Nuclear Engineering and Design 316 (2017) 198–208

Contents lists available at ScienceDirect

Nuclear Engineering and Design

journal homepage: www.elsevier.com/locate/nucengdes

# Fault-weighted quantification method of fault detection coverage through fault mode and effect analysis in digital I&C systems

Jaehyun Cho, Seung Jun Lee *, Wondea Jung

*Korea Atomic Energy Research Institute, 1405 Daedeok-daero, Yuseong-gu, Daejeon 305-353, Republic of Korea*

## HIGHLIGHTS

- We developed the fault-weighted quantification method of fault detection coverage.
- The method has been applied to specific digital reactor protection system.
- The unavailability of the module had 20-times difference with the traditional method.
- Several experimental tests will be effectively prioritized using this method.

## ARTICLE INFO

*Article history:*
Received 21 August 2015
Received in revised form 9 March 2017
Accepted 10 March 2017
Available online 17 March 2017

*Keywords:*
Digital I&C system
Probabilistic safety assessment
Fault injection
Fault-tolerant technique
Fault detection coverage

## ABSTRACT

The one of the most outstanding features of a digital I&C system is the use of a fault-tolerant technique. With an awareness regarding the importance of thequantification of fault detection coverage of fault-tolerant techniques, several researches related to the fault injection method were developed and employed to quantify a fault detection coverage. In the fault injection method, each injected fault has a different importance because the frequency of realization of every injected fault is different. However, there have been no previous studies addressing the importance and weighting factor of each injected fault. In this work, a new method for allocating the weighting to each injected fault using the failure mode and effect analysis data was proposed. For application, the fault-weighted quantification method has also been applied to specific digital reactor protection system to quantify the fault detection coverage. One of the major findings in an application was that we may estimate the unavailability of the specific module in digital I&C systems about 20-times smaller than real value when we use a traditional method. The other finding was that we can also classify the importance of the experimental case. Therefore, this method is expected to not only suggest an accurate quantification procedure of fault-detection coverage by weighting the injected faults, but to also contribute to an effective fault injection experiment by sorting the importance of the failure categories.



## 1. 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 main outstanding features of digital I&C systems is the fault-tolerant technique, i.e. a system's capability for helping the system, through software, to correctly perform specific required functions despite the presence of faults. In order to assess safety and identify vulnerability of the digitalized system in an NPP, fault-tolerant techniques and their fault detection coverage should be considered in the PSA fault tree (FT) (Kang et al., 2009).

In their study of sensitivity, Kang and Sung (2001) showed the important role of fault detection coverage in the quantification of digital system unavailability. With the awareness of importance for quantification of fault detection coverage, there have been several studies that aimed to quantify fault detection coverage. One of the current techniques for quantifying it is the fault injection method. Software-implemented fault injection and a limited hardware-implemented fault injection techniques have been developed and employed to quantify fault detection coverage corresponding to fault-tolerant techniques of a specific digital I&C system (Choi et al., 2012; Lee et al., 2006, 2010).

However, up to now, fault injection experiments were conducted for a feasibility study; therefore, the experiments were

* Corresponding author.
*E-mail address:* sjlee420@unist.ac.kr (S.J. Lee).

http://dx.doi.org/10.1016/j.nucengdes.2017.03.016
0029-5493/

conducted under limited conditions (Lee et al., 2010). The first limitation is that a limited memory area was examined due to the limited experimental time. The experiments used only 1.33% of the whole memory area. However, this problem will be solved by using a high efficiency computing tool by increasing the test speed. Furthermore, the software-implemented fault injection test method recently developed by Lee and Jung (2015) can rapidly perform the experiment test. The second limitation of prior research is that each fault injection test was given the same weighting. Each injected fault in the experiments has a different importance, as the frequency of realization of every injected fault is different. With the increase of likelihood of realization, its importance and weighting also increase. However, since no previous studies addressed the importance and weighting factor of each injected fault, the values computed by fault injection experiments were not correct, in this view.

### 1.1. Configuration of digital I&C systems

In view of safety concerns, a digital I&C system consists of three parts: hardware, software, and fault-tolerant techniques. Fig. 1 shows the configuration of digital I&C systems highlighting the function of fault-tolerant techniques. The hardware consists of several processors, such as bistable processors (BP) and coincidence processor (CP). These processors have specific functions that are essential for an I&C system, such as data processing, voting, and data communication. A processor consists of several modules, such as digital input module (DIM) and process module (PM). These modules conduct specific tasks, including data gathering, data transferring, and power supply management; furthermore, each module may be distributed to processors to perform its own function. A module consists of a number of components, such as a chip ceramic capacitor or a switching diode (NEA, 2014). Some of the components may have trouble over time (i.e. a fault). The trouble can propagate to a module and further a processor. When the fault leads to a malfunction of the essential function of hardware, it is regarded as a failure of hardware. On the other hand, software initially has internal faults made during the software designing phase. The internal faults can cause a failure in certain conditions, such as an operator mistake or a data input error.

In order to improve the reliability of digital I&C systems, various fault-tolerant techniques, such as self-diagnostics of each component, a heartbeat check of the watchdog timer, and periodic automatic testing, have been implemented in digital I&C systems. Some of the failures originating from hardware and software could be detected by fault-tolerant techniques; then, a system automatically becomes a safe state. However, since fault-tolerant techniques cannot detect all possible failures, they are not perfect. Furthermore, each fault-tolerant technique has a different detection period. Some fault-tolerant techniques make the system automatically generate fail-safe signals for the system to enter a safe state; others simply provide an abnormal status warning to human operators. Due to imperfection of fault-tolerant techniques, a digital I&C system may fail (Choi et al., 2012). Fault detection coverage is defined as the probability of detecting failures.

### 1.2. Fault injection experiments

Fault injection is used by engineers to test fault-tolerant techniques. These include hardware-implemented fault injection and software-implemented fault injection (Hsueh et al., 1997). Hardware-implemented fault injection uses additional hardware to inject faults into the target system's hardware, so it is obviously a time- and cost-consuming method. Considering that reliability of digital systems in NPP, fault injection experiments require exhausting testing that covers the whole range of the system. Therefore, using hardware-implemented fault injection is practically impossible. In this study, only software-implemented fault injection was considered.

In the fault injection experiment developed by Lee et al. (2010), faults are intentionally injected into the memory, of the several modules. It is assumed that all faults in a system are reflected in the faults in the memory, because a fault should affect the memory related to the calculation process, reading input variables, generating output variables, and so on. After fault injection, we can measure whether injected faults affect or do not affect the system output. The fault injection experiment is illustrated in Fig. 2.

The bit changed by an injected fault may cause a wrong output or no output of the system. Otherwise, a system does not fail in spite of an injected fault. This is because some memory area is not assigned to any program code or variable; also, the changed bit is not directly related to the output generation. A wrong output or no output caused by the injected faults can be detected by fault-tolerant techniques or not. The fault detection coverage is obtained by dividing the number of detected faults by the number of injected faults.

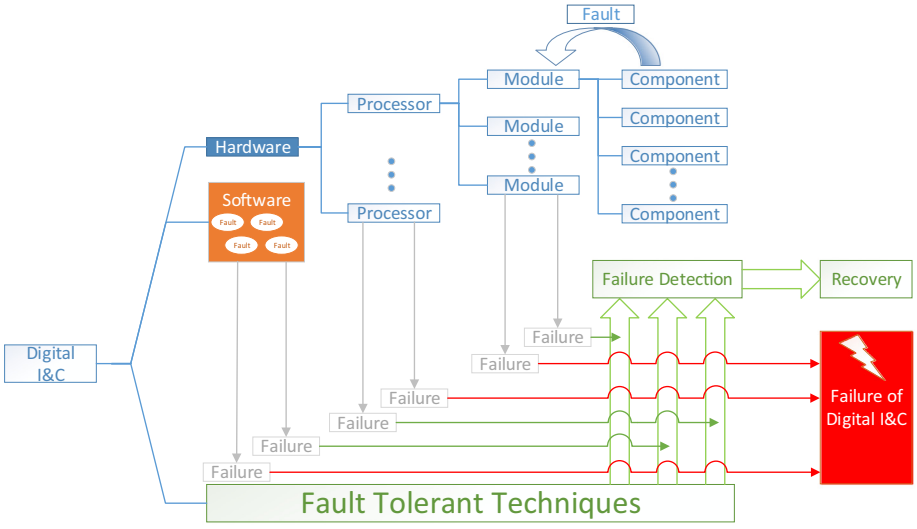


**Fig. 1.** Configuration of the digital I&C systems.