

Modeling, Developing and Testing Protective Relays using MATLAB, Programmable Relays and Digital Simulators

Mladen Kezunovic

Fellow, IEEE

Bogdan Kasztenny

Senior Member, IEEE

Zijad Galijasevic

Student Member, IEEE

Texas A&M University
Department of Electrical Engineering
College Station, TX 77843-3128

Test Laboratories International, Inc.
1701 Southwest Parkway, Suite 99
College Station, Texas 77840

Abstract — This paper discusses application of MATLAB, programmable relays and digital simulators in modeling, developing and testing of protective relays. First, the issue of modeling of protective relays is addressed. The new MATLAB-based software capable of modeling digital relays by providing detailed models of the common components of contemporary digital relays is presented. Second, application of programmable relays for fast prototyping of certain class of protective relays and for developing real-time hardware models of digital relays is outlined. Third, application of digital simulators for testing physical relays is addressed. The way of using modeling, developing and testing to optimize the design and application is discussed.

Keywords: Modeling and Simulation, Protective Relaying, Programmable Relays, Digital Simulators, MATLAB

I. INTRODUCTION

Contemporary protective relays are becoming more and more sophisticated since the vendors are applying more powerful digital technology, using more involved operating principles, and adding more elaborate communication options.

From the perspective of relay application, all this creates new engineering needs for improvements in understanding of new operating principles, determining proper way of using a given relay, calculating the settings, and finally, performing relay design and application testing.

From the vendor perspective, the new product implementation and evaluation require elaborate tools for modeling, developing and testing as well.

From the educational perspective, both design and application issues are important since students learn best by emulating (modeling), implementing (developing) and simulating (testing) the algorithm of a given relay.

All the aforementioned needs can be met by appropriate modeling, developing and testing tools used together in one programming environment which is the primary interest of this paper.

First, the paper presents new SIMULINK [1] libraries developed for modeling, initial testing and evaluating of protective relays [2,3]. Section II presents the new software in detail. Second, this paper demonstrates an example of the use of programmable relays for developing certain class of relays and building real-time hardware models of digital relays (section III) [4]. Third, the paper shows an example of the use of digital simulation and testing technology for setting calibration, testing and evaluating of physical protective relays (section IV) [5]. Finally, the way of combining, for educational purposes, the modeling (MATLAB), prototyping and testing technology in optimizing the design and application is discussed (section V).

II. MODELING PROTECTIVE RELAYS IN MATLAB

A. MATLAB — the Simulation Environment

MATLAB has been selected as the shell for the new protective relay modeling software [1]. Particular factors that support the selection of MATLAB are:

- (a) MATLAB is a de-facto standard that already dominates the university environment and is more and more recognized in industry including the power engineering field.
- (b) Flexible software structure of MATLAB comprising libraries, models and programs enables one to integrate different model components in one package conveniently.
- (c) MATLAB and its time domain solver, SIMULINK, create a friendly and open system. New models and libraries may be just added to the package without deep knowledge nor modification of the existing parts. This is very useful if the user is involved in continued development of the software.
- (d) Fast-development with MATLAB using powerful calculation and visualization means of the package enables one to expand the software quickly and efficiently without developing any extra programming tools.
- (e) A wide selection of TOOLBOXes - comprehensive collections of pre-defined functions for solving application-specific problems - is already available with MATLAB and is likely to grow even faster in the future.
- (f) Power System Blockset, one of the latest extensions of MATLAB enabling modeling the basic components of power systems. It provides computations similar to EMTP/ATP [6] permitting modeling of both the power

system and its controls in the same environment, and thus, facilitating closed-loop simulation.

B. The Basic Software Concept

The developed software emulates very closely the general philosophy of SIMULINK. The hardware and software elements of digital relays have been reduced to their generic models and implemented as input-output blocks. The review of the most important software development issues follows. More detailed discussion of some of the software implementation and application issues is given in other papers published by the authors recently [2,3].

Partitioning a digital relay for modeling

When partitioning a digital relay (i.e. selecting the blocks to be used in modeling) the objectives were:

- to keep the number of blocks at minimum;
- the blocks should correspond to typical hardware and software modules of actual relays;
- the variations within a given block (digital filter, for example) should be accomplished by setting-up options and input data associated with the block, not by generating new versions of the same block.

As a result, a collection of blocks have been developed that can be combined (with occasional support of the standard SIMULINK libraries) into a model of almost any digital protective relay.

Generic models of relay elements

For each block representing either a part of the relay hardware or software, complete relay or protection system, certain generic model has been assumed. For example, the model of the front-end Data Acquisition Board (DAB) provides signal conditioning, analog filtering and sampling. A variety of options has been offered for all the operations included in the model.

Variety of options and parameters

A variety of options has been assigned to each block in order to model various relay designs. For example, the analog filter of the DAB can be enabled or disabled, its order can be set between 2 and 4, the cut-off frequency is a parameter, the filter's transfer function approximation can be assigned to be of four different types including a "free expression" transfer function. In the same block, the sampling frequency and conditioning gain are parameters. The vertical resolution of the A/D converter can be disabled or enabled. In the latter case, the number of bits and the signal range are the parameters. This shows that each block in order to enable modeling of various digital relays can assume tens of different design characteristics depending on the selected options and data.

Input and output signals

One of the important choices when developing specialized SIMULINK blocks is to assume diverse representation of the input/output signals. The following data structures are used for the signals in the new MATLAB software: real signals,

complex signals and data windows. In order to ensure connectivity of the blocks, the input and output signals of the interconnected blocks must be compatible.

Programming issues

The basic blocks (**Relay Elements, Tools and Input Signals**) have been developed as masked subsystems with s-functions [1] performing the required operations.

The more complex blocks (**Relays and Protection Systems**) have been developed as appropriate connections of the basic blocks.

Some of the software elements (file converters, for example) have been developed as MATLAB programs (m-files).

C. Relay Elements

This library includes models of the basic hardware and software components of a digital relay such as the front-end data acquisition board, digital filter, phasor computation algorithm, comparator, etc.

This SIMULINK library enables designing complete relays by selecting and connecting models of appropriate elements from the library and setting their parameters. Table I provides the basic description of the library.

D. Input signals

This part of the software enables generating the input signals for protective relays. This includes transient modeling of the power system for closed-loop simulation (MATLAB's Power System Blockset) as well as the analytical way of generating the signals. In addition, a file format conversion for data from major transient file formats is also provided (COMTRADE [7] and MATLAB's native format). Table II outlines a brief summary of the library.

E. Tools

This library contains elements that facilitate the analysis and testing procedures. The Phasor Display (PD) is perhaps the best example. This block can be fed by up to six phasors (multiplexed into a single input vector). It displays them in both the graphical and numerical forms. The display can be of the transient type (the plot refreshed every given time interval) or of the steady state type (the values are displayed when the simulation terminates). To use the display one puts its block in the model, connects the input and sets required options. Once the simulation is started and the block is set to active, a new window pops-up and the plot is created.

F. Relays and Protection Systems

These two libraries contain pre-set models of complete digital relays and protection terminals such as overcurrent, impedance and differential relays.

Those models are composed from the blocks of the **Relay Elements** library and the general SIMULINK libraries. Some of the relay models emulate actual relays to the extent possible using the publicly available design details.

Figs.1, 2 and 3 present an example of a three-phase overcurrent relay featuring three-phase instantaneous and inverse-

TABLE I. SUMMARY OF THE RELAY ELEMENTS LIBRARY

Element	Description
Data Acquisition Board (DAB)	An analog input signal is filtered for antialiasing, conditioned, sampled and forwarded as a data window of signal samples. The options include analog filtering and A/D vertical resolution enable/disable switches; type, order and cut-off frequency of the analog filter; range and number of bits of the A/D converter; conditioning gain and the length of the output data window.
Digital Filter (DF)	An input signal is filtered digitally. The pre-defined filters include Infinite Impulse Response (IIR) low-pass, high-pass, band-pass and band-stop filters (individually for each output quantity).
Digital Filter for Transient (DFTR)	Applied using either mean or median filters (individually for each output quantity).
Base Measurement (BM)	power are calculated. The post-filtering may be applied using either mean or median filters (individually for each output quantity).
Differential Equation based Impedance Measurement (DEIM)	The block measures the impedance based on the differential equation approach. The pre-filtering using either Walsh or Fourier filters of selectable window length may be applied. The post-filtering using either mean or median filters may be applied. Either Euler or trapezoidal method of numerical differentiation may be used.
Universal Comparator (UC)	The block is fed by two signals and does a comparison between either the signal and a threshold, the signal and time or the two signals. The direction of comparison may be alternated and the standard time characteristics are included.
Zone Comparator (ZC)	The block is fed by the resistance and reactance values and emulates four forward and one reverse impedance zones. Either the mho or "free-expression" shapes may be set.
TRiggering element (TR)	The block is fed by the data window of a signal and acts as a transient detector. The implemented methods include sample-to-sample, cycle-to-cycle and value-to-threshold checking.
Symmetrical Components (SC)	The block is fed by three phase signals and produces three symmetrical components signals. Either phasors or instantaneous values are utilized.
Vector Group Compensator for two-winding transformers (VG-2)	The block captures instantaneous values of six phase currents of a three-phase two-winding transformer and computes the differential and restraining currents. The parameters include the CT ratios, transformer rated voltages, connection type and vector group.
VG-3	Similar to VG-2, but for three-winding transformers.
Bias Characteristic (BC)	The block is fed by the operating and restraining signal and applies a "free-expression" (point by point defined) bias characteristic.

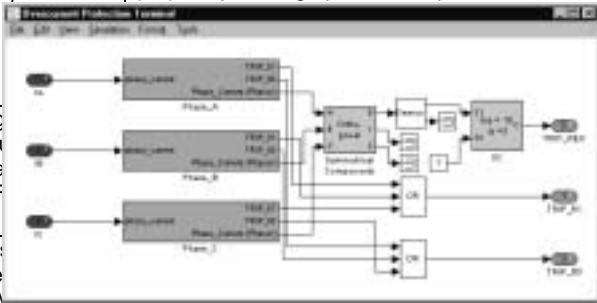


Fig. 1. Applied using either mean or median filters (individually for each output quantity).

time elements as well as instantaneous residual overcurrent element.

TABLE II. SUMMARY OF THE INPUT SIGNALS LIBRARY

Sub-Category	Elements	Structure
Signal Generators	General specification generator, limited frequency spectrum generator, phasor generators, etc.	Library
Data File Converters	ATP to MATLAB, COMTRADE to MATLAB, DFR to MATLAB	Programs
Power System Transient Model	Power System Blockset, Instrument Transformers, internal fault models	Libraries

As seen from the figures and Table I, this model can be assembled from the available libraries in a very convenient and quick way. Modifications of its structure and parameters are very easy. Once developed, the model can be "zoomed-out" to a subsystem [1], get assigned a name and an icon and may be included into a library of **Relays** as a new block (Fig.1).

In the same way the models of larger structures of **Protection Systems** are built out of **Relays** and **Relay Elements**.

III. PROTOTYPING WITH PROGRAMMABLE PROTECTIVE RELAYS

By being able to use models of digital relays one meets the educational need for initial design verification, testing and evaluation. In order to obtain more information on the performance of actual relays one needs to test the physical relay or protection system using realistic input signals and well-established test scenarios.

This engineering activity for evaluating and testing existing relays is straightforward. For educational purposes, when designing new relays, this step requires an extra tool for convenient and fast prototyping. Programmable relays meet this requirement [4].

A. Programmable Relays

The programmable relay used is the protection and control platform with a completely modular hardware design [4]. The hardware can be configured to provide whatever combination of I/O is necessary for certain class of applications, but primarily for protection of distribution systems with relaxed operating time requirements. The twelve front panel meters, eight targets, eight control pushbuttons, and LCD display are fully programmable (Fig.4).

B. User Interface Software

The software provides a common, graphical, interactive interface to the programmable relays [4]. Settings and scheme logic can be quickly modified using an intuitive graphical interface.

C. Available Libraries

Elements

The elements include relays (Table III), timers, oscillography (phasor plot, time plot, event capture, fault locator), symmetrical component calculations, logic (gates, latch, sample and hold), math functions, metering and targets.

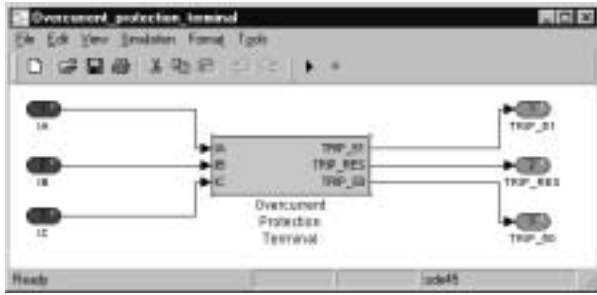


Fig.1. Overcurrent relay model (IA, IB, IC - input currents, TRIP_51, TRIP_50, TRIP_RES - targets).

This part of the programmable relay corresponds to the MATLAB's **Relay Elements, Relays and Tools** libraries and standard SIMULINK libraries.

Pre-programmed protection schemes

A number of pre-programmed schemes are available including feeder protection, line distance protection, transformer protection and capacitor bank protection [4].

This part of the programmable relay corresponds to the MATLAB's **Protection Systems** library.

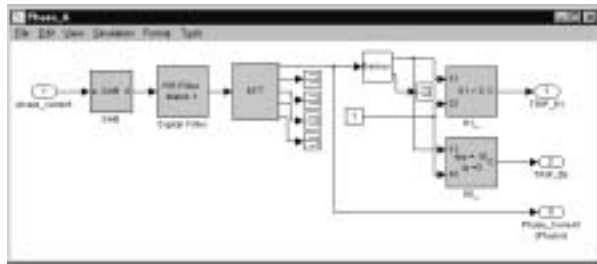


Fig.3. Content of the phase A channel from Fig.2.

D. Virtual Test Set

A Virtual Test Set (VTS) is available for initial testing of the scheme prior to compiling and downloading to the hardware [4]. The VTS is pre-defined for generating a waveform by providing the parameters of the fixed and simple analytical approximation of the signal.

The user interfacing and testing software corresponds to the MATLAB's **Input Signals** library but is by far more specialized.

IV. TRANSIENT TESTING OF PHYSICAL RELAYS

The final stage of the relay design and application testing is to use digital simulators to test physical relays with realistic waveforms obtained from Digital Fault Recorders (DFRs) or



Fig.4. Programmable relay hardware [4].

TABLE III. RELAYS LIBRARY OF THE PROGRAMMABLE RELAY [4].

Relay	Description
21	Distance relay: both phase and ground
25	Synchronism check
27	Undervoltage relay
32	Directional power relay: reactive, real, and total
46	Negative sequence current relay
47	Negative sequence voltage relay
50	Instantaneous overcurrent relay
51	Time overcurrent relay with eight pre-programmed curves and an extra user programmable curve
59	Overvoltage relay
79	Recloser relay: four-shot programmable with sequence coordination
81	Frequency relay
87	Differential relay

by digital simulation.

In the educational activities discussed in this paper, a PC-based open-loop simulator is used [5].

A. Hardware

With reference to Fig.5 the simulator uses a standard PC and offers two basic options for the interfacing hardware: cost-efficient commercial test sets or high-performance commercial amplifiers.

B. Software

The simulator software architecture is shown in Fig.6 [5].

Data generating

The waveform files used for testing usually originate either from DFRs or from transient simulation programs. A great variety of file formats is used. The simulator software includes a file format conversion layer to facilitate the use of the most commonly found file formats (COMTRADE, ATP/EMTP, native DFR formats - Rochester, Hathaway, MATLAB and ASCII).

Waveform processing

The waveform files generated by the transient simulation programs or recorded by DFRs usually require certain processing in order to be actually used for relay testing. The signal editing and processing functions such as cut, paste, insert, resample, rescale, invert, and filter are examples of the functions supported.

User interfacing

The graphical user interface (GUI) is the single most important element of the simulator. The functions for test and waveform handling as well as, signal processing and displaying are improving the productivity of the simulator user tremendously. In addition, GUI provides the required software/hardware transparency.

Waveform replaying

Prepared waveform files need to be played back to the relay under test through a digital to analog conversion system. De-

pending on the selection of the I/O hardware, various implementations of the replaying engines are available.

Processing test results

After replaying the waveform file, the software assists the user in the processing of the relay response. Processing has to extract as much information as possible from the raw relay trip data. To facilitate this, the test results along with the most important test data can be printed or saved in the form of a test report or exported in a form suitable to be loaded into an external database.

V. RELAY MODELING, DEVELOPING AND TESTING

Modeling, developing and testing are the three natural phases of implementing and evaluating a new relay as well as understanding and teaching protective relaying design and application.

As stated in the previous sections, MATLAB is a tool for modeling; programmable relays are the environment for both developing relays of certain class as well as for implementing real-time hardware models of relays (even though the final implementation may be done on different platforms); and finally, digital simulators are the tools for testing physical relays using various scenarios (phasors and transients).

As shown in Fig.7, the interaction between the three activities is bi-directional: certain element is tested and evaluated, while the results of testing generate changes towards improving the relay design. Eventually, the following loop is formed: modeling → testing by simulation → implementing → testing using an actual relay → modeling, etc. The interaction is elaborated below.

A. Modeling vs. Developing

By modeling one can develop the complete software model of a relay and test its performance using signals generated analytically or by digital simulation of the power system. Once the relay model is subjected to the development phase,

certain changes may be requested in order to facilitate the implementation or make the physical relay more robust and reliable. Using programmable relays one can test the algorithm using the VTS.

B. Implementing vs. Testing

A physical relay or the hardware model of a relay are subjected to phasor and transient tests with the use of a digital simulator. Since this is the most realistic test possible, the overall performance of the relay can be evaluated and certain recommendations for relay algorithms, software and hardware may come out at this stage.

C. Modeling vs. Testing

Certain recommendations for the initial design (MATLAB model) may also result from the use of digital simulators for testing a relay model. This interaction is, however, rare because the operating principles and the characteristics of the algorithms may be and should be tested by using MATLAB tools for testing.

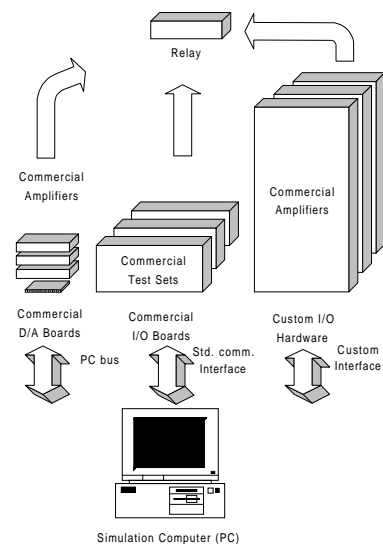


Fig.5. The hardware architecture of the simulator [5].

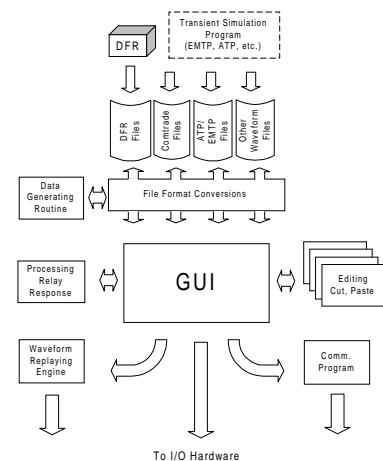


Fig.6. The software architecture of the simulator [5].

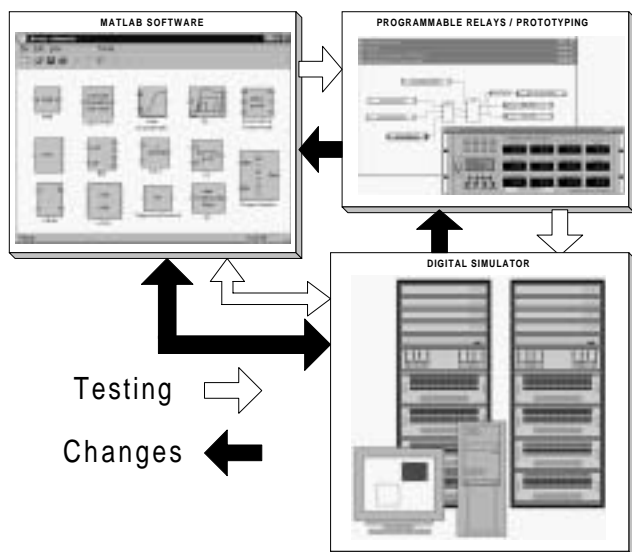


Fig.7. The iterative process of modeling, developing and testing involving MATLAB, programmable relays and digital simulator.

VI. CONCLUSIONS

This paper presents application of MATLAB, programmable relays and digital simulators for modeling, developing and testing protective relays. The new MATLAB software for modeling digital protective relays that facilitates understanding existing products, designing new relays, as well as testing of the relays and relay models using analytical signals and digital simulation is presented. The concept of using programmable relays for developing certain class of relays and/or hardware models of relays is discussed. The use of PC based open-loop digital simulator for the hardware testing and evaluating is outlined. A procedure for modeling, developing and testing that facilitates relay design and educational need in one programming environment is described

VII. REFERENCES

- [1] MATLAB. Reference manual, MathWorks, Inc., June 1997.
- [2] Kezunovic M., Kasztenny B., "New Educational MATLAB software for Teaching Protective Relaying Courses", *Proceedings of the IV Simposio "Ibero-americano Sobre Proteccion de Sistemas Electricos de Potencia"*, Monterey, Mexico, pp.1-6, 1998.
- [3] Kezunovic M., Kasztenny B., Martinez-Lagunes R., Suphasan N., "New Software Tools for Teaching Power System Relaying Utilizing Modeling and Simulation", *Proceedings of the ISCA 14th International Conference On Computers And Their Applications (CATA-99)*, April 7 - 9, 1999, Cancun, Mexico.
- [4] Kojovic L.A., McCall J., Newton T., "Modular Relay Architecture Unifies Protection and Control", *IEEE Computer Applications in Power*, Vol.11, No.3, July 1998, pp.30-35.
- [5] Kezunovic M., Galijasevic Z., "PC Based Dynamic Relay TEST Bench - State of the Art", *Proceedings of the International Conference on Modern Trends in the Protection*

Schemes of Electric Power Apparatus and Systems, October 28-30, 1998, Delhi, India.

- [6] Dommel H., *ElectroMagnetic Transient Program*, BPA, Portland, Oregon, 1986.
- [7] IEEE Standard C 37.111-1991: "Common Format for Transient Data Exchange (COMTRADE) for Power Systems", *Guides and Standards for Protective Relaying Systems*, IEEE 1995.

ACKNOWLEDGMENTS

This project is jointly funded by the National Science Foundation (Grant No. ECS-96-19294) and the Electric Power Research Institute (EPRI). The project is also co-funded by the following industry partners: ABB-TTI, Bailey Network Management, Reliant Energy HL&P, Incremental Systems, Pacific Gas & Electric, Puget Sound Power & Light, Schweitzer Engineering Laboratories, TU Electric. The project activities are a joint effort between Texas A&M University (Professors M.Kezunovic, A.Abur and G.Huang) and Washington State University (Professors A.Bose, K.Tomsovic, M.Venkatasubramanian). The participation of Texas A&M University students R.Martinez-Lagunes and N.Suphasan is also acknowledged.

BIOGRAPHIES

Mladen Kezunovic (S'77, M'80, SM'85, F'99) received his Dipl. Ing. degree from the University of Sarajevo, the M.S. and Ph.D. degrees from the University of Kansas, all in electrical engineering, in 1974, 1977 and 1980, respectively. Dr.Kezunovic's industrial experience is with Westinghouse Electric Corporation in the USA, and the Energoinvest Company in Sarajevo. He also worked at the University of Sarajevo. He was a Visiting Associate Professor at Washington State University in 1986-1987. He has been with Texas A&M University since 1987 where he is a Professor and Director of Electric Power and Power Electronics Institute. His main research interests are digital simulators and simulation methods for relay testing as well as application of intelligent methods to power system monitoring, control and protection. Dr.Kezunovic is a registered professional engineer in Texas.

Bogdan Kasztenny (M'95, SM'98) received his M.Sc. (89) and Ph.D. (92) degrees (both with honors) from Wroclaw University of Technology, Poland, where he is a faculty member of the Department of Electrical Engineering since 1989. In 1994 he was with Southern Illinois University at Carbondale as a Visiting Assistant Professor. During the academic year 1997/98 Dr.Kasztenny was with Texas A&M University as a Senior Fulbright Fellow. His research interests include protective relaying, digital signal processing, real-time computer application in power systems and artificial intelligence methods in power system protection. Currently he is with Texas A&M University as a Visiting Assistant Professor.

Zijad Galijasevic (S'97) has been with TLL, Inc. since 1996. Currently, he holds a position of a senior systems engineer and is responsible for software development. His prior professional experience was with Texas A&M University and Energoinvest Company in software engineering, power system protection, power system modeling, and computer aided testing. Mr. Galijasevic has received his B.Sc. and M.Sc. degrees in electrical engineering from the University of Sarajevo and is pursuing his Ph.D. degree in electrical engineering at Texas A&M University.