A Virtual Environment for Protective Relaying Evaluation and Testing

A. P. Sakis Meliopoulos
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, Georgia 30332

George J. Cokkinides
Department of Electrical and Computer Engineering
University of South Carolina
Columbia, SC 29208

Abstract: This paper presents the application of the virtual test bed for visualization and animation of protective relaying problems. The development of protective relaying animation and visualization objects as well as the interfacing with the virtual test bed is described. Two examples of protective relaying animation are presented: (a) modified mho relays and (b) transformer differential relays. The potential applications and utilization of the approach is discussed in the paper. Since any new relay is a digital relay, the proposed approach is amenable to directly interfacing the manufacturers "relay" with the Virtual Test Bed, thus providing a flexible testing tool for the plethora of relays and relay manufacturers. The resulting tool is extremely valuable for educational purposes.

Introduction

Relaying has always played a very important role in the security and reliability of electric power systems. Many events of outages and blackouts can be attributed to the misoperation of relaying schemes or inappropriate relaying settings. Traditionally, a two-step procedure is applied to minimize the possibility of such events. First, in the design phase, comprehensive analyses are utilized to determine the best relaying schemes and settings. Second, if such an event occurs, an exhaustive post mortem analysis is performed to reveal the root cause of the event and what "was missed" in the design phase. The post mortem analysis of these events is facilitated with disturbance recorders.

In this paper we propose a new approach to this old and perpetual problem. The proposed approach is driven by two factors: (a) recent developments in software engineering and visualization of power system dynamic responses and (b) the new generation of power system digital-object oriented relays. Specifically, it is possible to integrate simulation of the power system, relay testing and animation of relay response. This new approach has been integrated into the Virtual Test Bed, a multitasking simulation program, specialized for electric power systems. Relays are objects that have been interfaced to the virtual test bed.

The paper presents a brief description of the virtual test bed. The application of this system to study relaying performance is described in detail. Specific examples of animation and visualization of differential relays and distance (impedance) relays are presented. It is important to note that the presentation of the paper will be augmented with live demonstration of these examples. One important feature of the system is that the user can apply disturbances to the system while the system operates, i.e. faults, load shedding, motor start-up, etc. The response of the relays is instantaneously observed. The paper describes the mathematical modeling required to achieve this feature in a multitasking environment. The virtual test bed is also a great tool to study adaptive relaying schemes.

Description of the Virtual Test Bed Organization

The virtual test bed utilizes a dynamic system simulator that operates in a multitasking environment. The addition of graphical user interface tools and hardware-accelerated graphics make the final product an indispensable tool to the understanding of the operation of the system. The system has the following features:

1. Continuous simulation of the system under study.
2. Ability to modify the system under study during the simulation, and immediately observe the effects of the changes.
3. Advanced output data visualization options such as animated 2-D or 3-D displays that illustrate the operation of any device in the system under study.

The above properties are fundamental for a virtual environment and for the main subject of this paper, i.e. animation and virtual testing of protective relaying. The first property guarantees the uninterrupted operation of the system under study in the same way as in a physical laboratory: once a system has been assembled, it will continue to operate. The second property guarantees the ability to connect and disconnect devices into the system without interrupting the simulation of the system. This property duplicates the capability of physical laboratories where one can connect a component to the physical system and observe the reaction immediately. For example connecting a motor to the power supply and observing the startup transients and the protective relaying logic during the transients, etc. The third property duplicates the ability to observe the simulated system operation, in a similar way as in a physical laboratory.

The virtual test bed environment supports visualization objects that have access to the instantaneous conditions of the components and the overall system. Thus detailed 3-D "movies" of the instant-by-instant operation of the component or the system can be generated. This paper focuses on using this property for animation and visualization of protective relays. We first present the organization of the virtual test bed and then focus on the animation and visualization of protective relays with specific examples.

The virtual test bed implementation is based on the MS Windows multi-document-view architecture, illustrated in Figure 1.

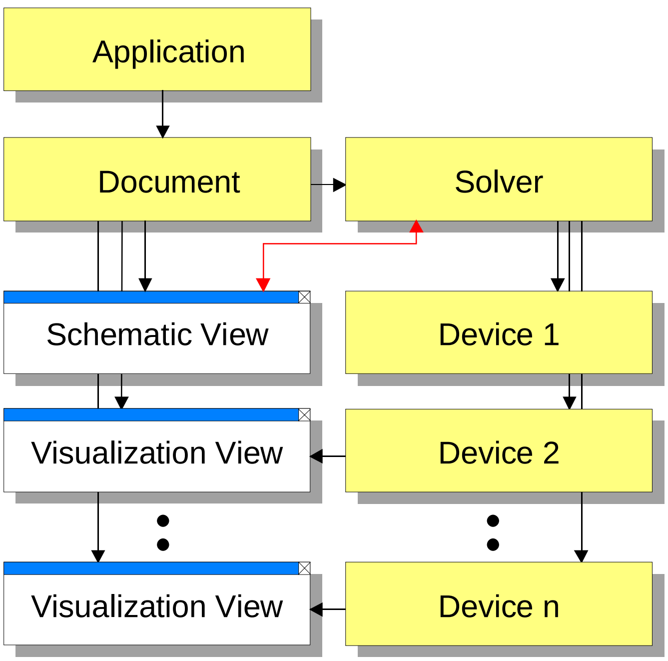

Figure 1. The Virtual Test Bed Architecture

Each document object constructs a single Solver object, which handles the simulation computations. The simulated system is represented by a set of objects – one for each system device (i.e. generator motors, transmission lines etc). The document object can generate any number of view window objects. Two basic view classes are available: (a) schematic views and (b) result visualization views. Schematic view objects allow the user to define the simulated system connectivity graphically, by manipulating a single line diagram using the mouse. Device parameters are also accessible via the schematic editor via pop-up dialog windows. Result visualization views allow the user to observe calculated results in a variety of ways. Several types of result visualization views are provided, including generic oscilloscope style displays, as well as specialized 2-D or 3-D graphics windows controlled by specific device objects.

The solver object is implemented as an independent background computational thread, allowing both schematic editor and visualization views to be active during the simulation run. The solver continuously updates the visualization views so that they always reflect the latest system state. The visualization view update frequency can be user selected, thus allowing speeding-up or slowing-down the simulation speed. Furthermore, the user can modify the system topology or any device parameter during the simulation, and immediately observe the system response in the visualization views.

The network solver is based on the representation of each system device with its *algebraic companion form (ACF)* [1]. The ACF is developed from the integro-differential equations of a component by numerical integration. The ACFs of all components in a system are related via the connectivity constraints. Application of the connectivity constraints yields a quadratic network equation for each time step of the simulation that is solved using Newton's method.

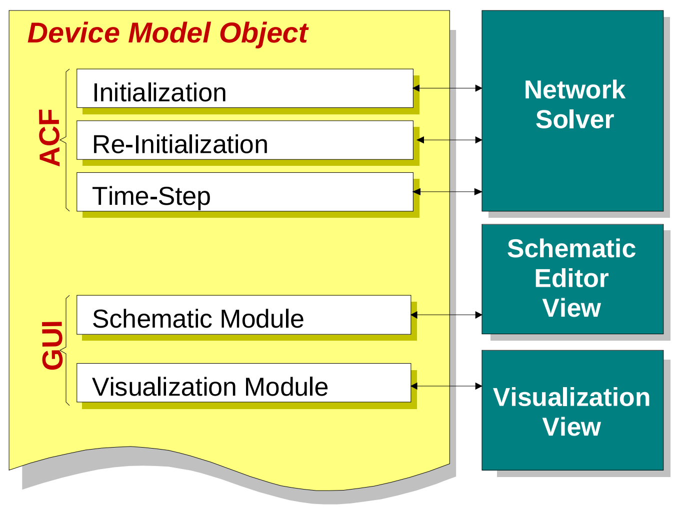

Figure 2. The Device Model Organization

Figure 2 illustrates the organization of device objects and the interaction between the device objects and the rest of the simulation environment. Each model is required to have several standard virtual functions, which facilitate the interface to the network solver, schematic editor and visualization view. The network interface requires at minimum three virtual functions:

1. Initialization. The solver calls this function once before the simulation starts. Device objects can use this function to initialize all dependent parameters needed during the simulation.

2. Re-initialization. The solver calls this function any time the user modifies any device parameter, allowing the device object to update dependent parameters affected by the change.

3. Time step. The solver calls this function at every time step of the simulation. In this function, the device object receives the across variables computed by the solver during the previous iteration and updates its through and internal state variables.

The above three function constitute the minimum requirement for performing time domain simulation. Additional virtual functions are defined for performing small signal stability analysis, and sensitivity analysis concurrently with the time domain simulation [9], [10].

In addition to the above functions, a device object has a set of virtual functions comprising the **schematic module** interface. These functions allow the user to manipulate the device within the schematic editor graphical user interface. Specifically, the device diagram can be moved, resized, and copied using the mouse. Also, a function is included in this set, which implements a device parameter editing dialog window. Furthermore, the **schematic module** interface allows for device icons that reflect the device status. For example a breaker schematic icon can be implemented to indicate the breaker status.

Finally, each device class may optionally include a **visualization module**, consisting of a set of virtual functions that handle the visualization output associated with the device. The visualization module interface allows for both 2-D and 3-D graphics. Presently 2-D output is implemented via the Windows *Graphical Device Interface* (GDI) standard. The 3-D output is implemented using the *Open Graphics Library* (OpenGL). Both 2-D and 3-D outputs generate animated displays, which are dynamically updated by the solver to reflect the latest device state.

There are many potential applications of 3-D animated visualization objects. These objects can generate photo-realistic renderings of electromechanical components that clearly illustrate their operation. The component parts can be viewed from any desired perspective, slowed down or paused for better observation. Furthermore, quantities, that cannot be easily monitored and displayed in a physical laboratory, such as electromagnetic fields and temperature distributions, can be clearly illustrated.

We use this environment to visualize and animate the operation of relays. The system provides the capability to animate not only the inputs to the relay but also the internal workings of the relay by displaying as many as desirable intermediate results of a relay processing of data and logic. These results can be displayed in any desirable form: (a) oscilloscopic views, (b) 3-D visualization of relay mechanical parts, (c) visualization of logic flow, etc. The following section presents two specific application examples listed below:

1. Distance and Mho relay operation.
2. Transformer Differential Relay.

We present the basic animation and visualization modules for these relays and a number of possible applications.

Protective Relaying Example 1: Distance and Mho Relay Operation

Figure 3 illustrates the basics of an example application of the virtual test bed for visualization of protective relay operation. The example system consists of a generator, a transmission line, a step-down transformer, a passive electric load (constant impedance load), an induction motor and a mechanical load of the motor (fan). A modified distance relay (mho relay) monitors the transmission line. This example illustrates the visualization of the operation of the modified impedance relay. The operation of this relay is based on the apparent impedance that the relay 'sees' and the trajectory of this impedance.

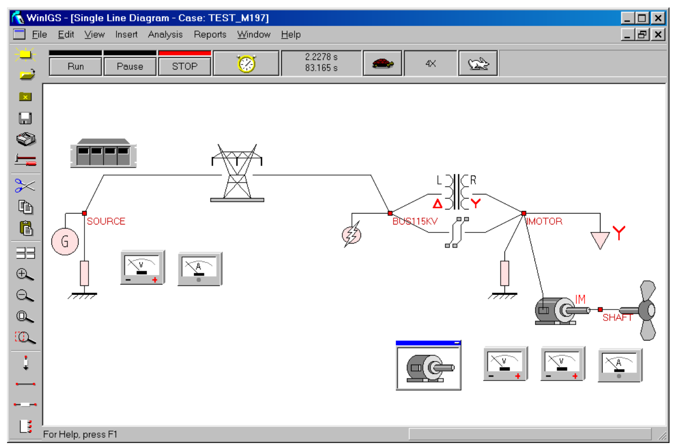

Figure 3. Example Test System For Mho Relay Animation

The visualization object of this relay displays what the relay 'sees' during a disturbance in the system and superimposes this information on the relay settings. Typical examples are illustrated in Figures 4 and 5. The relay monitors the three phase voltages and currents at the point of its application. The animation model retrieves the information that the relay monitors from the simulator at each time step. Subsequently, it computes the phasors of the voltages and currents as well as the sequence components of these voltages and currents. Part of the visualization displays these phasors (see left side of Figure 4). It is important to note that the user may select what phasors to display, i.e. phase voltages or currents or any of the sequence components of the voltages or currents. From this information, the positive sequence voltage and current are constructed and displayed. Figure 4 displays the positive sequence of the voltage and current.

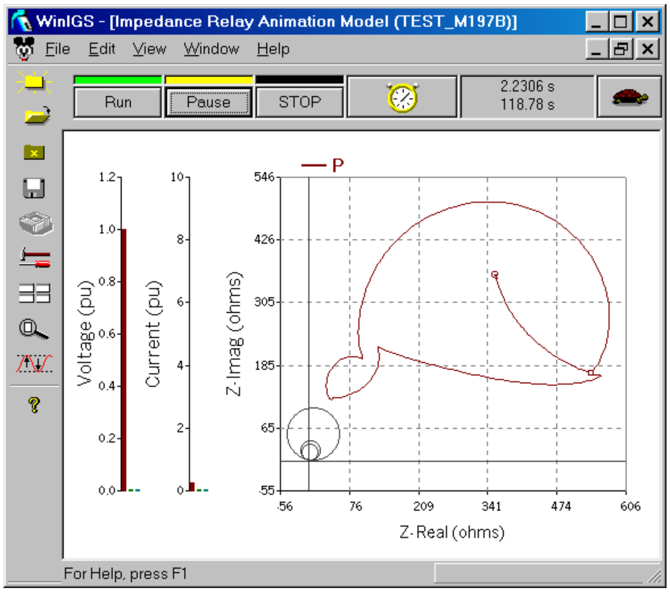

Figure 4. Animation of a Modified Impedance Relay for a Single Line to Ground Fault on the 115 kV Bus

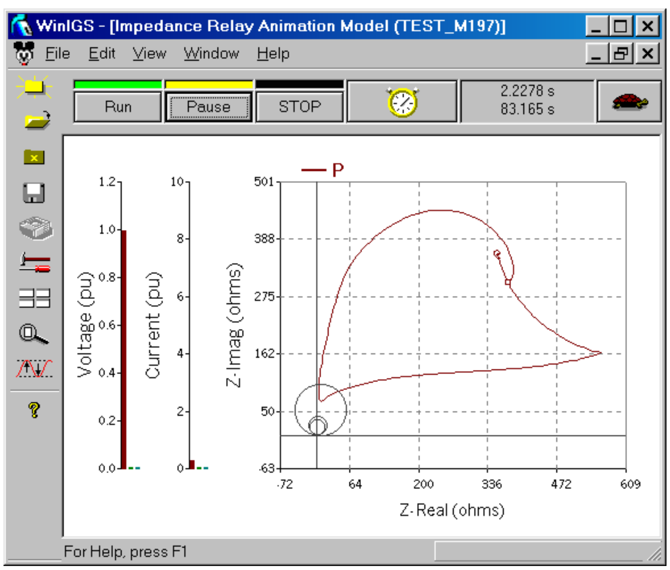

Figure 5. Animation of a Mho Relay for a Three Phase Fault on the 13.8 kV Bus

The apparent impedance is also computed and displayed. Again, the displayed impedance can be the phase impedances or the sequence impedances. As the system state evolves, the visualization displays are dynamically updated, providing a pictorial view of the operation of this relay. In this particular case, the system animates the positive sequence phasors of the voltage and current as well as the positive sequence of the apparent impedance. For example, Figure 4 provides the recorded positive sequence impedance trajectory for the combined event of an induction motor start-up followed by a single-phase to ground fault near the 115 kV bus of the transformer. The impedance trajectory is superimposed on the trip characteristics of this relay. In this case the impedance trajectory does not visit the trip "region" of the relay. Figure 5 provides the recorded positive sequence impedance trajectory for the combined event of an induction motor start-up followed by a three-phase fault near the low voltage bus of the transformer. The impedance trajectory is superimposed on the trip characteristics of this relay. In this case the impedance trajectory does visit the trip "region" of the relay.

Protective Relaying Example 2: Transformer Differential Relay

Another important protective relaying example is the differential relay. In this example we present the animated operation of a differential relay scheme for a delta-wye connected transformer with tap changing under load. The system may operate under steady state or under transient conditions. The effects of tap changing on the operation of the relay are demonstrated. The example system is shown in Figure 6. It consists of an equivalent source, a transmission line, a 30 MVA delta-wye connected transformer, a distribution line and an electric load. A transformer differential relay is protecting the transformer. The differential relay has as inputs the transformer terminal currents. The animation module of the relay, provides a pictorial view of the input currents and the internal currents in the differential relay. The animation uses the electromechanical equivalent differential relay for display purposes. For example, Figure 7 illustrates the "operating" coils and "restraining" coils and the currents that flow in these coils at any instant of time. Both instantaneous values as well as rms values of these currents are displayed. Figure 7 illustrates one snapshot of the system. In reality, as the system operation progresses, this figure is continuously updated providing an animation effect.

The importance of this animation module is that one can study the effects of various parameters and phenomena on the operation of the relay. Examples are: (a) effects of tap setting. The differential relay settings are typically selected for the nomionaltap setting. As the tap setting changes under load, the current in the operating coil changes and may be nonzero even under normal operating conditions. It is very esy to change the tap setting and observe the operation of the relay in an animated fashion. It is also easy to observe the operation of the relay during a through fault for different values of tap settings. Thus this tool is very useful in determining the optimal level of percent restraint for the relay. (b) effects of inrush currents. One can perform energization simulations of the transformer by various types of breaker closing schemes. Since the transformer model includes the nonlinear magnetization model of the transformer core, the magnetization inrush currents will appear in the terminals of the transformer and therefore in the differential relay. The display of Figure 7, provides a full picture of the evolution of the electric currents. One the study the effects by bypassing the even harmonic filters as well as by implementing a number of harmonic filters and observing the effectiveness of the filters.

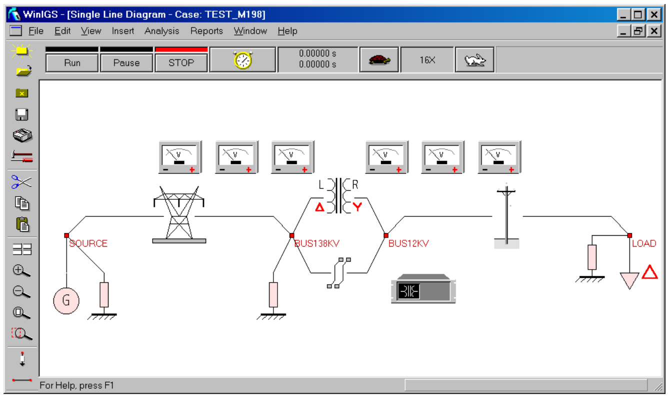

Figure 6. Example Test System For Transformer Differential Relay Animation

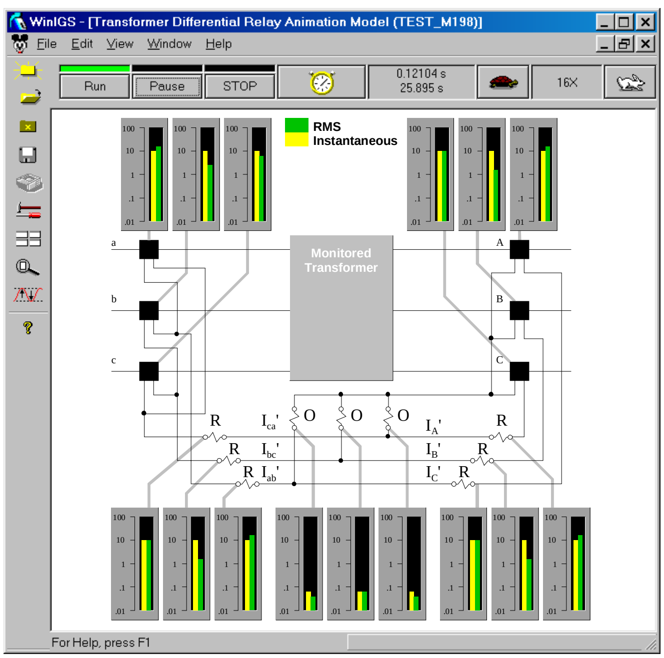

Figure 7. Animation of a Transformer Differential Relay for a Single Phase to Ground Fault on the 115 kV Bus

Interfacing Digital Relays

The virtual test bed uses object oriented programming. As such it is an open architecture and can accept Dynamic Link Libraries of third parties. We propose that the natural extension of the work reported in this paper is to use this feature to interface with commercially available digital "relays". The word "relay" is in quotation marks to indicate that the relay is simply a digital program that takes inputs of voltages and currents, performs an analysis of this data, applies logic and issues a decision. This program is an object and can be converted into a Dynamic Link Library. If this DLL is "linked" with the virtual test bed, in the sense that the inputs come from the virtual test bed, then the specific relay can be evaluated within the virtual test bed environment. The paper proposes an interfacing procedure. This procedure consists of modeling in the virtual test bed the instrumentation channel (i.e. instrument transformers, control cable, attenuators, etc.) and the output of the instrumentation channel is input to the relay object (or DLL). Note that the relay object is the manufacturers software and therefore the response of this relay will be identical with the actual relay in the field. The paper proposes a specific standard for this interfacing. Assuming acceptance of this standard by the relay manufacturers, the virtual test bed could be used to evaluate any commercially available relay that meets this standard. This tool will be invaluable in two respects: (a) to test commercial relays within the virtual laboratory, an inexpensive testing procedure, and (b) to train students and young engineers in the art of protective relaying.

Conclusions

This paper has discussed and presented the virtual test bed and its application for visualization and animation of protective relaying. We have discussed our recent work towards the development of animation and visualization objects of various protective relays. Two examples of protective relay visualization objects have been presented: (a) a distance relay and (b) a transformer differential relay. From these examples, it is clear that virtual laboratories can be quite beneficial from the educational point of view as they can provide insight of the system under study that are impossible in a physical laboratory. In addition, the virtual test bed is valuable for testing commercially available digital relays assuming that they can be converted into a DLL. The presentation of the paper includes a live demonstration of these examples. It is important to note that much more work remains to develop a comprehensive library of relay visualization objects for the plethora of existing power system relaying devices.

Acknowledgments

The work reported in this paper has been partially supported by the ONR Grant No. N00014-96-1-0926. This support is gratefully acknowledged.

References

1. A. P. Sakis Meliopoulos and G. J. Cokkinides, ''A Time Domain Model for Flicker Analysis', *Proceedings of the IPST '97*, pp. 365-368, Seattle, WA, June 1997.

2. Eugene V. Solodovnik, George J. Cokkinides and A. P. Sakis Meliopoulos, “Comparison of Implicit and Explicit Integration Techniques on the Non-Ideal Transformer Example”, *Proceedings of the Thirtieth Southeastern Symposium on System Theory*, pp. 32-37, West Virginia, March 1998.
3. Eugene V. Solodovnik, George J. Cokkinides and A. P. Sakis Meliopoulos, “On Stability of Implicit Numerical Methods in Nonlinear Dynamical Systems Simulation”, *Proceedings of the Thirtieth Southeastern Symposium on System Theory*, pp. 27-31, West Virginia, March 1998.
4. A. P. Sakis Meliopoulos, *Power System Grounding and Transients*, Marcel Dekker, Inc., 1988.
5. A. P. Sakis Meliopoulos, G. J. Cokkinides and A. G. Bakirtzis, “Load-Frequency Control Service in a Deregulated Environment”, *Decision Support Systems*, Vol. 24, No. 3-4, pp. 243-250, January 1999.
6. A. P. Sakis Meliopoulos, Murad Asad and George J. Cokkinides, ‘Issues of Reactive Power and Voltage Control Pricing in a Deregulated Environment’, *Proceedings of the 32st Annual Hawaii International Conference on System Sciences*, p. 113 (pp. 1-7), Wailea, Maui, Hawaii, January 5-8, 1999.
7. Ben Beker, George J. Cokkinides, Roger Dougal and A. P. Sakis Meliopoulos, ‘The Virtual Test Bed for PEBB Based Systems’, *Proceedings of the 3rd International Conference on Digital Power System Simulators*, Vasteras, Sweden, May 25-28, 1999.
8. A. P. Sakis Meliopoulos, David Taylor, George J. Cokkinides and Ben Beker ‘Small Signal Stability Analysis in PEBB Based Systems’, *Proceedings of the 3rd International Conference on Digital Power System Simulators*, Vasteras, Sweden, May 25-28, 1999.
9. A. P. Meliopoulos and George J. Cokkinides ‘Small Signal Stability Analysis in PEBB Driven Motion Systems’, *Proceedings of the ELECTROMOTION ’99 Symposium*, pp. 273-278, Patras, Greece, July 8-9, 1999.

Biographies

A. P. Sakis Meliopoulos (M '76, SM '83, F '93) was born in Katerini, Greece, in 1949. He received the M.E. and E.E. diploma from the National Technical University of Athens, Greece, in 1972; the M.S.E.E. and Ph.D. degrees from the Georgia Institute of Technology in 1974 and 1976, respectively. In 1971, he worked for Western Electric in Atlanta, Georgia. In 1976, he joined the Faculty of Electrical Engineering, Georgia Institute of Technology, where he is presently a professor. He is active in teaching and research in the general areas of modeling, analysis, and control of power systems. He has made significant contributions to power system grounding, harmonics, and reliability assessment of power systems. He is the author of the books, *Power Systems Grounding and Transients*, Marcel Dekker, June 1988, *Ligthning and Overvoltage Protection*, Section 27, Standard Handbook for Electrical Engineers, McGraw Hill, 1993, and the monograph, *Numerical Solution Methods of Algebraic Equations*, EPRI monograph series. Dr. Meliopoulos is a member of the Hellenic Society of Professional Engineering and the Sigma Xi.

George Cokkinides (M '85) was born in Athens, Greece, in 1955. He obtained the B.S., M.S., and Ph.D. degrees at the Georgia Institute of Technology in 1978, 1980, and 1985, respectively. From 1983 to 1985, he was a research engineer at the Georgia Tech Research Institute. Since 1985, he has been with the University of South Carolina where he is presently an Associate Professor of Electrical Engineering. His research interests include power system modeling and simulation, power electronics applications, power system harmonics, and measurement instrumentation. Dr. Cokkinides is a member of the IEEE/PES.