LA-UR-15-27110
Approved for public release; distribution is unlimited.

Title: Simulator for an Accelerator-Driven Subcritical Fissile Solution System

Author(s): Klein, Steven Karl
Day, Christy M.
Determan, John C.

Intended for: Report

Issued: 2015-09-14

Simulator for an Accelerator-Driven Subcritical Fissile Solution System

Background

For the past five years Los Alamos National Laboratory (LANL) has examined the theoretical characteristics of fissile solution systems of uranium. This effort was driven by a desire to utilize these systems to produce the important medical isotope Molybdenum-99 (Mo^{99}). Current techniques for production of Mo^{99} involve use of metal targets of Highly Enriched Uranium (HEU) irradiated in large power or research reactors, which are generally more than five decades old and, therefore, nearing end of operational life. The continued use of HEU targets in the future is not desirable due to non-proliferation concerns. In addition, adverse economic factors of the cost of continuous supply line of uranium targets coupled with the high cost of construction, licensing, and operating large power reactors without government subsidies as is the current practice, suggests alternative means of Mo^{99} production is highly desirable.

Reactors fueled by fissile solutions of uranium operated continuously at LANL from 1944 until 2004. These Aqueous Homogeneous Reactor (AHR) systems were of a variety of designs, utilized fuels of uranium sulfate, nitrate, and fluoride. In the late 1990's Mo99 was successfully extracted from SHEBA, a uranium fluoride fueled AHR. These AHR were generally used to produce large quantities of thermal neutrons for a variety of physics experiments including determination of cross sections of isotopes and for evaluation and calibration of dosimeters and criticality alarms to be used in process facilities. The underlying reactor physics, especially related to long term steady-state operation required for isotope production was not a subject of much study. Since SHEBA, the last operating AHR in the United States, was decommissioned in 2004 only theoretical examination of reactor physics, based on historical experimental data, has since been possible.

Initial effort was directed at modeling SUPO (<u>Su</u>per <u>Po</u>wer), an AHR that operated at LANL from 1951 to 1974, accumulating over 600,000 kWh of operation. This uranyl nitrate (HEU) fueled system was chosen as the benchmark for steady-state operation of an AHR. Similarly, SILENE, an AHR that operated in Valduc, France, was selected as the benchmark for pulsed operations. KEWB systems (<u>K</u>inetics <u>E</u>xperiment <u>W</u>ater <u>B</u>oiler), which represent five separate evolutions of AHR operated by North American Atomics in the 1960's and 1970's, were chosen as confirmatory systems to evaluate the modeling technique. Results of this effort were published by LANL in 2013[1] documenting the validity of the dynamic system simulation (DSS) technique in faithfully reproducing the time dependent behavior of these systems in steady-state, slow kinetics, free evolution, pulse and boiling modes.

During 2012 interest developed in utilizing a DT (deuterium on tritium) accelerator to drive a fissile solution in a subcritical configuration for Mo^{99} production. Using the DSS technique that had been verified on historical AHR, LANL modeled a variety of generic system configurations for these subcritical

---

[1] "Dynamic System Simulation of Fissile Solution Systems"; LA-UR-14-22490; Steven Klein, Robert Kimpland & Marsha Roybal; March 2013

accelerator-driven systems. Initial results were published in late 2012[2] with subsequent operating details provided in separate reports. The first[3] provided specific estimates of performance compared to the AHR record showing that the general operating characteristics such as large negative reactivity feedback mechanisms due to fuel temperature rise and radiolytic gas void were common attributes. The second[4] extended the analysis to include explicit model elements for radiolytic gas handling and a full cooling loop with primary and secondary heat exchangers. Once again, it was shown that the operation of fissile solution systems exhibit great commonality of behavior. The important question of stability of these systems was examined through using the DSS models as a basis for generating transfer functions appropriate for each configuration. These transfer functions could then be evaluated using traditional linear systems analysis techniques. This effort documented that fissile solution systems of uranium were unconditionally stable in the linear approximation due to their large negative reactivity feedback[5].

Designer's Toolbox

The DSS models are useful to examine the underlying physics of systems operations. However, they are not ideally configured to support design of a specific fissile solution system; the primary reason being that intimate familiarity with the DSS scripting language is required. For design team management, direct access to the underlying physics model may also not be desirable. Accordingly, LANL decided to develop a technique to rapidly construct a version of the DSS model for a specific system. The approach chosen was to convert the DSS model in the native scripting language to C++ code that could be embedded into a user's interface implemented in Visual Studio. This allows the designer to manipulate a range of meaningful design characteristics such as physical configuration and operational parameters to assess the impact of proposed design decisions on overall system performance without having to manipulate either computer code or the underlying physics model.

Visual Studio models for both SUPO and a generic accelerator-driven subcritical system have been developed[6]. Validation of the conversion was accomplished through extensive testing over a variety of operating scenarios where the Visual Studio implementation was compared to the DSS version to ensure that each version provides identical results using the same initial conditions.

The Visual Studio implementation provides utilities to the designer that extends the DSS models. For example, the DSS model generally provides a graphical output showing the time history of the system evolution. The Visual Studio version stores a host of operator selected parameters in a database allowing the designer a detailed view of this history. Also, the stability analysis is embedded in the Visual Studio version. This allows a designer to simulate system operation to a point where steady-state is

---

[2] "System Model for an Accelerator-Driven Fissile Solution Assembly"; LA-UR-12-25461; Steven Klein & Robert Kimpland; November , 2012
[3] "Operating Characteristics of Accelerator-Driven Fissile Solution Systems"; LA-UR-14-27027; Steven Klein, Robert Kimpland & Marsha Roybal; September, 2014
[4] "Operating Characteristics of a Subcritical Accelerator-Driven Fissile Solution System"; LA-UR-15-25067; John Determan, Steven Klein, Robert Kimpland, & Marsha Roybal; May, 2015
[5] "Stability of Fissile Solution Systems"; LA-UR-15-23832; Steven Klein & Robert Kimpland; May, 2015
[6] "Executable Models of Fissile Reactor Systems for Hardware Simulation and Design"; LA-UR-15-20648; John Determan, Christy Day, Steven Klein, & Marsha Roybal; May, 2015.

achieved. At this point the designer may choose to utilize the state of the system at that condition to generate a view of the system stability. Traditional Bode Amplitude, Bode Phase, and Nichols graphs are provided. This then provides the designer with a rather complete view of the system performance that may be expected, which in turn may provide a path to system optimization.

System Simulator

Modern computer technology allows the development of a rather complete simulation of a dynamic system. Using this concept, simulators have been developed ranging in complexity from one running on a single personal computer to a full mock-up of a control room. The system simulator developed by LANL for fissile solution systems currently is implemented in a single personal computer format using the National Instruments LabVIEW system design software[7].

The purpose of the LabVIEW implementation is to provide system developers the ability to evaluate human factors characteristics of the design. In addition, operator training related to normal dynamic events such as start-up and shutdown may be conducted. With the addition of an "instructors screen" that interfaces with the basic simulator, off-normal events may be simulated, allowing assessment of the operator controls required to identify and respond to an event.

The simulator is based on a two-tier architecture, as shown in Figure 1. The simulation executes in a C++ module. The operator and instructor screens are two specific LabVIEW user interfaces. An interface layer exists between these two levels providing commands that allow the operator and instructor to affect the course of the simulation. Data returned via this interface is displayed in the operator's control panel and also lets the instructor know the current state of the system.

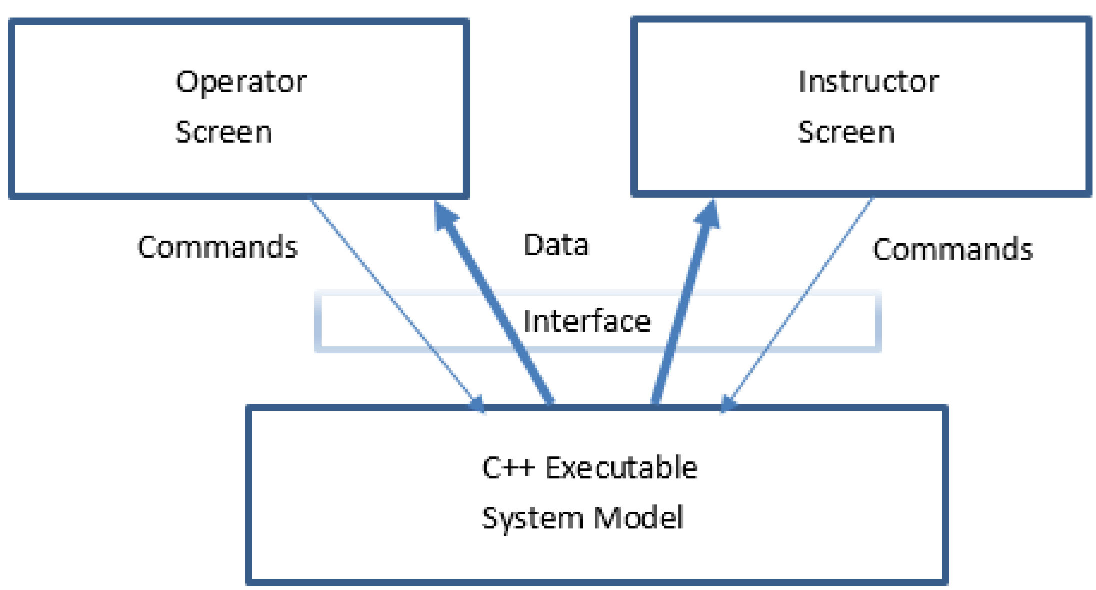

Figure 1: Simulator Two-tier Architecture Diagram.

[7] "Simulator for an Accelerator-Driven Fissile Solution System– Notional Design; LA-UR-14-26996; Christy Day & Steven Klein; September, 2014

Case Example: Accelerator-Driven Subcritical System

A side-by-side comparison of the dynamic system simulation and Visual Studio conversion of start-up of a generic accelerator-driven subcritical system is shown in Figure 2. As can be seen from the comparison, the graphical presentation of the Visual Studio implementation closely matches the behavior of the DSS model.

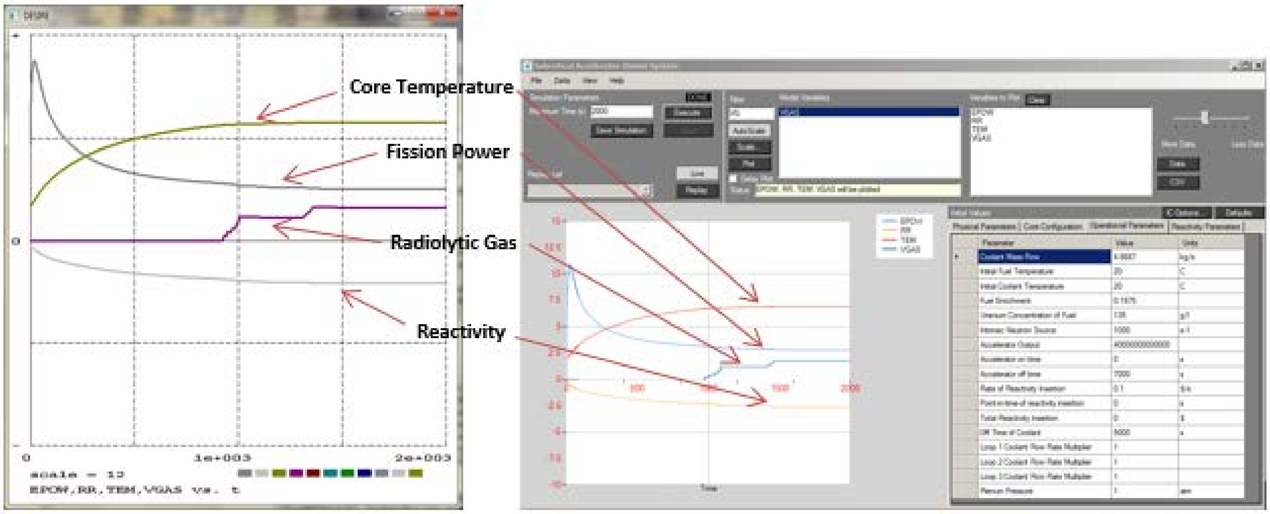

Figure 2: DSS and Visual Studio Implementation Comparison.

The Design Tool is also a two-tier model. The simulation is executed in the same C++ system model used by the simulator. A different programming interface connects the design tool to the C++ system model. The design tool is a user interface developed in C# using Visual Studio, and consists of blocks for simulation control, initial conditions specification, plot and data table parameter specification, plot display, and data table display. Not all of these components are shown in Figure 2 – the table in the lower right quadrant is the initial conditions specification block; this quadrant may optionally display a data table containing user specified data. The data plot shown in the lower left quadrant may also be swapped for the stability plots.

Figure 3 provides a view of the same startup scenario of the accelerator driven system as shown in Figure 2, but this time using the LabVIEW simulator. The graphical display screen is split to provide two scales, which show the absolute values of the various parameters. In this particular case, the time scale has been compressed to match the 2000 seconds of data presented for the DSS and the Visual Studio implementations. In the actual simulator, as will be seen below, the time base is real time so the events are not visible on a single screen. For example radiolytic gas saturation and release only appears after ten to 15 minutes of operation. Note the controls present. An operator is required to follow the proper startup sequence in order to achieve startup. In this implementation SCRAM must be first reset then the fuel vessel may be filled. Subsequent to vessel fill, the accelerator and coolant flow loop controls become available.

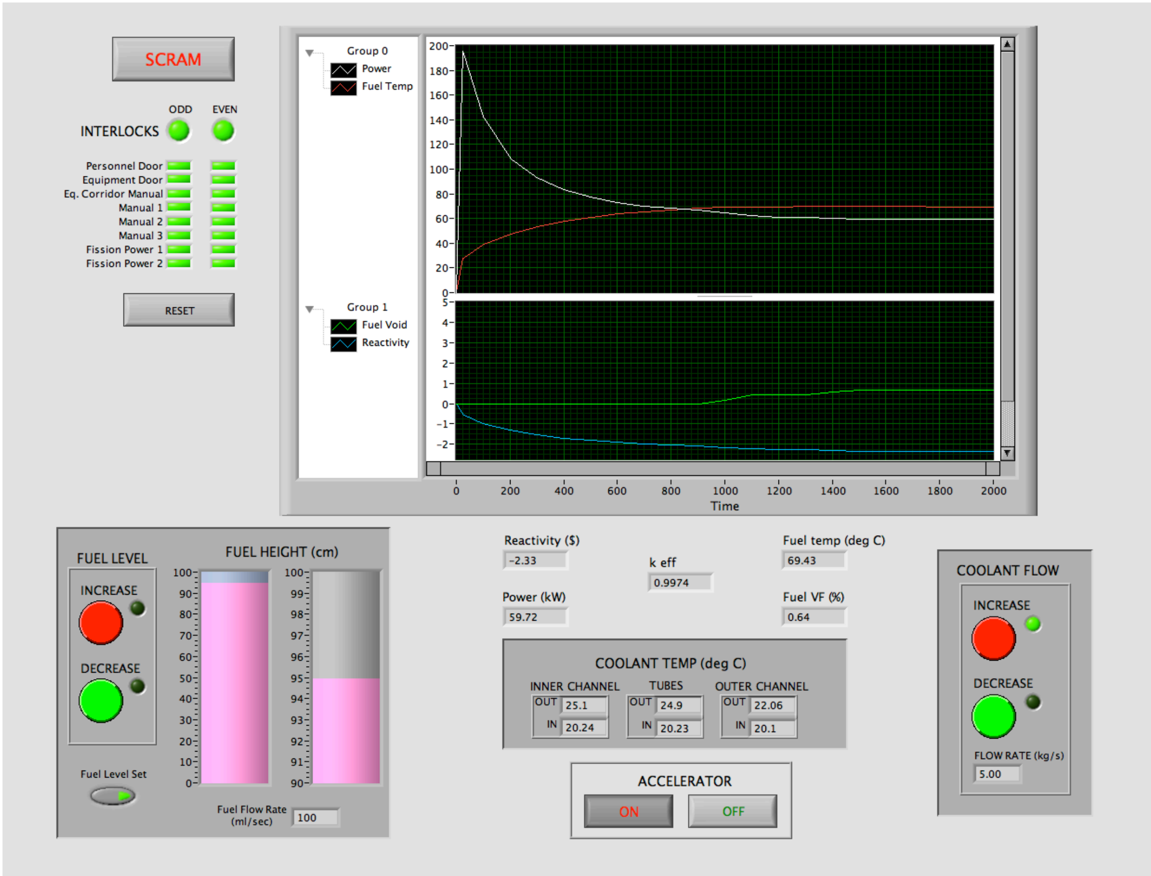

Figure 3: LabVIEW Implementation of Accelerator Driven System Model.

Figure 4 is a view of the Instructors Screen that is the companion to the Operator's screen. This allows an instructor to induce events into a running simulation to evaluate operator's response.

Several control sets are provided in this version, each of fundamental importance to the operation of an accelerator driven subcritical system. The coolant flow set allows variation in the rate of coolant flow, which directly affects the ability of the system to extract fission heat from the system, including hard and soft fail options. Hard fail is an instantaneous loss of coolant while the soft fail mode is a "spin down" of the coolant pump over time. Note that since reactivity feedback due to temperature in a fissile solution system is negative with rising temperature, such failures result in decreased fission power and rising fuel temperature, which if not corrected by the operator will result in boiling and system shutdown.

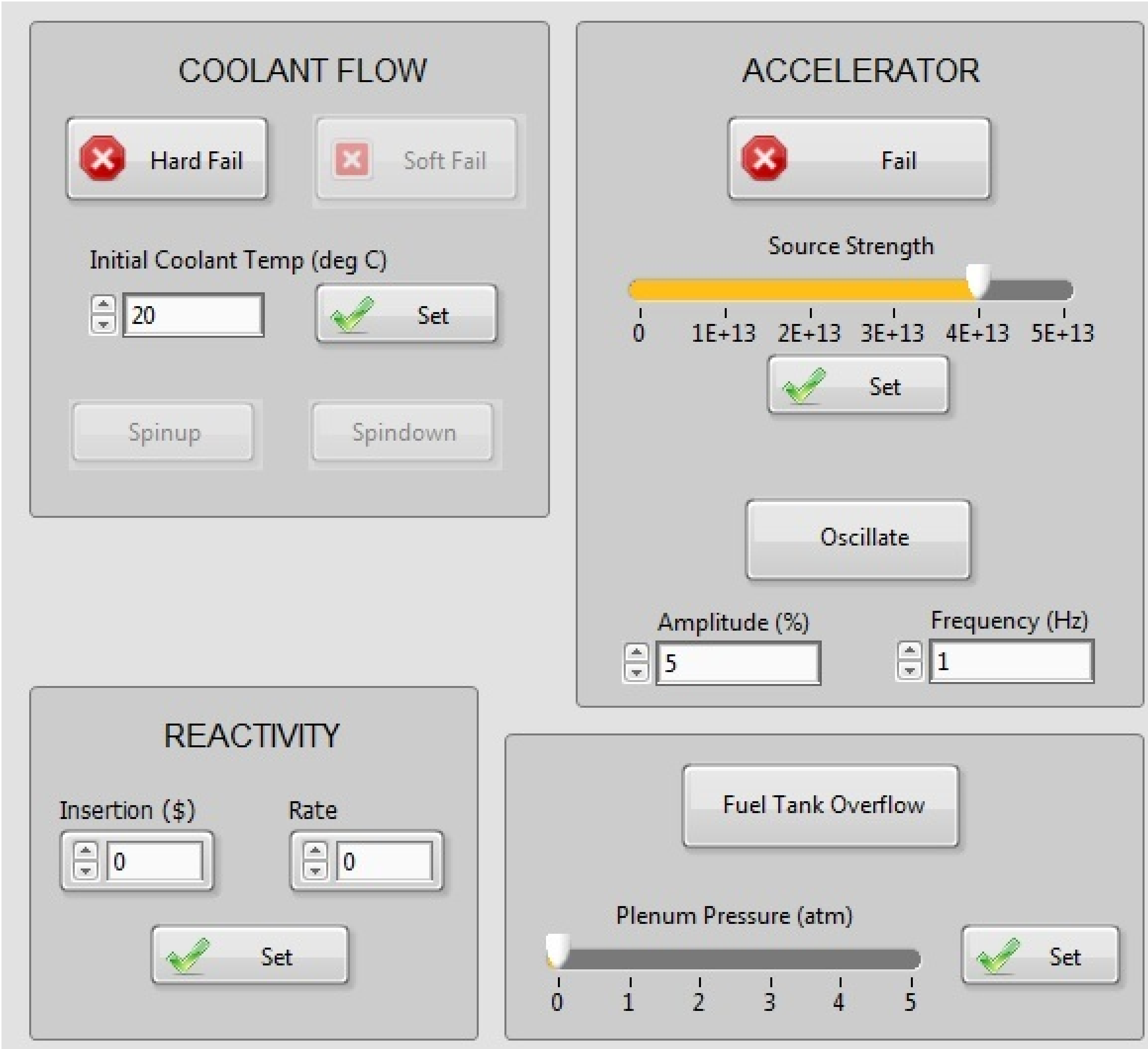

Figure 4: Instructors Screen for Accelerator Driven System.

Figure 5 shows one off-normal event of particular interest to operation of a subcritical system. In this case, the coolant temperature is below the initial starting temperature of the fuel and the accelerator fails to operate on demand. The result is fuel cooling, which results in increase in solution density, hence reactivity. Since these systems start close to critical such cooling will result in a criticality excursion if unchecked by an operator.

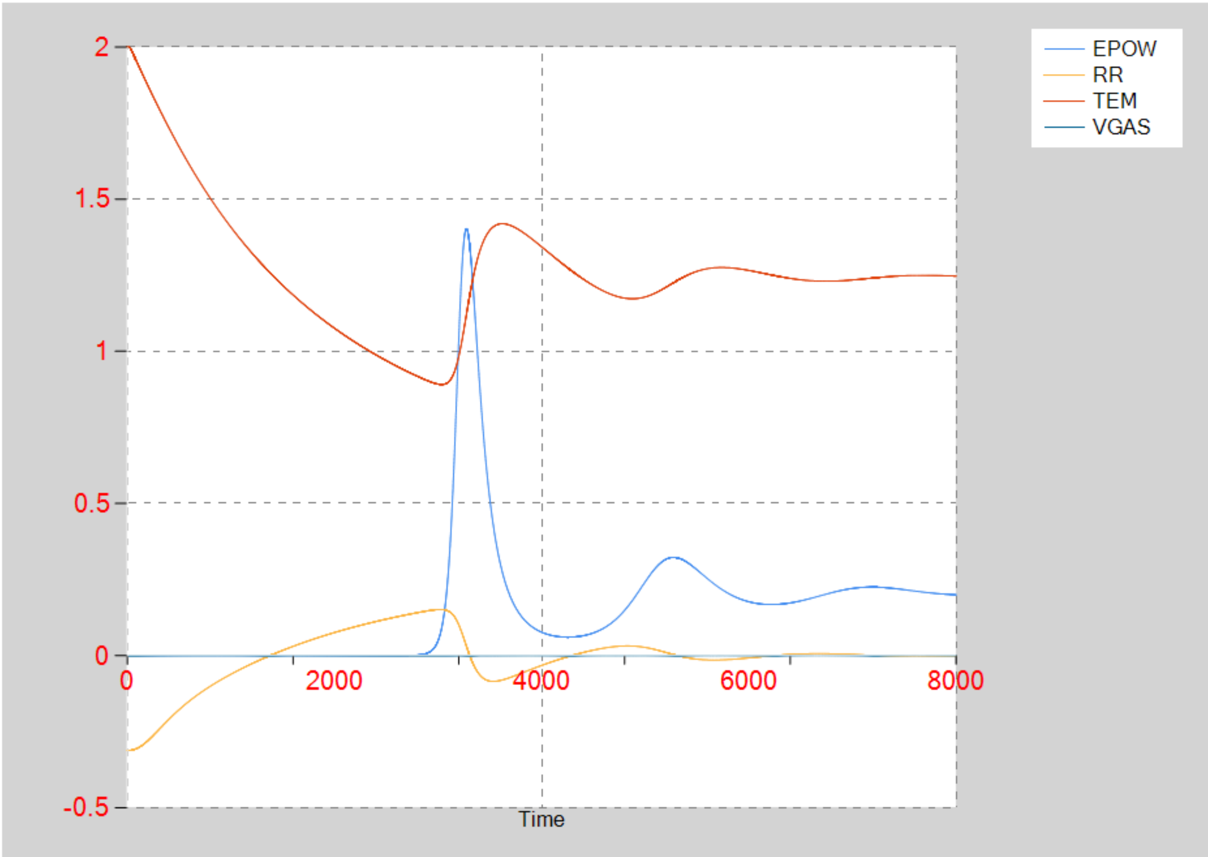

Figure 5: Off-normal scenario, reduced coolant inlet temperature and failure of accelerator at start-up.

Reactivity insertion events may be induced through the Reactivity control element. Both the amount of inserted reactivity and its rate may be controlled to evaluate operator response. Figure 6 is an example of this occurrence. In this scenario, Reactivity is inserted starting at 1000 s with a rate of 0.1 $/s and a maximum reactivity insertion of 1 $.

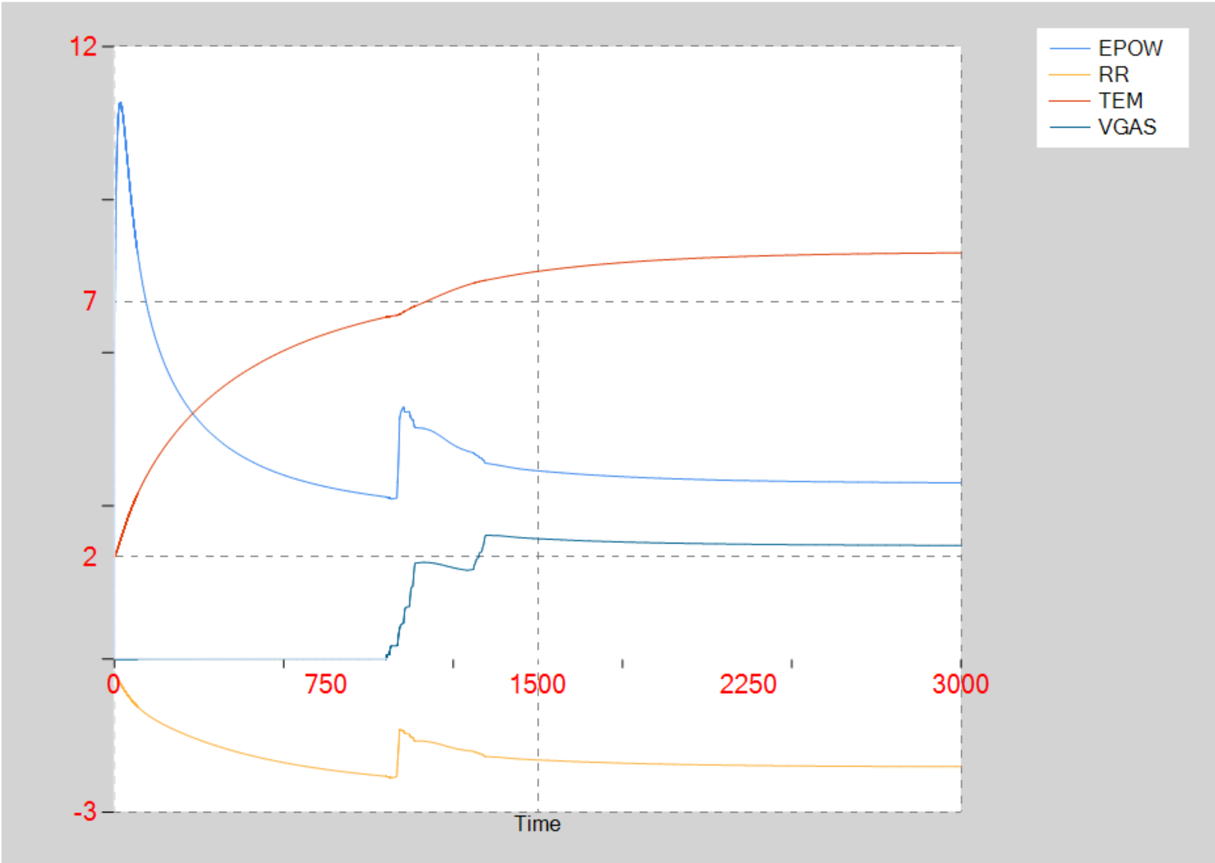

Figure 6: Off-normal scenario, reactivity insertion at 1000 s.

Typical accelerator anomalies are available to the instructor. These include variations in source strength (number of neutrons per second), hard accelerator failure, and oscillations in source strength. In the latter case both the magnitude and frequency of oscillations can be controlled. Figure 7 is an example of system response to an oscillating accelerator.

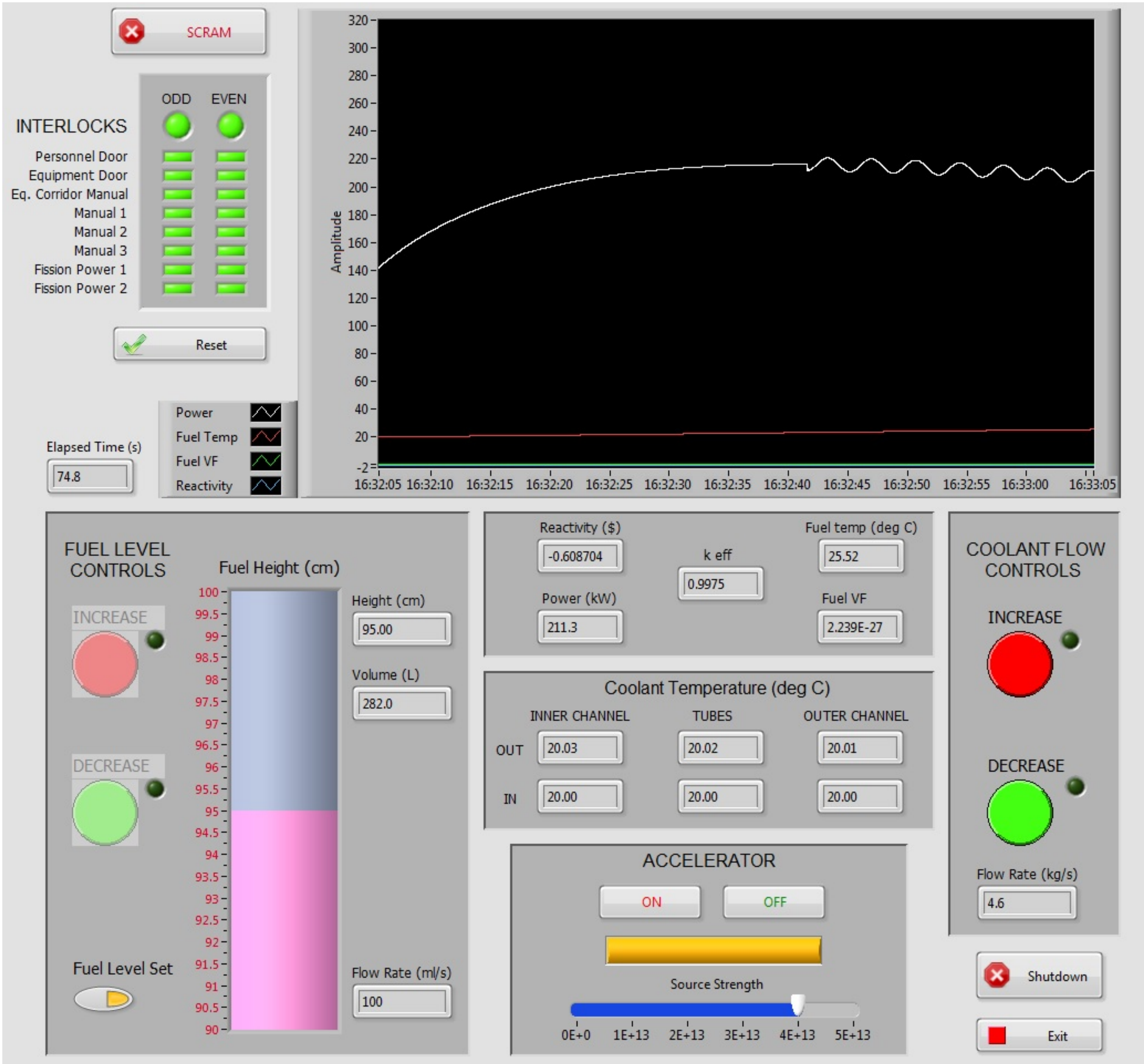

Figure 7: System Response to Oscillating Accelerator.

The final control set available to the instructor includes plenum pressure variations, which will suppress radiolytic gas void, hence increasing reactivity of the system, and a runaway overflow of the fuel tank during fill. The latter case is similar to the one describe above for cold water injection into the cooling system with simultaneous accelerator failure; the system achieves criticality and proceeds to function as an AHR.

Conclusions

LANL has developed a process to generate a progressive family of system models for a fissile solution system. This family includes a dynamic system simulation comprised of coupled nonlinear differential equations describing the time evolution of the system. Neutron kinetics, radiolytic gas generation and transport, and core thermal hydraulics are included in the DSS. Extensions to explicit operation of cooling loops and radiolytic gas handling are embedded in these systems as is a stability model.

The DSS may then be converted to an implementation in Visual Studio to provide a design team the ability to rapidly estimate system performance impacts from a variety of design decisions. This provides a method to assist in optimization of the system design.

Once design has been generated in some detail the C++ version of the system model may then be implemented in a LabVIEW user interface to evaluate operator controls and instrumentation and operator recognition and response to off-normal events.

Taken as a set of system models the DSS, Visual Studio, and LabVIEW progression provides a comprehensive set of design support tools.