GENERIC PROTECTION ANALYZER FOR POST-PROCESSING OF EMTP SIMULATIONS

Khalil El-Arroudi, Geza Joos, Donald McGillis, and Reginald Brearley

Department of Electrical and Computer Engineering, McGill University, 3480 University Street, Montreal, Quebec, Canada, H3A-2A7 (e-mail: kelarr@po-box.mcgill.ca, joos@ece.mcgill.ca, donmcgillis@sympatico.ca, reginald@pubnix.net)

***Abstract* – This paper presents a Generic Protection Analyzer (GPA) to run and analyze EMTP protection system simulations. This GPA is a generic software protection model that exchanges data with EMTP programs through the input case data files and the output simulation files. The proposed technique results in less computational complexity and faster execution due to the manner in which the individual relays are represented in the GPA. This is the key feature of this proposed technique. The proposed GPA was implemented using the MATLAB object-oriented programming language and linked with the EMTDC simulation program.**

Keywords – Power System Protection, Power System Relaying, Power System simulation, Simulation Relay Modeling, EMTP, EMTDC, MATLAB

I. INTRODUCTION

Protection system studies ensure that the protection equipment is selected to withstand the operating levels, the protective functions are dependable and secure, and the relays are set to respond to faults within their zones of protection. These studies should include both transient and steady state analyses. Therefore, Electromagnetic Transient Programs (EMTP) are an important tool for the simulation of the protection systems.

EMTP protection system simulation consists of two main modeling tasks namely the modeling of power system network and the modeling the protection system. The protection system simulations include relay performance testing, relay mis-operation analysis, and protection system design. In such simulations where it is required to analyze contingencies in a network containing hundreds of relays on different branch-ends, the burden of modeling and configuring of each of these relays introduces time and effort constraints to protection engineers in conventional procedures. Furthermore, the simulation computational complexity will depend on the number and the functions of the modeled relays within the simulated network. In all these simulations, the relay model can be operated in either closed-loop or open-loop simulations.

In the closed loop simulations, the relay model is combined with EMTP simulations. The relay model can be implemented using TACS (Transient Analysis of Control System) [1], ATP-MODELS [2], EMTP Add-in custom FORTRAN code [3, 4, 5], or MATLAB functions [6]. In these simulations, the relay is modeled to work interactively within EMTP and therefore the modeled relay operates switches in the EMTP network simulations.

On the other hand, the open loop simulations (playback) do not allow the modeled relay to operate switches in the EMTP network simulations. In these simulations, the relay model is stand-alone and can be implemented by any programming language including C, FORTRAN, MATLAB, etc. The output simulation files from the EMTP are converted to the required format which are then fed to the relay model to analyze these files.

In large networks modeling of significant number of relays in TACS is sometimes inconvenient, and open-loop simulation is possibly one of the most effective ways to analyze relay operation. However, there is also a drawback of open-loop simulations in that it is impossible to model situations such as fault clearing and reclosing. In order to model such situations in open-loop simulation, switches should be operated in "manual" mode by adjusting opening and closing times rather than controlling the switches by TACS or MODELS as in closed-loop simulations.

Several studies have been published on using the EMTP for protection system simulations. The early work in [7] described a procedure for analyzing fault detection algorithms using EMTP. The study reported in [8] presented a method to test the transient performance of a digital relay by modeling relay components within EMTP program. In [3], Models for current transformers and capacitive voltage transformers within EMTP were developed and incorporated in the EPRI/DCG EMTP Version 2.0. The technique in [9], presented an approach in which the relay simulation combines the advantages of both the closed-loop and open-loop approaches.

In the concept proposed in this paper, one Generic Protection Analyzer (GPA) is created to serve as a generic software computational engine which can analyze EMTP results. This GPA acquires its input voltage and current information by capturing the reading of the target relays (the required relays under study) located at given branch-ends in the simulated network. The target relays exist virtually in the simulated network and carry the tag of their corresponding location in the simulated network. These target relays take their settings and required protective functions by accessing a system database of the network under study. This proposed technique results in less computational complexity and faster execution due to the manner in which the individual relays are represented in the GPA. This GPA operates on open-loop simulations, but the concept can be extended to operate on closed-loop simulations.

II. THE CONCEPT OF THE PROPOSED GPA

The concept of the GPA is based on the creation of virtual relays to represent the actual relays in a network. The mechanism whereby each virtual relay is created is, in fact, a software model which acquires the voltage and current information at a given target relay location (the required relay under study) for some specific event and transforms this information into protection system parameters through its mathematical algorithms. These parameters represent the response of the system at target relay to the each simulated event such parameters are overvoltage (magnitude and duration), overcurrent (magnitude and duration), transient apparent impedances, zero and negative sequence unbalances, etc.

The purpose of the GPA is to analyze the consequences of every simulated event in the EMTP and generate the results in terms of protection system parameters. These consequences are acquired by the GPA in terms of voltages and currents for a targeted relay location leading to the creation of its corresponding virtual relay. As a result of this action of the GPA, it is not necessary to configure relays in the actual simulated network but only to tag their location in the network through the action of the user.

The relationship between the EMTP, and the GPA is illustrated in Figure 1.

The operation of the GPA concept can be illustrated in Figure 2 for any given target relay R_x. The following steps describe this operation:

1. The user chooses the target relay R_x for which the relay response is required for each simulated event.
2. A series of events is simulated in order to determine the consequences at this target relay location.
3. The consequences of these events in terms of voltage and current are stored into data files.
4. At the same time, the GPA creates a virtual relay R_x which acquires the data stored in the data file and its properties from the corresponding target relay data present in the system database.
5. Thus, the virtual relay R_x is an instance of the GPA from which the system parameters of the target relay are determined.
6. The system parameters of the virtual relay are determined from an analysis of the input data using the algorithms inherited from the GPA model.
7. As a result, the virtual relay represents the actual relay R_x and is, in fact, a GPA for R_x.
8. Thus the GPA creates a model of itself for each physical relay in the network, the output of which are the system parameters.
9. The user then compares these system parameters with the relay characteristics for each simulated event.

The features of the proposed Generic Relay are summarized in Table 1. This table indicates that the GPA modeling results in less engineering time and effort, as well as less computational complexity. This is because the relay configuration is done dynamically at the creation of each virtual relay.

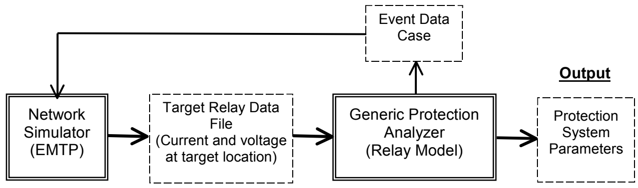

Fig. 1 The relationship between the EMTP and the GPA

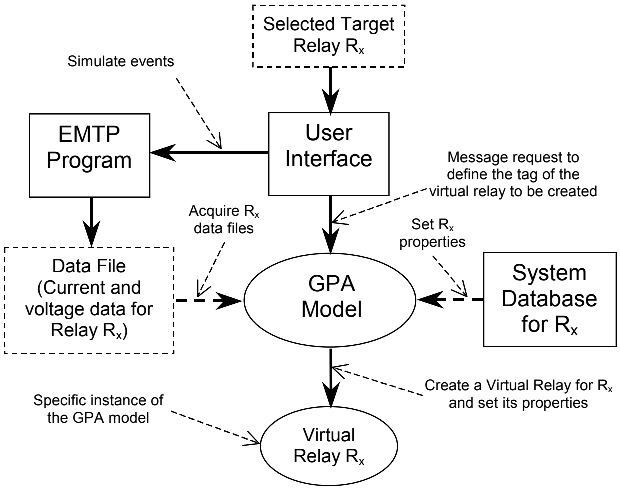

Fig. 2 The concept of the GPA

Table I The Features of the GPA

Item	Generic Relay Features
Relay representation	No physical relays are modeled within network data cases. Instead, only tagged current and voltage monitors are defined.
Relay Functions	Relay functions combined in one single package relay to serve as a single multifunction relay.
Relay configuration	Relay configuration is done dynamically at the creation of each virtual relay.
Reusability	All created virtual relays use (share) one computational engine which is embedded in the GPA model.
Modularity (All in one)	Each created virtual relay acquires the properties of the protected equipment where this relay is located. Such data includes line data, circuit breaker data, series equipment data, stability data, etc). Therefore, each created virtual relay is, in fact, a compendium of related data at the specific location.

III. THE GPA COMPONENTS

The design of the GPA is based on cooperation of basic units to calculate system parameters through protective relaying algorithms. Such basic units include: current unit, voltage unit, and timer unit. Each of these units has a structure which has states and functions.

The algorithms that operate on the abovementioned basic units can include: current and voltage digital filters, time-overcurrent, instantaneous overcurrent, under/overvoltage, symmetrical component filtering, directional power, apparent impedance, fault classification, current differential, phase comparison, overcurrent time durations, undervoltage time durations, and transient time duration.

Figure 3 shows the components of the proposed GPA which contains the following.

a) *The Inputs* to the GPA are a) the target relay current and voltage data files from the EMTP program and b) the target relay database.

b) *The output* of the GPA is a list of system parameters for all the prescribed events that have been simulated.

c) *The Relay Configuration* is performed through the user interaction to: a) set the tag of the virtual relay to be created for the target relay, and b) select the required algorithms for analysis of the case study.

d) *The GPA Manager* is a program that manages all the cooperation between the units of the GPA in order to implement the algorithms of the GPA. In addition, it acts as an interface of the GPA to the user through which the user can communicate with the GPA. In this respect, the GPA receives message requests from the user and performs the required computations in response.

e) *The GPA Computational Engine (GPACE)* processes protection algorithms that calculate the system parameters. The computational engine is a multifunction protection software model developed specifically as a part of this study. This multifunction model can be used for performing a variety of protection system studies. The protection algorithms within this software model are represented by both transient generic model (which calculates relay response at each instant of time for specific current and voltage) as well as phasor model (uses phasor equations). The selection between these types of modeling depends on the event to be studied. For example, the steady-state events require phasor model while the transient events require transient model. This computational engine is used by each virtual relay for the calculation of the system parameters associated with this virtual relay. The general form of the algorithms that compute the system parameters can be written as follows:
$\{\boldsymbol{SP}\} = f(\boldsymbol{I}, \boldsymbol{V}, \boldsymbol{T})$

Where:
$\boldsymbol{SP}$ is a set of the GPA system parameters
$\boldsymbol{I}$ is the current unit object
$\boldsymbol{V}$ is the voltage unit object
$\boldsymbol{T}$ is the timer unit object

The following three examples show how algorithms are implemented from the cooperation of the GPA units:

- time-overcurrent algorithm:
  $\{\boldsymbol{SP}_{toc}\} = \boldsymbol{f}_{toc}(\boldsymbol{I}, \boldsymbol{T})$
  and $\boldsymbol{T} = f(\boldsymbol{I}, k1, k2, \ldots, kn)$
  where *k1* to *kn* are constants which govern the current-time characteristics of the time-overcurrent algorithm.

- RMS detector algorithms:
  These algorithms estimate the rms magnitude and phase angle of the currents and voltages. They can contain different mathematical models such as Discrete Fast Fourier Transform, Walsh functions, Least Squares methods, Kalman filters which are implemented in the proposed GPA.
  $\{\boldsymbol{SP}_{DFI}\} = \boldsymbol{f}_{DF}(\boldsymbol{I}, f, N)$ is the current digital filtering
  $\{\boldsymbol{SP}_{DFV}\} = \boldsymbol{f}_{DF}(\boldsymbol{V}, f, N)$ is the voltage digital filtering
  Where:
  N = is the number of samples per cycle.
  f =is the fundamental frequency [Hz].
  $\{\boldsymbol{SP}_{DFI}\} = \{I_{a1}, I_{b1}, I_{c1}, I_{n1}, \varphi_{a1}, \varphi_{b1}, \varphi_{c1}, \varphi_{n1}\}$ is a set of fundamental frequency magnitude and phase angle of the currents.

- Apparent Impedance algorithm:
  It calculates the apparent impedances of the fault loops based on the voltage and current units for different faults such as phase-to-ground, phase-to-phase, and three-phase.
  $\{\boldsymbol{SP}_Z\} = \boldsymbol{f}_Z(\boldsymbol{I}, \boldsymbol{V})$
  $\{\boldsymbol{SP}_Z\}$ is a set of all calculated loop apparent impedances.

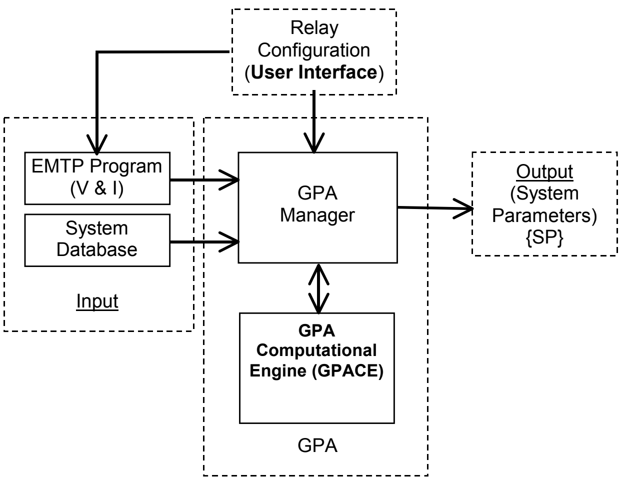

Fig. 3 GPA Components

IV. THE IMPLEMENTATION OF THE GPA

The implementation of the proposed GPA is based on the Object-Oriented design technique. This technique has been selected because it allows: a) building of systems that evolve over time (open system architecture), b) reusing of classes and objects and promoting a great degree of code reuse that avoids replication of codes and c) extending classes by inheritance to achieve reuse at the class level.

Figure 5 shows the implementation of the proposed GPA using the MATLAB [10].

V. CASE STUDY

The purpose of this case study is to demonstrate the effectiveness of the proposed GPA.

The system studied is shown in Figure 4. This case study was obtained from [11] with the parameters stored in a system database file.

The system database contains data for every component in the actual network shown in Figure 4. This data can include data for generation systems, transmission systems, distribution systems, and protection systems as follows:

a) data for generation and transmission systems (the values are given in per-unit):

- $S_{base} = 500MVA$ (Base Power).
- $V_{base} = 220Kv$ (Base Voltage).
- $f = 50\ Hz$
- Line Length = 400 Km.
- $E_S = 1.0ej^{\delta S}$ (Sending-End source voltage).
- $E_R = 1.0ej^{\delta R}$ (Receiving-End source voltage).
- $Z_f = 0.04$ (fault resistance).
- *TL* = Transmission line (PI-model)
- *SC* = Series Capacitor
- Z_S and Z_R are sending-end and receiving-end source impedances.
- $R_{0L} = 0.001\ pu/Km$ (line zero sequence resistance).
- $R_{1L} = 0.00033\ pu/Km$ (line positive sequence resistance).
- $X_{0L} = 0.0114\ pu/Km$ (line zero seq. inductive reactance).
- $X_{1L} = 0.00382\ pu/Km$ (line pos. seq. inductive reactance).
- $X_{0CL} = 4504\ pu.Km$ (line zero seq. cap. reactance).
- $X_{1CL} = 2252\ pu.Km$ (line pos seq. cap. reactance).
- $X_{0S} = 0.153\ pu$ (sending-end zero seq. reactance).
- $X_{1S} = 0.382\ pu$ (sending-end pos. seq. reactance).
- $X_{0R} = 0.153\ pu$ (receiving-end zero seq. reactance).
- $X_{1R} = 0.382\ pu$ (receiving-end pos. seq. reactance).
- $X_{1SC} = 0.6089\ pu$ (series capacitor reactance).
- *line_conductor_rated_current = 1200*
- *line_breaker_rated_current = 1500*
- *line_switch_rated_current = 1500*
- *line_trap_rated_current = 1000*
- *line_series_compensator_rated_current = 1600*

b) Data for protection relays is given in data files. For example the protection data for the relay R1 shown in Figure 4 includes the following sample data:

Relay Database file: *R1.dat*

- *relay_tag : R1*
- *station_name: S*
- *line_name: Line1*
- *line_length: 400*
- *rated_freq: 50*
- *current_transformer_ratio: 1200/1*
- *voltage_transformer_ratio: 220000/110*
- *minimum_load_current: 1000*
- *maximum_load_current: 1500*
- *system_rated_line_voltage: 220000*
- *line_characteristic_angle: 85*
- *overcurrent_pickup_setting: 1.0*
- *distance_relay_characteristic: mho*
- *zone_one_setting: 41*
- *zone_two_setting: 58*
- *zone_three_setting: 96*

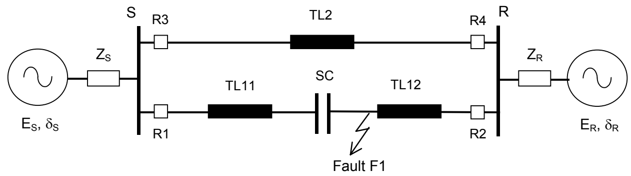

Figure 4 The power system of the case study

The case study consists of simulation of a transient single-phase-to-ground fault at F1 shown in Figure 4. The fault is initiated at t = 0.25 sec, cleared at t = 0.3 sec. The required target relay was selected to be the relay R1 shown in Figure 4. The required analyses of this target relay are a) transient fault current, b) transient voltage in the unfaulted phases, and c) the transient impedances seen by this relay.

The simulation time and step size were selected 1.0 sec and 50e-6 sec respectively. Some of the results of this case study are shown in Figure 5.

V. CONCLUSIONS

In this paper, the concept of the Generic Protection Analyzer (GPA) is presented along with its components. The concept of the GPA is to create a virtual relay for each target relay. The virtual relay thus created is, in fact, the GPA for this targeted relay since it inherits all the properties and behavior of the GPA. The essential feature of the GPA is a mechanism whereby the prescribed events simulated in the EMTP program are analyzed and the generation of the protection system parameters of the relays are then passed to the user.

ACKNOWLEDGEMENT

The authors wish to express their gratitude to the General Electricity Company GECOL (Libya), Hydro Quebec (Canada), and the National Science and Engineering Research Council (NSERC) of Canada for their support in the preparation of this work.

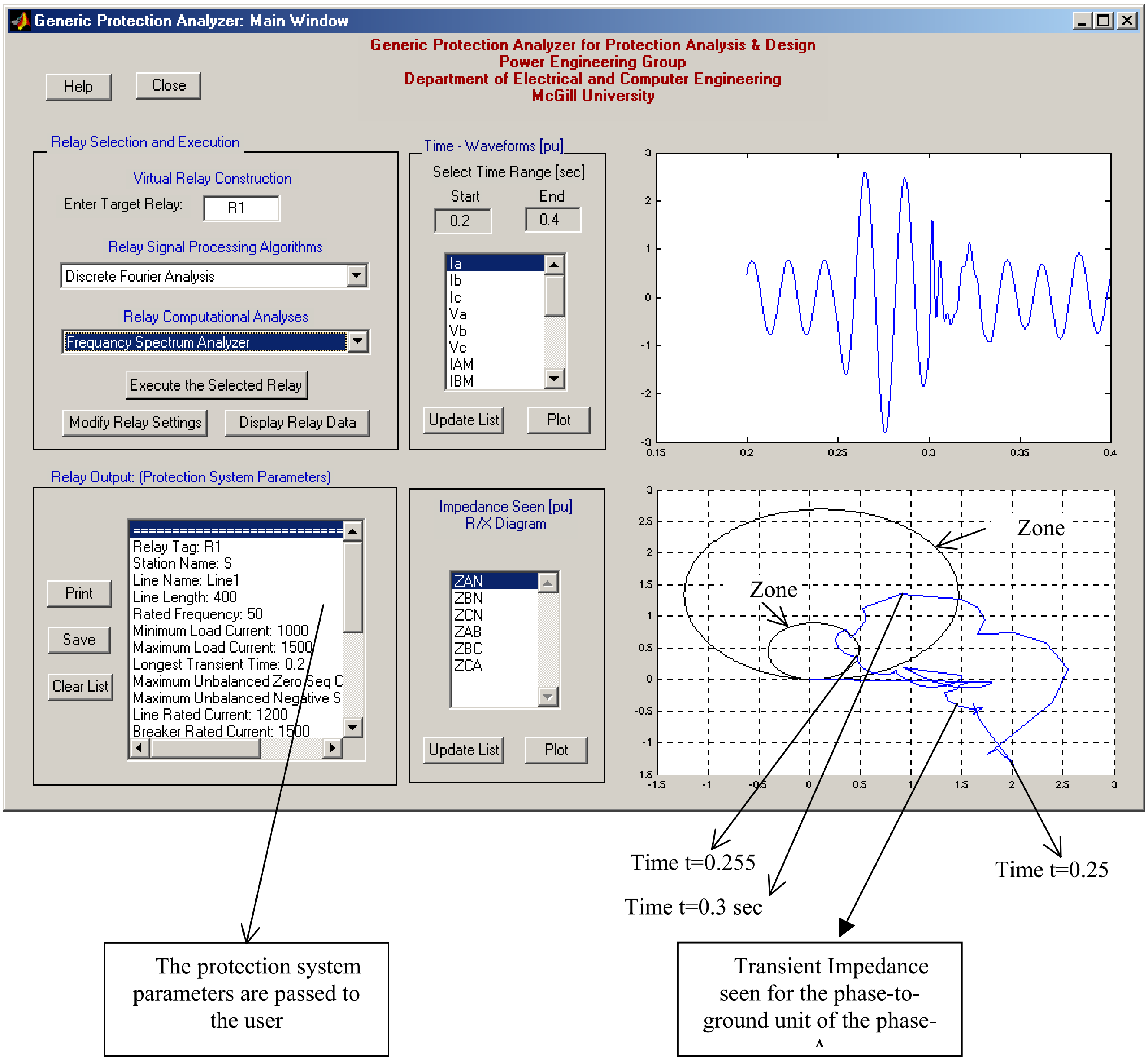

Fig. 5 Implementation of the proposed GPA using the MATLAB program

REFERENCES

[1] L. Dube, H. Dommel, "Simulation of Control System in an Electromagnetic Transient Program with TACS, IEEE Trans. on Power Industry and Computer Applications, 1977.

[2] CAN/AM EMTP User Group, Models in ATP-Rule Book, Portland, Oregon, Aug. 1995.

[3] A. Chaudhary, K. Tam, and A. Phadke, "Protection system representation in the Electromagnetic Transients Program," IEEE Trans. on Power Delivery, Vol.9, No.2, April 1994, pp700-711.

[4] J. N. Peterson, R. Wall, "Interactive Relay Controlled Power System Modeling," IEEE Trans. on Power Delivery, Vol.6, No.1, Jan. 1991, pp96-102.

[5] L. X. Bui, et al, "EMTP TACS-FORTRAN Interface Development for Digital Control Modeling", IEEE Trans. on Power System, Vol. 7, No. 1, Feb. 1992.

[6] Jean Mahseredjian, et al, "A link between EMTP and MATLAB for user-defined modeling,", IEEE Trans. on Power Delivery, Vol.13, No.2, April 1998, pp667-674.

[7] B. Dixon, H. Dommel, "Relay Studies With a Computer Program for Electromagnetic Transients", Proceedings, Western Protective Relay Conference, Washington State University, 1975.

[8] R. E. Wilson, J. Nordstom, "EMTP Transient Modeling of a Distance Relay and a Comparison with EMTP Laboratory Testing," IEEE Trans. on Power Delivery, Vol.8, No.3, July 1993 pp984-992.

[9] M. Kezunovic, Q. Chen, "A Novel Approach for Interactive Protection System Simulation," IEEE Trans. on Power Delivery, Vol.12, No.2, April 1997, pp668-674.
[10] MATLAB, Version. 6.1 Reference Manual, MathWorks, Inc, 2001.
[11] H. W. Dommel, "Introduction to the Use of MicroTran and other EMTP versions", University of British Columbia, 1998.
[12] T. S. Sidhu, et al, "Generating Relay Models for Protection and System Studies", IEEE Computer Applications in Power, Vol. 11, No. 4, Oct. 1998, pp 33-38.
[13] R. Folkers, "Determine Current Transformer Suitability Using EMTP Models", 26th Westren Protective Relay Conference, Oct. 26-28, 1999.
[14] Simpson, W. D. Humpage, " Modeling and Simulation for Protection", CIGRE, Jan 11-17, 1995, Paper No. 34-101.
[15] P. G. Mclaren, et al, "Closed Loop Relay testing with Real Time Electromagnetic Transients Simulation", Trans. CEA E&O Div., Vol. 30, 1991, Paper No. 91-SP-154.
[16] B. Dixon, H. Dommel, "Digital Simulations of Transients for Relay Applications", Canadian Electrical Association, Engineering Operating Division, Vancouver, BC, March 1975.
[17] B. W. Garrett, "Digital Simulation of Power System Protection under Transient Condition," Ph.D. Dissertation, University of British Columbia, April 1987.
[18] R. E. Wilson, "A New Method Using Relay Macro-models for the Simulation of the Transient Response of Distance Relays," Ph.D Dissertation, University of Idaho, Moscow, Idaho, Sept. 1992.
[19] A Domijan, et al, " State Space Relay Modeling and Simulation Using the Electromagnetic Transients Program and its Transient Analysis of Control Systems Capability," IEEE Trans. On Energy Conversion, Vol. 5, No. 4, Dec. 1990, pp.697-702.
[20] Working Group C1 of the System Protection sub-committee, "Software Models for Relays," IEEE Trans. On Power Delivery, Vol. 16, No. 2, April 2001, pp. 238-245.
[21] Power System Relaying Committee, " Mathematical Models for Current, Voltage, and Coupling Capacitor Voltage Transformers," IEEE Trans. on Power Delivery, Vol.15, No.1, Jan. 2000, pp62-72.
[22] P. M. Anderson, "Power System Protection," IEEE Power Engineering Society, Sponsor, IEEE Press 1999.
[23] GEC Measurement, Protective Relays Application Guide. GEC, 1987.
[24] Walter A. Elmore, Protective Relaying Theory and Applications, ABB Power T&D Company Inc., 1994.
[25] A. Phadke, J. Thorp, "Computer Relaying for Power System", Research Studies Press Ltd., England, 1988.
[26] EMTDC/PSCAD Simulation Software, Version 3.04, HVDC Research Centre, Manitoba-Hydro, Canada, 1999.
[27] K. Dattari, C++ Effective Object-Oriented Software Constructor, Second edition, Prentice Hall PTR, 2000.