FAST CONTINGENCY RANKING USING ARTIFICIAL NEURAL NETWORKS

K L Lo  L J Peng  J F Macqueen  A O Ekwue  D T Y Cheng

University of Strathclyde  The National Grid Company plc.
United Kingdom

ABSTRACT

Contingency ranking attempts to predict the impact on a power system of various outages without actually performing a full ac load flow. Existing methods suffer from either masking effects in approximate approaches or slow execution in more accurate approaches. This paper proposes a fast contingency ranking approach which is based on pattern recognition technique using artificial neural networks. The power system operating state is defined by a set of variables used as a pattern. The corresponding performance indexes of various contingencies can be recognised by the properly trained artificial neural networks, backpropagation and counterpropagation networks. The ranking performance compared with one of full ac load flow shows the artificial neural networks can achieve high speeds of execution and have good pattern recognition ability.

Keyword: Contingency ranking; Contingency analysis; Pattern recognition; Artificial neural networks; Backpropagation networks and Counterpropagation networks.

INTRODUCTION

Contingency ranking is very commonly used for contingency analysis in modern energy management control centres. It attempts to predict the impact on the power system of various contingencies without actually performing a full ac power flow. Contingencies can be ranked according to a system-wide scalar Performance Index (PI) which is calculated by approximate simulations of post-contingency effects to quantify the severity of each contingency.

Since Ejebe and Wollenberg in 1979 first proposed the idea of contingency ranking,1,2 a variety of algorithms have been developed. To summarize, contingency ranking consists of two steps. The first step is to define a proper performance index for measuring the severity of contingencies on the power system. Some definitions are based on line MW overload, which is referred to as MW type PI; the other definitions are based on voltage limit violation or reactive power limit violation of generator unit, which is referred to as voltage type PI (or MVar type PI). The second step is how to calculate PI value by approximate computing methods. The principal approaches of computing PI values are basically grouped as followings: (a) DC load flow;2 (b) Local solution based on concentric relaxation;3 (c) Network performance index;4,5 (d) 1P-1Q iteration of load flow;6 (e) Linear distribution factor;7 (f) Bounding method;8,9 (g) Zero mismatch method;10 (h) Energy function method.11 These techniques may be successful in many specific implementations. However, Lo and et al did two comprehensive studies on MW type and Voltage type PI ranking techniques in 1988 and 1989 respectively,12,13 and concluded that ‘masking’ effects which are inherent in these techniques could lead to misclassification. Later on, contingency screening approaches14 are proposed for improving the accuracy of contingency ranking. A performance evaluation was also made to conclude that performance index based methods are efficient but vulnerable to misranking while screening methods are accurate but inefficient.15

In addition, the emergence of artificial intelligence (AI) as a pattern recognition technique motivated many researchers to investigate its applicability to power system.16 Many novel approaches, such as fuzzy theory17 and expert system (ES)18,19, also were developed to heuristically estimate the severity of MW and voltage contingencies. However, these approaches suffer from slow execution, knowledge base insufficiency, etc.

Existing contingency ranking methods suffer from either masking effects in approximate methods or slow execution in more accurate ranking methods. Although the advancement of computer technology reduces the contingency ranking time significantly, it is still a heavy burden in contingency analysis. Researchers continue to look for new contingency ranking approaches which provide higher computational efficiency without sacrificing accuracy. Since Sobajic and Pao began to use artificial neural networks (ANNs) as an alternative approach to power system studies,20 a number of literatures21-30 have proposed the applications of ANNs approaches to security monitoring, contingency selection, contingency evaluation and contingency screening. Most
of these approaches can indicate that the present operating state is situated in Normal state or Alert state, but they cannot demonstrate which contingencies will cause the violation of constraints. Some approaches used Hopfield neural network to select only the most severe contingency out of the contingency set\(^{[37-38]}\), but more detailed and accurate security assessments are required.

In order to improve this situation and meet new demands, this paper proposes a fast contingency ranking scheme based on pattern recognition using artificial neural networks. The power system state of pre-contingencies is considered as a pattern. The performance indexes are calculated according to some variable values of post-contingency power system state. The system state of pre-contingency and the corresponding performance indexes of post-contingency condition are composed as training patterns. The performance indexes of a set of contingencies under various power system operating conditions are recognised from the trained ANNs which stored some similar patterns. In this paper the first section introduces how the contingency ranking is considered as a pattern recognition problem which can be solved by artificial neural networks. The second section briefly describes two artificial neural networks, backpropagation and counterpropagation networks, which are used for contingency ranking. In the last section, the two kind of artificial neural networks are tested on 5 busbar 7 line power system. The results demonstrate that the proposed approach has a good performance for contingency ranking.

**PROBLEM FORMULATION**

Before the introduction, notation is given as follows:

\[ P_{Gi}, Q_{Gi} \] the active and reactive power generation at bus \( i \);
\[ P_{Di}, Q_{Di} \] the active and reactive power demand at bus \( i \);
\[ P_i, Q_i \] the active and reactive line flow from bus \( i \) to bus \( j \);
\[ P_{i}^{\text{Max}} \] the maximum limit of active power flow between bus \( i \) and bus \( j \);
\[ P_i \] the real power flow on line \( i \);
\[ P_{i}^{\text{M}} \] the MW capacity of limit on line \( i \);
\[ \Delta P_{i}^{\text{M}} \] the maximum allowable MW deviation on line \( i \);
\[ \theta_y \] the voltage angle difference between bus \( i \) and bus \( j \);
\[ \theta_{y}^{\text{Max}} \] the maximum phase angle difference limit between bus \( i \) and bus \( j \);
\[ V_i \] the voltage magnitude at bus \( i \);
\[ V_i^{\text{spec}} \] the specified voltage magnitude at bus \( i \);
\[ \Delta V_i^{\text{lim}} \] the voltage deviation limit, above which voltage deviations are unacceptable;
\[ V_{i}^{\text{Min}}, V_{i}^{\text{Max}} \] the lower and upper limit of voltage magnitude at bus \( i \);
\[ \Delta V_i^{\text{M}} \] the maximum allowable voltage deviation of lower limit at bus \( i \);
\[ \Delta V_i^{\text{M}} \] the maximum allowable voltage deviation of upper limit at bus \( i \);
\[ W_r \] the line active power weighting coefficient;
\[ W_v \] the voltage magnitude weighting coefficient;
\[ \alpha \] set of overload lines;
\[ \beta \] set of buses whose voltage magnitude is below a specific lower limit;
\[ \gamma \] set of buses whose voltage magnitude is exceeded a specific upper limit;
\[ NL \] the total number of transmission lines;
\[ N \] the total number of buses;
\[ n \] an integer.

The primary concept for measuring power system security was proposed by Liacco in 1967\(^{[33]}\). According to his tentative idea, three states, i.e. normal, emergency and restorative, were defined for describing the operating states. Later on, alert state was added by Cihlar and et al in 1969\(^{[34]}\). A in-extremis operating condition (system starts to split into smaller sections) was then extended as the fifth states in 1978\(^{[35]}\), but most researches were based on the definition of the former four states.

Power system operating state can be described by a set of variables (e.g. line flows, bus voltages and so on). Maintaining power system static security is accomplished by ensuring that all the system state variables are within operating constraints. The basic equality and inequality constraints are described as following:

\[
P_{Gi} = P_{i} + \sum_{j \neq i} P_{j} \tag{1}
\]
\[
Q_{Gi} = Q_{i} + \sum_{j \neq i} Q_{j} \tag{2}
\]
\[
P_{Gi} ^{\text{Min}} \leq P_{Gi} \leq P_{Gi} ^{\text{Max}} \tag{3}
\]
\[
Q_{Gi} ^{\text{Min}} \leq Q_{Gi} \leq Q_{Gi} ^{\text{Max}} \tag{4}
\]
\[
V_{i} ^{\text{Min}} \leq V_{i} \leq V_{i} ^{\text{Max}} \tag{5}
\]
\[
\theta_{y} = \theta_{i} - \theta_{j} \leq \theta_{y} ^{\text{Max}} \tag{7}
\]

If any constraint violation is detected, the present operating state may be classified as emergency or restorative. The basic goal for the security operation is to keep operation within these constraints. However, in practice, it is not sufficient just to maintain a system within these constraints. If no violation is detected in the present state, security of the power system also needs to be assessed by performing an on-line contingency analysis which includes the following three aspects:

i) Contingency definition: as many contingencies need to be considered as possible. In common use, the single component outage must be included in
contingency set. The members of contingency set are chosen depending on the security standard required;

ii) Contingency selection: the main task of contingency selection is to identify those critical contingencies which are likely to result in constraint violations. Currently, contingency ranking is a preferred method used in modern energy management control centres;

iii) Contingency evaluation: only those selected contingencies are further investigated by the conduct of load flow calculation. If the evaluation results in any constraint violation, the system is classified as insecure or Alert state. Otherwise, the system is classified as secure state or Normal state.

The contingency selection and ranking is one of the most time-consuming functions in on-line security assessment. The first step of contingency ranking is to define a proper performance index for measuring the relative security correctly. A contingency that causes a line MW overload may not necessarily cause a bus voltage problem, and vice versa[6]. Thus, two different performance indexes which give measures for line overloads and bus voltage violations, respectively, are defined for MW and voltage contingency ranking. These indexes are usually of the following form:

\[ PI_p = \sum_{i=1}^{N} W_i \left( \frac{P_i}{P_{i,\text{Max}}} \right)^2 \]  
\[ PI_v = \sum_{i=1}^{N} W_i \left( \frac{V_i - V_{i,\text{Min}}}{\Delta V_{i,\text{Min}}} \right)^2 \]  

This kind of performance indexes detect variation of monitored variables from the pre-contingency state to quantify the severity of contingencies. However, a contingency causes a large variation in line flows or voltage magnitude may not be an exact representative of the severity of the contingency[12-13]. It would also result in 'masking problem'. To avoid the undesirable effect, in this research, only those overloaded lines and buses with voltage magnitude violations are taken into account in the computation of MW performance index \( PI_p \) and voltage performance index \( PI_v \), respectively.

\[ PI_p = \sum_{i=1}^{N} W_i \left( \frac{P_i}{P_{i,\text{Max}}} \right)^2 \]  
\[ PI_v = \sum_{i=1}^{N} W_i \left( \frac{V_i - V_{i,\text{Min}}}{\Delta V_{i,\text{Min}}} \right)^2 \]  

According to the definition of the performance indexes, it has two contents. If the PI value is greater than zero, the corresponding contingency is identified as insecure; otherwise it is secure. The PI value also represents the severity of the contingencies. The greater the value, the more severe the contingency would be. Thus, it is very convenient to implement contingency ranking in accordance with the value of this kind of defined performance indexes.

A power system operating condition is uniquely described by a set of variables such as line power flows and bus voltage magnitudes which can be obtained from the state estimation. These variables (supposed \( n \) in number) can be arranged into an \( n \)-dimensional vector. The operating condition is characterised by these variables which can be then be viewed as a pattern in \( n \)-dimensional Euclidean space. In existing methods, the PI values of various contingencies for a given pattern are calculated by approximate computing methods. As mentioned previously, these methods suffer from masking effects. Using pattern recognition technology, the PI values can be calculated by an off-line full AC load flow program. Many patterns and corresponding performance indexes are composed to form a set of training patterns. After training, the PI of various contingencies under different operating conditions can be recognised from the fully trained ANNs which stored some similar patterns. According to the definition of the PI in equation (10) and (11), the PI value are not only able to select these critical contingencies, but also quantify the severity of each contingency. From the discussion, it is clear that contingency ranking can be viewed as a pattern recognition problem.

ARTIFICIAL NEURAL NETWORKS

The section is aimed to develop an efficient approach of calculating the PI values of the post-contingency under various load levels. In this research we proposed to use two different artificial neural networks algorithms for recognising their performance indexes of a set of specific contingencies under different load levels.

Artificial neural networks are the implementation of various algorithms inspired by research into the human brains. From the view of computing methods, artificial neural networks are computational models built around massively parallel interconnected neural units (referred to as processing elements, PE). An artificial neural network processes a set of inputs and corresponding output samples to learn the similarities. This kind of learning process allows the ANN to generalise a reasonable output for inputs not provided during the training process. The basic learning algorithms of the backpropagation and counterpropagation networks are briefly described as followings.

Backpropagation Network

The invention of the Backpropagation (BP) algorithm has played an important role in the resurgence of interest in artificial neural networks. Backpropagation is a systematic method for training multilayer artificial neural networks. The basic architecture of a simple feedforward neural network is shown in Figure 1.
Backpropagation is a supervising training method. The objective of training the network is to adjust the weights so that application of a set of inputs produces the desired set of output. The backpropagation algorithm is briefly presented in the following five steps.

**Step 1** Initialise weights and thresholds;
Set all weights and thresholds to small random values.

**Step 2** Present input and desired output;
Present input $X_p = x_1, x_2, x_3, \ldots, x_n$ and target output $T_p = t_1, t_2, \ldots, t_m$ where $n$ is the number of input nodes and $m$ is the number of output nodes. For supervising training, $X_p$ and $T_p$ represent a training pattern to be associated.

**Step 3** Calculate actual outputs;
Each layer calculates every node’s actual output:
$$y_{pi} = f \left( \sum_{i=1}^{k} w_{ij} x_i \right)$$
and passes them as inputs to the next layer. The final layer output values are $y_{pj}$ and $k$ in equation (12) is the number of nodes in the layer.

**Step 4** Adapt weights;
Start from the output layer, and work backwards.
$$w_{ij}(t+1) = w_{ij}(t) + \eta \delta_{pj} y_{pj}$$
$w_{ij}(t)$ represents the weight from node $i$ to node $j$ at time $t$, $\eta$ is a gain term, and $\delta_{pj}$ is an error term for pattern $p$ on node $j$.

For output units:
$$\delta_{pj} = \alpha(t) y_{pj} (1 - y_{pj}) (t_{pj} - y_{pj})$$
where $t_{pj}$ represents the target output for pattern $p$ on node $j$, $y_{pj}$ represents the actual output at that node; $\alpha(t)$ is a gain term which starts at value between 0.4 and 1.0, then decreases in time as training progresses.

For hidden units:
$$\delta_{pj} = \alpha(t) y_{pj} (1 - y_{pj}) \sum_{k} \delta_{pk} w_{jk}$$
where the sum is over the $k$ nodes in the layer above node $j$.

**Step 5** Repeat by going to Step 2 until all weights change within a certain tolerance.

### Counterpropagation Network

The counterpropagation network (CPN) developed by Hecht-Nielson is a combination of the Kohonen self-organising map, which performs unsupervised clustering, and a supervised Grossberg outstar layer which transforms inputs into the correct mapping function, $\Phi$. A simple feedforward version of the CPN for on-line recognition of performance indexes of various contingencies under different load levels is shown in Figure 2. A brief description of this network’s architecture, functional operation and learning capability is given below.

The architecture of CPN consists of three layers: an input layer containing $n$ neurons which simply multiplex the input signals $X_1, X_2, \ldots, X_n$; a middle Kohonen layer with $N$ neurons with output signals $Z_1, Z_2, \ldots, Z_n$; and a final Grossberg outstar layer with $m$ neurons with output signals $Y_1, Y_2, \ldots, Y_m$. Each neuron $i$ in the middle Kohonen layer receives the complete input pattern $X$ through a set of connections which have weights $W_i$. Similarly, each neuron $k$ on the Grossberg layer receives a Kohonen layer output through a set of connections which have weight vector $U_k$. The outputs of the Grossberg layer have primes on them because they represent approximations to the components $Y_1, Y_2, \ldots, Y_m$ of $Y = \Phi(X)$. During training, these desired values $Y$ are also supplied to some neurons of the input layer.
The operation of the network is usually thought of as consisting of two successive stages: training and normal operation. During training, the network is exposed to examples of the mapping function $F : R^n \rightarrow R^m$. It is assumed that input training vectors $X$ are drawn from $R^n$ in accordance with a fixed probability density function. After each $X$ is selected, $Y = F(X)$ is determined and both $X$ and $Y$ are input to the network. This kind of training is called supervised training.

Counterpropagation networks are trained in two-phase process. In the first phase, the Kohonen neuron's weights are adjusted to match the input. After this, the second training phase helps to adjust the Grossberg weights in order to fit the desired neuron output. The counterpropagation algorithm is described in the following steps:

**Step 1** Initialise network;

All the weights are initialised to small random values.

**Step 2** Present input for training Kohonen layer;

Present input $X_i(t), X_2(t), \ldots, X_n(t)$, where $X_i(t)$ is the input to neuron $i$ at time $t$.

**Step 3** Calculate the Euclidean distances;

Compute the distance $d_j$ between the input and each Kohonen neuron $j$ from the following:

$$d_j = \sqrt{\sum_{i=1}^{n}(x_i(t) - w_{ij}(t))^2}$$

**Step 4** Determine Kohonen neuron’s output $Z_i$;

After the neuron with the minimum distance $d_j$ is identified as the winner neuron $k$, its output signal $Z_k$ is set to 1 and the other neuron’s output signals are set to 0, e.g.

$Z_i = \begin{cases} 1 & d_k = \text{Min } d_j \text{ for } j = 1, \ldots, N \\ 0 & \text{otherwise} \end{cases}$

**Step 5** Update weights;

$$W_i^{\text{new}} = W_i^{\text{old}} + \alpha(t)(X - W_i^{\text{old}})Z_i$$

The winner neuron's weight vector $W_i$ is adjusted in accordance with Equation (18) to a new position $W_i^{\text{new}}$ in the direction of the input vector. At the beginning of training, the gain term $\alpha(t)$ ($0 \leq \alpha(t) \leq 1$) usually starts at a value between 0.4 and 0.9, then decreases in time as training progresses.

**Step 6** Repeat by going back to Step 2 until the Kohonen layer is stabilised, then go to Step 7;

**Step 7** Present input and desired output for training Grossberg layer;

**Step 8** Update weights;

$$u_{ki}^{\text{new}} = u_{ki}^{\text{old}} + \beta(y_k - u_{ki}^{\text{old}})z_i$$

Where $\beta$ is a network parameter ($0 < \beta < 1$) and $y_k$ is the $k$th component of $Y$ (the desired output vector for the input $X$). In this way only the weights associated with the connections from the winning neuron are modified.

**Step 9** Calculate the actual output of Grossberg neuron:

$$Y_k = \sum_{i=1}^{N} u_{ki}^{\text{new}}Z_i$$

**Step 10** Repeat by going back to Step 7 until the complete Counterpropagation network reaches an acceptable error.

During normal operation, the counterpropagation network can be operated in an interpolation mode. In this mode, more than one node can win the competition on the Kohonen layer. The sum of these Kohonen winning nodes’ output $Z_i$ remains equal to 1 and the non-winning nodes still have zero output signals. The Grossberg layer output is determined by only these winning Kohonen nodes.

**TEST RESULTS**

**Test system**;

To demonstrate the suitability of ANNs as pattern recognition techniques for contingency ranking in power system security analysis, the backpropagation and counterpropagation networks are employed to recognise the performance indexes of various contingencies under different operating conditions on a 5-bus 7-line power system, which is represented in Figure 3:

![Figure 3 The one-line diagram of the 5-bus 7-line system](image-url)
37 seconds for training the BP-V and CPN-V networks with the 101 (voltage type PI) training patterns, respectively. It can conclude that the training time of counterpropagation networks is less than backpropagation networks.

**The flow chart of contingency analysis;**
The flow chart of on-line contingency analysis based on artificial neural networks for fast contingency ranking is shown in Figure 6.

![Figure 6](image)

**Contingency ranking performance;**
After training, there are 300 untrained patterns for testing the four trained ANNs. MW type performance indexes can be recognised by BP-MW and CPN-MW; voltage type performance indexes can be recognised by BP-V and CPN-V. The two kinds of trained neural networks can recognise the performance indexes of 13 contingencies for every test pattern. The performance indexes obtained by the neural networks are compared with the results computed by a full ac load flow program. Table I and Table II shows MW type and voltage type PI ranking lists and performance indexes for a test patterns under heavy loading condition (149.5% load level of the base case), respectively.

The effectiveness of the contingency ranking schemes is usually represented by its capture ratio (CR) and false alarm ratio (FAR) which are defined as following:

1) Capture ratio (CR):

\[
CR(100\%) = \frac{N_{id}}{N_T} \times 100
\]  

Where, \(N_{id}\) is the number of the contingencies to be in violation identified by ANN;

\(N_T\) is the total number of the contingencies which actually result in violation.

2) False alarm ratio (FAR):

\[
FAR(100\%) = \frac{N_{fa}}{N_{TS}} \times 100
\]  

Where, \(N_{fa}\) is the number of false alarms where secure states are identified as insecure ones; \(N_{TS}\) is the total number of actual secure states computed by a full ac load flow program.

**Table I MW PI ranking lists and performance index**

<table>
<thead>
<tr>
<th>Contingency</th>
<th>Full AC</th>
<th>BP</th>
<th>CPN</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID. Number</td>
<td>Rank</td>
<td>PI</td>
<td>Rank</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>9.6</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>5.4</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>3</td>
<td>3.8</td>
<td>3</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>0</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>7</td>
<td>0</td>
<td>7</td>
</tr>
<tr>
<td>7</td>
<td>8</td>
<td>0</td>
<td>8</td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>11</td>
<td>11</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>12</td>
<td>12</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>13</td>
<td>13</td>
<td>0</td>
<td>13</td>
</tr>
</tbody>
</table>

**Table II Voltage PI ranking lists and performance index**

<table>
<thead>
<tr>
<th>Contingency</th>
<th>Full AC</th>
<th>BP</th>
<th>CPN</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID. Number</td>
<td>Rank</td>
<td>PI</td>
<td>Rank</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>9.3</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>0</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>7</td>
<td>0</td>
<td>7</td>
</tr>
<tr>
<td>7</td>
<td>8</td>
<td>0</td>
<td>8</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>11</td>
<td>11</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>12</td>
<td>12</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The four trained artificial neural networks have been tested for both MW and voltage contingency ranking performance on the 5 bus-7 line system. The results are shown in Table III.
Contingency Set:
The members of the set of contingencies will depend on the system involved and the standard of security required. A set of most probable contingencies is first specified. Often these contingencies begin as single component outages. In this research, single line or single load/generator outage are also considered. The set of contingencies on this test system includes 13 outages as follows: 7 single line outages; 4 single load outages; 2 generators outages; When the generator at bus North trips, the bus South will be used as the slack busbar.

Input and output vectors:
The features of input vector should respect the power system operating states. In the test system, the voltage magnitude of every busbar and the active power flow of every line and are chosen as the features of input vector

\[ X = [V_E, V_L, V_M, V_N, P_{LM}, P_{MB}, P_{RS}, P_{RS}, P_{SM}, P_{SE}] \]

The components of output vector \( Y \) are defined as the corresponding performance indexes of 13 contingencies:

\[ Y = [P_{I_1}, P_{I_2}, P_{I_3}, \ldots, P_{I_{12}}, P_{I_{13}}] \]

The architectures of ANNs:
Usually, in the input layer, the number of neuron is depended on the number of features of an input pattern. In the output layer, each neuron corresponds to a performance index of the contingency set. The backpropagation network consists of five layers: 12 neurons in the input layer, 13 neurons in the output layer, 20 neurons in each of the 3 hidden layers. The topology of the backpropagation network is similar in Figure 1. The input and output layers of the counterpropagation network used in this test have the same structure as ones of the backpropagation network. The number of neurons in the Kohonen layer of the counterpropagation network is based on the number of the training patterns in the training set. In the test system 101 training patterns is used. Different size of Kohonen layer has been tried in order to achieve the best performance. Finally 200 neurons are employed in the Kohonen layer. The counterpropagation network architecture is similar in Figure 2.

Training and testing ANNs:
It must be addressed that the training patterns obtained should be representative of the whole range of the power system operating conditions. It is an important requirement. Otherwise, a trained ANN could not recognise the pattern correctly. In this research, the training and test patterns are created with a fixed base case topology while loads and generators are varied between 50% and 150% of the base case in 401 randomly selected steps. This basically reflects on the different load levels which may be encountered in practice. The power system state in each step will be defined by a set of variables as a pattern. The corresponding performance index of 13 contingencies in each step is obtained from a full ac load flow program. The ac power flow program is run 13 times for each of the 401 steps, in other words, the ac power flow program is run 401 times for each of the 13 contingencies. The loads and generations are varied from 50%, 51%, ..., 149%, 150%, then there are 101 patterns which are selected as training patterns. The backpropagation and counterpropagation networks are trained with the 101 patterns. Due to the use of separate MW and voltage type PI to evaluate the severity of contingencies, two different artificial neural networks in either BP or CPN algorithms can be used. One for recognising MW type PI is referred to as ANN-MW; another one for recognising voltage type PI is referred as to ANN-V. The 101 patterns (associating desired MW type performance indexes) trains the ANN-MW, and another 101 patterns (associating desired voltage type performance indexes) trains the ANN-V. The other randomly remaining 300 patterns are fed to the four trained neural networks to evaluate their performances.

The convergence characteristic:
Using a stopping criteria of \( \varepsilon = 0.001 \) the proposed four artificial neural networks converged for all network configurations but required various numbers of iterations. The convergence characteristics of the MW type and voltage type ANNs were studied and shown in Figure 4 and Figure 5, respectively. The study results indicate that the counterpropagation neural network is much faster than the backpropagation neural network during the training process.

Training times:
The program is run on a Pentium 60-MHz PC with 8 MB RAM for evaluating the performance of the proposed neural network approaches. Different networks require different times for training neural networks. In this research, backpropagation networks need about 50000 iterations in order to get the better ranking performance, while counterpropagation networks need only 20000 iterations. It took 8 minutes 20 seconds and 4 minutes 32 seconds for training the BP-MW and CPN-MW networks with the same 101 (MW type PI) training patterns, respectively; it took 8 minutes 36 seconds and 4 minutes...
Table III Ranking Performance based on backpropagation and counterpropagation networks

<table>
<thead>
<tr>
<th></th>
<th>Backpropagation Networks</th>
<th>Counterpropagation Networks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MW PI Ranking</td>
<td>Voltage PI Ranking</td>
</tr>
<tr>
<td>( N_{id} )</td>
<td>276</td>
<td>155</td>
</tr>
<tr>
<td>( N_I )</td>
<td>276</td>
<td>155</td>
</tr>
<tr>
<td>( N_{is} )</td>
<td>11</td>
<td>10</td>
</tr>
<tr>
<td>( N_{TS} )</td>
<td>287</td>
<td>165</td>
</tr>
<tr>
<td>CR(100%)</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>FAR(100%)</td>
<td>3.83275%</td>
<td>6.0606%</td>
</tr>
</tbody>
</table>

Summary of test results:

i) Artificial neural network approach is actually a kind of adapted pattern recognition technique. It can solve various tasks which traditional methods may fail;

ii) The computation time of ANN based contingency ranking methods is much less than existing contingency ranking methods;

iii) The test results show that both backpropagation and counterpropagation networks can provide a high capture ratio for inputs which are not encountered in training process; but the false alarm ratio of the counterpropagation network is not as good as the one of the backpropagation network;

iv) Due to the learning phase of Kohonen layer in counterpropagation networks is an unsupervised clustering phase, the convergence of the counterpropagation networks is much better than the backpropagation network, but some discriminatory information could also be lost in this phase. Maybe it is why the false alarm ratio of the counterpropagation networks is not as good as the backpropagation network;

v) The accuracy of the backpropagation network is very close to the one of the full ac load flow method. Backpropagation network is much more suitable for future on-line applications;

vi) The proposed approach is currently being tested on a specific region of the National Grid Company plc. of United Kingdom. The region consists of 71 busbars and 150 lines. Test results will be reported in due course.

CONCLUSION

This paper proposes a fast contingency ranking scheme which has been formulated as a pattern recognition problem using artificial neural networks. Backpropagation and counterpropagation networks are tested. A comparative study of results by using the proposed approach and a full load flow program is presented. The test results show that backpropagation networks have much better contingency ranking performance than counterpropagation networks.

ACKNOWLEDGEMENTS

The authors wish to acknowledge the financial assistance received from the National Grid Company plc. of the United Kingdom.

REFERENCE


**BIOGRAPHIES**

**Professor L. K. Lo** obtained his MSc and PhD degrees from the University of Manchester Institute of Science and Technology. Currently he is professor of Power Systems and Head of the Power Systems Research Group at the University of Strathclyde, UK. His major research interest is in power systems analysis, monitoring and control using modern mathematical techniques including genetic and evolutionary algorithms, ANN and expert systems. He has been a member of many organising committees of international conferences, committees of professional institution, and is currently visiting and consultant professor to several universities. He is a Fellow of the IEE and is the author/co-author of over 230 technical publications.

**Mr Liangjian Peng** obtained the BSc and the MSc degree in Electrical Engineering from the Hunan University, P. R. China in 1982 and 1988, respectively. He then joined the Department of Electrical Engineering of Hunan University as a lecturer. At present, he is pursuing his PhD degree at the Department of Electronic and Electrical Engineering, University of Strathclyde (UK). His current research interests include power system equivalent, steady state security analysis, applications of artificial neural networks to power system security problems.

**Dr. John Macqueen** graduated as a mathematician from St. Andrews University in 1965, and completed his PhD on differential equations in 1970. He joined the CEGB to do mathematical modelling of their environmental impact. Since privatisation of the electricity industry in 1990, he has worked with the National Grid Company plc in its Technology and Science Division where he heads the Power Systems and Business Modelling Section. Dr Macqueen is a Chartered Engineer and Chartered Mathematician. He is currently an Industrial Visiting Professor in the Department of Cybernetics at Reading University (UK).

**Dr Arthur Ekwue** obtained the BSc(Eng.) degree in Electrical Engineering from the University of Nigeria, Nsukka. He came to Imperial College in 1979 and was awarded the PhD degree in Power Systems in 1982. He obtained the MBA degree in 1994 from City University. Dr Ekwue is a Chartered Engineer and currently with National Grid Company plc's Technology and Science Division as the Senior Project Engineer responsible for Artificial Intelligence Applications with the Power Systems and Business Modelling Section.

**Dr Daniel Cheng** obtained the BSc(Eng.) degree in Electrical Engineering from the University of Hong Kong. He completed his MSc degree in 1978 at the University of Manchester Institute of Science and Technology and was awarded the PhD degree in Power Systems in 1982. Dr Cheng has been involved in the development of Energy Management Systems and commissioned the EMS at the National Grid Company plc. Dr Cheng is a Chartered Engineer and currently with National Grid Company plc's Technology and Science Division as the Senior Project Engineer within the Power Systems and Business Modelling Section.