Nuclear Engineering and Technology 51 (2019) 1289–1296

Contents lists available at ScienceDirect

Nuclear Engineering and Technology

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

Original Article

Design of safety critical and control systems of Nuclear Power Plants using Petri nets

Pooja Singh [a, *], Lalit Kumar Singh [b]

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

ARTICLE INFO

Article history:
Received 14 December 2018
Received in revised form
11 February 2019
Accepted 22 February 2019
Available online 2 March 2019

Keywords:
Safety-critical systems
Performance analysis
Petri nets
Markov chain

ABSTRACT

Non-functional requirements plays a critical role in designing variety of applications domain ranges from safety-critical systems to simple gaming applications. Performance is one of the crucial non-functional requirement, especially in control and safety systems, that validates the design. System risk can be quantified as a product of probability of system failure and severity of its impact. In this paper, we devise a technique to do the performance analysis of safety critical and control systems and to estimate performance based risk factor. The technique elaborates Petri nets to estimate performability to ensure system dependability requirements. We illustrate the technique on a case study of Nuclear Power Plant system. The technique has been validated on 17 safety critical and control systems of Nuclear Power Plant.

1. Introduction

Validation of non functional requirement still poses a challenge in practice [1]. Relatively, a very less effort is devoted to address these requirements, considering an impractical approach of “Fix-it-Later”. Unfortunately, such practices must never be acceptable to safety critical and control systems. Today's such systems are complex in nature, being composed of heterogeneous hardware, firmware and software components. These systems measure a big set of process parameters to ensure the healthiness of the system, and data routes through such complex architectural environment to fulfill the intended requirements [1–3].

System designers have started taking benefits of recent advancements of high speed microprocessors, their performance, large memory and low cost to design distributed systems. Such systems use more number of microprocessors and heterogeneous components to meet the performance requirements of the system. However the reliability of such system actually depends on the effectiveness of the synchronization among such different and heterogeneous components.

Not meeting performance requirements, a safety system is declared as fail. Failure of any safety system can lead to catastrophic disasters and hence there is an utmost requirement to deal with. For this reason, researchers are continuously putting their tireless efforts to deal with performability [2] of safety-systems.

The objective of this paper is to introduce the concept of performance-based risk in case of safety critical systems. In order to explain our technique, we first explain the risk that can be involved in safety critical and control system. Then we consider a safety critical distributed system as a set of loosely or tightly coupled components working in synchronized fashion to meet the intended requirements. The performance analysis can be done by using deterministic or stochastic models. However, deterministic models do have several unpractical assumptions like task execution times, task arrival times, and level of synchronization. However, if these parameters are known, a very accurate estimate of performance can be predicted, which will be very useful for safety critical and control systems as generally such systems are real time in nature.

In case stochastic models for performance evaluation, task execution times, task arrival times, etc are usually determined by probability distribution functions. Moreover in case of safety critical and control systems, the synchronization among the tasks can also be modeled because such systems are very small in size and hence there state space is very small. The benefit of using stochastic models is prediction of performance at the early stages of system development life cycle when all the system characteristics are not known and well understood. This predicted value saves a

* Corresponding author.
E-mail addresses: poojasingh1615@gmail.com (P. Singh), lalit.rs.cse@iitbhu.ac.in (L.K. Singh).

https://doi.org/10.1016/j.net.2019.02.014
1738-5733/

significant effort and avoid delay in system development.

In this work, we use Petri nets (PN) to perform the performance analysis to validate the design of safety-critical systems of Nuclear Power Plant (NPP). For this we have taken a real case study of NPP, known as Shutdown System.

In section2, the related work along with their shortcomings is brought out. Section 3 provides an overview of basic terminologies. Section 4 discusses Bayesian Updating mechanism, on which the proposed technique based on. Section 5 discusses Shutdown System-2 (SDS-2), as a case study, in detail. Our approach for critical component identification of SDS2 is shown in section 6. In section 7, we apply our technique on the case study and show the experimental results along with its validation. Section 8 concludes this paper.

2. Related work

Lalit Kumar Singh et al. [1] proposed a methodology for verifying the design of Instrumentation & Control (I&C) systems, for which authors have address the reliability attribute. The proposed methodology is based on Petri nets and demonstrated step-by-step to ease its usage to theoreticians, researchers and practitioners. However, reliability is only one important attribute of dependability. There is a need to address more dependability attributes like safety, security, and performability.

Lalit Kumar Singh and Hitesh Rajput [3] proposed Petri nets based model for dependability analysis, in which reliability and safety attributes are considered. Some of the system properties are also verified like deadlocks, liveness, and boundedness. The proposed method is verified on several safety critical and control systems of NPP. However, being such systems real time in nature, there is a need to perform the performance analysis.

Singh, L. et al. [4] uses Petri net modelling technique to quantify the reliability of safety critical and control systems. The proposed technique transforms Petri net into Markov chain, where evaluation of transition probabilities among states is a challenge. Authors demonstrate a technique to compute transition probabilities on a safety critical system of NPP. However, nothing has been discussed on performance attribute.

Lalit et al. [2] proposed techniques to quantify performance of a test facility system of NPP. Authors have well described the case study of NPP along with system architecture, control flow, etc. The considered system is distributed in nature, where an embedded processing node is exchanging the data with display unit. Although the system is not safety critical in nature but embedded unit is real time in nature. However, token transferred execution time is not accurately modeled, and evaluated only on the assigned delay associated with timed transitions.

Lalit Singh and Hitesh Rajput [5] proposed an effective technique to perform safety analysis during the design phase of safety critical computer based system of NPP. Shutdown system 2 is taken as case study to demonstrate the technique, step by step. However, there are two issues in this method. It very well perform safety analysis qualitatively and also propose technique to address the shortcomings, in case safety analysis results any sort of threat. However, it would prove more beneficial, if a quantitative figure of safety could be derived to gain a significant amount on the system with respect to safety. Also, performance has not been considered in this paper.

V. Kumar et al. [6] proposed a technique for parameter estimation for quantitative dependability analysis of safety critical and control systems. Authors identified an important issue of assumed state transition probabilities, in state-space models that were used for quantification for several attributes of dependability. In this paper, the system requirements are captured using UML approach, which is transformed into Petri nets. Possible failures are identified in concurrence with system designer. Thereafter, quantitative safety assessment of the system was derived. The technique is demonstrated on Digital Feed Water Control System of NPP. However, performance analysis subject was remain silent, which needs to be addressed.

Vinay et al. [7] proposed a technique for transformation of deterministic models into state space models to perform safety analysis of safety critical systems. The method is validated on many safety critical systems of NPP and demonstrated on Reactor Core Isolation Cooling System. In this method, the probabilities of all the possible hazardous states are computed. However, nothing is dealt as far as performance of the system is concerned.

Raj et al. [8] have taken another interesting dependability attribute – security. This attribute is very important in two situations: (i) when the system is distributed in nature and (ii) when the system is connected with external network. Authors propose threat-driven framework to perform security analysis of safety critical systems. The proposed methodology is demonstrated step-by-step on Digital Feed Water Control System of NPP, in which Petri nets are used to model the system. Security threat may effect performance of the system, which have not been discussed in this paper.

Kumar P et al. [9] proposed an effective and optimized technique to predict the reliability of safety critical systems using Markov chain. In this technique, state explosion problem is attempted to be addressed by minimizing the number of possible states, devising some rules. The technique is validated on shut sown system of NPP. However, there are some limitations of Markov chain like transition rates among states follows exponential distribution, concurrent tasks cannot be modeled, etc. Further, performability issue is not addressed.

3. Background of Petri nets

A Petri net is a mathematical and visualization tool, which can model flow of information and control of any kind of system [10]. Mathematically, it is a 5-tuple $PN = (P, T, F, W, M_0)$ where P is a finite set of places, T is a finite set of transitions, F is a set of arcs such that $F \subseteq (P \times T) \cup (T \times P)$, W is a weight function that ranges from 1 to infinity, M_0 is the initial marking that ranges from 0 to infinity. Also, $P \cap T = \varnothing$ and $T \cap P = \varnothing$.

A place, denoted by circle, represents conditions and a transition, denoted by rectangular bar, represents events. A black solid circle, embeds in a place, is known as token and represents the holding of the condition of the place. The number of tokens in each place of PN at a given time represents state of the system and is called marking.

The state of the system depends on firing of transitions that depends on the marking. Firing of transitions causes movement of the tokens. The firings of transitions are governed by the following rules.

a) A transition to be fired, needs to be in enable state, which is possible only when each of its input places contains sufficient number of tokens. The sufficiency of the tokens is equal to weight on its input arc.
b) On firing of transitions, tokens are removed from each of its input places that depends on the weight of its input arc and tokens are deposited into each of its output arcs that also depends on the weight of its output arc.

The working of PN can be understood by Fig. 1. This scenario represents a reader writer process, where k tokens in place p_1 represents k processes which can read or write in place, p_3, which represents a shared memory. *k* processes can read concurrently but only one process can write at a time and during that period, no other processes can write or read.

From the figure, it can be seen that p_2 that represents ‘reading’, can contains maximum up to *k* token that represents processes if no token is in p_4. Also only one token can be in p_4 that represents ‘writing’ because all k tokens in p_3 will be removed through k weight arc when transition t_2 fires once.

PN are widely used to verify the control flow of the process or system and to know about the insights of the system. Several dependability properties can be analyzed such as liveness, boundedness, safeness, liveness, etc.

To study the performance of the system, traditional PN were extended to include the timing information [11] to the transitions. For example in Fig. 1, there could be a time delay in firing of t_2 to do some processing, even after t_2 is in enable condition. In such case writing process will be delayed further. The same situations may arise for rest of the transitions as well. These delays can be accumulated that can result in low performance of the system, even may fail to meet the system requirements.

In this paper, a method is devised to find the cycle time to process a task of the system. Several computational complexities are involved in analysis of system performance.

4. Case study: shut down system

4.1. Shut down system

The shut downs are designed to shut down the reactor for all plant conditions to prevent potentially hazardous situation from occurring. To ensure high reliability of shutdown operation, two completely independent and diversified systems are provided known as Shutdown System-1(SDS-1) and Shutdown System-2 (SDS-2).

SDS1 drops cadmium rods into the calendria to stop the fission reaction and hence reactor shuts down. SDS2 is designed to operate at higher ‘trip’ setpoints than SDS1 to ensure that reactore would shutdown if SDS-1 is unavailable or fails. SDS2 injects liquid poison into the calandria that eats the neutrons and hence sufficient amount of neutrons are not available to sustain the fission reaction. Both shutdown systems are designed to quickly insert sufficient negative reactivity into the calandria to reduce the reactor power out to a safe, subcritical, low level.

In order to meet shutdown requirements, there is a need of continuous monitoring of critical parameters that are known as trip parameters [12], which may be either absolute or conditional. Absolute trip parameters are valid at all the states of reactor power while conditional trip parameters are valid only when reactor power is greater or equal to 2% full power.

The simplified schematic diagram of SDS2 is shown in Fig. 2.

SDS2 is a computer based system, which contains several heterogeneous components such as sensors, digital input & output cards, relay output modules, data acquisition software, data processing software, graphical user interface. When it senses any trip parameter, it injects the liquid poison into the calandria. The working mechanism to inject the poison is follows.

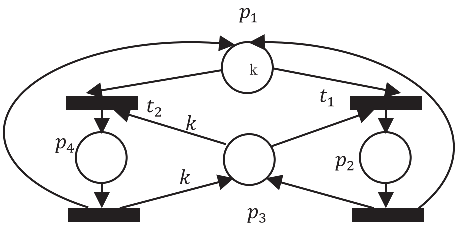

Fig. 1. Reading-writing process.

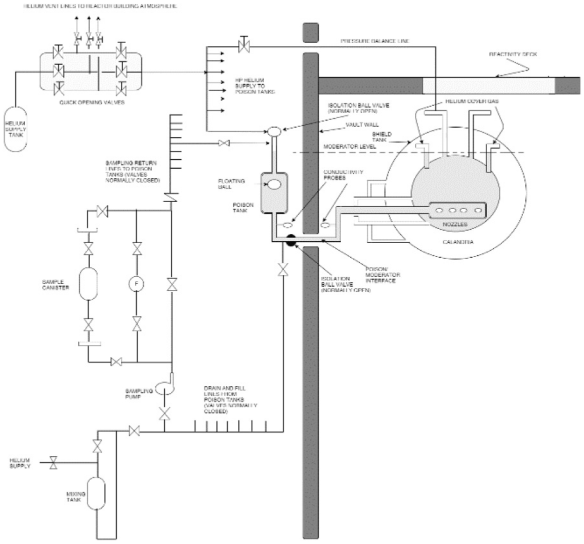
HELIUM VENT LINES TO REACTOR BUILDING ATMOSPHERE
PRESSURE BALANCE LINE
REACTIVITY DECK
HP HELIUM SUPPLY TO POISON TANKS
QUICK OPENING VALVES
HELIUM SUPPLY TANK
ISOLATION BALL VALVE (NORMALLY OPEN)
HELIUM COVER GAS
VAULT WALL
SHIELD TANK
MODERATOR LEVEL
SAMPLING RETURN LINES TO POISON TANKS (VALVES NORMALLY CLOSED)
FLOATING BALL
CONDUCTIVITY PROBES
POISON TANK
NOZZLES
CALANDRIA
SAMPLE CANISTER
F
ISOLATION BALL VALVE (NORMALLY OPEN)
POISON/ MODERATOR INTERFACE
SAMPLING PUMP
DRAIN AND FILL LINES FROM POISON TANKS (VALVES NORMALLY CLOSED)
HELIUM SUPPLY
MIXING TANK

Fig. 2. Simplified diagram of SDS2.

On deviation of trip parameter(s), all the 3 vent valves close and all the 6 Fast Acting Valves (FAV) open to connect Helium tank with the poison tank. Helium pressurizes floating ball in the poison tank to inject the poison into the calandria through a horizontal injection tube nozzle which spans calandria and is immersed in the moderator. After poison injection, floating ball sits at the bottom position in poison tank to prevent the release of high pressure helium to the calandria.

4.2. Architecture and critical functions of shutdown system

Fig. 3 represents the architecture of SDS-2. It is a 3-tier architecture. First tier represents field sensors, which sense the trip parameters and send their raw values in form of current, voltage or Resistance thermometers (RTD) signals to the embedded unit, known as Data Acquisition Module (DAM), placed in second tier. Two DAM are put into architectural design, one for redundancy purpose, so that if one DAM fails, other can take over to meet the reliability requirements. DAM processes raw data and convert into engineering data, for example voltage into pressure. DAM also performs critical functions like closing of vent valves, opening of FAV, reading current state of all equipments like valves, floating ball and sending it to Graphical User Interface, which is placed in third tier. Two Local Area Networks (LAN) are used for redundancy. Graphical User Interface (GUI) displays the process parameters and status of all equipments. Operator can perform control operations

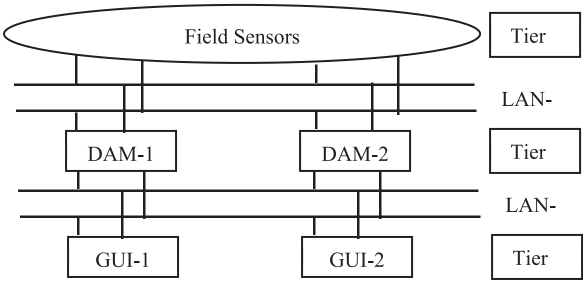

Fig. 3. Architecture of SDS-2.

also through GUI. GUI also has feature to log the history and generate alarms in case of unwanted scenario. Two LAN are used to ensure high reliability. Components of all the 3 tiers works in synchronized manner to achieve the reliability requirement of the overall SDS-2 system of 10^{-3} year/year. The DAM is an embedded system, software is developed on VxWorks, using C libraries, which is burned on Erasable programmable read only memory. The GUI software is developed in VC++ on Windows platform.

5. Performance quantification & validation

The success criteria of SDS-2 is that the rate of insertion of liquid poison from all the poison tanks shall be transferred into the calandria within 32 msecs. We devise a framework for performance quantification, given in Fig. 4.

5.1. Phase1: Petri net model creation

In this phase we create PN model of the system, for which performance quantification is to be computed. Several techniques are available [13] for constructing PN model from system specification. PN model for SDS-2 is shown in Fig. 5. The places and transitions are defined in Table 1. A token in place. p_{pd} represents that trip parameter(s) values are out of process range and there is a need to shut down the reactor immediately. The following activities in sequence are followed to shutdown the reactor.

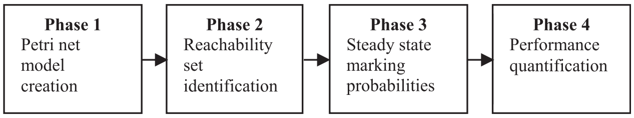
Phase 1
Petri net model creation
Phase 2
Reachability set identification
Phase 3
Steady state marking probabilities
Phase 4
Performance quantification

Fig. 4. Performance quantification framework.

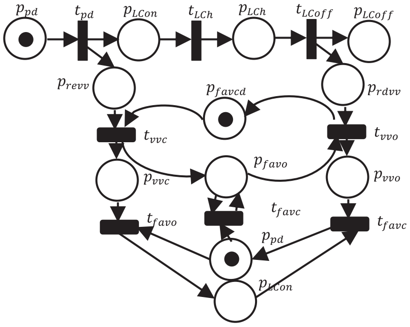

Fig. 5. PN model of SDS2.

1. As described in Fig. 3, DAM continuously reads process and neutronic parameters and compares them with the stored set points. If current parameters are out of the process range, a token gets deposited in p_{pd}.
2. When there is a token in p_{pd}, required logic conditions gets created, represented by a token in p_{Lcon} and a relay gets energized to close the vent valves, represented by a token in p_{revv}.
3. Thereafter logic condition remains on hold position, represented by a token p_{Lch}, till the poison injection process completes and vent valves closes, represented by a token in p_{vvc}.
4. On closing of all vent valves, all FAV become in open state, represented by a token in p_{favo}, to inject the poison into the calandria.

It is to be noted that place I is not performing any functional requirements but has been included to ensure the safety of the system by prioritising t_{favo} over t_{LCh} during race condition. The constructed PN contains sequential transitions such as (t_{LCh},t_{Lcoff}), parallel transitions such as (t_{LCh},t_{vvc}), forking such as t_{pd}, joining such as t_{LCh}. It can also be analyzed for critical attributes such as liveness, deadlocks, boundedness, safeness as described in section 3. The PN model also allows computation of steady state probabilities of different markings.

5.2. Phase2: reachability set identification

The constructed PN model gives the boundary conditions of the system i.e. it can identify the number of possible states of the system throughout its operational life [5]. As token movement takes place, marking changes. Each marking represents a state of the system. The total possible markings represents the total possible states, a system can undergo. For the SDS-2 PN, shown in Fig. 5, the 18 possible states are possible as given in equation (1).

Table 1
SDS-2 PN model places and transitions.

TYPE	NOTATION	DESCRIPTION
p_{pd}	Place	Trip parameters deviates
p_{Lcon}	Place	Logic condition creates
p_{LCh}	Place	Logic condition holds
p_{LCoff}	Place	Logic condition off
p_{revv}	Place	Vent valve closing relay in energized state
p_{favcd}	Place	Redundant place for FAV close state
p_{vvc}	Place	Vent valve in closed state
p_{favc}	Place	FAV is in closed state
p_{favo}	Place	FAV is in open state
p_{rdvv}	Place	Vent valve opening relay in de-energized state
p_{favod}	Place	Redundant place for FAV open state
p_{vvo}	Place	Vent valve in open state
I	Place	Condition of prioritization of t_{favo} over t_{LCh}
t_{pd}	Transition	Signal sent to create logic condition & energize vent valve close relay
t_{LCh}	Transition	Signals to hold Logic condition in ON state
t_{LCoff}	Transition	Signals to OFF Logic condition
t_{vvc}	Transition	Sends signal to close vent valves
t_{favo}	Transition	Sends signal to open all FAV
t_{vvo}	Transition	Sends signal to open vent valves
t_{favc}	Transition	Sends signal to close all FAV
t_{sfav}	Transition	Retrying sending signal to open FAV

Markov chain derived from these possible states is given in Fig. 6. It is to be noted that for different initial markings, we obtain a different Markov Chain and Markov chain size increases with number of tokens in PN model.

5.3. Phase3: steady state marking probabilities

The derived Markov chain from constructed PN model of SDS-2 is ergodic, which can be solved for steady state "marking probabilities.

$$\left[\begin{array}{cccccccccccccc} States & p_{pd} & p_{Lcon} & p_{LCh} & p_{LCoff} & p_{revv} & p_{favcd} & p_{vvc} & p_{favc} & p_{favo} & p_{rdvv} & p_{favod} & p_{vvo} & I \\ M_0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ M_1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ M_2 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ M_3 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ M_4 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ M_5 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ M_6 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ M_7 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ M_8 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ M_9 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ M_{10} & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ M_{11} & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ M_{12} & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ M_{13} & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ M_{14} & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ M_{15} & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ M_{16} & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ M_{17} & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \end{array}\right] \tag{1}$$

Since the arriving packets are queuing up with time slot of t, the next transmission is possible only during $t+1$. Let us assume that queue size is K. When queue becomes full, upcoming packets will drop.

Let every link i has queue length $\overrightarrow{Q_t} = \{Q_{i,t}\}_{i\in D}$ and number of packets arrived to link i during t is represented by $A_{i,t}$. This is Poisson stationary process with mean $\lambda_i \Delta t$. Then queuing process evolves as given in equation (2)

$$Q_{i,t+1} = \min\left[K, \max\left[0, Q_{i,t} - \frac{r_{i,t}\Delta T}{B}\right] + A_{i,t}\right] \tag{2}$$

Therefore, Signal to Interference and Noise Ratio (SNR) depends on other links in system. Assuming, $\overrightarrow{\Theta_t} = \{\Theta_{i,t}\}_{i\in D}$ represents status of queue, whether occupied by the packets, of link i during starting duration t. Here $\Theta_{i,t} = 1\ \forall Q_{i,t} > 0$. There are 2^D possible

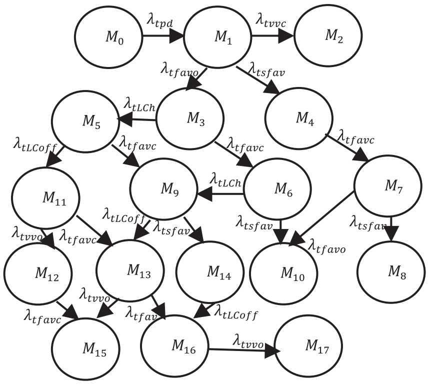

Fig. 6. Equivalent MC of Fig. 5.

values for $\overrightarrow{\Theta_t}$. Let subset of other links excluding i with non zero length of the queue is represented by $\xi_{i,t} = \{j\in D\{i\} : \Theta_{j,t} = 1\}$.

$$\therefore SNR_{i,t} = \frac{P_i G_{ii,t}}{N_i + \sum_{j\in D\{i\}} P_i G_{ji,t}\Theta_{j,t}}$$

where $\frac{P_i G_{ii,t}}{N_i}$ is SNR of link i and N_i is noise on i. Denoting $\gamma_{ii,t} = \frac{P_i G_{ii,t}}{N_i}$,

$$SNR_{i,t} = \frac{\gamma_{ii,t}}{1 + \sum_{j\in D\{i\}} \gamma_{ji,t}\Theta_{j,t}} \tag{3}$$

Let the transition probability from state $(\overrightarrow{l}, \overrightarrow{k})$ to $(\overrightarrow{n}, \overrightarrow{h})$ of Markov chain is $p^{(\overrightarrow{n},\overrightarrow{h})}_{(\overrightarrow{l},\overrightarrow{k})}$, where $\overrightarrow{l} := \{l_i\}_{i\in D}$, $\overrightarrow{k} := \{k_i\}_{i\in D}$, $\overrightarrow{n} := \{n_i\}_{i\in D}$, $\overrightarrow{h} := \{h_i\}_{i\in D}$. The $p^{(\overrightarrow{n},\overrightarrow{h})}_{(\overrightarrow{l},\overrightarrow{k})}$ can be decomposed as:

$$\begin{aligned} p^{(\overrightarrow{n},\overrightarrow{h})}_{(\overrightarrow{l},\overrightarrow{k})} &= P\left\{\overrightarrow{Q_{t+1}} = \overrightarrow{h} | \overrightarrow{H_t} = \overrightarrow{l}, \overrightarrow{Q_t} = \overrightarrow{k}\right\} \\ &\times P\left\{\overrightarrow{H_{t+1}} = \overrightarrow{n} | \overrightarrow{H_t} = \overrightarrow{l}, \overrightarrow{Q_t} = \overrightarrow{k}, \overrightarrow{Q_{t+1}} = \overrightarrow{h}\right\} \\ &= p^{\overrightarrow{h}}_{(\overrightarrow{l},\overrightarrow{k})} p^{\overrightarrow{n}}_{(\overrightarrow{l},\overrightarrow{k},\overrightarrow{h})} \end{aligned} \tag{4}$$

Here $p^{\overrightarrow{h}}_{(\overrightarrow{l},\overrightarrow{k})}$ represents transition probability of the queue state from $\overrightarrow{Q_t} = \overrightarrow{k}$ to $\overrightarrow{Q_{t+1}} = \overrightarrow{k}$, given the channel state $\overrightarrow{H_t} = \overrightarrow{l}$ and $p^{\overrightarrow{n}}_{(\overrightarrow{l},\overrightarrow{k},\overrightarrow{h})}$ represents transition probability of the channel state from $\overrightarrow{H_t} = \overrightarrow{l}$ to $\overrightarrow{H_{t+1}} = \overrightarrow{n}$, given queue states $\overrightarrow{Q_t} = \overrightarrow{k}$ and $\overrightarrow{Q_{t+1}} = \overrightarrow{h}$.

From (1), we can have:

$$p^{h_i}_{l_i,k_i} = P\{Q_{i,t+1} = h_i | H_{i,t} = l_i, Q_{i,t} = k_i\}, Vr_{i,t} = R_{l_i}$$

$$= \begin{cases} P(A_{i,t} = h_i - k_i + \eta_i), k_i > \eta_i, h_i \neq K \\ P(A_{i,t} = h_i), k_i \leq \eta_i, h_i \neq K \\ P(A_{i,t} \geq K - k_i + \eta_i), k_i > \eta_i, h_i = K \\ P(A_{i,t} \geq K), k_i \leq \eta_i, h_i = K \end{cases} \tag{5}$$

From Poisson assumptions. $\eta_i = \frac{R_{l_i} \varDelta T}{B}, P(A_{i,t} = a) = \frac{(\lambda_i \varDelta T)^a}{a!} e^{-\lambda_i \varDelta T}$
Also, $p^{h_i}_{l_i,k_i}$ depends only on the server through which it is connected and queue states.

$$\therefore p^{\overrightarrow{h}}_{(\overrightarrow{l}, \overrightarrow{k})} = \prod_{i=1}^{D} p^{h_i}_{l_i,k_i} \tag{6}$$

Theorem 1. *$\forall\, \overrightarrow{Q_t} =, \overrightarrow{k} \in S_{Q_i} \times S^{\bar{i}}_{\theta_v}$ and $\forall\, \overrightarrow{Q_{t+1}} = \overrightarrow{h} \in S_{Q_i} \times S^{\bar{i}}_{\theta_\omega}$, the values of $p^{n_i}_{l_i, \overrightarrow{k}, \overrightarrow{h}}$ are identical and can be represented as $p^{n_i}_{l_i,\theta_v,\theta_\omega}$.*

Proof *- the proof of this Theorem is straightforward from* equation (3). *We can write*

$$p^{n_i}_{(l_i,\theta_v,\theta_\omega)} = \frac{p^{(l_i,n_i)}_{(\theta_v,\theta_\omega)}}{p^{l_i}_{\theta_v}} \tag{7}$$

We can create finite state Markov model with respect to SNR for each link, which also considers the noise of Electromagnetic Interference. The transition in states of SNR based finite state Markov model can be because of variations of SNR elements in SNR vector $\{\gamma_{ji}\}_{j\in\{i\}\cup\xi_t}$. We make an practical assumption that state transition can occur only in adjacent states.

$$\therefore p^{(l_i,n_i)}_{(\theta_v,\theta_\omega)} = 0, \vee |n_i - l_i| > 1$$

Let $LCR(\mathrm{X}_{l_i})$ is rate of level crossing, defined as number of times per second SNR passes downward across X_{l_i} and $LCR(\mathrm{X}_{l_i})\Delta T$ is probability that SNR passes downward across X_{l_i} in a time slot interval ΔT.

$$\therefore LCR(\mathrm{X}_{l_i})\Delta T < 1$$

For $\Delta t \to 0$, $LCR(\mathrm{X}_{l_i})\Delta t$ will represent the probability that SNR passes downward across X_{l_i} interval small interval of Δt.
Therefore,

$$p^{(l_i,l_i+1)}_{(\theta_v,\theta_\omega)} \approx LCR(\mathrm{X}_{l_i})\Delta T, p^{(l_i,l_i-1)}_{(\theta_v,\theta_\omega)} \approx LCR(\mathrm{X}_{l_i-1})\Delta T \tag{8}$$

Let E_j is defined as event that SNR passes downward/upward across a threshold, then we can write:

$$LCR(\mathrm{X}_{l_i})\Delta t = \sum_{j\in D} P(E_j) \tag{9}$$

From equations (8) and (9), *we get:*

$$p^{(l_i,l_i+1)}_{(\theta_v,\theta_v)} \approx \sum_{j=1}^{D} \int \ldots \int_0^\infty N_j\left(\Gamma_{j,l_i}\right)\Delta T \prod_{k\in\{i\}\cup\xi_t\{j\}} f(\gamma_{ki})d\gamma_{ki}$$

and

$$p^{(l_i,l_i-1)}_{(\theta_v,\theta_v)} \approx \sum_{j=1}^{D} \int \ldots \int_0^\infty N_j\left(\Gamma_{j,l_i-1}\right)\Delta T \prod_{k\in\{i\}\cup\xi_t\{j\}} f(\gamma_{ki})d\gamma_{ki} \tag{10}$$

where $N_j(\Gamma)$ is the level crossing rate of SNR.
We can decompose $p^{(l_i,n_i)}_{(\theta_v,\theta_\omega)}$ as follows.

$$p^{(l_i,n_i)}_{(\theta_v,\theta_\omega)} = p^{(l_i,l_i+1)}_{(\theta_v,\theta_v)} \times \widehat{p}^{n_i}_{(l_i+1,\theta_v,\theta_\omega)} + p^{(l_i,l_i-1)}_{(\theta_v,\theta_v)} \times \widehat{p}^{n_i}_{(l_i-1,\theta_v,\theta_\omega)} + p^{(l_i,l_i)}_{(\theta_v,\theta_v)} \times \widehat{p}^{n_i}_{(l_i,\theta_v,\theta_\omega)} \tag{11}$$

And we can have:

$$p^{\overrightarrow{n}}_{(\overrightarrow{l}, \overrightarrow{k}, \overrightarrow{h})} = \prod_{i=1}^{D} p^{n_i}_{(l_i, \overrightarrow{k}, \overrightarrow{h})} \tag{12}$$

Theorem 2. *The stationary distribution of the Markov Chain $(\overrightarrow{H_t}, \overrightarrow{Q_t})$ exists; π is unique, and $\pi > 0$.*
Proof*- In order to prove the Theorem, we need to propose lemmas.*

Lemma 1. *The Markov chain $(\overrightarrow{H_t}, \overrightarrow{Q_t})$ is irreducible, if $K < R_L \varDelta T$.*
Proof of Lemma 1: Lemma 1 *can be proved by proving that there exists a multi-transition path - $(\overrightarrow{l}, \overrightarrow{k}) \to (\overrightarrow{n}, \overrightarrow{h})$ with probability greater than 0 for each transition from state $(\overrightarrow{l}, \overrightarrow{k})$ to $(\overrightarrow{n}, \overrightarrow{h})$. The proof is as follows.*

a) ***Case 1:*** $(\boldsymbol{\overrightarrow{l}}, \boldsymbol{\overrightarrow{k}}) \to (\boldsymbol{\overrightarrow{l^*}}, \boldsymbol{\overrightarrow{k}}), \forall \boldsymbol{\overrightarrow{l^*}} = \{\boldsymbol{\overrightarrow{l_i^*}} \in \{1, \ldots, \boldsymbol{L}\} : \boldsymbol{R_{\overrightarrow{l_i^*}} \varDelta T \geq k_i}\}_{\boldsymbol{i\in D}}$

We start by proving $p^{\{l_i+1\}_{i\in D}}_{(\overrightarrow{l}, \overrightarrow{k}, \overrightarrow{k})} > 0, p^{\{l_i-1\}_{i\in D}}_{(\overrightarrow{l}, \overrightarrow{k}, \overrightarrow{k})} > 0$ and $p^{\overrightarrow{l}}_{(\overrightarrow{l}, \overrightarrow{k}, \overrightarrow{k})} > 0$. From equation (12), *there is a need to prove $p^{(l_i+1)}_{(l_i, \overrightarrow{k}, \overrightarrow{k})} > 0, p^{(l_i-1)}_{(l_i, \overrightarrow{k}, \overrightarrow{k})} > 0$ and $p^{l_i}_{(l_i, \overrightarrow{k}, \overrightarrow{k})} > 0$, which is equivalent to prove that $p^{(l_i+1)}_{(l_i,\theta_v,\theta_v)} > 0, p^{(l_i-1)}_{(l_i,\theta_v,\theta_v)} > 0$ and $p^{l_i}_{(l_i,\theta_v,\theta_v)} > 0$, according to* Theorem 1. *Here $\overrightarrow{k} \in S_{Q_i} \times S^{\bar{i}}_{\theta_v}$. This statement has strong validation from* equation (7), *as $p^{l_i}_{\theta_v} > 0$. Also from* equation (10), *we get $p^{(l_i,l_i+1)}_{(\theta_v,\theta_v)} > 0, p^{(l_i,l_i-1)}_{(\theta_v,\theta_v)} > 0$ and $p^{(l_i,l_i)}_{(\theta_v,\theta_v)} > 0$.*

Next, we need to prove that $p^{\overrightarrow{k}}_{\overrightarrow{l}, \overrightarrow{k}} > 0$. Since $A_{i,t} = k_i - \max[0, k_i - R_{l_i} \varDelta T] \geq 0$, from equations (5) and (5) *we have $p^{k_i}_{l_i,k_i} > 0$ and $p^{\overrightarrow{k}}_{(\overrightarrow{l}, \overrightarrow{k})} > 0$.*
Therefore from equation (4),

$$p^{\left(\{l_i+1\}_{i\in D}, \overrightarrow{k}\right)}_{(\overrightarrow{l}, \overrightarrow{k})} > 0, p^{\left(\{l_i-1\}_{i\in D}, \overrightarrow{k}\right)}_{(\overrightarrow{l}, \overrightarrow{k})} > 0,$$

$$p^{(\overrightarrow{l}, \overrightarrow{k})}_{(\overrightarrow{l}, \overrightarrow{k})} > 0$$

Also, there exists a multi-transition path from $(\overrightarrow{l}, \overrightarrow{k})to(\overrightarrow{l^*}, \overrightarrow{k})$ as following two ways, with non zero probability of each transition.

$$\left(\overrightarrow{l}, \overrightarrow{k}\right) \to \left(\{l_i + 1\}_{i\in D}, \overrightarrow{k}\right) \to \ldots \to \left(\left\{l_i^* - 1\right\}_{i\in D}, \overrightarrow{k}\right) \to \left(\overrightarrow{l^*}, \overrightarrow{k}\right) \tag{13}$$

$$\left(\overrightarrow{l}, \overrightarrow{k}\right) \to \left(\{l_i - 1\}_{i\in D}, \overrightarrow{k}\right) \to \ldots \to \left(\left\{l_i^* + 1\right\}_{i\in D}, \overrightarrow{k}\right) \to \left(\overrightarrow{l^*}, \overrightarrow{k}\right) \tag{14}$$

b) ***Case 2:*** $(\boldsymbol{\overrightarrow{l^*}}, \boldsymbol{\overrightarrow{k}}) \to (\boldsymbol{\overrightarrow{l^*}}, \boldsymbol{\overrightarrow{h}})$

In this case, we prove that $p^{\vec{h}}_{\vec{l^*},\vec{k}} > 0$.

$\because A_{i,t} = h_i - \max[0, k_i - R_{l_i}\Delta T] \geq 0$, *given* $R_{l_i^*}\Delta T > k_i$

From equations (5) and (6),

$$p^{h_i}_{l_i^*,k_i} > 0, p^{\vec{h}}_{(\vec{l^*}\,\vec{k})} > 0$$

Next we prove that $p^{\vec{l^*}}_{\vec{l^*},\vec{k},\vec{h}} > 0$. *It is clear from* equation (12), *we only need to prove that* $p^{l_i^*}_{l_i^*,\vec{k},\vec{h}} > 0$. *It is equivalent to prove that* $p^{l_i^*}_{(l_i^*,\theta_v,\theta_\omega)} > 0$, *where* $\vec{k} \in S_{Q_i} \times S^{\bar{i}}_{\theta_v}$ *and* $\vec{h} \in S_{Q_i} \times S^{\bar{i}}_{\theta_\omega}$. *From* equation (7), *and* $\pi_{l_i^*|\theta_v} > 0$, *we only have to prove that* $p^{l_i^*,l_i^*}_{(\theta_v,\theta_\omega)} > 0$. *We have already proved that* $p^{(l_i^*,l_i^*+1)}_{(\theta_v,\theta_v)} > 0, p^{(l_i^*,l_i^*-1)}_{(\theta_v,\theta_v)} > 0$ *and* $p^{(l_i^*,l_i^*)}_{(\theta_v,\theta_v)} > 0$. *Therefore we only need to prove*:

$$\exists\ \widehat{p}^{l_i}_{(l_i^*+1,\theta_v,\theta_\omega)} \cup \widehat{p}^{l_i^*}_{(l_i^*-1,\theta_v,\theta_\omega)} \cup \widehat{p}^{l_i^*}_{(l_i^*,\theta_v,\theta_\omega)} > 0$$

$\widehat{p}^{l_i^*}_{(l_i^*,\theta_v,\theta_\omega)}$ *is always greater than zero, irrespective of* θ_v *and* θ_ω *relationship.*

$$\therefore p^{\vec{l^*}}_{\vec{l^*},\vec{k},\vec{h}} > 0$$

Therefore, from equation (4),

$$\left(\vec{l^*}, \vec{k}\right) \rightarrow \left(\vec{l^*}, \vec{h}\right) > 0 \tag{15}$$

c) ***Case 3:*** $(\vec{\boldsymbol{l}^*}, \vec{\boldsymbol{h}}) \rightarrow (\vec{\boldsymbol{n}}, \vec{\boldsymbol{h}})$

The proof is similar to case 1.

From equations (2)–(4), *we can prove the following*:

$$\exists \left(\vec{l}, \vec{k}\right) \rightarrow \left(\vec{l^*}, \vec{k}\right) \rightarrow \left(\vec{l^*}, \vec{h}\right) \rightarrow \left(\vec{n}, \vec{h}\right)$$

Where

$$R_{l_i^*}\Delta T \geq k_i$$

$\because K \leq R_L \Delta T, \exists l_i^*$ *that satisfies this condition.*

The stationary or steady state distribution of the ergodic process can be uniquely determined by the following balance equations.

$$\vec{\pi} = [\pi(M_0), \pi(M_1), \ldots, \pi(M_{17})]$$

$$\vec{\pi} = \vec{\pi} P, \sum_{i \in M} \pi(i) = 1$$

Where $\pi_i(t)$ be the probability that the system is in state i at time t and it is considered that system executes for a very long time $t \rightarrow \infty$. *The steady state probabilities, for each marking is given in* Table 2.

5.4. *Phase4: performance quantification*

In our case the reachability set of PN is finite. However, for several systems there may be infinite reachability set, for example in communication systems, acknowledgement may take longer time. Analysing such systems is complicated due to infinite reachability set. However, researchers have proposed many techniques to solve such cases empirically and analytically. For example, infinite reachability set can be truncated [9], ensuring integrity of system specification. The resultant reachability set will be finite which can give approximated results. Empirically, one could analyse the reachability set sincerely to eliminate practically possible states [3,5]. Also reachability set can be reduced by eliminating states which are not safe or by prioritising the transitions, which are in race condition. However, a care should be taken that such modifications destroys the Markovian property.

To quantify the performance, we consider two calculations.

Table 2
Steady state marking probabilities of SDS2 PN Places

$P[M_0] = 3.311 \times 10^{-3}$	$P[M_1] = 8.2 \times 10^{-4}$	$P[M_2] = 3.311 \times 10^{-3}$
$P[M_3] = 3.311 \times 10^{-3}$	$P[M_4] = 1.655 \times 10^{-3}$	$P[M_5] = 0.163$
$P[M_6] = 3.344 \times 10^{-5}$	$P[M_7] = 1.66 \times 10^{-5}$	$P[M_8] = 3.311 \times 10^{-3}$
$P[M_9] = 8.34 \times 10^{-10}$	$P[M_{10}] = 8.32 \times 10^{-6}$	$P[M_{11}] = 0.0815$
$P[M_{12}] = 0.0815$	$P[M_{13}] = 0.164$	$P[M_{14}] = 4.17 \times 10^{-10}$
$P[M_{15}] = 2.455 \times 10^{-3}$	$P[M_{16}] = 0.082$	$P[M_{17}] = 8.2 \times 10^{-4}$

5.4.1. *Communication channel to capture trip parameters values*

Token deposition in p_{pd}, on deviation of trip parameters process range. During communication CRC calculation is being done. The transitions reading value, sending, receive acknowledgement have exponentially distributed execution times. There is a fixed timeout time, in which the message needs to be send again. Therefore timeout transition does not follow exponential distribution. It is to be noted that random variable time that denotes timeout has Erlangian probability density function. The values communicate to system, represented by a token in p_{pd} at Poisson rate λ. Therefore, the throughput of the communication system is $\lambda(1-\rho)$. In our case, the baud rate of communication network is 9600 with 5% of error probability and packet size is 128 bytes. Then performance measures for packet containing the values of trip parameters are: (i) for sending packet, the rate is 8.325 firings/second (ii) for lost packets (data packet of acknowledgement), the rate is 4.01 firings/second (iii) for CRC check, the rate is 80.23 firings/second (iv) for timeout condition, the rate is 1.00 firings/second.

5.4.2. *Poison injection process*

Once token gets deposited in p_{pd}, poison injection takes place as per Petri net model, shown in Fig. 5. We assume that the service times are independent and exponentially distributed with parameter $\mu_i, i = 0, \ldots, 7$. The objective function in this process is total throughput $\sum_{i=0}^{7}\lambda_0$. The numerical results with 95% of confidence interval are presented in Table 3.

The average delay may be computed when the system is busy or packet acknowledgement is lost or on-hold. In such case, delay can be computed using Little's result as $N = \lambda T$, where λ is throughput. For the values used above, the average delay in the entire process of injection of poison on trip parameter deviation is 0.2576 s.

The results have been validated on 17 systems of Nuclear Power Plants, out of which 9 systems are control systems, 6 systems are

Table 3
Throughput (firings/sec) of SDS2 transitions.

$\lambda[t_{pd}] = 4.22$	$\lambda[t_{LCh}] = 5.21$	$\lambda[t_{LCoff}] = 4.81$	$\lambda[t_{vvc}] = 3.89$
$\lambda[t_{vvo}] = 5.20$	$\lambda[t_{favo}] = 4.77$	$\lambda[t_{sfa}] = 4.98$	$\lambda[t_{favc}] = 3.98$

safety critical systems and 2 systems are monitoring systems.

6. Conclusion

The work focuses on a technique for evaluating the performance of the system using Petri nets, to validate the design, especially of safety critical and control systems. The analysis was based on Petri nets, Markov chain and proposing new related theorems along with their proofs. The proposed technique has a potential to address the challenges and limitations of the existing techniques that are discussed in section 2. The technique is applied on 17 different safety critical, control and monitoring systems of NPP and is demonstrated on a SDS2. The case study is well described along with its functions, architecture and boundary conditions. Our experimental results strongly validate our approach.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.net.2019.02.014.

References

[1] Lalit Kumar Singh, Gopika Vinod, A.K. Tripathi, Design verification of instrumentation and control systems of NPP, IEEE Trans. Nucl. Sci. 61 (April 2014) 921–930.

[2] L.K. Singh, G. Vinod, A.K. Tripathi, Modeling and prediction of performability of safety critical computer based systems using Petri nets, in: 2012 IEEE 23rd International Symposium on Software Reliability Engineering Workshops, Dallas, TX, 2012, pp. 85–94.

[3] Lalit Kumar Singh, Hitesh Rajput, Dependability analysis of safety critical real-time systems by using Petri nets, IEEE Trans. Control Syst. Technol. (March 2017) 1–12.

[4] L. Singh, H. Rajput, G. Vinod, A.K. Tripathi, Computing transition probability in Markov chain for early prediction of software reliability, Qual. Reliab. Eng. Int. (2015), https://doi.org/10.1002/qre.1793.

[5] Lalit Singh, Hitesh Rajput, Ensuring safety in design of safety critical computer based systems, in: Annals of Nuclear Energy, vol. 92, Elsevier, June 2016, pp. 289–294.

[6] V. Kumar, L.K. Singh, P. Singh, K.V. Singh, A.K. Maurya, A.K. Tripathi, Parameter estimation for quantitative dependability analysis of safety-critical and control systems of NPP, IEEE Trans. Nucl. Sci. 65 (5) (May 2018) 1080–1090.

[7] Vinay Kumar, Lalit Singh, A.K. Tripathi, Transformation of deterministic models into state space models for safety analysis of safety critical systems: a case study of NPP, in: Annals of Nuclear Energy, vol. 105, Elsevier, July 2017, pp. 133–143.

[8] Raj Kamal, Lalit Singh, Babita Pandey, Security analysis of safety critical and control systems: a case study of nuclear power plant system, Nucl. Technol. 197 (3) (2017) 296–307, https://doi.org/10.1080/00295450.2016.1273702.

[9] P. Kumar, L.K. Singh, C. Kumar, An optimized technique for reliability analysis of safety-critical systems: a case study of nuclear power plant, Qual. Reliab. Eng. Int. (2018) 1–9. https://doi.org/10.1002/qre.2340.

[10] T. Murata, Petri nets: properties analysis and applications, Proc. IEEE 77 (1989) 541–580.

[11] C. Ramchandani, Analysis of Asynchronous Concurrent Systems by Petri Nets, M.I.T., Cambridge, MA, 1974. Project MAC, TR-120.

[12] CANDU 6 Program Team, CANDU 6 Tech. Summary, May 2005.

[13] Lalit Singh, Gopika Vinod, A.K. Tripathi, An approach for Parameter estimation in Markov model of software reliability for early prediction: a case study, IET Softw. 9 (3) (June 2015) 65–75.