Journal of Control, Automation and Electrical Systems

Autonomous Neural Models for the Classification of Events in Power Distribution Networks

André E. Lazzaretti · Vitor H. Ferreira · Hugo Vieira Neto · Rodrigo J. Riella · Julio S. Omori

The original publication is available at www.springerlink.com. DOI 10.1007/s40313-013-0064-8

Abstract This paper presents a method for automatic classification of faults and transients in power distribution networks, based on voltage oscillographies of the distribution networks feeders. For signal preprocessing, the Discrete Wavelet Transform was used with the performances of several families of wavelet functions being compared. In the classification stage, three neural models were assessed: Multi-Layer Perceptrons, Radial Basis Function Networks, and Support Vector Machines. The models were trained autonomously, i.e., using automatic model selection and complexity control. Promising results were obtained using a set of simulations generated using the Alternative Transients Program. Initial results obtained for real data acquired from a set of oscillograph loggers installed in a distribution network are also presented.

Keywords Event Classification · Wavelet Transform · Neural Networks · Complexity Control · Autonomous Models.

André E. Lazzaretti and Rodrigo J. Riella
Institute of Technology for Development (LACTEC), Avenida Comendador Franco 1341, Curitiba-PR, Brazil
Tel.: +55-41-33616014
E-mail: lazzaretti@lactec.org.br

Vitor H. Ferreira
Federal University Fluminense (UFF), Rua Passo da Pátria 156, Niterói-RJ, Brazil

Hugo Vieira Neto
Federal University of Technology – Paraná (UTFPR), Avenida Sete de Setembro 3165, Curitiba-PR, Brazil

Julio S. Omori
Energy Company of Paraná (COPEL), Rua José Izidoro Biazetto 158, Curitiba-PR, Brazil

1 Introduction

Several events are responsible for changes in voltage and current waveforms in electrical power systems. In the particular case of voltage waveforms (oscillographic records) in a power distribution system, there is a range of events with relevant impact regarding equipment failure or consumer damage. These events involve changes in the waveforms, whose correct identification is desirable – in particular, the following events are of interest: short-circuits, lightning discharges, switching transients, and the start of heavy-duty engines.

In power distribution utilities, variations in voltage waveform cause increasing concern about supply disruptions and their duration, number of outages, voltage levels, frequency deviations, transients, and harmonic contents. In several countries there are standards that specify the expected quality of service for distribution networks – the extrapolation of product limits can incur in fines for power utilities, imposed by regulatory agencies. In this context, the Energy Company of Paraná (COPEL) and the Institute of Technology for Development (LACTEC) have developed several projects under the Research and Development Program from the Brazilian Electricity Regulatory Agency (ANEEL), with the objective of providing continuous monitoring of voltage waveforms in distribution networks (Riella et al., 2008; Lazzaretti et al., 2011).

A set of oscillograph loggers was designed and installed in a distribution network, in such a way that every time that some voltage waveform parameter exceeds some defined threshold, the referred waveform is logged for later analysis. A notorious problem with this method is the large amount of data logged by the monitoring systems, which ends up being unfeasible to be manually analyzed by maintenance and protection en-

gineers. This scenario suggests the need of an automatic data classification scheme, similar to the ones that have been used for event classification in power systems. In this case, the classification process is conducted in two steps: preprocessing and classification itself.

Preprocessing is performed for signal feature extraction in order to reduce the dimensionality of the input space. At this stage, a transformation of the input waveform (voltage or current) in the time domain to the frequency domain domain via Discrete Fourier Transform (DFT) or time-frequency domain via Discrete Wavelet Transform (DWT) is normally used (Costa et al., 2010). After that, some straightforward techniques can be applied, e.g., computing the signal energy in the new domain in order to extract the most relevant information from the signal (Lazzaretti et al., 2009) while maintaining the commitment to reduce dimensionality (Dong et al., 2009). It has been observed that the DWT has many advantages for signal characterization when compared to the DFT (Mallat, 1999), especially when power system signals are concerned (Costa et al., 2010).

At the supervised learning stage, the literature presents neural networks (Oleskovicz et al., 2003; Costa et al., 2010; Malathi et al., 2010), classification using fuzzy systems (Mahanty and Gupta, 2007), and hybrid models (Zhang and Kezunovic, 2007). Besides these, there are systems that perform classification using rules based on the analysis of the preprocessed signal (Dong et al., 2009). An important observation is that the final classifier is strongly dependent on the preselected parameters at the classifier construction stage, especially with regard to the structure of neural models (number of neurons). In (Demir, 2010; Manimala et al., 2011), techniques that optimize the parameters of Support Vector Machine (SVM) neural models for classification problems in power quality were applied, providing an autonomous feature (complexity control capability) for the training process. The chosen method is based on cross-validation, which is computationally intensive and substantially dependent on the length of the training set, but is applicable for the classification purposes in the present study. Shortcomings of cross-validation were fully analyzed in (Cataltepe et al., 1999).

Considering the aspects for classifier development mentioned earlier, the present study proposes the classification of events of different nature, such as short-circuits and transient events in a distribution system using voltage oscillographies. The proposed method is based on a DWT using several different wavelet functions and three different neural models, namely Multi-Layer Perceptron (MLP), Radial Basis Function (RBF), and SVM. This method is flexible and novel to the extent of the author's knowledge, and aims to be complementary to current methods in the following aspects:

- A new approach for training supervised classifiers autonomously is presented, achieving input selection and complexity control of the structure during the training stage (Ferreira and da Silva, 2007) and providing automatic generalization error control;
- The input selection, which is embedded in the training stage of each classifier, allows the selection of the most significant inputs, corresponding to the most prominent frequency bands of the DWT. At the end of this procedure, it is possible to characterize each event according to their frequency content, without requiring a dedicated step for such characterization;
- The autonomous features used for training make the extension of the method to new types of events possible, inherently maintaining the ability to generalize;
- All classification models (MLP, RBF and SVM) were designed to minimize user intervention in the main parameters of trained models, without the need of cross-validation or validation datasets. Therefore, the application to real data is facilitated, since specific adjustments for each different set of records are not required;
- The classification problem itself includes events of different nature and real data, which is observed in only a few works in this context.

The present paper is an extension of (Lazzaretti et al., 2009), where the MLP and SVM models were evaluated in simplified versions. The autonomous features of the classifiers, different types of signal preprocessing, and application to real data were not analyzed in (Lazzaretti et al., 2009) – in order to present new contributions in these senses, this paper is divided as follows. In Section 2, the theoretical aspects of the method are presented. Section 3 describes the experimental setup, from modeling to classification. In Section 4, the results obtained in the classification of events are presented, and finally, in Section 5, the conclusions are summarized and future work is outlined.

2 Theoretical Aspects

2.1 Discrete Wavelet Transform

The main motivation for DWT is the time-scale signal decomposition in frequency sub-bands, using orthonormal bases obtained from digital filter banks (Mallat, 1999). The input signal is processed by a series of high- and low-pass filters that separate frequency components in different subspaces. The construction of the filters

is based on the wavelet function properties, as defined in (Mallat, 1999). Thus, orthonormal bases of discrete wavelet functions are not only associated with the mother wavelet function, but also with the scale function – the mother wavelet function is associated with signal details (high-pass filters), whereas the scale function is associated to signal approximations (low-pass filters), forming an orthonormal basis.

Decomposition of the input signal into approximation and detail coefficients is the foundation of multiresolution analysis (Mallat, 1999), and can be done using a pair of finite impulse response (FIR) filters – a high- and a low-pass filter for the decomposition process, as well as their conjugates for the reconstruction process. In this way, the resolution analysis can be associated with filtering operations, and the scale analysis can be associated with downsampling and upsampling operations during decomposition and reconstruction, respectively. Once the signal is decomposed, the most prominent frequency components result in high amplitudes in the DWT sub-band coefficients that include these particular frequencies, retaining the temporal localization of the frequency components, differently from what occurs when the DFT is used.

The wavelet decomposition procedure presents good time domain resolution for high-frequency components and good frequency domain resolution for low frequency components. These properties constitute an alternative spectral representation to the one given by the DFT, using nonlinearly spaced frequency sub-bands that allows temporal localization of specific components of the signal under analysis, a very important characteristic for power distribution waveform analysis. More details of the DWT procedure can be found in (Mallat, 1999).

2.2 Multi-Layer Perceptron

Among the several methods proposed for specification and training of MLP networks, the Bayesian inference framework originally proposed by David J. C. Mackay in 1992 (Bishop, 1995) has been used in this study. This choice is mainly motivated by the concept of evidence maximization, once it is possible to use three hierarchical levels of inference. The process starts with the estimation of parameters followed by the estimation of hyperparameters, which allows the development of an input selection method. The last stage of the process is the selection of the most probable model for a given training dataset (Bishop, 1995).

Once the number of hidden layers, the number of neurons in each layer, and the type of neuron activation function of the MLP neural network are defined, the model training process, from a Bayesian inference point of view, is associated to the estimation of the parameter vector $\boldsymbol{w}$ that maximizes the posterior probability $p(\boldsymbol{w}|X, D)$. Considering classification problems with C mutually exclusive classes and defined by the set of N input-output pairs $\{X, D\}$, $X = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$, $D = \{\boldsymbol{d}_1, \ldots, \boldsymbol{d}_N\}$, $\boldsymbol{x} \in \mathbb{R}$, $x = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N]^t$, and $\boldsymbol{d} \in [0, 1]^C$, where 1-of-$C$ codification is used to represent class C of each input pattern $\boldsymbol{x}_i$ of length t, the posterior probability $p(\boldsymbol{w}|X, D)$ is represented by:

$$p(\boldsymbol{w}|X, D) = \frac{p(D|X, \boldsymbol{w})p(\boldsymbol{w})}{p(D|X)}, \tag{1}$$

where $p(D|X)$ is a normalization factor, $p(\boldsymbol{w})$ represents the prior probability of $\boldsymbol{w}$ and $p(D|X, \boldsymbol{w})$ is a likelihood function, which is related to the probability distribution of $\boldsymbol{x}_i$ being in a given class. As the classes are mutually exclusive, the prior probability $p(\boldsymbol{d}_i|\boldsymbol{x}_i, \boldsymbol{w})$ of a vector $\boldsymbol{x}_i$ belonging to class i, given the input pattern $\boldsymbol{x}_i$ and vector parameter $\boldsymbol{w}$, is given by:

$$p\left(\boldsymbol{d}_i|\,\boldsymbol{x}_i, \boldsymbol{w}\right) = \prod_{k=1}^{C} \left[f_k\left(\boldsymbol{x}_i, \boldsymbol{w}\right)\right]^{d_{ik}}, \tag{2}$$

with $f\left(\boldsymbol{x}_i, \boldsymbol{w}\right)$ being the MLP output as follows:

$$f_k\left(\boldsymbol{x}_i, \boldsymbol{w}\right) = y_{ik} = \frac{\delta_k}{\sum\limits_{i=1}^{k} \delta_i}. \tag{3}$$

In (3), y_{ik} represents the probability of $\boldsymbol{x}_i$ belonging to class C_k, with δ_k representing the output of the activation function of the k^{th} output layer neuron:

$$\delta_k = \varphi_{output}\left[\sum_{j=1}^{m} w_{kj}\varphi_h\left(\sum_{l=1}^{n} w_{jl}x_l + b_j\right) + b_k\right], \tag{4}$$

where $\varphi_{output} = exp(x)$. The function $\varphi_h(x)$ is associated to the output of each neuron in the hidden layer, represented by the logistic function $\varphi_h\left(x\right) : \Re \rightarrow [0, 1]$.

To choose $p(\boldsymbol{w})$, one must take into account the fact that different sets of weights are expected to present different behaviors during the estimation process. Thus, it is reasonable to use specific distributions for each set of weights, defining the prior distribution as follows:

$$p(\boldsymbol{w}) = \frac{1}{\prod\limits_{i=1}^{g}\left(\frac{2\pi}{\alpha_i}\right)^{\frac{M_i}{2}}} exp\left(-\frac{1}{2}\sum_{i=1}^{g}\alpha_i\|\boldsymbol{w}_i\|^2\right), \tag{5}$$

where $\boldsymbol{w}_i$ represents the set containing M_i parameters, α_i is the hyperparameter given by the inverse of the zero mean Gaussian distribution variance used for the prior representation of $\boldsymbol{w}_i$ and g is the number of parameter sets.

From the distributions $p(D|X, \boldsymbol{w})$ and $p(\boldsymbol{w})$, the posterior probability $p(\boldsymbol{w}|X, D)$ is given by:

$$p(\boldsymbol{w}|X, D) = \frac{1}{Z_S} exp(-S(\boldsymbol{w})), \tag{6}$$

with

$$S(\boldsymbol{w}) = \sum_{i=1}^{N}\sum_{k=1}^{C} d_{ik} \ln\left[f_k(\boldsymbol{x}_i, \boldsymbol{w})\right] + \frac{1}{2}\sum_{i=1}^{g}\left(\alpha_i \sum_{j=1}^{M_i} w_{ij}^2\right), \tag{7}$$

where $Z_s = \int exp(-S(\boldsymbol{w}))\mathrm{d}\boldsymbol{w}$ consists in a normalization factor. Hence, to maximize $p(\boldsymbol{w}|X, D)$, it is necessary to minimize $S(\boldsymbol{w})$, which consists in two terms. The first term, $\sum_{i=1}^{N}\sum_{k=1}^{C} d_{ik} \ln\left[f_k(\boldsymbol{x}_i, \boldsymbol{w})\right]$, regards empirical risk, which is related to how the model fits a given training set. The second term, $\frac{1}{2}\sum_{i=1}^{g}\left(\alpha_i \sum_{j=1}^{M_i} w_{ij}^2\right)$, is associated to weight decay regularization. Thus, the maximization of $p(\boldsymbol{w}|X, D)$ is equivalent to minimizing training error, taking into account model complexity (Bishop, 1995).

From a specific group of parameters, which defines the neural model, the relation between the hyperparameter α_i and the magnitude of $\boldsymbol{w}_i$ can be used to measure the relevance of each input in the model output estimation. This procedure is called automatic relevance determination. The inputs within the group of parameters $\boldsymbol{w}_i$ with smaller magnitude (higher α_i) can be considered insignificant. Therefore, it is necessary to define a limit for input relevance. In this study, the method presented in (Ferreira and da Silva, 2007) was used, which comprises starting the training process by inserting a random proof variable not correlated to the model output. This insertion provides the reference threshold α_0 for the random input at the end of the first stage of the training process, which can be used for input relevance determination. Thus, inputs with α_i exceeding α_0 can be considered irrelevant – these inputs are discarded from the final model.

Bayesian inference can also be used for model selection. One can use Bayes' rule to estimate the posterior probability $p(H_h|D)$ of the H_h hypothesis:

$$p(H_h|D) = \frac{p(D|H_h)\, p(H_h)}{p(D)}. \tag{8}$$

As $p(D)$ represents a normalization factor and assuming that all hypotheses H_h are equally probable, the evidence $p(D|H_h)$ can be used for model selection, with the model with higher posterior probability, i.e., higher evidence, being selected. Considering MLPs with one hidden layer with m neurons and a Gaussian distribution approximation for $\boldsymbol{\alpha}$, the logarithm of model evidence is given by:

$$\ln p(D|H_h) = -S(\boldsymbol{w}) - \frac{1}{2}\ln|\mathbf{A}(\boldsymbol{w})| + \frac{1}{2}\sum_{i=1}^{g} M_i\alpha_i + 2\ln m + \ln m! + \frac{1}{2}\sum_{i=1}^{g}\ln\left(\frac{2}{\gamma_i}\right), \tag{9}$$

where γ_i is the effective number of estimated parameters for the i^{th} set of weights $\boldsymbol{w}_i^* = [w_{i1}^*, \ldots, w_{iM}^*]^T$ and γ is the effective number of estimated parameters for the model, which are given by:

$$\gamma_i = \alpha_i \sum_{j=1}^{M_i}(w_{ij}^*)^2;\ \gamma = \sum_{i=1}^{g}\gamma_i. \tag{10}$$

The estimation of α_i and γ_i in (10) and (11) are the most probable values for the given training data. These values are calculated by an iterative method, based on evidence maximization for the hyperparameters. More details about this method can be found in (Ferreira and da Silva, 2007).

2.3 Radial Basis Function

Unlike the MLP network, RBF networks aim to address training using a curve fitting problem in a space of high dimensionality. The main goal of the RBF training is to minimize the two portions of the Tikhonov functional, namely training error and regularization term (Bishop, 1995). A possible solution uses the concept of Green functions $G(\boldsymbol{x}, \boldsymbol{x}_i)$, according to:

$$F_\lambda(\boldsymbol{x}) = \sum_{i=1}^{N} \boldsymbol{w}_i G(||\boldsymbol{x} - \boldsymbol{x}_i||). \tag{11}$$

The solution of the Tikhonov functional is given by the expansion of $F_\lambda(\boldsymbol{x})$ in terms of the Green function – in this case, given as a radial basis function. The weight vector $\boldsymbol{w}$ is given by the solution of:

$$\boldsymbol{w} = (\boldsymbol{G} + \lambda\boldsymbol{I})^{-1}\boldsymbol{d}. \tag{12}$$

In (12), $\boldsymbol{G}$ represents the Green matrix, which is responsible for applying the differential operator to various combinations of $\boldsymbol{x}_i$, called the expansion centers of $F_\lambda(\boldsymbol{x})$, and the input vector $\boldsymbol{x}$. The matrix $\boldsymbol{I}$ is the identity matrix and $\boldsymbol{d}$ is the output vector of the training set. In the special case of $\lambda = 0$, the solution is

$\boldsymbol{w} = \boldsymbol{G}^{+}\boldsymbol{d}$, where the matrix $\boldsymbol{G}^{+}$ is the pseudo-inverse of $\boldsymbol{G}$, given by:

$$\boldsymbol{G}^{+} = \left(\boldsymbol{G}^{T}\boldsymbol{G}\right)^{-1}\boldsymbol{G}^{T}. \quad (13)$$

The Green function normally used in RBF networks, which acts as a linear differential operator subject to the constraints of invariance to translation and rotation, is the multivariate Gaussian function given by:

$$G(\boldsymbol{x}, \boldsymbol{x}_i) = \exp\left(-\frac{1}{2\sigma_i^2}||\boldsymbol{x} - \boldsymbol{x}_i||^2\right). \quad (14)$$

Thus, the output of the network is given by the activation of the m_1 RBFs and their linear weights $\boldsymbol{w}_i$:

$$y = \sum_{i=1}^{m_1} w_i \phi_i(\boldsymbol{x}), \quad (15)$$

where the RBFs $\phi_i(\boldsymbol{x})$ basically use the Euclidean distances between an input pattern and the centers $\boldsymbol{t}_i$ of the basis functions as arguments:

$$\phi_i(\boldsymbol{x}) = G(||\boldsymbol{x} - \boldsymbol{t}_i||). \quad (16)$$

The RBF training can be defined with the estimation of linear output weights, the definition of the location of centers $\boldsymbol{t}_i$ in the hidden layer, and the selection of widths σ_l of the RBFs. In general, there are three basic approaches for this purpose: random selection of centers, self-organized selection of centers, and supervised selection of centers. A method that uses characteristics from two of these learning processes was applied, i.e., the proposed method basically consists of a self-organized selection of the centers, followed by a supervised learning stage in which the centers and the linear weights are readjusted from the first stage of the training process. This method can be best understood as follows:

1. In the first stage, or self-organized stage, a clustering method based on the Grow-When-Required (GWR) neural network (Marsland et al., 2002) is used, resulting in the automatic choice of the number of centers, as well as their respective locations.
2. In the second stage, or supervised learning stage, the location of the centers from the first stage are used as a starting point for an optimization method that uses a multi-objective genetic algorithm (Konaka et al., 2006) for the adjustment of centers and respective widths. Finally, the estimation of the linear weights is carried out by the pseudo-inverse method.

The use of the pseudo-inverse method is justified by its simplicity and efficacy. However, the use of this estimator takes into account that the regularization parameter is null, therefore disregarding the portion associated to network generalization. To circumvent this problem, an approach based on modifying the cost function by supervising the screening method of centers was adopted. In this approach, the function that initially only took the training error into account, now also considers network complexity by inserting a complexity control term that is conceptually similar to weight decay, as in:

$$\varepsilon = \frac{1}{N}\sum_{j=1}^{N}[t_i - f(x_i)]^2 + \frac{1}{N}\sum_{j=1}^{N} w_j^2, \quad (17)$$

where $f(x_i)$ represents the network output. Eq. (17) is optimized by multi-objective genetic algorithm (Deb et al., 2002), so that both minimizations – training error and network complexity – are guaranteed.

The last stage of the training process concerns input selection. An important observation regards the choice of basis functions. In this context, a variation of the Gaussian function in (14) was used, in such a way that a method of input selection can also be used in RBF network approach, as shown in:

$$G\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = exp\left(-\sum_{l=1}^{n}\frac{1}{2}\left(\frac{\sigma_l x_{il} - \sigma_l x_{jl}}{\sigma_i}\right)^2\right). \quad (18)$$

The σ_l parameters for each input are similar to the α_i parameters of the method based on the insertion of a random input in the MLP (described in section 2.2), and are used in order to achieve automatic input selection in RBF networks. By comparing σ_l to the threshold value σ_0 determined for the inserted random input, the relevant inputs of the model are determined, just like in the case of MLP networks. In the RBF network model, only inputs with σ_l exceeding σ_0 are considered relevant – the other inputs are discarded from the model.

2.4 Support Vector Machines

Support vector machines were developed based on a machine learning paradigm known as statistical learning. Unlike the classical approach for classification problems, statistical learning theory was developed to solve problems where the quantity of available data is limited, which represents a common characteristic in real applications (Bishop, 1995).

For classification problems, the learning process of SVMs is based on the concept of an optimum separation hyperplane, which maximizes the separation margin ρ between classes. The motivation for maximizing ρ is related to a complexity measurement known as Vapnik-Chervonenkis dimension (Bishop, 1995), whose upper limit is inversely proportional to ρ. The output of an SVM can be expressed as:

$$f(\boldsymbol{x}, \boldsymbol{W}, b) = \text{sgn}[\boldsymbol{W}^T \boldsymbol{\Phi}(\boldsymbol{x}) + b]; \quad (19)$$
$$\boldsymbol{W} = [W_1, \ldots, W_N]^T;$$
$$\boldsymbol{\Phi}(\boldsymbol{x}) = [\phi_1(\boldsymbol{x}), \ldots, \phi_N(\boldsymbol{x})]^T,$$

where $\boldsymbol{\Phi}(\boldsymbol{x}) : \mathbb{R}^n \rightarrow \mathbb{R}^N$ represents a nonlinear input mapping in feature space, with $\boldsymbol{W}$ and b being the parameters that define the hyperplane and sgn$[a]$ the sign function.

One way to formulate the maximization of the separation margin ρ for nonlinearly separable patterns is using the following restrict optimization problem:

$$\min_{\boldsymbol{W}, b, \boldsymbol{\xi}} E_s(\boldsymbol{W}) = \frac{1}{2}\boldsymbol{W}^T\boldsymbol{W} + C\sum_{i=1}^{N}\xi_i \quad (20)$$

s.t.

$$\begin{cases} d_i[\boldsymbol{W}^T\boldsymbol{\Phi}(\boldsymbol{x}) + b] \geq 1 - \xi_i \\ \xi_i \geq 0 \end{cases}, i = 1, 2, \ldots, N. \quad (21)$$

In (20), the first term, $\frac{1}{2}\boldsymbol{W}^T\boldsymbol{W}$, is responsible for complexity control of the model by means of maximization of ρ. The second term, $C\sum_{i=1}^{N}\xi_i$, is related to the classification error for the dataset. The variables ξ_i measure the deviation from $\boldsymbol{x}_i$ to the complete data classification. Input patterns that are correctly classified, i.e., that are in the correct side of the separation hyperplane and outside of ρ, have $\xi_i = 0$. Furthermore, input patterns with $0 \leq \xi_i \leq 1$ are correctly classified, i.e., they are in the correct side of the separation hyperplane, but inside the separation margin ρ. Patterns with $\xi_i > 1$ are in the incorrect side of the separation hyperplane and outside ρ, i.e., they are incorrectly classified.

The hyperparameter C is responsible for the balance between model complexity and goodness-of-fit to the training data and therefore is denominated as regularization parameter (Cherkassky and Mulier, 1998). The quadratic optimization problem in (20) can be solved by the Lagrange multipliers method, whose dual formulation is given by:

$$\max_{\boldsymbol{\alpha}} \Psi(\boldsymbol{\alpha}) = \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} d_i d_j K(\boldsymbol{x}_i, \boldsymbol{x}_j)\alpha_i\alpha_j \quad (22)$$

s.t.

$$\begin{cases} 0 \leq \alpha_i \leq C \\ \sum_{i=1}^{N}\alpha_i d_i = 0 \end{cases}, i = 1, 2, \ldots, N, \quad (23)$$

where $\boldsymbol{\alpha}$ represents the set of Lagrange multipliers and $K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ denotes the dot product kernel in feature space, as follows:

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = [\boldsymbol{\Phi}(\boldsymbol{x_i})]^T\boldsymbol{\Phi}(\boldsymbol{x_j}). \quad (24)$$

There are several types of kernel $K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ which abide to the conditions of Mercer's theorem (Bishop, 1995), such as polynomials, Gaussians and sigmoids. In this work, the Gaussian kernel was used because it allows automatic input selection (Ferreira and da Silva, 2007). The Gaussian kernel used is given by:

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = exp\left[-\sum_{l=1}^{N}\frac{\sigma_l^2}{2}(x_{il} - x_{jl})^2\right], \quad (25)$$

where σ_l^2, $l = 1, 2, \ldots, N$ are kernel hyperparameters.

The kernel definition and the vectors for which α_i^* is not equal to zero are called support vectors – they define the decision surface of the SVM as follows:

$$f(\boldsymbol{x}, \boldsymbol{W}, b) = \text{sgn}\left[\sum_{i=1}^{N_S}\alpha_i d_i K(\boldsymbol{x}_i, \boldsymbol{x}) + b\right], \quad (26)$$

where N_S is the number of support vectors.

The last stage of the SVM training process is related to hyperparameter specification, such as the regularization constant C and kernel hyperparameters σ_l^2. These parameters are commonly selected via cross-validation, or even user-specified, but in this work they were selected by means of minimization of the upper limit of the estimated generalization error in a leave-one-out approach (Ferreira and da Silva, 2007), by considering the hyperparameter C as the kernel parameter. It is important to notice that this limit is the least upper bound for the leave-one-out estimated error without the need of a validation set, which justifies its choice here. This upper limit was analytically developed in (Vapnik and Chapelle, 2000) and is conceptually founded on the span of support vectors:

$$T[f(\boldsymbol{x}, \boldsymbol{W}, b)] = \sum_{i=1}^{N_S}\alpha_i S_i^2, \quad (27)$$

where S_i^2 represents the extension of the i^{th} support vector, given by:

$$S_i^2 = \frac{1}{(\tilde{\mathbf{K}}^{-1})_{ii}}, \quad (28)$$

where $(\tilde{\mathbf{K}}^{-1})_{ii}$ represents the i^{th} diagonal element of the inverse matrix of $\tilde{\mathbf{K}}$, as follows:

$$\tilde{\mathbf{K}} = \begin{bmatrix} \mathbf{K}_{N_S} & \boldsymbol{u} \\ \boldsymbol{u}^T & 0 \end{bmatrix}, \quad (29)$$

where $\mathbf{K}_{N_S}$ is the dot product kernel matrix for all support vectors and $\boldsymbol{u} \in \mathbb{R}^{N_S}$ is the unit vector.

Given the multimodal characteristics of the function $T[f(\boldsymbol{x}, \boldsymbol{W}, b)]$ (Chapelle et al., 2002) and the computational effort for exhaustive search, genetic algorithms

were used for the minimization of $T[f(\boldsymbol{x}, \boldsymbol{W}, b)]$ as well as estimation of C and σ_l^2.

The analysis of the estimated σ_l allows the implementation of a method for measuring the relevance of the inputs in the calculation of the output, similarly to the RBF procedure described before, where the insertion of artificial random proof variables allows the estimation of empirical thresholds of relevance, which are used for identification of irrelevant inputs.

3 Experimental Setup

3.1 ATP Modeling

For dataset generation using an ATP model, the basic elements of a particular distribution substation from COPEL were considered, as well as the elements needed for event simulation. These elements are a substation transformer, capacitor bank, grounding transformer, bar feeders and the equivalent of the electric circuit up to the substation transformer, as can be seen in Fig. 1. Parameters for model simulation were based on (Lazzaretti et al., 2009).

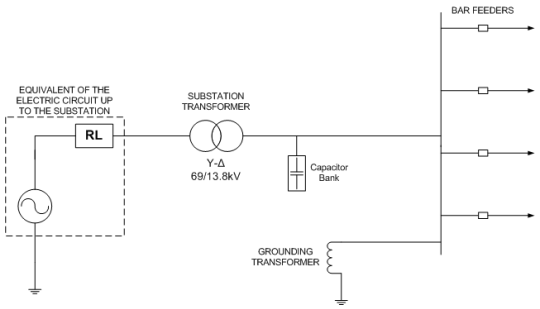

Fig. 1 Schematic diagram of the ATP simulated model.

Having the basic ATP model defined, the following events were simulated: single-line-to-ground faults, two-phase-line-to-ground faults, three-phase-line-to-ground faults, two-phase faults, three-phase faults, feeder circuit breaker switch-off, feeder reclosing and capacitor bank switching. The instant and location of the generated events were varied and, when concerning short-circuits, the fault resistance was also varied. Moreover, feeder load cycles were considered to simulate events for different load values. Finally, the typical harmonic distortion of the substation was considered in order to model the system operation as similar as possible to reality.

It must be emphasized that every event instance was automatically generated from a single ATP base file, adding up 6480 cases (432 cases for each event type, for separate training and test datasets). The instant when the events occurred was varied from 0° to 180°, as well as the load cycle and the distance where the events occurred with respect to the substation. In addition, the fault resistance was also varied, between 5 and 2000Ω in the training set, and between 20 and 1700Ω in the test set. The fault resistance values used were in the typical range of the events that happen in the real substation that was modeled.

3.2 Preprocessing

With the aim of extracting signal features with minimal loss of relevant information, preprocessing using the calculation of the energy in various sub-bands for different wavelet functions was carried out using a sampling rate of 7680 Hz. The voltage signals of the three phases were decomposed into 10 levels, obtaining 10 detail signals and the approximation signal to each phase, resulting in 33 signals in the wavelet domain. Once the sampling rate was 7680 Hz, ten levels of the DWT were used – with this number of levels, it was possible to cover most of the frequency contents of the signal with proper resolution.

After decomposing the input signals, the energy contents in each DWT level was calculated in order to achieve dimensionality reduction. This calculation was done based on the energy contents before and during the event, i.e., the ratio between the energy contents in the cycle in which the fault occurred and the energy contents in the cycle immediately before it was obtained for each DWT level. The energy ratio used in this work is given by the following expression:

$$E = \frac{1 + E_{DE}}{1 + E_{BE}} - 1, \tag{30}$$

where E_{DE} and E_{BE} are the wavelet energy levels of the cycles during and before the event, respectively.

The formulation in (30) is a modification of the direct energy ratio E_{DF}/E_{BF} (Oleskovicz et al., 2003), so that divisions by zero are avoided. Normalized energy levels were assumed, so that $E \in [-1, 1]$. By using this technique, it was possible to identify precisely those DWT sub-bands in which there was an increase or decrease in the energy contents, which was able to characterize the events. This ability was not observed when using the direct energy ratio E_{DE}/E_{BE}, which was very susceptible to noise (Lazzaretti et al., 2009). In order to use (30), it was necessary to normalize the sum of the energy contents in every DWT level to one.

4 Results and Discussions

4.1 Simulated Data

In this section the main results obtained for each classifier using different wavelet functions in the preprocessing stage are presented. For these results, the total accuracy for the training set and the total accuracy for the test set were considered, as well as the classifier generalization ability, model structure and input selection. Table 1 presents the classification results for the MLP.

Table 1 Results for the MLP.

Preproc.	Training Acc. (%)	Test Acc. (%)	Hidden Neurons	Selected Inputs
db1	87	87	19	33
db4	83	83	11	33
db8	**96**	**96**	**20**	**33**
db12	64	64	16	33
db15	95	95	18	33
coif1	93	92	13	33
coif3	93	93	8	33
coif5	**96**	**96**	**17**	**33**
sym2	**97**	**96**	**20**	**33**
sym4	95	94	17	33
sym8	87	86	19	33
bior1.1	95	93	19	33
bior1.5	78	77	19	33
bior2.2	87	87	20	33
bior2.8	91	90	18	33
bior3.1	89	88	19	33
bior3.9	89	89	18	33
bior4.4	93	93	19	33
bior5.5	89	88	20	33
bior6.8	91	90	19	33

In terms of total accuracy, it was observed that for the test set, the wavelet functions *db8*, *coif5* and *sym2* yielded the best performance, i.e., 96% of accuracy. The performance for other preprocessing wavelet functions was similar, except for the preprocessing based on *db12*, which yielded poor accuracy when compared to the average accuracy of the other wavelet functions.

Regarding generalization ability, the use of weight decay, shown in Eq. (8), ensured that a trade-off between training error minimization and complexity control was maintained during the process of parameter and hyperparameter estimation. Comparative analysis of the training and test accuracy provides an idea of the generalization ability – in general, the MLP has shown good generalization ability for all wavelet functions.

The structure defined in the hidden layer of MLP networks is the result of the model selection through evidence maximization. The networks were tested by varying the number of neurons in the hidden layer and the structure selection was based on evidence calculation, i.e., the most likely model for the training data was the model with the highest evidence. All the preprocessing wavelet functions were assessed in the range of one to 25 neurons in the hidden layer.

As for input selection, the automatic relevance determination method kept all the inputs for all wavelet functions, i.e., no irrelevant input variables were found. The comparative analysis between $\boldsymbol{\alpha}_i$ obtained for the input variables and α_0 of the inserted random variable, showed that the irrelevance level associated to the random variable was not sufficient to eliminate any inputs.

Table 2 presents the classification results using the RBF approach. The wavelet functions with the best performance were *db4* and *db12*, with an average accuracy of 90% for the test set. In the RBF networks, it was observed that the performance for the test set was quite close to the performance for the training set for all preprocessing wavelet functions. This observation confirms the efficacy of multi-objective optimization during the training process, including a factor that is indirectly responsible for complexity control of the network, as shown in Eq. (17).

Table 2 Results for the RBF.

Preproc.	Training Acc. (%)	Test Acc. (%)	Hidden Neurons	Selected Inputs
db1	63	62	223	14
db4	**92**	**90**	**610**	**33**
db8	89	87	532	27
db12	**92**	**91**	**536**	**29**
db15	79	77	393	16
coif1	78	76	352	18
coif3	81	80	337	18
coif5	76	75	309	18
sym2	86	83	463	25
sym4	84	83	347	18
sym8	78	77	381	18
bior1.1	74	73	252	18
bior1.5	87	85	493	29
bior2.2	70	69	350	18
bior2.8	78	77	324	18
bior3.1	70	69	409	18
bior3.9	82	80	273	18
bior4.4	88	85	473	26
bior5.5	73	72	344	18
bior6.8	86	85	419	21

The structure selection was done by a GWR neural network, which provides an automatic selection of the number of neurons in the hidden layer of the RBF network, as well as their respective center locations. The resulting number of RBF neurons in the hidden layer was relatively high – 380 hidden neurons in average. This quantity was found to be directly related to the

GWR network parameters used, which were selected according to the suggestions in (Marsland et al., 2002).

The number of neurons inserted during the training of the GWR network can be considered as the number of new features – novelties – identified during the presentation of input patterns. These novelties can be interpreted as patterns that are not close enough to any center already established in the GWR space, causing a new center to be allocated. Each new center spans a new region in the GWR space, which accounts for certain characteristics in the input patterns.

The method for input selection adopted in the RBF networks proved to be very efficient in the number of selected inputs, resulting in an average selection of 18 inputs to the classifier. However, the interpretation regarding frequency sub-bands has very particular characteristics for each wavelet function.

For *db4*, the method of input selection found no irrelevant variables, taking into account all 33 frequency bands of the three-phase voltage signals. For *db12*, the B-phase detail signals *D2*, *D5*, *D8* and C-phase *D10* were considered irrelevant for the classification process. It is important to mention that *db12* obtained a slightly higher performance than *db4* for the test set, in spite of discarding four input features.

The input selection for the *db12* wavelet function has a very specific characteristic for the chosen frequency sub-bands. In this case, no DWT energy level corresponding to the A-phase was eliminated, unlike the B-phase, where three different frequency sub-bands were excluded. The eliminated sub-bands did not provide a specific characterization of the simulated events, considering that the input selection basically happened in the B-phase despite the fact that the events were generated in a balanced manner for all three phases.

For the *coif5* wavelet function, the detail signal *D10* and the approximation signal *A1* were automatically eliminated from the decomposition of voltage signals of all phases. In this process, the frequency range from 0 to 7 Hz was not considered relevant for the classifier, demonstrating that the frequency sub-bands in question did not characterize the events of interest, which is consistent with practical observations.

A common feature for all wavelet functions is that high frequency sub-bands were never eliminated. The main features of the transient events under analysis occur especially in the first two levels of decomposition (*D1* and *D2*). Moreover, the signal detail *D10* was excluded in most of the wavelet functions. The frequency in this level was not associated with any particular feature of the simulated events, because it was in a frequency range below the fundamental frequency of the voltage signals (60 Hz).

Table 3 presents the classification results using the SVM approach, in which performances for each wavelet function was very similar for training and test sets. Among the various wavelet functions, *db1* and *coif1* yielded 98% of accuracy for the training set, and *db1* and *bior2.8* yielded 97% of accuracy for the test set.

Table 3 Results for the SVM.

Pre-proc.	Training Acc. (%)	Test Acc. (%)	Support Vectors	Selected Inputs
db1	**98**	**96**	**74**±122	**25**±6
db4	**97**	**97**	**80**±106	**23**±7
db8	**96**	**96**	**78**±103	**24**±7
db12	95	94	96±114	24±6
db15	92	90	102±130	24±6
coif1	**98**	**96**	**74**±81	**23**±7
coif3	96	94	91±112	25±6
coif5	**96**	**95**	**65**±92	**25**±6
sym2	97	94	80±105	25±6
sym4	97	96	78±101	23±6
sym8	96	95	80±106	24±6
bior1.1	97	95	81±127	23±7
bior1.5	96	93	78±112	22±7
bior2.2	97	95	85±117	25±7
bior2.8	**97**	**97**	**94**±120	**24**±6
bior3.1	97	94	88±117	24±6
bior3.9	95	94	78±109	24±7
bior4.4	96	94	77±101	24±7
bior5.5	96	95	83±115	25±6
bior6.8	96	95	87±114	25±7

Once training of the SVM network is based on the structural risk minimization method, it is guaranteed that the trade-off between training error and generalization ability is maintained during the training process of the model. This result is evident by comparing the performances obtained for the training and testing sets in this approach. In addition, the training method takes into account the minimization of an upper limit of the generalization error estimated via leave-one-out cross-validation without the need of a validation set and computation of the leave-one-out error itself. One of the results of this method is the automatic selection of the regularization parameter C and kernel parameters σ_l, with the possibility of automatic input selection as well.

In Table 3, mean values and standard deviations for the selected inputs are presented because the SVM training is done in pairs of classes (one-versus-one approach). In the case of support vectors, the method selected an average of 80 support vectors for the several SVM networks in their various combinations. Once these values refer to pairs of classes, their interpretation is very particular according to each case. The complexity among several models with different preprocessing wavelet functions is quite different, i.e., the number

of support vectors selected for the various models and their different pairs of classes differed significantly from the overall mean of support vectors.

Regarding the number of selected inputs, the SVM method was efficient, considering an average of 24 inputs as relevant to the classifiers. Unlike the interpretation applied to the RBF networks, in the case of SVM networks the main frequency sub-band selection analysis becomes unfeasible because relevant sub-bands are only valid for specific pairs of classes.

When analyzing the average accuracy of all forms of preprocessing for the three neural networks tested, it was observed that the SVM approach achieved higher performance than the other ones. Its average accuracy for the test set was 94%, while that of the MLP and RBF networks was 90 and 77%, respectively.

In all assessed models, the complexity control was efficient, allowing good generalization ability. Overall, the three models presented very similar results in terms of generalization, showing no limitation to their application. With respect to autonomous characteristics, the three models were designed with a particular technique for structure definition, aiming at little (or less impacting) user intervention during the training process, because each model needed some manual parameter definitions in at least one stage.

For input selection, the RBF model presented a very promising estimation for the most relevant frequency sub-bands for classification. This estimation provided an important characterization of the analyzed events and highlights that the DWT frequency division is able to characterize the signals under analysis very well. The initial results for this technique could form the basis for further investigation of relevant event features in distribution networks, facilitating the understanding, identification and location of occurrence of these events.

Also, a large number of wavelet functions were compared at the preprocessing stage for all neural models. The comparison shows that all wavelet families have very similar average performance. This fact indicates that specific features of a single wavelet function may not exist in order to justify its choice. However, one possible choice can be based on the wavelet function that yielded better average performance for all three models – in this case, the *db8* wavelet is suggested. As an additional criterion, it is possible to select wavelets with associated filters that have less coefficients, in order to reduce the processing burden at this stage.

4.2 Real Data

For experiments using real data, a database generated by the four oscillography loggers presented in (Lazzaretti et al., 2011) was used. Those four logging systems were designed to measure fast electromagnetic transients on energized distribution networks on both medium and low voltage circuits. They were installed on class 15 kV feeders during eight months. Of the 340 recorded events with significant variations on the waveform, 57% were related to lightning discharges, 20% were classified as short-circuits, 12% as feeder reclosings and 11% as others events.

Due to the large difference in the number of samples per class, it was decided to include simulated data from single-line-to-ground faults and automatic feeder reclosings, balancing the number of samples per class. Thus, 26 training samples and 15 test samples were obtained for the following classes: single-line-to-ground fault (Phase A), single-line-to-ground fault (Phase B), single-line-to-ground fault (Phase C), automatic feeder reclosing, and lightning-related transients. Half of the events generated by simulation were used to compose the training set and the other half, the test set.

All the three-phase voltage signals were decomposed using the approach outlined in the section 3.2. However, in this case the signals were sampled with approximately 500 kHz in order to acquire lightning-induced transients properly. Furthermore, it was chosen to decompose the signals of each voltage phase into 12 levels (11 for signal detail and one for signal approximation), leaving all frequencies below 120 Hz in the signal approximation. Thus, the input vector was composed by 36 features.

Using the SVM model and the *db8* wavelet, **78%** of global accuracy was obtained for test set. This performance is quite different from the performance obtained for simulated data only. One possible explanation for that is because of the small number of real patterns available for the development of the automatic classification process. In addition, the authors were not able to find any references in the literature for classification performance on real waveforms with distinct nature events, as presented here. Therefore, it can be stated that these preliminary results for real data were very satisfactory.

5 Conclusions

Although the methods for event classification are quite widespread in the literature, especially those regarding power quality, peculiarities of each system demand specific classifier development. In this work, the most difficult task was to use the same classifier for events of different nature. The performance of the proposed classification methods suggest that such an approach is feasible, even for a large number of event classes.

With respect to the neural models themselves, this work demonstrates the use of autonomous training processes. Autonomy, as defined here, means automatic complexity control and automatic selection of model structure, namely input selection and definition of the number of hidden neurons. The autonomous strategy for neural network specification and training without the use of a validation set and specific adjustments (e.g., cross-validation) is a novel approach for event classification in electric distribution networks.

In future work, new real data will be acquired and processed, since the logging systems mentioned before (and also new logging systems) are being installed on a distribution network. It will be possible to check the performance of the proposed model for a database with more samples and, eventually, more classes. It is expected that with this new dataset the results for real data acquired in a distribution network can be improved when compared to the results obtained so far.

Acknowledgements The authors would like to thank Copel and ANEEL for their funding of this research, and Cleverson L. S. Pinto for his contribution in this work.

References

C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

Z. Cataltepe, Y. S. Abu-Mostafa, and M. Magdon-Ismail. No free lunch for early stopping. *Neural Computation*, 11:995–1009, 1999.

O. Chapelle, V. Vapnik, O. Bousquet, and S. Mukherjee. Choosing multiple parameters for support vector machines. *Machine Learning*, 46(1-3):131–159, 2002.

V. Cherkassky and F. F. Mulier. *Learning from data: Concepts, theory and methods*. Wiley, 1998.

F. B. Costa, N. S. D. Brito, and B. A. Souza. Detecção de faltas evolutivas e múltiplos distúrbios em registros oscilográficos baseada na transformada wavelet discreta. *Controle e Automação*, 21:173–184, 2010. In Portuguese.

K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Trans. on Evolutionary Computation*, 6:182–197, 2002.

Y. Demir. Wavelet-based feature extraction and selection for classification of power system disturbances using support vector machines. *Electric Power Systems Research*, 80:743–752, 2010.

X. Dong, W. Kong, and T. Cui. Fault classification and faulted-phase selection based on the initial current traveling wave. *IEEE Trans. on Power Delivery*, 24: 552–559, 2009.

V. H. Ferreira and A. P. A. da Silva. Toward estimating autonomous neural network load forecasters. *IEEE Trans. on Power Systems*, 22(4):1554–1562, 2007.

A. Konaka, D. W. Coit, and A. E. Smith. Multi-objective optimization using genetic algorithms: A tutorial. *Reliability Engineering and System Safety*, 15:992–1007, 2006.

A. E. Lazzaretti, V. H. Ferreira, H. Vieira Neto, R. J. Riella, and J. Omori. Classification of events in distribution networks using autonomous neural models. In *Proc. of the 15th Int. Conf. on Intelligent System Applications to Power Systems*, 2009.

A. E. Lazzaretti, M. A. Ravaglio, L. F. R. B. Toledo, J. A. Teixeira-Júnior, P. M. Rojas, and C. L. S. Pinto. Measurements of lightning discharges in overhead distribution feeders. In *Anais do XI Simpósio Internacional Proteção Contra Descargas Atmosféricas*, 2011. In Portuguese.

R. N. Mahanty and P. B. D. Gupta. A fuzzy logic based fault classification approach using current samples only. *Electric Power Systems Research*, 77:501–507, 2007.

V. Malathi, N. S. Marimuthu, and S. Baskar. Intelligent approaches using support vector machine and extreme learning machine for transmission line protection. *Neurocomputing*, 73:2160–2167, 2010.

S. Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, 2nd edition, 1999.

K. Manimala, K. Selvi, and R. Ahila. Hybrid soft computing techniques for feature selection and parameter optimization in power quality data mining. *Applied Soft Computing*, 11:5485–5497, 2011.

S. Marsland, J. Shapiro, and U. Nehmzow. A self-organising network that grows when required. *Neural Networks*, 15:1041–1058, 2002.

M. Oleskovicz, R. K. Aggarwal, and D. V. Coury. O emprego de redes neurais artificiais na detecção, classificação e localização de faltas em linhas de transmissão. *Controle e Automação*, 14:138–150, 2003. In Portuguese.

R. J. Riella, V. P. Ferrari, G. Paulillo, M. R. Ortega, and J. G. Pereira. Desenvolvimento de um sistema de monitoramento contínuo da qualidade da energia elétrica para subestações de distribuição. In *Anais do XVIII Seminário Nacional de Distribuição de Energia Elétrica*, 2008. In Portuguese.

V. Vapnik and O. Chapelle. Bounds on error expectation for support vector machines. *Neural Computation*, 12(9):2013–2036, 2000.

N. Zhang and M. Kezunovic. A real time fault analysis tool for monitoring operation of transmission line protective relay. *Electric Power Systems Research*, 77:361–370, 2007.