Application of Genetic Algorithm and Rough Set Theory for Knowledge Extraction

Chinglai Hor, *Member, IEEE,* Peter A. Crossley, *Member, IEEE,* and Dean L. Millar

***Abstract*— This paper proposes a hybrid approach using the rough set theory and genetic algorithm (RS-GA) for knowledge extraction as one part of a substation level decision support system. The technique involved a process which learns and extracts knowledge from a set of events into a form of rules to identify the most probable faulted section in a network. Numerous case studies performed on a simulated distribution network [1] that consists of several relays models [2] using PSCAD/EMTDC have revealed the usefulness of the proposed technique for fault diagnosis. The test results demonstrated that the extracted rules are capable of identifying and isolating the faulted section and hence improve the outage response time.**

***Index Terms*— rule induction, fault diagnosis, knowledge base, expert system, genetic algorithm, rough sets, reduct computation.**

I. Introduction

WITH the advent of AI, rule based expert systems are able to offer capabilities of powerful inference and explanation for symbols recognition and knowledge intensive problem of fault diagnosis. However, they suffer from some bottlenecks. The size of a conventional knowledge base in a substation can be very large depending on the task it is in use. Therefore, the process of knowledge acquisition, knowledge base construction and maintenance for a great number of rules can be quite tedious and often take a long time. Moreover, such process often involves an intensive interview with experts and engineers and thus significant amount of efforts are required to establish a rule based system with a substantial good performance. As such, the cost of developing and/or maintaining a knowledge base system is generally high.

We need an approach that is capable of extracting good quality of knowledge in the form of rules from different sets of events data. The rule induction system should be an automated process with subsequent classification used for verifying the extracted rules without the presence of an expert [3]. Clearly, the experts are still required to perform a final verification check before these rules are applied to the knowledge base system. The proposed technique can reduce the time needed to install a knowledge based system whenever a new substation configuration has taken place and hence reduce the maintenance cost.

Manuscript received April 11, 2007.

C.L. Hor and D.L.Millar are currently with Camborne School of Mines, School of Geography Archaeology & Earth Resources, University of Exeter in Cornwall, Tremough Campus, Penryn, Cornwall, TR10 9EZ, United Kingdom (e-mail: c.l.hor@exeter.ac.uk and d.l.millar@exeter.ac.uk).

P.A. Crossley is currently with the School of Electrical and Electronic Engineering, Ferranti Building, The University of Manchester. PO Box 88, Manchester M60 1QD, United Kingdom (e-mail: p.crossley@manchester.ac.uk).

II. Rough Sets (RS)

Rough set theory (RST) is a technique for knowledge discovery that has been successfully applied in data mining. The concept of rough sets is based on the idea of using upper and lower approximation of a set to deal with discernibility. These two approximations provide crisp and rough descriptions of a dataset. If a universe can be formed as a union of some elementary sets, it is called crisp otherwise, it is rough. A lower approximation defines the collection of events in which the equivalence classes are fully contained in the set of events which is to be reduced to its essential attributes. The upper approximation however defines the collection of events in which the equivalence classes are at least partially contained in the set of events to be reduced. The literatures of rough sets are available in [4], [5] which proposed the use of a supervised and unsupervised rough classification for handling large numbers of messages received during an emergency.

In this paper, we are however more concerned about knowledge base construction for online intelligent switching, fault identification and service restoration. Detailed rules covering every possible scenario in a substation are thus needed. A classical information system may not cope well with high volumes of data and often it relies on particular data sources. If these data sources are not available, such a system will not perform accurately. Rules generated from a decision table are based on the reduct set computation. A reduct is a reduced set of attributes, e.g voltages and currents, that essentially provides the same amount of information about a data set as a complete set of attributes [3]. Computing the reduct set can be very time consuming particularly when the decision table has too many attributes or different attribute values. This is because some problems do not always lend themselves to solutions by deterministic algorithms. These are generally categorised as an NP (*non-polynomial*) problem. In case of the complexity class of decision problems that are intrinsically hard to solve, they are referred to as an NP-hard [6]. An effective genetic algorithm is proposed in this paper to optimise the number of reducts computed by the rough set approach. The approach is complementary to our proposed rough classification method for knowledge base reduction and rule induction [3].

III. Hitting Sets

Finding reducts for data reduction is commonly associated with minimal hitting sets. Given a collection of sets S, a hitting set is defined as a set B⊆A such that the intersection between B and every set in S is non-empty (i.e. contains at least one element from all the sets in S). Let HS(S) denotes the collection of all the hitting sets of S [7] as in Eq. (1).

$$HS\,(S) = \{B \subseteq A \mid B\ \cap S_i \neq \emptyset\ for\ all\ S_i\ in\ S\} \quad (1)$$

If no element can be removed from S without violating the hitting set property, it is considered as the minimal hitting set denoted MHS(S) [7]. Determining a minimal cardinality element of MHS(S) is called the minimal hitting set problem.

Finding the minimal hitting sets is sometimes NP-hard. Its NP-hard complexity makes the computation of an optimal solution infeasible. There are a number of hitting sets but only one or some of them are needed [9]. In case of large sets of conflicts, the computation of the hitting sets will result in both time and space consumption. The discernibility matrix is needed in order to make the application useful for rough set purpose [10]. To achieve a minimal reduct finding in an information system, a discernibility function is interpreted as a multiset $S\,(h)$. A multiset is conceptually an unordered collection of elements where the same elements may occur more than once. The multiset constructor S is a trivial matter of reinterpretation as $S\left(\left(\bar{a}+\bar{b}\right)\cdot\left(\bar{a}+\bar{b}\right)\cdot(\bar{c})\right) = [\{a,b\},\{a,b\},\{c\}]$ and *h* represents any Boolean product of sum function of *m* Boolean variables $\left\{\bar{b}_1,\bar{b}_2,......,\bar{b}_m\right\}$ composed on *n* sums $\{s_1,.......,s_n\}$ as defined in Eq. (2) [8]:

$$S(h) = \left\{S_i \mid S_i = \left\{b_j \in B \mid \bar{b}_j\ occurs\ in\ s_i\right\}\right\} \quad (2)$$

where $i = \{1,...,n\}$, $j = \{1,...,m\}$ and

$$s_i = \sum_{j=1}^{m} \bar{b}_j \bar{w}_{ij} \quad (3)$$

where $\bar{w}_{ij} \in \{0,1\}$ denotes an indicator variable that states whether $\bar{b}_j$ occurs in s_i, $b_j\,(x_i) = \begin{cases} 0 & if\ i\ =\ 0 \\ \bar{w}_{ij} & otherwise. \end{cases}$ and

$$h = \prod_{i=1}^{n} s_i \quad (4)$$

A hitting set of $S\,(h)$ is an implicant of *h* constructed on the basis of a discernibility matrix, M_B. Consequently, a minimal hitting set corresponds to a prime implicant (equivalently to a reduct) [8]. It is often desirable to compute approximations of minimal hitting sets, i.e. attribute subsets that hit *enough* but not necessarily all of the sets in S. Thus, a hitfraction is used.

A hitfraction (*hf*) is defined as the measure of approximation, which has a ratio of the sum of weights of the sets in S intersected by B to the total sum of weights of elements in S [7] [8]. For each S_i in set *S* computed from the data, we can associate it with a weight $w(S)$ as given in Eq. (5): -

$$hf\,(B,S) = \frac{\sum\limits_{S_i \in S} w\,(HS\,(S))}{\sum\limits_{S_i \in S} w\,(S)} \quad (5)$$

If a minimal hitting set has a hitfraction = 1, this implies a proper minimal set. Those good enough hitting sets, i.e. have a hitting fraction of at least ε, which signifies a minimal value for the hitting fraction are known as approximate hitting sets.

Approximate hitting set (AHS) is a set that hits "enough" elements of the bag or multiset S. It provides an approximate solution to the hitting set problem. The set of ε-approximate hitting sets of the multiset S is defined AHS(S,ε): -

$$AHS\,(S,\varepsilon) = \left\{B \subseteq A| \frac{|\{S_i \text{ in } S|S_i \cap B \neq \emptyset\}|}{|S|} \geq \varepsilon\right\} \quad (6)$$

where the parameter ε and k control the degree of approximation decision ability. The set is a minimal ε-approximate hitting set if it ceases to be so when any of its elements are removed. The set of all minimal ε-approximate hitting set denoted MAHS(S,ε) is collected in a *keep list* with the size that can be specified [8]. *k* denotes the number of extra keep lists in use by the algorithm. $w(S) = k$ for a sum that occurs *k* times in *S*. If *k* = 0, then only the minimal hitting sets with a hitting fraction of approximately ε are returned. If $k > 0$, then $k + 1$ groups of the minimal hitting sets are returned, each group having an approximate (but not smaller) hitting fraction is evenly spaced between ε and 1 in which ε = 1 implies proper minimal hitting sets [8].

IV. Genetic Algorithm

A genetic algorithm (GA) is a stochastic global search algorithm that reflects in a primitive way some of the processes of natural biological evolution [11]. It has been successfully used in reduct finding in rough set theory as it works very well on combinatorial problems [12]. A GA starts by generating a large set of possible solutions to a given problem. It then evaluates each of those solutions, and decides on a fitness level for each solution set. These solutions then breed new solutions. The parent solutions that have better fitness level are more likely to reproduce, while those that have less fitness level are more unlikely to do so. GAs evolve the search space scope over time or generation to a point where the solution can be found. The reduct computation process using GAs begins with initialising the population of chromosomes (or individuals) from the discernibility functions *f*(B) via rough sets. This can be represented in the following steps: -

1) **Generate an initial population**. An initial population is created from a random selection of solutions. Bi-vectors are used to represent the sets and their hitting sets. These bi-vectors are called *chromosome*, and each bit is called *gene*, and all of the chromosomes are called *population*. Each of the chromosomes are assigned a weight to record the number of times it appears in *f*(B) before the fitness of all the chromosomes are evaluated. A straightforward choice of population is a set P of elements from the power set of A, written P(A) or 2^A encoded as bit-vectors where each bit indicates the presence of a particular element in the set. Given a set A, 2^A is the set of all subsets of A. As 2 can be defined as $\{0,1\}$, 2^A is the set of all functions from A to $\{0,1\}$. Assume that we have 10 IED relays in our network $\{\text{IED1}, \text{IED2}, \ldots, \text{IED10}\}$ and a reduct candidate A = $\{\text{IED1}, \text{IED4}, \text{IED9}\}$. Then, 2^A is $\{\}, \{\text{IED1}\}, \{\text{IED4}\}, \{\text{IED9}\}, \{\text{IED1}, \text{IED4}\},$

$\{\mathrm{IED1}, \mathrm{IED9}\}, \{\mathrm{IED4}, \mathrm{IED9}\}, \{\mathrm{IED1}, \mathrm{IED4}, \mathrm{IED9}\}$. The reduct candidate can be represented as 1001000010.

2) **Evaluate fitness**. Fitness is a value assigned to each solution (chromosome) depending on how close it actually is to solving the problem. These solutions are possible characteristics that the system would employ to reach the answer. The fitness function depends on the number of attributes (should be kept as low as possible) and decision ability (should be kept as high as possible).
3) **Reproduce, selection, mutation, inversion and crossover**. A higher fitness value of those chromosomes are more likely to reproduce offspring. In the reproduction, crossover, mutation and inversion are performed to create offsprings. The mutation operator chooses a random location in the bit string and changes that particular bit. The inversion operator inverts the order of the elements between two randomly chosen points on a single chromosome. It rearranges the bits in a string allowing linked bits to migrate and move close together by keeping track of the original positions of the genes. The crossover operator recombines two individuals by exchanging parts of their strings to form two new individuals (offspring).
4) **Control next generation**. If the new generation contains a solution that produces an output that is close enough or equal to the desired answer then the problem has been solved. Otherwise, the process repeats until the predetermined number of generations have been exceeded or an acceptable performance level has been reached.

The fitness function $f(B)$ evaluates an individual candidate solution based on how well it performs at solving the problem. It drives the search in GAs by optimizing the solution that determines how good a solution is. The better a solution performs, the higher rated fitness value it receives. The fitness function in use by the algorithm is defined in Eq. (7), where B$\subseteq$ A and S is the set of sets corresponding to the discernibility function. The function cost specifies the cost of an attribute subset. If no cost information is used, then a default unit cost is effectively defined as $cost(B) = |B|$ [8].

$$f(B) = (1-\rho) \times \frac{\cos t(A) - \cos t(B)}{\cos t(A)} + \rho \times hf(B,S) \quad (7)$$

$$\text{where :} \quad hf(B,S) = \min\left\{\varepsilon, \frac{|[S_i \text{ in } S | S_i \cap B \neq \emptyset]|}{|S|}\right\}$$

The subsets B of A that are found through the evolutionary search driven by the fitness function. The parameter ρ defines a weighting between subset cost and hitfraction (*hf*). The first term of Eq. (7) rewards the shorter elements and the second term ensures that hitting sets are rewarded to guarantee the decision ability.

V. Example I

Figure 1 shows a typical 132/11kV substation model developed using PSCAD/EMTDC [3]. The directional relays at R5 and R6 also include non-directional time graded earth fault elements to protect the 11kV busbar and provide backup for the 11kV feeders [1]. Relays R1, R2, R3 and R4 all gave an identical pattern for the fault F1 and therefore these relays are regarded as one and labelled as "Rx" in which x = $\{1, 2, 3, 4\}$. Vx and Ix represent the three phase voltage and current. H1 indicates the current is flowing in the direction that would trigger one of the directional relays, i.e. R6 and R5. To assist our fault section estimation, we defined 4 protection zones in Figure 1. Breakers BRK5 and BRK7 are regarded one breaker zone labelled as "BZ2". Similarly, BRK6 and BRK8 are regarded one labelled as "BZ3". Zone 1 represents the protection zones of R1, R2, R3 and R4. Zone 2 includes the protection zones of R5, R7, R9 and R11 and Zone 3 covers the protection zones of R6, R8, R10 and R12. Zone 4 is the busbar protection zone which is not considered in this scenario. Zone 25 indicates the regional Zone 2 supervised by the relay R5.

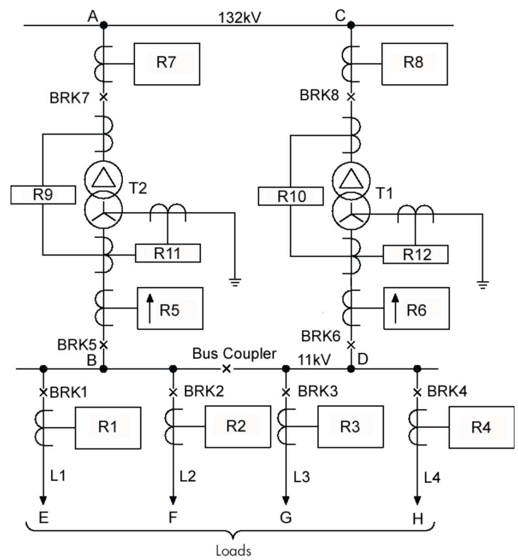

Fig. 1. 132/11kV Substation model

Table I and Table II laid out a list of recorded voltage and current patterns as well as the switching actions caused by the protection system(s) subject to various faults at different locations in the substation. Bx is the short form of the breaker BRKx in which x = $\{1, 2, 3, 4\}$. The auxiliary contacts are used to determine the condition of a breaker and relay. '01' indicates that the contact of the breaker/relay is closed. '10' indicates that the breaker/relay is open/tripped, '00' indicates failure of the breaker/relay and '11' indicates an undefined breaker/relay state.

Table III and IV listed the decision rules generated by the hybrid RS-GA model. The former considers a relay box that contains a voltage and current measurement while the latter considers the two measurements separately. Combining the relay trip and breaker data from Table II, we can analyse the rules produced. In Table III, the extracted rules 5, 8, 21 and 28 are not that useful because $\mathrm{I}_9, \mathrm{I}_{10}, \mathrm{I}_{11}$ and I_{12} are from the unit protections which are only important if there is a fault on the transformer. Rule 11, 18, 27 and 35 are not good enough to be considered. This is likely to happen when the data set does not contain adequate information for knowledge

TABLE I

LIST OF VOLTAGE AND CURRENT PATTERNS WITH ESTIMATED PROTECTION ZONES FOR VARIOUS FAULT SCENARIOS

R1		R2		R3		R4		R5		R6		R7		R8		R9	R10	R11	R12	ZONE
V_1	I_1	V_2	I_2	V_3	I_3	V_4	I_4	V_5	I_5	V_6	I_6	V_7	I_7	V_8	I_8	I_9	I_{10}	I_{11}	I_{12}	
L	H	L	L	L	L	L	L	L	H	L	H	N	H	N	H	L	L	L	L	Z11
L	L	L	L	L	L	L	L	L	H1	L	H	N	H	N	H	L	L	L	L	Z25
L	L	L	L	L	L	L	L	L	H	L	H1	N	H	N	H	L	L	L	L	Z36
L	L	L	L	L	L	L	L	L	L	L	L	L	H	L	L	L	L	L	L	Z27
L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	H	L	L	L	L	Z38
L	L	L	L	L	L	L	L	L	H1	L	H	L	H	L	H	H	L	L	L	Z2T
L	L	L	L	L	L	L	L	L	H1	L	H	L	H	L	H	L	L	H	L	Z2T
L	L	L	L	L	L	L	L	L	H1	L	H	L	H	L	H	H	L	H	L	Z2T
L	L	L	L	L	L	L	L	L	H	L	H1	L	H	L	H	L	H	L	L	Z3T
L	L	L	L	L	L	L	L	L	H	L	H1	L	H	L	H	L	L	L	H	Z3T
L	L	L	L	L	L	L	L	L	H	L	H1	L	H	L	H	L	H	L	H	Z3T

TABLE II

LIST OF SWITCHING ACTIONS WITH ESTIMATED PROTECTION ZONES FOR VARIOUS FAULT SCENARIOS

R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	B1	B2	B3	B4	BZ2	BZ3	ZONE
10	01	01	01	01	01	01	01	01	01	01	01	10	01	01	01	01	01	Z11
01	10	01	01	01	01	01	01	01	01	01	01	01	10	01	01	01	01	Z12
01	01	10	01	01	01	01	01	01	01	01	01	01	01	10	01	01	01	Z13
01	01	01	10	01	01	01	01	01	01	01	01	01	01	01	10	01	01	Z14
01	01	01	01	10	01	01	01	01	01	01	01	01	01	01	01	10	01	Z25
01	01	01	01	01	10	01	01	01	01	01	01	01	01	01	01	01	10	Z36
01	01	01	01	01	01	10	01	01	01	01	01	01	01	01	01	10	01	Z27
01	01	01	01	01	01	01	10	01	01	01	01	01	01	01	01	01	10	Z38
01	01	01	01	01	01	01	01	10	01	10	01	01	01	01	01	10	01	Z2T
01	01	01	01	01	01	01	01	10	01	01	01	01	01	01	01	10	01	Z2T
01	01	01	01	01	01	01	01	01	01	10	01	01	01	01	01	10	01	Z2T
01	01	01	01	01	01	01	01	01	10	01	10	01	01	01	01	01	10	Z3T
01	01	01	01	01	01	01	01	01	10	01	01	01	01	01	01	01	10	Z3T
01	01	01	01	01	01	01	01	01	01	01	10	01	01	01	01	01	10	Z3T

extraction. These rules should thus be verified and refined by experts to improve their coverage. Some rules generated are also redundant due to similar patterns in V_5 and V_6 as well as V_7 and V_8 and they are denoted as 'R'. The rules may look predictable for a small substation in Figure 1. However, when considering a larger substation or a complex power network with a significant number of protection system(s), extracting rules manually may be time-consuming and require significant resource. As such, this hybrid method will be useful to power utilities for exploiting substation rules. Relying on the switching actions for estimating fault section might not always be adequate when considering relay failures and the complexity of a power network. Therefore, we believe that voltage and current components should also be considered in a fault section estimation procedure.

VI. EXAMPLE II

Unlike the data given in Table I and II, the real time operational data captured in a substation is usually in time series. Thus it is necessary to exract knowledge from this type of dataset. This example is to evaluate the data pattern when the breaker BRK1 has failed to operate in Figure 1. Due to its large size, the time series dataset is not presented in the paper. A single phase A to earth fault was applied at 1.0s to the load feeder L1. The failure of breaker BRK1 at 1.091s has caused the upstream relays IED5 and IED6 to trip at 1.339s. Both IED relays, which serves as a backup to the downstream protection saw the fault at the busbar. Both tripped the breakers BRK5 and BRK6 and inter-tripped the respective breakers BRK7 and BRK8 simultaneously. Table VI showed a list of decision rules that imply IED1 and either breakers BRK5 - BRK7 or BRK6 - BRK8 are the main information source for this events. However, there is a concern if the given rules are adequate to classify the events correctly. Without the data from fault indicator, it is difficult to confirm that the breaker BRK1 has failed. The fault indicator data is a crucial piece of information that can provide reliable information when any protection system failed and triggered other protection systems to respond. The trouble is that if the fault indicator data is considered in the rule generation process, more rules are obtained and also it may be difficult to fit into the structure of the state classification, which only takes account of the voltage, current and breakers data [3]. To overcome this, the fault indicator is considered separately from the rules generation process, meaning that if the inputs match with the rules listed in Table VI, the system will acquire the relay trip data to confirm its status before the decision for the breaker failure BRK1 is fired. Due to the space constraints in this paper, the entire rule sets that characterise the overall events from the dataset are not included. The rules presented in Table VI (ignore the time column) are derived only from the first reduct set in Table V. The data pattern of V_1 = V_5 = V_6, V_2 = V_3 = V_4 and breakers BRK5 = BRK6 = BRK7 = BRK8. Therefore, some rules listed in Table V are carrying the same information. We thus picked the shorter reduct set which is $\{I_1, V_1, BRK5 - BRK7\}$. Each reduct has also an associated support count that measures the *strength* of the reduct equivalent to the reduct's hitting fraction multiplied by 100, which is the percentage of sets in sum S that attribute B has a non-empty intersection with [8].

A. Classifier performance

The set of rules derived from the reducts must be assessed on its classification performance, readability and usefulness before it can be verified by experts and used for online diagnosis. For assessing the classifier performance, the dataset

TABLE III

RULES GENERATED BY RS-GA (RELAY BOX)

Index	List of rules to identify the faulted section	Zone	Yes
1	$\{V_1(L), I_1(H)\}$	Z11	$\surd$
2	$\{V_5(L), I_5(H)\} \cdot \{V_6(L), I_6(H)\}$	Z11	$\surd$
3	$\{V_5(L), I_5(H1)\} \cdot \{V_7(N), I_7(H)\}$	Z25	$\surd$
4	$\{V_5(L), I_5(H1)\} \cdot \{V_8(N), I_8(H)\}$	Z25	R
5	$\{V_5(L), I_5(H1)\} \cdot I_9(L) \cdot I_{11}(L)$	Z25	$\times$
6	$\{V_1(L), I_1(L)\} \cdot \{V_6(L), I_6(H)\} \cdot \{V_7(N), I_7(H)\}$	Z25	$\surd$
7	$\{V_1(L), I_1(L)\} \cdot \{V_6(L), I_6(H)\} \cdot \{V_8(N), I_8(H)\}$	Z25	R
8	$\{V_1(L), I_1(L)\} \cdot \{V_6(L), I_6(H)\} \cdot I_9(L) \cdot I_{11}(L)$	Z25	$\times$
9	$\{V_5(L), I_5(L)\} \cdot \{V_7(L), I_7(H)\}$	Z27	$\surd$
10	$\{V_6(L), I_6(L)\} \cdot \{V_7(L), I_7(H)\}$	Z27	R
11	$\{V_8(L), I_8(L)\}$	Z27	$\times$
12	$\{V_5(L), I_5(H1)\} \cdot \{V_7(L), I_7(H)\}$	Z2T	$\surd$
13	$\{V_5(L), I_5(H1)\} \cdot \{V_8(L), I_8(H)\}$	Z2T	R
14	$\{V_6(L), I_6(H)\} \cdot \{V_7(L), I_7(H)\}$	Z2T	$\surd$
15	$\{V_6(L), I_6(H)\} \cdot \{V_8(L), I_8(H)\}$	Z2T	R
16	$I_9(H)$	Z2T	$\surd$
17	$I_{11}(H)$	Z2T	$\surd$
18	$\{V_7(L), I7(H)\} \cdot \{V_8(L), I_8(H)\} \cdot I_{10}(L) \cdot I_{12}(L)$	Z2T	$\times$
19	$\{V_6(L), I_6(H1)\} \cdot \{V_8(N), I_8(H)\}$	Z36	$\surd$
20	$\{V_6(L), I_6(H1)\} \cdot \{V_7(N), I_7(H)\}$	Z36	R
21	$\{V_6(L), I_6(H1)\} \cdot I_{10}(L) \cdot I_{12}(L)$	Z36	$\times$
22	$\{V_1(L), I_1(L)\} \cdot \{V_5(L), I_5(H)\} \cdot \{V8(N), I8(H)\}$	Z36	$\surd$
23	$\{V_1(L), I_1(L)\} \cdot \{V_5(L), I_5(H)\} \cdot \{V7(N), I7(H)\}$	Z36	R
24	$\{V_1(L), I_1(L)\} \cdot \{V_5(L), I_5(H)\} \cdot I_{10}(L) \cdot I_{12}(H)$	Z36	$\surd$
25	$\{V_6(L), I_6(L)\} \cdot \{V_8(L), I_8(H)\}$	Z38	$\surd$
26	$\{V_5(L), I_5(L)\} \cdot \{V_8(L), I_8(H)\}$	Z38	R
27	$\{V_7(L), I_7(L)\}$	Z38	$\times$
28	$\{V_8(L), I_8(H)\} \cdot I_9(L) \cdot I_{10}(L) \cdot I_{11}(L) \cdot I_{12}(L)$	Z38	$\times$
29	$\{V_5(L), I_5(H)\} \cdot \{V_8(L), I_8(H)\}$	Z3T	$\surd$
30	$\{V_5(L), I_5(H)\} \cdot \{V_7(L), I_7(H)\}$	Z3T	R
31	$\{V_6(L), I_6(H1)\} \cdot \{V_8(L), I_8(H)\}$	Z3T	$\surd$
32	$\{V_6(L), I_6(H1)\} \cdot \{V_7(L), I_7(H)\}$	Z3T	R
33	$I_{10}(H)$	Z3T	$\surd$
34	$I_{12}(H)$	Z3T	$\surd$
35	$\{V_7(L), I_7(H)\} \cdot \{V_8(L), I_8(H)\} \cdot I_9(L) \cdot I_{11}(L)$	Z3T	$\times$

is divided into a training set and a test set. Since there is no standard on how to choose the relative sizes of these sets, 10%, 30% and 50% for the test sets have been randomly selected and the average result was determined. The average accuracy of the classifier for the 10% test data is remarkably good, **98.4%** (see Table VII). The results demonstrated that most of the rules generated from the training sets are able to classify the events in the test set. The inaccuracy occurred when the rules generated from the training set 3 were used to classify the test set 3. Only one event with the decision class E_1 was unclassified and the classifier was not able to distinguish it from the events with the decision class E_0. The cause of this problem could be due to the small dataset used in this case study (only 35 events). This leads to the classification errors since the 10% we used for testing may be different than the 90% we used for training.

The same procedure is repeated again, this time with different training sets and split of data. For the training set that consists 70% of the data and 10% for the test set, the accuracy of classification is approximately **98.9%** (see Table VIII). In the final assessment, the accuracy of classification dropped expectedly to approximately **91.2%** (see Table IX) when the smaller variety of training set is used (only 50% of the data). The overall average accuracy for all the three experiments is (0.984 + 0.989 + 0.912)/3 = 0.9617.

The accuracy of 30% and 50% are expected to drop because the system has been trained on a smaller variety of events and therefore it may not recognise a larger variety of events for classifications. The small rise in accuracy using the 70% of data as a training set compare to the 90% partition might be

TABLE IV

RULES GENERATED BY RS-GA (INDIVIDUAL MEASUREMENTS)

Index	List of rules to identify the faulted section	Zone	Yes
1	$I_1(H)$	Z11	$\surd$
2	$I_5(H) \cdot I_6(H)$	Z11	$\surd$
3	$I_5(H1) \cdot V_7(N)$	Z25	$\surd$
4	$I_5(H1) \cdot V_8(N)$	Z25	R
5	$I_1(L) \cdot I_6(H) \cdot V_7(N)$	Z25	$\surd$
6	$I_5(H1) \cdot I_9(L) \cdot I_{11}(L)$	Z25	$\times$
7	$I_6(L) \cdot I_7(H)$	Z27	$\surd$
8	$I_8(L)$	Z27	$\times$
9	$I_5(H1) \cdot V_8(L)$	Z2T	$\surd$
10	$I_6(H) \cdot V_7(L)$	Z2T	$\surd$
11	$I_6(H) \cdot V_8(L)$	Z2T	R
12	$I_9(H)$	Z2T	$\surd$
13	$I_{11}(H)$	Z2T	$\surd$
14	$I_6(H1) \cdot V_8(N)$	Z36	$\surd$
15	$I_6(H1) \cdot V_7(N)$	Z36	R
16	$I_6(H1) \cdot I_{10}(L) \cdot I_{12}(L)$	Z36	$\times$
17	$I_1(L) \cdot I_5(H) \cdot V_8(N)$	Z36	$\surd$
18	$I_7(L)$	Z38	$\times$
19	$I_6(L) \cdot I_8(H)$	Z38	$\surd$
20	$I_5(L) \cdot I_8(H)$	Z38	R
21	$I_5(H) \cdot V_8(L)$	Z3T	$\surd$
22	$I_5(H) \cdot V_7(L)$	Z3T	R
23	$I_6(H1) \cdot V_8(L)$	Z3T	$\surd$
24	$I_6(H1) \cdot V_7(L)$	Z3T	R
25	$I_{10}(H)$	Z3T	$\surd$
26	$I_{12}(H)$	Z3T	$\surd$

$\surd$: rules that are good enough to be considered. $\times$ indicates the bad rules. 'R': rules that are redundant.

TABLE V

REDUCTS GENERATED USING RS-GA (TIME SERIES DATA)

Index No.	Number of reducts	Support	Length
1	I_1, V_1, BRK5-BRK7	100	3
2	I_1, V_1, BRK6-BRK8	100	3
3	I_1, V_2, BRK6-BRK8	100	3
5	I_1, V_3, BRK6-BRK8	100	3
6	I_1, V_3, BRK5-BRK7	100	3
7	I_1, V_4, BRK5-BRK7	100	3
8	I_1, V_4, BRK5-BRK8	100	3
9	I_1, V_6, BRK5-BRK7	100	3
10	I_1, V_6, BRK6-BRK8	100	3

TABLE VI

RULES GENERATED BASED ON THE FIRST REDUCT SET IN TABLE V

Index No.	*Time t/s*	IED1 V1	IED1 I1	Breaker BRK	Decision *d*	System states
1	*0.139*	N	•	0	N	N
2	*1.004*	L	N	•	A_0	A
3	*1.005*	•	H	0	E_0	E
4	*1.419*	•	H	1	E_1	
5	*1.443*	•	L	•	S_1	S

System states: N(NORMAL), A(ALERT), E(EMERGENCY) and S(SAFE)
Decision: A_0(Alert and Breaker Closed), E_1(Emergency and Breaker Opened), S_1(Safe and Breaker Opened), BRK: BRK5 - BRK7. It can also be BRK6 - BRK8

slighty misleading here. Only one event was unclassified in the 10% of data for testing whereas in the 30% partition, two events are actually unclassified. However, the larger size of the test set is used it tends to have more events i.e. 11 events in the 30% partitions, but only 4 events in the 10%. As the strength of the rule relies upon the large support basis that describes the number of events, which support each rule, the accuracy estimated from the 30% partition is therefore higher than the 10% partition even though it has two unclassified events. The number of unclassified and/or misclassified events in power system should be treated with high priority as it indicates the failure in the classifier and hence it needs to be verified by experts. The results showed us that the overall classification rates are very good even though a small dataset has been used.

TABLE VII

CLASSIFIER RESULT USING THE 90% TRAINING SET AND 10% TEST SET

Training Set (90%)	Test Sets (10%) Split 1	Split 2	Split 3	Split 4	Mean Accuracy
1	1.000	1.000	1.000	1.000	1.000
2	1.000	1.000	1.000	1.000	1.000
3	1.000	1.000	0.750	1.000	0.938
4	1.000	1.000	1.000	1.000	1.000
		Measure of Accuracy			**0.984**

TABLE VIII

CLASSIFIER RESULT USING THE 70% TRAINING SET AND 30% TEST SET

Training Set (70%)	Test Sets (30%) Split 1	Split 2	Split 3	Split 4	Mean Accuracy
1	0.909	1.000	1.000	1.000	0.977
2	1.000	1.000	1.000	1.000	1.000
3	1.000	1.000	0.909	1.000	0.977
4	1.000	1.000	1.000	1.000	1.000
		Measure of Accuracy			**0.989**

TABLE IX

CLASSIFIER RESULT USING THE 50% TRAINING SET AND 50% TEST SET

Training Set (50%)	Test Sets (50%) Split 1	Split 2	Split 3	Split 4	Mean Accuracy
1	0.941	1.000	0.941	1.000	0.971
2	1.000	1.000	1.000	1.000	1.000
3	0.706	0.765	0.471	0.765	0.676
4	1.000	1.000	1.000	1.000	1.000
		Measure of Accuracy			**0.912**

VII. CONCLUSION

This paper suggests the use of a hybrid RS-GA method to process and extract implicit knowledge from operational data derived from relays and circuit breakers. The proposed technique simplifies our rule generation process, reduces the rule maintenance costs, the outage response time and resources required to develop a rule-based diagnostic system. Two given examples have demonstrated how knowledge can be induced from data sets and from these simplified examples, the results has shown promise for practical application. The approach is also able to extract rules from a time series data. The advantage of integrating RST and GA approach is that the former can effectively reduce a large quantity of data and generate efficient rules whereas the latter provides us flexibility to generate a number of rules subject to the parameter settings we provide. It results a simple problem formulation that is able to produce global optimal solutions. More importantly, the approach is able to adapt itself to the new situation after a few generations [12]. The methodology is more attractive than some techniques i.e. Bayesian approach because no assumption about the independence of the attributes is necessary nor is any background knowledge about the data required [13].

Classic expert systems were developed to handle a specific task for a *fixed* network topology. Increased penetration of distributed generation (DGs) will alter the distribution network topology from time to time and could complicate the fault diagnosis task. Rules developed in the conventional knowledge base systems are no longer efficient enough of supervising the active network. An automated rule induction system is being developed to extract rules from various simulated topologies in a distribution system caused by the connection/disconnection of DGs. The generic rules applicable to all the simulated topologies are identified and stored in one common database while other rules unique to the topological changes (via mapping) are kept separately in other databases.

REFERENCES

[1] C. Hor, A. Shafiu, P. Crossley and F. Dunand. Modelling a substation in a distribution network: real time data generation for knowledge extraction. *IEEE Power Engineering Society Summer Meeting*, Chicago, Illinois, USA, July 21-25, 2002.

[2] C. Hor, K. Kangvansaichol, P. Crossley and A. Shafiu. Relay modelling for protection studies. *IEEE Bologna Power Tech 2003 Conference*, Bologna, Italy, June 23-26, 2003.

[3] C. Hor, P. Crossley and S. Watson. Building Knowledge for Substation based Decision Support using Rough Sets. *IEEE Transactions on Power Delivery. In press*, April 2006.

[4] C. Hor and P. Crossley. Extracting Knowledge from Substations for Decision Support. *IEEE Transactions on Power Delivery*, 20(2), pp.595-602, April 2005.

[5] C. Hor and P. Crossley. Unsupervised Event Extraction within Substations using Rough Classification. *IEEE Transactions on Power Delivery*, 21(4), pp.1809-1816, October 2006.

[6] J. Bazan, A. Skowron and P. Synak. Discovery of Decision rules from Experimental data. *Third Internationl Workshop on Rough Sets and Soft Computing Proceedings (RSSC '94)*(T. Lin and A. Wildberger eds.), (San Jose State University, San Jose, California, USA), pp. 276-279, November 1994.

[7] S. Vinterloo and A. Øhrn. Approximate Minimal Hitting Sets and Rule Templates. *International Journal of Approximate Reasoning*, Vol. 25, pp. 123-143, 2000.

[8] A. Øhrn. Discernibility and Rough Sets in Medicine: Tools and Applications. PhD thesis, Department of Computer Science and Information Science, Norwegian University of Science and Technology, Trondheim, Norway, February 2000.

[9] L. Li and J. Yunfei. Computing Minimal Hitting Sets with Genetic Algorithm. *International Workshop on Principles of Diagnosis (DX'02)*, Austria, May 2000.

[10] A. Skowron and C. Rauszer. The discernibility matrices and functions in information systems,. Intelligent Decision Support - Handbook of Applications and Advances of the Rough Sets Theory, Kluwer Academic Publishers, Dordrecht, 1992, pp331-362.

[11] IEEE Tutorial on Modern Heuristic Optimisation Techniques with Applications to Power Systems. IEEE PES Tutorial 02TP160.

[12] J. Wroblewski. Finding minimal reducts using Genetic Algorithm. *Second Annual Join Conference on Information Sciences*, pp. 186-189, October 1995.

[13] Ø. Aasheim and H. Solheim. Rough Sets as a Framework for Data Mining. Technical report, Knowledge Systems Group, Faculty of Computer Systems and Telematics, The Norwegian University of Science and Technology, Trondheim, Norway, May 1996.

Chinglai Hor is a lecturer in Renewable Energy at the University of Exeter in Cornwall (UEC). He received his B.Eng (Hons) in Electrical & Electronic Engineering from the University of Manchester Institute of Science and Technology (UMIST), UK in 1997 and his Ph.D. degree from the Queen's University of Belfast, UK in 2004. He worked as an electrical engineer in ALSTOM Power from 1997 to 2000. He joined UEC in Oct 2006 after working for 3 years as a postdoctoral research associate with the Centre for Renewable Energy Systems Technology (CREST), Loughborough University, UK.

Peter A. Crossley is the Professor of Power Systems at the University of Manchester. He graduated with a B.Sc degree from the University of Manchester Institute of Science and Technology (UMIST), UK in 1977 and a PhD degree from the University of Cambridge, UK in 1983. He has been involved in various research projects on the applications of GPS in electrical networks and the technical problems associated with the connection and transport of electrical energy. He joined Manchester in 2006 after 4 years at Queen's University of Belfast, 11 years with UMIST and 9 years in Industry at GEC/ALSTHOM.

Dean L. Millar is a senior lecturer at the University of Exeter in Cornwall (UEC), with expertise in Mining Engineering, Rock Mechanics and Renewable Energy. He is the programme director for UEC's B.Sc. Renewable Energy course and is also a Renewable Energy Business Fellow. His current research interests focus on themes within the renewable energy field, specifically: marine renewables (including wave and tidal energy) and transport biofuels. He has developed a robust genetic algorithm software during his Ph.D. in Imperial College, UK.