POWER SYSTEM FAULT ANALYSIS BASED ON INTELLIGENT TECHNIQUES AND INTELLIGENT ELECTRONIC DEVICE DATA

A Dissertation

by

XU LUO

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

May 2007

Major Subject: Electrical Engineering

POWER SYSTEM FAULT ANALYSIS BASED ON INTELLIGENT TECHNIQUES AND INTELLIGENT ELECTRONIC DEVICE DATA

A Dissertation

by

XU LUO

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Chair of Committee,	Mladen Kezunovic
Committee Members,	Chanan Singh
	Edgar Sanchez-Sinencio
	William Lively
Head of Department,	Costas N. Georghiades

May 2007

Major Subject: Electrical Engineering

ABSTRACT

Power System Fault Analysis Based on Intelligent Techniques and Intelligent Electronic Device Data. (May 2007)

Xu Luo, B.S., Xi'an Jiaotong University;

M.S., Xi'an Jiaotong University

Chair of Advisory Committee: Dr. Mladen Kezunovic

This dissertation has focused on automated power system fault analysis. New contributions to fault section estimation, protection system performance evaluation and power system/protection system interactive simulation have been achieved. Intelligent techniques including expert systems, fuzzy logic and Petri-nets, as well as data from remote terminal units (RTUs) of supervisory control and data acquisition (SCADA) systems, and digital protective relays have been explored and utilized to fufill the objectives.

The task of fault section estimation is difficult when multiple faults, failures of protection devices, and false data are involved. A Fuzzy Reasoning Petri-nets approach has been proposed to tackle the complexities. In this approach, the fuzzy reasoning starting from protection system status data and ending with estimation of faulted power system section is formulated by Petri-nets. The reasoning process is implemented by matrix operations. Data from RTUs of SCADA systems and digital protective relays are used as inputs. Experiential tests have shown that the proposed approach is able to perform accurate fault section estimation under complex scenarios.

The evaluation of protection system performance involves issues of data acquisition, prediction of expected operations, identification of unexpected operations and diagnosis of the reasons for unexpected operations. An automated protection system performance evaluation application has been developed to accomplish all the tasks. The application automatically retrieves relay files, processes relay file data, and performs rule-based analysis. Forward chaining reasoning is used for prediction of expected protection operation while backward chaining reasoning is used for diagnosis of unexpected protection operations. Lab tests have shown that the developed application has successfully performed relay performance analysis.

The challenge of power system/protection system interactive simulation lies in modeling of sophisticated protection systems and interfacing the protection system model and power system network model seamlessly. An approach which utilizes the "compiled foreign model" mechanism of ATP MODELS language is proposed to model multifunctional digital protective relays in C++ language and seamlessly interface them to the power system network model. The developed simulation environment has been successfully used for the studies of fault section estimation and protection system performance evaluation.

To my wife and my parents

ACKNOWLEDGMENTS

I am sincerely grateful to my advisor Dr. Mladen Kezunovic for his advice, support and patience, which made this dissertation possible. During my study at Texas A&M University, I learned from him much knowledge of power systems as well as unique methodology to explore, formulate and express new ideas.

Special thanks are extended to my committee members Dr. Chanan Singh, Dr. Edgar Sanchez-Sinencio, and Dr. William Lively for their precious time and valuable comments.

I also appreciate all the help from my colleagues Mr. Goran Latisko, Mr. Zarko Djekic, Mr. Yang Wu, Dr. Nan Zhang and Dr. Hongbiao Song.

Acknowledgement is extended to NSF I/UCRC Power Systems Engineering Research Center (PSerc) and Electric Power Research Institute (EPRI) for their research funding and CenterPoint Energy for their contribution of experimental data.

TABLE OF CONTENTS

LIST OF TABLES

TABLE Page

LIST OF FIGURES

FIGURE Page

FIGURE Page

CHAPTER I

INTRODUCTION

A. Introduction

This chapter introduces the scope and solution for power system fault analysis. It serves as an explanation of the problem domain which the dissertation focuses on. First, power system fault analysis is classified into two major categories, and sub-categories are further differentiated. Then the relationship between the two major categories and the relationship among sub-categories are discussed. Finally the current trend towards automated fault analysis is emphasized, and the techniques and input data are introduced with examples.

B. Scope of Power System Fault Analysis

The increasing competition in the utility industry requires maintaining power delivery service with minimum interruption. The goal of power system fault analysis is to provide enough information to utility staff to be able to understand the reasons for the interruption better, and provide as quick as possible an action to restore the power delivery. The analysis should also provide enough understanding of the status of protection system components so that a preventive set of measures can be implemented to reduce the likelihood of service interruption and damages to equipment [1].

The scope of power system fault analysis can be generally classified into two categories: fault event analysis and protection system performance evaluation. Fault

---

The journal model is *IEEE Transactions on Automatic Control.*

event analysis focuses on determination of faulted section, fault type, fault location and fault inception angle. Protection system performance evaluation is to check whether a protection system has operated as expected, and if not, what are the causes.

Within the scope of fault event analysis, determination of faulted section is usually the first step. There are two reasons. First, it gives the quickest and the most important information about what happens in the power system. Second, the detailed fault information such as fault location, fault type and fault inception angle can be determined more easily and accurately by analyzing the data from the fault recording equipments close to the faulted section than from those far from the faulted section.

Fault event analysis and protection system performance evaluation have close relationship, because the fault events and protection systems operate in a cause and effect manner. Since a fault is the cause of the operation of a protection system, the status of the protection system will contribute to the analysis of the fault event. On the other hand, because the protection system is supposed to operate according to certain fault situation, fault event information such as fault type and fault location provides reference to the expected status of the protection system and thus contribute to the evaluation of the protection system. Fig. 1 illustrates the scope of power system fault analysis as well as the relationship among its sub-categories.

The results of power system fault analysis serve three groups of utility staff. System operators require the fault event information to conduct restoration procedures to return the system to a normal state as soon as possible. Protection engineers need the protection system performance information to assess the correctness of the response of a protection system to a given fault condition. Maintenance staff requires both the fault event information and protection system performance information to locate and repair faulted components [1].

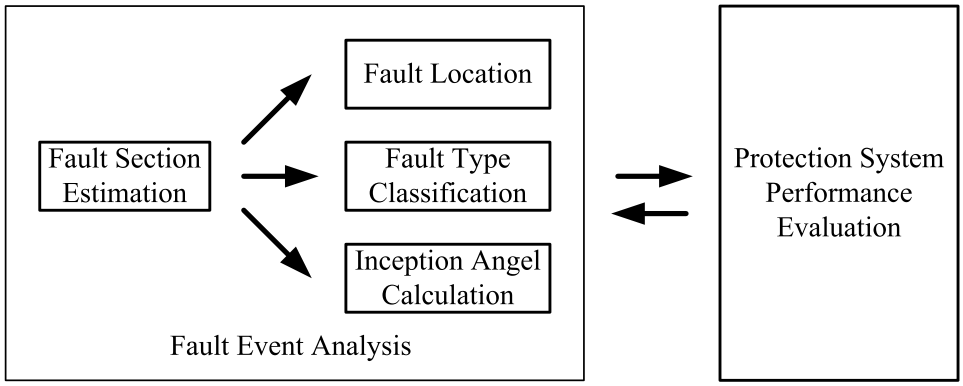

Fig. 1. Scope of power system fault analysis

C. Solutions for Automated Fault Analysis

As the scale of modern power systems grows dramatically, the traditional manual fault analysis becomes more and more difficult due to the complexity of systems and large volume of incoming data. To deal with such a dilemma, computer based automated fault analysis has gained significant attention around the world. The history on this subject dates to the late eighties when first expert systems for automated fault analysis based on the data from Remote Terminal Units (RTUs) of Supervisory Control and Data Acquisition (SCADA) Systems were introduced [1]. Since then, this field has advanced with new developments being pursued in two general directions. One direction was the introduction of a variety of intelligent techniques, besides expert systems, such as neural networks, fuzzy logic, Petri-nets, etc.. As examples, a hybrid expert system was developed for faulted section identification, fault type classification and selection of fault location algorithms [2]. An Adaptive Resonance Theory neural network with fuzzy decision rules was proposed to classify power system faults [3]. A Petri-nets combined with coding theory is used for fault diagnosis for substation automation [4].

The other direction was the use of data from Intelligent Electronic Devices

(IEDs), besides SCADA RTUs, such as Digital Fault Recorders (DFRs), Digital Protective Relays (DPRs), Sequence of Event Recorders (SERs), etc. As examples, reference [5] presents a solution for automated fault analysis of disturbance events and protection system operations using DFR Data. Reference [6] proposes an approach to perform comprehensive fault analysis by integrating DFR data and DPR data.

D. Summary

The scope and solution of power system fault analysis are introduced in this chapter. Power system fault analysis can be classified into fault event analysis and protection system performance evaluation. These two categories have close relationship due to the cause and effect relationship between the fault events and operations of protection systems. Among the sub-categories of fault event analysis, determination of faulted section is the first and the most important step. Because of the large volume of data in modern power systems, computer based automated analysis has been proposed as a solution to power system fault analysis. This field has advanced significantly with the application of intelligent techniques and intelligent electronic devices.

CHAPTER II

BACKGROUND

A. Introduction

This chapter provides the background knowledge for the dissertation study. First, different sections of a power system and their corresponding protection systems are explained. Then the theories of rule-based expert system, fuzzy logic and Petri-nets, which are the intelligent techniques used for fault analysis, are studied. Finally SCADA systems and digital protective relays, which are the data sources for fault analysis, are introduced.

B. Power System and Protection System

A power system is composed of a lot of sections such as generators, transformers, bus bars and transmission lines. These sections are protected by protective relaying systems comprising instrument transformers, protective relays, circuit breakers and communication equipments. In case of a fault occurring on a section, its associated protective relays should detect the fault and issue trip signals to open their associated circuit breakers to isolate the faulted section from the rest of the power system, in order to avoid further damage to the power system. Fig. 2 is an example of power system sections with their protection systems. G1 is a generator. T1 is a transformer. B1,...,B5 are bus bars. L45 is a transmission line. RG is a generator protective relay. RT is a transformer protective relay. RB is a bus protective relay. RL-4,...,RL-9 are transmission line protective relays. C1,..., C9 are circuit breakers.

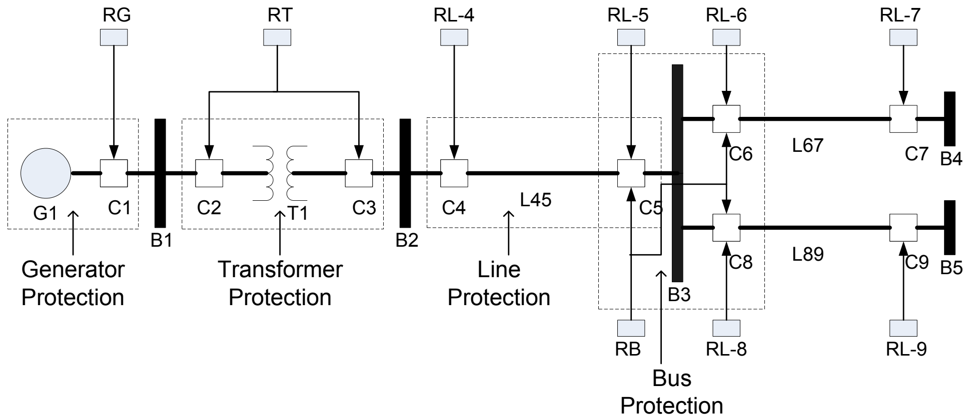

Fig. 2. An example of power system sections with their protection systems

C. Fault Analysis Techniques

1. Rule-based Expert System

An expert system is a computer system which emulates the decision-making ability of a human expert. When expert systems were first developed in the 1970's, they contained expert knowledge exclusively. Today, the term expert system is often applied to any solution which uses expert system technology. The knowledge in an expert system may be either expertise, or knowledge which is generally available from books and knowledgeable persons. The terms expert system and knowledge-based system are often used synonymously [7].

The knowledge of a knowledge-based system may be represented in the form of IF THEN type rules. Such a knowledge-based system is called a rule-based expert system. The elements of a typical rule-based expert system are shown in Fig. 3. The knowledge base contains the domain knowledge needed to solve problems coded in the form of rules. The working memory is a global database of facts used by the rules. The inference engine makes inferences by deciding which rules are satisfied by facts, prioritizes the satisfied rules, and executes the rule with the highest priority. The

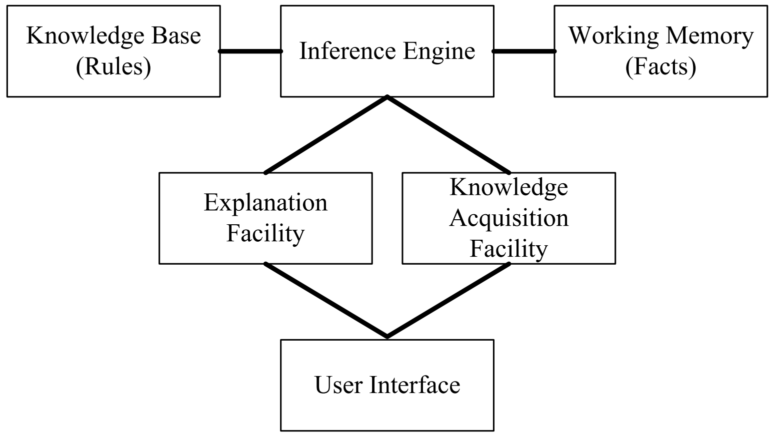

Fig. 3. Structure of a rule-based expert system

user interface is used for communication between the user and the expert system. The explanation facility presents the reasoning process to a user. The knowledge acquisition facility establishes an automatic way for the user to enter knowledge into the system [7].

Each rule in the knowledge base is identified by a name. Following the name is the IF part of the rule. The section between the IF and THEN part of the rule is called by various names such as antecedent, conditional part, pattern part, or left-hand-side(LHS). The individual condition is called a conditional element or a pattern. A rule whose patterns are all satisfied is said to be activated or instantiated. Multiple rules may be activated at the same time. In this case, the inference engine must select one rule for firing. Following the THEN part of a rule is a list of actions to be executed when the rule fires. These actions usually are insertion, deletion and modification of facts. This part of the rule is called a consequent or right-hand-side (RHS) [7]. As an example, a rule expressed in an equivalent pseudocode in an IF THEN format used to determine a fault on Bus 3 in Fig. 2 is as follows.

Rule: Bus 3 Fault

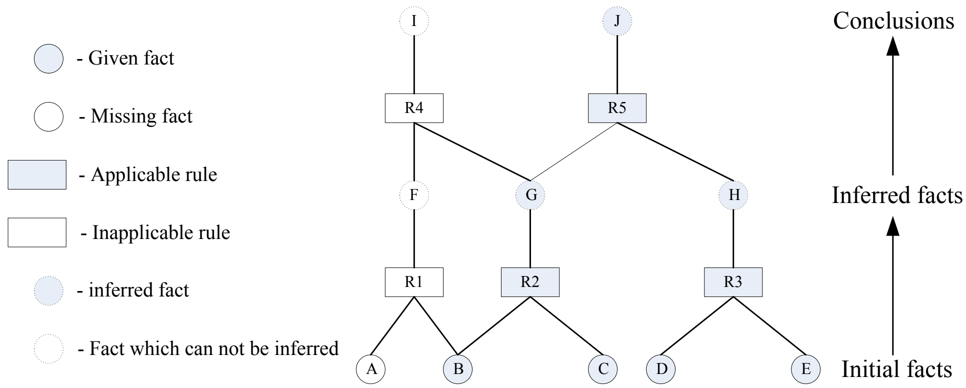

Fig. 4. An example of forward chaining

IF

Relay RB tripped AND

Circuit Breaker C5 opened AND

Circuit Breaker C6 opened AND

Circuit Breaker C8 opened

Then

A fault occured on Bus 3

A group of multiple inferences that connect a problem with its solution is called a chain. Forward chaining is reasoning from facts to conclusions [7]. Fig. 4 illustrates the concept of forward chaining in a rule-based system. Rules are triggered by the facts which satisfy their antecedent.For example, rule R1 must be satisfied by facts A and B for it to be activated. However, only fact B is present and rule R1 is not activated to produce the fact F. Then R4 is not activated because of the absence of the fact F. Rule R2 is activated by facts B and C which are present and so rule R2 produces the intermediate fact G. Other satisfied rules are rule R3 and R5. The execution of the rule R5 produce the conclusion which is the fact J.

Backward chaining is reasoning in reverse from a hypothesis, which is a potential conclusion to be proved, to the facts which support the hypothesis. A hypothesis

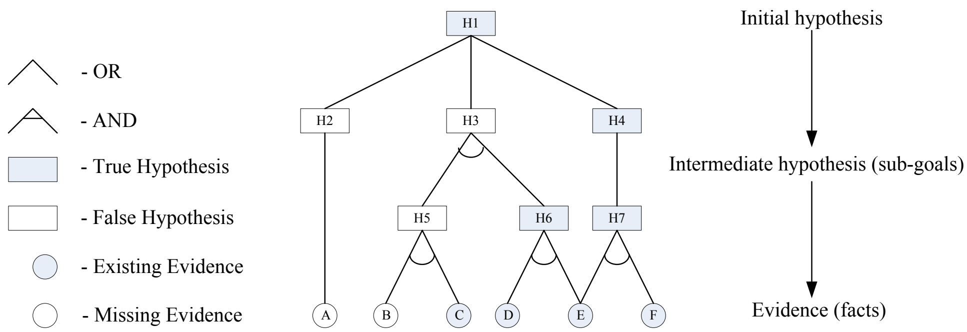

Fig. 5. An example of backward chaining

can be viewed as a fact whose truth is in doubt and needs to be established. The hypothesis can then be interpreted as a goal to be proven [7]. Fig. 5 illustrates the concept of backward chaining. In order to prove hypothesis H1, at least one of the intermediate hypotesis H2, H3 and H4 must be proven. To prove H2, fact A must exist. Since fact A is not present, H2 is disproven. To prove hypothesis H3, both hypothesis H5 and H6 must be proven. Since the absence of fact B will disprove hypothesis H5, Hypothesis H3 is disproven. To prove hypothesis H4, hypothesis H7 must be proven. The existence of fact E and F will prove hypothesis H7, hence hypothesis H4 is proven. Finally hypothesis H1 is proven.

2. Fuzzy Logic

Fuzzy logic is a logic based system that generalizes the classical two-valued logic for reasoning under uncertainty. The concept of fuzzy sets, the core of fuzzy logic, was first introduced by Lotfi A. Zadeh in his seminal paper in 1965 [8].

A classical Boolean set A, may be equated with its characteristic function:

$$\varphi_A : X \longrightarrow \{0, 1\} \tag{2.1}$$

which associates with each element x of a universe of discourse X a number $\varphi(x) \in \{0, 1\}$ such that $\varphi(x) = 0$ means that x does not belong to the set A, and $\varphi(x) = 1$

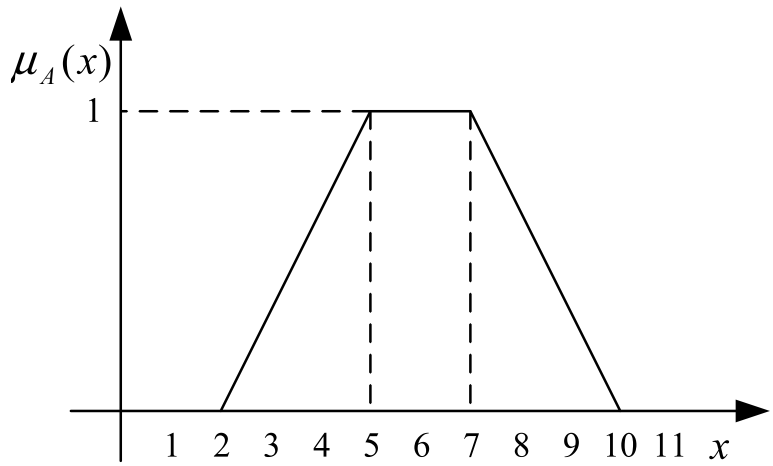

Fig. 6. Membership function of a fuzzy set "integer numbers which are more or less 6"

means that x belongs to the set A.

Unlike the classical Boolean set, elements of a fuzzy set may belong to it to partial degree, from full belongingness to the full nonbelongingness through all intermediate values. Thus the characteristic function: $\varphi_A : X \longrightarrow \{0, 1\}$ is replaced by a membership function:

$$\mu_A : X \longrightarrow [0, 1] \tag{2.2}$$

such that $\mu_A(x) \in [0, 1]$ is the degree to which an element x belongs to the fuzzy set A. $\mu_A(x) \in [0, 1]$ is called the grade of membership [9]. As an example, Fig. 6 describes a trapezoidal membership function of a fuzzy set "integer numbers which are more or less 6".

Similarly as in the classical Boolean set theory, the basic operations in fuzzy set theory are complement, intersection and union.

The complement of a fuzzy set A in X, written as $\neg A$, is defined as

$$\mu_{\neg A}(x) = 1 - \mu_A(x) \qquad \forall x \in X \tag{2.3}$$

The complement corresponds to the negation 'not'.

The intersection of two fuzzy sets A and B in X, written as $A \bigcap B$, is defined as

$$\mu_{A \bigcap B}(x) = \mu_A(x) \wedge \mu_B(x) \qquad \forall x \in X \tag{2.4}$$

where '$\wedge$' is usually a minimum operation, i.e. $a \wedge b = min(a, b)$. The intersection of two fuzzy sets corresponds to the connective 'and'.

The union of two fuzzy sets A and B in X, written as $A \bigcup B$, is defined as

$$\mu_{A \bigcup B}(x) = \mu_A(x) \vee \mu_B(x) \qquad \forall x \in X \tag{2.5}$$

where '$\vee$' is usually the maximum operation, i.e. $a \vee b = max(a, b)$. The union of two fuzzy sets corresponds to the connective 'or'.

It should be mentioned that beside the above conventional basic operations, some other definitions can also be used. As examples, for intersection, the algebraic product $\mu_{A \bigcap B}(x) = \mu_A \cdot \mu_B$ for intersection and , the probabilistic product $\mu_{A \bigcup B}(x) = \mu_A + \mu_B - \mu_A \cdot \mu_B$ for union are popularly employed [9]. An important issue is the adequacy of the operations on fuzzy sets, i.e. whether they do reflect the real human perception of their essence (the real semantics of 'not', 'and' and 'or') [10].

In order to properly represent real-world knowledge where ambiguous, vague and imprecise data are involved, fuzzy rules have been used for knowledge representation [11]. A fuzzy rule is a rule describing the fuzzy relation between two propositions. Let R be a set of fuzzy rules $R = \{R_1, R_2, ..., R_n\}$. The general formulation of the ith fuzzy rule is as follows:

$R_i(c_i)$: IF $P_j(\theta_j)$ THEN $P_k(\theta_k)$

where P_j and P_k are propositions which may contain some fuzzy variables. The truth of each proposition θ_j, θ_k are a real values. $\theta_j \in [0, 1]$, $\theta_k \in [0, 1]$. $c_i \in [0, 1]$. It represents the strength of the belief in the rule. The larger the value is, the more the rule is believed in.

If the antecedent part or the consequent part of a fuzzy rule contains ‘and’ and/or ‘or’ connectives, it is called a composite fuzzy rule. The composite fuzzy rule can be classified into the following types [12]:

Type 1: $R_i(c_i)$: $P_1(\theta_1)$ AND $P_2(\theta_2)$ AND ... AND $P_{k-1}(\theta_{k-1}) \longrightarrow P_k(\theta_k)$

Type 2: $R_i(c_i)$: $P_1(\theta_1) \longrightarrow P_2(\theta_2)$ AND ... AND $P_{k-1}(\theta_{k-1})$ AND $P_k(\theta_k)$

Type 3: $R_i(c_i)$: $P_1(\theta_1)$ OR $P_2(\theta_2)$ OR ... OR $P_{k-1}(\theta_{k-1}) \longrightarrow P_k(\theta_k)$

Type 4: $R_i(c_i)$: $P_1(\theta_1) \longrightarrow P_2(\theta_2)$ OR ... OR $P_{k-1}(\theta_{k-1})$ OR $P_k(\theta_k)$

Rules of Type 4 are unsuitable for deducing control because they make no specific implication. We will focus on the first three types of rules.

The reasoning results of the first three types of rules can be expressed as Eq. 2.6, Eq. 2.7, Eq. 2.8 respectively.

$$\theta_k = OP_{\cap}(\theta_1, \theta_2, ..., \theta_{k-1}) * c_i \tag{2.6}$$

where $OP_{\cap}$ is an operation corresponding to the connective ‘and’.

$$\theta_2 = \theta_1 * c_i \theta_3 = \theta_1 * c_i ... \theta_k = \theta_1 * c_i \tag{2.7}$$

$$\theta_k = OP_{\cup}(\theta_1, \theta_2, ..., \theta_{k-1}) * c_i \tag{2.8}$$

where $OP_{\cup}$ is an operation corresponding to the connective ‘or’.

As an example, the rule ‘Bus 3 Fault’ discussed in previous section is written as a fuzzy rule and is given certainty factor of the rule and truth values of antecedent propositions as follows:

Rule: Bus 3 Fault (0.9)

IF

Relay RB trips (0.9)AND

Circuit Breaker C5 opens (0.8)AND

Circuit Breaker C6 opens (0.8)AND

Circuit Breaker C8 opens (0.8)

Then

A fault occurs on Bus 3

If the $OP_\cap$ takes the form of minimum operation, the reasoning result will be as follows:

A fault occurs on Bus 3 with truth value of $\theta = min(0.9, 0.8, 0.8, 0.8) * 0.9 = 0.72$

3. Petri-nets

Petri-nets technique is first introduced by Carl A. Petri in 1962. It is a graphical and mathematical tool. The graphical aspect allows easy representation of the interaction between discrete events: parallelism, synchronism, precedence, alternatives and so on. The mathematical aspects allows formal modeling of these interactions and analysis of the properties of the modeled system. A formal definition of Petri-nets is as follows [9]:

Let N be the set of natural numbers and zero.

A Petri-nets is a 4-tuple

$$(P, T, Pre, Post)$$

where

1. $P = \{p_1, p_2, ..., p_n\}$ is a finite set of places.

2. $T = \{t_1, t_2, ..., t_m\}$ is a finite set of transitions.

3. Pre is the input incidence function:

   $Pre : P \times T \longrightarrow N.$

4. $Post$ is the output incidence function:

   $Post : P \times T \longrightarrow N.$

In the graphical representation of a Petri-net, places are denoted by circles and transitions by bars. Places are the nodes describing the states (a place is a partial state) and the transitions depict the state changes. The Pre incidence function describes the directed arcs connecting places to transitions. $Pre(p,t)$ is the weight of the arc (p,t). The absence of an arc between a place p and a transition t is denoted by $Pre(p,t) = 0$. The $Post$ incidence function describes the directed arcs connecting transitions to places. $Post(p,t)$ is the weight of the arc (t,p). The absence of an arc between a transition and a place is donated by $Post(p,t) = 0$.

In the matrical representation of a Petri-net, Pre is a $n \times m$ matrix of n rows (the places) and m columns (the transitions) whose elements belong to N. The vector $Pre(\cdot,t)$ denotes the input arcs of transition t with their weights. $Post$ is a $n \times m$ matrix of n rows (the places) and m columns (the transitions) whose elements belong to N. The vector $Post(\cdot,t)$ denotes the output arcs of transition t with their weights.

A marking M of a Petri-net $(P,T,Pre,Post)$ is a function $M : P \longrightarrow N$. It is a distribution of tokens in the places. It can be represented by a vector of dimension n of natural numbers. For $p \in P$, $M(p)$ is the token load of place p and represents a partial state of the system described by the Petri-nets. A marked Petri-net is a 2-tuple (N, M_0) where: N is a Petri-net and M_0 is its initial marking which is a function $M_0 : P \longrightarrow N$.

A transition t of a Petri-net is enabled for marking M if and only if $M \geq Pre(\cdot,t)$. This enabling condition expressed under the form of an inequality between two vectors is equivalent to

$$M(p) \geq Pre(p,t) \qquad \forall p \in P \tag{2.9}$$

Only enabled transitions can be fired. If M is a marking of a Petri-net enabling transition t and M' is the marking derived by the firing of t from M, then

$$M' = M + C(\cdot,t) \tag{2.10}$$

where

$C(\cdot,t) = Post(\cdot,t) - Pre(\cdot,t)$ is called the incidence matrix of the corresponding Petri-net.

As an example, a Petri-net modeling the operations of a protective relay is presented in Fig. 7. As shown in the figure, a protective relay has three states, dropout, pickup and operation which are represented by p_1, p_2 and p_3 respectively. The power system component protected by the relay has two states, absence of fault and existence of fault which are represented by p_4 and p_5 respectively. When a fault inception occurs, transition t_1 fires. The power system component protected by a relay goes into "existence of fault state" from "absence of fault" state. At the same time, the relay senses the fault and goes into "pickup" state from "dropout" state. When the relay's coordination timer is due, transition t_2 fires. The relay goes into "operation" state from "pickup" state. When the operation of the relay trips associated circuit breaker to clear the fault, transition t_3 fires. The power system component protected by the relay goes back to "absence of fault" state and the relay goes back to "dropout" state.

In the matrical representation, the structure of the Petri-net is given by the following matrices.

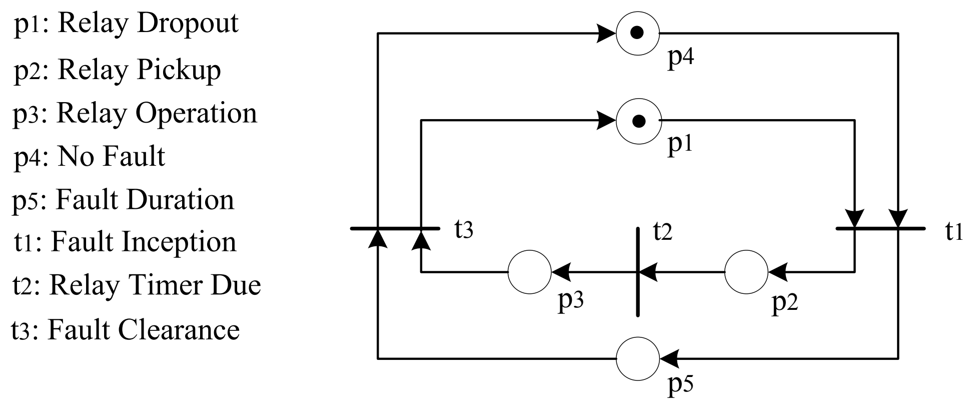

Fig. 7. A Petri-net describing the operations of a protective relay

$$Pre = \begin{array}{c} \begin{matrix} t_1 & t_2 & t_3 \end{matrix} \\ \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{array} \begin{matrix} p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \end{matrix} \qquad Post = \begin{array}{c} \begin{matrix} t_1 & t_2 & t_3 \end{matrix} \\ \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \end{array} \begin{matrix} p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \end{matrix}$$

The initial marking M_0 is given by the vector

$$M_0 = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 \end{bmatrix}^T$$

The initial marking will enable t_1 and the firing of t_1 will result in the marking M_1 according to Eq. 2.10.

$$M_1 = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 \end{bmatrix}^T + \begin{bmatrix} 0 & 1 & 0 & 0 & 1 \end{bmatrix}^T - \begin{bmatrix} 1 & 0 & 0 & 1 & 0 \end{bmatrix}^T = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 \end{bmatrix}^T$$

Such a marking represents the state when fault exists and relay picks up. The dynamics of t_2 and t_3 firing can also be described by Eq. 2.10 in a similar way.

Several extensions have been proposed for Petri-nets such as hierarchical nets, high level nets, temporal nets. An important extension comes with the investigation of

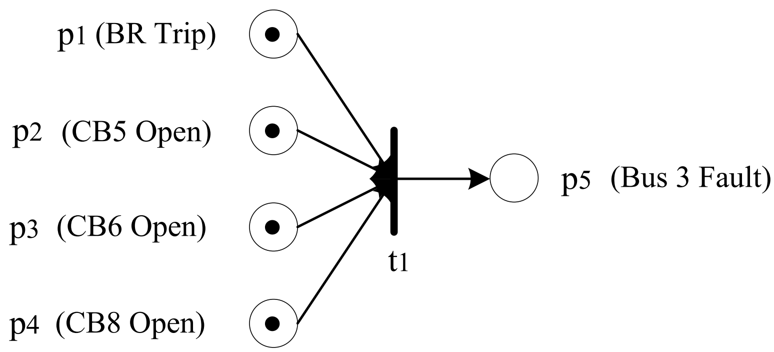

Fig. 8. A Petri-net representing the 'Bus 3 Fault' rule

the connection between logic and Petri-nets. A Petri-net can be applied to rule-based reasoning using proposition logic, where tokens represent the states of propositions [9]. As an example, the rule 'Bus 3 Fault' discussed in previous section can be represented by a portion of a Petri-net shown in Fig. 8.

D. Substation Equipments

1. SCADA System

Remote Terminal Unit (RTU) of Supervisory Control and Data Acquisition (SCADA) System is most widely used data acquisition equipment in substations. They are capable of recording status signals such as relay targets, circuit breaker status, transformer status and substation alarms, as well as analog signals such as bus voltages and line currents [13]. The recorded data from RTUs distributed in substations at different locations can be transferred to a central control center via certain communication links. Fig. 9 shows the common structure of SCADA systems.

There are several limitations of SCADA systems, which may restrict its performance in power system fault analysis applications.

1. The number of I/O ports of RTUs is limited. For monitoring of protection

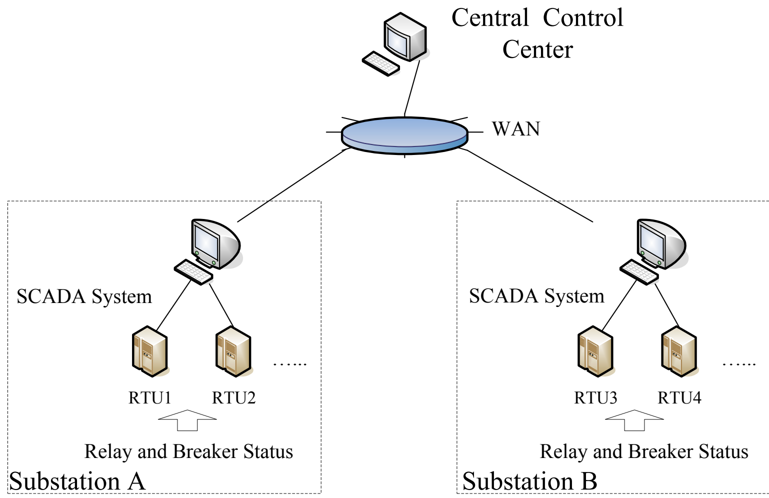

Fig. 9. Common structure of SCADA systems

system status, usually only relay trip signals and circuit breaker status signals are selected to be recorded. Due to the limitation, it is difficult to use additional information such as zone of operation, pickup, circuit breaker control circuit status to improve the determination of relay trip and circuit breaker switching status signals.

2. The timing accuracy of events recorded by RTUs is limited. RTUs usually have low scanning rates, which may be in the order of seconds. In practice, some RTUs use flags to label recorded events and then time-tag them using the scanning time. That means many events occurring in a short time interval may have the same time-stamp. Some other RTUs do not time-tag recorded events in the first place. Instead, the time when the master computer of the SCADA system receives event information is used as the time-stamp. Since there are always time delays because of data transmission, the timing accuracy of events is further degraded. Such limitation makes using sequence of events information difficult.

2. Digital Protective Relay

To achieve maximum flexibility, the firmware of digital relays is designed using the concept of functional elements. These elements usually include protection elements, control elements, and input and output contacts. The statuses of each element are represented by a set of predefined logic operands. As examples, Table I shows several logic operands for Ground Distance Zone 1 Element of GE's D60 relay [14].

Although relays from different manufacturers have different syntax for their logic operands, the functions of logic operands remain the same. First, logic operands can be used as logic variables to produce more complex schemes by logic operators in field programmable logic function provided by relays. Second, logic operands give information on the actual statuses of elements. Time-stamped logic operands are used as event record data to reflect detailed relay behaviors. Logic operands can also be recorded as digital oscillography data so that the statuses of elements can be visualized [14, 15].

When fault occurs, elements change their statuses according to their design principles and settings. A timed protection operation chain will be formed in order to trip the circuit breaker associated with the relay to interrupt fault currents in predetermined time. Fig. 10 illustrates the protection operation chain. In this chain, pickup of individual phases of elements is the first step and the current interruption by circuit breaker is the last step.

Along the chain, operation of any individual phase of an element will cause operation of the entire element. That is to say, operation of individual phases of a protection element triggers operation of the entire element through 'or' relation. Likewise, operation of several protection elements also triggers the relay trip through 'or' relation. Operation of an element may be blocked by pickup or operation of an-

Table I. Operands for Ground Distance ZONE 1 Element of D60 relay

Operand Syntax	**Description**
GND DIST Z1 PKP	Ground Distance Zone 1 has picked up
GND DIST Z1 OP	Ground Distance Zone 1 has operated
GND DIST Z1 PKP A	Ground Distance Zone 1 Phase A has picked up
GND DIST Z1 OP B	Ground Distance Zone 1 Phase B has operated
GND DIST Z1 DPO C	Ground Distance Zone 1 Phase C has dropped out

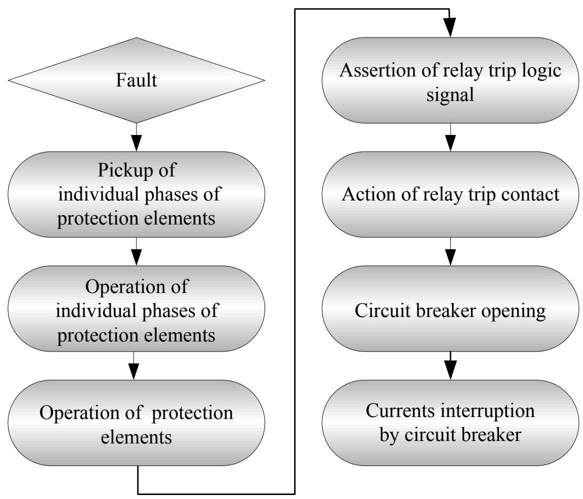

Fig. 10. Protection operation chain

other element or external block signals if pilot communication schemes are involved.

Most digital protective relays possess the capability of generating files which contain detailed data about power system fault disturbances and corresponding responses of protection system components. These data can be classified into four categories, namely oscillography data, setting data, fault data and event record data. Generally, oscillography data contain the records of what a relay "sees" during a disturbance event. Setting data specifies how the relay is configured. Fault data presents fault disturbance information calculated by the relay. Event data reveal how the relay and associated protection components actually respond to the disturbance event. Besides these relay-generated data, performance specification data such as the average pickup time for a phase distance element and average opening time for a circuit breaker are also important information. They are usually contained in user's manuals. Further description of relay file data is as follows:

1. Oscillography data: Oscillography data are generated by the fault recording function of a digital relay. Secondary voltages and currents coming into the relay are recorded as analog channels while statuses of both external contacts and internal states of the relay can be recorded as digital channels by users' selection.

2. Setting data: Setting data specify configuration parameters of a relay. Usually setting data configures the relay at three levels: selecting protection elements, deciding how the selected elements are logically combined, and setting operating parameters of each selected element.

3. Fault data usually include fault type, fault location, and voltage and current phasors during pre-fault and fault periods. They are calculated by a relay, but they may or may not be used for the relay decision making, depending on the relay design and application circumstances.

4. Event record data: Event record data are time-stamped logic operands in chronological order. It contains most of the information through which the external behavior of a relay and its associated protection system components and the internal states of the relay can be observed. According to our investigation, for some types of relays, not all logic operands that are important for analysis are reflected by event record data. This problem can be solved if users select these operands to be recorded in the oscillography files.

5. Performance specification data: Performance specification data define the relay

operating parameters which can be used to predict expected protection operation. Examples are the average pickup time of a phase distance element and the average opening time of a circuit breaker. Performance specification data are usually contained in user's manuals.

E. Summary

The background knowledge for the dissertation study is provided in this chapter. A power system consists of a lot of sections, which are protected by their corresponding protection systems. Based on such a relation, protection system data can be utilized for fault analysis. Rule-based expert system has strength in reasoning. Fuzzy logic excels in handling uncertainty. Petri-nets is an ideal graphical and mathematical tool to model and analyze discrete events. These intelligent techniques can be employed to deal with various complex fault analysis problems. SCADA systems are the traditional data source in power systems. Several limitations of SCADA systems may restrict their performance in fault analysis applications. Intelligent electronic devices such as digital protective relays provide abundant information about protection system operation as well as fault events for fault analysis applications.

CHAPTER III

PROBLEM TO BE SOLVED

A. Introduction

This chapter discusses the specific problems to be solved by the dissertation study. First, the problems of fault section estimation, protection system performance evaluation, and power system/protection system interactive simulation are presented and the complexities of the three problems are explained. Then the existing approaches to the three problems are presented and their shortcomings and disadvantages are emphasized. Finally the proposed approaches are outlined and their strengths to solve the problems are discussed.

B. Problem Statement

1. Fault Section Estimation

The problem of identification of faulted section is called fault section estimation. When a fault occurs on a certain section, the protection devices of protection systems will reach certain statuses accordingly. In the point view of a diagnosis problem, the fault on a given section is the cause, the statuses of the protection devices are effects. Thus the problem of fault section estimation can be defined as a diagnosis problem as follows: Given a set of observed statuses of protection devices, the goal is to identify the faulted power system section which explains those observations.

When a single fault occurs and all the statuses of protection devices are correctly observed, the fault section estimation problem is relatively simple. However, when multiple faults, failures of protection devices, and false data are involved, the task can be stressful and time consuming for system operators, because many situations can be hypothesized and the possibility of each of those situations needs to be examined. Multiple faults involve two or more faults which occur at the same time or in a short time interval at different locations of a power system. The multiple fault scenarios usually happen when cascading events occur. When protection devices fail to operate, backup protection devices will operate, which will cause more sections of a power system to be isolated. False data may be introduced by the logic of protection devices, measurement systems or communication systems. Multiple faults, failures of protection devices, and false data add uncertainty when identifying the actual faulted sections. When all the situations mix up, complexity of fault section estimation increases significantly.

A 14-bus power system and its protection systems shown in Fig. 11 are used as an example to explain the complexity. In Fig. 11, each bus bar is equipped with a bus relay. Each terminal of a transmission line is equipped with a main distance relay with forward zones. The distance relay also backs up the remote bus relay and the distance relays on the neighboring transmission lines in its forward direction. A bus bar is denoted as BXX. A bus relay shares the same number with its associated bus bar. A transmission line is denoted as LXX-XX, where XX is the number of the bus bar at each terminal of the transmission line. At each line terminal, a circuit breaker shares the same number with its associated distance relay.

A line fault F1 occurred on the line L13-14. The distance relays, RD38 and RD39, operated to send the trip signal to the circuit breakers, C38 and C39, respectively. Both of the two circuit breakers opened successfully. At this moment, a bus fault F2 occurred on the bus B13. The bus relay RB13 operated and sent trip signals to all the associated circuit breakers, C36, C37 and C38. However, the trip signal was not observed due to an error in the measurement system and the circuit breakers, C36 and C37, failed to open due to mechanical problems. Then the distance relay

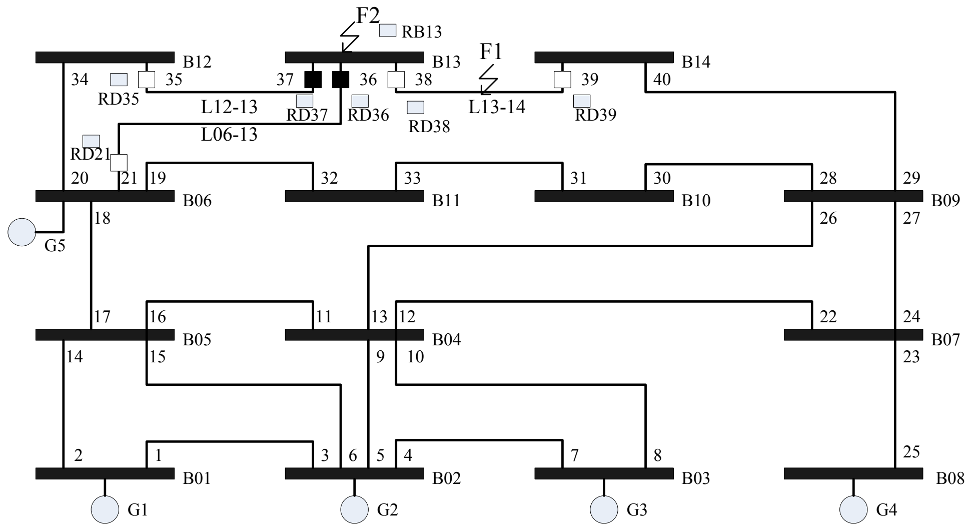

Fig. 11. An example of fault section estimation problem for a 14-bus system

Table II. Candidates of faulted sections for the 14-bus system example

Candidate No.	Faulted Section(s)	Failed Protection Device(s)	False Data
1	L13-14	C38	C38 status (should be reported closed but was falsely reported open)
2	L13-14, L12-13	RD37, C37	No false data
3	L13-14, L06-13	RD36, C36	No false data
4	L13-14, B13	C36, C37	RB13 trip signal (should be reported but was not reported)

RD35 on the line L12-13 and the distance relay RD 21 on the Line L06-13 operated as backup relays to open the circuit breakers, C35 and C21, respectively. Both of the two circuit breakers opened successfully. In such a scenario, several assumptions about the faulted sections can be made according to the observed relay trip signals and circuit breaker status signals. They are listed in Table II. As we can see, there are several candidates for faulted sections. Unless further investigations are made, it is difficult to tell where the actual faulted sections are.

2. Protection System Performance Evaluation

The well established criteria for protection system performance are dependability and security. Loss of dependability means the protection system is unable to trip when required, while loss of security means the protection system falsely trips when it is not expected to trip [16, 17]. In the context of fault analysis, the evaluation of protection system performance is to see if the system will operate or not operate as expected and diagnose the reasons for unexpected operations. The reasons for an unexpected operation may be a primary failure due to aging or random environmental stress, a secondary failure due to operating conditions that are out of design tolerance, and a command error due to incorrect input signals, settings and design [18]. The evaluation may involve several levels including the overall system, an individual device in the system and an element of a device.

The evaluation of protection system performance includes both identification of correct and incorrect operations, and diagnosis of the reasons for incorrect operations. There are several issues involved. First, a proper model of the protection system must be built in order to simulate the protection system operations. Second, an efficient mechanism needs to be employed so that the unexpected operations can be identified. Third, a sound strategy to trace the reasons for unexpected operations should be implemented. All the three issues render difficulties. To explain these difficulties, an EHV transmission line protection system using Directional Comparison Blocking (DCB) pilot scheme via Power Line Carrier (PLC) is shown in Fig. 12.

The protection system comprises two parts located at the terminal R and the terminal S of the transmission line respectively. The two parts are identical in functions and configurations but may interact with each other via communication signals. Each part mainly includes current transformers, voltage transformers, a protective re-

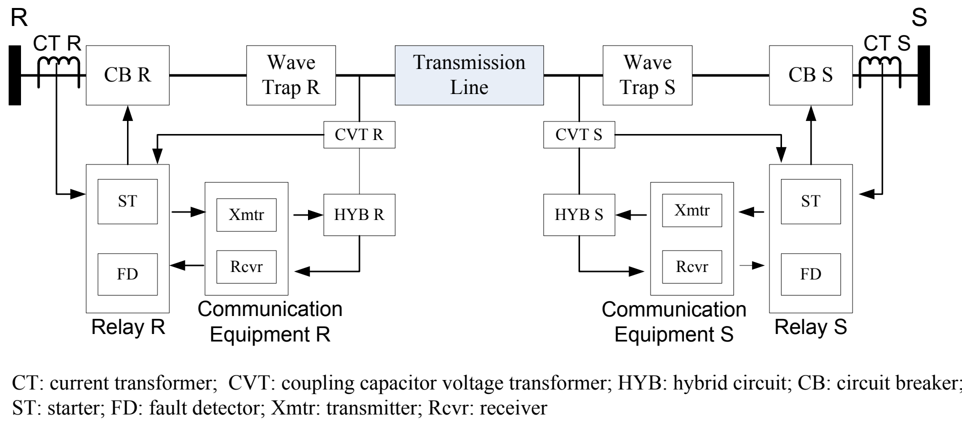

Fig. 12. An example of pilot protection system

lay, a circuit breaker and communication equipments. In the protective relay, several protection elements need to be configured to act as a starter and a fault detector to detect faults and discriminate the fault direction. The sensitivity, the reach of operating zone, and the timing of the starter and the fault detector should be coordinated well. The communication equipment includes a transmitter, a receiver and a hybrid circuit for impedance matching. They are in charge of sending and receiving block signals. The timing of the local trip signal must be carefully coordinated with the timing of the block signal sent from the remote terminal.

Modeling such a protection system requires that the protection scheme of the system, the behaviors of individual devices, and the dynamic interactions between individual devices be well understood and simulated to the necessary degree. To identify unexpected operations, not only the abnormal status needs to be observed but also the abnormal timing and sequence should be paid attention to. To diagnose the reasons for unexpected operations, the challenge lies in the fact that the unexpected operation of an element may be caused by a failure of the element itself or a failure of neighboring elements. To trace the ultimate reasons, the causal relations among elements should be taken into consideration.

To analyze the performance of such a system, as much as 20 designated check points are recommended by the Power System Relaying Committee of IEEE Power Engineering Society [19].

3. Power System/Protection System Interactive Simulation

The study of computer based power system/protection system interactive simulation generally covers power system network modeling, protective relay modeling and dynamic interaction between the power system network models and the relay models. The simulation is quite valuable for preliminary testing of relay algorithms, study of multi-terminal, coordinated relaying schemes, and evaluation of relay performance during cascade events [20, 21]. It plays important role in power system fault analysis because of two reasons. First, the cause and effect relation of power system faults and protection system behaviors can be can be studied in detail through interactive simulation. Second, the simulation can provide test data for various fault analysis applications.

The challenge of power system/protection system interactive simulation lies in modeling of sophisticated protection systems, and interfacing the protection system model and power system network model seamlessly. A protection system, especially a multifunctional digital protective relay consists of many functional components such as the interface to the power system and other protection systems, analog filters, analog to digital converters and protection elements implementing various protection algorithms. Besides, it is also capable of initializing its settings through relay set files and generating oscillography files and event reports to record what it "sees" and how it responds. To model such a sophisticated system requires a powerful programming language and good software design philosophy. The seamless interface between the protection system model and power system network model means that on one hand, the measurements from the power system model can be passed to the protection system model with minimal intervention and delay; on the other hand, the control signals from the protection system model can be passed to the power system model to open or close switches with minimal intervention and delay. In such a way, a real-time close loop simulation can be achieved [22].

C. Existing Approaches

1. Fault Section Estimation

Expert System (ES) is the earliest artificial intelligence technique applied to the problem of fault section estimation. Since the late eighties, various applications based on ES technique for fault section estimation have been reported in literature [2, 23–25]. Expert systems basically mimic the problem-solving behavior of experts using domain knowledge acquired during the knowledge acquisition process [26]. Since the fault section estimation is generally a diagnosis problem involving a number of fact-rule comparisons and consequent search steps which are usually used by fault analysis experts, the ES technique is well suited for that purpose. To achieve precise inference in complex cases, knowledge bases used in expert systems must involve a great number of rules covering all kinds of scenarios. The procedure of knowledge acquisition and knowledge base maintenance is quite burdensome. The response time of expert systems is usually not applicable to a real-time environment due to their conventional knowledge representation and inference mechanism.

Artificial Neural Network (ANN) technique is proposed as another potential solution to the problem of fault section estimation, as discussed in several papers [27–31]. ANN is a massively parallel distributed processor made up of simple processing units, which has a natural propensity for storing experimental knowledge [32]. The justification for application of ANN to the problem of fault section estimation lies in the fact that fault section estimation can be formulated as a problem of pattern recognition by mapping various combinations of statuses of protective relays and circuit breakers to faulted sections. This problem can be well solved by ANN's excellent non-linear input-output mapping capability. Some problems still remain unsolved in practical applications, such as slow convergence in the training process, and trivial determination process for the network parameters like hidden units, layers, learning rate and momentum value. The ANN approach has bad transparency, i.e., we can not determine how results are achieved. When any configuration of the power system or the protection system changes, the entire ANN needs to be re-trained.

In recent years, Petri-nets (PN) technique , which possesses the characteristics of graphic knowledge representation and parallel information processing, have gained researchers' strong interests, as demonstrated in the papers [4, 33–35]. Petri-nets are based on the concept that the relationships between the components of a system, which exhibits asynchronous and concurrent activities, could be represented by a net [33]. They are widely used to model and analyze discrete event systems. During a fault clearance process, the behavior of protection systems in terms of status changes of their components as well as the fault occurrence can be viewed as discrete events. Thus the behavior of protection systems and their relation to the fault occurrence can be modeled by Petri nets. This is the basic principle of Petri Nets approach to fault section estimation.

Besides the techniques discussed above, Fuzzy Logic technique is also employed to solve the problem of fault section estimation, as reported in the literature [36–38]. Fuzzy Logic offers a convenient means for modeling inexactness and uncertainties, hence a possible solution to handle the uncertainties due to unexpected operations of protective devices and false data in the problem of fault section estimation. The greatest inconvenience of Fuzzy Logic approach lies in the choice of the membership functions, usually defined based on empirical data.

Most of the solutions mentioned above are based on data from RTUs of SCADA systems. They only utilize rather limited data such as relay trip signals and circuit breaker status signals.

2. Protection System Performance Evaluation

Many fault section estimation solutions discussed in previous section are also able to identify failures and misoperations of protective relays and circuit breakers, which can be viewed as the overall performance evaluation for the protection systems. The principle is straightforward. After the exact fault section is figured out, the correct statuses of related protective relays and circuit breakers can be assumed and compared with the actual ones. However, the detailed evaluation of protection system performance can not be carried out because both the elaborated models of protection systems and the data reflecting detailed behaviors of protection systems are not used in those fault section estimation solutions.

In order to perform detailed evaluation of protection system performance, model-based approach is addressed in several papers [39–41]. However, the strategies discussed in these papers are quite different.

Reference [39] adopts consistency based reasoning mechanism. First, the correct behavior of each protection device and their interconnections are modeled. Then all the observed values are propagated through the modeled system, from inputs to outputs, and in the reverse direction. During the propagation, the output value of each device is predicted and the environment set which contains the path of devices employed to reach the predicted value is stored. If there are discrepancies within a set of predicted values for the output of a device, the union of the environment sets of the discrepancies creates a conflict set. Within a conflict set at least one device must be malfunctioned. Finally all the conflict sets are combined to produce candidate sets which show the possible combinations of malfunctioned devices. Reference [40] further proposes a temporal representation method and a toolset for reusing existing protection device models used to improve the solution discussed in [39]. How to couple the proposed temporal representation method into existing protection device models needs to be further investigated.

In [41], both the correct and faulty behavior of protective devices is specified by Augmented Reactive Model (ARM) and the timing constraints are represented by time intervals. Then a linear equation solver and a linear programming algorithm are employed to search for a set of transition paths which best justify the observed behavior. Thus the malfunctioned devices can be identified and diagnosed. The difficulty which such a method has to face is that all faulty behavior of protective devices must be defined and modeled in advance. Given the complexity under certain conditions, it may be impractical to classify and model all faulty modes.

3. Power System/Protection System Interactive Simulation

Previous research explored various options related to the software programs for modeling of power system networks and protective relays, and the schemes for interfacing the power system network models and the protective relay models [42]. They generally fall into three categories. The use of electromagnetic transients program (EMTP) for power system network modeling, and the transient analysis of control system (TACS) functions of EMTP for protective relay modeling is reported in the early literature [43]. Complied FORTRAN subroutine called from TACS in the EPRI/DCG version of EMTP is also used to develop protective relay models as reported in [44]. The MODELS language of the alternative transient program (ATP) version of EMTP, which is an enhancement to TACS, is employed for protective relay modeling as reported in [45, 46]. A prominent advantage of these approaches is the easy interfacing between the power system network models and the protective relay models because the TACS and MODELS are inherently embedded in EMTP/ATP [47–49].

A scheme which uses an "interaction buffer" for interfacing power system networks modeled by EMTP and protective relays modeled by MATLAB is described in [20]. Another method for establishing the link between EMTP and MATLAB is discussed in [50]. It is an interconnection where the internal computation engine of MATLAB is directly accessed by the FORTRAN code in EMTP. By these approaches, the high-level computation facilities of MATLAB can be utilized for protective relay modeling while the interconnection between the relay models and the power system network models is maintained.

An approach where power system network models are created in MATLAB/Power System Blockset and protective relay models are developed in MATLAB/SIMULINK is presented in [51]. The interfacing is easily achieved since both the power system network models and the protective relay models are under context of MATLAB/SIMULINK [52–54].

Despite the obvious advantages, the approaches discussed above have their inherent limitations. With respect to the first category of approaches, sophisticated relay models are difficult to be developed by TACS, MODELS and FORTRAN due to their limited flexibility and programmability. The "interaction buffer" and the programmed link discussed in the second category will cost excessive simulation time. They also cause the entire simulation program lack of integrity and portability. The problem of the third category lies in the slow simulation speed when the power system networks modeled by MATLAB/Power System Blockset are of large scale.

D. Proposed Approaches

1. Fault Section Estimation

Rule-based expert systems have demonstrated powerful reasoning capability in various diagnosis problems including fault section estimation. Their disadvantages lie in the burdensome procedure of rule base building and maintenance. It has been proved that Petri-nets can be translated into production rule systems [55]. So it is feasible to realize rule-based reasoning using Petri-nets formalism. Such Petri-nets for knowledge representation not only hold the strength of rule-based reasoning, but also overcome the disadvantages of conventional rule-based expert systems in that [12]:

1. Petri-nets' graphical nature allows one to visualize the structure of a rule-based system and make the models relatively simple and legible.

2. Petri-nets' mathematical foundation allows one to express the dynamic behavior of a system in algebraic forms.

In order to deal with uncertainty, fuzzy logic has been introduced into Petri-nets for knowledge representation to form Fuzzy Reasoning Petri-nets (FRPN) [12]. FRPN technique is quite well suited to deal with the complexities in the problem of fault section estimation because:

1. Multiple faults can be identified as the members of a candidate set by the virtue of fuzzy set theory.

2. False or uncertainty information can also be tackled by fuzziness of data.

3. Various backup protection operations due to unexpected operations of protection devices can be handled by the parallel reasoning capability.

4. The rule base and parameters are all represented in matrix forms and the whole reasoning process is implemented by matrix operations, which significantly facilitates the procedure of rule base building and maintenance.

In the study of the dissertation, FRPN diagnosis models for a 14-bus system based on relay trip signals and circuit breaker status signals acquired by RTUs of SCADA systems are formulated. As an improvement, the logic operand data of digital protective relays such as pickup and operation information of protection elements, which are more reliable than SCADA data, are used as additional inputs to the diagnosis models.

2. Protection System Performance Evaluation

The issues of protection system performance evaluation include acquiring data to observe the actual operations of the protection system, modeling the protection system to simulate its operations, employing a mechanism to identify unexpected operations and implementing a strategy to trace the reasons for unexpected operations. In the study of the dissertation, an automated protection system performance evaluation application has been developed to accomplish all the tasks.

The application automatically retrieves relay files upon their generation based on certain file transfer mechanism, processes relay file data through text parsing and signal processing techniques, and performs analysis by a rule-based expert system, in which forward chaining reasoning is used for prediction of expected protection operation while backward chaining reasoning is used for diagnosis of unexpected protection operations.

3. Power System/Protection System Interactive Simulation

In the study of the dissertation, a novel power system/protection system interactive simulation approach is proposed. In this approach, the power system network is modeled by the ATP program while the "compiled foreign model" mechanism of MODELS language is employed to model the digital protective relay in C++ language, which allows relay modeling in an "object-oriented" way as well as building a "seamless" interface between the power system network model and the relay model. An ATP/MinGW software package is used to facilitate the entire compilation and link process. A setting program is developed to facilitate the fault scenario setup, relay settings and user-defined error insertion. As a result, the enhanced relay model representation, the "seamless" interface between the power system network model and the relay model, and the easy scenario setup, make the overall interactive simulation more powerful and flexible.

E. Summary

The three problems to be solved by the dissertation study are discussed in this chapter. In the problem of fault section estimation, the complexities lie in the mix of multiple faults, failures of protection devices, and false data. In the problem of protection system performance evaluation, there are difficulties in building a proper model, identifying unexpected operations and tracing the causes of unexpected operations. In the problem of power system/protection system interactive simulation, challenges are the protection system modeling and the "seamless" interaction between the power system network models and the protection system models. Various approaches solving these three problems been presented in the literature. Their disadvantages limit their functionality and implementations. In the dissertation, by identifying their advantages over existing approaches, Fuzzy Reasoning Petri-nets, rule-based expert system and "compiled foreign model" mechanism are proposed as solutions to the problems of fault section estimation, protection system performance evaluation, and power system/protection system interactive simulation respectively.

CHAPTER IV

FAULT SECTION ESTIMATION

A. Introduction

This chapter discusses a Fuzzy Reasoning Petri-net (FRPN) approach to solve the problem of fault section estimation. First, the formal definition of FRPN is described and its algorithm is detailed [12]. Then the fault diagnosis models for a 14-bus system is developed based on FRPN formalism and SCADA data. Finally the fault diagnosis models based on both SCADA data and digital protective relay data as an improvement is further discussed.

B. Fuzzy Reasoning Petri-net Approach

1. Fuzzy Reasoning Petri-net Algorithm

A Fuzzy Reasoning Petri-net (FRPN) can be defined as an 8-tuple [12]:

$$(P, R, I, O, H, \theta, \gamma, C)$$

where

1. $P = \{p_1, p_2, ..., p_n\}$ is a finite set of places or called propositions.

2. $R = \{r_1, r_2, ..., r_m\}$ is a finite set of transitions or called rules.

3. $I : P \times R \longrightarrow \{0, 1\}$ is an $n \times m$ input matrix defining the directed arcs from propositions to rules. $I(p_i, r_j) = 1$, if there is a directed arc from p_i to r_j, and $I(p_i, r_j) = 0$, if there is no directed arcs from p_i to r_j, for $i = 1, 2, ..., n$, and $j = 1, 2, ..., m$.

4. $O : P \times R \longrightarrow \{0,1\}$ is an $n \times m$ output matrix defining the directed arcs from rules to propositions. $O(p_i, r_j) = 1$, if there is a directed arc from r_j to p_i, and $O(p_i, r_j) = 0$, if there is no directed arcs from r_j to p_i, for $i = 1, 2, ..., n$, and $j = 1, 2, ..., m$.

5. $H : P \times R \longrightarrow \{0,1\}$ is an $n \times m$ matrix defining the complementary arcs from propositions to rules. $H(p_i, r_j) = 1$, if there is a complementary arc from p_i to r_j, and $H(p_i, r_j) = 0$, if there is no complementary arcs from p_i to r_j, for $i = 1, 2, ..., n$, and $j = 1, 2, ..., m$.

6. θ is a truth degree vector. $\theta = (\theta_1, \theta_2, ..., \theta_n)^T$, where $\theta_i \in [0,1]$ means the truth degree of p_i, $i = 1, 2, ..., n$. The initial truth degree vector is denoted by θ^0.

7. $\gamma : P \longrightarrow \{0,1\}$ is a marking vector. $\gamma = (\gamma_1, \gamma_2, ..., \gamma_n)^T$. $\gamma_i = 1$, if there is a token in p_i, and $\gamma_i = 0$, if p_i is not marked. An initial marking is denoted by γ^0.

8. $C = diag\{c_1, c_2, ..., c_m\}$. c_j is the confidence of r_j, $j = 1, 2, ..., m$.

The 5-tuple (P, R, I, O, H) is the basic FRPN structure that defines a directed graph. The updates of the truth degree vector θ through the firing of a set of rules describe the dynamic reasoning process of the modeled system. If the truth degree of a proposition is known at a certain reasoning step, a token is assigned to the corresponding proposition, which is associated with the value between 0 and 1. The token is represented by a dot. When a proposition p_i has no token, which means that the truth degree is unknown at that step, $\theta_i = 0$. Hence, $\theta_i = 0$ implies two possible situations: 1) the absence of token, which means truth degree of proposition p_i, is unknown; 2) a token with zero value, which means that the truth degree of proposition p_i is known and equals zero. Marking vector γ can be used to distinguish the two situations.

In order to represent the execution rules of FRPN formally, some operators are used.

1. $\bigoplus$: $\mathbf{A} \bigoplus \mathbf{B} = \mathbf{D}$, where $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{D}$ are all $m \times n$-dimensional matrices, such that $d_{ij} = max\{a_{ij}, b_{ij}\}$.

2. $\bigotimes$: $\mathbf{A} \bigotimes \mathbf{B} = \mathbf{D}$, where $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{D}$ are $(m \times p)$, $(p \times n)$, $(m \times n)$-dimensional matrices respectively, such that $d_{ij} = max_{1 \leq k \leq p}(a_{ik} \cdot b_{kj})$.

3. $\cdot *$: $\mathbf{A} \cdot * \mathbf{B} = \mathbf{D}$, where $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{D}$ are all $m \times n$-dimensional matrices, such that $d_{ij} = a_{ij} \cdot b_{ij}$.

Similar to an ordinary Petri-nets, the execution rules of a FRPN include enabling and firing rules.

1. A rule $r_j \in R$ is enabled if and only if p_i is marked, or $\gamma_i = 1$, $\forall p_i \in \{$input propositions of $r_j\}$.

2. Enabled at marking γ, r_j firing results in a new γ'

   $\gamma'(p) = \gamma(p) \bigoplus O(p, r_j), \qquad \forall p \in P.$

   The truth degree vector changes from θ to θ'

   $\theta'(p) = \theta(p) \bigoplus c_j \cdot \rho_j \cdot O(p, r_j), \qquad \forall p_i \in P$

   where

   $\rho_j = \sum_{p_i \in \dot{r_j}} x_i w_i$

   where

   $\dot{r_j} = \{p_i | I(p_i, r_j) = 1 or H(p_i, r_j) = 1, p_i \in P\}$

   and

$x_i = \theta_i$ if $I(p_i, r_j) = 1$; $x_i = 1 - \theta_i$ if $H(p_i, r_j) = 1$; w_i is the weight of p_i regarding the rule r_j.

3. All the enabled rules can fire at the same time. A firing vector μ is introduced such that $\mu_j = 1$ if r_j fires. After firing a set of rules, the marking and truth degree vectors of the FRPN become

$$\gamma' = \gamma \oplus [O \otimes \mu] \tag{4.1}$$

$$\theta' = \theta \oplus [(O \cdot C) \otimes \rho] \tag{4.2}$$

where

$\rho = [\rho_1, \rho_2, ..., \rho_m]^T$, which is called control vector. $\mu : T \longrightarrow \{0, 1\}$ is the firing vector. $\mu = (\mu_1, \mu_2, ..., \mu_m)^T$.

From Eq. 4.1 and Eq. 4.2, we notice that as long as μ and ρ are known, the next step marking and truth degree vectors can be derived from the current values. To obtain μ, an 'neg' operator is used. μ^k can be calculated as follows:

$neg\gamma^k = 1_m - \gamma^k = \overline{\gamma^k}$

$neg\theta^k = 1_m - \theta^k = \overline{\theta^k}$

$$\mu^k = \overline{(I + H)^T \otimes \overline{\gamma^k}} \tag{4.3}$$

$$\rho_k = ((I^T . * W^T) \cdot \theta^k + (H^T . * W^T) \cdot \overline{\theta^k}) \cdot *\mu^k \tag{4.4}$$

where

$1_m = (1, 1, ..., 1)^T$, k is the kth reasoning step, neg θ^k is an n-dimensional vector. Its components express the confidence of proposition p_i being false at the kth reasoning step, $i = 1, 2, ..., n$. γ^k is the marking. μ_k is an m-dimensional firing vector. $\mu_k = 1$, if r_j is enabled, and $\mu_k = 0$, if r_j is not enabled, $j = 1, 2, ..., m$. W is the weight matrix. ρ^k is an m-dimensional control vector at the kth reasoning step. Its components express the truth degrees of enabled rule r_j's preconditions. $\rho_k = 0$, if rule r_j is not enabled.

From Eq. 4.1 and Eq. 4.3:

$$\gamma^{k+1} = \gamma^k \oplus [O \otimes \overline{(I+H)^T \otimes \overline{\gamma^k}}] \tag{4.5}$$

From Eq. 4.2, Eq. 4.3 and Eq. 4.4:

$$\theta^{k+1} = \theta^k \oplus [(O \cdot C) \otimes (((I^T . * W^T) \cdot \theta^k + (H^T . * W^T) \cdot \overline{\theta^k}) \cdot * \mu^k)] \tag{4.6}$$

To summarize, the FRPN algorithm can be described as follows:

1. Read initial inputs I,O,H,C,γ^0, and θ^0.

2. Let $k = 0$.

3. Compute γ^{k+1} from γ^k according to Eq. 4.5; Compute θ^{k+1} from θ^k according to Eq. 4.6.

4. If $\theta^{k+1} \neq \theta^k$ or $\gamma^{k+1} \neq \gamma^k$, let $k = k + 1$, and return to Step 3; Otherwise, the reasoning is over.

2. Fault Section Estimation Model

We will focus on the fault section estimation problem on a 14-bus system as shown in Fig. 13. The system consists of 34 sections, including 14 buses and 20 transmission lines. The buses are denoted as Bnn, where nn is a two-digit number ranging from 01 to 14. The transmission lines are denoted as Lnnmm, where nn and mm are the two-digit numbers of the two buses connected by the transition line and nn is always smaller than mm. The protection system of the 14-bus system consists of

174 protection devices, including 40 circuit breakers, 40 main transmission line relays, 40 primary backup transmission line relays and 40 secondary backup transmission line relays and 14 bus relays. Only the 40 circuit breakers are shown in Fig. 13. They are all installed on the two ends of the 20 transmission lines. The 40 circuit breakers are denoted as CBnnmm, where nn is the two-digit number of the bus where the circuit breaker is located and mm is the two-digit number of the bus at the remote end of the transmission line. The 40 main transmission line relays are denoted MLRnnmm, where nn is the two-digit number of the bus where the relay is located and mm is the two-digit number of the bus at the remote end of the transmission line. The 40 primary backup transmission line relays are associated with the 40 main transmission line relays respectively. They are denoted as BLRnnmm, where nn and mm have the same meaning as those of the main relays. The 40 secondary backup transmission line relays are associated with the 40 main transmission line relays respectively. They are denoted as SLRnnmm, where nn and mm have the same meaning as those of the main relays. The 14 bus relays are denoted as BRnn, where nn is the two-digit number of the bus protected by the relay.

A bus relay protects its associated bus. It will operate to trip all the circuit breakers connected to the bus if a fault occurs on the bus. A main transmission line relay has forward protection zone and protects the whole transmission line. It will operate to trip its associated circuit breaker to clear a fault on the transmission line. A primary backup transmission line relay is the local backup of the main transmission line relay and has the same protection zone as that of the main relay. If the fault clearance by the main transmission line relay fails, the primary backup transmission line relay will operate to trip its associated circuit breaker to clear the fault. A secondary backup transmission line relay is the remote backup of the main and primary backup transmission line relays just beyond the remote end of the transmission line.

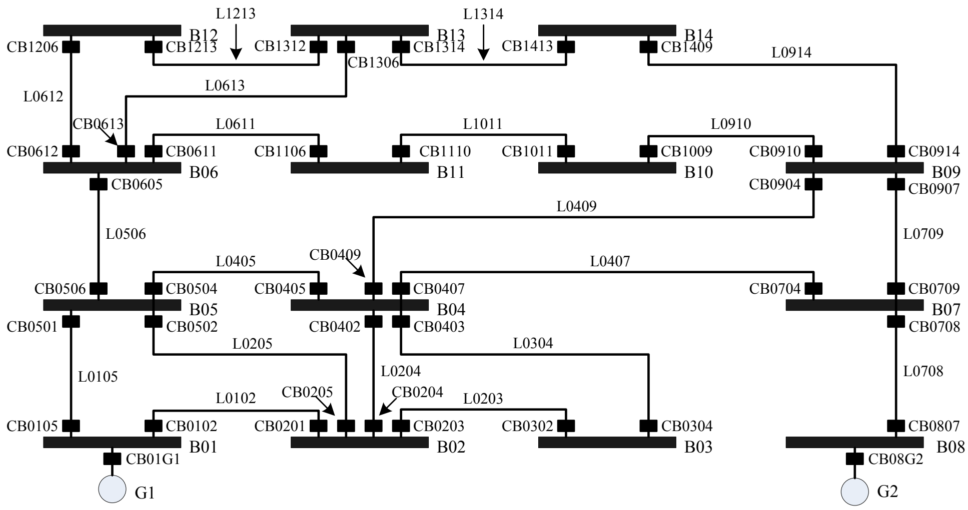

Fig. 13. A 14-bus power system model

If the fault clearance by both the main and the primary backup transmission line relays just beyond the remote end of the transmission line fails, it will operate to trip its associated circuit breaker to clear the remote transmission line fault. A secondary backup transmission line relay is also the remote backup of the bus relay at the remote end of the transmission line. If the fault clearance by the bus relay at the remote end of the transmission line fails, it will operate to trip its associated circuit breaker to clear the remote bus fault.

For each section of the 14-bus system, a FRPN model will be built. The FRPN model establishes the reasoning from the observed statuses of protection devices to a faulted section based on the protection rules associated with the particular section. There are two categories of models: 1) for transmission lines and 2) for buses. As examples, the FRPN model for the transmission line L1314 is shown in Fig. 14 and the FRPN model for the bus B13 is shown in Fig. 15.

In Fig. 14, the places p_1, p_2, ..., p_{12} represent the input propositions, which are the operations of protection devices associated with the transmission line L1314.

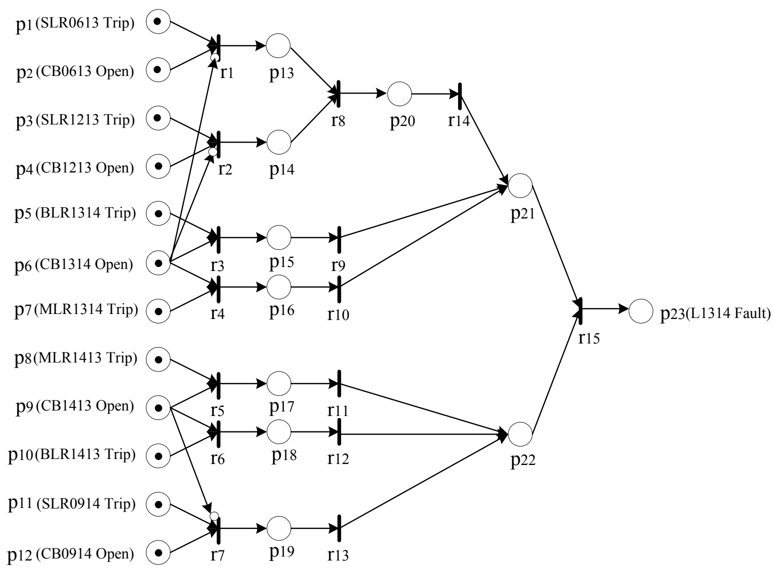

Fig. 14. A FRPN model for L1314 fault based on SCADA data

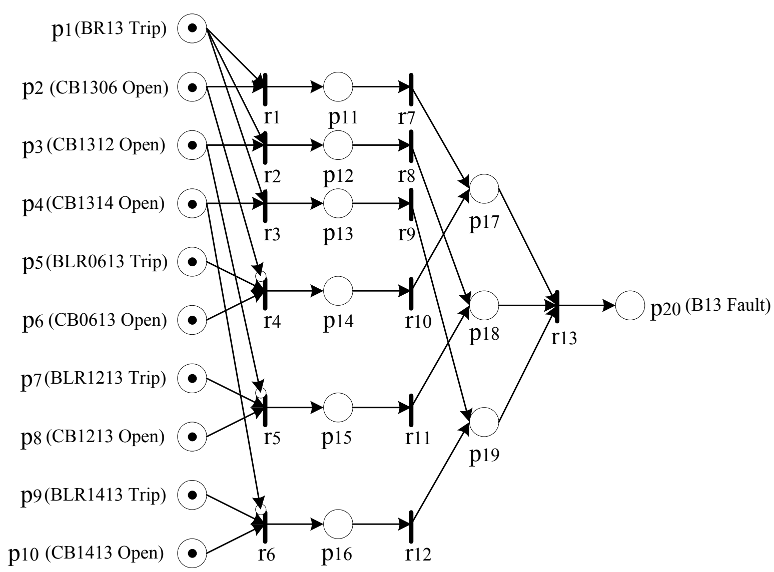

Fig. 15. A FRPN model for B13 fault based on SCADA data

For example, p_5 represents the proposition "BLR1314 trips" and p_6 represents the proposition "CB1314 opens". Initially all of these places contain a token, which means that the truth degrees of these propositions are known. Each such proposition will be assigned a truth degree value describing the certainty of observation of the operation of the protection device. Under such an assumption, if the operation of a protection device is actually observed, the proposition will have a truth degree value θ_i which is bigger than 0. On the contrary, if the operation of the protection device is not observed, the proposition will have a 0 truth degree value. θ_i can be given by experience based on the reliability of the indication mechanism of the protection device, the measurement channel for the protection device and the data communication system for the protection device. In this example, θi will be given the same value of 0.9.

The places p_{13}, p_{14}, ..., p_{22} represent the propositions which are intermediate reasoning results. For example, p_{15} represents the proposition "main protection of the transmission line L1314 at the bus B13 end operates for a fault on the transmission line L1314". p_{22} represents the proposition "protection of the transmission line L1314 at the bus B14 end operates for a fault on the transmission line L1314". The place p_{23} represents the output proposition "a fault exists on the transmission line L1314".

The transitions r_1, r_2, ..., r_{15} represent rules in which antecedent propositions implicate consequent propositions. Each rule r_j is associated with a certainty factor c_j, which describes the confidence level of the rule. $c_j, j = 1, 2, ..., 7$ can be given by experience based on the reliability of relays. Usually a main relay has higher reliability than that of a primary backup relay. A primary backup relay has higher reliability than that of a second backup relay. In this example, c_1, c_2, c_3, c_4, c_5, c_6, c_7 will be given the values 0.7, 0.7, 0.8, 0.9, 0.9, 0.8, 0.7 respectively. $c_j, j = 8, 9, ..., 15$ will be given the same value 1.0.

It should be mentioned that from p_6 to r_1 and from p_6 to r_2, there are two complementary arcs, which means that if the opening of the circuit breaker CB1314 is observed, the operation of the corresponding secondary backup protection should be discredited. On the contrary, if the opening of the circuit breaker CB1314 is not observed, the operation of the corresponding secondary backup protection should be credited. Similarly, the complementary arc from p_9 to r_7 have the same meaning.

Regarding each rule, each of its antecedent propositions is given a weight, which stands for the relative significance of the antecedent proposition in implicating the consequent propositions. For example, regarding the rule r_1, the proposition p_1 "SLR0613 Trip" will be given a weight 0.4; the proposition p_2 "CB0613 Trip" will be given a weight 0.3; the absence of the proposition p_6 "CB1314 Open" will be given a weight 0.3.

According to the discussion in previous section, the matrical representation of the FRPN model can be given as follows:

As an example, when a fault occurs on the transmission line L1314, its associated protection system operated to respond to the fault. The following signals are observed in SCADA data: SLR0613 Trip, CB0613 Open, SLR1213 Trip, CB1213 Open, BLR1314 Trip, MLR1314 Trip, MLR1413 Trip and CB1413 Open. γ^0 and θ^0 are given as:

$\gamma^0 = [\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ ]^T$

$\theta^0 = [\ 0.9\ 0.9\ 0.9\ 0.9\ 0.9\ 0\ 0.9\ 0.9\ 0.9\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ ]^T$

The first reasoning step will result in

$\gamma^1 = [\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ ]^T$

$\theta^1 = [\ 0.9\ 0.9\ 0.9\ 0.9\ 0.9\ 0\ 0.9\ 0.9\ 0.9\ 0\ 0\ 0\ 0.791\ 0.791\ 0.324\ 0.342\ 0.855\ 0.486\ 0.026\ 0\ 0\ 0\ 0\ ]^T$

The second reasoning step will result in

$$
I = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\qquad
O = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
$$

$$
H = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\qquad
C = \begin{bmatrix}
0.85 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.85 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.9 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.95 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.95 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0.9 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0.85 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
$$

$$
W = \begin{bmatrix}
.4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
.3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & .4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & .3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & .4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
.3 & .3 & .6 & .6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & .4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & .4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & .6 & .6 & .3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & .4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & .4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & .3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & .5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & .5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & .5 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & .5 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

$\gamma^2 = [\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 0\ ]^T$

$\theta^2 = [\ 0.9\ 0.9\ 0.9\ 0.9\ 0.9\ 0\ 0.9\ 0.9\ 0.9\ 0\ 0\ 0\ 0.791\ 0.791\ 0.324\ 0.342\ 0.855\ 0.486\ 0.026$
$0.791\ 0.342\ 0.855\ 0\ ]^T$

The third reasoning step will result in

$\gamma^3 = [\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ ]^T$

$\theta^3 = [\ 0.9\ 0.9\ 0.9\ 0.9\ 0.9\ 0\ 0.9\ 0.9\ 0.9\ 0\ 0\ 0\ 0.791\ 0.791\ 0.324\ 0.342\ 0.855\ 0.486\ 0.026$
$0.791\ 0.791\ 0.855\ 0.599\ ]^T$

The final reasoning step will result in

$\gamma^4 = [\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ ]^T$

$\theta^4 = [\ 0.9\ 0.9\ 0.9\ 0.9\ 0.9\ 0\ 0.9\ 0.9\ 0.9\ 0\ 0\ 0\ 0.791\ 0.791\ 0.324\ 0.342\ 0.855\ 0.486\ 0.026$
$0.791\ 0.791\ 0.855\ 0.823\ ]^T$

So the conclusion will be that a fault occured on the transmission line L1314 with a truth degree value 0.823.

In the same example as above, if MLR1413 Trip is missing in the SCADA data due to data transmission error, the conclusion will be that a fault occured on the transmission line L1314 with a truth degree value 0.652.

For each section of the 14-bus system, a FRPN model is built. So there are totally 34 FRPN models. SCADA data are input into these models. The conclusion of a fault occurrence on a section has a truth degree value. The conclusion with the highest truth degree value is the final conclusion in case of single fault. In case of multiple faults, the several conclusions with the highest truth degree values which are greater than a threshold are the final conclusions.

3. Improvement Based on Digital Protective Relay Data

When a digital protective relay responds to a power system fault, a protection operation chain will be formed to trip the circuit breaker to interrupt the fault currents. Along the chain, the fault is the initial cause, and the pickup and operation of relay elements, the relay trip and the circuit breaker opening are the consequent effects. In a fault section estimation problem, all the effects can be used to infer the cause. We have discussed the use of relay trip signal and circuit breaker opening status signal in SCADA data for fault section estimation. In this section we will discusses the use of pickup and operation information of relay elements.

In a digital relay, the pickup and operation information of relay elements is usually in the form of logic operands. These logic operands are in essence digital bits and are usually observed in two ways. First, they can be directly transmitted in the form of register values via a digital communication system based on a certain communication protocol. Second, they are contained in relay files such as the event report and the oscillography file and these files can be transmitted via a digital communication system based on a certain file transfer protocol.

The pickup and operation information of relay elements can be used to improve the fault section estimation based on the relay trip signal and circuit breaker opening status signal from SCADA data. The justification is as follows:

1. Along the protection operation chain, the pickup and operation of relay elements are more directly related to the fault than the relay trip and circuit breaker opening because the fault directly triggers the pickup and operation of relay elements and the relay trip and the circuit breaker opening are further effects of the pickup and operation of relay elements. The latter effects may be influenced more by uncertain factors. For example, even if a relay element successfully picks up and operates to respond to a fault, the relay trip contact may fail to assert due to electrical or mechanical problems; the circuit breaker may also fail to open due to electrical or mechanical problems. Obviously, it is desirable to use information with less uncertainty.

2. In a multifunctional digital relay, several protection elements may pick up and operate to respond to the same fault. For example, the neutral instantaneous over-current element and the ground distance element may pick up and operate to respond to a ground fault. Meanwhile, a logic operand representing the pickup or operation of a protection element may exist in several relay files. In case of communication errors which may generate false data, the redundancy makes the relay pickup and operation information more reliable than the trip signal and circuit breaker status signal measured by RTUs of SCADA systems.

3. A logic operand is originally in a digital form. If they are transmitted in the form of register values or files through digital communication systems. They do not need to be measured. On the contrary, the relay trip contact signal and the circuit breaker status signal are originally analog signals. they are measured

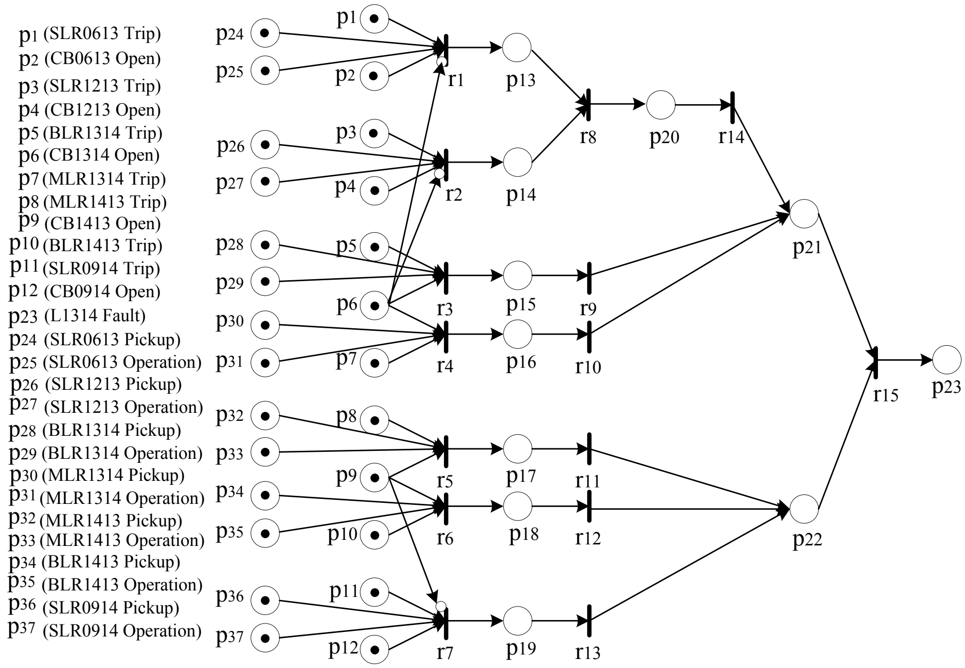

Fig. 16. A FRPN model for L1314 fault based on SCADA and digital protective relay data

and converted to digital signals by RTUs of SCADA systems. Measurement errors may exist, which may cause false data. Furthermore, the circuit breaker status signal is generated by complex electrical and mechanical mechanism. A problem in such a mechanism may also cause false indication of the circuit breaker status.

Fig. 16 illustrates how the pickup and operation information is added into the FRPN model built for diagnosing a fault on the transmission line L1314.

The matrical representation of the FRPN model described by Fig. 16 can be easily generated based on the matrical representation of the FRPN model described by Fig. 14. The following are the updated matrices I, O, H, W. There is no change on matrix C. The weight assignment in W is adjusted to reflect the relative significance of input signals in determination of the occurrence of a protection operation. The operation of relay element has the largest weight and the pickup of relay element has the second largest weight. The relay trip and the circuit breaker opening have smaller weights. When the absence of the circuit breaker opening for the main protection and primary backup protection is taken into consideration of the secondary backup protection, it has the largest weight.

We take the same example given in previous section. when a fault occurs on the transmission line L1314, its associated protection system operated to respond to the fault. In addition to the observed SCADA data, the following relay signals are also observed: SLR0613 Pickup, SLR0613 Operation, SLR1213 Pickup, SLR1213 Operation, BLR1314 Pickup, BLR1314 Operation, MLR1314 Pickup, MLR1314 Operation, MLR1413 Pickup, MLR1413 Operation, BLR1413 Pickup, SLR0914 Pickup. Since the relay data are more reliable than the SCADA data, they are given a larger truth value 0.98. γ^0 and θ^0 are given as:

$$\gamma^0 = [\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ ]^T$$

$$\theta^0 = [\ 0.9\ 0.9\ 0.9\ 0.9\ 0.9\ 0\ 0.9\ 0.9\ 0.9\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0.98\ 0.98\ 0.98\ 0.98\ 0.98\ 0.98\ 0.98\ 0.98\ 0.98\ 0.98\ 0.98\ 0\ 0.98\ 0\ ]^T$$

The final conclusion will be that a fault occurs on the transmission line L1314 with a truth degree value 0.848.

In the same example as above, if MLR1413 Trip is missing in the SCADA data due to data transmission error while MLR1413 Pickup and MLR1413 Operation are observed, the conclusion will be that a fault occurs on the transmission line L1314 with a truth degree value 0.827.

$$
I = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\qquad
O = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

$$
H = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\qquad
W = \begin{bmatrix}
.05 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
.3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & .05 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & .3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & .05 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
.3 & .3 & .6 & .6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & .05 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & .05 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & .6 & .6 & .3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & .05 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & .05 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & .3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & .5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & .5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & .5 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & .5 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
.15 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
.2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & .15 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & .2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & .15 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & .2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & .15 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & .2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & .15 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & .2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & .15 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & .2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & .15 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & .2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

C. Summary

A Fuzzy Reasoning Petri-nets (FRPN) approach to solve the problem of fault section estimation is discussed in this chapter. In this approach, the fuzzy reasoning from protection system status data to faulted power system sections is formulated by Petri-nets. The reasoning process can be graphically represented in a form of Petri-nets and implemented by matrix operations. Data acquired by RTUs of SCADA systems, including relay trip signals and circuit breaker status signals, are the inputs to the diagnosis models. The logic operand data of digital protective relays such as pickup and operation information of protection elements are more reliable than the SCADA data to reflect relay trip status. They can be utilized as additional inputs to the diagnosis models based on SCADA data and the required matrices representing the new diagnosis models can be easily generated by augmenting and modifying the original matrices.

CHAPTER V

PROTECTION SYSTEM PERFORMANCE EVALUATION

A. Introduction

This chapter discusses a rule-based reasoning approach to solve the problem of protection system performance evaluation. First, the overall structure of an automated protection system performance evaluation application which are based on the Rule-based reasoning technique is described. Then each module of the application is further detailed.

B. Rule-based Reasoning Approach

1. Overall Structure

The overall structure of the Automated Protection System Performance Evaluation Application is represented in Fig. 17. The application consists of a relay file retrieval module, a relay file data processing module, and a protection operation validation and diagnosis module. The relay file retrieval module communicates to relays to check if any new relay files are generated. If new relay files are generated, the module will automatically download them to specified destinations. Then the relay file data processing module processes incoming relay files so that relay data can be converted into initial facts used by the protection operation validation and diagnosis module. The expert system based protection operation validation and diagnosis module analyzes relay data to validate correct protection operations and diagnose the reasons for unexpected protection operations. As a result, an analysis report is generated to serve relevant users.

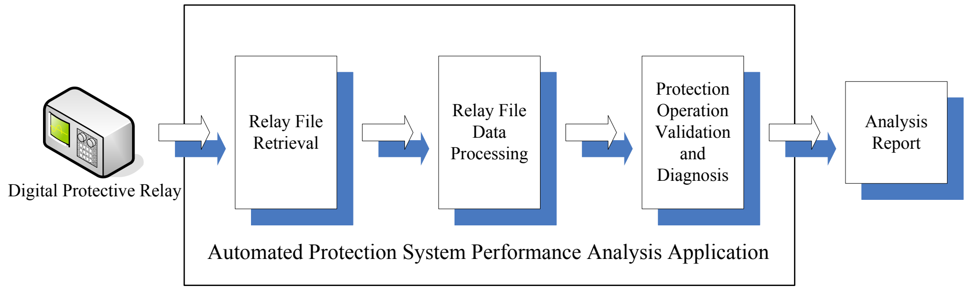

Fig. 17. Functional structure of automated protection system performance evaluation application

2. Automated Relay File Retrieval

Many digital relay vendors provide PC software programs which can communicate to their relays for relay settings, file retrieval, and real-time data view. For example, GE has EnerVista UR Setup software for its UR series relays and SEL has AcSELerator software for most of its relay series [14, 15]. To retrieve relay files, users need to manually run those software programs and initiate the file retrieval process. This obviously renders difficulty in automated data retrieval and analysis. In order to solve the problem, a relay file retrieval module is developed to automatically retrieve relay files upon their generation without user intervention, which makes the automated data analysis possible.

Typically, digital relays support two modes of automated file retrieval mechanism, namely polling and report by exception. In the polling mode, a software program initiates communication to a relay at certain time interval to check if there are new files generated. If new files are identified, they will be downloaded by the software program based on certain file transfer protocol. If there are no new files, the communication will be halted till the next checking due time. The report by exception mode means that upon the triggering of new event, a relay initiates communication to notify a software program of the new event. Then the software program will download the new files based on certain file transfer protocol. Both of the two modes have advantages and disadvantages. In a polling mode, since new files are checked repeatedly at certain time interval, if a file transfer process fails due to certain error, it is still possible to get those files in the next file transfer process. This makes the risk of missing files small. In a report by exception mode, if the one-time new event notification is not captured by the software program or the file transfer process fails due to certain error, there is no remedy to get the new files. A polling mode occupies more computer time than a report by exception mode because a polling mode repeats check routines, while a report by exception mode just initiates communication upon new event trigger. A polling mode also has slower response than a report by exception mode. In a polling mode the file transfer process has to be initiated at scheduled due time which may lag the new event trigger time, while in a report by exception mode the file transfer process is initiated immediately after the new event trigger. Table III summaries the advantages and disadvantages of the two modes.

The Automated Relay File Retrieval module is capable of automatically retrieving files from GE's UR series relays and SEL's SEL421 relays. The module is embedded in the application as a library. For each relay there is a master sub-module. A configuration file is created to supply configuration information for those sub-modules. Fig. 18 illustrates the functional structure.

Each relay master sub-module requires information about relay identification,

Table III. Comparison of digital relay file retrieval modes

Criteria	**Polling**	**Report by Exception**
Reliability	Higher	Lower
Computing Load	More	Less
Response Time	Slower	Faster

communication port settings, file retrieval settings and last recorded event number. The relay identification specifies relay ID and manufacturer information. The communication port settings specify communication parameters such as communication port number and baud rate. The file retrieval settings specify parameters related to file transfer such as file retrieval mode and polling time interval. The last recorded event number is compared with current event number in a relay to decide if a new event is recorded by the relay. This number is updated when the file retrieval is done for each new event.

GE UR series relays support Modicon Modbus RTU protocol over their RS232 or RS485 serial links. Appendix A details the UR relay file transfer mechanism and gives the program flow chart for UR relay file retrieval in polling mode.

SEL421 relays support SEL ASCII Command protocol over their RS232 serial links. Several SEL ASCII commands are involved in initiation of a file transfer from the relay to external software. Once a file transfer is initiated, the Ymodem protocol is used to perform the file transfer process. Appendix B details the SEL421 relay file transfer mechanism and gives the program flow charts for SEL421 relay file retrieval in polling mode and report by exception mode.

3. Relay File Data Processing

The relay file data processing module performs three functions. First, it parses text information to extract performance specification data, setting data, fault data, and event record data. Second, by applying Discrete Fourier Transform (DFT), it extracts fundamental frequency phasors from analog oscillography data to determine the exact fault inception time and fault clearance time, which are a portion of fault data. Third, it determines the status changes of digital oscillography data and converts them into event record data. Finally all the data are converted into CLIPS

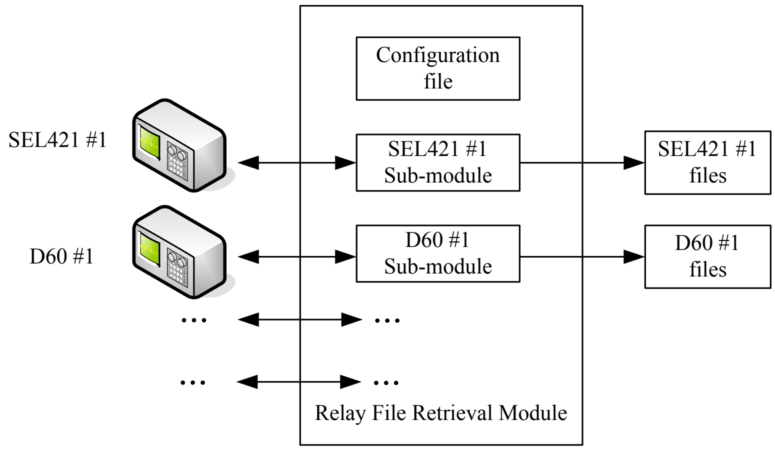

Fig. 18. Functional structure of automated relay file retrieval module

expert system fact format. Fig. 19 illustrates the functional structure.

Fig. 20 is the program flow diagram for calculation of fault inception and clearance time. It is assumed that the phase current phasor amplitudes calculated at the sample index number which is 5% of the total sample number are normal state phase current phasor amplitudes. If any phase current phasor amplitude calculated at certain sample index number is greater than 1.2 times of its corresponding normal state phase current phasor amplitude, fault inception is detected and the fault inception time is recorded. If all the phase current phasor amplitudes are smaller than 0.1 times of their corresponding normal state phase current phasor amplitudes, fault clearance is detected and the fault clearance time is recorded. Fig. 21 is the program flow diagram for determination of status changes of digital oscillography data.

4. Rule-based Reasoning

a. Problem Domain

For the evaluation of protection system performance, three levels of diagnosis problems are involved. First, the expected statuses and timings of operation of el-

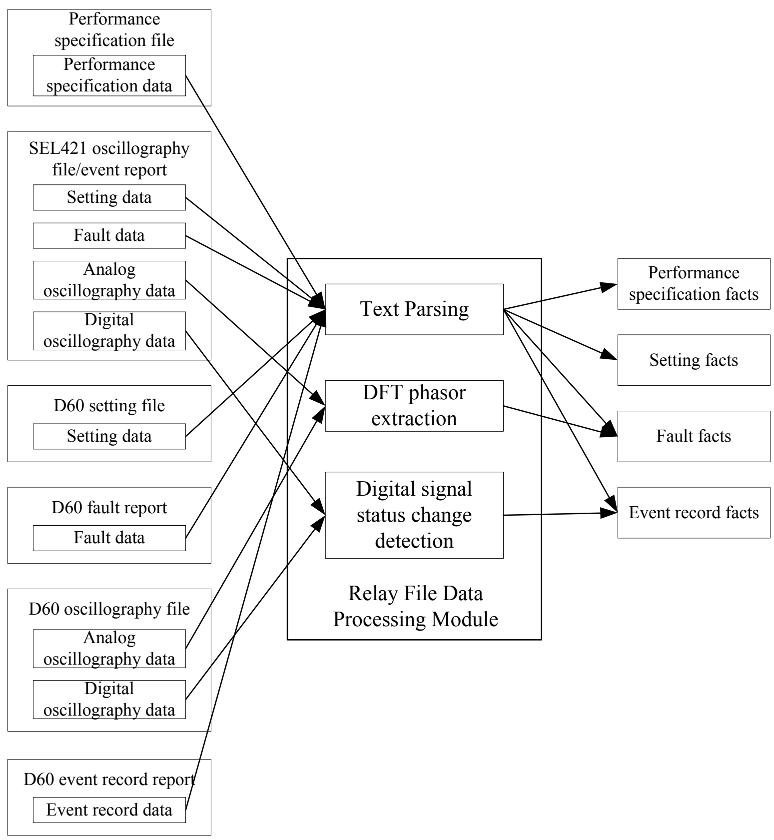

Fig. 19. Functional structure of file data processing module

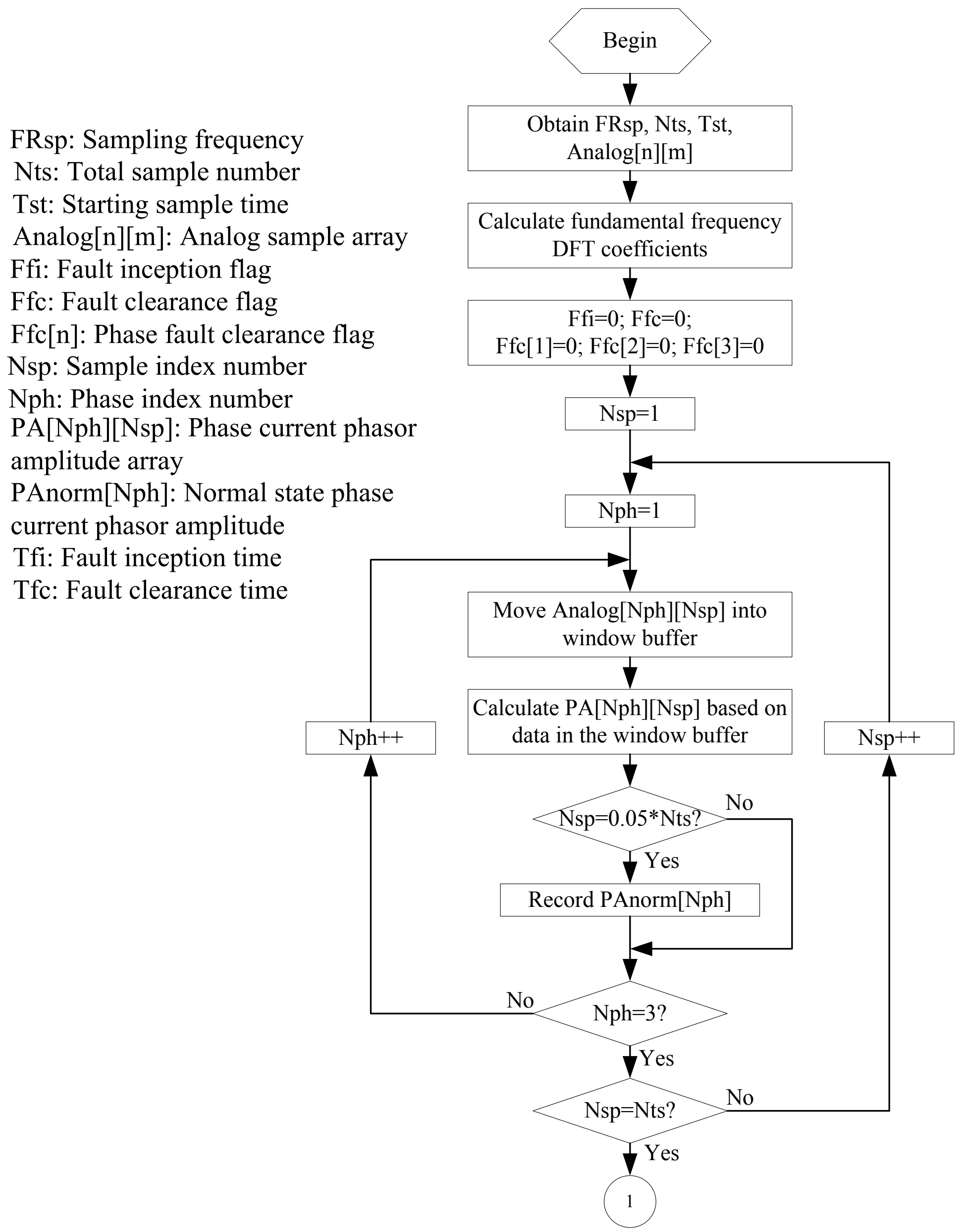

Fig. 20. Program flow diagram for calculation of fault inception and clearance time

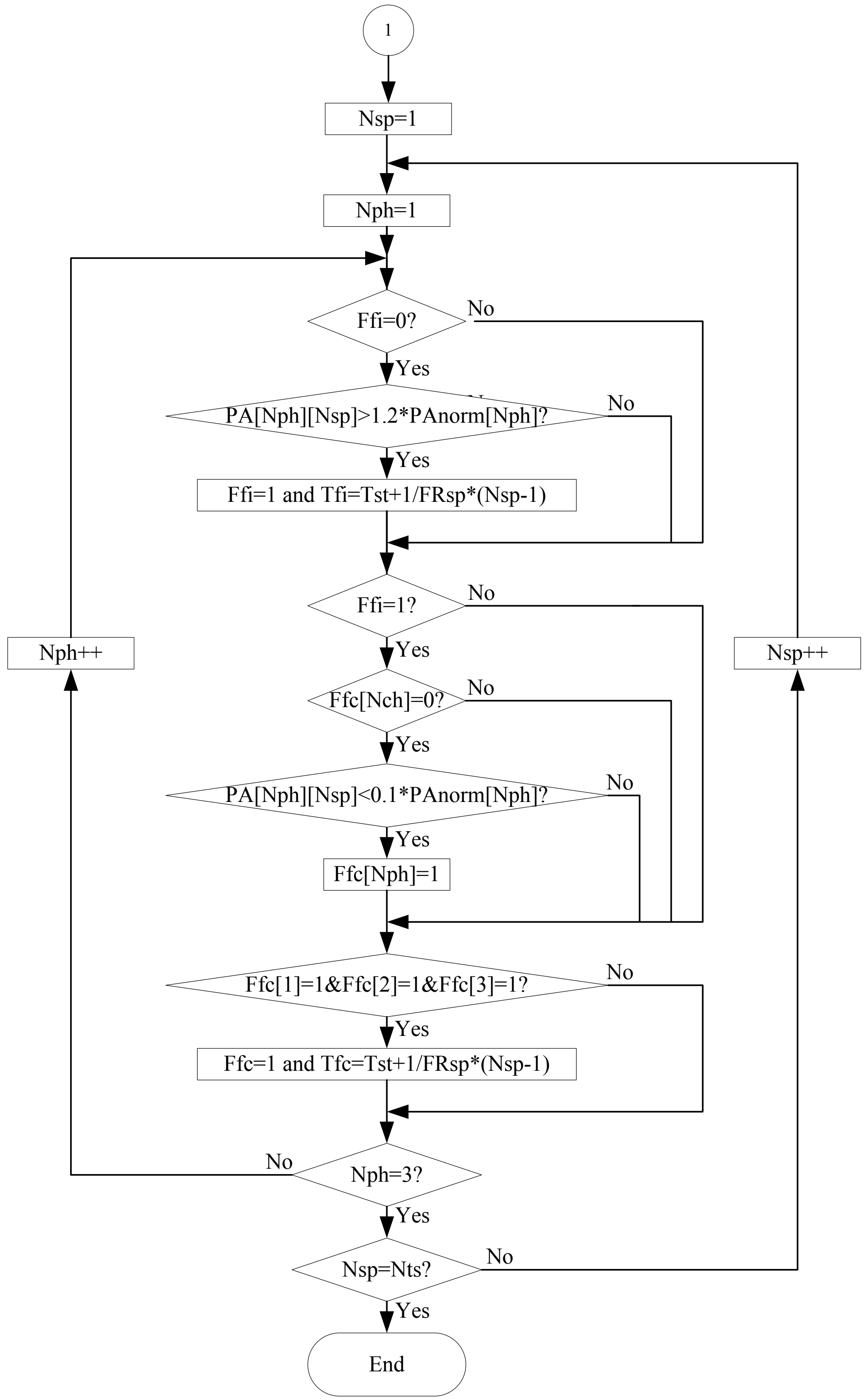

Fig. 20 Continued

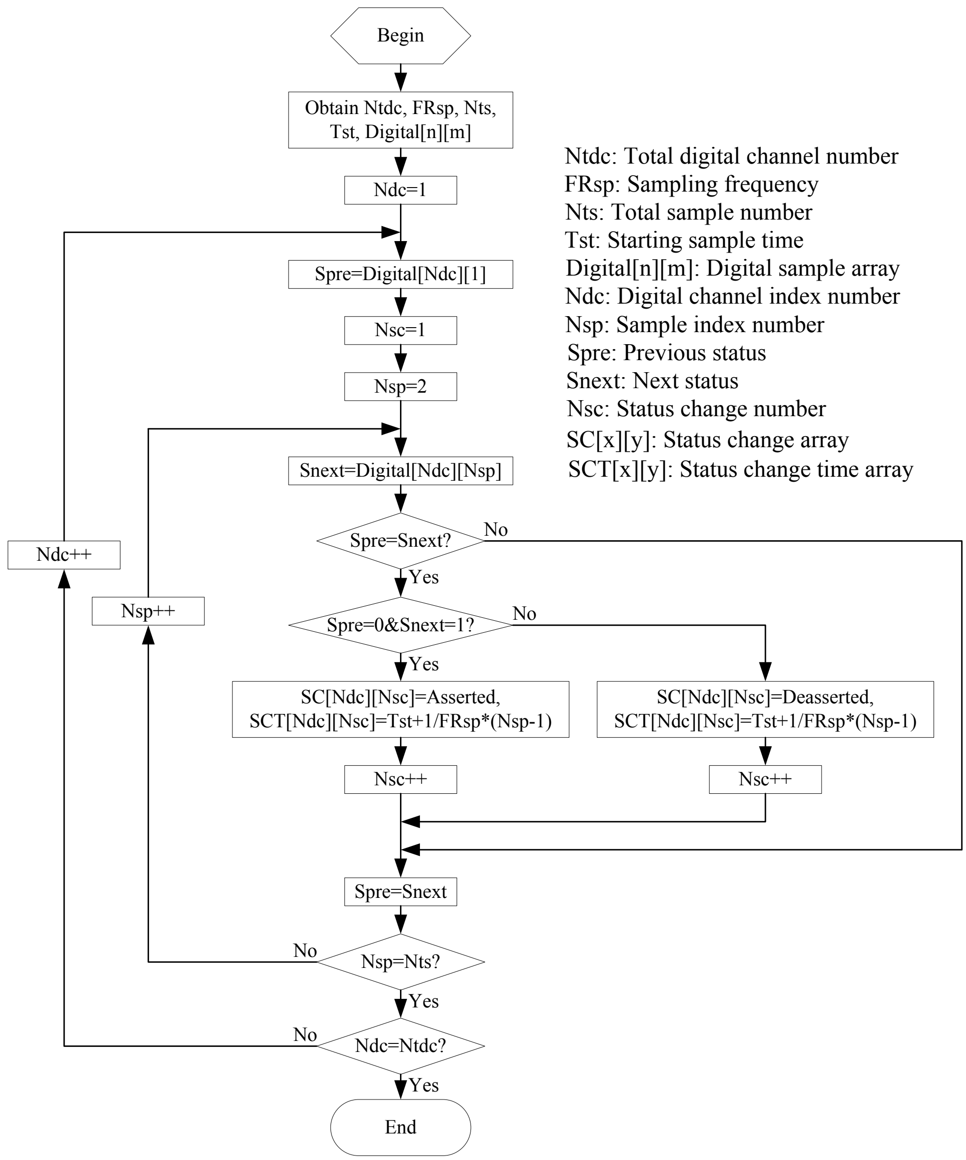

Fig. 21. Program flow diagram for determination of status changes of digital oscillography data

ements should be simulated. Second, all the unexpected statuses and timings of operation of elements should be identified. Third, a symptom of unexpected status or timing of an element may be caused by a malfunction of its directly related logic component. It may also be caused by a malfunction of unrelated logic components, because abnormities can propagate through the protection operation chain to cause several occurrences of symptoms. The malfunctioned logic components should be traced out by analyzing the relations of these symptoms.

The first problem can be solved by building a protection operation logic model which generates expected statuses and timings of elements. The second problem can be solved by comparing the expected statuses and timings of elements with the actual ones. To solve the third problem, an efficient way is to reason from the effect side to the cause side in cause-effect relations. In terms of the protection operation chain, a reasoning mechanism is needed to traverse from its higher level down to its lower level.

The rule base built to solve the three levels of problems consists of three modules called Expected Protection Operation Prediction, Protection Operation Validation and Symptom Diagnosis respectively. Fig. 22 illustrates the conceptual strategy of diagnosis reasoning. The expected protection operation is predicted by the Expected Protection Operation Prediction Module. Inputs to the module are performance specification facts, relay setting facts and fault facts. Within this module, the expected statuses and timings of active logic operands are inferred. The results are regarded as hypothesis of protection operation. Event record facts obtained from the relay file processing module are the actual statuses and timings of logic operands. With both hypothesis and facts of protection operation as inputs, the Protection Operation Validation Module performs validation of the correctness of statuses and timings of logic operands based on hypothesis-fact matching. All the inconsistencies of expected

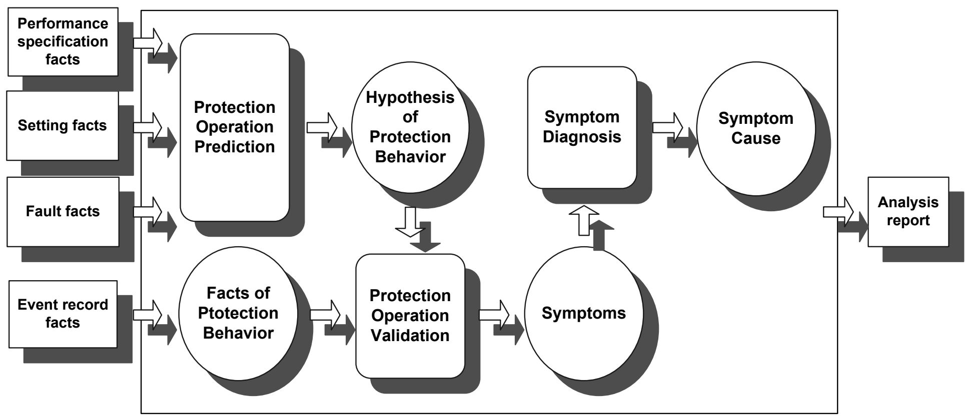

Fig. 22. Conceptual strategy of reasoning

and actual statuses as well as timings of logic operands are regarded as symptoms. The Symptom Diagnosis Module takes symptoms as inputs, trace the ultimate cause of a symptom. An analysis report will be generated which contains the results from all the three modules.

b. Expected Protection Operation Prediction

In order to predict the expected statuses and timings of elements, forward chaining reasoning is employed to simulate the protection operation chain. Forward chaining reasoning is also called bottom-up reasoning. It reasons from lower level facts to top level conclusion. The protection operation chain fits into this concept. The disturbance information, relay settings and performance specifications are lower level facts. Rules are written to simulate the transition of statuses of elements. The inferred statuses and corresponding timings, combined with relay settings and performance specifications are the inputs to the next transition. Thus, the whole protection operation chain can be simulated until the final conclusion is reached, which reads as "Fault currents are interrupted by the circuit breaker at time T". Fig. 23 illustrates the forward chaining reasoning for prediction of protection operation, which only details the operation of a phase distance element.

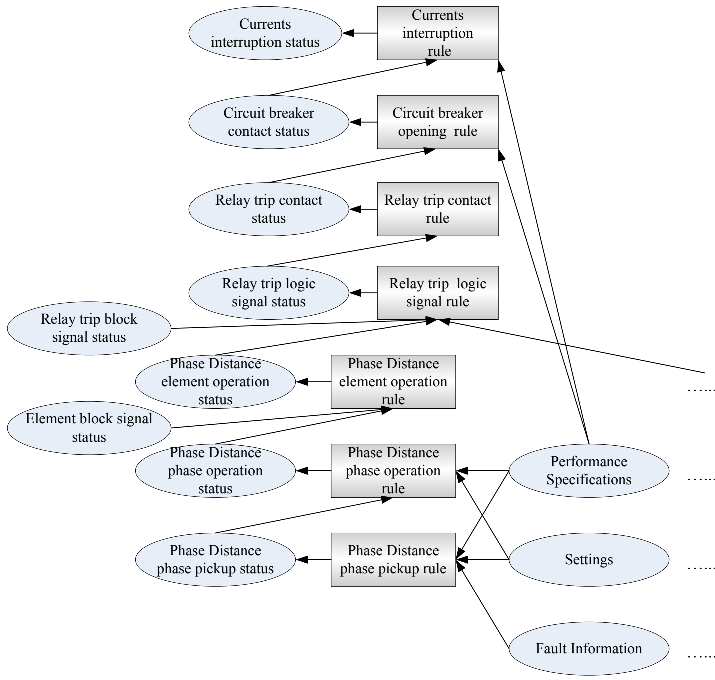

Fig. 23. Forward chaining reasoning for prediction of protection operation

c. Protection Operation Validation

Validation of Protection Operation is based on comparison of predicted statuses and timings of elements with the actual ones. The status comparison is based on the existence and non-existence of predicted status and actual status of an operand. The predicted status is regarded as a hypothesis and the actual status is regarded as a fact. If both the hypothesis and the fact exist, the correctness of the status is validated. If the hypothesis exists and the fact does not exist or the hypothesis does not exist but the fact exists, a symptom will be identified. Fig. 24 illustrates the reasoning process for the lower three levels of the protection operation chain. The reasoning process for other levels of the protection operation chain is similar. Fig.4 aims to deal with such symptoms: A status of a logic operand should have existed but it does not exist. There is also a counterpart of the reasoning process, which aims to deal with such symptoms: A status of a logic operand should have not existed but

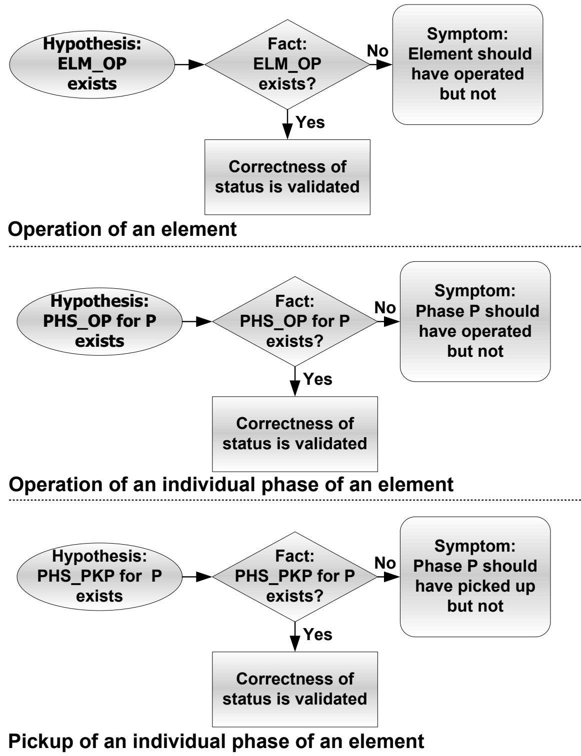

Fig. 24. Reasoning process for validation of status of logic operands

it exists.

The operating speed of protection elements and the associated circuit breaker is evaluated by examining the timing of operands. Fig. 25 shows the logic for evaluating the operating speed of protection elements. The logic for evaluating the operating speed of the circuit breaker is similar.

d. Diagnosis of Symptoms

A symptom of unexpected status or timing may be caused by a malfunction of its directly related logic component or by a malfunction of logic components at lower level of protection operation chain due to the propagation of abnormality. Backward chaining reasoning is employed to trace out the malfunctioned logic components. Backward chaining reasoning is also called top-down reasoning. Along the reasoning

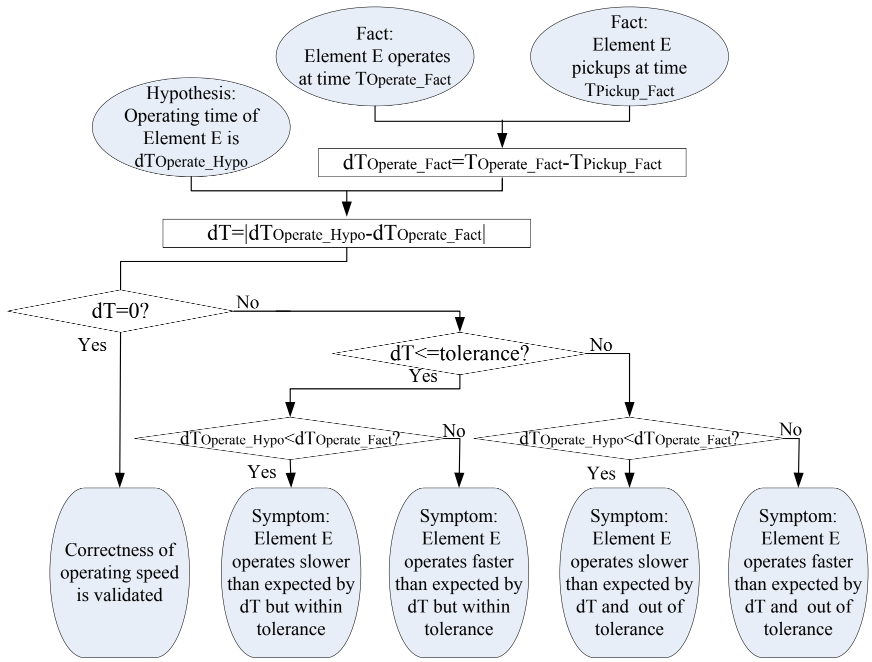

Fig. 25. Logic reasoning for evaluating operating speed of protection elements

chain, in order to prove higher level hypotheses, the intermediate hypotheses must be proven. Thus the reasoning will trace the basic facts to prove the hypotheses. This mechanism is quite suitable for a diagnosis problem.

In the context of our problem domain, the backward reasoning chain is defined in terms of a goal which can be accomplished by satisfying sub-goals. We use Fig. 26 and Table IV to explain the reasoning process. Suppose the symptom "Circuit breaker currents interruption failed" is identified, finding the reason for this symptom will be set as the initial goal (Goal 1). Then the existence of the symptom "Circuit breaker failed to open" will be tested. If it does not exist, it proves that the contact signal indicated circuit breaker opening but in fact the circuit breaker did not interrupt the fault currents. Obviously the diagnosis will be "Circuit breaker malfunctioned". But if the symptom "Circuit breaker failed to open" exists, it proves that the circuit breaker failed to interrupt fault currents because the circuit breaker failed to open. A

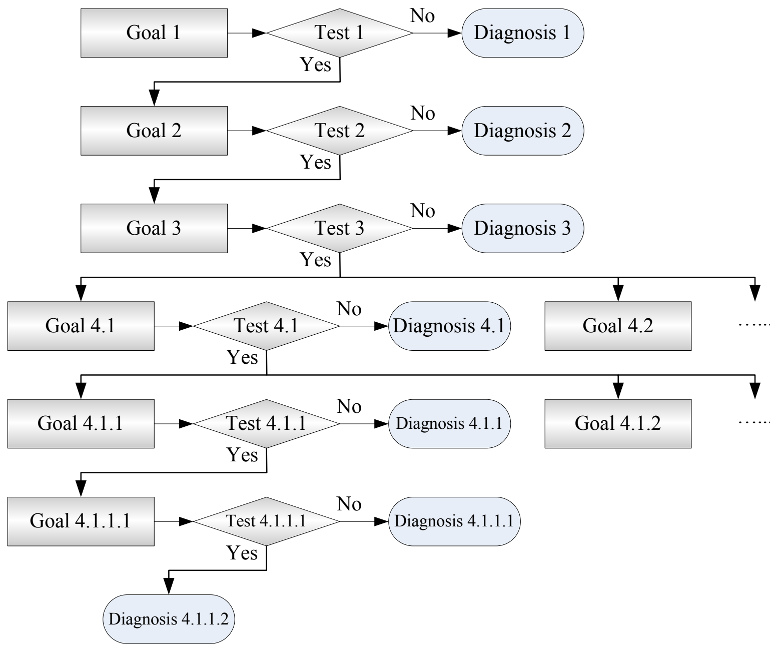

Fig. 26. Backward chaining reasoning for diagnosis of symptoms

sub-goal (Goal 2) will be created to find the reason for the symptom "Circuit breaker failed to open". Following this pattern, the second or more sub-goals will be created and the malfunctioned logic components will be finally traced. It should be noticed that because the relay trip can be triggered by the operation of any enabled protection element, if the symptom "Relay failed to trip" is identified, sub-goals (Goal 4.1, Goal 4.2,) for diagnosis of operation of each enabled element may be created at the same time. Likewise, sub-goals (Goal 4.1.1, Goal 4.1.2,) may be created for diagnosis of operation of individual phase of an enabled protection element.

C. Summary

An automated protection system performance evaluation application which is based on rule-based reasoning technique is presented in this chapter. The application consists of a relay file retrieval module, a relay file data processing module, and a rule-

Table IV. Explanation of backward reasoning process

Goal	Description
Goal 1	To find the reason for the symptom "Fault currents interruption failed"
Goal 2	To find the reason for the symptom "Circuit breaker failed to open"
Goal 3	To find the reason for the symptom "Relay failed to trip"
Goal 4.1	To find the reason for the symptom "Phase Distance Element (Zone 1, Zone 2,) failed to operate"
Goal 4.2	To find the reason for the symptom "Phase IOC Element failed to operate"
Goal 4.1.1	To find the reason for the symptom "Phase A-B of Phase Distance Element (Zone 1, Zone 2,) failed to operate"
Goal 4.1.2	To find the reason for the symptom "Phase B-C of Phase Distance Element (Zone 1, Zone 2,) failed to operate"
Goal 4.1.1.1	To find the reason for the symptom "Phase A-B of Phase Distance Element (Zone 1, Zone 2,) failed to pickup"
Goal 4.1.1.2	To find the reason for the symptom "Over-current supervision of Phase A-B of Phase Distance Element (Zone 1, Zone 2,) failed"
Test	**Description**
Test 1	Does the symptom "Circuit breaker failed to open" exist?
Test 2	Does the symptom "Relay failed to trip" exist?
Test 3	Does the symptom "All the protection elements expected to operate failed to operate" exist?
Test 4.1	Does the symptom "All the phases of Phase Distance Element (Zone 1, Zone 2,) expected to operate failed to operate" exist?
Test 4.1.1	Does the symptom "Phase A-B of Phase Distance Element (Zone 1, Zone 2,) failed to pickup" exist?
Test 4.1.1.1	Did the symptom "Over-current supervision of Phase A-B of Phase Distance Element (Zone 1, Zone 2,) failed" exist?
Diagnosis	**Description**
Diagnosis 1	Circuit breaker malfunctioned
Diagnosis 2	Wire connection between the relay and the circuit breaker is broken
Diagnosis 3	Logic component for Relay Trip malfunctioned
Diagnosis 4.1	Logic component for Operation of Phase Distance Element (Zone 1, Zone 2,) malfunctioned
Diagnosis 4.1.1	Logic component for Operation of Phase A-B of Phase Distance Element (Zone 1, Zone 2,) malfunctioned
Diagnosis 4.1.1.1	Logic component for Pickup of Phase A-B of Phase Distance Element (Zone 1, Zone 2,) malfunctioned
Diagnosis 4.1.1.2	Logic component for Over-current Supervision of Phase A-B of Phase Distance Element (Zone 1, Zone 2,) malfunctioned

based protection operation validation and diagnosis module. The relay file retrieval module utilize file transfer mechanisms of digital protective relays to automatically retrieve relay files. The data and information contained in these files are processed and converted into proper format in the relay file data processing module. In the rule-based protection operation validation and diagnosis module, forward chaining reasoning is used for prediction of expected protection operation while backward chaining reasoning is used for diagnosis of unexpected protection operations.

CHAPTER VI

DEVELOPMENT OF POWER SYSTEM/PROTECTION SYSTEM INTERACTIVE SIMULATION ENVIRONMENT

A. Introduction

This chapter discusses a "compiled foreign model" approach to solve the problem of power system/protection system interactive simulation [56]. First, the "compiled foreign model" mechanism of ATP MODELS language is introduced. Then a generic digital protective relay model which is based on the "compiled foreign model" mechanism is described. The programming structure which enables reuse of the generic relay model is further presented. Finally the implementation of the generic relay model is described with emphasis on the solution of the "seamless" interfacing between the relay model and the power system model.

B. ATP Compiled Foreign Model Approach

1. Compiled Foreign Model of ATP MODELS Language

MODELS language is a general-purpose description language of the ATP program [48, 49]. It provides a format which focuses on description of the structure of a model, and the function of its elements. Compared with high-level programming languages such as C/C++, its flexibility and programmability are relatively limited. To overcome the disadvantage, MODELS provides a "compiled foreign model" mechanism to expand its flexibility and programmability. This mechanism can be utilized for modeling a protective relay in high-level languages, and interfacing the relay model with the power system model.

MODELS provides a pre-defined interface to link procedure called a "foreign model" which is written in other programming languages to the ATP simulation program [49]. The interface is defined as four arrays carrying the values of data, input, output and history variables. Each "foreign model" should provide both an execution procedure and an initiation procedure corresponding to the EXEC procedure and INIT procedure of a model defined in MODELS. A "foreign model" must be compiled and linked to the ATP simulation program before it can be called by MODELS. An interface routine in a FORTRAN file called "formod.for" is where the user registers the correspondence between the identification name used in the "foreign model" declaration in MODELS, and the actual name of the procedure in the "foreign model". Once declared and named, a "foreign model" can be used independently in as many separate uses as required. The inputs and outputs of the "foreign model", along with the directives controlling its simulation, are specified in a regular USE statement in MODELS.

The newly developed ATP/MinGW program package has convenient tools to compile a "foreign model" written in FORTRAN and C/C++, and link it with the ATP simulation program [57]. The Minimalist GNU for Windows (MinGW) is a compiler package for windows operating system [58]. In the ATP/MinGW program package, the source code of the ATP program is compiled by the FORTRAN compiler and C compiler to generate object files. The compilers are also used to compile the user-supplied source code of a "foreign model" written in FORTRAN or C/C++ to generate its object file. Then all the object files and libraries are linked together to produce a new executable ATP program, which takes the ATP data case file as input to run the simulation. Fig. 27 illustrates the whole make process which includes the compilation and linking. It should be mentioned that the users can easily complete the make process in dialogs in the ATP/MinGW program package.

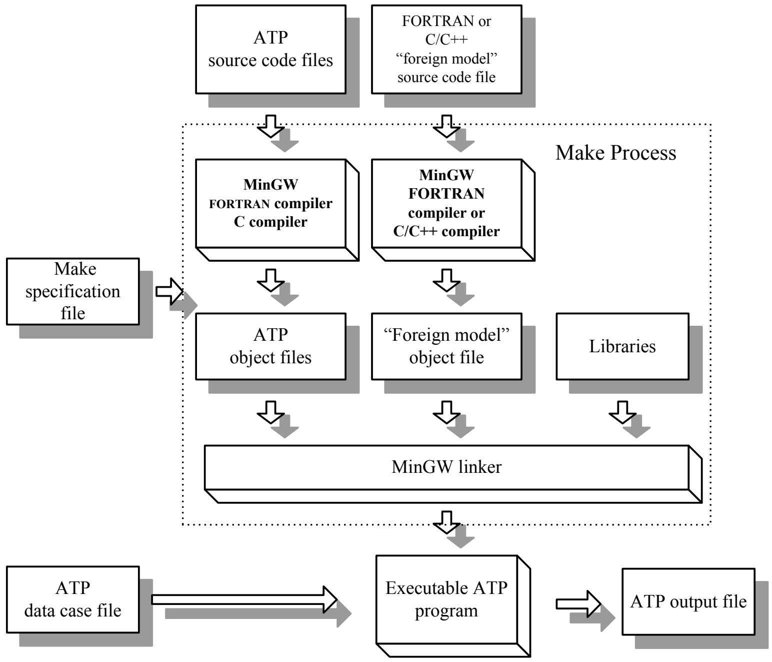

Fig. 27. Make process in the ATP/MinGW program package

2. Generic Digital Protective Relay Model

a. Main Features of a Generic Relay Model

Protective relays at different locations in a power system may have different inputs, outputs, sampling rates and settings, but they usually have similar design architectures. In order to efficiently realize the interaction between the power system network model and each relay model associated with a specific location, our strategy is to employ the "compiled foreign model" mechanism to build a generic digital protective relay model as a "foreign model" and reuse the model with different configuration of inputs, outputs, sampling rates and settings.

The generic relay model not only satisfies the common functional requirements for components, interface, and protection functions, but also is capable of inserting user-defined errors and generating event reports. Table V lists the main features of the relay model.

Table V. Main features of the generic relay model

Requirements	**Features**
Components	analog filter, A/D converter, implementation of protection algorithms
Interface	15 channels of node voltages and branch currents inputs, 1 channel of trip signal output
Protection Functions	phase distance, ground distance, differential
Others	user-defined error insertion, setting file reading, generation of event reports

b. Programming Structure of the Relay Model

By virtue of the “compiled foreign model” mechanism, the advanced features of C++ language such as object-oriented concepts, direct access to windows libraries, and powerful file I/O capability can be utilized to model the relay. It is possible to realize all the functions of the relay model in the C++ “foreign model”. However, since the MODELS language itself has some unique features which facilitate modeling of some components of the relay, we adopt a hybrid approach to realize the relay functions in both the MODELS section of ATP data case file and the C++ “foreign model”. The interfacing to power system network model, analog signal filtering, and the A/D conversion is implemented in the MODELS section, while all other functions of the relay model are realized in the C++ “foreign model”. In order to reuse the “foreign model” with different configuration of inputs, outputs and sampling rates while applying the common analog signal filtering to all reused “foreign models”, we employ an “inheritance” modeling architecture. A model named “RLY” which represents a generic relay model is declared in the MODELS section. In the execution procedure of “RLY”, the analog filtering function is defined, which is followed by the definitions of use of “RLY” with different inputs, outputs and sampling rates. Fig. 28 illustrates the programming structure of the relay model.

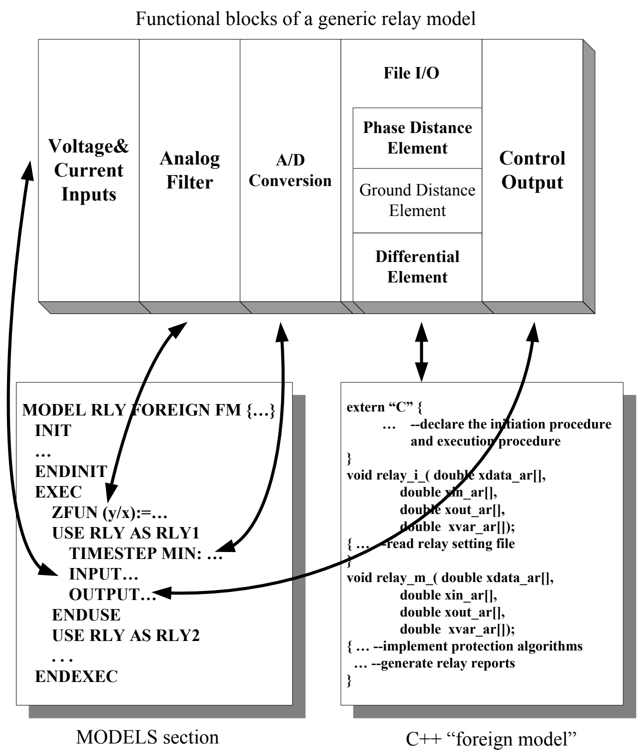

Fig. 28. Programming structure of the relay model

c. Implementation of the Relay Model

1. Interface to the Power System Network Model: The inputs from the power system network model are three phase voltages measured at bus nodes and three phase currents measured through circuit breaker switches. The outputs of the power system network model are control variables of the control nodes of circuit breaker switches. The names of these nodes and switches are declared in the INPUT and OUTPUT directives of the MODELS section. In the USE statement of each of the reused models, the inputs and outputs associated with a specific relay location are defined by particular names of bus nodes, switches and control nodes. A distance transmission line relay model has three phase voltages and three phase currents as inputs. A differential bus relay has three phase currents of each transmission line connecting to the bus bar as inputs.

2. Analog Filtering: In order to meet the sampling theorem, the sampling rate of the relay model should be twice the maximum frequency of the input analog signals. Sampling with a lower sampling rate will result in errors due to the aliasing effect in the frequency domain. The anti-aliasing filters, which in practice are analog filters, should be used to minimize such aliasing effect as well as attenuate the high frequency components. In the relay model, analog second order Butterworth low-pass filter is employed. From a modeling point of view, such a filter can be represented by the Z-plane digital transfer function, which can be easily realized by the Z-transform transfer function of MODELS language.

3. A/D conversion: The sample and hold circuit of A/D converters is realized by the TIMESTEP MIN "time step" directive in the USE statement of the generic relay model. This will actually perform the decimation in the original simulation time step at the rate of the specified time step.

4. Protection Algorithms: All the protection algorithms are implemented as the C++ "foreign model". For a distance transmission line relay model, Fourier Transform is used to extract the fundamental frequency phasors for phase voltages and currents, line voltages and currents, and zero sequence currents. The phasors for line voltages and currents are used to calculate the line impedances for comparison with the MHO characteristic of the Phase Distance Elements. The phasors for phase voltages and currents, and zero sequence currents are used to calculate the phase impedances for comparison with the quadrilateral characteristic of the Ground Distance Elements. For a differential bus relay model, the instantaneous phase currents of each transmission line connecting to the bus bar are summed to compare with a predefined threshold to detect the

Table VI. User-defined error types of the relay model

Device	Zone	Status	Error Type
Distance relay	Zone 1, Zone 2	Pickup, operation, dropout, trip	failure, data missing
Bus relay	N/A	Pickup, operation, dropout, trip	failure, data missing
Circuit breaker	N/A	open	failure, data missing

occurrence of a bus fault. For both types of relay models, timers are simulated to ensure the required time coordination between the pickup and operation of protection elements.

5. Relay File Generation: In the relay model, the digital signals representing pickup and operation of protection elements are stored in the arrays. The status changes of digital signals are detected and used for event report generation. At the end of the simulation, the file I/O functions of C++ are employed to generate the time-stamped event reports.

6. Relay Setting Reading: In the initiation procedure, settings for all instantiated relay models are read from a relay setting file by using the file I/O functions of C++. Since the settings are not hard-coded, they can be easily changed from case to case, which facilitates studies involving a large amount of cases.

7. Error Insertion: In the initiation procedure, code numbers for all kinds of user-defined errors are read from an error code file by using the file I/O functions of C++, which also facilitates easy setup of scenarios. The types of errors are listed in Table VI.

C. Summary

A “compiled foreign model” approach to the problem of power system/protection system interactive simulation is discussed in this chapter. The “compiled foreign model” mechanism of ATP MODELS language provides convenient method for modeling a sophisticated digital protective relay using C++, interfacing the relay model and the power system network model, and reusing such a relay model. The flexibility of C++ language greatly facilitates the interfacing. Its file I/O capability is quite useful for relay setting reading, relay file generation, and user-defined error insertion.

CHAPTER VII

CASE STUDY

A. Introduction

This chapter presents the case study of the proposed Fuzzy Reasoning Petri-net approach for fault section estimation and protection system performance evaluation application. Test environment and scenarios are described. Test results are presented and discussed.

B. Fault Section Estimation

1. Test Environment

The test environment used for the case study is the 14-bus system shown in Fig. 13 in Chapter IV. Fig. 29 only shows its power system model created by an ATPDraw program. The whole power system/protection system interactive simulation environment is developed by the approach described in Chapter VI.

2. Test Cases and Results

a. Case 1

The scenario of Case 1 is described in Table VII. The observed SCADA data are listed in Table VIII. The observed relay data are listed in Table IX.

Based on the SCADA data in Table VIII, the only candidate for the fault section is estimated as the transmission line L0910, with a truth degree value 0.855. Based on both the SCADA data in Table VIII and relay data in Table IX, the only candidate for the fault section is estimated as the transmission line L0910, with truth degree value 0.882.

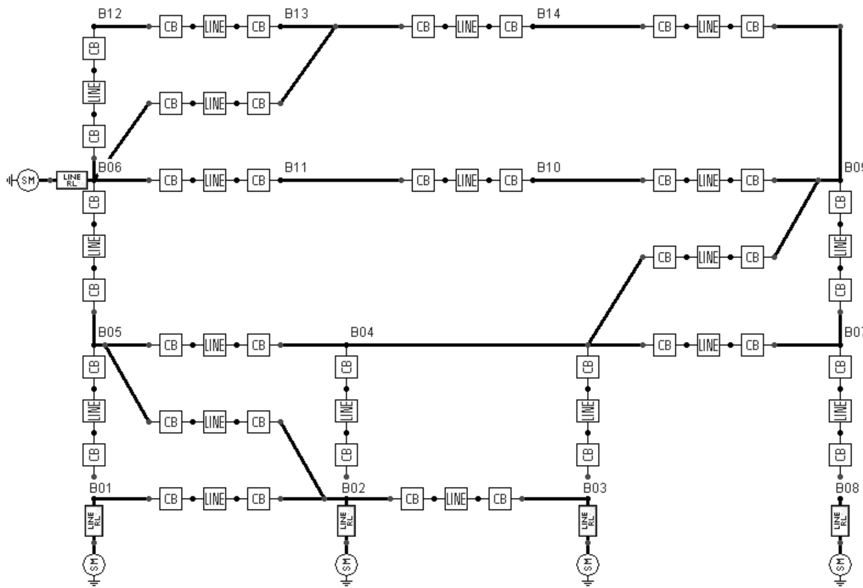

Fig. 29. 14-bus power system model created by an ATPDraw program

Table VII. Scenario description of Case 1 for the fault section estimation study

Scenario	**Description**
Power System Fault	A permanent fault occurred on the transmission line L0910 at 0.05 second
Protection Device Failure	No protection device failed
False Data	No false data occurred

b. Case 2

The scenario of Case 2 is described in Table X. The observed SCADA data are listed in Table XI. The observed relay data are listed in Table XII.

Based on the SCADA data in Table XI, the candidates for the fault section are estimated and results are listed in Table XIII. Based on both the SCADA data in Table XI and relay data in Table XII, the candidates for the fault section are estimated and the results are listed in Table XIV.

c. Case 3

The scenario of Case 3 is described in Table XV. The observed SCADA data are

Table VIII. SCADA data of Case 1 for the fault section estimation study

Sequence No.	Time Stamp (Sec)	Observed Signal
1	0.1000	MLR0910 TRIP
2	0.1000	MLR1009 TRIP
3	0.2000	CB0910 OPEN
4	0.2000	CB1009 OPEN

Table IX. Relay data of Case 1 for the fault section estimation study

Sequence No.	Time Stamp (Sec)	Observed Signal
1	0.0662	SLR0409 PKP
2	0.0677	SLR0709 PKP
3	0.0693	BLR0910 PKP
4	0.0698	MLR0910 PKP
5	0.0703	MLR1009 PKP
6	0.0703	BLR1009 PKP
7	0.0703	SLR1110 PKP
8	0.0724	SLR1409 PKP
9	0.0740	MLR0910 OP
10	0.0745	MLR1009 OP

listed in Table XVI. The observed relay data are listed in Table XVII.

Based on the SCADA data in Table XVI, the candidates for the fault section are estimated and results are listed in Table XVIII. Based on both the SCADA data in Table XVI and relay data in Table XVII, the candidates of the fault sections are estimated and the results are listed in Table XIX.

Table X. Scenario description of Case 2 for the fault section estimation study

Scenario	Description
Power System Fault	A permanent fault occurred on the bus B04 at 0.05 second. A second permanent fault occurred on the bus B09 at 0.09 second.
Protection Device Failure	No protection device failed
False Data	No false data occurred

Table XI. SCADA data of Case 2 for the fault section estimation study

Sequence No.	Time Stamp (Sec)	Observed Signal
1	0.1000	BR04 TRIP
2	0.2000	CB0402 OPEN
3	0.2000	CB0403 OPEN
4	0.2000	CB0405 OPEN
5	0.2000	CB0407 OPEN
6	0.2000	CB0409 OPEN
7	0.2000	BR09 TRIP
8	0.2000	CB0904 OPEN
9	0.2000	CB0907 OPEN
10	0.2000	CB0910 OPEN
11	0.2000	CB0914 OPEN

Table XII. Relay data of Case 2 for the fault section estimation study

Sequence No.	Time Stamp (Sec)	Observed Signal
1	0.0537	BR04 PKP
2	0.0625	SLR0304 PKP
3	0.0651	SLR0904 PKP
4	0.0667	SLR0204 PKP
5	0.0667	SLR0504 PKP
6	0.0677	SLR0704 PKP
7	0.0703	BLR0704 PKP
8	0.0703	BLR0904 PKP
9	0.0766	BLR0204 PKP
10	0.0766	BLR0504 PKP
11	0.0771	BLR0304 PKP
12	0.0938	BR09 PKP
13	0.0964	SLR0709 PKP
14	0.1000	BR04 OP
15	0.1063	BLR0709 PKP
16	0.1115	SLR1009 PKP
17	0.1115	SLR1409 PKP
18	0.1115	SLR0409 PKP
19	0.1224	BLR1009 PKP
20	0.1224	BLR1409 PKP
21	0.1224	BLR0409 PKP
22	0.1401	BR09 OP

Table XIII. Candidates for estimated fault sections based on SCADA data of Case 2

Candidate No.	Fault Section	Truth Degree Value
1	B04	0.855
2	B09	0.855
3	L0409	0.513

Table XIV. Candidates for estimated fault sections based on SCADA data and relay data of Case 2

Candidate No.	Fault Section	Truth Degree Value
1	B04	0.882
2	B09	0.882
3	L0409	0.618

Table XV. Scenario description of Case 3 for the fault section estimation study

Scenario	Description
Power System Fault	A permanent fault occurred on the transmission line L1314 at 0.05 second; A second permanent fault occurred on the bus B13 at 0.11 second.
Protection Device Failure	The circuit breakers CB1312 and CB1306 failed to open
False Data	The BR13 TRIP signal should be observed but it was not observed.

Table XVI. SCADA data of Case 3 for the fault section estimation study

Sequence No.	Time Stamp (Sec)	Observed Signal
1	0.1000	MLR1314 TRIP
2	0.1000	MLR1413 TRIP
3	0.2000	CB1314 OPEN
4	0.2000	CB1413 OPEN
5	0.3000	BLR0613 TRIP
6	0.3000	BLR1213 TRIP
7	0.3000	CB0613 OPEN
8	0.3000	CB1213 OPEN

Table XVII. Relay data of Case 3 for the fault section estimation study

Sequence No.	Time Stamp (Sec)	Observed Signal
1	0.0641	SLR1314 PKP
2	0.0651	SLR1413 PKP
3	0.0683	BLR1314 PKP
4	0.0688	BLR1413 PKP
5	0.0693	SLR0914 PKP
6	0.0698	MLR1314 PKP
7	0.0698	MLR1413 PKP
8	0.0703	SLR0613 PKP
9	0.0703	SLR1213 PKP
10	0.0740	MLR1314 OP
11	0.0740	MLR1413 OP
12	0.1141	BR13 PKP
13	0.1193	SLR0613 PKP
14	0.1204	SLR1213 PKP
15	0.1271	BLR0613 PKP
16	0.1297	BLR1213 PKP
17	0.1605	BR13 OP
18	0.2433	BLR0613 OP
19	0.2459	BLR1213 OP

Table XVIII. Candidates for estimated fault sections based on SCADA data of Case 3

Candidate No.	Fault Section	Truth Degree Value
1	L1314	0.855
2	B13	0.729
3	L1213	0.647
4	L0613	0.647

Table XIX. Candidates for estimated fault sections based on SCADA data and relay data of Case 3

Candidate No.	Fault Section	Truth Degree Value
1	L1314	0.882
2	B13	0.854
3	L1213	0.722
4	L0613	0.722

3. Discussion

As shown in Case 1, if the scenario is a single fault without protection device failure and false data, the faulted section can be accurately identified. The truth degree value of the result based on both the relay data and SCADA data are higher than that based on only the SCADA data, because the relay data are assigned higher truth degree values due to their higher reliability. Case 2 is more complex than Case 1, because multiple faults occur. As shown in Table XIII and Table XIV, besides the bus B04 and the bus B09, on which faults actually occur, the transmission line L0409, which has no fault, is included in the candidate set. The transmission line L0409 has a far smaller truth degree value than the other two candidates, which indicates small possibility of fault occurrence. Similar to Case 1, the truth degree values of the candidates based on both the relay data and SCADA data are higher than those based on only the SCADA data. Case 3 has additional complexity, because not only multiple faults but also protection device failure and false data are involved. As shown in Table XVIII and Table XIX, besides the transmission line L1314 and bus B13, on which fault actually occur, the transmission line L1213 and transmission line L0613, which have no fault, are included in the candidate set. The use of relay data increases the truth degree values for all candidates. It should be noticed that although the truth degree values of L1213 and L0613 are increased to some extent, the truth degree value of B13 are largely increased. The actual faulted sections can still be identified.

C. Protection System Performance Evaluation

1. Test Environment and Relay Configuration

In a lab environment, Schweitzer Engineering Laboratories' SEL421 and General Electric' D60 relays have been configured with the same settings as those of the two relays protecting a 345 KV transmission line in a substation of CenterPoint Energy, Houston. A digital simulator is used to generate the voltage and current inputs to the two relays. It also simulates the breaker status signals which are monitored by the two relays.

In the substation, the 345 KV transmission line of interest with a length of 31.5 miles is protected by a SEL421 relay and a D60 relay. The SEL421 relay acts as main protection which implements a Directional Comparison Blocking scheme (DCB). The D60 performs backup protection which implements distance and over current protection. The trip contacts are combined together to trip the circuit breakers F280 and F290.

The principle of DCB scheme is described in Fig. 30. Line RS is protected by protection elements located at Terminal R and Terminal S. The directional fault detection elements at Terminal R and Terminal S, designated as FD(R) and FD(S) respectively, are set to overreach the remote terminals so that they will pick up for all internal faults on Line RS. Usually they are set to overreach by 120-150% of the line length. Starting elements, designated as ST(R) and ST(S), are set with different reach than the fault detection elements. ST(R) and ST(S) can be directional or non-directional. If they are directional, ST(R) will only pick up for external faults to the left of Terminal S and within its reach. Similarly, ST(S) will only pick up for external faults to the right of Terminal S and within its reach. If ST(R) and ST(S) are non-directional, they will pick up for both internal and external faults within their reach. If

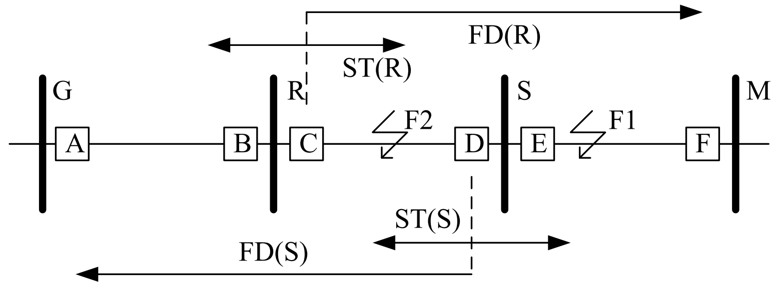

Fig. 30. Principle of DCB scheme

ST(R) and ST(S) pick up, they will key the communication equipment on to transmit the blocking signal to the fault detection element at the remote terminal. Non-directional starting elements do not process a directional decision, so non-directional starting elements are always faster than directional starting elements.

If an internal fault occurs, FD(R) will pick up at Terminal R. The pickup of FD(R) will also key its associated communication equipment off to prevent sending the blocking signal in case of the pickup of ST(R). Similarly, at Terminal S, FD(S) will pick up and operate. The pickup of FD(S) will also key its associated communication equipment off to prevent sending the blocking signal in case of the pickup of ST(S). Consequently, there is no blocking signal sent in either direction, and Circuit Breaker C and D will trip to open and clear the fault.

If an external fault to the left of Terminal R occurs, FD(R) will not pick up. But FD(S) will pick up if the fault is within its reach. ST(R) will pick up to key its associated communication equipment on to send the blocking signal to the remote FD(S). Thus the tripping of Circuit Breaker D by FD(S) will be blocked. If an external fault to the right of Terminal S occurs, the tripping of Circuit Breaker C by FD(R) will be blocked in a similar manner.

According to the SEL421 relay configuration, Phase Distance Zone 2 Element M2P and Neutral Directional Over-current Level 2 Element 67G2 are configured as

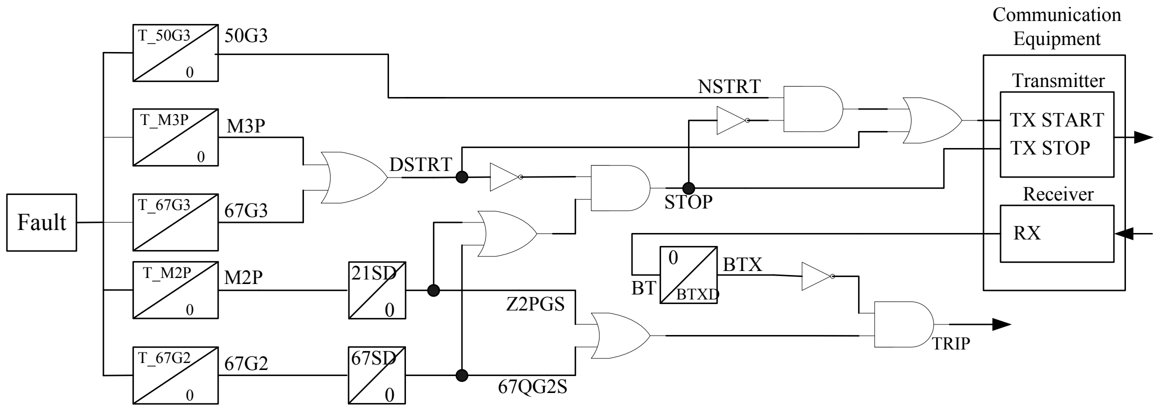

Fig. 31. SEL421 operation logic

phase and ground fault detection elements. Phase Distance Zone 3 Element M3P and Neutral Directional Over-current Level 3 Element 67G3 are configured as directional starting elements. Neutral Non-directional Over-current Level 3 Element 50G3 is configured as a non-directional starting element. Fig. 31 describes the operation logic. As shown in Fig. 31, the operation of M2P or 67G2 may trigger the relay to trip if the blocking signal is not received. The pickup of directional starting elements M3P or 67G3 will directly key the communication equipment on to send the blocking signal and can not be stopped by fault detection elements. While directional starting elements have priority over fault detection elements, fault detection elements have priority over non-directional starting elements. That is to say, the operation of fault detection elements will prevent non-directional starting elements to key the communication equipment on to send the blocking signal. Besides logic issues, some important timing issues need to be carefully considered.

Timers 21SD and 67SD are used to set the carrier coordination time delay for M2P and 67G2 respectively. The delay should allow the blocking signal from the remote relay to arrive before the local circuit breaker trips for external faults behind the remote terminal. The setting for the timers is a sum of the following three time

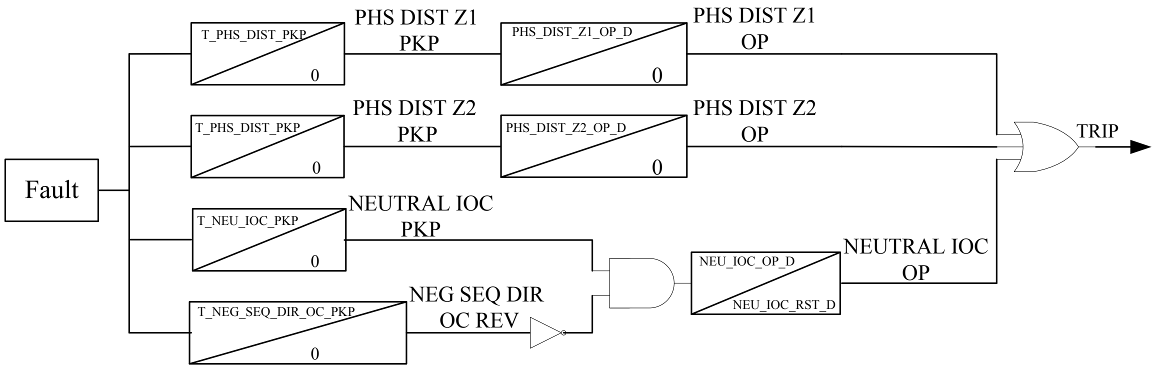

Fig. 32. D60 operation logic

intervals: the maximum pickup time of the remote starting elements, the maximum communication channel operating time, the contact input recognition time of the local relay.

Timer BTXD is used to set the delay of dropout of the contact input assigned to logic operand BT (Blocking Signal Received). This input must remain asserted to block fault detection elements from triggering the relay trip after the carrier coordination timers expire. If the blocking signal drops out momentarily, fault detection elements can trigger the relay trip for external faults behind the remote terminal. The delay by the timer helps avoid unexpected trip during momentary lapses of the blocking signal (carrier holes).

According to the D60 relay configuration, Phase Distance Zone 1 Element and Phase Distance Zone 2 Element are configured to look at phase faults in the forward direction. Neutral Instantaneous Over-current Element is configured to look at ground faults. Since the Neutral Instantaneous Over-current Element is non-directional, the Reverse Directional Negative Sequence Over-current Element is used to block the Neutral Instantaneous Over-current Element from operating for reverse ground faults. Fig. 32 describes the D60 operation logic.

2. Test Cases and Results

a. Case 1

The scenario of Case 1 is described in Table XX. Fig. 33 is the analysis report generated by the protection system performance evaluation application.

b. Case 2

The scenario of Case 2 is described in Table XXI. Fig. 34 is the analysis report generated by the protection system performance evaluation application.

c. Case 3

The scenario of Case 3 is described in Table XXII. Fig. 35 is the analysis report generated by the protection system performance evaluation application.

3. Discussion

The scenario of Case 1 assumes that no malfunction or error happens in the protection system and hence it represents a normal situation. As shown in Fig. 33, the analysis report shows that all the input and output contacts, and protection elements of both the D60 relay and SEL421 relay are operated as expected. In Case 2, the opening of a circuit breaker is assumed to be 1 cycle slower. As shown in Fig. 34, the analysis report reveals that the slower circuit breaker opening is identified based on either D60 relay data or SEL421 data. In Case 3, the pickup settings of phase distance elements of the SEl421 relay are assumed to be incorrect, which makes the relay fail to issue trip signals, while the D60 relay works correctly to trip the circuit breakers. As shown in Fig. 35, the analysis report tells that the pickup of phase distance elements and the assert of relay trip signals of SEl421 relay have failed. The reason is incorrect pickup settings of the phase distance element.

Table XX. Scenario of Case 1 for the protection system performance evaluation study

Fault Type	Permanent AB fault
Fault Location	50 % (15.75 miles)
Device Mulfunction/Error	None
Description	The main and backup relays tripped circuit breakers to clear the fault as expected.

```
DISTURBANCE INFORMATION FROM D60:
---------------------------------
Fault Inception Time: 08/16/2006,21:59:45.7792
Fault Clearance Time: 08/16/2006,21:59:45.8350
Fault Type: AB
Fault Location: 16.4 miles

DISTURBANCE INFORMATION FROM SEL421
-----------------------------------
Fault Inception Time: 08/16/2006,21:59:45.7796
Fault Clearance Time: 08/16/2006,21:59:45.8356
Fault Type: AB
Fault Location: 16.39 miles

CONCLUSIONS ON D60:
-------------------
52a2 turned off as expected <-- CB < 52a2 > opened as expected
52a1 turned off as expected <-- CB < 52a1 > opened as expected
H1 TRIP1 TC1 turned on as expected <-- Relay tripped H1 TRIP1 TC1 as expected
H3 TRIP2 TC1 turned on as expected <-- Relay tripped H3 TRIP2 TC1 as expected
DIRECT TRIP turned on as expected
PH DIST Z1 operated as expected
NEG SEQ DIR OC1 FWD picked up as expected
PH DIST Z1 picked up as expected
PH DIST Z2 picked up as expected

CONCLUSIONS ON SEL421:
----------------------
IN101 deasserted as expected <-- CB < IN101 > opened as expected
IN102 deasserted as expected <-- CB < IN102 > opened as expected
OUT102 asserted as expected <-- Relay tripped OUT102 as expected
OUT106 asserted as expected <-- Relay tripped OUT106 as expected
TRIP asserted as expected
MPT Z1 asserted as expected
67P Z1 asserted as expected
50P Z1 asserted as expected
MP Z4 asserted as expected
MP Z2 asserted as expected
MP Z1 asserted as expected
Z2PGS asserted as expected
STOP asserted as expected
```

Fig. 33. Analysis report of Case 1 for the protection system performance evaluation study

Table XXI. Scenario of Case 2 for the protection system performance evaluation study

Fault Type	Permanent AG fault
Fault Location	40 % (12.60 miles)
Device Mulfunction/Error	Opening of the middle circuit breaker F280 is delayed for 1 cycle.
Description	The main and backup relays tripped circuit breakers to clear the fault. The fault clearance is delayed due to slow opening of the middle circuit breaker F280.

```
DISTURBANCE INFORMATION FROM D60:
---------------------------------
Fault Inception Time: 08/18/2006,16:35:46.2780
Fault Clearance Time: 08/18/2006,16:35:46.3483
Fault Type: AG
Fault Location: 13.1 miles

DISTURBANCE INFORMATION FROM SEL421
-----------------------------------
Fault Inception Time: 08/18/2006,16:35:46.2785
Fault Clearance Time: 08/18/2006,16:35:46.3490
Fault Type: AG
Fault Location: 13.18 miles

CONCLUSIONS ON D60:
-------------------
52a2 turned off slower than expected <-- CB < 52a2 > opened slower than expected
52a1 turned off as expected <-- CB < 52a1 > opened as expected
H1 TRIP1 TC1 turned on as expected <-- Relay tripped H1 TRIP1 TC1 as expected
H3 TRIP2 TC1 turned on as expected <-- Relay tripped H3 TRIP2 TC1 as expected
DIRECT TRIP turned on as expected
NEUTRAL IOC1 operated as expected
NEG SEQ DIR OC1 FWD picked up as expected
NEUTRAL IOC1 picked up as expected

CONCLUSIONS ON SEL421:
----------------------
IN101 deasserted as expected <-- CB < IN101 > opened as expected
IN102 deasserted slower than expected <-- CB < IN102 > opened slower than expected
OUT102 asserted as expected <-- Relay tripped OUT102 as expected
OUT106 asserted as expected <-- Relay tripped OUT106 as expected
TRIP asserted as expected
67GT Z1 asserted as expected
51S Z1 asserted as expected
67G Z2 asserted as expected
67G Z1 asserted as expected
67P Z1 asserted as expected
50P Z1 asserted as expected
67QG2S asserted as expected
NSTRT asserted as expected
STOP asserted as expected
```

Fig. 34. Analysis report of Case 2 of the protection system performance evaluation study

Table XXII. Scenario of Case 3 for the protection system performance evaluation study

Fault Type	Permanent AB fault
Fault Location	50 % (15.75 miles)
Device Malfunction/Error	The pickup settings of Phase Distance Zone 1 Element, Phase Distance Zone 2 Element and Phase Distance Zone 4 Element of the main relay are not correct.
Description	The main relay failed to trip circuit breakers due to incorrect settings. The backup relay tripped the circuit breakers to clear the fault.

```
DISTURBANCE INFORMATION FROM D60:
---------------------------------
Fault Inception Time: 08/16/2006,22:08:53.4587
Fault Clearance Time: 08/16/2006,22:08:53.5144
Fault Type: AB
Fault Location: 16.5 miles

DISTURBANCE INFORMATION FROM SEL421
-----------------------------------
Fault Inception Time: 08/16/2006,22:08:53.4590
Fault Clearance Time: 08/16/2006,22:08:53.5149
Fault Type: AB
Fault Location: 16.41 miles

CONCLUSIONS ON D60:
-------------------
52a2 turned off as expected <-- CB < 52a2 > opened as expected
52a1 turned off as expected <-- CB < 52a1 > opened as expected
H1 TRIP1 TC1 turned on as expected <-- Relay tripped H1 TRIP1 TC1 as expected
H3 TRIP2 TC1 turned on as expected <-- Relay tripped H3 TRIP2 TC1 as expected
DIRECT TRIP turned on as expected
PH DIST Z1 operated as expected
NEG SEQ DIR OC1 FWD picked up as expected
PH DIST Z1 picked up as expected
PH DIST Z2 picked up as expected

CONCLUSIONS ON SEL421:
----------------------
IN101 deasserted as expected <-- CB < IN101 > opened as expected
IN102 deasserted as expected <-- CB < IN102 > opened as expected
OUT102 failed to assert <-- Relay failed to trip OUT102
OUT106 failed to assert <-- Relay failed to trip OUT106
TRIP failed to assert
67P Z1 asserted as expected
50P Z1 asserted as expected
MP Z4 failed to assert <-- Check MP Z4 setting
MP Z2 failed to assert <-- Check MP Z2 setting
MP Z1 failed to assert <-- Check MP Z1 setting
Z2PGS failed to assert
STOP failed to assert
```

Fig. 35. Analysis report of Case 3 for the protection system performance evaluation study

D. Summary

Case study of the proposed FRPN approach for fault section estimation and protection system performance evaluation application is presented in this chapter. Experiential tests have shown that the proposed FRPN approach for fault section estimation is able to perform accurate fault section estimation under complex scenarios. It was also shown that the developed protection system performance evaluation application has successfully performed relay performance analysis.

CHAPTER VIII

IMPLEMENTATION SOLUTIONS

A. Introduction

This chapter proposes implementation solutions for the problems of fault section estimation, protection system performance evaluation, and power system/protection system interactive simulation. First, a fault section estimation application implemented in a control center and its SCADA system support infrastructure are presented. Then a protection system performance evaluation application implemented in a substation is illustrated and some prototype software development is demonstrated. Finally the process of power system/protection system interactive simulation is described and a GUI prototype for the setting program is demonstrated.

B. Implementation Solutions

1. Fault Section Estimation

The fault section estimation application will be implemented in a control center to assist the system operator in rapidly identifying faulted sections for restoration process. The structure of the application as well as its SCADA support infrastructure are illustrated in Fig. 36.

In such a solution, input data such as relay trip signals and circuit breaker status signals are acquired by RTUs of the SCADA system. Relay logic operand signals are defined in their data memories and retrieved from relays by the SCADA front-end computers in substations. The data are acquired from different substations and are transmitted to the control center through selected communication link such as microwave or optical fiber. In the control center, the SCADA master computer

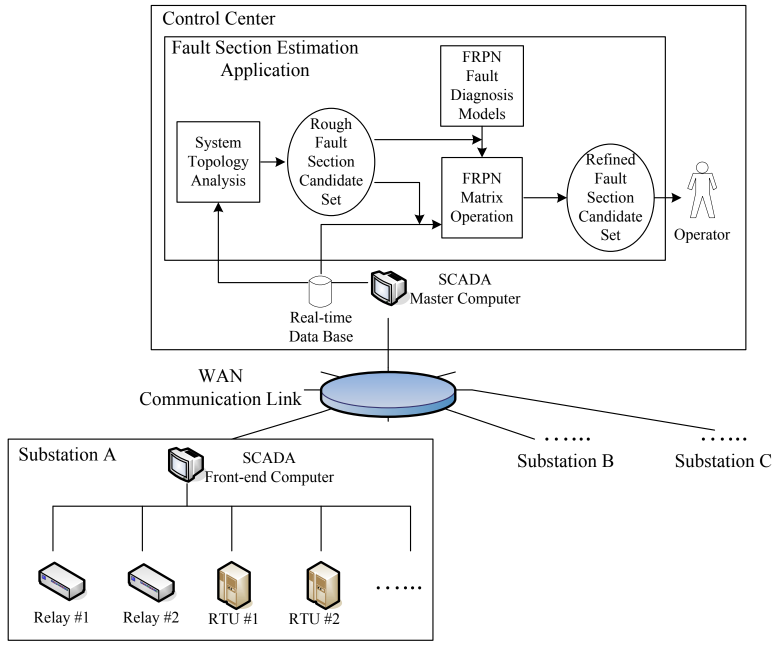

Fig. 36. Implementation of fault section estimation application

puts the input data into a real-time data base and keeps updating them at each scan time.

The fault section estimation application includes two stage analysis. In the first stage, the system topology is analyzed based on circuit breaker status data in the real-time data base. The analysis will include all sections isolated by the opening of circuit breakers into a rough candidate set. The set is rough because it more likely includes sections which are not faulted but are isolated due to backup relay operation. In the second stage analysis, the Fuzzy Reasoning Petri-net diagnosis model as well as data in the real-time data base corresponding to each section in the rough candidate set are used and Fuzzy Reasoning Petri-net matrix operation is implemented. As a result, each section will be associated with a truth degree value. The section with a truth degree value greater than a certain threshold will be included in the refined candidate set. Such a refined candidate set is presented to the system operator for decision-making.

In such a solution, the FRPN models which are represented by all kind of matrices are separated from FRPN matrix operations. This is analogous to an expert system whose rule-base is separated from its inference engine. The FRPN models can be built in advance based on power system and protection system configurations and stored in files. In such a way, the FRPN models can be easily modified according to the change of power system and protection system configuration.

2. Protection System Performance Evaluation

The Protection System Performance Evaluation Application will be implemented in a substation in assisting protection engineers to assess protection system performance in the post-fault analysis. Fig. 37 shows the implementation structure.

In such a solution, an automated relay file retrieval module communicates to relays through serial communication links according to communication port settings and file retrieval settings. It monitors new relay files triggered by the operations of relays and downloads them into specified file repository. Fig. 38 shows a GUI for the automated relay file retrieval module.

A file data processing module monitors the incoming file repository, processes file data and converts them into initial facts of CLIPS expert system. Another category of initial facts is performance specification data which are input from GUI by protection engineers. Fig. 39 shows a GUI for the performance specification input.

In our solution, the inference engine of CLIPS expert system is complied as a Dynamic Link Library (DLL) and embedded in the application. The API functions for the DLL are used as bridges for initial facts, rule-base and inference engine. The rule-base is created by protection engineers and stored in a text file.

The final analysis report generated is displayed in the GUI and also stored into a

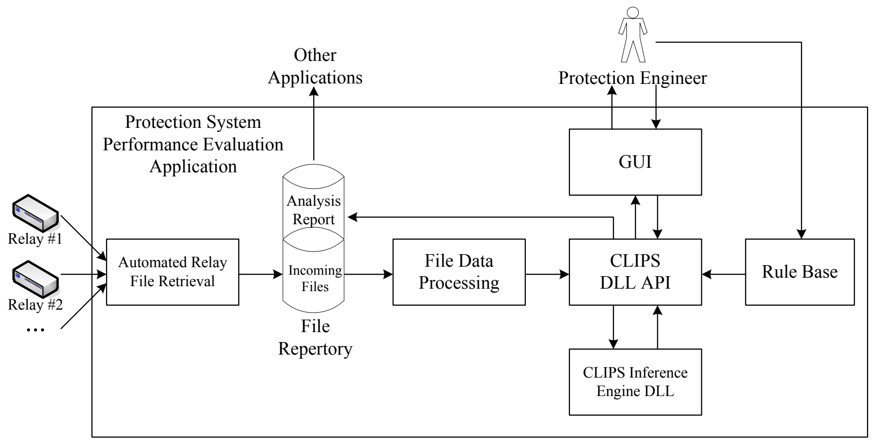

Fig. 37. Implementation of protection system performance evaluation application

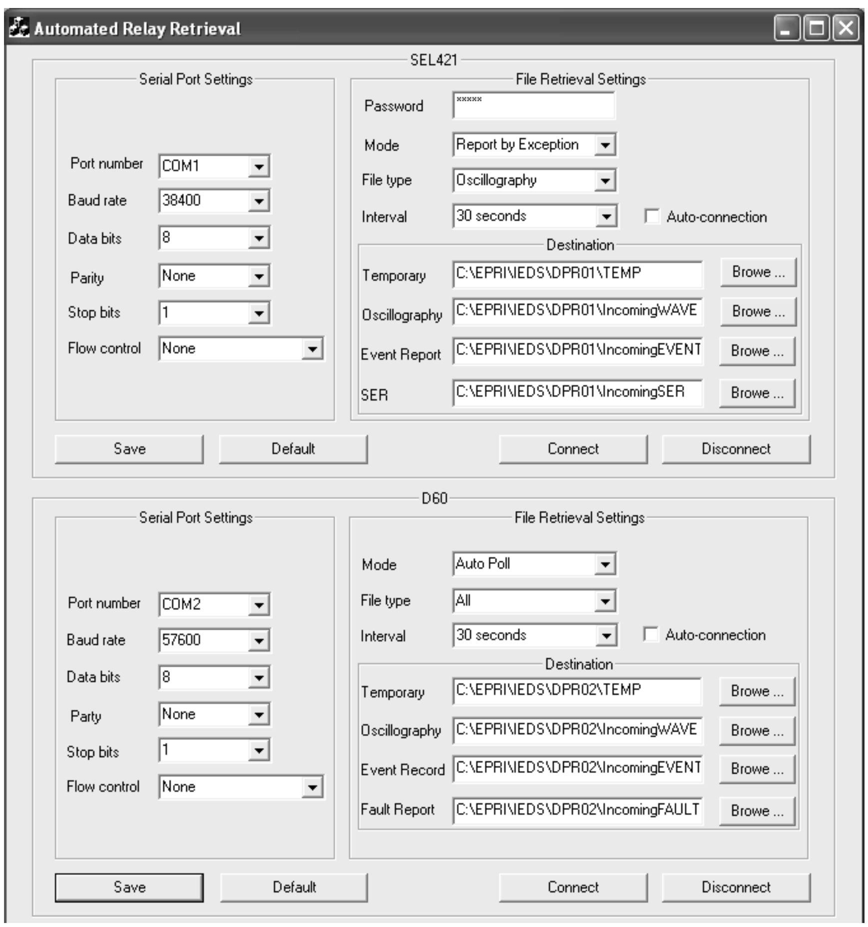

Fig. 38. GUI of automated relay file retrieval module

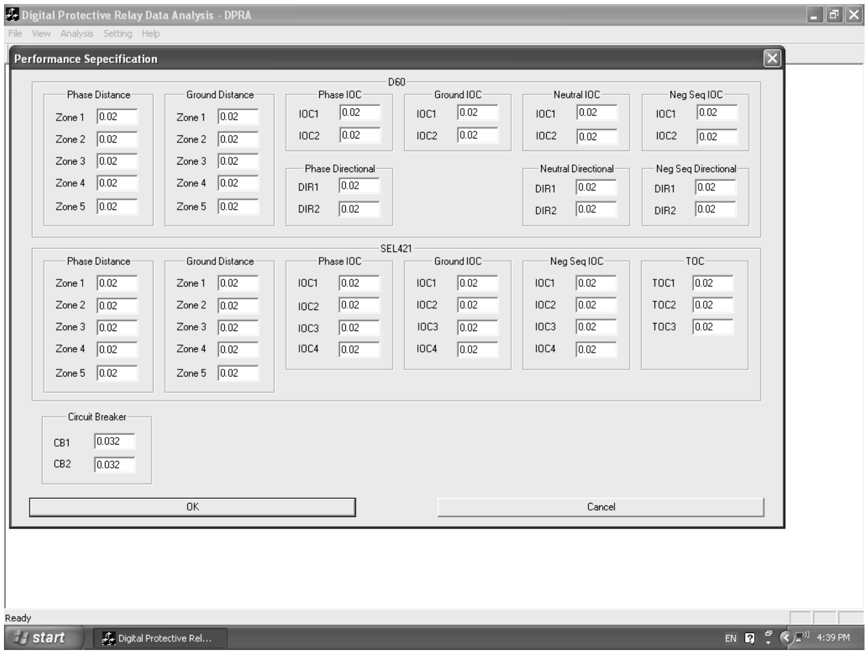

Fig. 39. GUI of protection system performance evaluation application

report repository. The analysis report can be utilized by other applications to perform more comprehensive analysis.

3. Power System/Protection System Interactive Simulation

In the procedure of power system/protection system interactive simulation, the user will fist prepare the ADP file which models the power system, the MODELS data case which specifies the interface between the power system model and the protection system model, and the C++ source code which models the protection system. Then an ATPDraw program, an ATP/MinGW program, and a simulation setting program will be utilized in the process [57,59]. Fig. 40 illustrates the structure of the implementation.

The ADP file is graphically created in the ATPDraw program. The user can build a power system model by adding in all kinds of power system components.

The model can be built in a hierarchical way in that a group of components can be represented by a user-created icon. It should be mentioned that the models of circuit breakers, current transformers (CT) and voltage transformers (VT) which belong to protection systems are included in the ADP file. Fig. 29 in Chapter VII is an example of an ADP file.

By using the ATPDraw program, the ADP file can be converted to an ATP data case file. Then an MODELS data case is inserted into the ATP data case file to add the interface between the power system model and the protection system model. The augmented file is called a template ATP data case file.

A simulation setting program is used to modify the template ATP data case file in order to specify the simulation parameters such as the total simulation time and the length of simulation time step, and the fault parameters such as fault section, fault type, fault location and fault inception time. The program is also utilized to specify protection system settings and user-defined errors. As the outcome, a modified ATP data case file, a protection system setting file and a protection system error code file are created. Fig. 41 shows the GUI of the simulation setting program.

The C++ source code file which models the protection system is compiled as an object and linked with the original ATP object and some libraries by the ATP/MinGW program. As the outcome, an executable ATP program will be created. Such a program takes the modified ATP data case file, the protection system setting file and the protection system error code file as inputs, performs the simulation, and generates the ATP PL4 file which stores the power system measurement data and the protection system event report which reveals the protection system behavior.

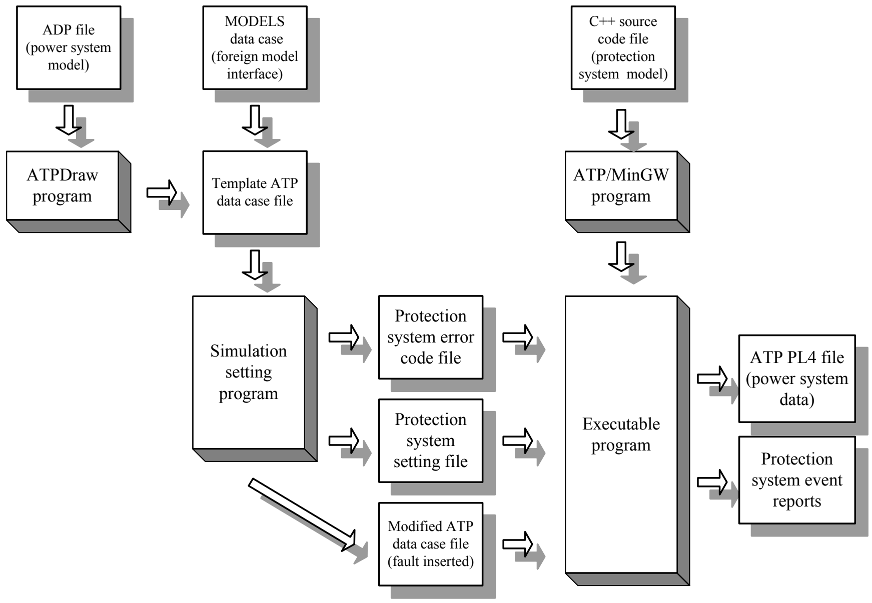

Fig. 40. Implementation of power system/protection system interactive simulation

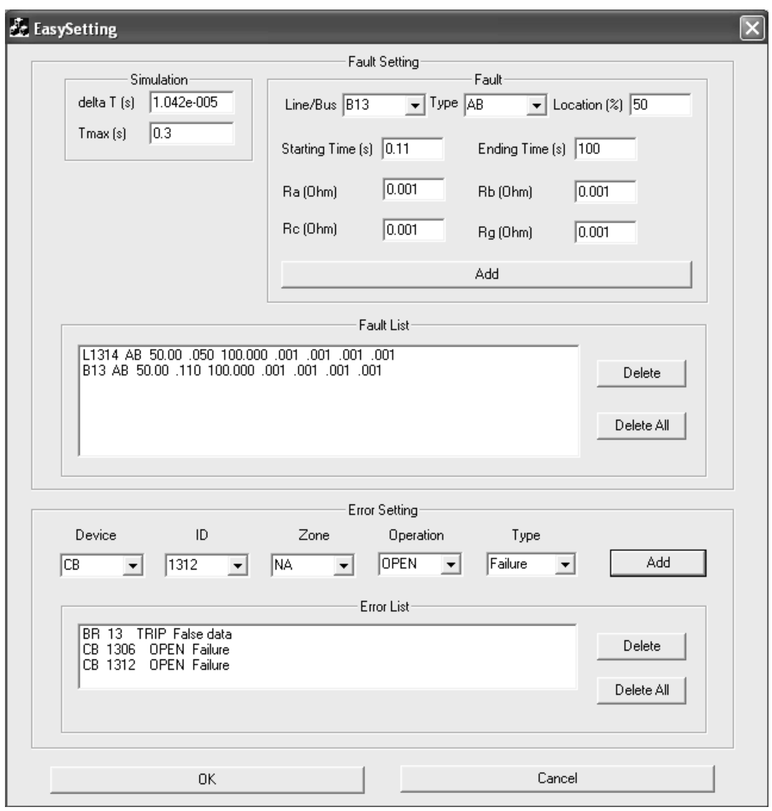

Fig. 41. GUI of the simulation setting program

C. Summary

The implementation solutions for the problems of fault section estimation, protection system performance evaluation, and power system/protection system interactive simulation are proposed in this chapter. In the fault section estimation application, two-stage analysis is proposed for implementation. In the first stage, a system topology analysis results in a rough fault section candidate set and in the second stage, FRPN models and matrix operations are implemented to refine the candidate set. In the protection system performance evaluation application, an automated relay file retrieval module and a relay file data processing module are implemented to get the initial facts. An inference engine of CLIPS expert system is compiled as a DLL to perform the reasoning based on initial facts and rules. The results are displayed in user interface and stored in file repository. In the process of power system/protection system interactive simulation, the ADP file which models the power system, the MODELS data case which specifies the interface between the power system model and the protection system model, and the C++ source code which models the protection system need to be prepared. Then ATPDraw program, ATP/MinGW program, and simulation setting program will be utilized in the process. The results are the ATP PL4 file and the protection system event report.

CHAPTER IX

CONCLUSIONS

A. Expected Benefits

This dissertation has focused on three fundamental problems in power system fault analysis, namely fault section estimation, protection system performance evaluation, and power system/protection system interactive simulation. Although there are existing solutions to these problems, new approaches proposed in the dissertation have their unique strength in solving these problems and have demonstrate their advantages. The expected benefits to be gained from the proposed approaches are summarized as follows:

1. The Fuzzy Reasoning Petri-nets technique has combined strength of uncertainty processing, rule-based reasoning, symbolic representation, and parallel computing. It makes fault section estimation more accurate, fast and adaptive to system changes. Especially, the reasoning process can be visualized in a form of graphical representation of Petri-nets. The rule base and parameters are saved in matrix forms and the whole reasoning process is implemented by matrix operations. This will significantly facilitates the procedure of rule base building and maintenance. It will provide system operators a fast and reliable tool for identifying fault sections in the restorative stage.

2. IED Data are more reliable than SCADA data. The proposed approach to combine IED data and SCADA data will further enhance the accuracy of fault section estimation.

3. The developed protection system performance evaluation application stream-

lines the process of assessing the protection system operations. It automates acquisition of data, validation of protection system operations and diagnosis of unexpected operations, which are previously done manually by protection engineers in the post-fault analysis. This allows fast and reliable assessment of a large number of protection system operations.

4. The "compiled foreign model" approach of power system/protection system interactive simulation allows modeling of sophisticated protection systems in an "object-oriented" way as well as building a "seamless" interface between power system models and protection system models. It will provide a convenient experimental platform for various research activities related to protection systems and power system fault analysis.

B. Research Contribution

Power system fault analysis provides critical information to utility staff to be able to understand the reasons for power system interruption better and provide action to restore the power delivery quicker. Three fundamental problems in power system fault analysis, namely fault section estimation, protection system performance evaluation, and power system/protection system interactive simulation have been researched. Although there are existing solutions to these problems, the dissertation study has made new contributions to these areas.

A Fuzzy Reasoning Petri-nets (FRPN) approach to solve the problem of fault section estimation is discussed in Chapter IV. In this approach, the fuzzy reasoning from protection system status data to faulted power system sections is formulated by Petri-nets. The reasoning process can be graphically represented in a form of Petri-nets and implemented by matrix operations. Data acquired by RTUs of SCADA systems are the inputs to the diagnosis models. The logic operand data of digital protective relays, which are more reliable than the SCADA data, are utilized as additional inputs. The matrix formalism of implementing Fuzzy Reasoning Petri-net based fault section estimation is not reported in existing literature and the concept of utilizing IED data for fault section estimation is new in this area.

An automated protection system performance evaluation application which is based on rule-based reasoning technique is presented in Chapter V. The application consists of a relay file retrieval module, a relay file data processing module, and a rule-based protection operation validation and diagnosis module. The relay file retrieval module utilizes file transfer mechanisms of digital protective relays to automatically retrieve relay files. The data and information contained in these files are processed and converted into proper format in the relay file data processing module. In the rule-based protection operation validation and diagnosis module, forward chaining reasoning is used for predicting expected protection operations while backward chaining reasoning is used for diagnosing unexpected protection operations. Such a completely automated application is rather new in the power system industry.

A "compiled foreign model" approach to the problem of power system/protection system interactive simulation is detailed in Chapter VI. This approach enables modeling of a sophisticated digital protective relay using C++, and "seamless" interfacing of the relay model and power system network model. The resulting digital relay model can be reused and has capability of reading settings , generating event reports, and inserting user-defined errors. As an outcome, the power system/protection system interactive simulation environment is more convenient and powerful than existing solutions.

C. Suggestion for Future Work

Although significant achievements have been made in the dissertation study, due to limitation of time, many research topics still remain to be explored. Some further work is suggested as follows:

1. For the Fuzzy Reasoning Petri-nets based fault section estimation, timing information of the logic operands of digital relays can be utilized to further improve the estimation accuracy. Sparse matrices largely exist in Petri-nets representation of fault section diagnosis models. Techniques for processing sparse matrices can be utilized to improve efficiency of memory usage and computation.

2. The automated protection system performance evaluation application mainly uses relay file data as inputs. It can be integrated with other analysis applications such as DFR data analysis application and circuit breaker monitor data analysis application to achieve more accurate and comprehensive analysis.

3. MATLAB and some other intelligent system shells provide run-time access routine for C/C++ language. In the "compiled foreign model" based power system/protection system interactive simulation, the digital relay model can be improved to utilize the functions in the MATLAB and the intelligent system shells such as an expert system shell. Thus an improved platform to study intelligent system application to analysis of protection system operation based on interactive simulation can be implemented.

REFERENCES

[1] M. Kezunovic, C. C. Liu, J. McDonald, and L. E. Smith, "Automated fault analysis," IEEE Tutorial, IEEE Power Engineering Society, 2000.

[2] A. A. Girgis and M. B. Johns, "A hybrid expert system for faulted section identification, fault type classification and selection of fault location algorithms," *IEEE Trans. Power Delivery*, vol. 4, no. 2, pp. 978–985, April 1989.

[3] S. Vasilic and M. Kezunovic, "Fuzzy ART neural network algorithm for classifying the power system faults," *IEEE Trans. Power Delivery*, vol. 20, no. 2, pp. 1306–1314, April 2005.

[4] H. Ren, Z. Q. Mi, H. S. Zhao, and Q. X. Yang, "Fault diagnosis for substation automation based on Petri nets and coding theory," in *Proc. IEEE Power Engineering Society General Meeting*, Denver, CO, June 2004, vol. 1, pp. 1038–1042.

[5] M. Kezunovic, P. Spasojevic, C. Fromen, and D. Sevcik, "An expert system for transmission substation event analysis," *IEEE Trans. Power Delivery*, vol. 8, no. 4, pp. 1942–1949, October 1993.

[6] X. Luo and M. Kezunovic, "Fault analysis based on integration of digital relay and DFR data," in *Proc. IEEE Power Engineering Society General Meeting*, San Francisco, CA, June 2005, vol. 1, pp. 746–751.

[7] J. Giarratano and G. Riley, *Expert Systems: Principles and Programming*, PWS Publishing Company, Boston, MA, 2nd edition, 1994.

[8] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, no. 1, pp. 1–6, 1965.

[9] J. Cardoso and H. Camargo, *Fuzziness in Petri Nets*, Physica-Verlag Heidelberg, New York, NY, 1999.

[10] G. J. Klir and B. Yuan, *Fuzzy Sets and Fuzzy Logic: Theory and Application*, Prentice-Hall, Englewood Cliffs, NJ, 1995.

[11] S. M. Chen, J. S. Ke, and J. F. Chang, "Knowledge representation using fuzzy Petri nets," *IEEE Trans. Knowledge and Data Engineering*, vol. 2, no. 3, pp. 311–319, September 1990.

[12] M. M. Gao, M. C. Zhou, X. G. Huang, and Z. M. Wu, "Fuzzy reasoning Petri nets," *IEEE Trans. Systems, Man and Cybernetics, Part A*, vol. 33, no. 3, pp. 314–324, May 2003.

[13] W. R. Block, "Fundamentals of supervisory systems," IEEE Tutorial, IEEE Power Engineering Society, EH0337-6 PWR, 1991.

[14] General Electric Company, Fairfield, CT, *Instruction Manual for D60 Line Distance Relay*, 2004.

[15] Schweitzer Engineering Laboratories, Inc., Pullman, WA, *Instruction Manual for SEL421 High-Speed Line Protection, Automation, and Control System*, 2004.

[16] E. A. Udren, J. A. Zipp, G. L. Michel, K. K. Mustaphi, and S. L. Nilsson et al., "Proposed statistical performance measures for microprocessor-based transmission-line protective relays, Part I: Explanation of the statistics," *IEEE Trans. Power Delivery*, vol. 12, no. 1, pp. 134–143, January 1997.

[17] E. A. Udren, J. A. Zipp, G. L. Michel, K. K. Mustaphi, and S. L. Nilsson et al., "Proposed statistical performance measures for microprocessor-based

transmission-line protective relays, Part II: Collection and uses of data," *IEEE Trans. Power Delivery*, vol. 12, no. 1, pp. 144–156, January 1997.

[18] P.M. Anderson, *Power System Protection*, pp. 1055–1056, IEEE Press, New York, NY, 1999.

[19] IEEE PES Power System Relaying Committee, "Pilot relaying performance analysis," *IEEE Trans. Power Delivery*, vol. 5, no. 1, pp. 85–102, January 1990.

[20] M. Kezunovic and Q. Chen, "A novel approach for interactive protection system simulation," *IEEE Trans. Power Systems*, vol. 12, no. 2, pp. 668–674, April 1997.

[21] M. Kezunovic, S. Vasilic, and D. Ristanovic, "Interfacing protective relays and relay models to power system modeling software and data files," in *Proc. International Conference on Power System Technology - PowerCon*, Kunming, China, October 2002, vol. 1, pp. 253–259.

[22] Q. Chen, "Interactive distance protection simulation for computer relay evaluation and studies," Ph.D. dissertation, Texas A&M University, College Station, TX, 1996.

[23] Y. L. Zhu, Y. H. Yang, B. W. Hogg, W. Q. Zhang, and S. Gao, "An expert system for power systems fault analysis," *IEEE Trans. Power Systems*, vol. 9, no. 1, pp. 503–509, February 1994.

[24] E. Vazquez M., O. L. Chacon M., and H. J. Altuve F., "An on-line expert system for fault section diagnosis in power systems," *IEEE Trans. Power Systems*, vol. 12, no. 1, pp. 357–362, February 1997.

[25] Y. C. Huang, "Fault section estimation in power systems using a novel decision support system," *IEEE Trans. Power Systems*, vol. 17, no. 2, pp. 439–444, May

2002.

[26] Power System Relaying Committee Working Group D10 of the Line Protection Subcommittee, "Potential applications of expert systems to power system protection," *IEEE Trans. Power Delivery*, vol. 9, no. 2, pp. 720–728, April 1994.

[27] H. T. Yang, W. Y. Chang, and C. L. Huang, "A new neural networks approach to on-line fault section estimation using information of protective relays and circuit breakers," *IEEE Trans. Power Delivery*, vol. 9, no. 1, pp. 220–230, January 1994.

[28] M. Kezunovic, "A survey of neural net applications to protective relaying and fault analysis," *Engineering Intelligent Systems*, vol. 5, no. 4, pp. 185–192, December 1997.

[29] Z. E. Aygen, S. Seker, M. Bagnyanik, F. G. Bagnyanik, and E. Ayaz, "Fault section estimation in electrical power systems using artificial neural network approach," in *Proc. IEEE Transmission and Distribution Conference*, New Orleans, LA, April 1999, vol. 2, pp. 466–469.

[30] T. S. Bi, Y. X. Ni, and F. F. Wu, "Distributed adaptive fault section estimation system for large-scale power networks," in *Proc. IEEE Power Engineering Society Winter Meeting*, New York, NY, January 2002, vol. 2, pp. 1350–1353.

[31] G. Cardoso, J. G. Rolim, and H. H. Zurn, "Application of neural-network modules to electric power system fault section estimation," *IEEE Trans. Power Delivery*, vol. 19, no. 3, pp. 1034–1041, July 2004.

[32] S. Haykin, *Neural Networks, a Comprehensive Foundation*, p. 2, Prentice Hall, Englewood Cliffs, NJ, 2nd edition, 1999.

[33] K. L. Lo, H. S. Ng, and J. Trecat, "Power systems fault diagnosis using Petri nets," *IEE Proc. Generation, Transmission and Distribution*, vol. 144, no. 3, pp. 231–236, May 1997.

[34] Y. C. Huang, C. M. Huang, C. C. Liao, J. F. Chen, and H. T. Yang, "A new intelligent fast Petri-net model for fault section estimation of distribution systems," in *Proc. International Conference on Power System Technology - PowerCon*, Perth, Australia, December 2000, vol. 1, pp. 217–222.

[35] A. A. El Fergany, M. T. Yousef, and A. A. El Alaily, "Fault diagnosis of power systems using binary information of breakers and relays through DPNs," in *Proc. International Conference on Power System Technology - PowerCon*, Kunming, China, October 2002, vol. 2, pp. 1122–1126.

[36] H. J. Cho and J. K. Park, "An expert system for fault section diagnosis of power systems using fuzzy relations," *IEEE Trans. Power Systems*, vol. 12, no. 1, pp. 342–348, February 1997.

[37] W. H. Chen, C. W. Liu, and M. S. Tsai, "On-line fault diagnosis of distribution substations using hybrid cause-effect network and fuzzy rule-based method," *IEEE Trans. Power Delivery*, vol. 15, no. 2, pp. 710–717, April 2000.

[38] J. Sun, S. Y. Qin, and Y. H. Song, "Fault diagnosis of electric power systems based on fuzzy Petri nets," *IEEE Trans. Power Systems*, vol. 19, no. 4, pp. 2053–2059, November 2004.

[39] J. R. McDonald S. D. J. McArthur, R. Dysko, "The application of model based reasoning within a decision support system for protection engineers," *IEEE Trans. Power Delivery*, vol. 11, no. 4, pp. 1748–1754, October 1996.

[40] J. R. McDonald E. M. Davidson, S. D. J. McArthur, "A toolset for applying model-based reasoning techniques to diagnostics for power systems protection," *IEEE Trans. Power Systems*, vol. 18, no. 2, pp. 680–687, May 2003.

[41] A. Aldea G. Tornielli T. Wyatt A. Jolley M. Chantler, P. Pogliano, "The use of fault-recorder data for diagnosing timing and other related faults in electricity transmission networks," *IEEE Trans. Power Systems*, vol. 15, no. 4, pp. 1388–1393, November 2000.

[42] T. S. Sidhu, "Relay software models for use with electromagnetic transient analysis programs," *ELECTRA*, vol. 1, no. 226, pp. 48–54, June 2006.

[43] A. Domijan and M. V. Emami, "State space relaying modeling and simulation using the electromagnetic transient program and its transient analysis of control capability," *IEEE Trans. Energy Conversion*, vol. 5, no. 4, pp. 667–702, December 1990.

[44] A. K. S. Chaudhary, K. S. Tam, and A. G. Phadke, "Protection system representation in the electromagnetic transients program," *IEEE Trans. Power Delivery*, vol. 9, no. 2, pp. 700–711, April 1994.

[45] C. H. Kim, M. H. Lee, R. K. Aggarwal, and A. T. Johns, "Educational use of EMTP MODELS for the study of a distance relaying algorithm for protecting transmission lines," *IEEE Trans. Power Systems*, vol. 15, no. 1, pp. 9–15, February 2000.

[46] J. Y. Heo, C. H. Kim, K. H. So, and N. O. Park, "Realization of distance relay algorithm using EMTP MODELS," in *Proc. International Conference on Power System Transients - IPST*, New Orleans, LA, September 2003.

[47] H. W. Dommel, *Electromagnetic Transients Program (EMTP) Theory Book*, Bonneville Power Administration, Portland, OR, 1986, [Online] Available: http://www.eeug.org/files/secret/EMTP_TheoryBook.

[48] CanAm EMTP User Group, *Alternative Transient Program (ATP) Rule Book*, Bonneville Power Administration, Portland, OR, 2001, [Online] Available: http://www.eeug.org/files/secret/ATP_RuleBook.

[49] L. Dube, *MODELS in ATP, Language Manual*, 1996, [Online] Available: http://www.eeug.org/files/secret/MODELS.

[50] J. Mahseredjian, G. Benmouyal, and X. Lombard, "A link between EMTP and MATLAB for user-defined modeling," *IEEE Trans. Power Delivery*, vol. 13, no. 2, pp. 667–674, April 1998.

[51] M. Kezunovic and B. Kasztenny, "New SIMULINK libraries for modeling digital protective relays and evaluating their performance under fault transients," in *Proc. International Conference on Power System Transients - IPST*, Budapest, Hungary, June 1999.

[52] The MathWorks, Inc., Natick, MA, *Getting Started with MATLAB, Version 7*, September 2006, [Online] Available: http://www.mathworks.com/access/helpdesk/help/pdf_doc/matlab/getstart.pdf.

[53] The MathWorks, Inc., Natick, MA, *SimPowerSystems User's Guide, Version 4*, September 2006, [Online] Available: http://www.mathworks.com/access/helpdesk/help/pdf_doc/physmod/powersys/powersys.pdf.

[54] The MathWorks, Inc., Natick, MA, *Using SIMULINK, Version 6*, September 2006, [Online] Available:http://www.mathworks.com/access/helpdesk/help/

pdf_doc/simulink/sl_using.pdf.

[55] X. Li, W. Yu, and F. L. Rosano, "Dynamic knowledge inference and learning under adaptive fuzzy Petri net framework," *IEEE Trans. Systems, Man, Cybernetics, C*, vol. 30, no. 4, pp. 442–450, November 2000.

[56] X. Luo and M. Kezunovic, "Interactive protection system simulation using ATP MODELS and C++," in *Proc. IEEE Transmission and Distribution Conference*, Dallas, TX, May 2006, vol. 1, pp. 874–879.

[57] Japanese ATP User Group, *Introduction to ATPDraw+ATP/MinGW Easy Installer*, [Online] Available: http://www.jaug.jp/mingw32/atpeasye.html.

[58] MinGW User Group, *Introduction to MinGW*, [Online] Available: http://www.mingw.org.

[59] SINTEF Energy Research, Trondheim, Norway, *ATPDRAW Users' Manual, Version 3.5*, 2002, [Online] Available: http://www.eeug.org/files/secret/ATPDraw.

APPENDIX A

GE UR SERIES RELAY FILE TRANSFER MECHANISM

GE UR series relays support Modicon Modbus RTU protocol over their RS232 or RS485 serial links. A UR relay always acts as a slave device, which only listens and responds to requests issued by a master computer. Communications takes place in packets which are groups of asynchronously framed byte data. The master transmits a packet to the slave and the slave responds with a packet. The end of a packet is marked by "dead-time" on the communications line. Table XXIII describes general format of both transmit and receive packets.

Modbus officially defines function codes from 1 to 127. Only two are used by UR relays for file transfer. Table XXIV summarizes these two function codes.

The UR relays have a generic file transfer facility, meaning that users can use the same method to obtain all of the different types of files such as oscillography file, event report and fault report. The Modbus registers that implement file transfer are found in the "Modbus File Transfer (Read/Write)" module, starting at address 0x3100 and "Modbus File Transfer (Read Only)" module, starting at address 0x3200. They are listed in Table XXV. The following steps are used to read a file from a UR relay:

1. Write the filename to the "Name of file to read" register in "Modbus File Transfer (Read/Write)" module using a write multiple registers command. The file name format of different file types is listed in Table XXVI.

2. Repeatedly read all the registers in "Modbus File Transfer (Read Only)" module using a read multiple registers command. The "Character position of current

Table XXIII. Modbus RTU packet format

Frame Name	**Size**	**Description**
Slave Address	1 byte	Each slave on a communication bus must have a unique address which is programmable from 1 to 254. A master transmit packet with slave address 0 indicates a broadcast command
Function Code	1 byte	This tells the slave what action to perform.
Data	N bytes	This includes a variable number of bytes depending on the function code. This may include actual values, settings, or addresses sent by the master to the slave or by the slave to the master.
CRC	2 bytes	This is an error checking code generated by a 16-bit cyclic redundancy check algorithm (CRC-16).
Dead Time	3.5 bytes transmission time	A packet is terminated when no data is received for a period of 3.5 byte transmission time. The transmitting device must not allow gaps between bytes longer than this interval.

block within file" register is initially zero and thereafter indicates how many bytes have been read so far. The "Size of currently-available data block" register indicates the number of bytes of data remaining to read, to a maximum of 244. The "Block of data" registers contain file data. The entire block behaves like a stack, which is updated by next block of file data after each reading of the current one.

3. If a block of file data needs to be re-read, only the "Size of currently-available data block" and "Block of data" registers should be read. The file pointer is only incremented when the "Character position of current block within file" register is read, so the same block of file data will be returned as was read in the previous operation.

4. Keep reading until the "Size of currently-available data block" register is smaller than 244, the number of bytes of data read from data block each time. This condition indicates end of file.

Table XXIV. Modbus RTU function codes for file transfer

Function Code	GE Relay Definition	Description
0x04	Read actual values or settings	This function code allows the master to read one or more consecutive data registers (actual values or settings) from a UR relay.
0x10	Write multiple settings	This function code allows the master to modify the contents of a one or more consecutive setting registers in a UR relay.

Table XXV. UR relay registers used for file transfer

Address	Name
0x3000	Oscillography number of triggers
0x3100 to 0x3127	Name of file to read (40 registers)
0x3200	Character position of current block within file
0x3202	Size of currently-available data block
0x3203 to 0x327C	Block of data (122 registers)

Table XXVI. UR relay file name format

File Type	Format
COMTRADE Oscillography File	OSCnnn.HDR, OSCnnn.CFG, OSCnnn.DAT (Replace nnn with the desired oscillography trigger number)
Event Report	EVTnnn.TXT (Replace nnn with the desired starting record number)
Fault Report	faultReportnnn.TXT (Replace nnn with the desired fault report number)

If only the files corresponding to the latest event are retrieved, the current event number recorded by the relay must be known. This can be done by reading the "Oscillography number of triggers" at address 0x3000.

Fig. 42 is the program flow chart for UR relay file retrieval in polling mode.

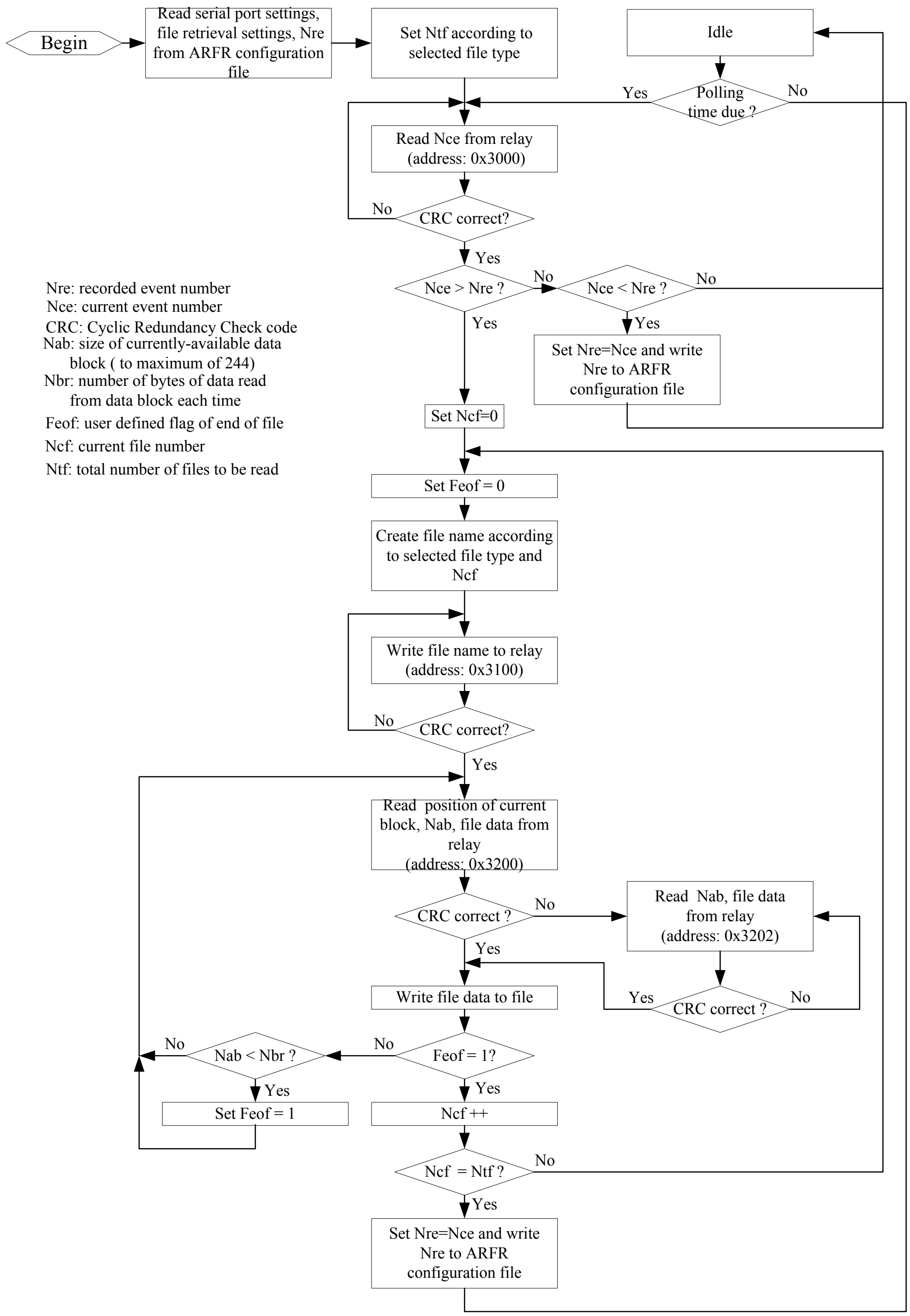

Fig. 42. Program flow chart for UR relay file retrieval in polling mode

APPENDIX B

SEL421 RELAY FILE TRANSFER MECHANISM

SEL421 relays support SEL ASCII Command protocol over their RS232 serial links. Several SEL ASCII commands are involved in initiation of a file transfer from the relay to external software. Table XXVII describes these commands and their functions regarding to initiation of a file transfer. The file name format of different file types is listed in Table XXVIII.

If only the files corresponding to the latest event are retrieved, the current event number recorded by the relay can be determined from the summary report responding to the "SUM" command at each polling time in the polling mode of file retrieval. The summary report can also be automatically sent by the relay to a serial port after a new event occurs if the auto-message function is enabled for the serial port. This function can be used for the report by exception mode of file retrieval.

Once a file transfer is initiated by the "FILE READ EVENTS filename" command, the Ymodem protocol is used to perform the file transfer process. The Ymodem protocol is a receiver driven, asynchronous, 8 data bit protocol. When the receiver sends a byte for initiation, positive acknowledgment or negative acknowledgment, the sender responds with a packet which is a group of byte data. In the SEL421 file transfer process, the receiver refers to the ARFR program and the sender refers to the SEL421 relay. The bytes sent by a receiver are described in Table XXIX. Table XXX describes general format of a sender packet.

The following steps describe a file transfer process:

1. The receiver first sends a "C" byte to initiate a file transfer. The sender responds with an information packet which contains the file name.

Table XXVII. SEL ASCII COMMANDS for initiation of file transfer

Command	**Description**
ACC	Go to Access Level 1, which is a monitoring level
SUM	Return the most recent event summary in order to get the current event number
FILE READ EVENTS filename	Initiate a file transfer

Table XXVIII. SEL421 file name format

File Type	**Format**
COMTRADE Oscillography File	HR_nnnnn.HDR, HR_nnnnn.CFG, HR_nnnnn.DAT (Replace nnnnn with the desired event number)
Compressed 8 Sample/Cycle Event Report	C8_nnnnn.TXT (Replace nnnnn with the desired event number)
Compressed 4 Sample/Cycle Event Report	C4_nnnnn.TXT (Replace nnnnn with the desired event number)

Table XXIX. Ymodem receiver bytes

Symbol	**Value**	**Description**
C	0x43	Character "C"
ACK	0x06	Positive acknowledgment
NAK	0x15	Negative acknowledgment

Table XXX. Ymodem sender packet format

Frame Name	**Size**	**Description**
Head	1 byte	SOH (0x01) indicates a 128 bytes data frame length; STX(0x02) indicates a 1024 bytes data frame length; EOT (0x04) indicates the end of transmission. No other frames follow EOT.
Packet Number	1 byte	0x00 indicates an information packet, which contains file name in the data frame; 0x01 to 0xFF indicates a file data packet. The number starts at 0x01, increments by 1 and wraps from 0xFF to 0x01.
Packet Number Complement	1 byte	0xFF minus the packet number
Data	128 bytes or 1024 bytes	File name data or file data
CRC	2 bytes	This is an error checking code generated by a 16-bit cyclic redundancy check algorithm (CRC-16) for the data frame.

2. After the filename has been transmitted, the receiver asks for file data by sending another “C”. The sender responds by sending a data packet. If the receiver receives the packet and decides the received data are correct by checking the CRC code, it sends back an ACK. If the receiver does not received the packet in a certain time limit or decides the received data are not correct, it sends back a NAK. If the sender receives an ACK in a certain time limit, it sends the next data packet. If the sender does not receive an ACK in a certain time limit or receives a NAK, it re-sends the data packet. This process continues until the sender sends an EOT to indicate the end of transmission of file data.

3. After the file has been transmitted, the receiver asks for the next file by sending a “C”. Transmission of an information packet with null filename by the sender indicates termination of the entire file transfer process.

Fig. 43 and Fig. 44 are the program flow charts for SEL421 relay file retrieval in polling mode and report by exception mode respectively.

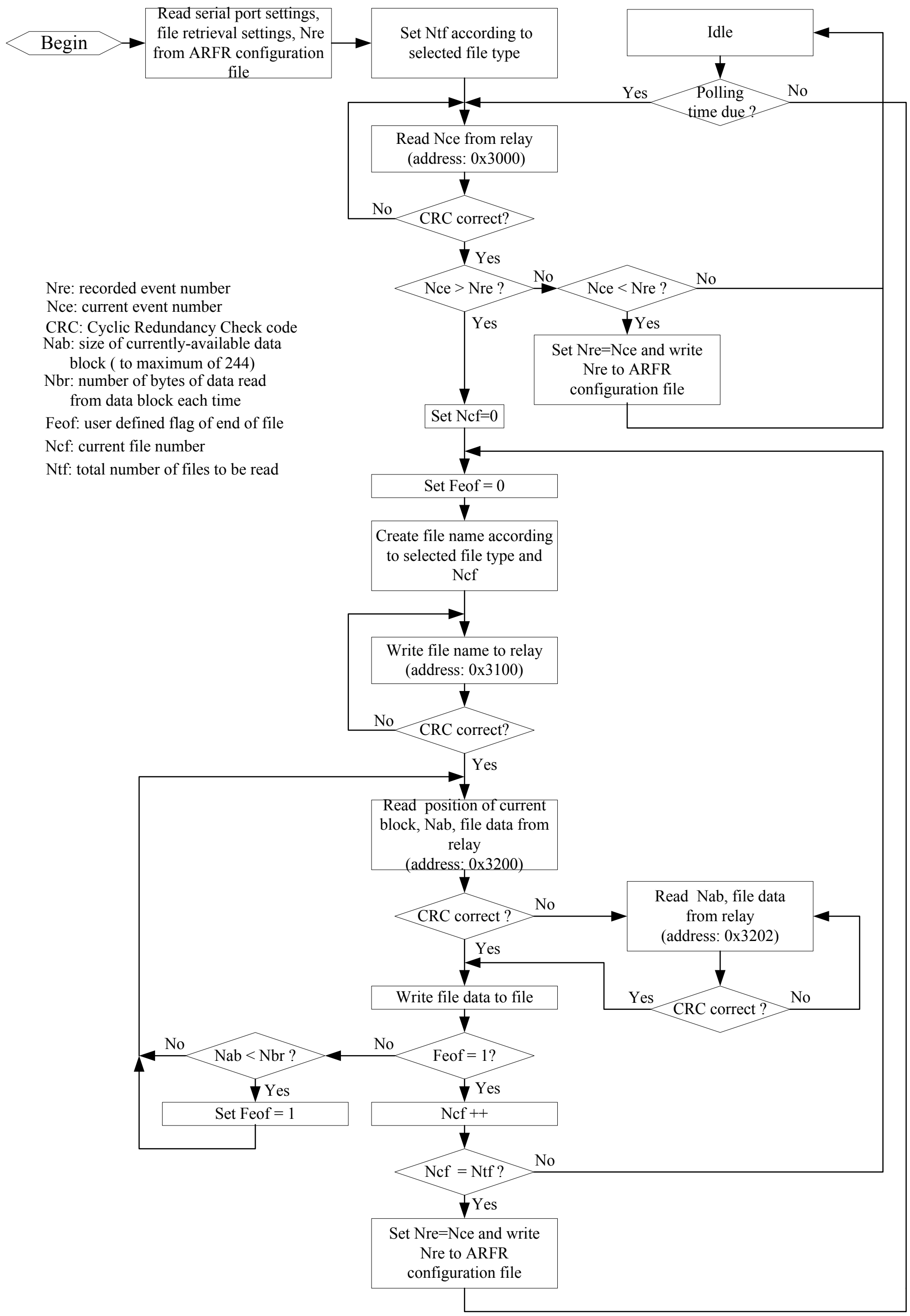

Fig. 43. Program flow chart for SEL421 relay file retrieval in polling mode

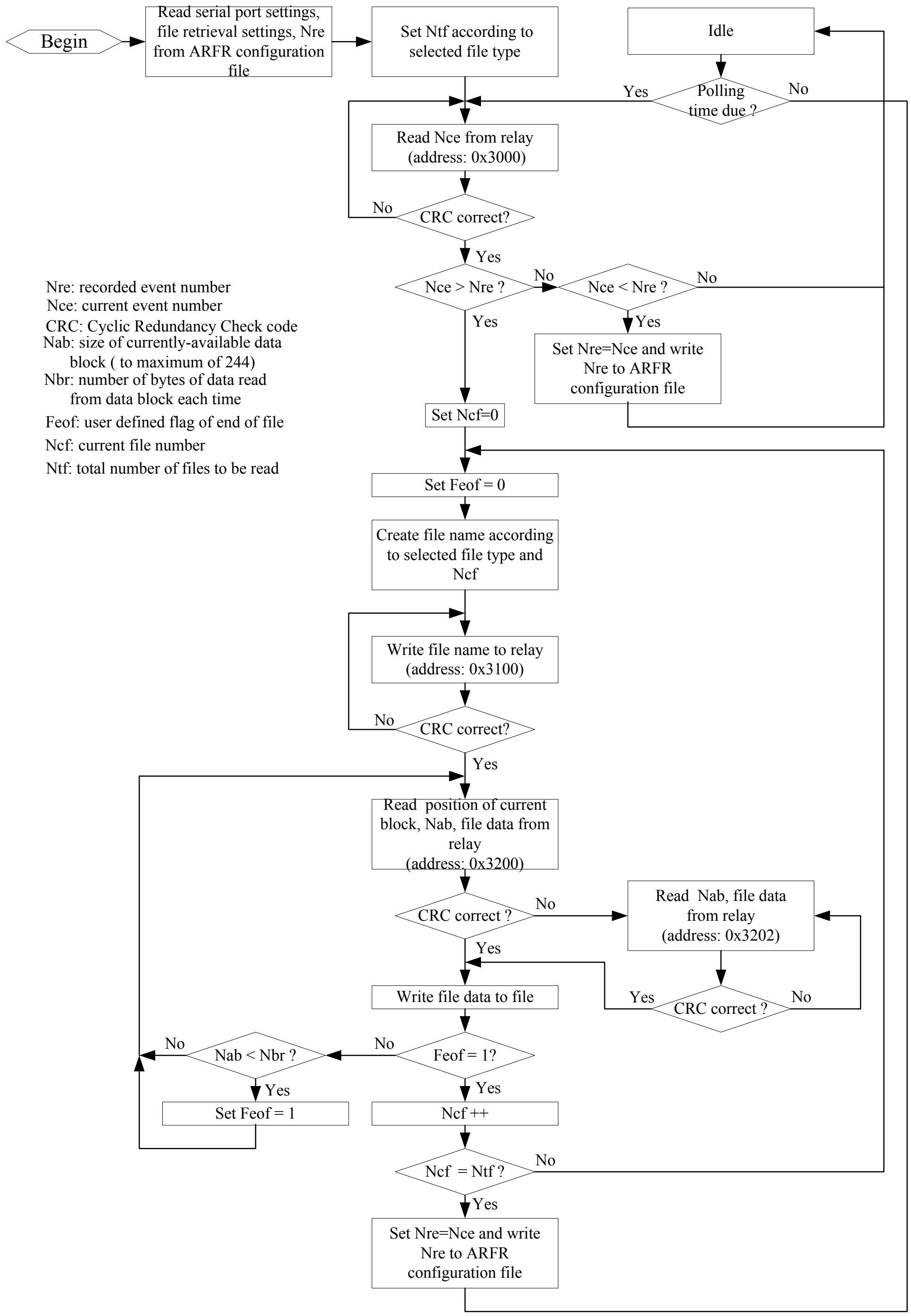

Fig. 44. Program flow chart for SEL421 relay file retrieval in report by exception mode

VITA

Xu Luo received his B.S. and M.S. degrees from Xi'an Jiaotong University, Xi'an, China, both in electrical engineering in 1999 and 2002 respectively. He has been at Texas A&M University and received his Ph.D. degree in May 2007. His research interests are power system protection, substation automation, and artificial intelligence applications in power system monitoring, control and protection. His permanent address is: Department of Electrical and Computer Engineering, Texas A&M University, 214 Zachry Engineering Center, College Station, TX 77843-3128.

The typist for this thesis was Xu Luo.