BNL-112743-2016

A Statistical Testing Approach for Quantifying Software Reliability: Application to an Example System

Tsong-Lun Chu, Athi Varuttamaseni, Joo-Seok Baek

2016 ANS Winter Conference
Las Vegas, NV

November 2016

Nuclear Science and Technology Department

Brookhaven National Laboratory

U.S. Department of Energy
Office Of Science
American Nuclear Society

DISCLAIMER

A Statistical Testing Approach for Quantifying Software Reliability; Application to an Example System

Tsong-Lun Chu, Athi Varuttamaseni, and Joo-Seok Baek

Brookhaven National Laboratory, 33 N. Renaissance Road, Upton, NY 11973, chu@bnl.gov

I. INTRODUCTION

The U.S. Nuclear Regulatory Commission (NRC) encourages the use of probabilistic risk assessment (PRA) technology in all regulatory matters, to the extent supported by the state-of-the-art in PRA methods and data. Although much has been accomplished in the area of risk-informed regulation, risk assessment for digital systems has not been fully developed. The NRC established a plan[1] for research on digital systems to identify and develop methods, analytical tools, and regulatory guidance for (1) including models of digital systems in the PRA's of nuclear power plants (NPPs), and, (2) incorporating digital systems in the NRC's risk-informed licensing and oversight activities.

Under NRC's sponsorship, Brookhaven National Laboratory (BNL) explored approaches for addressing the failures of digital instrumentation and control (I&C) systems in the current NPP PRA framework. Specific areas investigated included PRA modeling digital hardware[2], development of a philosophical basis for defining software failure[3], and identification of desirable attributes of quantitative software reliability methods[4 7044]. Based on the earlier research, statistical testing is considered a promising method for quantifying software reliability.

It is widely recognized that software failures are due to the triggering of pre-existing defects by the software's operational environment. Software defects can arise from errors made in user requirements or coding errors introduced during the developmental process. The software's operational environment includes factors such as the time history of digital system inputs, communication interfaces, the internal state of the digital system, and external conditions. Thus, software reliability is a function of both the number of pre-existing defects and the presence of a triggering condition caused by the manner in which the software is used.

In this paper, we describe a statistical software-testing approach for quantifying software reliability and applied it to the loop-operating control system (LOCS) of an experimental loop of the Advanced Test Reactor (ATR) at Idaho National Laboratory (INL). The work involved collaboration between BNL and INL.

The objectives of the study include:

(1) Development of a statistical testing approach for estimating software failure probability on demand, the results of which are suitable for including in a probabilistic risk assessment (PRA); and,

(2) Application of the approach to the LOCS to estimate its failure probability, and obtain insights into the feasibility, practicality, and usefulness of the estimation in models of digital systems for inclusion in nuclear power plants' PRAs.

II. SUMMARY OF THE STATISTICAL TESTING APPROACH

The research described in this paper utilizes a statistical testing method (STM) to represent the operational environment and test the software to determine if this environment is capable of triggering pre-existing defects. The test results (number of failures) thus represent operational software failures. Since digital I&C systems (including the software) will be modeled in the NPP PRA sequences, the ways in which the digital system is used will be determined by each PRA sequence. For instance, if one postulated that the digital reactor protection system (RPS) appears in both the primary loss of coolant accident and steam generator tube rupture sequences, the inputs to this RPS and its software (such as the reactor's temperature, pressure, steam-generator's level, steam pressure) would follow different patterns, and different parts of the RPS software would be challenged; consequently, the probability of RPS failure might differ for each sequence. The STM method developed in this research produces test scenarios specific to each sequence and tests the RPS system against these scenarios to generate the sequence-specific probability of software failure.

The STM method consists of the following steps, which assumes that a PRA and an appropriate thermal-hydraulic model have been developed:

1. Select a system under test (SUT);
2. Identify SUT-related PRA sequences (represented by the cutsets);
3. Determine the thermal-hydraulic simulation boundary conditions corresponding to the selected cutsets;

4. Run the thermal-hydraulic model to calculate a time history of the reactor and the plant physical conditions. Such outputs are test scenarios to the SUT;
5. Execute test scenarios and collect test results; and,
6. Analyze the test results to quantify the probability of software failure.

III. APPLICATION TO AN EXAMPLE SYSTEM

In this study, BNL selected a Loop Operating Control System (LOCS) for the ATR at INL as the SUT. The ATR has six in-pile tubes (IPTs) through which water circulates at a set pressure, temperature, and flow rate. The LOCS normally controls the condition of an experimental loop, and will generate a reactor trip signal when the pressure, temperature or flow exceeds its threshold. Figure 1 shows the logic involved for the IPT inlet's temperature-protection channel. It illustrates how the different protective functions are logically connected to each other via an OR gate. Hence, regardless of the channels that initiate the trip, the three digital output-modules normally should be in the same state (i.e., the status of all three should show either a trip or non-trip). In all, two 2/3 logics are used in the protection channels: one is used at the level of a sensor/analog input module; a second is at the level of the digital-output modules.

INL provided BNL with the ATR PRA and RELAP5 models relevant to the LOCS system. BNL revised these models to make them STM-friendly. Based on PRA accident sequence information (e.g., LOCS-relevant cutsets), the thermal-hydraulic model was used to simulate the experimental loop conditions (e.g., pressure, temperature, and flow) during the selected accident sequences in order to provide realistic input signals to a LOCS test platform developed at INL. To ensure that the test cases provided adequate coverage of operational conditions, thirteen probabilistic failure process models were developed to represent the variabilities associated with timing, component failure modes, and process variable control (See Table 1).

An important reactivity insertion scenario is a loss of secondary cooling to the experimental loop. It causes a heat up of the loop and an increase in reactivity in the core. The LOCS needs to detect the thermal hydraulic condition in the loop and generate a trip signal. The thermal hydraulic model of the loop does not model the components of the secondary side. It simply models the secondary side in terms of a heat transfer coefficient at the heat exchanger and a secondary side coolant temperature. In order to capture the variability of different loss of secondary cooling scenarios, a probabilistic failure model was developed. It models the rate at which the heat transfer coefficient changes from its normal value to zero.

Figure 2 gives a high-level view of the testing process. RELAP5 simulations of reactivity-insertion scenarios derived from a probabilistic risk assessment (PRA) were used to generate input files for testing the LOCS. Each such file consists of time-stamped records with values of physical parameters. In addition, INL added a time-pulse analog-signal for estimating the cycle time of the LOCS. The Testing Host Computer took the time-stamped records, converted the values of the physical parameters into analog signals, and fed them to the LOCS. It also received the output trip signals and an output heartbeat signal from LOCS as test results. The host computer then generated time-stamped records of these outputs, saving them in a file with the test results. These results then were evaluated to determine if a trip signal was generated in time, based on a predefined success criterion. The successes and failures of the tests were used in estimating the system's failure probability.

For this study, 10,000 different test cases were used to demonstrate a reliability level consistent with PRA and design requirement assumptions. The test cases did not trigger any pre-existing software defects. The testing did identify one potential failure that was not reproducible and was determined to be caused by the test equipment setup. However, a few early trips and delayed trips were observed and were further analyzed. The analysis demonstrated that the anomalies were likely caused by the inaccuracy of the analog input/output modules. The LOCS system was tested as a black box in this study and therefore both hardware failures and software failures could be detected. Therefore, the per-demand failure probability determined by this testing approach considered both hardware and software failure likelihood in an integrated fashion.

ACKNOWLEDGMENTS

The authors acknowledge the collaboration of INL staff on the research and the participation of the NRC project manager.

REFERENCES

1. T.L. CHU, et al., "Development of a Statistical Testing Approach for Quantifying Software Reliability and Its Application to An Example System," Draft report, available at NRC website, ML14294A232 (2014).
2. U.S. NUCLEAR REGULATORY COMMISSION, "NRC Digital System Research Plan FY2010-FY2014," (2010).
3. T.L. CHU, et al., "Modeling a Digital Feedwater Control System Using Traditional Probabilistic Risk Assessment Methods," NUREG/CR-6997 (2009).

4. T.L. CHU, et al., “Workshop on Philosophical Basis for Incorporating Software Failures into a Probabilistic Risk Assessment,” Brookhaven National Laboratory, Technical Report, BNL-90571-2009-IR (2009).
5. T.L. CHU, M, YUE, G. MARTINEZ-GURIDI, and J. LEHNER, “Development of Quantitative Software Reliability Models for Digital Protection Systems of Nuclear Power Plants,” NUREG/CR-7044 (2013).

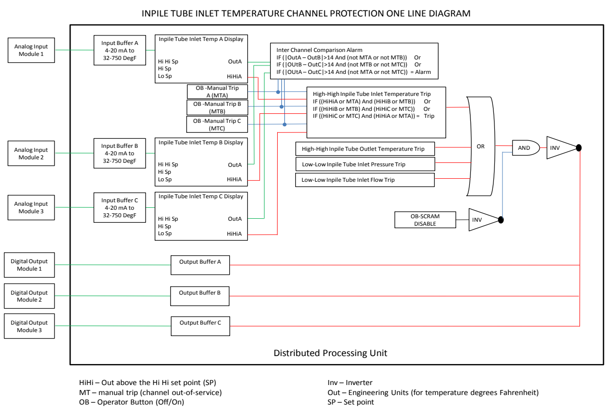

Fig. 1 Typical processing logic of the loop-protective channel

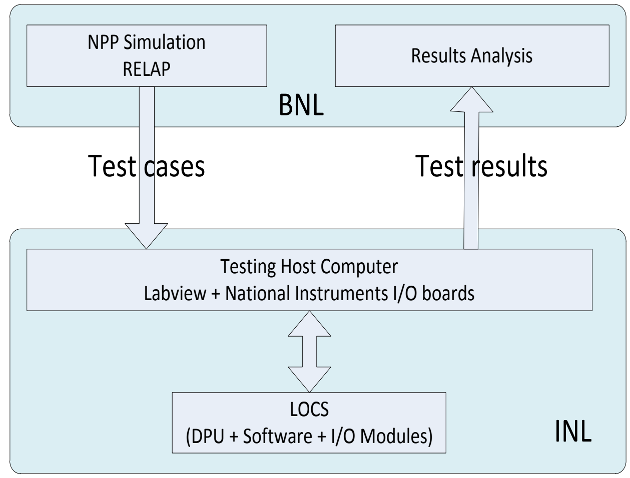

Fig. 2 Work flow associated with performing the tests

TABLE I. Probabilistic Modeling of Failure Effect Categories.

No.	Failure Effect Category	Subcategory [Frequency of Subcategory]	Parameter	Lower Bound	Upper Bound
1	Loss of HX cooling	-	Time at which the heat transfer coefficient reaches zero. [s]	0	1670
2	Pump Failure	Trip [49%]	Multiplication constant to the time variable for the pump's coastdown curve	0.5	1.5
3		Seizure [51%]	Time for pump to reach complete stop [s]	0.001	2.04
4	Pipe Plugging	Plugging at FE1 [33.3%]	Flow area at junction 855 [ft^2]	1.00E-08	6.3580E-04
5		Plugging at FE2 [33.3%]	Flow area at junction 856 [ft^2]	1.00E-08	6.5630E-04
6		Plugging at S145 [33.4%]	Flow area at junction 857 [ft^2]	1.00E-08	6.3580E-04
7	Pipe Break	Break at IPT Inlet [50%]	Flow area at valve 851 [ft^2]	6.3840E-06	7.5100E-04
8		Break at IPT Outlet [50%]	Flow area at valve 853 [ft^2]	6.1500E-06	9.4300E-04
9	Loss of flow control - input	-	CV-240 (Flow sensor input) [gpm]	30.06	35.1
10	Loss of flow control - output	-	CV-24 (Flow controller output) [flow area ratio]	0	0.382423
11	Loss of line heater control – input	-	CV-1 (490°F - Temperature sensor input) [^{o}F]	45	490
12	Loss of line heater control – output	-	CV-4 (Line heater controller output) [W]	1.799637E+05	2.16E+05
13	Loss of TCV control	-	Time for valve TCV-3-1 to be fully closed. [s]	15	45