High-impedance fault detection system using an adaptive element model

C.J. Kim, PhD
B.D. Russell, PhD

Indexing terms: Adaptive filter, Fault detection, Scoring rule, Uncertainty reasoning

Abstract: An adaptive-element model approach for high impedance fault detection is presented. The system employs multiple detection algorithms using different frequency parameters. To obtain a final fault/nonfault classification for a power distribution feeder, a revised adaptive-element model is applied using expert weights associated with each algorithm–parameter pair. The adjustment of the weight, assigned by experts, is performed by an elliptic formula. This revised adaptive-element model is different from the original adaptive-element scheme in two components: the combination method for multiple inputs and the learning algorithm for weight correction. This system when tested using examples shows good performance, especially in terms of security against false identification.

1 Introduction

The clearing of distribution line faults is usually accomplished by devices which can sense the overcurrent produced by a fault and react to disconnect the faulted section of the feeder from the healthy section. However, high-impedance faults do not draw sufficient fault current to be detected by such a conventional protective scheme. Such faults may be cause by a downed conductor on the ground or in contact with a grounded object. Arcing is often associated with these faults, which may result in a fire hazard or damage to the publics.

The behaviour of an arc has been studied by many scientists and for a power system can be summarised as follows. If two conductors are separated by a small gap and have a small potential difference between them, the air acts as an insulator. As the potential difference is increased, the resistance of the air gap decreases and the current flows between the conductors. Rapid ionisation accounts for the sudden ability of the air to conduct current [1]. However, because of the varying conditions of the gap distance, loading level, and surface types, it is most difficult to formalise an arc model to calculate the arcing current. Therefore, almost universally, researchers in the detection of high-impedance faults concentrate on the harmonic and noise currents generated by the arc. Current harmonics generated by arcing have their origin in the nonlinear voltage–current characteristics of the arc. The harmonic current characterised by an arc is variable and odd, even, and subharmonic components are present.

Many techniques have been proposed for dealing with the long standing problem of undetected downed distribution conductors. Some of the detection techniques proposed are: energy algorithm [2], randomness algorithm [3], phase relationship algorithm [4], sequnce component algorithm [5], and amplitude ratio algorithm [6]. Some involve the use of only standard substation relaying inputs; others require the addition of special equipment, either at the substation or at distributed locations on the feeder. Several of these techniques have been implemented, either at the prototype level or at the production level; others have only been suggested. The discussion on each detection technique and of its performance can be found in References 7 and 8.

It is relatively easy to detect the presence of any fault, including a high-impedance fault, on a distribution feeder. However, it is a most difficult task to distinguish high-impedance faults from many normal system events and activity. In addition to the characteristic of very low fault currents, high-impedance faults behave differently under different fault situations and, in many cases, emulate the current signatures of normal switching events. The discrimination of faults from these normal events determines to a large extent the balance between security and dependability for a distribution protection system. Research over the past few years has led to the development of several different detection methods which have shown good individual performance [9–12]. However, it has become apparent to these researchers that no single algorithm will offer a sensitive and discriminatory fault detection. Given our knowledge of the behaviour of low current faults and the satisfactory detection performance of several algorithms, we decided to use multiple algorithms with several different electrical parameters in the construction of a detection system [13].

Because of the different performance of each algorithm, when we design a detection system we consider the performance index of each detection algorithm. The performance of each detection algorithm is expressed in terms of experts' confidence (or basic weight). One of the many difficulties associated with the development of a detection system is the fact that the information from each detection algorithm contains a considerable degree of uncertainty. Therefore, the output indication from each algorithm is converted, by experts' confidence on its performance, into the level of weight expressed by a number between 0 and 1. To accommodate these multiple weights of the algorithms, an intelligent combination of multiple pieces of evidence is needed.

The evidence of a status can be derived from the

Paper 9149C (P11), first received 18th May and in revised form 20th August 1992

The authors are with the Department of Electrical Engineering, Texas A&M University, College Station, Texas 77843-3128, USA

experts' basic weights and the algorithm's input to the detection system. The supportive and nonsupportive evidence of fault are combined into a final combined evidence for fault identification. In addition, we show how basic weights can be adjusted using experts' experience combined with an adaptive classification feature for online adaptation to changing feeder situations. The principal technology in this practice is very much similar to a neural network model: the adaptive element or filtering approach. However, the components of the overall system are not the same. Our detection system is based on the combination of the weights (confidence or beliefs) on each detection algorithm and the calibration of these weights with known patterns of events. The combination of the weights is accomplished with the evidence combination formula proposed by Dempster and Shafer [14]. For learning algorithm, we used a scoring rule instead of well known LMS (least mean square) algorithm. The revised adaptive-element model is good especially when the input vector X, which is produced by the detection algorithms, contains uncertain information.

2 Adaptive filtering network

The adaptive filtering network is said to originate from B. Widrow and M.E. Hoff's adaptive linear element. As the name implies, the idea behind this network is to make a system so that it can adjust filtering noise from a signal. The significance of the adaptive filtering approach is that this network learns through an iteration procedure. This significance is well reflected in the frequent use of this network as a neural network model. This adaptive linear element also introduced a learning law, the LMS algorithm. More on the history and basic introduction of this network can be obtained from the book written by Caudill and Butler [15].

In a two-input adaptive linear element (Adaline), we separate input patterns into two categories depending on the values of the weights. A critical threshold condition occurs when the linear output S equals zero:

$$S = X_1 W_1 + X_2 W_2 + W_0 = 0$$

therefore,

$$X_2 = -\frac{W_1}{W_2} X_1 - \frac{W_0}{W_2} \tag{1}$$

where W_k indicates the kth weight and X_k the kth input element, and W_0 is a bias weight connected to a constant input $X_0 = 1$. This bias weight is to effectively control the threshold level. This relation forms a graph of a line having slope of $-W_1/W_2$ and intercept $-W_0/W_2$.

The LMS algorithm of a single Adeline follows the minimal disturbance principle. The weight update equation for the original form of the algorithm is

$$W_{k+1} = W_k + \alpha \frac{\varepsilon_k X_k}{|X_k|^2} \tag{2}$$

The present error ε_k is defined to be the difference between the desired response d_k and the linear output $S_k = W_k^T X_k$ before adaptation:

$$\varepsilon_k = d_k - W_k^T X_k \tag{3}$$

In accordance with the LMS rule of equation, the weight change is

$$\Delta W_k = W_{k+1} - W_k = \alpha \frac{\varepsilon_k X_k}{|X_k|^2} \tag{4}$$

Therefore, the error is reduced by a factor of α as the weights are changed while holding the input pattern fixed (α-LMS rule) [16]. A single Adaline with n binary inputs and one binary output, is depicted in Fig. 1. This Adaline system has been applied to weather forecasting with a single neuronal model, speech recognition with real-time base, noise cancelling, and echo cancelling [17].

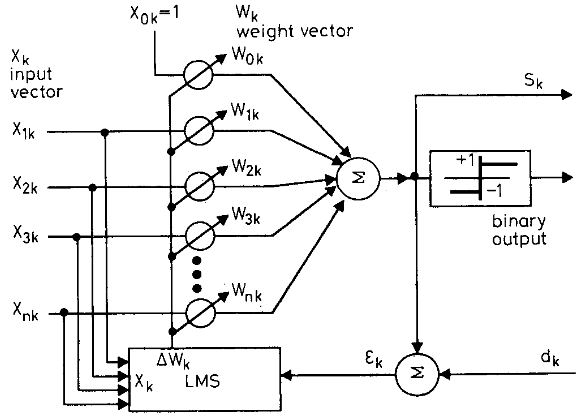

Fig. 1 *Single Adaline with n binary inputs and one binary output*

This adaptive element seems to hold promise for an adaptive detection of high impedance faults. However, with uncertainty inputs, we cannot use the original adaptive element model without changes. When the input vector contains some degree of uncertainty expressed by experts' beliefs or confidence levels, the combination method of the adaptive element is not appropriate. The reasoning method for detection with multiple algorithms requires a method for combining the support for a hypothesis (i.e. fault status), or for its negation. Two components are revised: the combination method and the weight correction learning algorithm. We used an uncertainty reasoning method for input combination and the performance maximising scoring rule for weight correction.

The Dempster–Shafer model recognises the requirements for the combination of the uncertainty information and also provides a formal proposal for its management [14]. The proposed scoring rule for a learning algorithm is very different in principle: while LMS is to minimise the disturbance, the scoring rule is to maximise the overall performance of the detection system [18].

3 Fault detection with adaptive element model

In our detection system, we have multiple pieces of information from the algorithms. And each algorithm produces, as its attribute, the number of fault indication during a certain time interval. From the weights of the algorithms and the number of fault indications, the adjusted weights are obtained. Then, with these adjusted weights and the fault or nonfault indicating information from the detection algorithms, we calculate the supportive evidence and nonsupportive evidence. The combination of the multiple information is actually the combination of the two pieces of evidence: supportive and nonsupportive evidence about the fault.

The variables used in the revised adaptive-element model are three column vectors: the input vector (X) which consists of the information from multiple algorithms, the basic weight vector (W_b) which is originally assigned to each algorithm, and adjusted weight vector ($\boldsymbol{W}_a$) derived by an elliptic formula using the basic weight and the number of fault indications of the algorithms. The function units are the weight adjustment unit, the evidence calculation and combination unit which calculates the supportive and nonsupportive evidence based on the adjusted weights, and the score calculation and calibration unit which derives the performance score with given weights and, if necessary, corrects the weights into the calibrated weights. Fig. 2 sketches the structure of the detection system with four inputs. The detailed explanation of the system follows.

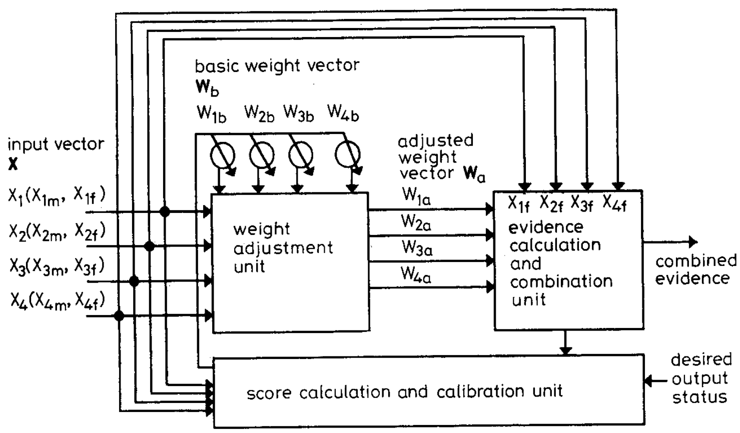

Fig. 2 *Structure of adaptive detection system*

3.1 Input vector

This individual input has two components: the maximum indication number (X_m) and the number of fault indications (X_f). To explain these two input components, we need to mention about decision time window. In the detection of high impedance faults, we do not want to make a quick and wrong decision based on only one set of indications from detection algorithms, for the reason of security (against false indication of fault). Utilities representatives have stressed that one of their main concerns is the minimisation of false detections. The reason for this position is that, while energised downed conductors are a public safety hazard, frequent unnecessary service interruptions can pose safety problems of their own: traffic signals off, lighting lost in homes and businesses, the interruption of factory processes, and so on. Certain normal events like switching might invoke detection algorithms to indicate a fault [19]. So we may have a long (when we compare with an overcurrent relaying situation) decision time window of, for example, 30 seconds. However, we generate a final decision of classification every second based on the matrix of the set of the fault/nonfault indications of the detection algorithms, by adding the newest indication and removing the oldest indication.

Because each detection algorithm has different processing time for its indication, in the long decision time window, there are multiple numbers of fault/nonfault indications. Within this decision time window, theoretically, we have the maximum indication number of each detection algorithm and the number of fault indications. Therefore, each algorithm input (X_i) in the input vector $\{\boldsymbol{X} | X_1, X_2, \ldots, X_n\}$ holds two pieces of information: the maximum indication number X_{im} and the number of fault indications X_{if} where $i = [1, n]$.

3.2 Basic weight vector

The output indications of each detection algorithm have different uncertainty levels, so experts assign basic weights on the detection algorithms. A basic weight indicates expert's confidence level on the detection algorithm in the assumed situation where only one detection algorithm was employed to detect a fault. Therefore, a basic weight is independent of other basic weights similarly derived from corresponding detection algorithms. These basic weights form a basic weight vector $\{\boldsymbol{W}_b | W_{1b}, W_{2b}, \ldots, W_{nb}\}$ of n detection algorithms.

One unique aspect of this basic weight vector $\boldsymbol{W}_b$ is that, admitting that this scheme is not perfect, the same basic weight is used to express our confidence on both classes of status: one's confidence of a fault when $X_f > 0$ and one's confidence of nonfault when $X_f = 0$. This seems illogical. It may be ideal to have the second weight which tells only of nonfault confidence. However, this kind of weight is not easily derivable when we consider the environment that, during overall running time of a certain detection algorithm, the indication of the detection algorithm will be nonfault most of the time. In other words, normal status is dominant; fault events do not frequently happen. When we map this situation into experiments, we find ourselves asking how many experiments are enough to get a reasonable weight on nonfault. Experts do not agree. This kind of information is simply not obtainable.

Using the same weight on both classes of status does not cause practical difficulty. When we have a certain basic weight, say 0.75, used on fault confidence, we work with this number to calculate only supportive (on fault) evidence. This means that we have 0.75 confidence of fault: this does not mean we have 0.25 confidence of nonfault. Similarly, if we use 0.75 on nonfault confidence, we say that we have 0.75 confidence of nonfault and use this number to calculate only nonsupportive (on fault) evidence: this does not mean 0.25 confidence of fault. They are quite independently used.

3.3 Adjusted weight vector

The reason we have 'adjusted weight' is the fact that when $X_f > 1$ during a decision time window, the weight set based on the original scheme of each detection algorithm with just one fault indication (i.e. $X_f = 1$) is not appropriate. We need to adjust our basic weights. The adjustment of basic weights is not easy. Literature is silent on this subject. So we decided to follow our intuition. Actually we want to have our detection system resemble human behaviour on this matter. It seems to work correctly when tested at our Downed Conductor Test Facility and applied in a substation computer which monitors feeder current.

With a certain value of basic weight, say 0.60, derived in an assumed conditions of $X_f = 1$ or $X_f = 0$, how will our confidence on any detection algorithm be changed when $X_f = X_m$? Experts say that, no matter what the basic weight is, they strongly believe that fault actually occurred about 0.99 or 1.0 confidence. However, when X_f is between 2 and X_m, the situation is so fuzzy that our experts are hesitant to give any exact number on their confidence. In a sense, this is a situation of measuring a fuzzy environment quantitatively.

For quantification of this fuzzy 'beliefs of experts', quite intuitively we could agree that those adjusted weights are on the approximate line of an exponential curve w increasing along the x-axis (the axis of actual number of fault indications, X_f). Therefore, we have the following requirements for a hypothetical curve for weight adjustments:

(i) a discrete curve w which has a value of basic weight at $x = 1$ and has an exact value, say 1.0 or 0.999, at $x = X_m$

(ii) a discrete curve w which resembles a exponential curve or parabola.

A true exponential or parabola is not an adequate candidate because it converges to a certain value (i.e. 1.0) at $x = \infty$. We found a better curve which has a form of exponent and maximum value at a fixed value of x: ellipse. By changing the eccentricity, while setting the minor axis to a certain value, say 0.999, and changing the length of the major axis, we could put the centre of it x equal to the maximum possible number of fault indications, X_m. Then we have the elliptic formula given below. We set 0.999 as our maximum confidence at $X_f = X_m$.

$$\frac{(x - X_m)^2}{a^2} + \frac{w^2}{0.999^2} = 1.0 \tag{5}$$

where a is half of a sum of distances from any point on the ellipse to the foci. The variable x indicates the actual number of fault indications, X_f.

When we apply the other condition that when $x = 1.0$ the curve w has a value of basic weight, we have the following weight adjustment formula. In the formula, w_0 indicates a basic weight. This relationship is well illustrated in Fig. 3.

$$w = \sqrt{\left[0.999^2 - \frac{(X_f - X_m)^2}{(1 - X_m)^2}(0.999^2 - w_0^2)\right]} \tag{6}$$

An example is shown for the weight adjustment. Assume that an algorithm–parameter has a basic weight of 0.60 and $X_m = 4$. Then the adjusted weights on each case of actual fault indication (X_f) is

X_f	$W_a(w)$
1	0.600
2	0.845
3	0.963
4	0.999

Conclusively speaking, the adjusted weight vector $\boldsymbol{W}_a(w)$ is made by both the basic weight vector $\boldsymbol{W}_b(w_0)$ and two elements of the input vector X (i.e. X_f and X_m). The adjusted weight is hereafter called just weight, $\boldsymbol{W}$.

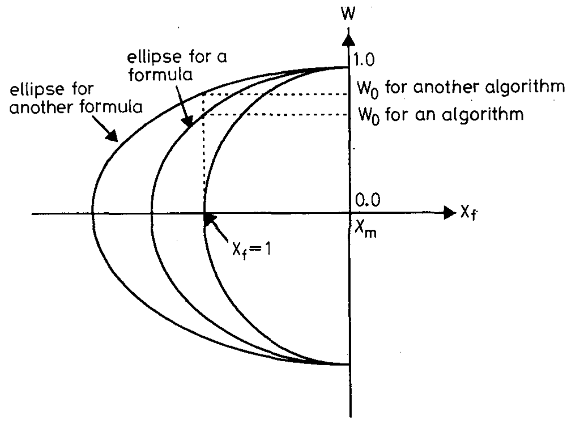

Fig. 3 *Illustration of ellipses for weight adjustment formula*

3.4 Weight combination

To calculate the combined evidence we need to find supportive evidence from fault indicating detection algorithms which meet the condition that $X_f > 0$, and nonsupportive evidence from all nonfault indicating detection algorithms which hold the condition of $X_f = 0$. The combinaton of these two pieces of evidence is done by a revised and easily computable version of the Dempster–Shafer theory [20].

A belief function assigns a measure of our total belief in the proportion represented by the subset of the frame of discernment. The frame of discernment is a set of propositions about exclusive and exhaustive possibilities in a domain, and is represented by the symbol Θ. In our problem of simple proportions of fault and nonfault as our Θ, we represent our belief in terms of degree of support like

$$Bel_1(\text{fault}) = S_1 \quad Bel_2(\text{fault}) = S_2$$

These are simple support functions with respective degrees of support S_1 and S_2. Then our overall supportive evidence can be expressed by $1 - (1 - S_1)(1 - S_2)$. Therefore, general supportive evidence of a fault $SE(f)$ and nonsupportive evidence of a fault $NE(f)$ can be expressed by the following formulas

$$SE(f) = 1 - \prod_{k=1}^{k=N_f}(1 - W_{ka}) \tag{7}$$

where W_{ka} is the adjusted weight of the kth algorithm and N_f indicates the total counts of the case where the actual number of fault indication is greater than or equal to 1, and

$$NE(f) = 1 - \prod_{j=1}^{j=N_n}(1 - W_{ja}) \tag{8}$$

where N_n indicates the total counts of the case where the actual number of fault indication is equal to zero. Then a combined evidence on fault $CE(f)$ and a combined evidence on nonfault $CE(n)$ can be expressed by the following formulas [14]

$$CE(f) = \frac{SE(f)(1 - NE(f))}{1 - SE(f) * NE(f)} \tag{9}$$

$$CE(n) = \frac{NE(f)(1 - SE(f))}{1 - SE(f) * NE(f)} \tag{10}$$

A very simple example is shown following this idea. If the following data is collected during a decision time window,

Algorithms	W_b	X_{im}	X_{if}
AP1	0.60	9	2
AP2	0.80	9	1
AP3	0.75	9	0
AP4	0.75	4	0
AP5	0.85	4	1
AP6	0.65	4	2

then according to the previous discussion on weight adjustments, supportive evidence, and nonsupportive evidence.

Algorithms	W_a	Supportive/nonsupportive
AP1	0.71	supportive
AP2	0.80	supportive
AP3	0.75	nonsupportive
AP4	0.75	nonsupportive
AP5	0.85	supportive
AP6	0.86	supportive

From this data, supportive evidence is

$$SE(f) = 1 - (1 - 0.71)(1 - 0.80)(1 - 0.85)(1 - 0.86) = 0.999$$

and nonsupportive evidence is

$$NE(f) = 1 - (1 - 0.75)(1 - 0.75) = 0.938$$

Then, a combined evidence of fault is

$$CE(f) = 0.0624/0.0636 = 0.981$$

This says our combined final evidence of fault is 0.981 or 98.1%. This very strongly indicates that there is a fault.

3.5 Weight calibration

The basic weights asigned to detection algorithms summarise the experts' personal, subjective, and experienced beliefs. Though experts have good experience and good sense of assigning weights on the detection algorithms, we need to assess their subjective beliefs. In other words, we want to see whether experts' weights are well calibrated.

However, unfortunately, not all well-calibrated weights give a useful indication. This is well shown in the example of the comparison of weathermen's forecast performance [18]. Suppose a certain weatherman always says there is a 50% chance of rain next day. Even when this belief comes out as a well-calibrated one, this forecast is next to useless: more like flipping a coin for a head or tail guess. Therefore, we need to calibrate basic weights so that detection algorithms are realistic about their abilities to identify the status. To measure a detection algorithm's ability given a basic weight, we adopted an idea of scoring rule. Suppose we reward each detection algorithm an amount of $U(r, w)$ when it has guessed on an event with a weight of w, and $r = 1.0$ if its guess is right, and $r = 0$, otherwise. This function $U(r, w)$ is called a scoring rule [18], and is defined by

$$U(r, w) = 1.0 - (r - w)^2 \qquad (11)$$

We calculate the score from the basic weights. With this score, we calibrate the basic weights backward. When there is a large enough number of events with actual status confirmed, we calculate the score of each detection algorithm. If the score calculated is very close to about 0.90, the highest calculated score with a chosen weight, we keep the original basic weight. Otherwise, a basic weight is changed to the value which will earn the highest calculated score.

For example, the following hypothetical experimental results demonstrate score calculation. The number of total experiments is $N = 20$. The algorithm pair we use is the AP1 with maximum indication number $X_m = 4$. And we assume its basic weight is $w_0 = 0.75$. Each experimental datum is classified according to the output X_f of this algorithm and the number of correct guesses recorded by this algorithm in each output X_f case. Each number of correct guesses by this algorithm corresponding to the value of X_f is shown below.

	Weight vector				
	Basic weight	Adjusted weight			
	w_0 0.75 $X_f = 0$	w_1 0.75 $X_f = 1$	w_2 0.94 $X_f = 2$	w_3 0.98 $X_f = 3$	w_4 0.99 $X_f = X_m$
Total case (NT_i), $i = [0, 4]$	6	8	4	1	1
Number of correct guess (NC_i), $i = [0, 4]$	4	3	1	0	0

Then the total score can be calculated using the score formula using the following practical equation. The maximum total score in this example, with given basic weights, is

$$U_{max} = 14 * (1 - (1 - 0.75)^2) + 4 * (1 - (1 - 0.94)^2) + (1 - (1 - 0.98)^2) + (1 - (1 - 0.99)^2) = 19.1101$$

The total score U is calculated with the following formula

$$U = \sum_{i=0}^{4} NC_i\{1 - (1 - w_i)^2\} + \sum_{j=0}^{4} (NT_j - NC_j)\{1 - (0 - w_j)^2\} \qquad (12)$$

In the right-hand part of the equation, the first term is the score for the correct guess and the second term for incorrect guess with this algorithm AP1. Then we have the score of $U = 11.2534$, about 59% of the U_{max}. Since this score is not satisfactory, we change the basic weight w_0 to give the highest possible score with the experiment. By changing the basic weight, and with corresponding w_i, $i = [2, 4]$ using the elliptic formula for adjustment, we calculated scores and compared them to find the highest score. The score was 12.585 and the corresponding basic weight w_0 was 0.45. With this new basic weight, the detection performance will be increased by 7%.

4 Example test of the detection system

We tested our detection system using staged fault test experiments and feeder monitoring experiences from the same site. The fault data was collected from staged fault tests done in January 1991 and the rest, normal data, was collected during January and February 1991 from the normal monitoring data which invoked algorithm–parameter pairs and caused at least one fault indication. Data number 1 through 5 belong to a downed conductor on a concrete surface and the rest of the fault data 6 through 20 are the case of the downed conductor on the sod surface. We used six algorithm–parameter pairs during this time. They are the combinations of two algorithms, energy algorithm [2] and randomness algorithm [3], and three parameters. The parameters we chose are a composite even harmonic, a composite odd harmonic, and a composite subharmonic (or called 'nonharmonic') in an interested frequency band of 0–480 Hz with the 60 Hz component eliminated.

Table 1 shows the result of the test. Fault status indicates the event of stages faults performed in our Downed Conductor Test Facility in the remote campus of Texas A&M University. No-fault status shows the data of nonfault which invokes at least one detection algorithms to falsely indicate a fault. We used six detection algorithms in this example. Each digit of the six-digit numbers in the second column indicates each detection algorithm's X_f.

Table 1: Test result of performance comparison between the original weight (W_b) and the calibrated weight (W_a)

No	Number of fault indication						$CE(f)$ with W_b	$CE(f)$ with W_a	Correct status
	AP1	AP2	AP3	AP4	AP5	AP6			
1	0	0	0	1	0	0	0.00	0.00	fault
2	0	0	0	1	0	0	0.00	0.00	fault
3	1	0	0	0	0	0	0.01	0.00	fault
4	0	0	0	1	0	0	0.00	0.00	fault
5	0	0	0	1	0	0	0.00	0.00	fault
6	1	0	0	1	0	0	0.05	0.02	fault
7	0	0	1	0	0	1	0.36	0.96	fault
8	0	0	1	0	0	1	0.36	0.96	fault
9	0	0	3	0	0	1	0.82	0.99	fault
10	0	0	3	0	0	1	0.82	0.99	fault
11	0	0	1	0	0	2	0.59	0.96	fault
12	0	0	2	0	0	1	0.53	0.98	fault
13	0	0	2	0	0	1	0.53	0.98	fault
14	3	0	0	2	0	0	0.79	0.91	fault
15	2	0	3	3	0	3	0.99	0.99	fault
16	2	0	3	3	0	3	0.99	0.99	fault
17	0	0	2	2	0	2	0.98	0.99	fault
18	0	0	1	2	0	1	0.92	0.99	fault
19	0	0	1	1	1	1	0.90	0.99	fault
20	1	0	1	3	0	1	0.99	0.99	fault
21	0	0	1	0	0	0	0.02	0.00	nonfault
22	0	0	0	1	0	0	0.02	0.00	nonfault
23	0	0	0	1	0	0	0.02	0.00	nonfault
24	0	0	0	1	0	0	0.02	0.00	nonfault
25	0	0	0	1	0	0	0.02	0.00	nonfault
26	0	0	0	1	0	0	0.02	0.00	nonfault
27	1	0	0	2	0	0	0.23	0.04	nonfault
28	0	0	0	1	0	0	0.02	0.00	nonfault
29	0	0	0	1	0	0	0.02	0.00	nonfault
30	0	0	0	0	2	0	0.01	0.00	nonfault
31	0	0	0	1	0	0	0.02	0.00	nonfault
32	0	0	0	1	0	0	0.02	0.00	nonfault
33	0	0	0	1	0	0	0.02	0.00	nonfault
34	0	0	0	1	0	0	0.02	0.00	nonfault
35	0	0	0	1	0	0	0.02	0.00	nonfault
36	0	0	0	0	0	1	0.02	0.00	nonfault
37	0	0	0	1	0	0	0.02	0.00	nonfault
38	0	0	0	1	0	0	0.02	0.00	nonfault
39	1	0	2	0	0	0	0.43	0.00	nonfault
40	0	0	1	0	0	0	0.02	0.00	nonfault

1–5 = concrete, 6–20 = sod

Each number in the second column indicates the number of fault indication (X_{if}) of APi, the ith algorithm–parameter pair, where $i = [1, 6]$. The corresponding pair of the algorithm–parameter is

$i = 1$, AP1: randomness algorithm–even harmonics

$i = 2$, AP2: randomness algorithm–odd harmonics

$i = 3$, AP3: randomness algorithm–sub harmonics

$i = 4$, AP4: energy algorithm–even harmonics

$i = 5$, AP5: energy algorithm–odd harmonics

$i = 6$, AP6: energy algorithm–sub harmonics

For the maximum indication number X_m, the energy algorithm has a maximum possible indication of 4 (i.e. $X_{im} = 4$, $i = [4, 6]$) and the randomness algorithm, 9 (i.e. $X_{im} = 9$, $i = [1, 3]$), during a chosen decision time window of 30 seconds. In this example, the original basic weight vector of $\{W_b \mid 0.7, 0.6, 0.8, 0.5, 0.5, 0.75\}$ is used. To test the performance, we used our internal threshold level for fault indication as 0.90 or above the combined evidence. In other words, if the combined evidence is 0.90 or above, the detection systems indicates the fault status as its final decision.

From the second column of Table 1, the range of the rate for each individual algorithm is 5–65% for the correct indication rate, and 5–70% for the false indication. (We excluded the rates of AP2, because the AP2 is of 0% correct and false indication rates. Actually this pair is of no use). The third column shows the combined evidence with the original weights. The correct indication of fault is only six out of 20 (30% correct indication rate) and the false indication is zero out of 20 (0% false indication rate). Conclusively, the detection with the original weights is of low dependability and high security. Even though six cases of faults have the same pattern of normal case, and the first five faults data are from the concrete surface (the difficulties of the detection of the downed conductor on the surface of concrete or asphalt are well known: see Reference 10, we have to improve our correct indication of faults.

Then with a total of 40 events of experience, we calculate a score and find a new set of calibrated basic weights: $\{W_b \mid 0.34, 0.10, 0.98, 0.98, 0.16, 0.88\}$. The same procedure was performed with this set of calibrated weights. The result is shown in the fourth column. With the calibrated weights, the correct indication moves up to 70%, while keeping the false indication rate 0%. This is quite a dramatic increase in the performance without jeopardising the 0% rate of false indication of fault.

With the set of calibrated basic weights in this example, our adaptive detection system may not guarantee good performance at other sites, because the training data and the testing data we used are from the same site (see Reference 10 for the dependency of the detection algorithms on the different sites). However, this example sufficiently shows the detection system's adaptability to other situations.

The calibration scheme can be applied in two stages. In the first stage, we tune the detection system with training data before we install it in the field. This will provide a better start for a higher score with calibrated basic weights. These calibrated weights are most appropriate whether they are coincidental or contradictory to those of the experts. In the other stage, while in the operation process, we calibrate basic weights if the performance is unacceptable. The information we use for calibration in the second stage is the detection system's outputs and the operator's responses. The operator's response can be drawn only when the detection system's output is fault and when a fault actually occurred, therefore, this system will gradually improve security and dependability in a supervised adaptive manner.

5 Conclusions

We have discussed a perplexing situation in the detection of high impedance faults. To overcome the problem, we used multiple algorithms with several different electrical parameters. To accommodate these multiple different beliefs of the same event, we devised an adaptive detection system using our revised adaptive element model. The detection system derives supportive evidence and nonsupportive evidence and combines them for a final evidence about a distribution feeder status. An uncertainty reasoning method is adopted for evidence combination and a performance scoring rule was employed to correct the weights as a learning algorithm. The adaptive detection system evolved, securing higher performance, into new situations with calibrated weights. We expect this overall system to give flexibility, dependability, and security in the classification and detection of high impedance faults.

6 References

1 BEASLEY, W.L.: 'An investigation of radial signals produced by small sparks on power lines'. PhD dissertation, Texas A&M University, 1970

2 'Detection of arcing faults on distribution feeders'. EPRI Final Report (prepared by Texas A&M University), *EPRI EL-2757*, 1982

3 BENNER, C., CARSWELL, P., and RUSSELL, B.D.: 'Improved algorithm for detecting arcing faults using random fault behavior', *Electr. Power Syst. Res.*, 1989, **17**, pp. 49–56

4 'High impedance fault detection using third harmonic current'. EPRI Final Report (prepared by Hughes Aircraft Co.), *EPRI EL-2430*, 1982

5 'Detection of high impedance faults'. EPRI Final Report (prepared by Power Technologies, Inc.), *EPRI EL-2413*, 1982

6 KIM, C.J., and RUSSELL, B.D.: 'Harmonic behavior during acing faults on power distribution feeders', *Elec. Power Syst. Res.*, 1988, **14**, pp. 219–225

7 KIM, C.J.: 'An intelligent decision making system for detecting high impedance faults'. PhD dissertation, Texas A&M University, 1989

8 'Detection of downed conductors on utility distribution systems'. IEEE Tutorial Course Text, 90EH0310-3-PWR, 1989

9 RUSSELL, B.D., MEHTA, K., and CHINCHALI, R.P.: 'An arcing fault detection technique using a low frequency current components-performance evaluation using recorded field data', *IEEE Trans.*, 1988, **PWRD-3**, pp. 1485–1500

10 KIM, C.J., and RUSSELL, B.D.: 'Classification of faults and switching events by inductive reasoning and expert system methodology', *IEEE Trans.*, 1989, **PWRD-4**, pp. 1631–1637

11 JEERINGS, D.I., and LINDERS, R.J.: 'Ground resistance — revisited', *IEEE Trans.*, 1989, **PWRD-4**, pp. 949–956

12 BALSER, S.J., CLEMENTS, K.A., and LAWRENCE, D.J.: 'A microprocess-based technique for detection of high impedance faults', *IEEE Trans.*, 1986, **PWRD-1**, pp. 252–258

13 KIM, C.J., RUSSELL, B.D., and WATSON, K.L.: 'A parameter-based process for selecting high impedance fault detection techniques using decision making under incomplete knowledge', *IEEE Trans.*, 1990, **PWRD-5**, pp. 1314–1320

14 GORDON, G., and SHORTLIFF, S.H.: 'The Dempster-Shafer theory of evidence', *in* BUCHANAN, B.G., and SHORTLIFF, E.H. (Eds.): 'Rule-based expert systems' (Addison-Wesley, Reading, MA, USA, 1984)

15 CAUDILL, M., and BUTLER, C.: 'Naturally intelligent systems' (MIT Press, Cambridge, MA, USA, 1990)

16 WIDROW, B., and LEHR, M.: '30 Years of adaptive neural networks: perceptron, Madaline, and backpropagation', *Proc. IEEE*, 1990, **78**, pp. 1415–1442

17 WIDROW, B., and WINTER, R.: 'Neural nets for adaptive filtering and adaptive pattern recognition', *Computer*, 1988, **21**, pp. 25–39

18 SMITH, J.Q.: 'Decision analysis — a Bayesian approach' (Chapman and Hill, New York, 1988)

19 RUSSELL, B.D., CHINCHALI, R.P., and KIM, C.J.: 'Behavior of low frequency spectra during arcing fault and switching events', *IEEE Trans.*, 1988, **PWRD-3**, (4), pp. 1485–1492

20 BARNETT, J.A.: 'Computational methods for a mathematical theory of evidence'. Proceedings of the seventh international joint conference on *Artificial intelligence*, 1981, vol. 81, pp. 868–875

Errata

MUNSHI, S., ROY, C.K., and BISWAS, J.R.: 'Computer studies of the performance of transformer windings against chopped impulse voltages', *IEE Proc.-C*, 1992, **139**, (3), pp. 286–294

The authors wish to point out the following errors in their paper:

In Section 2, eqn. 2 should read as follows:

$$h_2(t) = -\bar{A}_2[e^{-p_3(t-T_c)} - e^{-p_4(t-T_c)}] \quad \text{for } t \geqslant T_c$$
$$= 0 \quad \text{for } t < T_c \qquad (2)$$

In Section 3.4, line eight should read as follows:

$$\bar{\lambda} = \lambda l = 4.68$$

In Section 5, part (vi), line four should read as follows:

. . . the IEC recommendation [6], irrespective of . . .

In Appendix 7.1, line eight should read as follows:

$$h_2(t) = \bar{A}_2[e^{-p_3(t-T_c)} - e^{-p_4(t-T_c)}] \quad \text{for } t \geqslant T_c$$
$$= 0 \quad \text{for } t < T_c$$