Early Warning Fault Detection Using Artificial Intelligent Methods

K C P Wong, H M Ryan, J Tindle

University of Sunderland, United Kingdom.

ABSTRACT

This paper describes a research investigation to access the feasibility of using an Artificial Intelligence (AI) method to predict and detect faults at an early stage in power systems. An AI based detector has been developed to monitor and predict faults at an early stage on particular sections of power systems. The detector for this early warning fault detection device only requires external measurements taken from the input and output nodes of the power system. The AI detection system is capable of rapidly predicting a malfunction within the system. Artificial Neural Networks (ANNs) are being used as the core of the fault detector. A simulated medium length transmission line has been tested by the detector and the results demonstrate the capability of the detector. Furthermore, comments on an evolutionary technique as the optimisation strategy for ANNs are included in this paper.

1.0 INTRODUCTION

As a result of increasing competition in the electrical power industry and the requirement for quality power supply, adequate fault detection is becoming vitally important to both companies and consumers. One of the strategies to ensure low running cost is the general avoidance of supply interruption wherever possible. Conventional fault studies are concerned mainly with a 'what if ' scenario i.e. on considering what would happen after a fault occurred, identifying its location and accessing the nature and degree of damage. In contrast, if potential faults could be identified by an appropriate early warning system before a catastrophic fault actually occurs, the chance of power interruption would be greatly reduce. However, few studies have been made concerning early fault detection (EFD) techniques as used by Wong [1]. In a typical power system, the states (voltages and currents) of most bus bar nodes are monitored and gradual changes are analysed. However, in general, because of the complexity of recorded data, faults cannot be easily recognised at an early stage. These faults can often be disguised initially by the complexity of power system operational data.The purpose of using the EFD system is to provide an early warning to the operator when potential faults are identified.

The heart of the EFD system is a hierarchical ANNs structure [2], which consists a network of ANNs, which can be employed to monitor the states of some important components in power networks, such as switchgear and transformers. Each of the individual ANNs is trained to detect minor changes to a component or a equivalent circuit component in a power system model. In the training process, a sufficient number of training patterns are presented to the ANNs repeatedly until the problem is generalised. A training pattern effectively consists of a series of numbers which inform the ANNs what the outputs states should be when identical, or similar, patterns are presented at the input of the ANN. In this case, the training patterns are the states at both ends of the monitored power system section under slightly different 'known' working conditions. The small variations of voltages and currents resulting from small degradation of component values, at sending and receiving ends of the monitored power system section, can be derived, under simulation, or selected from the power industry recorded data. As some of the equivalent components of a power system model do not physically exist, or are inaccessible, they cannot be measured directly by simple measurement methods. Thus, the application of an intelligent technique, such as an ANN method , is obviously desirable. The principle of the EFD method can be applied to various sections of a power system. A typical simplified example will now be given.

High voltages and currents carried by transmission lines in power systems are subject to small changes in state, caused by partial faults, often too trivially small to trigger the conventional protection warming systems. However, these small scale changes may develop and eventually lead to major faults. If, for example, the protective layer for an underground transmission line had an undiscovered partial fault, due to road works damage, the cable could become progressively corroded and, in time, its electrical characteristics could change gradually. A circuit breaker would trip if underground flooding caused a short-circuit and this fault could "black-out" a large area. However, for a network with early warning fault monitoring, the interruption of power supply to certain sections of network could possibly be prevented. The gradual change of impedance of the transmission line provides vital information which can be continuously monitored and analysed by the EFD technique to provide an early detection capability. This approach could alert the operator before the main fault actually occurs enabling, in some situations, appropriate action to be taken, e.g. providing power supply from another circuit and switching out the endangered line prior to a more detailed investigation.

2.0 ARTIFICIAL NEURAL NETWORK

An ANN may be considered as a greatly simplified model of the human brain which can be used to perform a particular task or function of interest.[3] The network is usually implemented using electronic components or simulated in software on a digital computer. The massively parallel distributed structure and the ability to learn and generalise makes it possible for ANNs to solve complex problems that otherwise are currently intractable. A brief description of what and how neural networks are being employed is given below. For more information relating to ANNs, see [3-6].

2.1 Neurons And Synapses

A neuron is an information processing unit that is fundamental to the operation of an ANN. Two basic elements can be identified from a neuron; an adder and an activation function. An adder is used to sum up the input signals, weighted by the respective synapses of the neuron. The activation function limits the amplitude of the output of a neuron. It compresses the permissible amplitude range of the output signal to some finite value [3]. For some gradient descent learning algorithms, such as the Back-Propagation learning method (BP), the activation functions are required to be bounded and differentiable.[4] In this work, the standard sigmoid function was selected which bounds its output range between zero and one.

Synapses are simple connection that can either impose excitation or inhibition on the receptive neuron. Knowledge is acquired by the network through a learning process. The synaptic weights are used to store the knowledge. Through the learning process, the synaptic weights of the network are modified in such a way to map the input patterns to the output patterns. [3]

2.2 Structure of Artificial Neural Networks

The standard Multi-Layer Feedforward (MLF) network is employed as the network architecture in this project. The MLF network is a network of neurons and synapses organised in the form of layers; the input layer, hidden layer and output layer.

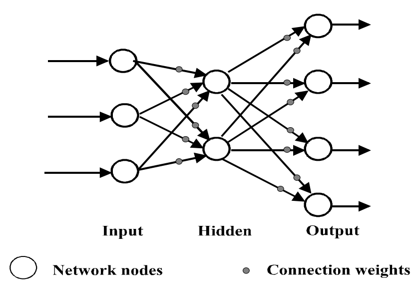

Figure 1. A 3-2-4 multi-layer feedforward network

The function of the input layer is simply to buffer the external inputs to the network. The hidden neurons have no direct connections to the outside world. However, they empower the extraction of higher-order statistics as the network acquires a global perspective despite its local connectivity by virtue of the extra set of synaptic connections and the extra dimension of neural interactions (Churchland and Sejnowski, 1992)[5]. Figure 1 shows the structure of a MLF network.

The source nodes in the input layer of the network supply respective elements of the activation pattern, which constitute the input signals applied to the neurons (computation nodes) in the second layer. The output signals of the second layer are used as inputs to the third layer, and so on, for the rest of the network.

2.3 Learning Algorithm

The procedure used to perform the learning process is called a learning algorithm, the function of which is to modify the synaptic weights of the network in an orderly fashion so as to attain a desired design objective. Many learning methods have been developed in the last few decades. Detailed information concerning possible ANN learning algorithms can be obtained from [3-5].

Two learning algorithms have been used for this project; the standard back-propagation (BP) method and the genetic algorithm (GA). BP has already been successfully applied by several researchers to solve some difficult and diverse problems by training ANNs in a supervised manner. With regard to the BP method, a training set is applied to the input of the network, signals propagate through the network and emerge as a set of output states. An error term is derived from the difference between the desired and actual output values and synaptic weights are then adjusted in accordance with an error correction rule. As the iteration proceeds, the overall error normally approaches zero.[3]

However, the slow rate of learning and the possible premature convergence are limitations of the standard BP learning method. An alternative to the BP is the GA which is an evolutionary algorithm based on the concept of natural selection and evolution, described in [7], [8]. Evolutionary computing techniques are based upon Darwin's theory of evolution where a population of individuals, in this case potential solutions, compete with one another over successive generations, 'survival of the fittest'. After a number of generations, the best solutions survive and the less fit are gradually eliminated from the population [8]. As the GAs can prevent the local solutions when guided by the parallel search strategy, premature convergence can be avoided. The synaptic weights of ANNs are considered to be the chromosomes of an individual. A population of individuals constitutes a pool of potential solutions.

Chromosomes are traditionally represented by binary numbers and standard crossover and mutation are employed as the reproduction operators under a selection scheme. A mapping process is required to convert binary chromosomes back into real numbers. Each individual (synaptic weights) of the new population will be transferred to the ANNs for evaluation. After the fitness values are calculated, a new generation of weights will be genetically created. This process is repeated many times until the pre-defined precision is met. In many cases, standard GAs cannot be used to solve complex problems [10]. Figure 2 shows a representation of a modified GA training method developed by Wong [1].

2.4 Hierarchical Distributed ANNs (HDANNs)

Recently developed HDANNs are advanced neural network architectures [2]. HDANNs consist of several interconnected level of ANNs. The outputs of lower level ANNs are connected to the inputs of higher level ANNs. A typical schematic of a HDANN is shown on Figure 3.

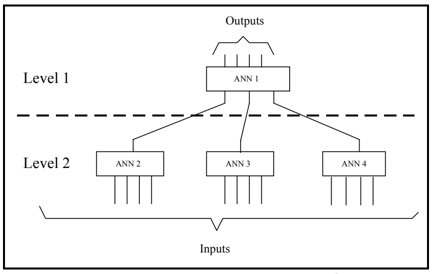

Figure 3. A schematic diagram of HDANNs

HDANNs manifest a number of superior characteristics over conventional methods, exhibiting faster learning rate, smaller training sets and usually lessen the memory storage required. Most importantly, each of the individual ANNs can be trained separately, 'the divide and conquer approach', with the aid of parallel training, the overall training time can be greatly reduced. HDANNs have been applied to solve multiple fault detection problems in this project. Three independent ANNs are employed to monitor the same section of a power system. Each ANN produces an output identifying the potentially faulty components. The outputs from level 2 are fed into the level 1 (Figure 3) decision making ANNs which will further analyse the output data to produce meaningful instructions for the system operator.

3.0 EXPERIMENTS AND RESULTS

The characteristics of most passive components of a power systems can be considered to be a system of connected equivalent resistances, inductances or capacitances. A RLC ∏ circuit was used to construct a model which simulates these characteristics. The schematic diagram is shown in Figure 4. This simple circuit can represent various equivalent power system components or sectors such as a medium length transmission line, low pass filter. In order to verify the capability and reliability of the early warning fault detection system, the RLC ∏ circuit has been intensively tested under many diverse working condition. Soft faults which are caused by slight degradation of one or more impedances were introduced at various location into the ∏ circuit. For single fault testing, an ANNs unit was employed to monitor both the sending and receiving end states. Its outputs would which impedance (resistance, inductance or capacitance) represents a potential problem. In the case of multiple faults testing, the situation were too complex for an single ANNs unit to efficiently handle. Therefore, advanced HDANNs were employed. The experimental results are shown in Table 1 at the end of the paper.

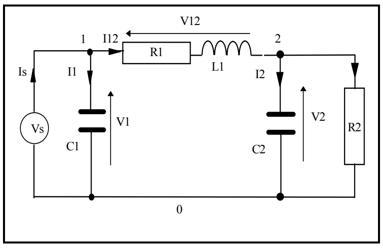

Figure 4. The schematic diagram of the ∏ circuit.

Before the early warning fault detection system can be applied, it must be fully trained. To 'train' the network a training set is applied to the ANN inputs and the learning algorithm invoked so that the output produces the desired response to identify 'soft' faults. Various patterns obtained from different operating conditions are repeatedly submitted to the ANN input units until the problem is fully generalised.

For this ∏ circuit example, training patterns for the ANNs unit were constructed based on the following principle. The values of the impedances in the ∏ circuit were varied in small incremental steps. The changes of impedance of the ∏ circuit affected states at both the sending and receiving ends of the circuit. By specifying a certain degree of impedance change, for example 10% (see Table 1 and Figure 4), as a 'soft' fault, the states at both ends of the transmission line can be used to form the training pattern for the ANNs unit. The numerical data generated often differs only by very small magnitude. These relatively small differences in current and voltage values makes it very difficult for the ANNs fault analyser to detect a 'soft' fault state. The solution to this problem is to pre-process the input data for the ANNs unit. A normalisation pre-process method, which maps the input data to the range of 0 to 1, is applied to maximise the differences among the data. The results from this experiment demonstrated that the pre-process (normalisation) method is essential to obtain a solution and it also helps to reduce the training process time. Table 1 shows the ANNs normalised training pattern for the multiple faults testing. The first two columns list the percentage of change of impedance values, while the next eight columns of numbers show the sending (Vs, Is) and receiving end (Vr, Ir) voltages and currents, respectively. The desired outputs which represent the fault states of the ANNs unit then are listed in the next three columns, where a fault is represented by a logic '1'. Each row of data was produced by changing the value of resistance, inductance and capacitance by only a small step

3.1 Experiment Results of Multiple Faults Testing

A HDANN structure, which consists of four ANN units, is employed in this experiment. Every impedance in the ∏ circuit are monitored by an independent ANN unit. The voltages and currents of both sending and receiving end of the ∏ circuit (see Figure 4) are fed to the three ANN units. The outputs of the each is then connected to the inputs of the final decision ANN unit which further analyses the data and produce meaningful results, such as which impedance(s) of the actual ∏ circuit exceeds the allowable tolerance limits, for the system operator.

Each of the ANN units in the HDANN structure were trained by the BP training method, within approximately 1,000 iterations. A total of 57 sets of different testing cases were used to evaluate the detector. One-third of these were not part of the training set. The last three columns of Table 1 show the results produced by the early warning fault detection system. The desired and corresponding fully simulated results are closely matched with an average accuracy of 99%. Therefore, it is considered that the system can be used to accurately identify soft fault states.

4.0 DISCUSSIONS AND CONCLUSIONS

The concept and methods for EFD or soft fault detection has been outlined and tested. The preliminary results presented have been encouraging and show good potential for the algorithm to be successfully implemented in real power system environments. As fault detection is one of the important processes for reliable operation of any power system, an effective soft detection algorithm could eventually become a standard monitoring application essential for power system operational processes.

During the development process of the early warning fault detection system, it was found that adequate data pre-processing is absolutely essential so that ANN units can efficiently trained. The normalisation method employed in these experiments made the training patterns more distinctly different from each other, hence facilitating analysis by the ANN units.

Experiments on single and multiple fault testing under both DC and AC ∏ circuit have been carried out. Surprisingly, this research has shown that the soft fault detector performs better with more complex AC system (processing complex numbers rather than just scalar values). A possible reason for this is that the addition phase information in the training pattern enhance the pattern distinction.

Another important result of this research is the demonstration of the effectiveness use of HDANN, 'the divide and conquer approach'. Instead of employing a single large ANN unit, several small, independent ANNs were employed. The resulting training time required is about ten times less for the multiple fault testing experiment. In addition, each independent ANN units can be trained separately. In the case of system reconfiguration, only the affected ANNs units need to be retrained.

Evolutionary computing techniques are being evaluated as an ANN optimisation strategy. Preliminary results show that the standard GAs method works very well for simple logic problems, five to ten time better than BP during training. However, as the complexity increases, performance declines markedly and standard GAs are unable to solve most complex analogue electrical problems. The authors can independently confirm the findings of Yao and Liu [10]; which describe the limitations of GAs for ANN applications, (a) the noisy fitness evaluation problem, (b) the permutation problem (competing conventions problem). The use of Evolutionary Programming (EP) algorithms are currently being investigated to avoid crossover operators which are not effective due to the permutation problem. In order to improve the learning efficiency of the EP algorithm operators designed specifically for the problem domain are being devised. Several new EP operators have been developed, and preliminary results are encouraging.

5.0 REFERENCES

[1] Wong P.K.C. & Ryan H.M. & Tindle J : "Power System Fault Prediction Using Artificial Neural Networks", International Conference on Neural Information Processing, September 1996.

[2] Kim K H & Park J K : "Application of hierarchical neural networks to fault diagnosis of power systems", International journal of electrical power and energy systems Vol.15, No.2, p65-70, 1993.

[3] Haykin S : "Neural Networks, A Comprehensive Foundation", Macmillan College Publishing Company, 1994

[4] Patterson D.W. : "Artificial Neural Networks Theory and Application", Prentice Hall, 1996

[5] Chunchland P.S. & Sejnowski T.J. : "The Computational Brain", Cambridge, MA : MIT Press, 1992.

[6] Dasgupta D & McGregor D.R. : "Designing Application-Specific Neural Networks using the Structured Genetic Algorithm", Proceeding of COGANN92 (International Workshop on Combinations of Genetic Algorithms and Neural Networks), IEEE Computer Society Press, 1992.

[7] Goldberg D.E. : "Genetic Algorithms in Search, Optimisation & Machine Learning", Addison-Wesley Publishing Company, INC, USA, 1989.

[8] Paul H & Tindle J : "Passive optical network planning in local access network – An optimisation approach utilising genetic algorithms. British Telcom Technology Journal, April, 1996.

[9] Whitney D & Dominic S & Das R &Anderson C.W, : "Genetic Reinforcement Learning for Neurocontrol Problems", Mechine Learning, Vol 13, P259-284, Kluwer Academic Publishers, 1993.

[10] Xin Yao & Yong Liu, : "Towards Designing Artificial Neural Networks by Evolution", International Symposium on Artificial Life and Robotics, Japan, February 1996.

ADDRESS OF CONTACTING AUTHOR:

K.C.P.Wong
School of Engineering and Advanced Technology,
University of Sunderland
Edinburgh Building, Chester Road, Sunderland,
SR1 3SD, United Kingdom.
Tel : +44 (0)191 515 2865
Fax : +44 (0)191 515 3307

Email and home page :
patrick.wong@sunderland.ac.uk,
http://www.sunderland.ac.uk/~ts0pwo/index.htm

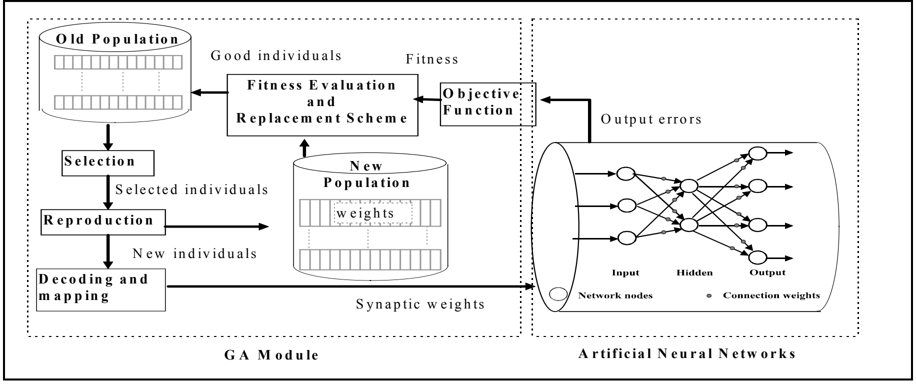

Figure 2. Block diagram of GA training method.

										Desired			Simulated		
Degradation		Vs		Is		Vr		Ir		Fault	Fault	Fault	Fault	Fault	Fault
Component	%	a	bj	a	bj	a	bj	a	bj	(R)	(L)	(C)	(R)	(L)	(C)
R	-50%	1.0000	1.0000	0.4795	0.3268	0.6031	0.2383	0.4687	0.0000	1	0	0	1.00	0.00	0.00
L	-50%	1.0000	1.0000	0.9381	0.6245	0.8227	0.9790	0.9482	0.7650	0	1	0	0.00	0.99	0.01
C	-50%	1.0000	1.0000	0.4226	0.1896	0.4382	0.4715	0.4405	0.5965	0	0	1	0.00	0.01	1.00
R,L	-50%	1.0000	1.0000	1.0000	0.4239	1.0000	0.7792	1.0000	0.1991	1	1	0	1.00	0.99	0.01
L,C	-50%	1.0000	1.0000	0.9271	0.2841	0.8060	1.0000	0.9442	0.8216	0	1	1	0.00	0.99	1.00
R,C	-50%	1.0000	1.0000	0.4413	0.0000	0.5727	0.2591	0.4541	0.0771	1	0	1	0.99	0.03	1.00
R,L,C	-50%	1.0000	1.0000	0.9805	0.0748	0.9821	0.7925	0.9919	0.2460	1	1	1	1.00	0.99	1.00
R	-30%	1.0000	1.0000	0.4695	0.4017	0.5484	0.3236	0.4627	0.2097	1	0	0	1.00	0.00	0.00
L	-30%	1.0000	1.0000	0.7392	0.5691	0.6822	0.7496	0.7442	0.6382	0	1	0	0.00	0.99	0.00
C	-30%	1.0000	1.0000	0.4343	0.3179	0.4495	0.4611	0.4447	0.5641	0	0	1	0.00	0.00	1.00
R,L	-30%	1.0000	1.0000	0.7681	0.4541	0.7781	0.6297	0.7672	0.3145	1	1	0	1.00	0.99	0.01
L,C	-30%	1.0000	1.0000	0.7274	0.3693	0.6691	0.7631	0.7399	0.6782	0	1	1	0.00	0.99	1.00
R,C	-30%	1.0000	1.0000	0.4486	0.2068	0.5307	0.3374	0.4549	0.2571	1	0	1	1.00	0.01	1.00
R,L,C	-30%	1.0000	1.0000	0.7533	0.2516	0.7645	0.6408	0.7615	0.3519	1	1	1	1.00	0.99	1.00
R	-10%	1.0000	1.0000	0.4583	0.4748	0.4937	0.4056	0.4552	0.4145	1	0	0	0.97	0.00	0.00
L	-10%	1.0000	1.0000	0.5463	0.5271	0.5387	0.5416	0.5469	0.5479	0	1	0	0.00	0.95	0.00
C	-10%	1.0000	1.0000	0.4463	0.4464	0.4607	0.4506	0.4489	0.5314	0	0	1	0.00	0.00	0.95
R,L	-10%	1.0000	1.0000	0.5535	0.4906	0.5675	0.5020	0.5523	0.4453	1	1	0	1.00	0.97	0.00
L,C	-10%	1.0000	1.0000	0.5409	0.4620	0.5335	0.5466	0.5450	0.5632	0	1	1	0.00	0.98	0.97
R,C	-10%	1.0000	1.0000	0.4520	0.4103	0.4880	0.4106	0.4530	0.4307	1	0	1	0.98	0.00	0.96
R,L,C	-10%	1.0000	1.0000	0.5478	0.4252	0.5622	0.5068	0.5502	0.4604	1	1	1	1.00	0.98	0.98
R	-5%	1.0000	1.0000	0.4553	0.4928	0.4800	0.4256	0.4532	0.4649	0	0	0	0.01	0.00	0.00
L	-5%	1.0000	1.0000	0.4991	0.5186	0.5026	0.4929	0.4987	0.5305	0	0	0	0.00	0.01	0.00
C	-5%	1.0000	1.0000	0.4493	0.4785	0.4636	0.4480	0.4500	0.5232	0	0	0	0.00	0.00	0.00
R,L	-5%	1.0000	1.0000	0.5024	0.5005	0.5166	0.4731	0.5012	0.4798	0	0	0	0.02	0.02	0.00
L,C	-5%	1.0000	1.0000	0.4962	0.4862	0.4999	0.4955	0.4977	0.5384	0	0	0	0.00	0.02	0.03
R,C	-5%	1.0000	1.0000	0.4522	0.4606	0.4772	0.4282	0.4521	0.4731	0	0	0	0.01	0.00	0.02
R,L,C	-5%	1.0000	1.0000	0.4995	0.4681	0.5139	0.4756	0.5001	0.4877	0	0	0	0.03	0.03	0.03
All	0%	1.0000	1.0000	0.4523	0.5107	0.4664	0.4454	0.4510	0.5151	0	0	0	0.00	0.00	0.00
R	5%	1.0000	1.0000	0.4491	0.5285	0.4527	0.4650	0.4488	0.5649	0	0	0	0.02	0.00	0.00
L	5%	1.0000	1.0000	0.4059	0.5036	0.4301	0.3991	0.4038	0.5015	0	0	0	0.00	0.03	0.00
C	5%	1.0000	1.0000	0.4553	0.5429	0.4692	0.4428	0.4521	0.5069	0	0	0	0.00	0.00	0.03
R,L	5%	1.0000	1.0000	0.4031	0.5212	0.4168	0.4187	0.4019	0.5509	0	0	0	0.05	0.04	0.00
L,C	5%	1.0000	1.0000	0.4091	0.5356	0.4330	0.3964	0.4049	0.4931	0	0	0	0.00	0.04	0.04
R,C	5%	1.0000	1.0000	0.4521	0.5606	0.4555	0.4623	0.4499	0.5566	0	0	0	0.02	0.00	0.03
R,L,C	5%	1.0000	1.0000	0.4062	0.5531	0.4197	0.4160	0.4030	0.5424	0	0	0	0.05	0.04	0.04
R	10%	1.0000	1.0000	0.4460	0.5461	0.4391	0.4844	0.4466	0.6144	1	0	0	0.96	0.00	0.00
L	10%	1.0000	1.0000	0.3600	0.4971	0.3938	0.3541	0.3571	0.4899	0	1	0	0.00	0.96	0.00
C	10%	1.0000	1.0000	0.4583	0.5751	0.4720	0.4401	0.4532	0.4986	0	0	1	0.00	0.00	0.97
R,L	10%	1.0000	1.0000	0.3548	0.5319	0.3679	0.3932	0.3537	0.5872	1	1	0	0.99	0.97	0.00
L,C	10%	1.0000	1.0000	0.3667	0.5607	0.3998	0.3485	0.3594	0.4725	0	1	1	0.00	0.97	0.99
R,C	10%	1.0000	1.0000	0.4517	0.6103	0.4447	0.4789	0.4485	0.5978	1	0	1	0.96	0.00	0.98
R,L,C	10%	1.0000	1.0000	0.3612	0.5953	0.3739	0.3874	0.3558	0.5697	1	1	1	0.99	0.98	0.99
R	30%	1.0000	1.0000	0.4325	0.6157	0.3846	0.5602	0.4367	0.8096	1	0	0	1.00	0.00	0.00
L	30%	1.0000	1.0000	0.1810	0.4778	0.2487	0.1855	0.1751	0.4605	0	1	0	0.00	1.00	0.00
C	30%	1.0000	1.0000	0.4706	0.7040	0.4834	0.4295	0.4574	0.4657	0	0	1	0.00	0.00	1.00
R,L	30%	1.0000	1.0000	0.1708	0.5768	0.1789	0.3006	0.1700	0.7372	1	1	0	1.00	1.00	0.01
L,C	30%	1.0000	1.0000	0.2041	0.6642	0.2687	0.1669	0.1826	0.4023	0	1	1	0.00	1.00	1.00
R,C	30%	1.0000	1.0000	0.4481	0.8069	0.4009	0.5424	0.4417	0.7585	1	0	1	1.00	0.00	1.00
R,L,C	30%	1.0000	1.0000	0.1912	0.7617	0.1981	0.2805	0.1763	0.6782	1	1	1	1.00	1.00	1.00
R	50%	1.0000	1.0000	0.4181	0.6835	0.3304	0.6329	0.4258	1.0000	1	0	0	1.00	0.00	0.00
L	50%	1.0000	1.0000	0.0095	0.4678	0.1044	0.0347	0.0012	0.4561	0	1	0	0.01	1.00	0.01
C	50%	1.0000	1.0000	0.4831	0.8331	0.4947	0.4188	0.4616	0.4325	0	0	1	0.00	0.00	1.00
R,L	50%	1.0000	1.0000	0.0000	0.6243	0.0000	0.2220	0.0000	0.8925	1	1	0	1.00	1.00	0.01
L,C	50%	1.0000	1.0000	0.0524	0.7715	0.1405	0.0000	0.0143	0.3498	0	1	1	0.00	1.00	1.00
R,C	50%	1.0000	1.0000	0.4415	1.0000	0.3569	0.6013	0.4328	0.9128	1	0	1	1.00	0.01	1.00
R,L,C	50%	1.0000	1.0000	0.0359	0.9243	0.0338	0.1840	0.0099	0.7849	1	1	1	1.00	1.00	1.00

Where
Vs is sending end voltage; Is is sending end current; Vr is receiving end voltage; Ir is receiving end current;
F1 = 1 indicates a fault on R1; F2 = 1 indicates a fault on R2; F3 = 1 indicates a fault on R3.

Table 1. Training patterns for the ANNs based 'soft' fault detector.