ANN Based Relay Algorithm for the Detection of High Impedance Faults

L.A. Snider
Hong Kong Polytechnic University
Department of Electrical Engineering, Hong Kong

Y.S. Yuen
University of Strathclyde
Power Systems Research Group
Glasgow, Scotland

Abstract - This paper presents an artificial neural network (ANN) based digital relay algorithm for high impedance fault detection in distribution systems. A stochastic HIF model was simulated by using MODELS in EMTP, and the relay was trained to recognise the arcing characteristics of high impedance faults (HIFs). The input signals to the relay comprise low order harmonics of the three phase residual quantities and symmetrical components, which are calculated and fed to perceptron and backpropagation based neural networks. Digital sampling and anti-aliasing is incorporated, as well as 10% random noise. The performance of the relay was verified in simulations of different distribution systems and in the face of several potential confounding factors, and the relay was found to perform well. Since only low order harmonic signals are used, special instrument transformers would not be required, and this is of significant advantage over relays that propose to use the high frequency characteristics of arcing faults.

Keywords: Protection, High Impedance Faults, ANN, Modelling, EMPT

I. INTRODUCTION

High impedance faults are faults that do not draw enough current to operate conventional overcurrent protective devices (e.g. fuses, reclosers, relays). Unlike other faults that result in a high fault current, HIFs give a very low fault current with typical magnitudes ranging from 40A to 100A. Consequently, conventional non-unit detection will either result in false trips or totally fail to detect the presence of the faults. Although only a few percent of the faults are high impedance faults, the failure of HIF detection can lead to potential hazard to human beings and potential fire.

HIFs often occur when an overhead conductor breaks and falls to the ground or a tree. Usually the surface the downed conductor in contact with has a poor conductivity and arcing is involved. The resulting current and voltage waveforms are often characterised by cyclical patterns and distortions caused by the arcing and/or nonlinearity of the fault impedance.

Some detection schemes have been proposed [1-5] which are based on fractal techniques, digital signal processing, neural networks, high frequency noise pattern and dominant harmonic vector. They offer potential solutions to the problems currently associated with conventional schemes. However direct calculation of fractal dimensions is not effective owing to the relatively short data sets available for estimation. The use of high frequency harmonics is not practicable owing to the filtering by substation instrument transformers.

ANN based detection schemes offer robust relaying because of its ability to match pattern and tolerate noise. Moreover, these schemes also provide the potential for on-line training and customisation using actual field HIF data. In this paper, an ANN based digital relaying scheme is proposed which evaluates the low order harmonics of residual quantities and symmetrical components. Digital relaying was fully simulated by incorporating sampling, anti-aliasing and random noise. A stochastic high impedance fault model is also presented.

II. ARTIFICIAL NEURAL NETWORKS

An artificial neural network is a dynamic system with one-way interconnections. Its invention was first inspired by the actual learning process taking place in human brains. An ANN simulates this complex learning process. Perceptrons belong to nonbiological class of ANN and is one of the biggest branches in ANNs. The simplest perceptron is a single layer network whose weights and biases can be trained to produce a correct target vector when presented with the corresponding input vector. The advantages of using perceptron lie in their use as calculation tools and not in the insight they give to neural operation. Back propagation, branches from perceptron [6], was created by generalizing the Widrow-Hoff learning rule to multiple layer networks and nonlinear differentiable transfer functions. Indeed, both perceptron and back propagation are excellent tools for pattern classification.

Representative distribution systems were modelled using ATP-EMTP, and the MATLAB ANN Toolbox was selected for the implementation of perceptron and back propagation because of its simplicity and flexibility. The paper is aimed to show the feasibility of HIF detection using low order harmonics so the complexity of the neural networks is not of top priority. In fact, it was found that the ANNs from the ANN Toolbox could both be trained and verified satisfactorily.

Perceptron: The perceptron network is trained to respond to each input vector with a corresponding target output vector whose elements are either 0 or 1. The perceptron learning rule is applied to each neuron in order to calculate the new weight and bias. Convergence on a solution in finite time can be obtained if a solution exists. The perceptron neuron, which has a hard limit transfer function, is shown below in detail and in abbreviated notation as in Figure 1.

Each external input is weighted with an appropriate W, and the sum of the weighted inputs is sent to the hard limit transfer function, which also has an input of 1 transmitted to it through the bias. The transfer function returns a logic 0 or 1. The hard limit transfer function is shown in Figure 2.

Back propagation: In MATLAB, the back propagation learning rules are to adjust the weights and biases of networks so as to minimize the sum squared error (SSE) of the network. This is done by continually changing the value of the network weights and biases in the direction of steepest descent with respect to error. This is called a gradient descent procedure. Changes in each weight and bias are proportional to that element's effect on the sum-squared error of the network. Figure 3 shows an elementary back propagation neuron with R inputs.

In the actual implementation of perceptron and back propagation, the basic structure of the network consists of at least one input layer, one output and one hidden layer. More complicated classification usually requires more than one hidden layer with more neurons to match the inputs to the appropriate outputs.
Instead of hard limit function, log-sigmoid function as shown in Figure 4 was employed as the transfer function in the hidden layers. The main purpose of using the log-sigmoid function is to limit the output in the range of 0 to 1.

III. DETECTION SCHEMES

Perceptron and back propagation are renowned for pattern recognition. The aim here is to demonstrate that with proper training, the ANNs can acquire HIF detection. The networks were trained by feeding them with input vectors and the corresponding target vectors. An input vector is one consisting of magnitudes of low order harmonics, determined through a Fourier transform, which are considered to be able to reveal the presence of HIFs.

The data in the training set were obtained from simulation results based on a typical distribution system as shown in Figure 5. It depicts the sample study system of a radial distribution feeder with linear, nonlinear, solid state loads, voltage correction capacitor banks and an equivalent HIF arc model. Since disturbances resulting from HIFs may resemble those from other contingencies inherent to distribution systems such as capacitor switching and single phase load switching, it is therefore necessary to include these confounding factors in the training cases to ensure that the ANNs will not be confounded even under a high level of ambient harmonics.

Digital simulations were performed using EMTP for different types of faults, fault location and other contingencies such as capacitor switching, single phase load switching, etc.

IV. STOCHASTIC HIGH IMPEDANCE FAULT MODEL

The random nature of arcing during high impedance faults is implemented in the simulation through MODELS. A MODELS-controlled type-13 switch is connected to ground through the fault resistance R_f which has a different value in different data cases to produce a variety of fault current magnitudes. The simulation of arc instability is done by using the random number generator in MODELS. The values for fault arc voltage and extinction period are stochastically chosen out of a population of predefined values. The switch is open initially and it closes shortly after the start of the compilation. Following the start of the fault, a random value of arc voltage is chosen randomly to account for the stochastic variables related to arc voltage, including the nature of the distance between the transmission line and the object it is touching (e.g. a tree). Since the fault resistance is almost purely resistive, the fault voltage is almost in phase with the fault current. This implies that the voltage magnitude is relatively small at zero-crossings of the fault current, and the arc extinguishes when arc voltage gets larger than the system voltage. The MODELS routine keeps monitoring the system voltage of the faulty phase and the switch reopens once its absolute value gets lower than the corresponding arc voltage. The extinction period begins and an extinction duration is statistically chosen. The switch recloses after the extinction duration and another cycle begins with a new set of randomly chosen variables. The stochastic fault current is shown in Figure 6.

V. TEST CASES

Data for training were obtained by simulation based on the distribution system in Figure 5. A total of 30 test cases were built for training and another 23 cases were built for verification. Contingencies which may confound the relay include capacitor switching, single phase line switching and nonlinear load switching. The different events which were used for training and verification of the neural network include:

1. Capacitor switching at the line receiving end.
2. Capacitor switching at the line sending end.
3. Single phase load switching at the line receiving end.
4. Non-linear load switching at the line receiving end.
5. High impedance fault at the line sending end.
6. High impedance fault at the line receiving end
7. High impedance fault at the middle of the line.

Each of the 30 training cases has different combination of events and values of circuit parameters. The value of the fault resistance is also different in different cases. The distribution system of Figure 5 was used for training, while for verification, 15 cases were built from the distribution systems shown in Figures 5, 7 and 8. To further test the versatility of the trained ANNs, HIF current was simulated using simplified 2-diode model [8] and erratic fault model [9] in 8 cases in the verification set. These fault currents are shown in Figures 9 and 10.

VI. CHOICE OF INPUT SIGNALS

HIFs are generally non-symmetrical in nature. Residual quantities and symmetrical components were considered likely candidates because they are well known to represent the unbalance of power system. In order to find the best candidates, other quantities such as phase currents, phase voltages were also used at first. But they did not seem to contain substantial information for HIF detection and the chosen neural networks could not be trained successfully. To enhance the practicality of the detection scheme, low order harmonics were given priority over high order ones and the number of inputs required were kept as small as possible. After trying different combinations of the likely candidates, two sets of candidate inputs were selected. The first one consists of the magnitudes of the first and the third harmonics of residual current (I_R), the second harmonic of residual admittance (Y_R, defined as I_R/V_R), the first and third harmonics of residual voltage (V_R) and the second harmonic of residual power (P_R defined as I_RV_R). The second set comprises the magnitudes of the third harmonics of negative and zero sequence current and voltage (I_{-3}, I_{03}, V_{-3} and V_{03}).

VII. DIGITAL RELAYING

Digital relaying was simulated through the incorporation of sampling, anti-aliasing and random noise. Different sampling rates were used for digitisation and anti-aliasing filter, which is associated with the sampling process, was incorporated in the simulation circuit. Random noise was added to the input signals before they were fed to the ANN.

Results show that at least 20 samples had to be taken in one power cycle to result in recognisable signals at power frequency, i.e. 20 x 50 = 1kHz for 50Hz signals. The highest harmonic used is third harmonic and therefore a sampling rate of at least 3kHz was required. To enhance the performance of the ANNs, the final sampling rate adapted is 4kHz.

Anti-aliasing filter was implemented as a 2-stage RC low-pass filter with a cut-off frequency of half the sampling frequency, i.e. 2kHz. The ladder realization is shown in Figure 11.

During the AD process, the resulting signals are prone to different kind of errors. The total error altogether can be treated as a random process. To take this error into account, a 10% random noise was added to all outputs from the ADC. The effects of anti-aliasing filter on a phase voltage and the effects of anti-aliasing and noise on the outputs from ADC are shown in Figures 12, 13, 14, 15, 16 and 17.

VIII. ANN TRAINING

Both perceptron and back propagation were used for training. Both were successfully trained and verified. Perceptron involves a less complicated network structure, but its applicability is limited by the fact that its outputs can only take on two values: either 0 or 1. On the other hand, back propagation involves a more complex network architecture and gives outputs in a defined range which makes further analysis feasible.

In MATLAB, the implementation of ANNs is performed by means of matrix manipulation of the inputs, outputs, weights and biases vectors. Vectors from a training set are presented to the networks sequentially. If the network's output is correct no change has to be made. Otherwise the weights and biases are updated based on the network's training algorithm. The entire pass of all the input vectors is called an epoch.

Effective training of an ANN requires that all the vectors in different rows span similar numerical ranges. In this particular case, the required range of the elements in the input matrix is –1 to +1. The scaling for the selected magnitudes of residual quantities is I_R x 10, Y_R / 100, V_R x 100, P_R x 100 and that for symmetrical components is I_{-3} x 10, I_{03} x 10, V_{-3} x 100 and V_{03} x 100.

Apart from scaling, the choice of ANN architecture also facilitates the convergence of the solution. For perceptron, a one-layer network was employed. For feed forward networks, the number of hidden layers required is dependent on the complexity of the input and output vectors in the training set. Three logsigmoid hidden layers with 15 neurons in each hidden layer was found to be the optimum network structure. However, plain back propagation was too slow to achieve an error goal of 0.001. In some cases it did not even converge. The pitfalls of back propagation are mentioned in reference [10]. Nevertheless, the MATLAB toolbox provides some ways to improve the performance of back propagation. The use of momentum can prevent the solution from being trapped in one of the local minima and hence facilitate the reaching of the global minimum. Adaptive learning rate makes use of delta correction algorithm and attempts to keep the learning step size as large as possible while keeping learning stable. The improved network achieved the error goal of 0.001 after 130 epochs with an initial learning rate of 0.4.

IX. TRAINING RESULTS

The trained network was verified with a verification data set comprising 23 cases. Logic 0 and logic 1 represent the absence and presence of the fault respectively. Cases in the verification set have either different parameters using the same distribution system as in the training set or distribution system with completely different configuration. The success rate of verification was found to be 100% with a maximum error of 0.001. Both sets of candidate inputs gave satisfactory verification results with symmetrical components giving more reliable and accurate outcomes.

X. TRIPPING CRITERIA

For perceptron, outputs can only be 0 or 1. For back propagation, however, outputs can range from 0 to 1. In general, outputs equal to or more than 0.9 are regarded as an indication of the presence of HIF and the relay trips, while the outputs equal to or less than 0.5 are regarded as an indication of the absence of the fault. When the output is in the range of 0.5 to 0.9, a warning signal is issued and the operator will inspect the power system to judge what action should be taken.

Although the trained network could verify all those cases in the verification set, the network may still be confounded by contingencies which the network never encountered before. A simple threshold relay tripping criterion can be based on the number of HIF relay faulty classifications in a continuous period of ten cycles. The threshold of five positive identifications in ten consecutive cycles will result in tripping.

XI. CONCLUSIONS

The paper presents a practical application of ANN in HIF detection making use of low-order harmonics of residual quantities or symmetrical components as the inputs the ANN. The proposed ANN algorithm has been shown to be an effective relaying scheme that can be readily implemented using available ANN hardware chips or software. The trained ANN reacts promptly to HIFs and has a high success detection rate. The relay scheme can be trained using realistic data from digital computer simulation or field measurements. It can be retrained on-line using actual field measurements and can be implemented using available DSP and parallel hardware. The proposed detection scheme only utilises low harmonics as inputs which greatly enhances its feasibility and flexibility by avoiding the use of expensive specialised instrument transformers.

XII. ACKNOWLEDGEMENT

The authors would like to thank the University of Hong Kong, where the simulation work for this project was performed.

XIII. REFERENCES:

[1] A.V. Mamisev, B. D. Russell and Carl L. Benner, *Analysis of High Impedance Faults Using Fractal Techniques* (IEEE Transactions on Power Systems, Vol. 11, No. 1, February 1996) 435-440.

[2] D. J. Jeerings, J. R. Linders, *Ground Resistance Revisited* (IEEE Transactions on Power Delivery, Vol. 4, No. 2, April 1989) 949-956.

[3] Sonja Ebron, D. L. Lubeman and Mark White, *A Neural Network Approach to the Detection of Incipent Faults on Power Distribution Feeders* (IEEE Transactions on Power Delivery, Vol. 5, No. 2, April 1990) 905-1004.

[4] B. D. Russell and R. P. Chinchali, *A Digital Signal Processing Algorithm for Detection of Arcing Faults on Power Distribution Feeders* (IEEE Power Engineering Society 1988 Winter Meeting, NY, 88WM 123-2).

[5] A. M. Saraf, L. A. Snider, K. Debnath, *A Neuro-Fuzzy Based Relay for Global Ground Fault Detection in Radial Electrical Distribution Networks* (International Conference of Electrical Engineering, Tehran, Iran, May 1993).

[6] Robert L. Harvey, *Neural Network Principles* (Prentice Hall).

[7] Howard Demuth and Mark Eeale, *Neural Network Toolbox User's Guide for use with MATLAB* (The Math Works Inc).

[8] V. L. Buchhloz, M. Nagpal, J. B. Neilson, R. Parsi-Feraidoonian and W. Zarecki, *High Impedance Fault Detection Tester* (IEEE Transactions on Power Delivery Volume 11 p. 184-190 Jan 1996).

[9] Carl L. Benner, B. Don Russell, *Practical High Impedance Fault Detection for Distribution Feeders* (Rural Electric Conference, 1996).

[10] Anil K. Jain and Jianchang Mao, *Artificial Neural Networks: A Tutorial* (IEEE March 1996 Computer Theme Feature 0018-9162/96).

XIV. FIGURES

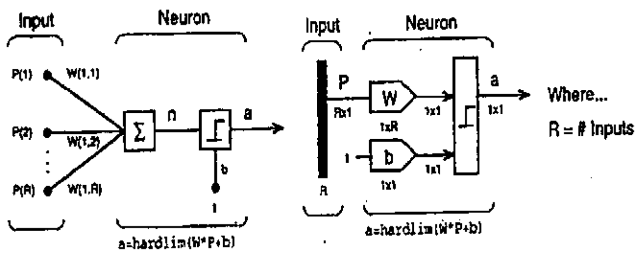

Fig. 1. MATLAB abbreviated notation for perceptron [7]

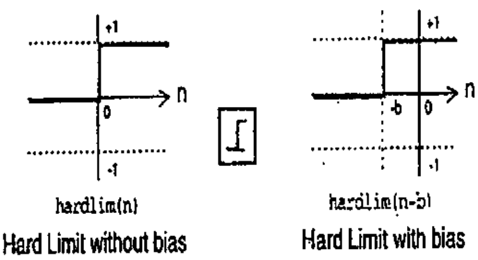

Fig. 2. Hard limit function in MATLAB [7]

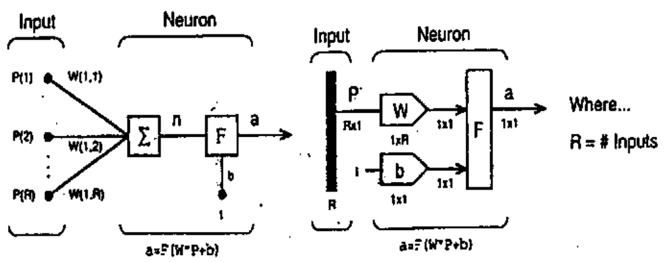

Fig. 3. MATLAB abbreviated notation for back propagation [7]

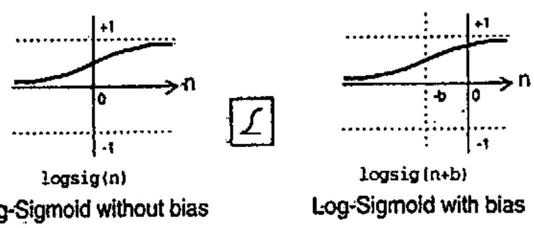

Fig. 4. The log-sigmoid function [7]

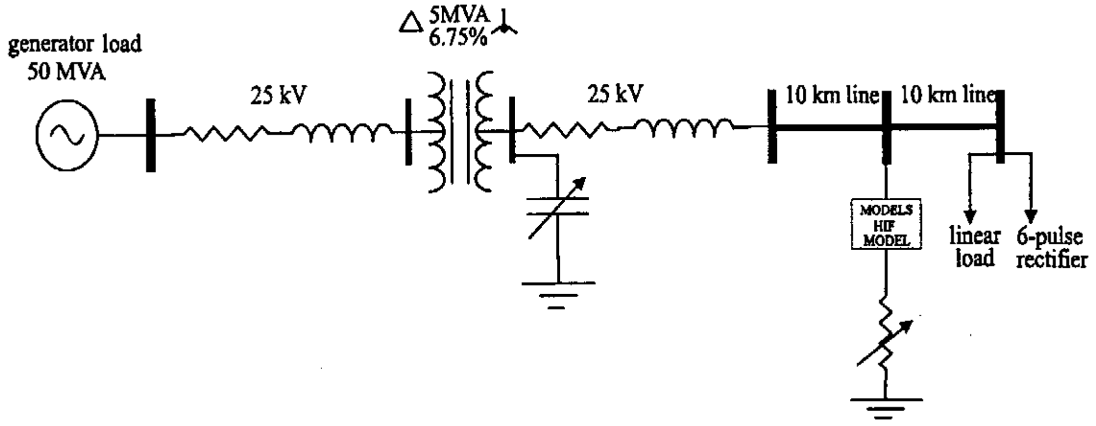

Fig. 5. Distribution system for training

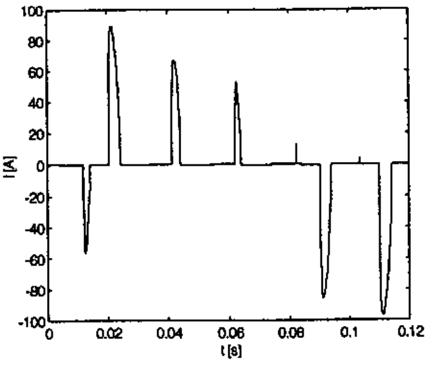

Fig. 6. Stochastic HIF current

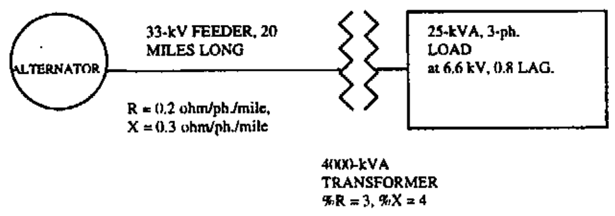

Fig. 7. Distribution system for verification

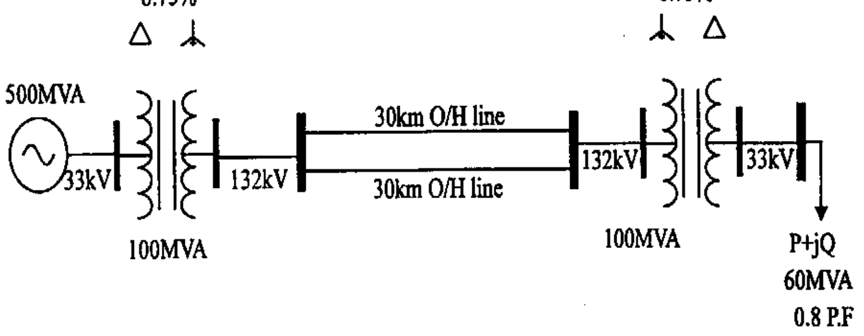

Fig. 8. Sub-transmission system for verification

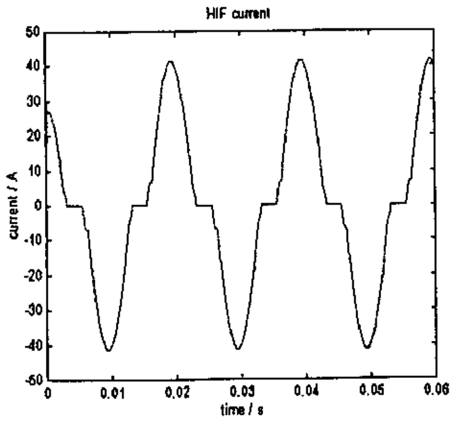

Fig. 9. Fault current by simplified 2-diode fault model

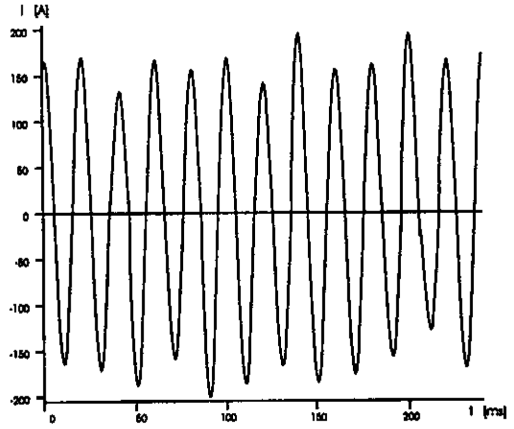

Fig. 10. Fault current by erratic fault model

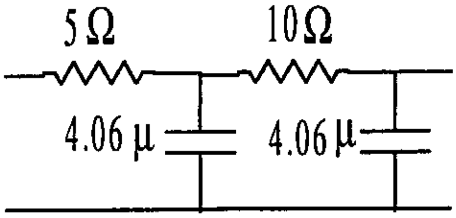

Fig. 11. Two-stage low pass filter

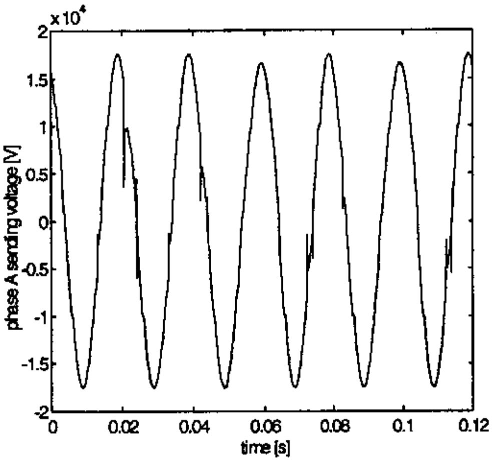

Fig. 12. Sending voltage without filter

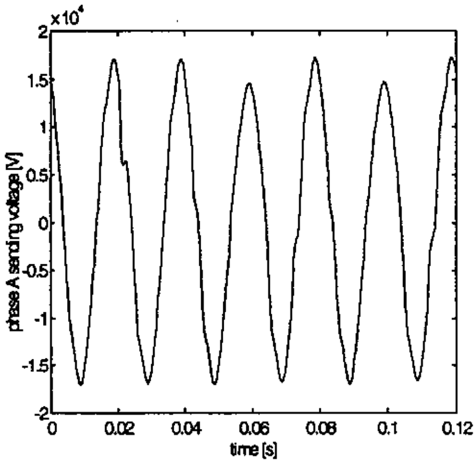

Fig. 13. Sending voltage with filter

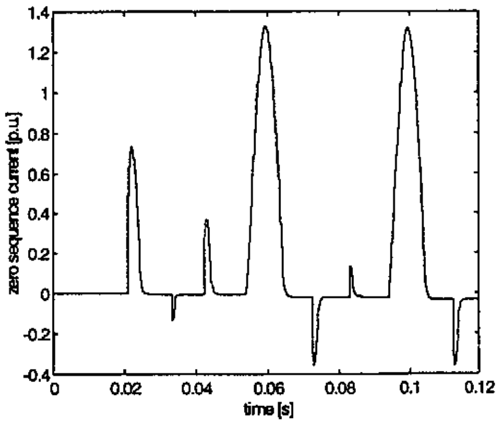

Fig. 14. 0-sequence current without filter

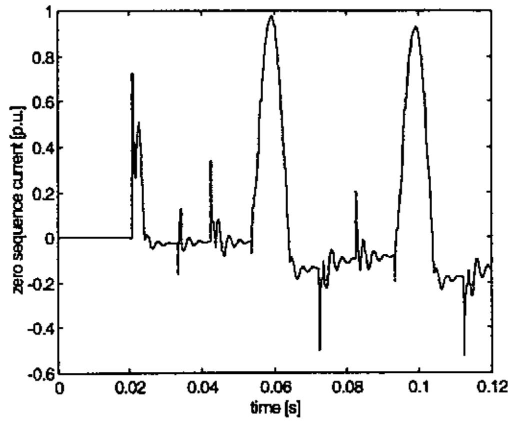

Fig. 15. 0-sequence current with filter

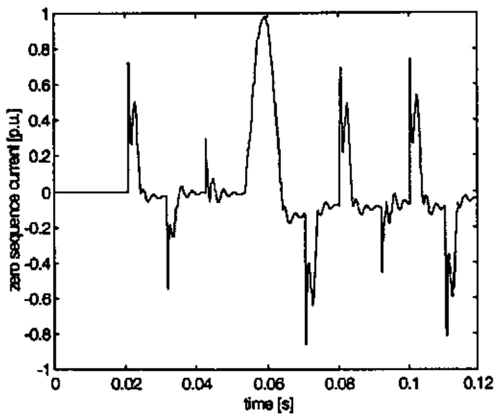

Fig. 16. 0-sequence current without noise

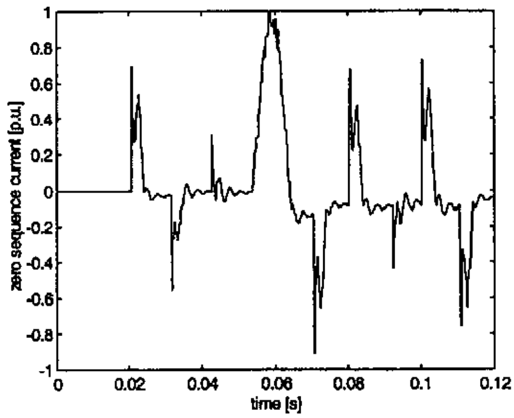

Fig. 17. 0-sequence current with noise