Published in IET Science, Measurement and Technology
Received on 17th January 2008
Revised on 29th April 2008
doi: 10.1049/iet-smt:20080009

ISSN 1751-8822

Artificial neural network optimisation methodology for the estimation of the critical flashover voltage on insulators

G.E. Asimakopoulou V.T. Kontargyri G.J. Tsekouras
F.E. Asimakopoulou I.F. Gonos I.A. Stathopulos

Electric Power Department, School of Electrical and Computer Engineering, National Technical University of Athens, 9, Iroon Polytecheniou Street, 157 80, Zografou Campus, Athens, Greece
E-mail: igonos@softlab.ntua.gr

Abstract: To describe an artificial neural network (ANN) methodology in order to estimate the critical flashover voltage on polluted insulators is the objective here. The methodology uses as input variables characteristics of the insulator such as diameter, height, creepage distance, form factor and equivalent salt deposit density, and it estimates the critical flashover voltage based on an ANN. For each ANN training algorithm, an optimisation process is conducted regarding the values of crucial parameters such as the number of neurons and so on using the training set. The success of each algorithm in estimating the critical flashover voltage is measured by the correlation index between the experimental and estimated values for the evaluation set, and finally the ANN with the correlation index closest to 1 is specified. For this ANN and the respective algorithm, the critical flashover voltage of the test set insulators is estimated and the respective confidence intervals are calculated through the re-sampling method.

1 Introduction

During the last few years, an increasing demand on energy consumption is observed, leading to an increase on the distributed power and, on occasion, to building new networks in order to support this demand. The consequent higher standards regarding the strength of the insulators comes as no surprise, since insulators are one of the most important components that greatly affect the reliability of the power system.

Heavy atmospheric pollution creates an electrolytic layer on the surface of the insulator. When combined with fog or rain, a leakage current flows along the conducting layer. Additionally, surface pollution and non-uniform potential distribution along the insulator surface cause glow discharges or quasi-stable arcs to appear. When the applied voltage exceeds a critical value, these discharges or quasi-stable arcs elongate through successive root formation over the polluted insulator surface until the flashover causes complete bridging. Therefore it is important to monitor the insulator's condition so as to ensure that maintenance takes place in due time.

For this purpose, several researches have been done in which mathematical and physical models are used [1, 2], experiments have been conducted [3, 4] or simulation programmes have been developed [5–7]. New technologies for the qualitative control of the insulators, such as artificial neural network (ANNs) and fuzzy logic, have been developed. Neural network algorithms have been successful in estimating the equivalent salt deposit density, by using information regarding temperature, humidity, pressure, rainfall and wind speed as input data, with the intention of establishing an effective maintenance policy.

Another ANN [8] has been applied in order to estimate the critical flashover voltage on polluted insulators. In [9], an ANN has been used in order to estimate the time-to-flashover when the applied voltage, the creepage length and the resistance per unit length are given. The experimental data were obtained by studies performed on a flat plate model for a polluted insulator under a power frequency voltage. In [10], an ANN was trained to deduce whether or not a breakdown is imminent using data collected from two pollution-related

monitoring devices. Several other ANNs have been developed in order to analyse the insulator surface tracking on solid insulators [11], to estimate the leakage current on silicone rubber insulators [12, 13] and the partial discharge inception voltage [14]. Another study [15] presents a multilayer ANN model that classifies the development of the arc gradient into three stages.

This paper describes a methodology that was developed for the estimation of the critical flashover voltage of polluted insulators by using an (ANN) and selecting the optimum training algorithm and the respective parameters. Its basic advantages are:

- the optimisation process to determine the most suitable parameters for each ANN training algorithm (such as the number of the neurons in the hidden layer, the kinds and the parameters of the activation functions),

- the comparison process between different ANN training processes (such as the deepest descent back-propagation (BP), adaptive (BP) conjugate gradient algorithm, scaled conjugate gradient algorithm, resilient algorithm, Newton algorithm),

- the use of three sets for the total optimisation process: the training set, which is used for the training of each ANN algorithm, the evaluation set, which is used for the selection of the ANN and the respective parameters with the biggest correlation index R^2 between the actual and estimated values of the critical flashover voltage, and the test set, which is the final set under estimation and proves the generalisation ability of the proposed methodology,

- the use of a small number of experimental results that are included in the training and evaluation set through the extension of the training set using the elder mathematical model [16],

- the calculation of the confidence intervals using the re-sampling method [17], so that the width of the confidence interval of the critical flashover voltage for each insulator is calculated beyond the estimated value.

The methodology is successfully implemented for the estimation of the critical flashover voltage of 24 artificially polluted insulators, whereas the training and evaluation sets are formed by 140 vectors from the mathematical model [16] and other four experimental vectors (different from the test set).

2 Proposed ANN methodology for the estimation of the critical flashover voltage

The estimation of the critical flashover voltage of the polluted insulators is achieved by applying an ANN methodology through the proper selection of the training algorithm and the optimisation of the respective parameters. This methodology has the following basic steps and its flow chart is shown in Fig. 1.

2.1 Data selection

The input variables are the maximum diameter D_m (in cm), the height H (in cm), the creepage distance L (in cm), the form factor F of the insulator and the layer conductivity σ_s (in μS), where as the output variable is the critical flashover voltage U_c (in kV). The data used for the training, evaluation and testing of the ANN were selected from various sources. Some of them were acquired by experiments that were carried out in the High Voltage Laboratory of Public Power Corporation's Testing, Research and Standards Centre in Athens [18] according to the IEC standard 507:1991 [19]. Artificial pollution was applied on the insulators before conducting the test to determine the critical flashover voltage. Apart from this set of experimental measurements, other measurements were also used, from experiments performed by Zhicheng and Renyu [20] and Sundararajan *et al.* [21].

A mathematical model was also used to enrich the training data [16]. The equivalent circuit for the evaluation of the critical flashover voltage is comprised by a partial arc spanning over a dry zone in series with a resistance that represents the pollution layer, as shown in Fig. 2, where V_{arc} is the arcing voltage, R_p the resistance of the pollution layer and U a stable voltage supply source. In this model, the critical flashover voltage is given by the following formula

$$U_c = \frac{A}{n+1} \cdot (L + \pi \cdot n \cdot D_m \cdot F \cdot K) \cdot (\pi \cdot A \cdot D_m \cdot \sigma_s)^{-(n/(n+1))} \quad (1)$$

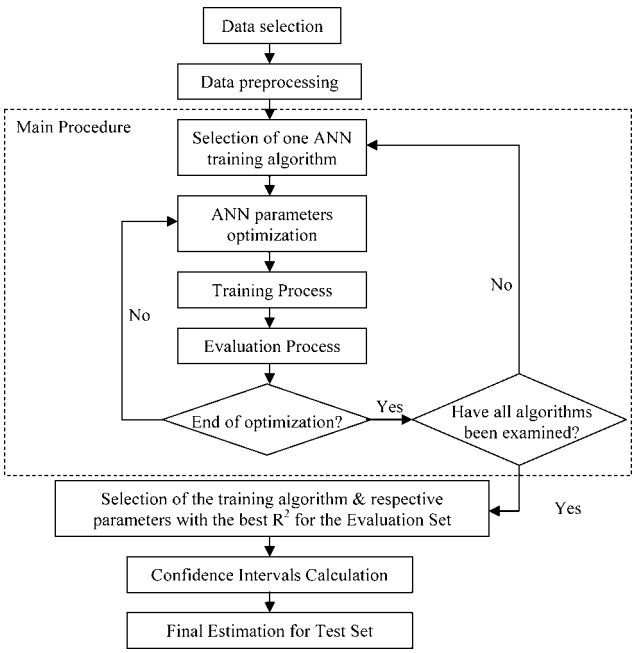

Figure 1 *Flowchart of the ANN optimisation methodology for the estimation of the critical flashover voltage of insulators*

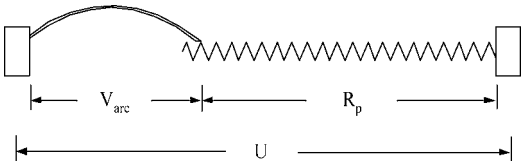

Figure 2 *Equivalent circuit for the evaluation of the flashover voltage*

The form factor of an insulator is determined from the insulator dimensions. For graphical estimation, the reciprocal value of the insulator circumference $(1/p)$ is plotted against the partial creepage distance l counted from the end of the insulator up to the point reckoned. The form factor is given by the area under this curve and calculated according to the formula [22]

$$F = \int_0^L \frac{dl}{p(l)} \tag{2}$$

The arc constants A and n have been calculated using a genetic algorithm model [23] and their values are $A = 124.8$ and $n = 0.409$. The surface conductivity σ_s (in Ω^{-1}) is given by the following type

$$\sigma_s = (369.05 \cdot C + 0.42) \cdot 10^{-6} \tag{3}$$

where C is the equivalent salt deposit density in mg/cm^2.

The coefficient of the resistance of the pollution layer K in the case of cap-and-pin insulators is given by

$$K = 1 + \frac{n+1}{2 \cdot \pi \cdot F \cdot n} \cdot \ln\left(\frac{L}{2 \cdot \pi \cdot R \cdot F}\right) \tag{4}$$

where R is the radius of the arc foot (in cm) and is given by

$$R = 0.469 \cdot \left(\pi \cdot A \cdot D_m \cdot \sigma_s\right)^{1/(2(n+1))} \tag{5}$$

2.2 Data preprocessing

Data are examined for normality, in order to modify or delete the values that are obviously wrong (noise suppression). Because of the great nonlinearity of the problem, nonlinear activation functions are preferably used. In that case, saturation problems may occur. These problems can be attributed to the use of sigmoid activation functions that present nonlinear behaviour outside the region $[-1, 1]$. To avoid saturation problems, the input and the output values are normalised as shown by the following expression

$$\hat{x} = a + \frac{b-a}{x_{max} - x_{min}}(x - x_{min}) \tag{6}$$

where $\hat{x}$ is the normalised value for variable x; x_{min}, and x_{max} are the lower and the upper values of variable x; and a and b are the respective values of the normalised variable.

2.3 Main procedure

For each training algorithm (Table 1), the respective parameters of the network are optimised through a set of

Table 1 Description of ANN's training algorithms

No.	Description of ANN's training algorithms
1	stochastic training with learning rate and momentum term (decreasing exponential functions) [9]
2	stochastic training, use of adaptive rules for the learning rate and the momentum term [9]
3	stochastic training, constant learning rate [9]
4	batch mode, constant learning rate [9]
5	batch mode with learning rate and momentum term (decreasing exponential functions) [9]
6	batch mode, use of adaptive rules for the learning rate and the momentum term [9]
7	batch mode, conjugate gradient algorithm with Fletcher–Reeves equation [24, 25]
8	batch mode, conjugate gradient algorithm with Fletcher–Reeves equation and Powell–Beale restart [24–26]
9	batch mode, conjugate gradient algorithm with Polak–Ribiere equation [24, 27]
10	batch mode, conjugate gradient algorithm with Polak–Ribiere equation and Powell–Beale restart [24, 26, 27]
11	batch mode, scaled conjugate gradient algorithm [28]
12	batch mode, resilient algorithm [29]
13	batch mode, quasi-Newton algorithm [30]
14	batch mode, Levenberg–Marquardt algorithm [31, 32]

doi: 10.1049/iet-smt:20080009

trials (such as the neurons can be varied from 2 to 25 or the activation function can be the linear, the logistic or hyperbolic). For each combination of the parameters, the ANN training process is actualised using the respective training set. After the respective convergence, the R^2 index between the experimental and estimated values of the critical flashover voltage for the evaluation set is calculated and the biggest index is chosen as the best one with the respective parameters. It is noted that

$$R^2 = r^2_{y-\hat{y}} = \frac{\left(\sum_{i=1}^{n}\left((y_i - \bar{y}_{\text{real}}) \cdot (\hat{y}_i - \bar{y}_{\text{est}})\right)\right)^2}{\sum_{i=1}^{n}\left((y_i - \bar{y}_{\text{real}})^2 \cdot \sum_{i=1}^{n}(\hat{y}_i - \bar{y}_{\text{est}})\right)^2} \quad (7)$$

where y_i is the experimental value of the critical flashover voltage, $\bar{y}_{\text{real}}$ the mean experimental value of the respective data set (training, evaluation or test), $\hat{y}$ the estimated value, $\bar{y}_{\text{est}}$ the mean estimated value of the data set and n the population of the respective data set. This index is a practical measure, which describes the approach of the estimated value to the actual one independently from the units and the quantity of the critical flashover voltage.

2.4 Selection of the best R^2 index from all the training algorithms

From all the training algorithms with the respective optimised parameters, the R^2 index that performs the highest is chosen as the best. This algorithm is used for the next steps.

2.5 Calculation of the confidence intervals

The confidence intervals for the evaluation set are calculated using the re-sampling method for a specific tail probability p. These intervals are extended in order to describe the values of the test set estimating the respective confidence interval with probability $(1 - 2p)$.

2.6 Final estimation for the test set

The critical flashover voltage (in kV) for the polluted insulators of the test set is finally estimated by using the training algorithm of the Section 2.4 with the respective ANN parameters. The prospective confidence intervals are the respective ones of the Section 2.5. It is also possible to calculate the confidence intervals for the test set for a comparative purpose.

In the following section, the ANN training algorithms are analysed.

3 Mathematical modelling of ANN

ANNs constitute a useful tool in the field of establishing relationships between quantities, that otherwise would have been difficult to model. A typical ANN is comprised of three layers, the input, the hidden and the output layer. The number of nodes of the input layer and the number of neurons at the output layer are equal to the number of input and output variables, respectively.

During the execution of each training algorithm, the ANN is presented with the patterns of the training set. Depending on the error between the estimated and the actual value, the weights of the ANN are adjusted until one of the stopping criteria, referred below, is fulfilled.

The evaluation criterion being used to evaluate the accuracy of the prediction is the average error for all the N patterns

$$G_{\text{av}} = \frac{1}{N}\sum_{n=1}^{N} G(n) \quad (8)$$

where $G(n) = 1/2\sum_{j\in C} e_j^2(n)$ is the sum of the square errors for all output neurons for the nth pattern.

Two different methods for presenting the patterns of the training set were used. In the first one, each input vector is randomly presented (stochastic training) minimising the error function per vector: $G(n) = 1/2\sum_{j\in C} e_j^2(n)$. In the second one, all input vectors are presented in series during the forward process (batch mode) and then, the weights are corrected minimising the average error function given by (8). In the case of the BP algorithm, with the steepest descent the correction of the weights' is calculated after the end of the respective epoch ep

$$\Delta\vec{w}(\text{ep}) = -\eta \cdot \nabla G(\vec{w}(\text{ep})) \quad (9)$$

where η is the learning rate.

If a momentum term a is added, then the equation for the correction of the weights' is

$$\Delta\vec{w}(\text{ep}) = -\eta \cdot \nabla G(\vec{w}(\text{ep})) + \alpha \cdot \Delta\vec{w}(\text{ep}-1) \quad (10)$$

3.1 BP algorithm variations

Apart from the original BP algorithm, several variations have been applied and have been compared, such as:

- Adaptive BP algorithm [9]. To achieve faster convergence, the learning rate and the momentum term are adaptively changed as

$$\eta(\text{ep}) = \begin{cases} \eta(\text{ep}-1), & G_{\text{av}}(\text{ep}) > G_{\text{av}}(\text{ep}-1) \\ \eta(\text{ep}-1)\cdot\exp\left(-1/T_\eta\right), & G_{\text{av}}(\text{ep}) \le G_{\text{av}}(\text{ep}-1) \end{cases} \quad (11)$$

$$a(\text{ep}) = \begin{cases} a(\text{ep}-1), & G_{\text{av}}(\text{ep}) > G_{\text{av}}(\text{ep}-1) \\ a(\text{ep}-1)\cdot\exp\left(-1/T_a\right), & G_{\text{av}}(\text{ep}) \le G_{\text{av}}(\text{ep}-1) \end{cases} \quad (12)$$

where T_η, $\eta_0 = \eta(0)$, T_a and $a_0 = a(0)$ are the time parameters and the initial values of the learning rate and the momentum term, respectively.

- Resilient algorithm [29]. The correction of the weights' is given by the formula

$$\Delta w_{ij}(\text{ep}) = \begin{cases} \delta_1 \cdot \Delta w_{ij}(\text{ep}-1), & \frac{\partial G_{\text{av}}}{\partial w_{ij}}(\text{ep}) \cdot \frac{\partial G_{\text{av}}}{\partial w_{ij}}(\text{ep}-1) > 0 \\ \Delta w_{ij}(\text{ep}-1), & \frac{\partial G_{\text{av}}}{\partial w_{ij}}(\text{ep}) \cdot \frac{\partial G_{\text{av}}}{\partial w_{ij}}(\text{ep}-1) = 0 \\ \frac{1}{\Delta_2} \cdot \Delta w_{ij}(\text{ep}-1), & \frac{\partial G_{\text{av}}}{\partial w_{ij}}(\text{ep}) \cdot \frac{\partial G_{\text{av}}}{\partial w_{ij}}(\text{ep}-1) < 0 \end{cases} \tag{13}$$

where δ_1, δ_2 are the increasing and the decreasing factor of change in the value of the weights between two successive epochs. In this method, the direction for correcting the weight is determined only by the sign of the derivative of the error function, thus providing an alternative solution for the saturation problem.

- Conjugate gradient algorithm [24]. The basic steps of this method are the following:

a. The first search direction $\vec{p}_0$ is selected to be the negative of the gradient

$$\vec{p}_0 = -\nabla G(\vec{w})|_{\vec{w}=\vec{w}_0} \tag{14}$$

b. The error function is minimised along the search direction

$$\Delta\vec{w}(\text{ep}) = -a_k \cdot \vec{p}_k \tag{15}$$

where the parameter a_k is computed by arithmetic methods, such as the golden section and bisection.

c. The next search direction is selected according to

$$\vec{p}_{k+1} = -\nabla G(\vec{w})|_{\vec{w}=\vec{w}_{k+1}} + \beta_{k+1} \cdot \vec{p}_k \tag{16}$$

where the parameter β_{k+1} is determined either by the Fletcher–Reeves equation (17) [25] or by the Polak–Ribiere equation (18) [26]

$$\beta_{k+1} = \frac{\nabla G(\vec{w})|^{\text{T}}_{\vec{w}=\vec{w}_{k+1}} \cdot \nabla G(\vec{w})|_{\vec{w}=\vec{w}_{k+1}}}{\nabla G(\vec{w})|^{\text{T}}_{\vec{w}=\vec{w}_k} \cdot \nabla G(\vec{w})|_{\vec{w}=\vec{w}_k}} \tag{17}$$

$$\beta_{k+1} = \frac{\Delta(\nabla G(\vec{w})|^{\text{T}}_{\vec{w}=\vec{w}_k}) \cdot \nabla G(\vec{w})|_{\vec{w}=\vec{w}_{k+1}}}{\nabla G(\vec{w})|^{\text{T}}_{\vec{w}=\vec{w}_k} \cdot \nabla G(\vec{w})|_{\vec{w}=\vec{w}_k}} \tag{18}$$

The second and third steps are repeated until the algorithm has converged. To achieve faster convergence, the algorithm should be restarted when the following criterion is fulfilled, as proposed by Powell [26]

$$|\nabla G(\vec{w})|^{\text{T}}_{\vec{w}=\vec{w}_k} \cdot \nabla G(\vec{w})|_{\vec{w}=\vec{w}_{k+1}}| \geq \lim_{\text{orthogonality}} \cdot \|\nabla G(\vec{w})|_{\vec{w}=\vec{w}_{k+1}}\|^2 \tag{19}$$

where the orthogonality limit $\lim_{\text{orthogonality}}$ can range from 0.1 to 0.9 – preferably 0.2.

The main disadvantage of this method is the calculation complexity per learning iteration.

- Scaled conjugate gradient algorithm [28]. It avoids the weakness of the last one by using the Levenberg–Marquardt approach. The steps of the algorithm are the following:

The first direction search is initialised as in (14) as well as the vector of the weights and biases $\vec{w}_0$ and the rest of the parameters (σ, λ_0, $\bar{\lambda}_0$, and flag) as

$$0 < \sigma \leq 10^4 \quad 0 < \lambda_0 \leq 10^{-6} \quad \bar{\lambda}_0 = 0 \quad \text{flag} = 1 \tag{20}$$

If flag = 1, then

$$\sigma_k = \sigma/\|\vec{p}_k\|$$
$$\vec{S}_k = (\nabla G(\vec{w})|_{\vec{w}=\vec{w}_k+\sigma_k\cdot\vec{p}_k} - \nabla G(\vec{w})|_{\vec{w}=\vec{w}_k})/\sigma_k \quad \delta_k = P_k^T \cdot \vec{S}_k \tag{21}$$

Parameter δ_k is given as

$$\delta_k = \delta_k + (\lambda_k - \bar{\lambda}_k) \cdot \|\vec{p}_k\|^2 \tag{22}$$

If $\delta_k \leq 0$, then the Hessian matrix is made positive

$$\bar{\lambda}_k = 2(\lambda_k - \delta_k/\|\vec{p}_k\|^2) \quad \delta_k = -\delta_k + \lambda_k \cdot \|\vec{p}_k\|^2$$
$$\lambda_k = \bar{\lambda}_k \tag{23}$$

The step size is calculated

$$\mu_k = -\vec{p}_k^{\,T} \cdot \nabla G(\vec{w})|_{\vec{w}=\vec{w}_k} \quad a_k = \mu_k/\delta_k \tag{24}$$

The comparison parameter is calculated

$$\Delta_k = 2 \cdot \delta_k \cdot (G(\vec{w})|_{\vec{w}=\vec{w}_k} - G(\vec{w})_{\vec{w}=\vec{w}_k+a_k\cdot\vec{p}_k})/\mu_k^2 \tag{25}$$

If $\Delta_k \geq 0$, then a successful reduction in error can be made as

$$\Delta\vec{w}_k = a_k \cdot \vec{p}_k \quad \vec{r}_{k+1} = -\nabla G(\vec{w})|_{\vec{w}=\vec{w}_{k+1}} \quad \bar{\lambda}_k = 0$$
$$\text{flag} = 1 \tag{26}$$

If k mod $N_{\text{w}} = 0$ (where N_{w} is the number of weights and

doi: 10.1049/iet-smt:20080009

biases), then the algorithm will be restarted

$$\vec{p}_{k+1} = -\nabla G(\vec{w})|_{\vec{w}=\vec{w}_{k+1}} \tag{27}$$

else

$$\beta_{k+1} = \left(\begin{array}{c} \left\| \nabla G(\vec{w})|_{\vec{w}=\vec{w}_{k+1}} \right\|^2 - \nabla G(\vec{w})|_{\vec{w}=\vec{w}_k}^{\mathrm{T}} \\ \cdot \nabla G(\vec{w})|_{\vec{w}=\vec{w}_k} \end{array} \right) \Big/ \mu_k \tag{28}$$

$$\vec{p}_{k+1} = \nabla G(\vec{w})|_{\vec{w}=\vec{w}_{k+1}} + \beta_{k+1} \cdot \vec{p}_k \tag{29}$$

If $\Delta_k \geq 0.75$, then $\lambda_k = 0.25 \cdot \lambda_k$, else $\bar{\lambda}_k = \lambda_k$, flag $= 0$.

$$\text{If } \Delta_k < 0.25, \text{ then } \lambda_k = \lambda_k + \delta_k(1 - \Delta_k)/\|\vec{p}_k\|^2 \tag{30}$$

If $\nabla G(\vec{w})|_{\vec{w}=\vec{w}_{k+1}} \neq \vec{0}$, then $k = k+1$ and step (2) is repeated, else the training process has been completed.

The main disadvantage of this method is its complexity ($O(6N_w^2)$ per iteration), whereas the complexity for the basic BP algorithm is $O(3N_w^2)$. If $\lambda_k = 0$, then the scaled conjugate gradient algorithm is identical to the conjugate gradient algorithm. The advantage of this method is that the error decreases monotonically, as an increase is not allowed. In case the error remains unchanged, the Hessian matrix is defined positively and λ_k increases.

- Newton algorithm [30]. The basic step of this method consists of inversing the Hessian matrix $\nabla^2 G(\vec{w})$ in order to estimate the weights and the biases

$$\Delta\vec{w}_k = -\nabla^2 G(\vec{w})|_{\vec{w}=\vec{w}_k}^{-1} \cdot \nabla G(\vec{w})|_{\vec{w}=\vec{w}_k} \tag{31}$$

Although this method is, usually, the fastest one, the inversion of the Hessian matrix according to the following formulas is complicated.

Hessian matrix:

$$\nabla^2 G(\vec{w}) = J(\vec{w})^{\mathrm{T}} \cdot J(\vec{w}) + \sum_{j \in C} e_j(\vec{w}) \cdot \nabla^2 e_j(\vec{w}) \tag{32}$$

Jacobian matrix:

$$J(\vec{w}) = \begin{bmatrix} \frac{\partial e_1}{\partial w_1} & \frac{\partial e_1}{\partial w_2} & \cdots & \frac{\partial e_1}{\partial w_{N_w}} \\ \frac{\partial e_2}{\partial w_1} & \frac{\partial e_2}{\partial w_2} & \cdots & \frac{\partial e_2}{\partial w_{N_w}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial e_{p_C}}{\partial w_1} & \frac{\partial e_{p_C}}{\partial w_2} & \cdots & \frac{\partial e_{p_C}}{\partial w_{N_w}} \end{bmatrix}_{p_C \times N_w} \tag{33}$$

One of the basic variations for the Newton method is the quasi-Newton method, where the second term of (32) is omitted. Alternatively, in the one-step secant algorithm, only the diagonal elements of the matrix are stored, thus making the inversion of the matrix an unnecessary task. The algorithm requires a greater number of iterations in order to converge. The computational complexity per iteration, however, is significantly compressed.

The Levenberg–Marquardt [31, 32] method suggests the following expression for the estimation of the value of the weights

$$\Delta\vec{w}_k = -\left(J^{\mathrm{T}} \cdot J + \lambda \cdot \mathrm{diag}[J^{\mathrm{T}} \cdot J]\right)^{-1} \cdot \nabla G(\vec{w})|_{\vec{w}=\vec{w}_k} \Rightarrow$$

$$\Delta\vec{w}_k = -\left(J^{\mathrm{T}} \cdot J + \lambda \cdot \mathrm{diag}[J^{\mathrm{T}} \cdot J]\right)^{-1} \cdot J^{\mathrm{T}} \cdot \vec{e}(\vec{w}_k) \tag{34}$$

Factor λ is given by the formula

$$\lambda(k+1) = \begin{cases} \lambda(k) \cdot \beta, & G_{\mathrm{av}}(k) > G_{\mathrm{av}}(k-1) \\ \lambda(k), & G_{\mathrm{av}}(k) = G_{\mathrm{av}}(k-1) \\ \lambda(k)/\beta, & G_{\mathrm{av}}(k) < G_{\mathrm{av}}(k-1) \end{cases} \tag{35}$$

where parameter β takes significant values, such as 10. This method gives satisfactory results in most problems, especially when the population of the weights and biases is less than a few hundreds.

3.2 ANN structure

Each ANN can be comprised by more than one hidden layer. According to Kolmogorov's theorem [33], an ANN can solve a problem using a single hidden layer, if the last one has the proper number of neurons. Under these circumstances, one hidden layer is used, however, the number of neurons has to be properly selected.

The following points need to be noted:

- Stopping criteria. In this study, three stopping criteria are used, namely the criterion of weight stabilisation criterion (36), the criterion of error function minimisation (37) and the criterion of the maximum number of epochs criterion (38). Analytically, the three criteria are, respectively, described by the following expressions

$$\left| w_{kv}^{(l)}(\mathrm{ep}) - w_{kv}^{(l)}(\mathrm{ep}-1) \right| < \mathrm{limit}_1, \ \forall k, v, l \tag{36}$$

$$\left| \mathrm{RMSE}(\mathrm{ep}) - \mathrm{RMSE}(\mathrm{ep}-1) \right| < \mathrm{limit}_2 \tag{37}$$

$$\mathrm{ep} \geq \mathrm{max\_epochs} \tag{38}$$

where $w_{kv}^{(l)}$ is the weight between k- the neuron of the $\ell-$ layer and v- the neuron of the $(\ell-1)$- layer, $\mathrm{RMSE} = \sqrt{1/(m_2 \cdot q_{\mathrm{out}}) \sum_{m=1}^{m_2} \sum_{k=1}^{q_{\mathrm{out}}} e_k^2(m)}$ is the root mean square error of the evaluation set with m_2 members and q_{out} neurons of the output layer (in this case $q_{\mathrm{out}} = 1$); max_epochs is the maximum number of the epochs. It is mentioned that for each algorithm two different approaches

are realised regarding the convergence. In case (a) all the three criteria were used, whereas in case (b) only the first and the third criterian were used.

• Activation function: In this study, three activation functions, also known as transfer functions, can be used

$$\text{Logistic:} \quad \phi(x) = 1/(1 + e^{-ax}) \tag{39}$$

$$\text{Hyperbolic tangent:} \quad \phi(x) = \tanh(ax + b) \tag{40}$$

$$\text{Linear:} \quad \phi(x) = ax + b \tag{41}$$

By making every possible combination for the activation functions of the hidden and the output layer and by changing the values of the parameters a and b, the most suitable combination for each method is selected.

3.3 Confidence intervals

The calculation of the confidence interval for the majority of the methods used for estimating unknown parameters is direct. For the ANNs, however, this calculation must be done according to one of the proposed methods, such as multilinear regression adapted to ANNs, error output and re-sampling. In this case, the re-sampling technique is applied, because the first one allows only the use of the linear activation function for the output layer and the second one doubles the number of the original outputs of the ANN. In the re-sampling method [17], the errors of the evaluation set are sorted in an ascending order considering the signs, and the cumulative sample distribution function of the prediction errors can be estimated as the following

$$S_{m_1}(z) = \begin{cases} 0, & z < z_1 \\ r/m_1, & z_r \le z < z_{r+1} \\ 1, & z_{m_1} \le z \end{cases} \tag{42}$$

When m_1 is large enough, $S_{m1}(z)$ is a good approximation of the true cumulative probability distribution $F(z)$. The confidence interval is estimated by keeping the intermediate z_r and discarding the extreme values, according to the desired confidence degree. The intervals are computed in order to be symmetrical in probability (not necessarily symmetric in z). The number of cases to discard in each tail of the estimation error distribution is $n \cdot p$, where p is the probability in each tail. If $n \cdot p$ is a fractional number, the number of cases to discard in each tail is $\lfloor n \cdot p \rfloor$ for the reasons of safety. In the case of the cumulative probability distribution $F(Z_p)$ to be equal to p, there is a probability p that an error is less than or equal to Z_p, which indicates that Z_p is the lower confidence limit. Consequently, Z_{1-p} is the upper limit and there is a $(1 - 2p)$ confidence interval for future errors.

4 Critical flashover voltage estimation using proposed methodology of the ANN

The proposed methodology is applied, as it has been presented in Section 2. There are several parameters to be selected, depending on the variation of the BP algorithm that is being used each time in order to train the ANN. The parameters that are common in all methods are: the number of neurons N_n, the type and the parameters (a, b) of the activation functions and the maximum number of epochs (max_epochs). For methods 1–6 (see Table 1), the additional parameters are: the time parameter and the initial value of the learning rate T_η, η_0 and the time parameter and the initial value of the momentum term T_a, a_0. The additional parameters are s, T_{bn} and T_{trix} for methods 7–10; σ and λ_0 for method 11; the increasing δ_1 and the decreasing δ_2 factor of change in the value of the weights between two successive epochs for method 12; and $\lambda(0)$ and β for method 14, respectively.

For each one possible combination of the input variables, the ANN parameters need to be specified. To reduce the combinations that need to be examined, two steps are taken. In the first step, the basic algorithm is executed separately for each parameter's range of values and the program registers the regions where satisfactory results for the current parameter are achieved. Specifically, the optimisation procedure for selecting the ANN's parameters is the following: first, the optimal number of neurons is determined by giving fixed values to the rest of the parameters and by varying the number of neurons from 2 to 25 with step 1. The optimal N_n is selected as the one that provides us with the smallest G_{av}. Next, the type of the activation functions that gives the best results is selected, while keeping the value for N_n as determined by the previous step. Then, for each one of the remaining parameters, the optimal value is similarly selected, while assigning to the rest of the parameters the optimal values that emerged from a previous step, if any. In Table 2, the respective intervals of values of all the parameters for every ANN training algorithm are presented.

In the second step, the main process is repeated for the reduced number of combinations, in which all parameters can take any value of their respective region, as determined in the first step. When this procedure is completed, the combination that presents the minimum error in the estimation of the evaluation set is selected. This combination is used for the estimation of the critical flashover voltage.

Table 3 summarises the optimal results for each training ANN algorithm with the respective values of the parameters provided by the optimisation process.

The optimisation process of the algorithm 1a (stochastic training with decreasing exponential functions for the

doi: 10.1049/iet-smt:20080009

Table 2 Values interval during the optimisation process of each parameter of every ANN training algorithm

No.	Intervals of values of each parameter of the respective ANN training algorithm (according to Table 1)
1	$\alpha_0 = 0.1,0.2, \ldots,0.9$, $T_\alpha = 200,400, \ldots,3000$, $\eta_0 = 0.1,0.2, \ldots,0.9$, $T_\eta = 200,400, \ldots,3000$
2	$\alpha_0 = 0.1,0.2, \ldots,0.9$, $T_\alpha = 200,400, \ldots,3000$, $\eta_0 = 0.1,0.2, \ldots,0.9$, $T_\eta = 200,400, \ldots,3000$
3	$\eta_0 = 0.01,0.02, \ldots,0.5,0.6, \ldots,2$
4	$\eta_0 = 0.1,0.2, \ldots,3$
5	$\alpha_0 = 0.1,0.2, \ldots,0.9$, $T_\alpha = 200,400, \ldots,3000$, $\eta_0 = 0.1,0.2, \ldots,0.9$, $T_\eta = 200,400, \ldots,3000$
6	$\alpha_0 = 0.1,0.2, \ldots,0.9$, $T_\alpha = 200,400, \ldots,3000$, $\eta_0 = 0.1,0.2, \ldots,0.9$, $T_\eta = 200,400, \ldots,3000$
7	$s = 0.04,0.1,0.2$, $T_{bv} = 20, 40$, $T_{\text{trix}} = 50, 100$, $e_{\text{trix}} = 10^{-6}, 10^{-5}$
8	$s = 0.04,0.1,0.2$, $T_{bv} = 20, 40$, $T_{\text{trix}} = 50, 100$, $e_{\text{trix}} = 10^{-6}, 10^{-5}$, $\text{lim}_{\text{orthogonality}} = 0.1,0.5,0.9$
9	$s = 0.04,0.2$, $T_{bv} = 20, 40$, $T_{\text{trix}} = 50, 100$, $e_{\text{trix}} = 10^{-6}, 10^{-5}$
10	$s = 0.04,0.2$, $T_{bv} = 20, 40$, $T_{\text{trix}} = 50, 100$, $e_{\text{trix}} = 10^{-6}, 10^{-5}$, $\text{lim}_{\text{orthogonality}} = 0.1,0.5,0.9$
11	$\sigma = 10^{-3}, 10^{-4}, 10^{-5}$, $\lambda_0 = 10^{-6}, 10^{-7}, 5{\cdot}10^{-8}$
12	$\delta_1 = 0.1,0.2$, $\delta_2 = 1,2$
13	–
14	$\lambda(0) = 0.1,0.2, \ldots, 1,2, \ldots, 5$, $\beta = 2,3, \ldots, 9,10,20, \ldots, 50$
Common	$N_n = \{2,3, \ldots,25\}$, activation function for hidden and output layers = linear, hyperbolic tangent, logistic, $a = 0.1,0.2, \ldots,1.5$, $b = -0.5, -0.4, \ldots,0.5$

learning rate and the momentum term with the use of all the three stopping criteria) is analytically presented below.

The optimisation process is comprised of consecutive steps in order to determine the values of the parameters of the ANN. In the first step, the number of neurons varies from 2 to 25 whereas the remaining parameters are assigned with fixed values. By the common graphs of the G_{av} against the number of neurons for the three sets (training, evaluation, test), as shown in Fig. 3, we draw the conclusion about the optimal selection for the parameter in question (number of neurons) (Table 3). The best results are achieved when the hidden layer is comprised of three neurons. It is observed that for $N_n > 2$, an oscillation in the values of G_{av} occurs, which supports the previous conclusion.

The next step is the determination of the optimal value for the time parameter and the initial value of the momentum term, while keeping the value for N_n as determined by the previous step and assigning fixed values to the rest of the parameters. Fig. 4 illustrates the variation of the G_{av} of the evaluation set for several values of α_0 and T_α. It is then self-evident that the preferable combination for α_0 and T_α is 0.9 and 1400, respectively.

The optimal values for η_0 and T_η are similarly selected, using as a criterion the minimum G_{av}. For $\eta_0 > 0.4$ and $T_\eta > 1000$, the G_{av} is satisfactorily low. The best results, however, are achieved for $\eta_0 = 0.9$ and $T_\eta = 1400$. Lastly, the proper combination for the activation functions used in the hidden and output layers is selected after examining all the nine possible alternatives (Table 4). It is obvious that the optimal selection is the hyperbolic tangent for the two layers. Accordingly, the values of the parameters of the function are selected: $a_1 = 0.9$, $a_2 = 0.4$, $b_1 = b_2 = 0$.

Finally, the correlation between the actual and the estimated values (R^2) for the evaluation set is 0.9955. The respective correlation between the actual and the estimated values (R^2) for the test set is 0.9753, whereas the respective mean absolute percentage error (MAPE) is 5.12%, where MAPE for a data set with m members is given by

$$\text{MAPE} = \frac{\sum_{i=1}^{m} |y_i - \hat{y}_i|/y_i}{m} \cdot 100\% \tag{43}$$

Fig. 5 illustrates the success of the estimation. Moreover, Fig. 5 depicts the confidence interval of the evaluation and the test sets. The effectiveness of the estimation provided by the developed ANN is clearly indicated by the fact that the interval of the test set is narrower than that of the evaluation set. In real-life applications, the confidence interval of the test set is not known. The only information is the confidence interval of the evaluation set and the estimated value of the critical flashover voltage. The fact that the interval of the test set is narrower than the interval of the evaluation set ensures that the region, to which the estimated value belongs, is accurate (meaning that the real value of the critical flashover

Table 3 Results of the ANN algorithms

	No. of ANN's training algorithm	G_{av} training set ($\times 10^{-4}$)	G_{av} evaluation set ($\times 10^{-4}$)	G_{av} test set ($\times 10^{-4}$)	Correlation index R^2 for evaluation set	N_n	Activation functions	Remaining parameters
Stochastic training	1a	3.0804	1.3118	0.5985	0.9955	3	$f_1 = \tanh(0.9x)$ $f_2 = \tanh(0.4x)$	$\alpha_0 = 0.9$, $T_\alpha = 1400$, $\eta_0 = 0.9$, $T_\eta = 1400$, max_epochs = 7000
	1b	4.4282	3.0511	2.0614	0.9819	3	$f_1 = \tanh(x)$ $f_2 = \tanh(x)$	$\alpha_0 = 0.8$, $T_\alpha = 1600$, $\eta_0 = 0.2$, $T_\eta = 400$, max_epochs = 7000
	2a	3.1550	1.9373	1.2398	0.9922	2	$f_1 = \tanh(0.9x)$ $f_2 = \tanh(0.5x)$	$\alpha_0 = \eta_0 = 0.9$, $T_\alpha = 1200$, $T_\eta = 800$, max_epochs = 7000
	2b	2.6312	2.7941	0.7893	0.9930	2	$f_1 = \tanh(x)$ $f_2 = \tanh(x)$	$\alpha_0 = 0.4$, $T_\alpha = 2800$, $\eta_0 = 0.1$, $T_\eta = 2600$, max_epochs = 7000
	3a	12.356	1.9286	4.3687	0.9693	3	$f_1 = \tanh(0.9x)$ $f_2 = \tanh(x)$	$\alpha_0 = 0.4$, $T_\alpha = 1000$, $\eta_0 = 0.32$, $T_\eta = 1000$, max_epochs = 7000
	3b	2.6369	2.5386	0.8043	0.9926	3	$f_1 = \tanh(x)$ $f_2 = 1/(1 + \exp(-0.6x))$	$\alpha_0 = 0.4$, $T_\alpha = 1000$, $\eta_0 = 0.13$, $T_\eta = 1000$, max_epochs = 7000
Batch mode training	4a	17.326	4.3633	3.7135	0.9792	18	$f_1 = \tanh(0.8x)$ $f_2 = 0.1x$	$\alpha_0 = 0.3$, $T_\alpha = 2000$, $\eta_0 = 3$, $T_\eta = 2000$, max_epochs = 5000
	4b	5.4084	3.5243	3.2704	0.9754	21	$f_1 = \tanh(x)$ $f_2 = 0.5x$	$\alpha_0 = 0.3$, $T_\alpha = 2000$, $\eta_0 = 3$, $T_\eta = 2000$, max_epochs = 5000
	5a	189.24	151.28	114.66	0.9149	21	$f_1 = 1/(1 + \exp(-x))$ $f_2 = 0.25x$	$\alpha_0 = 0.4$, $T_\alpha = 400$, $\eta_0 = 0.4$, $T_\eta = 200$, max_epochs = 7000
	5b	13.875	3.2073	4.2328	0.9651	21	$f_1 = \tanh(x)$ $f_2 = 1/(1 + \exp(-x))$	$\alpha_0 = 0.9$, $T_\alpha = 4800$, $\eta_0 = 0.9$, $T_\eta = 5600$, max_epochs = 7000
	6a	15.205	3.7728	3.8267	0.9765	21	$f_1 = \tanh(0.9x)$ $f_2 = 0.2x$	$\alpha_0 = 0.9$, $T_\alpha = 3000$, $\eta_0 = 0.8$, $T_\eta = 2600$, max_epochs = 7000
	6b	8.0111	2.9104	4.0557	0.9655	21	$f_1 = \tanh(x)$ $f_2 = \tanh(0.9x)$	$\alpha_0 = 0.9$, $T_\alpha = 5200$, $\eta_0 = 0.8$, $T_\eta = 5600$, max_epochs = 7000
	7a	1.1338	1.1916	0.3671	0.9972	10	$f_1 = \tanh(x)$ $f_2 = \tanh(0.4x)$	$s = 0.2$, $T_{bv} = 20$, $T_{trix} = 50$, $e_{trix} = 10^{-6}$, max_epochs = 7000

Continued

IET Sci. Meas. Technol., 2009, Vol. 3, No. 1, pp. 90–104

doi: 10.1049/iet-smt:20080009

Table 3 *Continued*

	No. of ANN's training algorithm	G_{av} training set ($\times 10^{-4}$)	G_{av} evaluation set ($\times 10^{-4}$)	G_{av} test set ($\times 10^{-4}$)	Correlation index R^2 for evaluation set	N_n	Activation functions	Remaining parameters
	7b	0.5700	0.5434	0.4441	0.9959	15	$f_1 = \tanh(x)$ $f_2 = \tanh(0.8x)$	$s = 0.2$, $T_{bv} = 20$, $T_{trix} = 50$, $e_{trix} = 10^{-6}$, max_epochs = 7000
	8a	12.409	1.6546	3.5029	0.9704	9	$f_1 = \tanh(1.2x)$ $f_2 = \tanh(0.04x)$	$s = 0.2$, $T_{bv} = 20$, $T_{trix} = 50$, $e_{trix} = 10^{-6}$, max_epochs = 7000
	8b	10.323	1.8298	3.9689	0.9657	7	$f_1 = \tanh(x)$ $f_2 = \tanh(0.07x)$	$s = 0.2$, $T_{bv} = 20$, $T_{trix} = 50$, $e_{trix} = 10^{-6}$, max_epochs = 7000
	9a	12.231	1.9136	4.0413	0.9651	7	$f_1 = \tanh(0.9x)$ $f_2 = \tanh(0.1x)$	$s = 0.2$, $T_{bv} = 20$, $T_{trix} = 50$, $e_{trix} = 10^{-6}$, max_epochs = 7000
	9b	17.575	3.5162	2.9477	0.9797	7	$f_1 = \tanh(0.6x)$ $f_2 = \tanh(0.1x)$	$s = 0.2$, $T_{bv} = 20$, $T_{trix} = 50$, $e_{trix} = 10^{-6}$, max_epochs = 7000
	10a	11.935	1.8242	3.9699	0.9658	7	$f_1 = \tanh(1.2x)$ $f_2 = \tanh(0.1x)$	$s = 0.2$, $T_{bv} = 20$, $T_{trix} = 50$, $e_{trix} = 10^{-6}$, max_epochs = 7000
	10b	12.409	1.6546	3.5029	0.9704	9	$f_1 = \tanh(1.2x)$ $f_2 = \tanh(0.04x)$	$s = 0.2$, $T_{bv} = 20$, $T_{trix} = 50$, $e_{trix} = 10^{-6}$, max_epochs = 7000
	11a	0.6138	0.7480	0.3289	0.9972	3	$f_1 = \tanh(0.325x)$ $f_2 = \tanh(0.1x)$	$\sigma = 10^{-5}$, $\lambda_0 = 5{\cdot}10^{-8}$, max_epochs = 7000
	11b	0.7959	0.9673	0.3669	0.9969	3	$f_1 = \tanh(0.35x)$ $f_2 = \tanh(0.2x)$	$\sigma = 10^{-5}$, $\lambda_0 = 5{\cdot}10^{-8}$, max_epochs = 7000
	12a-b	#	#	#	#	#	#	incapability of convergence
	13a-b	#	#	#	#	#	#	incapability of convergence
	14a-b	#	#	#	#	#	#	incapability of convergence

voltage belongs to it). Moreover, it is obvious that the upper and lower limits for both the evaluation and the test set are not symmetrically distributed around the actual values. This is attributed to the method for estimating the confidence interval. The re-sampling method that is used provides us with a confidence interval that in symmetric in probability (not necessarily symmetric in the estimated critical flashover voltage z).

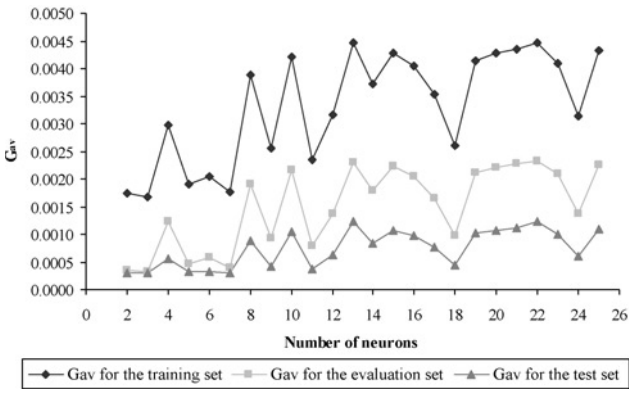

Figure 3 *G_{av} for the training, the evaluation and the test set against the number of neurons*

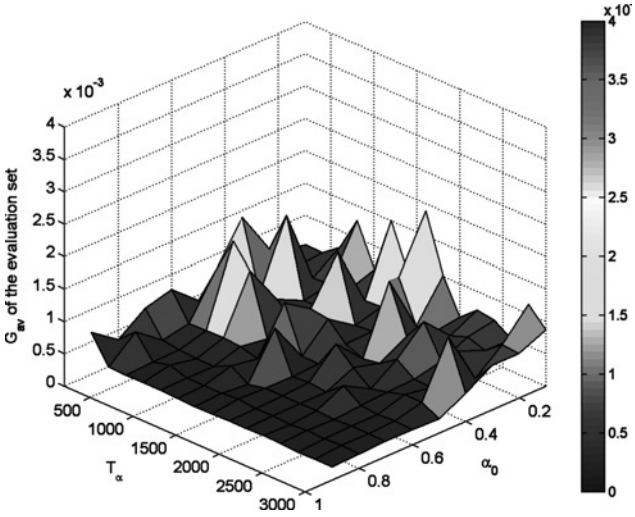

Figure 4 *G_{av} of the evaluation set for $\alpha_0 \in [0.1, 0.9]$ and $T_\alpha \in [200, 3000]$*

The same process is repeated for all training algorithms of Table 1. From the study of the results of Table 3, it is concluded that the best results are achieved when the scaled conjugate gradient algorithm is applied and all the three stopping criteria are used (algorithm 11a).

The optimisation process used in the case of algorithm 11a in order to achieve the minimum G_{av} is the following: first, the number of neurons is selected us the one that provides us with the smallest G_{av} for the three sets, which in this case is 3. Similarly, with method 1a, it is deducted that the best results are achieved when the activation functions are the hyperbolic tangent for both layers with $a_1 = 0.325$, $a_2 = 0.1$, $b_1 = b_2 = 0$. Additionally, different values for the parameters σ and λ_0 are tested ($\sigma = 10^{-5}$, 10^{-4}, 10^{-3}, $\lambda_0 = 10^{-7}$, 10^{-6}, 10^{-5}), until it is concluded that when $\sigma = 10^{-5}$ and $\lambda_0 = 5 \cdot 10^{-8}$, then the minimum G_{av} is achieved. It must be noted that, despite the final selection, there are several other combinations with similar results, meaning that the algorithm is not very sensitive to different values of the parameters σ and λ_0. The correlation between the actual and the estimated values R^2 for the evaluation set is 0.9972. The respective correlation between the actual and the estimated values R^2 for the test set is 0.9853, whereas the respective MAPE is 3.84%.

Fig. 6 depicts the success of the scaled conjugate gradient algorithm with three stopping criteria in estimating the critical flashover voltage. It is obvious that this method gives a satisfactory approach of the values for the critical flashover voltage. In the same picture, the confidence interval of the evaluation and the test set is shown. The effectiveness of the estimation provided by the developed ANN is clearly indicated by the fact that the interval of the test set is narrower than the interval of the evaluation set. This suggests that, if the only given information is the confidence interval of the evaluation set, then the estimation of the critical flashover voltage would have been satisfactorily accurate.

By using the mathematical model [16], the respective correlation between the actual and the estimated values (R^2) for the test set is 0.9801, whereas the respective MAPE is 4.59%, from which the superiority of the ANN optimisation methodology using the algorithm 11a is proved. The last one is confirmed in Figs. 7 and 8. Fig. 7 represents the actual and estimated values derived from the mathematical model [16] and from the proposed ANN methodology for 24 experimental vectors of the test set, whereas in Fig. 8, the respective absolute percentage errors from the mathematical model and from that ANN methodology are also presented.

Table 4 G_{av} ($\times 10^{-4}$) for the evaluation set for all the possible combinations between the activation functions for the hidden and the output layer ($a_1 = a_2 = 0.4$, $b_1 = b_2 = 0$)

		Activation function for the hidden layer		
		Logistic	Hyperbolic tangent	Linear
Activation function for the output layer	Logistic	2.4212	2.4839	4.1023
	Hyperbolic tangent	2.1859	1.7304	2.2577
	Linear	12.2654	22.4301	18.9700

IET Sci. Meas. Technol., 2009, Vol. 3, No. 1, pp. 90–104
doi: 10.1049/iet-smt:20080009

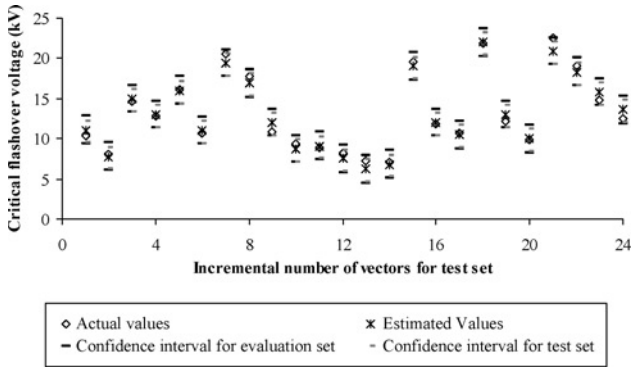

Figure 5 *Representation of the actual and the estimated values and of the confidence interval for the evaluation and test set with 5% probability in each tail for the stochastic training algorithm with learning rate and momentum term with three stopping criteria (algorithm 1a) for the test set*

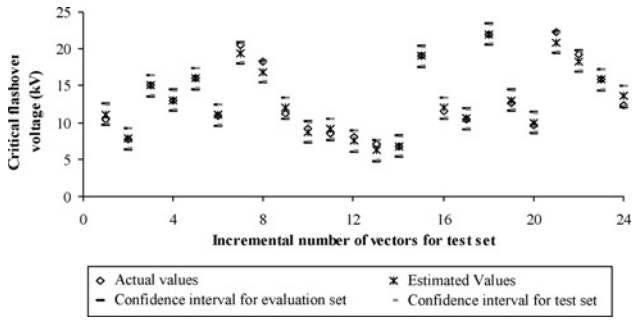

Figure 6 *Representation of the actual and the estimated values and of the confidence interval for the evaluation and test set with 5% probability in each tail for the scaled conjugate gradient algorithm with three stopping criteria (algorithm 11a) for the test set*

In Fig. 8, it is observed in most cases when the absolute percentage error values are big, the error of the ANN methodology is significantly better than the one derived from the mathematical model.

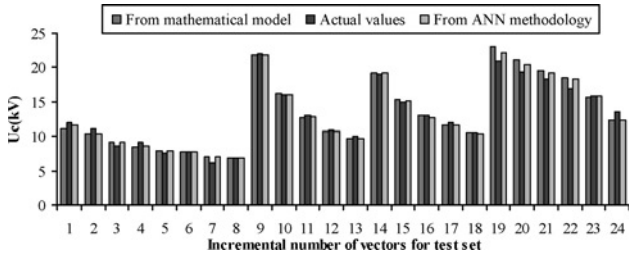

Figure 7 *Representation of the actual and estimated values from the mathematical model [16] and from the proposed ANN methodology (results from algorithm 11a) for the test set*

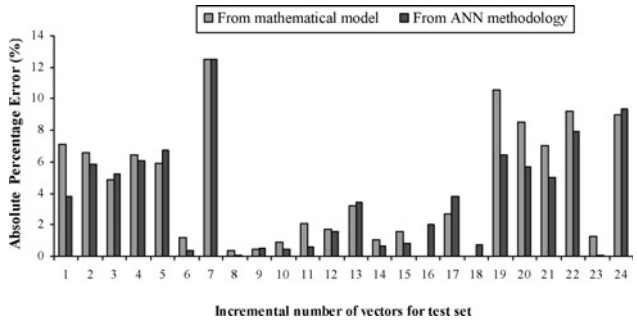

Figure 8 *Representation of the from the mathematical model [16] and from the proposed ANN methodology (results from algorithm 11a) for the test set*

5 Conclusions

A novel methodology for the estimation of the critical flashover voltage of polluted insulators was presented by an ANN. It performs an extensive search in order to select the optimum training algorithm and the respective parameters such as neurons, activation functions, learning rate for the weighting factors and momentum term. The generalisation ability of the proposed methodology is achieved by the use of three sets (training, evaluation and test). Moreover, the inclusion of a small number of experimental results in the training and evaluation sets and the successive estimation achieved on purely experimental results, of which the test set consists, suggests that practically only a small number of experimental results is required in order to train an ANN that provides us with more than satisfactorily accurate estimations. For the test set (24 experimental values), the estimated results are better than the ones calculated by the mathematical model [16], because the correlation (R^2) for the optimal selection of parameters for the scaled conjugate gradient algorithm using the three stopping criteria (the criterion of weight stabilisation the criterion of error function minimisation and the criterion of the maximum number of epochs' criterion) is 98.53%, in contrast to the correlation of the mathematical model (98.01%) [16]. In addition, the MAPE of the most effective ANN is 3.84% instead of 4.59% of the mathematical model [16]. Currently, the confidence interval of the critical flashover voltage for each insulator is calculated using the re-sampling method beyond the estimated value. This leads to a more accurate, generalised and objective estimation of the respective critical flashover voltage and makes the proposed methodology a powerful and useful tool.

6 References

[1] RIZK F.A.M.: 'Mathematical models for pollution flashover', Electra, No. 78. October 1981, pp. 71–103

[2] AYDOGMUS Z., CEBECI M.: 'A new flashover dynamic model of polluted HV insulators', *IEEE Trans. Dielectr. Electr. Insul.*, 2004, **11**, (4), pp. 577–584

[3] ENGELBRECHT C.S., HARTINGS R., TUNELL H., ENGSTRÖM B., JANSSEN H., HENNINGS R.: 'Pollution tests for coastal conditions on an 800 kV composite bushing', *IEEE Trans. Power Deliv.*, 2003, **18**, (3), pp. 953–959

[4] BOUDISSA R., HADDAD A., SAHLI Z., MEKHALDI A., BAERSCH R.: 'Performance of outdoor insulators under non-uniform pollution conditions'. 14th Int. Symp. High Voltage Engineering, China, August 2005, D-51

[5] RASOLONJANAHARY J.L., KRÄHENBÜHL L., NICOLAS A.: 'Computation of electric fields and potential on polluted insulaors using a boundary element method', *IEEE Trans. Magn.*, 1992, **28**, (2), pp. 1473–1476

[6] DE TOURREIL C.H., LAMBETH P.J.: 'Aging of composite insulators: simulation by electrical tests', *IEEE Trans. Power Deliv.*, 1990, **5**, (3), pp. 1558–1567

[7] CHENG Y., LI C.H., NIU C.H., ZHANG F.: 'Porcelain insulators detection by two dimensions electric field on high voltage transmission lines'. 15th Int. Symp. High Voltage Engineering, Slovenia, August 2007, T4-495

[8] KONTARGYRI V.T., GIALKETSI A.A., TSEKOURAS G.J., GONOS I.F., STATHOPULOS I.A.: 'Design of an artificial neural network for the estimation of the flashover voltage on insulators', *Electric Power Syst. Res.*, 2007, **77**, (12), pp. 1532–1540

[9] GHOSH P.S., CHAKRAVORTI S., CHATTERJEE N.: 'Estimation of time-to-flashover characteristics of contaminated electrolytic surfaces using a neural network', *IEEE Trans. Dielectr. Electr. Insul.*, 1995, **2**, (6), pp. 1064–1074

[10] CLINE P., LANNES W., RICHARDS G.: 'Use of pollution monitors with a neural network to predict insulator flashover', *Elect. Power Syst. Res.*, 1997, **42**, (1), pp. 27–33

[11] UGUR M., AUCKLAND D.W., VARLOW B.R., EMIN Z.: 'Neural networks to analyse surface tracking on solid insulators', *IEEE Trans. Dielectr. Electr. Insul.*, 1997, **4**, (6), pp. 763–766

[12] JAHROMI A.N., EL-HAG A.H., JAYARAM S.H., CHERNEY E.A., SANAYE-PASAND M., MOHSENI H.: 'A neural network based method for leakage current prediction of polymeric insulators', *IEEE Trans. Power Deliv.*, 2006, **21**, (1), pp. 506–507

[13] JAHROMI A.N., EL-HAG A.H., JAYARAM S.H., CHERNEY E.A., SANAYE-PASAND M., MOHSENI H.: 'Prediction of leakage current of composite insulators in salt fog test using neural network'. 2005 Annual Report Conf. Electrical Insulation and Dielectric Phenomena, 2005

[14] GHOSH S., KISHORE N.K.: 'Modeling PD inception voltage of epoxy resin post insulators using an adaptive neural network', *IEEE Trans. Dielectr. Electr. Insul.*, 1999, **6**, (1), pp. 131–134

[15] DIXIT P., GOPAL H.G.: 'ANN based three stage classification of Arc gradient of contaminated porcelain insulators'. 2004 Int. Conf. Solid Dielectrics, Toulouse, France, 5–9 July 2004

[16] TOPALIS F.V., GONOS I.F., STATHOPOULOS I.A.: 'Dielectric behaviour of polluted porcelain insulators', *IEE Proc. Gener. Trans. Distrib.*, 2001, **148**, (4), pp. 269–274

[17] SILVA A.P.A., MOULIN L.S.: 'Confidence intervals for neural network based short-term load forecasting', *IEEE Trans. Power Syst.*, 2000, **15**, (4), pp. 1191–1196

[18] IKONOMOU K., KATSIBOKIS G., KRAVARITIS A., STATHOPOULOS I.A.: 'Cool fog tests on artificially polluted suspension insulators'. 5th Int. Symp. High Voltage Engineering, Braunschweig, August 1987, vol. II, paper 52.13

[19] IEC 507: 'Artificial pollution tests on high-voltage insulators to be used on a.c. systems', 1991

[20] ZHICHENG G., RENYU Z.: 'Calculation of DC and AC flashover voltage of polluted insulators', *IEEE Trans. Electr. Insulat.*, 1990, **25**, (4), pp. 723–729

[21] SUNDARARAJAN R., SADHUREDDY N.R., GORUR R.S.: 'Computer-aided design of porcelain insulators under polluted conditions', *IEEE Trans. Dielectr. Electr. Insul.*, 1995, **2**, (1), pp. 121–127

[22] IEC 507: 'Artificial pollution tests on high-voltage insulators to be used on a.c. systems', 1991

[23] GONOS I.F., TOPALIS F.V., STATHOPOULOS I.A.: 'Genetic algorithm approach to the modelling of polluted insulators', *IEE Proc., Gener. Transm. Distrib.*, 2002, **149**, (3), pp. 373–376

[24] SAINI L.M., SONI M.K.: 'Artificial neural network-based peak load forecasting using conjugate gradient methods', *IEEE Trans. Power Syst.*, 2002, **17**, (3), pp. 907–912

[25] FLETCHER R., REEVES C.M.: 'Function minimization by conjugate gradients', *Comput. J.*, 1964, **7**, pp. 149–154

[26] POWELL M.J.: 'Restart procedures for the conjugate gradient method', *Math. Program.*, 1977, **12**, pp. 241–254

[27] POLAK E.: 'Computational methods in optimisation: a unified approach' (Academic Press, New York, 1971)

[28] MOLLER M.F.: 'A scaled conjugate gradient algorithm for fast supervised learning', *Neural Netw.*, 1993, **6**, pp. 525–533

[29] RIEDMILLER M., BRAUN H.: 'A direct adaptive method for faster backpropagation learning: the RPROP algorithm'.

doi: 10.1049/iet-smt:20080009

Proc. IEEE Int. Conf. Neural Networks, San Francisco, March 1993, vol. 1, pp. 586–591

[30] BATTITI R.: 'First and second order methods for learning: between steepest descent and Newton's method', *Neural Comput.*, 1992, **4**, (2), pp. 141–166

[31] LEVENBERG K.: 'A method for the solution of certain problems in least squares', *Q. Appl. Math.*, 1944, **2**, pp. 164–168

[32] MARQUARDT D.: 'An algorithm for least squares estimation of nonlinear parameters', *SIAM J. Appl. Math.*, 1963, **11**, pp. 431–441

[33] HAYKIN S.: 'Neural networks: a comprehensive foundation' (Prentice Hall, 1994)

7 Appendix

In this section, the theoretical and experimental data that were used in this work are presented.

Using the data given in Table 5 and the following values for the equivalent salt deposit density C (in mg/cm^2): {0.02, 0.03, 0.04, 0.05, 0.06, 0.13, 0.16, 0.23, 0.28, 0.34, 0.37, 0.49, 0.52, 0.55} and applying (1), the flashover voltage can be calculated. The experimental data is also given in Table 6.

Table 5 Values that were used in the mathematical model for the calculation of the flashover voltage

D_m, cm	26.8	26.8	25.4	25.4	29.2	27.9	32.1	28.0	25.4	20.0
H, cm	15.9	15.9	16.5	14.6	15.9	15.6	17.8	17.0	14.5	16.5
L, cm	33.0	40.6	43.2	31.8	47.0	36.8	54.6	37.0	30.5	40.0
F	0.79	0.86	0.90	0.72	0.92	0.76	0.96	0.80	0.74	1.29

Table 6 Experimental values

	D_m, cm	H, cm	L, cm	F	C, mg/cm^2	U_c (kV)
Test set	25.4	14.6	27.9	0.68	0.13	12.0
	25.4	14.6	27.9	0.68	0.16	11.1
	25.4	14.6	27.9	0.68	0.23	8.7
	25.4	14.6	27.9	0.68	0.28	9.1
	25.4	14.6	27.9	0.68	0.34	7.5
	25.4	14.6	27.9	0.68	0.37	7.8
	25.4	14.6	27.9	0.68	0.49	6.2
	25.4	14.6	27.9	0.68	0.52	6.8
	25.4	14.6	30.5	0.70	0.02	22.0
	25.4	14.6	30.5	0.70	0.05	16.0
	25.4	14.6	30.5	0.70	0.10	13.0
	25.4	14.6	30.5	0.70	0.16	11.0
	25.4	14.6	30.5	0.70	0.22	10.0
	25.4	14.6	43.2	0.92	0.05	19.0
	25.4	14.6	43.2	0.92	0.10	15.0
	25.4	14.6	43.2	0.92	0.16	13.0
	25.4	14.6	43.2	0.92	0.22	12.0
	25.4	14.6	43.2	0.92	0.30	10.5
	22.9	16.6	43.2	1.38	0.03	20.9
	22.9	16.6	43.2	1.38	0.04	19.4

Continued

Table 6 *Continued*

	D_m, cm	H, cm	L, cm	F	C, mg/cm^2	U_c (kV)
	22.9	16.6	43.2	1.38	0.05	18.3
	22.9	16.6	43.2	1.38	0.06	16.9
	22.9	16.6	43.2	1.38	0.10	15.8
	22.9	16.6	43.2	1.38	0.20	13.6
Training set	25.4	14.6	27.9	0.68	0.55	6.1
	25.4	14.6	30.5	0.70	0.30	8.5
	25.4	14.6	43.2	0.92	0.02	26.0
	22.9	16.6	43.2	1.38	0.02	23.5

IET Sci. Meas. Technol., 2009, Vol. 3, No. 1, pp. 90–104

doi: 10.1049/iet-smt:20080009