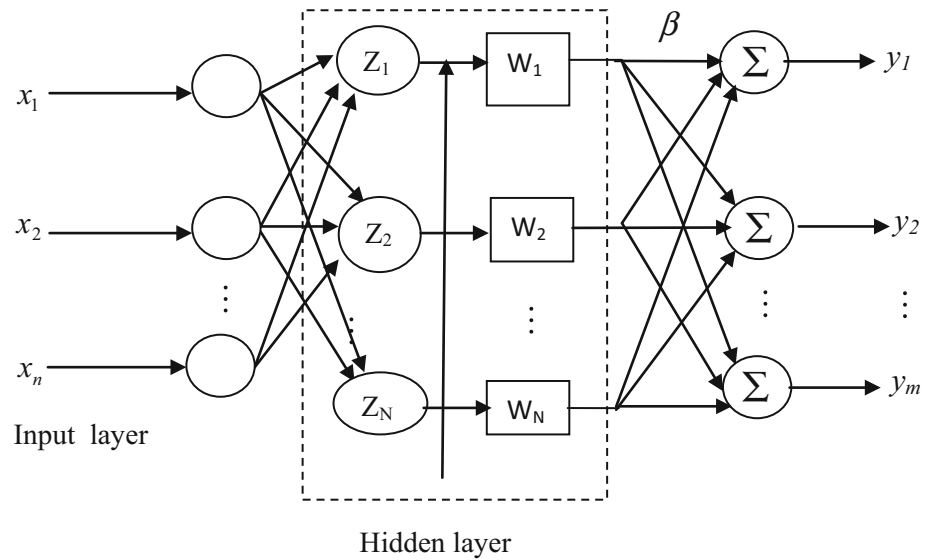
$$T = \begin{bmatrix} t_1^T \\ t_2^T \\ \vdots \\ t_N^T \end{bmatrix} = \begin{bmatrix} t_{11} & \dots & t_{1m} \\ t_{21} & \dots & t_{2m} \\ \vdots & \vdots & \vdots \\ t_{N1} & \dots & t_{Nm} \end{bmatrix} \quad (34)$$

Since seven classes of power network signals are considered for classification, the target vector becomes

$$T = \text{diag}[1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1] \quad (35)$$

The following optimization equation is solved to train the ELM for multiclass classification (ELMM):

**Fig. 4** Multiclass classification with more than one output neuron using ELM



$$\text{Minimize } L_P = \frac{1}{2} \|\beta\|^2 + \frac{1}{2} C \sum_{i=1}^N \xi_i^2, \quad (36)$$

$$\sum_{i=1}^N \sum_{j=1}^m \lambda_{i,j} (h(x_i) \beta_j - t_{i,j} + \xi_{i,j})$$

where  $\beta = [\beta_1, \beta_2, \dots, \beta_m]$ , and  $\beta_j$  represents the vector of weights linking the hidden layer to the  $j$ th output node. Thus, the output function of the multiclass classifier is obtained as

$$f_i(x) = h(x) H^T \left( \frac{I}{C} + H H^T \right)^{-1} T_i \quad (37)$$

where

$$T_i = [t_{i1}, t_{i2}, \dots, t_{im}]^T \quad (38)$$

In a multiclass classification, the number of output nodes is equal to  $m$  ( $m > 1$ ), and in such a case, the predicted class label of a given testing sample is the index number of the output node having the highest output value. Thus, the class label of  $x$  is given by

$$\text{Label}(x) = \arg \text{Max}(f_i(x)), \quad i = 1, 2, \dots, m$$

However, for large number of samples particularly when the tested system has a large dataset, the value of  $\beta$  is given by

$$\beta = C H^T \xi, \quad \text{and} \quad \xi = \frac{1}{C} (H^T)^{\Psi} \beta \quad (39)$$

and  $\xi = [\xi_1, \xi_2, \dots, \xi_N]$ . Finally, the value of  $\beta$  is obtained as

$$\beta = H^T \left( \frac{I}{C} + H^T H \right)^{-1} H^T T \quad (40)$$

For the ELM classifier, the following equation is used:

$$f(x) = h(x) H^T \left( \frac{I}{C} + H^T H \right)^{-1} H^T T \quad (41)$$

However, an online formulation for ELM for classification known as OS-ELM (online sequence ELM) is given in [29] using each new observation or chunk of observations as

$$\beta(k+1) = \beta(k) + P(k+1) H^T(k+1) [\beta(k+1) - H(k+1) \beta^T] \quad (42)$$

$$P(k+1) = P(k) - P(k) H^T(k+1) [I + H(k+1) P(k) H^T(k+1)]^{-1} H(k+1) P(k)$$

and  $P(k+1) = [K(k+1)]^{-1}$ ,  $I$  is an unity matrix of appropriate dimension.

The initial values are obtained using the Moore–Penrose generalized inverse as follows:

$$\beta(0) = \left[ \frac{I}{C} + H^T(0) H(0) \right]^{-1} H^T(0) T(0) \quad (43)$$

$$K(0) = H^T(0) H(0) \quad (44)$$

For classification power signal disturbances, the seven features are used as input to the LLRBFNN by using ELM classification algorithm. The weight parameters in the hidden layer are randomly chosen in the range of  $-1$  to  $+1$ . The value of  $C$  is selected to be equal to  $0.5$ . To study the convergence characteristic of the ELM classifier for power disturbance signals, the objective function to be minimized is chosen as

$$\text{Mean Square Error MSE} = \frac{1}{n} \sum_{k=1}^n (D_k - y_k)^2 \quad (45)$$

where “ $D_k$ ” is the desired vector,  $y_k$  is the output of the model, and  $n$  is the number of data. Another learning

scheme used for the LLRBFNN is the teaching–learning-based optimization technique, which is simple to implement for multiclass pattern recognition of power quality disturbances. The next section describes the TLBO algorithm for both the teacher phase and learner phase.

## 6 Weight update using TLBO algorithm

TLBO algorithm is the recently developed population-based optimization technique applied for mechanical design optimization problems. Teaching–learning-based optimization (TLBO) [20, 21] is basically divided into two parts: The first part is known as the “teacher phase,” while the second part is known as the “learner phase.” The “teacher phase” means learning from the teacher, and the “learner phase” means learners learn through the interaction between themselves.

### 6.1 Teacher phase

In the TLBO method, the teacher teaches the subject according to his capability and tries to enhance the ability of understanding of all learners to the level of his knowledge in the class, which is practically not possible. Thus, the teacher can only improve the mean level of the learners in a class depending on the capability of the learners. The procedure as to how the teacher improves the mean level of the learners follows a random process depending on different factors such as mean and teaching factor. In this paper, the TLBO technique is applied to train the weights of the LLRBFNN. In the teacher phase, the teacher tries to move the mean  $M_i$  toward his/her own knowledge level, and so the new mean is designated as  $M_{\text{new}}$ .

Assuming “ $W_n$ ” be the number of weights and “ $n$ ” is the number of learners, the best learner  $W_i^{\text{best}}$  is designated as the teacher in the subject  $i$ .

The teaching process is as follows:

$$W_i^{\text{best}} = M_{\text{new}} \quad (46)$$

The difference between the two mean is given by

$$D_f = \text{rand}(W_i^{\text{best}} - T_f \times M_i) \quad (47)$$

where  $T_f$  is the teaching factor that decides the value of mean to be changed, and in this program, the value of  $T_f$  is chosen as 2, and *rand* provides random number between 0 and 1.

Now adding the difference to the current solution, the update equation becomes:

$$W_i^{\text{new}} = W_i^{\text{old}} + \text{rand}(W_i^{\text{best}} - T_f \times M_i) \quad (48)$$

where  $M_i$  = mean of the population columnwise, which will give the mean for the particular subject.

Now considering  $M_{i,\text{new}} = \frac{W_i^{\text{best}} + W_i^{\text{min}}}{2}$ , and substituting  $M_i = M_{i,\text{new}}$  in the above Eq. (48), the Eq. (25) becomes

$$W_i^{\text{new}} = W_i^{\text{old}} + \text{rand}(W_i^{\text{best}} - T_f \times M_{i,\text{new}}) \quad (49)$$

By considering  $M_{i,\text{new}}$  in the above equation, we do not need to compute the mean value of population in each iteration and the diversity of population increases, because the new mean value  $M_{i,\text{new}}$  will be different for each individual.

To have fast convergence speed, a new exponential factor  $P_f = 0.9 - \exp\left(-\frac{W_i^{\text{best}} - W_i^{\text{min}}}{2}\right)$  is introduced in the update equation, hence Eq. (49) becomes

$$W_i^{\text{new}} = P_f W_i^{\text{old}} + \text{rand}(W_i^{\text{best}} - T_f \times M_{i,\text{new}}) \quad (50)$$

When the learner finishes learning from the teacher, Eq. (50) is updated, and the solution is accepted provided it is the optimal solution.

### 6.2 Learner phase

Learners put their efforts to increase their capability of understanding by interaction with each other. A learner learns new knowledge if the other learner has more knowledge than him.

Randomly select two learners  $W_i$  and  $W_j$ , where  $i \neq j$

If  $\text{objfun}(W_i) < \text{objfun}(W_j)$ , then the update equation is given by

$$W_i^{\text{new}} = P_f W_i^{\text{old}} + \text{rand}(W_i - W_j) \quad (51)$$

Else

$$W_i^{\text{new}} = P_f W_i^{\text{old}} + \text{rand}(W_j - W_i) \quad (52)$$

End ifAccept  $W_i^{\text{new}}$  if it gives a better solution.

### 6.3 Implementation of TLBO algorithm for weight optimization

Step 1: Initially, the features are taken as input to the model.

Step 2: Initialize the population size according to weights used in the model.

Step 3: Teacher phase:

According to TLBO algorithm in the teacher phase, the weights are selected randomly, and the mean of the population columnwise is calculated as  $W_i^{\text{best}} = M_{\text{new}}$  and then update the weights by using Eq. (26).

Step 4: Calculate the mean square error by using  $\text{MSE} = \frac{1}{n} \sum_{k=1}^n (D_k - y_k)^2$  where “ $D_k$ ” is the desired vector;  $y_k$  is the output of the model.

Accept  $W_i^{\text{new}}$  as the optimal solution.

Step 5: In the learner phase, randomly select two weights as  $W_i$  and  $W_j$ , where  $i \neq j$ , and then calculate  $MSE(W_i)$  and  $MSE(W_j)$ .

If  $MSE(W_i)$  is better than  $MSE(W_j)$

$$W_i^{\text{new}} = P_f W_i^{\text{old}} + \text{rand}(W_i - W_j) \quad (53)$$

Else

$$W_i^{\text{new}} = P_f W_i^{\text{old}} + \text{rand}(W_j - W_i) \quad (54)$$

End if Accept  $W_i^{\text{new}}$  as the optimal solution.

The procedure continues for all data points till the optimal solution for the weights of the LLRBFNN is obtained. It is found from the TLBO algorithm implementation that the teacher phase provides faster convergence speed and less computational time. However, the learner phase requires more computational burden than the teacher phase.

## 7 Results and discussion

For initial testing, power quality disturbance signals are synthesized using formulas given in Table 1. For the purpose of feature extraction, we have used synthetic signals generated using MATLAB software in a computer system having a configuration of dual core Intel processor, 2.93 GHz speed, and 1 GB RAM. Further, the relevant features are extracted using the cumulative sum average filter, and the statistical formulas are shown in Table 2. The extracted features shown in Table 3 are then fed as inputs to the LLRBFNN network sequentially. For each of the 10 classes of the simultaneous power quality disturbances, 100 data points are used making a total of 1000 for all the classes. In the case of PSO-based LLRBFNN, the error

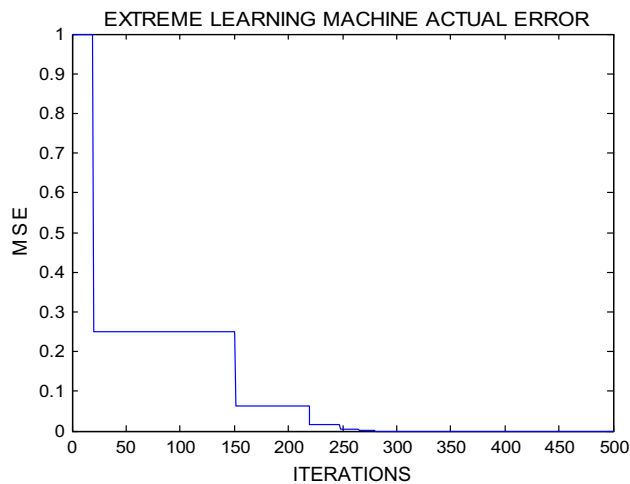
converges after near about 830 iteration steps. In TLBO-based LLRBFNN, it is found that after near about 450 iteration steps, convergence takes place, whereas in the proposed modified TLBO-based LLRBFNN, the error starts converging toward zero slowly after 160 iteration steps. It is found that the computational time for modified TLBO-based LLRBFNN is 11.5367 s. TLBO-based LLRBFNN and PSO-based LLRBFNN involve computational burden as 31.4275 and 43.4538 s, respectively. Table 4 shows the classification accuracy for the different networks. However, using the ELM approach, the mean square error convergences to zero in 300 iterations. Both ELM1- and ELM-based multiclass classifications were carried out for all the 1000 data points. The convergence characteristic of ELM1 is shown in Fig. 5. Similar characteristic is also obtained for ELM architecture. The MSE convergence to zero is obtained in nearly 300 iterations. The convergence characteristics for the TLBO and other evolutionary algorithms are shown in Fig. 6.

From Table 4, it is observed that the classification accuracy in the case of PSO-based LLRBFNN, the power signals Swell + Multiple spike and Sag + Multiple Notch is lesser in comparison with both the TLBO-based and LLRBFNN-based models. Modified TLBO-based LLRBFNN model provides good performance in terms of percentage of accuracy than the PSO-based LLRBFNN and TLBO-based LLRBFNN models for different multiple nonstationary power signals.

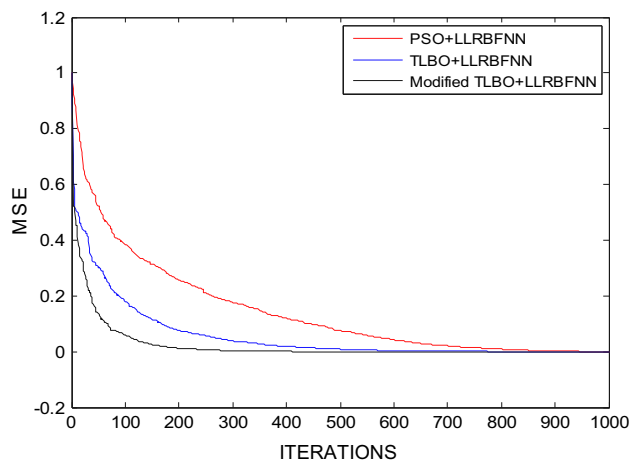
The accuracies of the proposed method are compared with other widely used power quality classifiers as shown in Table 5. Most of the power quality classifiers considered for comparison are based on time–frequency transforms such as DWT or ST hybridized with either neural network or fuzzy logic systems. Besides, there are a few power quality classifiers which use evolutionary techniques such

**Table 4** Multiple power signal classification accuracy in percentage

Sl. no	Simultaneous multiple power signal disturbances	Accuracy in percentage (%)				
		PSO-based LLRBFNN	TLBO-based LLRBFNN	Modified TLBO-based LLRBFNN	ELM1	ELMM
1	Multiple notch	98.32	99.40	99.85	98.6	99.3
2	Multiple spike + multiple notch	97.44	98.54	98.92	98.80	99.0
3	Multiple transient	97.87	98.47	99.87	98.68	99.5
4	Transient + multiple notch	97.34	97.85	99.17	98.45	98.90
5	Swell + multiple spike	96.42	97.51	98.47	98.30	98.60
6	Sag + multiple notch	95.46	97.91	98.86	98.20	98.75
7	Transient + notch + spike	97.18	98.23	98.97	99.0	99.2
8	Sag + harmonics	97.86	98.32	99.51	99.18	99.42
9	Swell + harmonics	97.75	98.15	98.93	98.96	99.12
10	Sag + transients	97.36	97.94	98.60	98.14	98.37
%	Average accuracy	97.30	98.23	99.11	98.63	99.02



**Fig. 5** Mean square error plot for the ELM



**Fig. 6** Mean square error plot using different evolutionary techniques

**Table 5** Comparison of proposed classifier performance with existing methods

Type of classifier	% Accuracy
Wavelet transform with neural network	94.4
Wavelet transform with neurofuzzy	96.5
Wavelet packets and support vector machines	97.25
Dyadic ST with DT	97.6
PSO-based LLRBFNN	97.14
TLBO-based LLRBFNN	98.29
Modified TLBO-based LLRBFNN	99.11
ELMM-based LLRBFNN	99.02

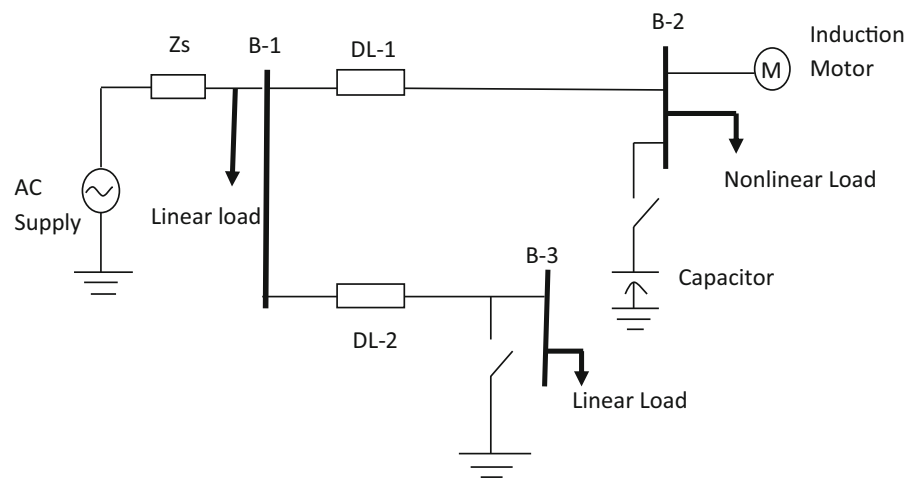
as decision trees (DT) along with DWT or ST. In comparison with the computationally intensive classifiers studied earlier, the new approach is simple and easy to

implement without extensive computations. Also, the accuracy of the proposed LLRBFNN trained by modified TLBO algorithm exhibits superior classification accuracy. The ELM-based LLRBFNN on the contrary produces slightly less accuracy in multiple power signal classification results in comparison with the TLBO approach but requires less computational overhead, thus making it suitable for processing large chunks power quality data. Also, ELMM (multiclass classification with 10 output neurons) produces better classification accuracy in comparison with the ELM1- (one output neuron) and the TLBO-based optimization. Both the algorithms were implemented in a computer with Core 2 Duo processor and 4-GB RAM, and it was found that the execution time for 1000 data points took 13.2 s in the case of TLBO learning-based localized RBF neural network. However, in the case of ELMM-based learning, it was found to be 2.25 s, which is much faster in comparison with the TLBO-based learning paradigm. The next section describes a laboratory implementation of the developed software for power quality disturbance detection. For measuring the learning speed of different algorithms, each of the power quality disturbance test cases comprised 100 specific observations in that class, and the overall speed was ascertained. This study revealed that the proposed ELMM-based learning scheme is nearly six times faster in comparison with the TLBO-based learning scheme for the LLRBFNN classifier.

## 8 Power quality test results from laboratory setup

To further verify the performance of the proposed algorithms, real power signal disturbance patterns are acquired from a test configuration shown in Fig. 7. A three-phase 400 V, 50 Hz power source supplies power to two linear loads comprising resistance and inductance (40 ohms, 70.54 mH) and a nonlinear voltage source-based converter load through two three-phase lines. A three-phase squirrel cage induction motor with a rating of 1.5 kW is also connected across the converter load. Different PQ disturbances were simulated by making faults in the line, additional load switching, and capacitor switching and varying the firing angle of the converter load. The feature extraction and classification algorithms were implemented on a 32-bit floating point digital signal processor from Texas instrument (TMS 320C6713 DSP Starter Kit (DSK) and embedded MATLAB coder. The internal program memory is structured so that a total of eight instructions can be fetched at every cycle. With a clock rate of 225 MHz, the C6713 processor is capable of fetching eight 32-bit instructions every  $1/(225 \text{ MHz})$  or 4.44 ns. Real-time classification of PQ waveforms was performed with the DSK using real-time



**Fig. 7** A practical laboratory setup for power quality disturbance detection**Table 6** Classification accuracy with real-time PQ events

Type of PQ disturbance	% Accuracy with TLBO	% Accuracy with ELM
Multiple notch	96.87	96.70
Multiple spike + multiple notch	96.5	96.2
Multiple transient	97.3	96.8
Transient + multiple notch	97.8	97.85
Swell + multiple spike	96.9	96.5
Sag + multiple notch	98.4	97.8
Transient + notch + spike	97.6	97.4
Sag + harmonics	97.38	97.41
Swell + harmonics	97.12	96.89
Sag + transient	97.20	97.12
Total	97.11	97.06

data exchange (RTDX) that allows for data exchange between the host PC and the target DSK, for analysis in real time without stopping the target. RTDX is achieved using the software tools TI C6000 DSP and code composer studio, to provide interface between PC and DSK. The host PC is used to monitor the classification results using the captured discrete data samples of the power quality disturbance signals, showing clearly a little reduction in classification accuracy due to some extraneous noise contaminating the data samples. The real-time data classification accuracy is shown in Table 6 from which it is observed that the TLBO-based classifier when used with practical data gives 97.11 % classification accuracy, while the ELM-based classifier produced 97.06 % accuracy, which is slightly lower than the former. Further, it can be seen that the classification accuracy gets degraded by nearly 2 % in comparison with the synthetic data. This discrepancy occurs due to the fact that the real-time data contain distortions and noise due to the measuring instruments used for the purpose. However, the classification accuracy is still high for simultaneous power quality disturbances.

## 9 Conclusion

In this paper, a combination of cumulative sum average filter and modified TLBO-based LLRBFNN model is proposed for the classification of multiple power quality disturbance patterns. The cumulative sum average filter has been designed in a recursive manner for multiple nonstationary power quality disturbance signals for feature extraction. It is found that the filter gives better localization of the power signal disturbances. Unlike the conventional neural architectures, the LLRBFNN requires a small number of nodes and uses a set of local linear models (a linear combination of weighted inputs and a bias) to provide a high-dimensional space for parsimonious interpolation of the nonlinear input–output relationship. The modified TLBO technique is used to adjust the weight parameters of the local linear models associated with radial basis functions and is computationally less involved and straightforward to implement. Mean square error plot shows that the modified TLBO-based LLRBFNN model requires less computation time in comparison with both the

TLBO-based LLRBFNN and PSO-based LLRBFNN models. On the other hand, the hybrid LLRBFNN and ELM are slightly less accurate than the TLBO-based LLRBFNN but requires less computation time, thereby making it suitable for large-scale data processing required for power quality disturbances. For better efficiency, it is required that the network should be presented with a particular set of input vectors belonging to the same class for training the network in a class by class manner. However, with multiple output nodes, ELM-based training produces better accuracy in comparison with one output neuron for multiclass classification. Regarding the practical data obtained in a laboratory environment, both the LLRBFNN-TLBO and LLRBFNN-ELMM classifiers perform almost similarly, while the former has a slight advantage in terms of accuracy over the latter. In comparison with the classification of synthetic power quality disturbance signals, the accuracy is found to be degraded by nearly 2 % due to the presence of noise in the signal.

## References

- Shaw SR, Laughman CR, Leeb SB, Lepard RF (2000) A power quality prediction system. *IEEE Trans Ind Electron* 47(3):511–517
- Zang H, Liu P, Malik OP (2003) Detection and classification of power quality disturbances in noisy conditions. *IEE Proc Gener Transm Distrib* 150(5):567–572
- Gaouda AM, Salama MMA, Kanoun SH, Chikhani AY (2002) Pattern recognition applications for power system disturbance classification. *IEEE Trans Power Deliv* 17(3):677–682
- Bollen MHJ, Gu IYH, Axelberg PGV, Styvaktakis E (2007) Classification of underlying causes of power quality disturbances: deterministic versus statistical methods. *EURASIP J Adv Signal Process* 2007(19):148
- Daubechies Ingrid (1990) The wavelet transform, time–frequency localization and signal analysis. *IEEE Trans Inf Theory* 36(5):961–1005
- Yang HT, Liao CC (2001) A de-noising scheme for enhancing wavelet-based power quality monitoring systems. *IEEE Trans Power Deliv* 16:353–360
- Chung J, Powers EJ, Lamoree J, Bhatt SC (2002) Power disturbance classifier using a rule based and wavelet-packet based Hidden Markov model. *IEEE Trans Power Deliv* 17:233–241
- Stockwell RG, Mansinha L, Lowe RP (1996) Localization of the complex spectrum: the S transform. *IEEE Trans Signal Process* 44(4):998–1001
- Faisal MF, Mohamed A (2009) Identification of multiple power quality disturbances using S-transform and rule based classification technique. *J Appl Sci* 9(15):2688–2700
- Nguyen T, Liao Y (2009) Power quality disturbance classification utilizing S-transform and binary feature matrix method. *Electr Power Syst Res* 79(4):569–575
- Looney CG (2002) Radial basis functional link nets and fuzzy reasoning. *Elsevier Sci Neurocomput* 48(4):489–509
- Albrecht S, Busch J et al (2000) Generalized radial basis function networks for classification and novelty detection: self organization of optimal Bayesian decision. *Neural Netw* 13:1075–1093
- Jayasree T, Devaraj D, Sukanesh R (2010) Power quality disturbance classification using hilbert transform and RBF neural networks. *Neurocomputing* 73(7–9):1451–1456
- Santoso S, Powers EJ, Grady WM, Parsons AC (2000) Power quality disturbance waveform recognition using wavelet-based neural classifier. I. Theoretical foundation. *IEEE Trans Power Deliv* 15:222–228
- Wang T et al (2000) A wavelet neural network for the approximation of nonlinear multivariable function. *Trans Inst Electr Eng C* 102-C:185–193
- Chen Y, Yang B, Dong J (2006) Time series prediction using a local linear wavelet neural network. *Elsevier Sci Neurocomput* 69:449–465
- Nekoukar V, Beheshti MTH (2010) A Local linear radial basis function neural network for financial time series forecasting. *Springer Appl Intell* 33:352–356
- Rao RV, Savsani VJ, Vakharia DP (2011) Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems. *Elsevier Sci Comput Aided Des* 43:303–315
- Rao RV, Savsani VJ, Vakharia DP (2012) Teaching–learning-based optimization: an optimization method for continuous nonlinear large scale problems. *Elsevier Sci Inf Sci* 183:1–15
- Tuo S, Yong L, Zhou T (2013) An improved harmony search based on teaching-learning strategy for unconstrained optimization problems. *Hindawi Math Probl Eng* 2013:1–29
- Rao RV, Patel V (2012) An elitist teaching-learning-based optimization algorithm for solving complex constrained optimization problems. *Int J Ind Eng Comput* 3:535–560
- Rao RV, Patel V (2013) Multi-objective optimization of heat exchangers using a modified teaching-learning based optimization algorithm. *Appl Math Model* 37(3):1147–1162
- Rao RV, Patel V (2013) Multi-objective optimization of two stage thermoelectric cooler using a modified teaching-learning based optimization algorithm. *Eng Appl Artif Intell* 26(1):430–445
- Rao RV, Savsani VJ, Balic J (2012) Teaching-learning-based optimization algorithm for unconstrained and constrained real parameter optimization problems. *Eng Optim* 44(12):1447–1462
- Huang G-B, Zhu Q-Y, Siew C-K (2006) Extreme learning machine: theory and applications. *Neurocomputing* 70(1–3):489–501
- Huang G-B, Chen L, Siew C-K (2006) Universal approximation using incremental constructive feedforward networks with random hidden nodes. *IEEE Trans Neural Netw* 17(4):879–892
- Huang G-B, Zhou H, Ding X, Zhang R (2012) Extreme learning machine for regression and multiclass classification. *IEEE Trans Syst Man Cybern Part B* 42(2):513–529
- Han F, Yao H-F, Ling Q-H (2013) An improved evolutionary extreme learning machine based on particle swarm optimization. *Neurocomputing* 116:87–93
- Huang GB, Wang DH, Lan Y (2011) Extreme learning machine-a survey. *Int J Mach Learn Cybernet* 2:107–122
- Mishra S, Panda G, Biswal B (2011) Use of multiple change detection in pattern recognition using relevant vector machine and moving sum average filter. *IEEE international conference on energy automation and signal, 28th–30th, ICEAS*
- Pradhan AK, Routray A, Mohanty SR (2006) A moving sum approach for fault detection of power system. *Electric Power Compon Syst* 34:385–399
- Nayak PK, Pradhan AK (2009) Transmission line fault classification using moving sum approach. *National conference on advances in computational intelligence applications in power, control, signal processing and telecommunication, NCACI*, pp 1–4