

Testing and Evaluating New Software Solutions for Automated Analysis of Protective Relay Operations

M. Kezunovic, X. Luo, N. Zhang, and H. Song

Abstract—This paper presents the test and evaluation environments for three new software solutions for automated analysis of protective relay operations. For each environment, the test requirements, test solution and benefits are discussed. A comprehensive modeling and simulation environment serving as the common test and evaluation platform for all the three software solutions is proposed.

Index Terms—test, software solution, automated analysis, protective relay

I. INTRODUCTION

New software solutions for automated analysis of protective relay operations include algorithms based on expert systems, neural networks, synchronized sampling, and static and dynamic contingency analysis [1-4]. The goal of these solutions is to distinguish between correct and incorrect operations or identify conditions that may cause unintended operations. Once developed, these solutions need to be thoroughly tested and evaluated. The challenge of test and evaluation lies in implementation of large number of test and evaluation scenarios. This requires methods for easily generating power system disturbances, interfacing relaying systems with power systems, modifying relay settings, inserting user-induced errors, recording and collecting transient waveforms and relay response events, and automatically executing batch tests. Since it is very difficult to implement such test and evaluation in real power systems, it is quite necessary to build modeling and simulation environments which fulfill the test and evaluation requirements. The existing solutions for building such modeling and simulation environments make use of power system models, relaying system models and/or digital simulators. Power system models are mainly developed using electromagnetic transients program (EMTP) while relay models are often developed using transient analysis of control system (TACS) functions of EMTP, MODELS language of alternative transient program (ATP) version of EMTP, or MATLAB [5-9]. Digital simulators are usually used with power system models to implement transient tests on physical relays [10].

This paper presents the modeling and simulation environments built for test and evaluation of new software solutions for automated analysis of protective relay operations [11-13]. These environments have enhanced features which are not possessed by existing solutions. For power system models, simulation setting programs are developed to facilitate fast and easy creation of a large number of disturbance scenarios. For relay models, dynamic relay setting extraction, relay event report generation and user-induced error insertion are implemented to set various relay conditions and reveal detailed relay behavior. For use of digital simulators, the batch test mode is implemented and an automated relay file retrieval program is developed to expedite the test process.

Sections II, III, IV of this paper present the modeling, simulation and test environments for the three algorithms. In each section, the functions of the software solution under test are introduced first. Then the test and evaluation requirements for the software solution are discussed. That is followed by the details of the modeling and simulation environment. Finally, the benefits of the environment are presented. In Section V, based on a summary of the test and evaluation requirements for each software solution and the strength of each corresponding modeling and simulation environment, a comprehensive modeling and simulation platform which may serve to test and evaluate all the three software solutions is proposed. Conclusions are drawn in Section VI.

II. TEST AND EVALUATION ENVIRONMENT FOR DIGITAL RELAY DATA ANALYSIS SOLUTION

Validation of correct operations and diagnosis of incorrect operations of protective relays is a vital task for protection engineers. Through the analysis, protection engineers can evaluate operation performance, identify design deficiency and incorrect settings, and trace component malfunctions. An expert system based solution to automate the analysis of digital protective relay operations, which enables quick and accurate assessment of operation of a large number of digital protective relays was developed [1].

To thoroughly test and evaluate the solution, several requirements should be satisfied. First, various fault scenarios need to be efficiently generated so that the relay to be analyzed can be triggered under different conditions. Second, the relay should be capable of generating files which contain detailed data about how it responds to a given fault. Third, the settings

M. Kezunovic, X. Luo, N. Zhang, and H. Song are with the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843-3128, USA, AREVA, Global Energy Decisions, and Bechtel respectively (e-mails: kezunov@ece.tamu.edu, xu.luo@areva-td.com, nzhang@globalenergy.com, hsong@bechtel.com).

of the relay must be easily modified and various types of malfunction options must be conveniently configured in the relay so that all kinds of user-induced errors can be set to demonstrate the diagnosis capability of the application. To fulfill the test requirements, a “compiled foreign model” approach has been used for modeling a multifunctional digital relay as well as implementing the power system and protection relay interactive simulation. This section presents the modeling and simulation environment developed based on the “compiled foreign model” approach [11].

A. Concept of Compiled Foreign Model

MODELS language is a general-purpose description language of ATP [7, 8]. It provides a pre-defined interface to link a “foreign model” which is written in other programming languages “foreign” to the ATP. A “foreign model” must be compiled and linked to the ATP before it can be called by MODELS. The newly developed ATP/MinGW package has convenient tools to compile a “foreign model” written in FORTRAN and C/C++, and link it with the ATP [14].

B. Digital Protective Relay Model

A multifunctional digital protective relay model is developed using C++ language based on the “compiled foreign model” mechanism. Besides possessing the features which satisfy the common functional requirements for components, interface, and protection functions, the relay model is also capable of reading settings from a setting file, enabling one to insert user-induced errors and generating relay reports.

A hybrid approach is adopted to realize the relay functions in both the MODELS section of ATP data case file and the C++ “foreign model”. The interfacing to the power system model, analog signal filtering, and 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”, an “inheritance” modeling architecture is used. 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. 1 illustrates the programming structure of the relay model.

C. Simulation Procedure

In the procedure of power system and protection system interactive simulation, the user will first prepare an ADP file of the ATPDraw package which models the power system, an MODELS data case which specifies the interface between the power system and relay models, and a C++ source code file which models the relay [15]. ATPDraw, ATP/MinGW, and a simulation setting program will be utilized in the simulation procedure. Fig. 2 illustrates the entire simulation procedure. The description is as follows:

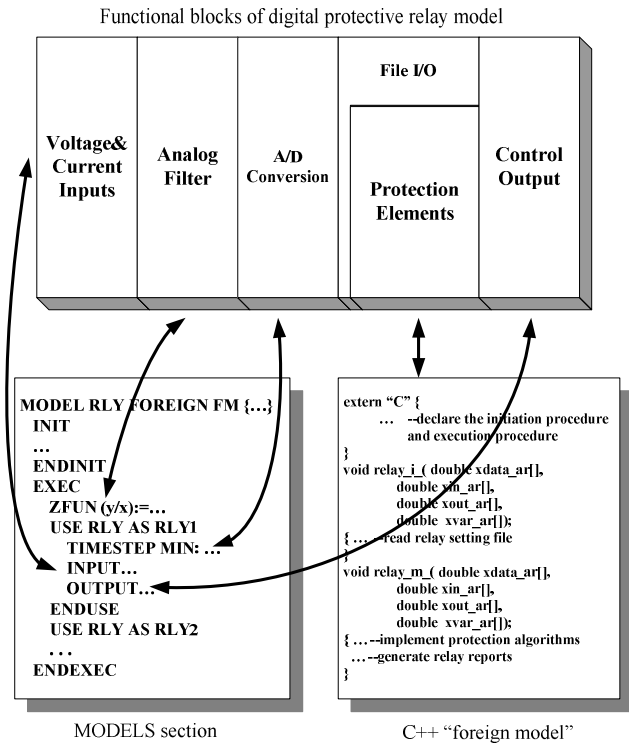


Fig.1 Programming structure of compiled relay model

- 1) The ADP file is graphically created in ATPDraw. The user can build a power system model by adding in all kinds of power system components. By using the ATPDraw, the ADP file can be converted to an ATP data case file.
- 2) An MODELS data case is inserted into the ATP data case file to add the interface between the power system and protection system models. The augmented file is called a template ATP data case file.
- 3) A simulation setting program developed in C++ 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 [11]. The program is also utilized to specify relay settings and user-induced errors. As the outcome, a modified ATP data case file, a relay setting file and a relay error code file are created.
- 4) The C++ source code file which models the relay is compiled as an object and linked with the original ATP object and some libraries by ATP/MinGW. As a result, an executable ATP program is created. Such a program takes the modified ATP data case file, the relay setting file and the relay error code file as inputs, performs the simulation, and generates an ATP .pl4 file which stores the power system measurement data and a relay event report that reveal the relay behavior.
- 5) The relay event report can be manually specified as the input for the digital relay data analysis solution. It can also be automatically identified as the input by the file searching function of the solution.

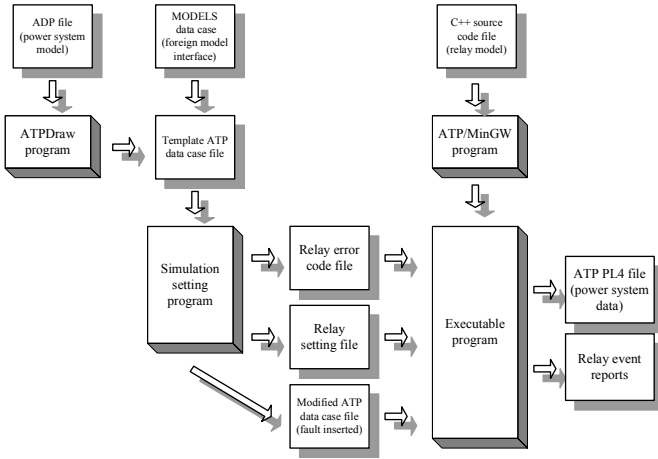


Fig. 2 Simulation procedure for test of digital relay data analysis solution

D. Benefits

The modeling and simulation environment introduced in this section greatly facilitates debugging, testing and evaluating the digital relay data analysis solution. The simulation setting program allows creating various fault conditions on the power system model. The relay model developed in C++ is capable of representing sophisticated protection functions, reading setting files, inserting user-induced errors, and generating event files. The “seamless” power system and protection relay interaction is especially useful for close-loop simulation. These features make the modeling and simulation environment more capable when testing relays with various conditions than existing solutions. Performance evaluation cases for the digital relay data analysis solution and other fault diagnosis applications based on this modeling and simulation environment can be found in [11].

III. TEST AND EVALUATION ENVIRONMENT FOR FAULT DIAGNOSIS ALGORITHMS

Most of the large scale system blackouts have relay misbehaviors involved. This includes relay correct/incorrect operation as well as relay failures and unintended operations. Operation of conventional relays needs to be evaluated under a variety of fault and no-fault scenarios to better understand the causes for misbehavior. The authors recently developed new fault diagnosis algorithms based on neural network, synchronized sampling and wavelets as a reference tool to evaluate operations of conventional relays [2, 3, 12, 13].

To make these new algorithms feasible for online application, a thorough testing based on thousands of different fault and system operating conditions has to be performed and the results need to be compared to those of conventional relays. A handy automated batch simulation environment is necessary for this purpose. This section introduces a batch simulation tool for studying relay and different fault diagnosis approaches based on ATP [7] and MATLAB [9].

A. Framework of the Simulation Tool

The interactive simulation tool takes the advantages from MATLAB with its powerful and easy-to-use programming and modeling feature and from ATP with its accurate and fast

transient calculation engine. The simulation consists of two parts: simulation of power systems and evaluation of fault diagnosis algorithm. The framework of the simulation tool is shown in Fig. 3.

B. Power System Model

The power system is first modeled using the graphic tool of ATP, ATPDraw [15]. The corresponding system parameters are set in MATLAB. The procedure is described as follows:

- 1) User sets the parameters for fault or non-fault scenarios in a MATLAB program. The scenarios can be set in a batch mode using deterministic/random methods. For a deterministic setting, the user defines a combination of fault types, locations, impedances, inception angles, etc. The program will sort out and create a list of scenarios by combining the fault parameters. For a random setting, the user only defines the range of different fault parameters and the program generates the parameters randomly for a scenario.
- 2) Based on the user definition, the MATLAB program sorts out the fault scenarios or non-fault scenarios according to the parameters.
- 3) For all fault scenarios, the program automatically copies the template *.atp* file to a temporary *.atp* file, modifies the parameters in temporary *.atp* file, and executes ATP. In our simulation tool, a universal routine is created for locating every system element’s row position in an *.atp* file. Another set of routines are created for modifying the parameters for each kind of element. By this approach, all system parameters can be automatically modified in MATLAB.
- 4) For each scenario, ATP will generate a *.pl4* file. Using the integrated software *pl42MAT.exe* in ATP, we can transform the *.pl4* file to *.mat* file and obtain related data in MATLAB format. For each scenario, the measured three-phase voltage and current data samples along with

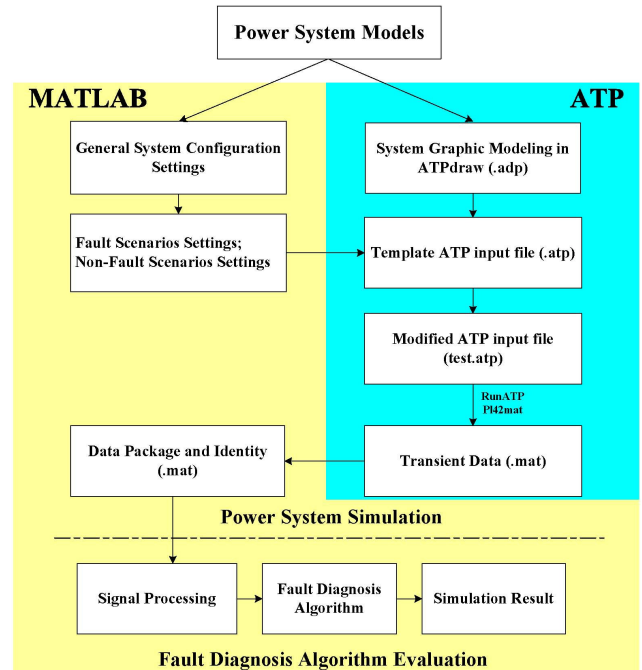


Fig. 3 Framework of the simulation tool for test of fault diagnosis algorithms

the time label are extracted in a *.mat* file, in which there is also an associated identity variable “C” recoding the fault parameters of this scenario. For a batch scenario group, there will be a header file generated, including the configuration and setting information, such as total number of the scenarios, sampling rate, etc.

- 5) Repeat the step 3) and 4) for non-fault scenarios.

C. Evaluation of Fault Analysis Algorithms

The data generated in power system simulations are used for evaluating different fault diagnosis algorithms. Different fault diagnosis algorithms, as well as the traditional relay algorithms can be programmed in MATLAB. The evaluation procedure is shown in Fig. 4 and described as follows:

- 1) General Settings. Preset data re-sampling rate, protection zones, and algorithm parameters.
- 2) Data Input. Load up the source data file, which is generated by simulation of the power systems, into MATLAB program.
- 3) Data Extraction. According to the requirement of the algorithm, extract the useful data from the source data file with re-defined sampling rate.
- 4) Signal Processing. Signal processing is different for different algorithms. For example, the distance relay requires that the waveform be conditioned using low pass filter, and then fundamental frequency phasors of voltages and currents be extracted using full cycle Fourier Transform. For neural network (NN) based approaches, the three-phase voltages and currents are normalized and arranged in a row to form a pattern. For synchronized sampling based approaches, only raw samples in time-domain are required.
- 5) Fault Analysis. The preprocessed data is sent to different algorithms to implement fault detection, classification and location.
- 6) Result Output. The results of fault analysis are compared with the scenarios to measure the analysis errors. The results are saved for further studies.

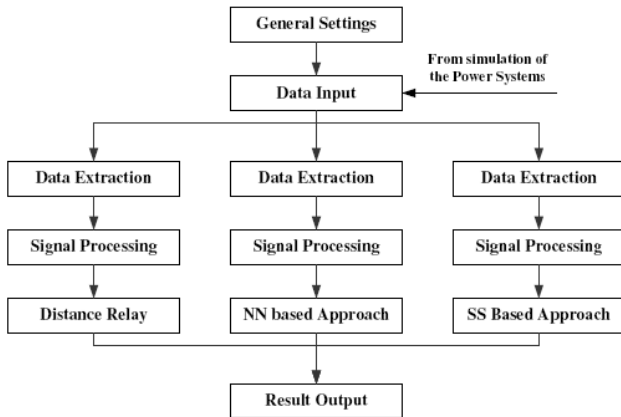


Fig. 4 The procedure for evaluating fault diagnosis algorithms

D. Benefits

The automated batch simulation environment introduced in this section can handle a variety of complex fault analysis tasks under different system conditions. The environment is more efficient than existing solutions because thousands of scenarios can be simulated at one time. The structure of the tool benefits from both the programming flexibility of MATLAB and simulation efficiency of ATP. The software implementation makes it easy to adapt different system models and evaluate different fault diagnosis algorithms. More comprehensive performance studies of the new fault diagnosis algorithms based on this simulation environment can be found in [2, 3, and 12].

IV. TEST AND EVALUATION ENVIRONMENT FOR ALGORITHMS AIMED AT IDENTIFICATION OF VULNERABLE RELAY LOCATION

Power system conditions and protection system behavior have close relations. Studying such interaction is an important subject for enhancement of power system security. The authors have studied approaches for identification of vulnerable relay locations and vulnerable conditions [4, 13]. In our study, dynamic bus voltage phasors are numerically calculated from time-domain transient stability analysis. The phasors are used to calculate the apparent impedance seen by a distance relay and obtain the dynamic impedance trajectory. During the transient stability analysis with critical clearing time (CCT) as the fault clearing time, the dynamic impedance trajectory is monitored. If the apparent impedance falls into the relay protection zone longer than the setting time, the relay is supposed to operate based on its designed characteristic. Thus vulnerable relay locations and vulnerable fault conditions can be identified. The algorithms are implemented in MATLAB.

The results can be tested and verified in a lab test environment. Such an environment requires a power system model, a digital simulator, a physical distance relay and related software programs. To implement a large number of tests in a fast way, the power system model should enable easy creation of various power system conditions. The digital simulator and associated software program should allow batch tests. The settings of distance relay needs to be conveniently configured and modified through a software program and the relay event reports need to be automatically retrieved. This section presents the power system model, setup of the lab test environment and, simulation procedure.

A. Power System Model

A 14-bus system with the same parameters as those used in the numerical calculation is built in ATP, as shown in Fig. 5. The model is composed of power system elements such as generators, transformers, transmission lines, loads, circuit breakers, current transformers and voltage transformers. The simulation setting program described in Section II is used to specify the simulation parameters such as simulation time, fault section, fault type, fault location, fault resistance and fault inception time [11]. Through the simulation, the transient voltage and current data are generated and stored in *.pl4* files.

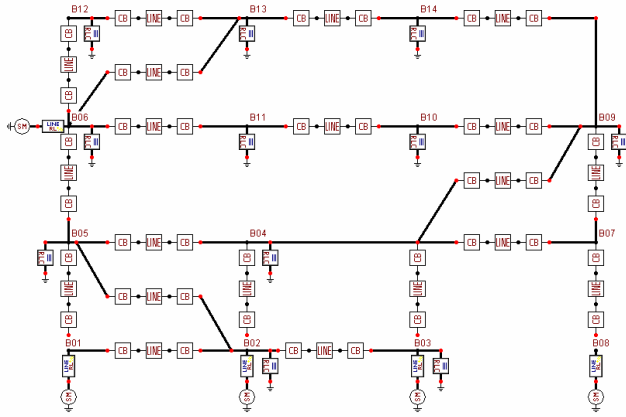


Fig. 5 A 14-bus power system modeled in ATP

B. Lab Setup

The lab setup is shown in Fig. 6. The major components include a PC used to run related software programs, a digital simulator used to generate “real” voltage and current signals and the physical distance relay under test. A commercial software program called Relay Assistant residing on the PC communicates with the digital simulator to send transient voltage and current data and receive contact status data [16]. The digital simulator consists of a digital to analog waveform reconstruction system and power amplifiers. It generates the “real” voltage and current input signals for the relay and records the relay trip contact status. A relay setting software program residing on the PC communicates with the relay to configure relay settings and an automated relay file retrieval software program residing on the PC communicates to the relay to automatically retrieve relay event reports triggered by certain pre-set conditions.

C. Simulation Procedure

The simulation procedure is described as follows:

- 1) The ATP power system model and its simulation setting program are used to generate .pl4 files which contain transient voltage and current data related to various test cases.
- 2) The .pl4 files are imported into the Relay Assistant program. The test cases corresponding to the same relay location are grouped into the same batch test session.
- 3) For a given relay location, the relay settings are configured in a setting group in the relay setting program. Then various groups of settings are uploaded to the relay through the relay setting program.

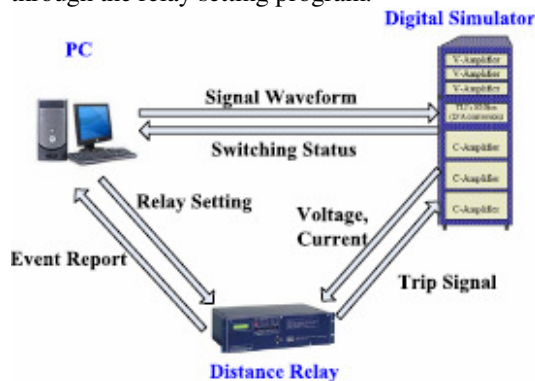


Fig. 6 Lab setup for physical relay test

- 4) The relay setting group corresponding to a given relay location is activated from the relay front-plate buttons or through the relay setting program. Then the batch test session corresponding to the given relay location is executed in the Relay Assistant program and the program drives the digital simulator which performs digital to analog conversion and power amplification to automatically generate transient voltage and current signals case by case. The relay responds to the input signals for each case and generates an event report containing the detailed operation information. The relay file retrieval program automatically detects the event report and downloads it to the PC.
- 5) Repeat step 4) for test cases corresponding to another relay location.

D. Benefits

The lab test environment introduced in this section allows verification of the results for the vulnerable relay location identification algorithms by using physical relays, which is more convincing than tests using relay models. By virtue of the batch test capability of the digital simulator and the automated file download capability of the automated relay file retrieval program, a large number of tests can be automatically implemented in a short period of time, which makes the environment more efficient than existing test solutions using digital simulators.

V. COMPREHENSIVE TEST AND EVALUATION ENVIRONMENT

In previous sections, the test requirements and test environments for three software solutions for automated analysis of protective relay operations have been detailed. Table I summarizes the discussions.

As shown in the table, while the three software solutions have their special test requirements, they have some common requirements as well. Accordingly, their test environments have some common features. In this section, based on the

Table I Summary of Test and Evaluation Environments

Solution	Test and Evaluation Requirements	Test Solution
Digital Relay Data Analysis	<ul style="list-style-type: none"> ■ Various fault scenarios ■ Easy modification of relay setting and user-induced error insertion ■ Relay event file generation 	<ul style="list-style-type: none"> ■ Power system modeling using ATP with simulation setting program in C++ ■ Sophisticated relay modeling using “compiled foreign model” approach
Fault Diagnosis	<ul style="list-style-type: none"> ■ A large number of fault and non-fault scenarios 	<ul style="list-style-type: none"> ■ Power system modeling using ATP with simulation setting program in MATLAB ■ Batch simulation
Vulnerable Relay Location Identification	<ul style="list-style-type: none"> ■ Various power system conditions ■ Various relay locations 	<ul style="list-style-type: none"> ■ Power system modeling using ATP with simulation setting program in C++ ■ Batch tests on physical relays using digital simulator

integration of the strength of each test environment, we propose a comprehensive modeling and simulation environment to serve as the common test and evaluation platform for all the three software solutions. The environment allows tests using relay models as well as physical relays. Fig. 7 illustrates the structure of the proposed environment.

As shown in Fig. 7, the power system is modeled by ATP. The “compiled foreign model” mechanism of ATP is utilized for relay modeling as well as for interfacing between the power system model and relay model. A simulation setting program developed in C++ is used as the user interface to modify the ATP file, relay model setting file and relay model error file to create various test cases. The program allows batch test case generation which can automatically create a large number of test cases based on a pre-set list. The simulation for each case will generate a relay model event report and a .pl4 file. The relay model event report provides detailed records of relay model behavior for all the three software solutions. The .pl4 file containing transient voltage and current data serves two purposes. First, a .mat file will be generated from the .pl4 file by a pl42MAT.exe software program and used as the inputs for the fault diagnosis algorithms. Second, the .pl4 file will be imported into the Relay Assistant software program to drive the digital simulator to trigger the physical relay operation. Through the automated relay file retrieval program, an event report generated by the physical relay is obtained to provide description of detailed physical relay behavior for all the three software solutions.

VI. CONCLUSIONS

This paper presents test and evaluation environments for three new software solutions for automated analysis of protective relay operations developed by the authors. For each environment, the test requirements are presented and test

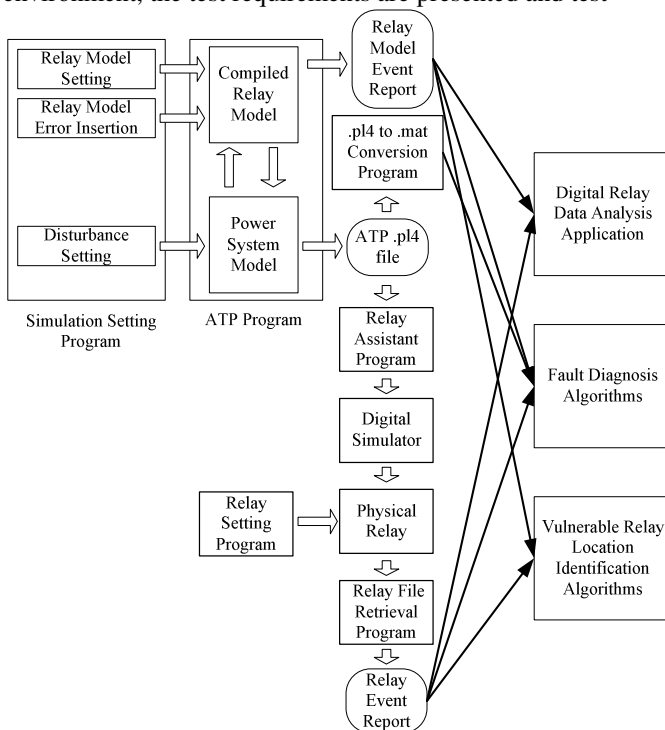


Fig. 7 A comprehensive test and evaluation platform

solutions are detailed. By analyzing the strength of each test solution, a comprehensive modeling and simulation environment is proposed to serve as the common test and evaluation platform for all the three software solutions. Such a platform will allow tests on relay models as well as physical relays. It will provide great capability and flexibility for tests of all kinds of new solutions for power system fault analysis and protection system operation analysis.

VII. REFERENCES

- [1] X. Luo, M. Kezunovic, “Automated analysis of digital relay data based on expert system”, in *Proc. PowerTech Conference*, St. Petersburg, Russia, June 2005.
- [2] N. Zhang and M. Kezunovic, “A real time fault analysis tool for monitoring operation of transmission line protective relay,” *Electric Power Systems Research Journal*, vol. 77, no. 3-4, pp. 361-370, March 2007.
- [3] N. Zhang, M. Kezunovic, “Transmission line boundary protection using wavelet transform and neural network,” *IEEE Trans. Power Delivery*, (Accepted, In Press).
- [4] N. Zhang, H. Song, M. Kezunovic, “New monitoring and control scheme for preventing cascading outage”, in *Proc. North American Power Symposium NAPS 2005*, Ames, Iowa, October 2005.
- [5] T. S. Sidhu, “Relay software models for use with electromagnetic transient analysis programs”, *ELECTRA*, no. 226, pp. 48-54, June, 2006.
- [6] H. W. Dommel, *Electromagnetic Transients Program (EMTP) Theory Book*, Bonneville Power Administration, Portland, OR, 1986 Available: http://www.eeug.org/files/secret/EMTP_TheoryBook.
- [7] CanAm EMTP User Group, *Alternative Transient Program (ATP) Rule Book*, Portland, OR, 2001. Available: http://www.eeug.org/files/secret/ATP_RuleBook.
- [8] L. Dube, *MODELS in ATP, Language Manual*, 1996. Available: <http://www.eeug.org/files/secret/MODELS>.
- [9] The MathWorks, Inc., “Using MATLAB”, Natick, Massachusetts, January, 1999.
- [10] M. Kezunovic, Y. Q. Xia, Y. Guo, C. W. Fromen, D. R. Sevcik, “Distance relay application testing using a digital simulator”, *IEEE Trans. Power Delivery*, vol. 12, no. 1, pp. 72-82, January 1997.
- [11] X. Luo, “Power system fault analysis based on intelligent techniques and IED data”, Ph.D. dissertation, Texas A&M University, College Station, Texas, May 2007.
- [12] N. Zhang, “Advanced fault diagnosis techniques and their role in preventing cascading blackouts”, Ph.D. dissertation, Texas A&M University, College Station, Texas, December 2006.
- [13] H. Song, “The detection, prevention and mitigation of cascading outages in the power system”, Ph.D. dissertation, Texas A&M University, College Station, Texas, December 2006..
- [14] Japanese ATP User Group, *Introduction to ATPDraw+ATP/MinGW Easy Installer*. Available: <http://www.jaug.jp/mingw32/atpeasye.html>.
- [15] L. Prikler and H.K. Høidalen, *ATPDraw Version 3.5 for Windows9X/NT/2000/XP - Users' Manual*, SINTEF Energy Research AS, Norway, TR F5680, ISBN 82-594-2344-8, August 2002. Available: <http://www.eeug.org/files/secret/ATPDraw>.
- [16] Test Laboratories International, Inc. *Relay Assistant -Digital Simulator Software*, 2004. Available: <http://www.tli-inc.com>.

VIII. BIOGRAPHIES

Mladen Kezunovic (S'77, M'80, SM'85, F'99) received his Dipl. Ing. Degree, the M.S. and Ph.D. all in electrical engineering, in 1974, 1977 and 1980, respectively. He has been with Texas A&M University since 1987 where he is the Eugene E. Webb Professor and Site Director of Power Systems Engineering Research Center, an NSF I/UCRC. His main research interests are digital simulators and simulation methods for equipment evaluation and testing as well as application of intelligent methods to control, protection and power quality monitoring. Dr. Kezunovic is a registered professional engineer in Texas, a member of CIGRE and a Fellow of the IEEE.

Xu Luo (S'05) has been with Texas A&M University pursuing his Ph.D. degree in electrical engineering since August 2002. He recently graduated and got employed by AREVA R&D division. His research interests are power system protection, substation automation, and artificial intelligence applications in power system monitoring, control and protection.

Nan Zhang (S'04) has been with Texas A&M University pursuing his Ph.D. degree in electrical engineering since June 2002. He recently graduate and got hired by Global Energy Decisions. His research interests are power system analysis, power system protection, power system stability, system-wide disturbances, as well as signal processing and artificial intelligence applications in power systems.

Hongbiao Song (S'04) has been with Texas A&M University pursuing his Ph.D. degree in electrical engineering since August 2002. He recently graduated and got hired by Bechtel. His research interests are power system analysis, simulation, stability, control, cascading events, protection, substation automation, and intelligent systems.