Exhausted Test Case for Software Reliability of Nuclear Digital Systems

Jaehyun CHO[1*], Sung Min SHIN[1], Seung Jun LEE[2], Wondea JUNG[1]

[1] Korea Atomic Energy Research Institute: 989-111 Daedeok-daero, Yuseong-gu, Daejeon, 305-353, Republic of Korea
[2] Ulsan National Institute of Science and Technology: UNIST-gil 50, Ulsan 689-798m Republic of Korea
**chojh@kaeri.re.kr*

Digital instrumentation and control (I&C) systems have several specific characteristics comparing with analog I&C system. One of the important part of digital I&C systems is a software part, however, even after several years of research, quantification of software reliability remains unresolved issue. This paper suggests the method to obtain the exhausted test cases of safety-critical software. Variables in the software is composed of plant variables and internal variables. If we consider the full combinations of all range of software variables, the number of test case is so tremendous that testing is physically not possible. In order to obtain the realistic number of test case having "exhausted" concept, the range of variables should be reasonably considered. In the developed method, for the plant variables, the available range of plant variables are obtained based on the results of best-estimated thermal-hydraulics code. For the internal variables, we conducted dependency analysis which is to identify the relationship between all internal variables. By doing this, internal variables that are directly or indirectly effectible to the trip were selected. With the testing of obtained exhausted test case, we can confirm the software's high reliability.

I. Introduction

The introduction of digital instrumentation and control (I&C) systems in nuclear power plants (NPPs) instead of an analog I&C system is a very natural line of development, as the application of digital I&C indeed offers many advantages, including stability from zero drift, huge data capacity, and design flexibility. One of the important and unique part of digital systems is a software. Because the software is composed of a number of logics, the reliability of it is totally different with conventional hardware reliability. In order to assess safety and identify vulnerability of the digitized system in NPPs, software reliability should be considered.

Software reliability is defined as the likelihood of the failure free function of software in a given period of time under some certain circumstances.[1] There has been several studies that aimed to quantify software reliability. Among the several researches, software reliability growth model (SRGM) has been widely used in analyzing and estimating the software reliability. In SRGMs, it is assumed that the software is fixed perfectly and instantaneously after failures, so that its reliability "grows" with time, that is, the software failure rate declines with time.[2] Based on the software reliability obtained by SRGMs and desired level of software reliability, software release time to customers and the required testing resources are determined.

In NPPs, however, safety is not negotiable one with economic aspect. For the licensing of digital systems in not only current NPPs but also new NPPs, almost zero-defect software has being required by regulatory body. Safety-critical software is required to be highly reliable such that the failure probability should be less than 10^{-5} per demand. As SRGMs is based on the statistical testing that does not cover entire range of software test case, SRGMs is not applicable to high-reliability-required filed such as safety-critical software of NPPs. To solve this limitation, Kang and co-authors suggested an exhausted test case development method.[3] In this method, input-profile reduces the number of test cases we should do to satisfy zero software failure drastically. However, the test case needs a combination of all variables used in software. Input-profile method does not consider the software internal variables.

The present study proposes a new method for confirmation the software reliability. The proposed method aims to obtain the exhausted test cases of safety-critical software including internal variables and plant response. The method has also been applied to specific digital reactor protection systems to obtain the exhausted test cases of the specific systems.

II. Method

Fig.1 shows the configuration of digital systems highlighting the all variables we should consider in the testing. For the exhausted testing, all variables regarding with digital systems should be identified. There are two parts of variables in the digital systems: plant variables, and software internal variables. Plant variables consist of analog type and digital type. Analog one is a signal of the sensors to monitor the plant transient such as pressurizer pressure, and steam generator water level. These signals are form of voltage or current. Thus, in order to use the analog signals in the digital systems, they should be transferred to a digital number. In NPP's digital systems, there is analog-to-digital converter (ADC) to convert a continuous physical quantity to a digital number. On the other hand, some of plant variables are converted to digital signals in other digital systems such as core protection calculator (CPC). The first-of-its-kind systems was installed in the United States at Arkansas Nuclear One Unit 2 (ANO-2) in 1980.[4] In NPP's of Korea, CPCs are installed in not only Optimized Power Reactor (OPR 1000) but also Advanced Power Reactor (APR 1400). CPC uses ex-core detector signals for core power calculation and calculates additional safety parameters such as departure from nucleate boiling ratio (DNBR) and local power density (LPD). The reliability of CPC is not considered in this research.

In digital systems, there are a number of software internal variables. These variables are connected with software logic. The software logic is expressed by unit function module. Every function module conduct its specific function. One function module may compare the analog plant variables with trip signal value which is already fixed in digital systems. Other one may just transfer the digits to other function module. In a one function module, there are several inputs and several outputs. The last function module gives the trigger signal to safety systems such as reactor protection system (RPS), safety coolant injection, and auxiliary feed water activation. Below two sub-chapter describe the method to obtain the exhausted test case for plant variables and software internal variables, respectively.

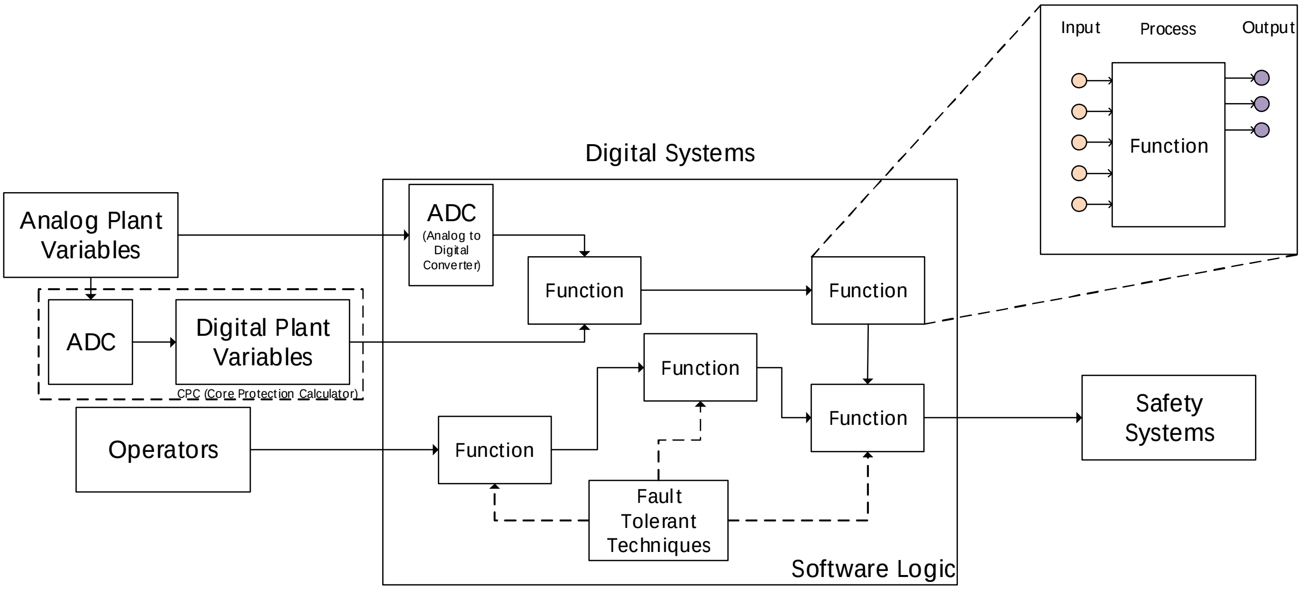

Fig. 1. Configuration of digital systems

II.A. Input Profile for Plant Variables

Fig. 2 shows the general idea of input profile for plant variables. This method was already developed by Kang and co-authors[3], to obtain the exhausted test case of plant variables. When abnormal situation occurs, plant variable is smoothly or rapidly changed by given core power and plant configuration, and finally reaches to trip set point (TSP). ADC converts continuous sensor signal into digit numbers every scan time. ADC scan interval which means the time difference between two successional scan times, is normally 100~200 milliseconds. In fig.2, the measured value at the time before S_5 are below than TSP, thus trip does not occur. At the time T_m which is right after S_5, measured value eventually reaches to TSP. Thus, we expect the trip occurs at T_m. However, trip does not occur due to scan time. At the time between S_5 and S_6, digital number is fixed to V_5 which is the measured value at S_5. The digital number is updated to V_6 which is the measured value at of S_6. Thus, trip occurs at the time of S_6.

In view of safety, software testing of digital system is valid for the situation only that trip should be occur. For the values less than trip set point, we does not expect trip, thus testing is not useless. The points from TSP to V_6 which is the maximum plant variables change during one scan time are required for testing. Thus, the number of test case is depended on plant transient rate, resolution, and scan interval. As transient velocity is faster, resolution is higher, and scan interval time is smaller, the number of test case is larger.

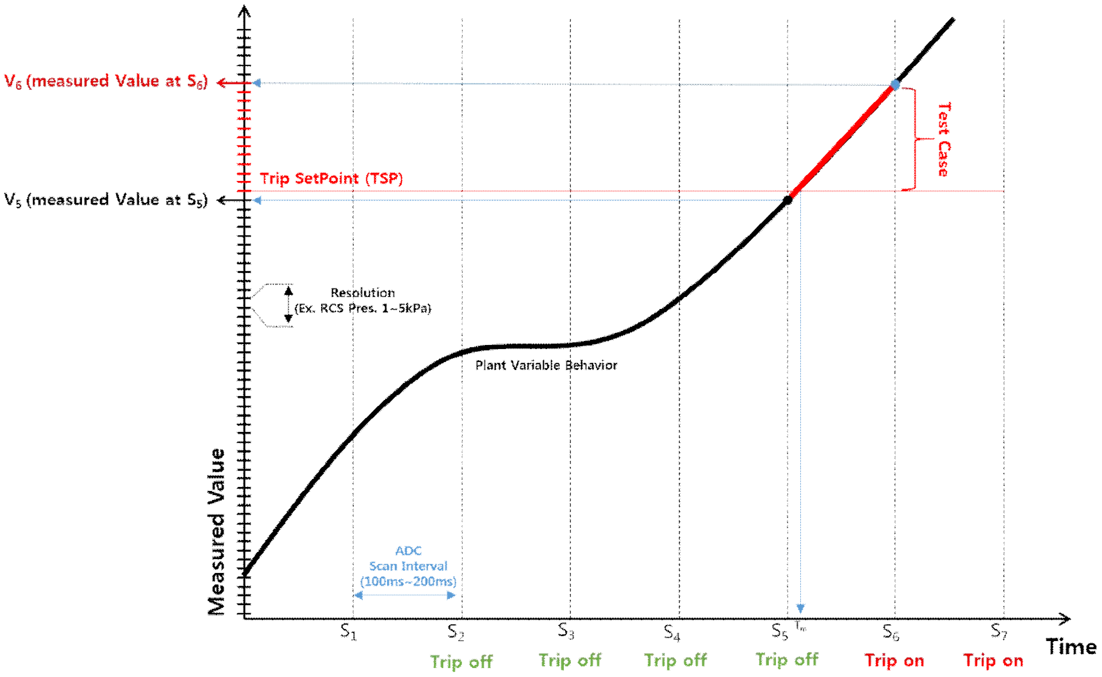

Fig 2. Diagram for input profile based test case development method

II.B. Software Internal Variables

Fig. 3 shows example for structure of digital software internal variables. There are three function blocks (indication A, B, and C) and each function has several inputs and outputs. Function block A has five inputs including two plant variables, and one operator action. Function block calculates its logic and two outputs also come out. Function block-B has three inputs and two outputs. Two inputs are directly connected to outputs of function block A. Function block C uses outputs from function block-A and -B as its inputs. The output of function block-C is connected to safety system.

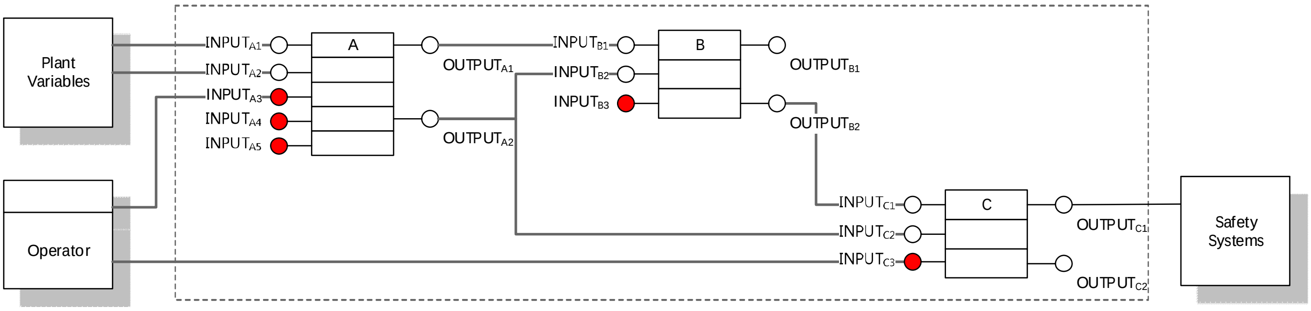

Fig. 3. Example for structure of digital software internal variables

If we consider all input space of all software variables in order to obtain exhausted test case, then, tremendous test cases will be obtained. It can be physically impossible in view of time and money. In view of the safety, we should select the variables which have an effect on activation of safety systems. In fig. 2, safety system will be activated by output$_{C1}$. Inputs of function block-C are output$_{B2}$ of function block-B, output$_{A2}$ of function block-A, and operator action. Output$_{C1}$ is dependent on input$_{C1}$, input$_{C2}$, and input$_{C3}$. Thus, output$_{C1}$ does not be considered in test case. Input$_{A1}$, and input$_{A2}$ are already considered in input profile for plant variables (Section II.A). In this manner, the critical variables which means give an effect to operation of safety systems can be selected. In this example, input$_{A3}$, input$_{A4}$, input$_{A5}$, input$_{B3}$, and input$_{C3}$ are critical variables. Then, we can make exhausted test case considering all input space of above five software variables.

III. Results

III.A. Input Profile for Plant Variables

The purpose of this research is to obtain the exhausted test case. In order to get the exhausted case for plant variables, we should consider all abnormal situation threaten plant's safety. In the nuclear deterministic safety field, NPP's accidents are divided by two categories: design-basis accidents (DBA) and beyond DBA. Currently, design extension conditions (DEC) are also introduced in beyond DBA category. On the other hand, in the probabilistic safety filed, probabilistic safety assessment (PSA) quantify the plant risk with consideration of both together DBA and beyond DBA. PSA considers all probable events and then, quantify the core damage frequency and large early release frequency. Based on frequency of initiating events, very low frequency-events may be sorted and neglected. Therefore, if we make input profile for all PSA initiating events and its scenarios, then it is exhausted.

The target PSA model is for the OPR 1000 RPS. There are 18 initiating events and a number of scenarios in each initiating event. From the preliminary thermal-hydraulics results, we knew that all scenarios in each initiating event has same input profile because time to reactor trip is always faster than any accidents sequences. Below initiating events do not be considered for input profiles. Table I shows the considered initiating events for input profile development.

- Large-break-size, medium-break-size LOCA, reactor vessel rupture: a large release of coolant accelerates the coolant voiding of the reactor core and large negative void coefficients cause the nuclear fission power shutdown.
- Interfacing System LOCA: because so many cases are available, and the plant behavior is very similar with small-break-size LOCA, so it is regarded as small-break-size LOCA.
- Anticipated transients without scram: this event is originally malfunction of reactor trip.
- Loss of off-site power and station black-out: because electricity of RPS is main problem, it is not concern with software logic testing.

TABLE I. Initiating events and its reactor trip type of OPR 1000 for input profile development

Initiating Event	Accident Sequence	Reactor Trip by RPS
Small-break-size LOCA	Reactor trip timing is always less than trigger time of any accident sequences	PZR pressure - low pressure trip
Steam generator tube rupture (SGTR)		PZR pressure - low pressure trip
Large secondary side break (LSSB)		SG pressure - low pressure trip
Loss of feed water (LOFW)		SG water level - low water level trip
General Transient (GTRN)		SG water level - low water level trip
Loss of condenser vacuum (LOCV)		SG water level - low water level trip
Loss of component cooling water (LOCCW)		SG water level - low water level trip
Loss of a 125 DC Bus (LODC)		SG water level - low water level trip
Loss of a 4.16 KV Bus (LOKVA)		SG water level - low water level trip

The ADC resolution is 12-bits in the RPS of the OPR 1000.[5] In the application, 12-bits and 14-bits for ADC resolution, 100 and 200 milliseconds scan times were considered. 12-bits and 14-bits ADC have resolution of 4,096 (2^{12}), 16,384 (2^{14}), respectively. It means that, in case of 12-bits resolution, ADC divides continuous signals from sensors into digital number of 4,096 equal parts. Measuring range of PZR pressure, SG pressure, and SG water level are 0 - 20.68 MPa, 0 - 10.50 MPa, and 0 – 100 %, respectively. Thus, in case of 12-bits resolution, unit measuring of PZR pressure, SG pressure, and SG water level are 5,049 Pa, 2,563 Pa, and 0.0244 %, respectively.

TALBE II shows the results of number of test case for all PSA initiating events. These numbers were obtained by each input profile. In order to develop the input profile of each event, we used MARS KS code which is developed by KAERI for thermal-hydraulic analysis of NPPs. In case of small-break-size LOCA and SGTR, the number of test case is very small (almost are just one) because small break size make plant transient slow. On the other hand, the number of test case of LSSB is relatively large because a large amount of steam released in secondary side make plant transient fast. When we assumed 14-bits resolution and 200ms scan interval, the number of test case of LSSB is 262.

TABLE II. The number of test case for all PSA initiating events
(100, and 200ms of scan interval, 12-bits and 14-bits of ADV resolution)

ADV resolution		12-bits	14-bits
Initiating Event	**Scan Interval**	**The number of test case required**	
Small-break-size LOCA (0.5inch)	100ms	1	1
	200ms	1	1
Small-break-size LOCA (2.0inch)	100ms	1	2
	200ms	1	4
SGTR	100ms	1	1
	200ms	1	1
LSSB	100ms	32	126
	200ms	66	262
LOFW, GTRN, LOCV, LOCCW, LODC, LOKVA	100ms	4	15
	200ms	8	31

III.B. Software Internal Variables

For the application study of software internal variables, function block diagram of Bistable Processor (BP) in Integral Digital Protection System-Reactor Protection System (IDiPS-RPS) was considered. IDiPS-RPS has been developed under the Korea Nuclear I&C Systems (KNICS) project.[6]

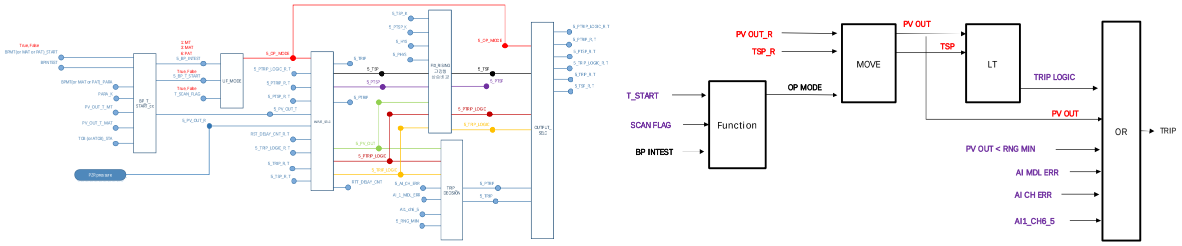

Fig. 4. Function block diagrams with variables in IDiPS-RPS BP
(a) Overall variables (b) Selected variables

TABLE III. Failure Mode and Criticality Level of Selected Variables

Failure Mode	Variable	Process	Criticality Level
Incorrect Input	**PV OUT**		**4**
	TSP		**4**
	RNG MIN		3
	AI MDL ERR		3
	AI CH ERR		3
	AI1_CH6_5		3
	T-START		2
	SCAN FLAG		**4**
	BP INTEST		2
Switched input	**PV OUT / TSP**	**LT**	**4**
	PV OUT / TSP	**MOVE**	**4**
	PV OUT / RNG MIN	OR	3
	T_STRAT / SCAN FLAG	**FUNCTION**	**4**
Extreme Input	PV OUT (maximum)		3
	PV OUT (minimum)		**4**
	TSP (maximum)		**4**
	TSP (minimum)		3

Fig. 4 illustrates function block diagrams with software variables. Left one (a) shows the function block diagram with all variables, and right one (b) shows the selected function block diagram which could give an effect to trip signal. From this analysis and consideration of dependency between variables, nine variables are selected: T START, SCAN FLAG, BP INTEST, PV OUT, TSP, RNG MIN, AL MDL ERR, AL CH ERR, and Al1_CH6_5. TABLE III is the results of failure mode and criticality level of selected variables. Criticality level 3 is spurious trip that trip occurs when trip is not required. Criticality level 4 is critical failure that trip does not occur when trip is required. In view of safety, we identify the variables related with critical level 4 failure. 4 variables were selected: PV OUT, TSP, SCAN FLAG, T-START.

IV. Summary and Conclusion

The aim of this study was to develop a quantification method of reliability for safety-critical system's software. It is because that software is essential part in the digital systems and SRGMs which is normally used in commercial software reliability, is not adequate to NPP's safety-critical systems. In order to assure the high reliability of software, software testing approach is needed. The research question was that how to obtain the exhausted test cases which is physically available. In order to solve the problem, test case for plant variables and test case for software internal variables were considered, respectively. For plant variables, input profile based test case development method was introduced. In application, input profile of all initiating event with all accident sequences in PSA were analyzed by thermal-hydraulics code simulation and the number of test case was estimated with two ADC resolution and two scan interval. For software internal variables, dependency of internal variables was considered and critical variables were obtained. The limitation of this research is that, in application of internal variables, we just apply the method to RPS BP. In further research, all processors of digital software should be considered.

ACKNOWLEDGMENTS

This work was supported by Nuclear Research & Development Program of the National Research Foundation of Korea grant, funded by the Korean government, Ministry of Science, ICT & Future Planning (Grant Code: 2012M2A8A4025991).

REFERENCES

1. ANSI/IEEE, Standard Glossary of Software Engineering Terminology, STD-729-1991, ANSI/IEEE, 1991
2. American National Standards Institute (ANSI)/ American Institute of Aeronautics and Astronautics (AIAA), "Recommended Practice for Software Reliability," American National Standards Institute, ANSI/AIAA R-013-1992, February 23, 1993.
3. H. G. Kang, H. G. Lim, H. J. Lee, M. C. Kim, S. C. Jang, "Input-profile-based software failure probability quantification for safety signal generation systems," Reliability Engineering and System Safety, 94, pp. 1542–1546, 2009.
4. Peter L. Hung, Core Protection Calculator System: Past, Present, and Future, 18th International Conference on Nuclear Engineering, ICONE18-29001, pp.585-592
5. Westinghouse Electric Company. Ulchin nuclear power plant units 5 and 6: digital plant protection system technical manual, 2002.
6. KAERI, "Reactor Protection System Design Specification", KNICS-RPS-DS101 (2006).