Annals of Nuclear Energy 92 (2016) 289–294

Contents lists available at ScienceDirect

Annals of Nuclear Energy

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

Technical note

Ensuring safety in design of safety critical computer based systems

Lalit Kumar Singh [a,*], Hitesh Rajput [b]

[a] Dept. of Computer Sc. & Engg., IIT (BHU), Varanasi, India
[b] Dept. of Mathematical Sciences, IIT (BHU), Varanasi, India

ARTICLE INFO

Article history:
Received 16 September 2015
Received in revised form 27 November 2015
Accepted 3 February 2016
Available online 16 February 2016

Keywords:
Nuclear power plant
Petri net
Reactor protection system
System safety
System reliability

ABSTRACT

Safety critical systems are designed to function in safe manner so that its failure should not lead to the catastrophic effects, including injury or death to humans, and harm to the environment. These systems take themselves to a safe state, thus ensuring goals of safety. Due to safety significance of such systems, these need to be designed carefully to ensure their reliability requirements. The strategy discussed the modeling and analysis techniques to safety critical computer based systems using Petri net for full proof design. The techniques to improve the faulty design are also proposed. The application of the proposed techniques is shown on a reactor protection system.

1. Introduction

Safety critical systems can cause injury or loss of human life, if they fail. Some of the prominent areas in which such systems are installed are aircrafts, nuclear power plants and medical systems. Due to their intolerable safety threats, such systems are designed such that their failure may not lead to the catastrophic disaster. Hence there should be a robust technique to model such systems and to analyze the safety issues in the constructed model to take preventive actions. We devise our technique on a protection system of NPP.

NPP has protection systems to meet the primary objectives, as per U.S. Nuclear Regulatory Commission (Glossary, 2015).

RPS is a complex control system comprising several safety electronic and mechanical components, known as nuclear safety components, to meet the above three objectives. The emergency shutdown of the reactor in need is known as SCRAM (Eide et al., 1999). RPS trips the reactor in two diverse ways, for which there are two separate systems. These systems are independent and are known as SDS1 and SDS2. In case of PHWR, SDS1 trips the reactor by inserting all control rods into the moderator. These rods are capable to absorb the neutrons and hence stop the nuclear chain reaction. SDS2 trips the reactor by inserting gadolinium nitrate solution into the moderator via liquid poison injection nozzles. Gadolinium nitrate solution is capable to absorb the neutrons to stop the nuclear chain reaction.

Considering the importance of RPS, it must be designed to meet the high reliability requirements, as per the regulatory body. The work in this paper focuses on the safety analysis of the RPS system during its design phase. This approach is valid for any kind of hardware, software or computer based systems. We have taken SDS2 as a case study.

The related work and their limitations, proposed in the current literature is given in Section 2. Section 3 discusses SDS2, as a case study, in detail. Our approach for safety analysis of SDS2 is shown in Section 4. In Section 5, we show the methodology to take preventive action, in case the design contains an acceptable level of risk. Section 6 concludes this paper.

2. Related work

Lalit et al. (2013, 2014, 2012) researchers attempted to do probabilistic analysis of the system for ensuring reliability. But the analysis done in these papers are to quantify the reliability, where the emphasis is not given to address the safety issues on the proposed design. We need to look into different perspective, if it is to be

Acronyms[1]*:* CBS, computer based system; NPP, nuclear power plant; PHWR, pressurized heavy water reactor; SDS, Shutdown system; FAV, fast acting valve; LPIS, liquid poison injection system; HMI, human machine interface; LC, logic condition; RTOS, real time operating system; FMEA, failure mode and effect analysis; PN, Petri net.

* Corresponding author.
E-mail addresses: lalit.rs.cse@iitbhu.ac.in (L.K. Singh), hrajput.rs.apm@iitbhu.ac.in (H. Rajput).

[1] The singular and plural of an acronym are always spelled the same.

http://dx.doi.org/10.1016/j.anucene.2016.02.002
0306-4549/

analyzed from safety perspectives. The set of failures for safety must be a subset of the failures for reliability.

Aljazzar et al. (2009) and Lee et al. (2011) tried to analyze system risk based on FMEA and other techniques but these techniques are limited to model the concurrency, parallelism and software failures, hence are not practically feasible for life critical systems.

Walter and Schneeweiss (2005) used PNs for risk and dependability analysis but he specifically put emphasis on reliability analysis.

Vernez et al. (2003) summarize perspective of using colored PNs for the risk analysis and an accident modeling but fails to cover a deep analysis from safety point-of-view.

Bobbio et al. (2001) and Khakzad et al. (2011) used Bayesian Networks, which can also accommodate uncertainty and has ability to model influences and complex interdependencies. But this approach has some known problems as a problematic modeling of temporal dependencies.

George and William (2005) converted the event trees using Bayesian networks to improve the approaches given in Bobbio et al. (2001), Khakzad et al. (2011) but the basic disadvantage of Bayesian Network is that large scenarios produce large networks, which require very large quantities of data.

From the above literature survey, the author must say that they have not found any work that would try to do the safety analysis of the complex systems that contain complex scenarios. Another fact is that almost no paper presents a case study with a large scenario – methods are usually shown only in simpler cases and those are too not specific to safety-critical systems in which safety aspect is at prime importance.

3. A case study: Shutdown system 2

There are two functionally independent and diverse SDS; each one is fully capable to shut down the reactor during any postulated accident condition. The following two different principles are used to provide functional independence:

1. Dropping all control rods in the moderator because of gravitational force.
2. Direct liquid poison injection tubes horizontally through the sides of the reactor.

The first and second mechanisms are achieved by SDS1 and SDS2 respectively.

SDS2 is a CBS system which senses the requirement for shutdown and opens the FAV to release high pressure helium to inject gadolinium nitrate solution into the moderator.

Apart from manual initiation, there are 9 parameters which can initiate SDS2 (Bereznani, 2005).

Fig. 1 is simplified diagram of SDS2-LPIS. A tank containing high pressure helium pressurizes the poison for rapid injection into the moderator. There are 4 FAV in between the helium tank and helium header which services the poison tanks. The FAV are air-to-close and spring-to-open, for ensuring their openings with high reliability, on demand. Each poison tank is connected to nozzle to inject it into the moderator.

4. Safety analysis

The failure of SDS2 will lead to the catastrophic disasters. The safety of the SDS2 is concerned with ensuring that a calamity does not occur, which has a potential to breach the safety that results in death, injury or damage of the objects. In case of safety critical system there are some failures which can never be tolerated at any cost, because of drastic consequences. For example, a flight may

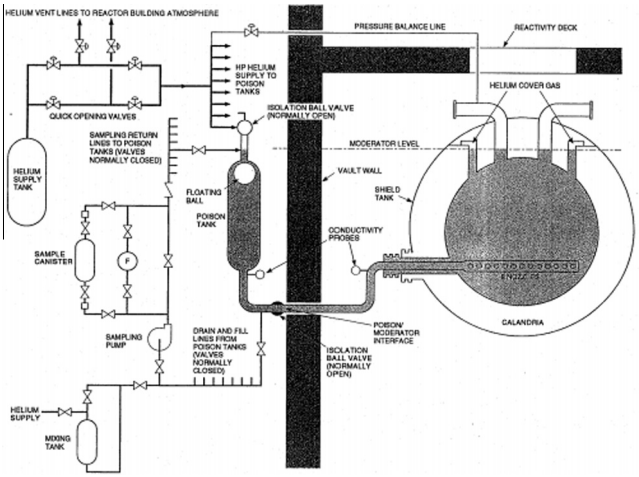

Fig. 1. Simplified diagram of Shutdown system-2 LPIS.

have to be canceled because of bad weather conditions. It means such systems have higher precedence of safety than reliability.

SDS2 system contains several components of different types that include sensors, logic, actuators, and a dedicated HMI to achieve its objective. These components can fail abruptly and hence the consequences to hardware and software failure, undesired environmental conditions must be modeled into the system design.

To do the safety analysis, an attempt must be made to identify all the system hazards and assess their consequences with respect to their severity. There could be some hazards which can lead to the risks that are acceptable to a certain limit while some hazards may lead to the risks that are unacceptable in any conditions. The technique to identify the system hazards in SDS2 is shown in this section.

There are two vent valves on each line of FAV, which are normally in open state to release the pressure in that line, if any, to avoid the spurious injection of poison. When any of the 9 parameters, discussed in Section 2, deviate from the normal limit, LC gets created. After creation of LC, both the vent valves are closed followed by opening of the FAV. This will pressurize the poison to get injected into the moderator. After poison injection, LC gets restored which closes the FAV followed by opening of vent valves.

To model SDS2 we use Time Petri net (Murata, 1989), as shown is Fig. 2a.

The places and transitions given in Fig. 2a are described in Table 1.

From the Table 1, it is to be noted that we keep redundant information to track the state of FAV because of its criticality for safety. FAV must be in open state to trip the reactor and must be in closed state for normal operating conditions. To identify the hazards, it is required to find out all the possible reachable states and which can be derived from the reachability graph. The reachability graph can be constructed from the PN model (Liao et al., 2013). Then we need to look into the behavioral states and failure states. The failure states could be low-risk or high-risk states. It is impossible to draw the complete reachability graph for a complex system due to its size. We can analyze the safety without drawing the full reachability graph by backtracking from the failure states to its originating state and apply some design techniques to ensure that from the originating state, that failure state cannot be reached.

From the reachability graph, shown in Fig. 2b, there are two threatening states S_3 and S_6, having triangular solid mark on the left upper corner which can cause disaster. In S_3, there are tokens in the places p_{LCh}, p_{revv}, p_{favcd}, p_{favc}. It means that LC exists but still