Session: 2677

Evaluation of a Flexible Simulator Structure for Nuclear Engineering Education

Patrick A. Tebbe, Ph.D.
Department of Engineering
The College of Engineering
Ewing, NJ 08628

Abstract

This paper will explore the design and structure of a distributed, multi-code, simulation program designed specifically for educational purposes. A brief review of current nuclear plant simulators will be covered. This will be followed by an examination of research aimed at interfacing a full scope simulator with new desktop interfaces. The results and recent technology improvements that support the concept of a distributed educational simulator will then be covered with examples of similar concepts from other fields given for comparison. Lastly the simulator concept (programming and classroom use) will be more fully described and progress to-date will be presented.

Introduction

Since the incident at the Three Mile Island nuclear power plant the use of simulation has been a major component of nuclear power plant operator training. Today all plants maintain a full scope simulator, which reproduces the thermal hydraulic, reactor physics, and control functions of the plant. The interface for these simulators is a duplication of the reactor control room, down to the last meter and switch. In the past these simulators were powered by large mainframe or minicomputers, while today they can be supplied with data from workstations and high-end desktop machines.

These advances in computer technology now allow the same fidelity of the full-scope simulation to be brought into the classroom. In nuclear plants operators must have a knowledge and understanding of the fundamental processes, as well as the procedures required for plant operation. Other personnel in the plant also benefit from a greater understanding of how the plant operates. The same is true of academic education in nuclear engineering, where the emphasis is on the understanding of theory. However, this type of education differs from the traditional use of the full scope simulator. The full scope simulator emphasizes the control room environment and its representation of plant operation (skill-based behavior). This representation is often not the best suited for building a person's understanding (knowledge-based behavior).

Research indicates that more abstract and hierarchical interfaces, which rely less on real world fidelity and more on psychological styles, supports knowledge based behavior and learning [1]. At the same time a high level of computational fidelity is required to obtain the data powering the interface. From a programming standpoint the best programming structure then exhibits a combination of computational abilities and interfacing options. A blending of programs, languages, and software packages, each tailored to a specific purpose (computation, interfacing, data storage, etc.) then becomes an attractive alternative to stand-alone, or all-in-one, simulation programs.

Simulation Background

Simulation has advanced considerably during the last several decades, aided by the increase in computer speed and the decrease in cost. The first nuclear plant simulator was built by General Electric for the Dresden 2 plant in 1968. At this time the simulator had limited capabilities and there were no requirements for modeling fidelity. Following the incident at Three Mile Island (TMI) in 1979 there has been a concerted effort at improving human performance. In 1981 all nuclear plants were required to operate plant specific simulators. At the time of TMI only 12 control room simulators were in use throughout the country [2]. Today all nuclear plants are required to maintain a full scope control room simulator pursuant to ANSI/ANS guidelines, either the 1985, 1993, or 1998 specifications. This ANS/ANSI-3.5 standard specifies that each simulator shall be capable of reproducing and predicting in great detail the operations and variable relationships within the plant [3].

Powering the simulations are a variety of computer programs. Some of the most important are those which model the thermal hydraulic and reactor physics present in the reactor core. There are several different codes in use today [4,5]. The National Energy Software Center maintains numerous programs covering reactor physics, engineering, and design [6]. In addition, individual plants often have simulation code created specifically for that plant.

When the full scope simulator is being used several different programs and workstations interact with the same database. This allows different tasks to be divided among different machines. For instance, a workstation can be established for the instructor. This provides monitoring of the simulation and allows different scenarios to be created by the instructor. Another workstation will exchange data between the database and the control boards and panels. Since the different workstations and programs (i.e. thermal hydraulics, core physics, control modeling) can interact with a common database through a communications protocol such as a structured query language they can operate efficiently together as one simulator.

In recent years computational power has increased greatly. Hardware performance (in terms of CPU and memory) has shown a doubling time of approximately 2 years. Along with this there has been a shift from mainframe and minicomputers to workstations and high performance desktop computers. Many utilities are now "porting" their simulation codes to Unix and Windows NT computers [7,8]. The new desktop capabilities are also fueling the emergence and use of different types of simulators (part task simulators, basic principle simulators, concept simulators, and special purpose simulators) [9]. Workstations are proving to be more flexible than the full scope simulators and interfacing is shifting from text to graphics with these new capabilities. In support of this the database management systems are also rapidly evolving. In some cases the simulators can be "enhanced by multimedia applications" to provide a better link to the underlying phenomena [10]. The use of graphical or soft panel controls has existed in fossil fuel power plants and other industries for many years now. However, due to licensing restrictions they have not been applied much in the nuclear power industry. This is now changing as more graphical interfaces are used in both control rooms and training. The Sta. Maria de Garona NPP (Burgos, Spain) is developing a prototype control room using touch screens and graphical interfaces. The control room elements are represented by ActiveX components developed in Visual C++ and Visual Basic. A DDE server manages communication between the interface objects and the OpenSim environment [11].

Initial Research

Academia is also making good use of computer simulation in the education of future nuclear engineers. Simulations can give the students a better understanding of the highly coupled variable relationships in a nuclear reaction [12]. In some cases old control room simulators from industry have found their way into academic settings [13]. However, many nuclear programs and individual courses still make use of older, more simplified, simulation codes. The origin of this work began as a project to examine how these older codes could be replaced or updated, mainly in order to improve the interfacing options. It was quickly realized that there was a parallel effort in the local nuclear industry to create new interfacing options for their programs as well.

In conjunction with the simulation staff of the Callaway Nuclear Power Station initial research was conducted into accessing the full scope simulator data with other computer programs. This was done in support of Callaway's efforts at creating a desktop variant of the full scope simulator and the University of Missouri – Columbia's efforts to improve classroom instruction. Following the concept often used for full scope simulators the interface was designed to interact with a database or other forms of shared memory. In this manner the actual data could be supplied from any computation source which was capable of interfacing with the database. The same interface could then be used with the full scope simulator and with older codes developed at the University.

To construct the interface the LabVIEW software package was chosen. While this software is traditionally used for data acquisition and controls it offers several advantages to this project. User interfaces can be easily produced using a point and click method. A wide range of common controls and indicators are already provided for this purpose. The software also comes equipped with various options for interfacing with data sources and other programs. Most importantly, however, both the Callaway plant and the Nuclear Engineering program were currently examining LabVIEW for other uses and this project fit in nicely with those plans.

Before the interface could be created the form of the data storage needed to be specified. Several interfacing options were explored including TCP/IP, Dynamic Data Exchange (DDE), and Open Database Connectivity (ODBC). It was determined that a common database format that could interact with a structured query language (SQL) would be the best choice. For simplicity a Microsoft Access database was then selected.

In order to test the interface certain functions were selected to be displayed. For output display several core variables were chosen, including temperature and flux. For input control the position of the control rods were simulated (Fig. 1). Options were specified so that the core data could be displayed as a three-dimensional contour or as an intensity plot, for a selected core "slab." The control rod indicators were constructed similarly to the actual indicators in the plant and the input device reproduced the lever used in the control room. Appropriate options were incorporated so that the lever would change appearance when pressed up or down and would operate the same as the real lever.

At the time this program was constructed it was not possible to test it dynamically with the full scope simulator, due to hardware and software licensing limitations. Instead the Access database was populated with "dummy" data taken from actual runs of the full scope simulator. This was deemed sufficient as the simulator was already known to interface properly with the database and what was needed was a verification of the new interface.

All of the evaluation goals of the initial research were met. However, several potential problems with full implementation were discovered. While the SQL commands used were very convenient there was the possibility of a serious time lag due to lengthy SQL searches. This was caused in part by the extreme size of the normal simulator database. By splitting up the key variables (temperature, flux, rod position, etc.) into individual Access tables this problem was largely resolved. A second time problem existed with the lever control. The visual update of the lever had a ½ to 1 second lag at times, more than long enough for the eye to catch. It was

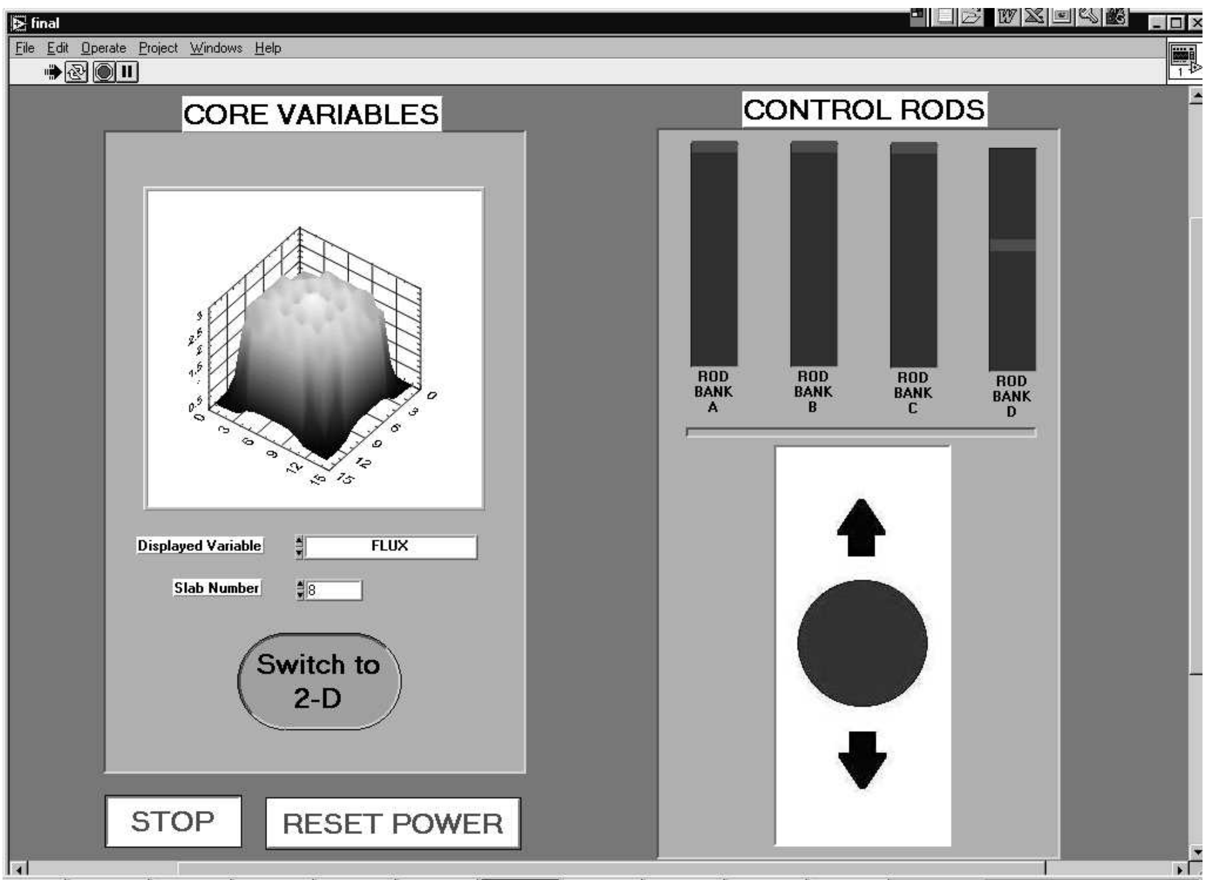

Figure 1: User interface developed to work with the full scope simulator data.

determined that this was largely due to inefficient LabVIEW programming and could be fixed with coding improvements.

Future Development Concept

Following the initial research the goals were reevaluated and the possibilities for the future were examined. It was realized that interfaces of this sort offered possibilities beyond the computational options available from older, existing programs. However, use of the full scope simulator programs was not possible, mainly due to licensing issues with the software and performance issues with the hardware. The task of creating all new coding which modeled the plant more in-depth appeared to be a sizable and daunting project. A solution was needed which offered comprehensive and realistic data, did not require extensive software or hardware resources, and which was not an all-or-nothing programming project. The solution was found in an extension of the original full scope simulator's distributed architecture.

As already described, many full scope simulators are actually a combination of programs which interact with a shared memory location or a database. The user interface that was developed was based on this format. An extension of this program distribution is to also divide the computations among various pieces of hardware. This approach has already been demonstrated by other researchers who have created a PWR simulator with "high processing performance using low cost hardware [14]." The same concept has been applied to the Modular Modeling System (MMS) which is used by industry to model power plants. By integrating remote access and control methods the MMS models can be integrated with a wide range of interface options [15].

In order to achieve all of our educational objectives this distributed concept was taken one step further. If students could access the simulator from any location remotely it was felt they would be more apt to make use of it. The key limitation would be the manner in which the inter-process communication (IPC) is handled between programs. Fortunately there is an answer to this with the IEEE Distributed Interactive Simulation (DIS) Protocol. The DIS protocol would allow the individual components (and data) of our simulator to be linked through the Java programming language. This approach has already been used in other fields. In particular, various military organizations around the world use this approach to generate equipment trainers and battlefield simulations.

Using this method the simulator could be constructed around, or modified to, use of a standard web browser as the interface mechanism. It also opens up the possibility of collaborative work on the computations side of the simulator as well. Modules for neutronic calculations and for thermal hydraulics could be developed and run on computers that are geographically separated, allowing a pooling of programming and hardware resources. With regard to classroom use, this approach allows different levels of computational complexity to be built and different user interfaces to be developed. Individual instructors would then be able to tailor the program to specific learning objectives.

Conclusions

Simulation has proven to be a valuable training and educational tool. This is perhaps more true for nuclear engineering where the opportunities for hands-on learning are limited and shrinking daily. While the sophistication and capabilities of full scope simulators has increased greatly in the last decade the standard nuclear engineering classroom simulator (for knowledge based learning) has not kept pace. This paper has presented initial research into updating legacy classroom codes and the technological advancements that allow a new breed of educational simulator to be constructed. By making use of a distributed simulation environment students will have easy access to high fidelity simulations. Hardware and software limitations, such as cost, are bypassed. The resulting simulation will be easily customizable and upgradable and new possibilities exist for collaboration between academic institutions. To date this approach has not been explored past the literature research stage. Possible programming collaborations and testing are possible between the University of Missouri and The College of New Jersey, however, additional partners may be needed. Currently simulator requirements are being assessed prior to creating programming action items.

Acknowledgements

The preliminary work presented here was funded through a fellowship from the Institute of Nuclear Power Operations (INPO) and the University of Missouri – Columbia. The assistance of Scott Halverson at the Fulton (Callaway) Nuclear Power Station was also greatly appreciated.

Bibliography

1. Vicente, K., Christoffersen, K., and Pereklita, A., "Supporting Operator Problem Solving Through Ecological Interface Design," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 25, 1995.
2. Smith, R. ed., "Twenty Years Later," *The Nuclear Professional*, 1st quarter issue, 1999.
3. ANSI, "Nuclear Power Plant Simulators for Use in Operator Training and Examination," revised ANSI/ANS-3.5 standard, 1998.
4. Agee, L., "Overview of Electric Power Research Institute Nuclear Safety Analysis Activities," *Nuclear Technology*, Vol. 121, 1998.
5. Shotkin, L., "Development and Assessment of U.S. Nuclear Regulatory Commision Thermal-Hydraulic System Computer Codes," *Nuclear Technology*, Vol. 116, 1996.
6. Glasstone S. and Sesonske, A., Nuclear Reactor Engineering, Chapman and Hall, 4th edition, Vol. 2., 1996.
7. Laughton, T., Belblidia, L., Andersen, P., and O'Farrell, D., "THOR ComEd Advanced Model Simulator Upgrade," *Western Multiconference 2000,* Society for Computer Simulation, 2000.
8. Boire, R., "Fourth Generation Full Scope Simulator on Windows NT," *Western Multiconference 2000*, Society for Computer Simulation, 2000.
9. Baumont, G., and Montes, G., "Current practices in operator training with simulators in OECD countries," *1997 IEEE 6th Conference on Human Factors and Power Plants*, IEEE, 1997.
10. Bartak, J., Chaumes, P., Gissinger, S., Houard J., and Van Houte, U., "Operator Training Tools for the Competitive Market," *Computer Applications in Power*, IEEE, Vol. 13, 2000.
11. Corcuera, P., Garces, M., and Ryan, J., "A Training Simulator with Soft Panels," *Western Multiconference 2000*, Society for Computer Simulation, 2000.
12. Doster, J., "Simulation as a Tool for Teaching Nuclear Reactor Systems," *ANS Transactions*, Vol. 81, 1999.
13. Ma, Y. and Edwards, R., "Undergraduate Simulator Design Course," *ANS Transactions*, Vol. 80,1999.
14. Kitamura, M., Ohi, T., Yamamoto, T., and Akagi, K., "Development of High Precision Plant Simulator for Pressurized Water Reactor Plants using Distributed Architecture," *Journal of Nuclear Science and Technology*, Vol. 3, 1999.
15. Jones, C.A., "Integrating Remote Access & Control Methods into Simulation Models," *Nuclear Plant Journal*, July-August, 1994.

PATRICK A. TEBBE
Patrick Tebbe is an Assistant Professor of Mechanical Engineering at The College of New Jersey. Patrick teaches undergraduate thermodynamics and thermal sciences laboratories. He received his Ph.D. in Mechanical Engineering from the University of Missouri-Columbia in 1997. Before moving to New Jersey he served as an Adjunct Assistant Professor at Missouri and also completed a M.S. in Nuclear Engineering. He is a member of ASME, ASEE, and ANS where he currently serves as the Secretary/Treasurer for the Human Factors Division.