http://dx.doi.org/10.5516/NET.03.2012.707

RELIABILITY ANALYSIS OF DIGITAL SYSTEMS IN A PROBABILISTIC RISK ANALYSIS FOR NUCLEAR POWER PLANTS

STEFAN AUTHÉN[1] and JAN-ERIK HOLMBERG[2*]
[1]Risk Pilot
Parmmätargatan 7, SE-11224 Stockholm, Sweden
[2]VTT
P.O.Box 1000, FI-02044 VTT, Finland
[*]Corresponding author. E-mail : jan-erik.holmberg@vtt.fi

Received May 05, 2012

To assess the risk of nuclear power plant operation and to determine the risk impact of digital systems, there is a need to quantitatively assess the reliability of the digital systems in a justifiable manner. The Probabilistic Risk Analysis (PRA) is a tool which can reveal shortcomings of the NPP design in general and PRA analysts have not had sufficient guiding principles in modelling particular digital components malfunctions.

Currently digital I&C systems are mostly analyzed simply and conventionally in PRA, based on failure mode and effects analysis and fault tree modelling. More dynamic approaches are still in the trial stage and can be difficult to apply in full scale PRA-models. As basic events CPU failures, application software failures and common cause failures (CCF) between identical components are modelled.The primary goal is to model dependencies. However, it is not clear which failure modes or system parts CCF:s should be postulated for. A clear distinction can be made between the treatment of protection and control systems. There is a general consensus that protection systems shall be included in PRA, while control systems can be treated in a limited manner.

OECD/NEA CSNI Working Group on Risk Assessment (WGRisk) has set up a task group, called DIGREL, to develop taxonomy of failure modes of digital components for the purposes of PRA. The taxonomy is aimed to be the basis of future modelling and quantification efforts. It will also help to define a structure for data collection and to review PRA studies.

KEYWORDS : Nuclear I&C, Digital I&C, Software, Probabilistic Risk Analysis, Probabilistic Safety Assessment, Reliability, PRA, PSA

1. INTRODUCTION

Digital protection and control systems are appearing as upgrades in older nuclear power plants (NPPs) and are commonplace in new NPPs. To assess the risk of NPP operation and to determine the risk impact of digital system upgrades on NPPs, quantifiable reliability models are needed along with data for digital systems that are compatible with existing probabilistic risk analyses (PRA) or probabilistic safety assessments (PSA). Due to the many unique attributes of these systems, several challenges exist in systems analysis, modeling and in data collection [1–3].

Currently there is no consensus on reliability analysis approaches. Traditional methods (event tree-fault tree approach) have clear limitations, but more dynamic approaches are still in the trial stage and can be difficult to apply in full scale PRA-models. Also the number of PRAs worldwide including reliability models of digital I&C systems, for instance, of the reactor protection system (RPS), are very few. Hence it is not possible at this stage to to identify a sound state-of-the-art regarding the reliability analysis approaches.

In current PRAs, distributed control systems are typically analysed and modelled rather simply. In many cases, the starting point for modelling is a reliability analysis made by the vendor, though incorporating the vendor's analysis in a PRA is not a straightforward task. Reviewing and evaluating the vendor's analysis can also be problematic, since the documentation sometimes lacks in transparency.

Digital control systems can be analyzed on several abstract levels, which raises additional questions, such as: which level of detail should be used, which failure modes should be considered, how to consider software failures, how to consider CCF, which dependencies should be considered, and how to account for human errors. The selection of plausible failure data, including common cause failure data for hardware and software is an open issue.

This paper presents an overview of the state-of-the-art of methods used in PRAs for nuclear power plants as well as interim results from the ongoing international project to develop guidelines for reliability analysis of digital I&C, i.e, OECD/NEA Working Group Risk Task Group "Development of the best practice guidelines on failure modes taxonomy for reliability assessment of digital I&C systems for PSA", called hereafter DIGREL.

2. SAFETY I&C SYSTEMS IN NUCLEAR POWER PLANT

In the last decades a variety of different safety-related digital I&C systems have been developed and implemented in nuclear installations and facilities around the world. Digital I&C architectures are deployed in several reactors worldwide, not only in turbine automation but also in safety automation, such as Chooz B (France), Sizewell B (United Kingdom), Ringhals-1 and -2 (Sweden), Temelin-1 and -2 (Czech Republic), and Tianwan (China). Also new designs such as the EPR developed by AREVA, the APWR by Mitsubishi Heavy Industries, Ltd. and the ESBWR by General Electric Hitachi also demonstrate the recent state of digital I&C architectures in NPPs. Descriptions of modern nuclear I&C can be found, e.g., in [4–6].

The architecture, the equipment (hardware) and software of the digital safety-related I&C (I&C platform) are designed to meet all safety-related I&C requirements in nuclear power plants. The dissimilarities between different I&C platforms may be significant. Not only the physical design but also the functional design, e.g. fault tolerant features and voting logic, may differ. On the other hand,the stringent safety requirements on design, manufacturing and operating of the safety systems and safety-related systems in the nuclear power plants lead consequently to recognizable similarities of the architecture of several digital safety-related I&C systems and of their functions.

The entire I&C architecture of the nuclear power plant can usually be divided into following levels of the interactions between technological process and process control functions: 1) process interface, 2) system automation and 3) unit supervision and control.

In the continuation of this paper, we will focus on the system automation level. The system automation level of a nuclear power plant usually consists of the reactor protection system (RPS), the safety automation system, the process automation system, and actuation and control equipment.The protection systems and the control systems are the two major parts of the safety automation.

Protection systems, belonging to the highest safety class (Cat. A in IEC 61226 [7]) are responsible for the primary safety functions consisting of reactor trip system and the engineered safety features actuation system (ESFAS). Protection systems (fig. 1) are composed of redundant divisions(or channels) running in parallel microprocessors and they actuate functions on demand (e.g., when process parameter limits are exceeded).

The divisions may be of the same or different architectures but in general all perform the same functions. Each division consists of multiple digital modules such as input module, processing module, communication module and output module (fig. 2). Each module comprises basic components such as an analog/digital converter, a multiplexer, a microprocessor and its associated components, a demultiplexer, and an A/D converter.

Control systems, e.g., turbine side automation, are versatile having both on demand and continuous functions. Control systems belong to a lower safety class (B or C). A control system is structured in the same manner as protection systems, except that control systems do not often have redundant channels.

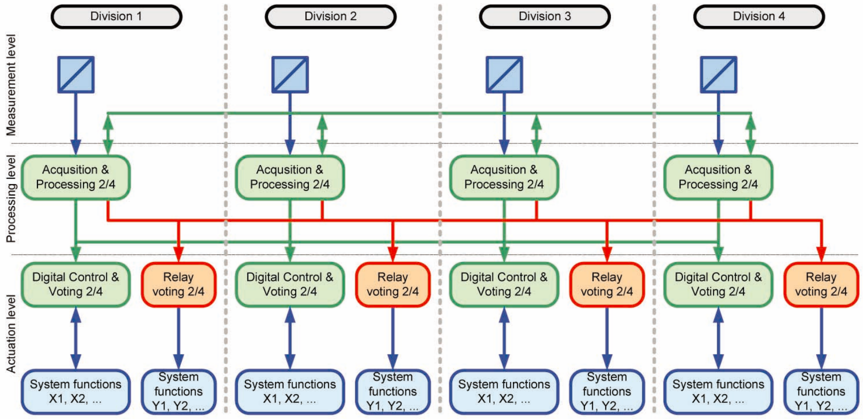
Division 1
Division 2
Division 3
Division 4
Measurement level
Processing level
Actuation level
Acqusition & Processing 2/4
Digital Control & Voting 2/4
Relay voting 2/4
System functions X1, X2, ...
System functions Y1, Y2, ...

Fig. 1. Example of a Digital I&C Protection System Architecture.

3. STATE-OF-THE-ART OF RELIABILITY ANALYSIS OF I&C SYSTEMS IN PRA CONTEXT

3.1 Overview

Digital I&C systems include unique features, such as complex dynamic interactions and the usage of software, that can be difficult to take into account with traditional PRA methods such as with the event tree-fault tree approach. Generally, dynamic methodologies provide a more accurate representation of probabilistic system evolution in time than the event tree/fault tree (ET/FT) approach. However, the dynamic models are on a trial stage and usually it is a difficult task to integrate dynamic models to existing PRAs.

A summary of experiences of modelling digital systems in CSNI member countries can be found in [1]. The report also presents a set of recommendations for method development, data collection and analysis, and international cooperation.

There is a general consensus that protection systems (RPS & ESFAS) shall be included in PRA, while control systems can be treated in a limited manner. The system architecture and the mode of operation of protection systems versus control systems are different, which creates a different basis for the reliability analysis and modelling. Since the reliability analysis of protection systems is considered prioritised, we will limit the discussion of this paper to the protection systems.

3.2 Modelling Digital I&C in PRA

The applicability of traditional PRA methods (event tree-fault tree and Markov modelling) for digital systems has been surveyed in [2]. Traditional methods are useful in the modelling but also indicates some limitations of the methods. The event tree-fault tree approach does not explicitly treat the timing of events in accident sequences and interactions with plant processes are implicitly and approximately considered. A set of desirable characteristics for a probabilistic model of a digital system has been identified. Additionally, a preliminary list of areas where additional research could enhance the state-of-the-art of modelling digital system is identified.

The incorporation of a model of a digital RPS into a PRA is discussed in [8]. The work demonstrated that modelling the digital RPS on an adequate level is challenging, and new approaches are required. An overview of the issues regarding the development of a static fault-tree-based risk model is presented in [9]. The complicated issues of digital system PRA are categorized into four groups based ontheir characteristics: hardware module, software, system, and safety function. The key issues related to modeling the PRA of nuclear safety digital I&C systems summarized in [10]. The probability risk quantification techniques is presented to each of the issues.

The utilization of traditional methods to model a digital feedwater control system is discussed in [11]. In the case study only the Markov method was used as the order of component failures was considered important. The study demonstrated that the proposed approach is feasible for analyzing digital system. However, the intergration with a PRA based on the ET/FT method may not be a trivial task.

Risk insights associated with digital upgrade is discussed in [11, 12]. In the development of the digital I&C PRA model a pragmatic approach was taken, as the quantification of software reliablity is a challenging problem. The research focused on important engineering insights that can be reach by understanding the role of the digital system with respect to the plant systems and the plant itself.

For representing the effect of I&C at a PRA level, EDF has since the 90's been developing the Compact Model [13]. The Compact Model of digital I&C is a

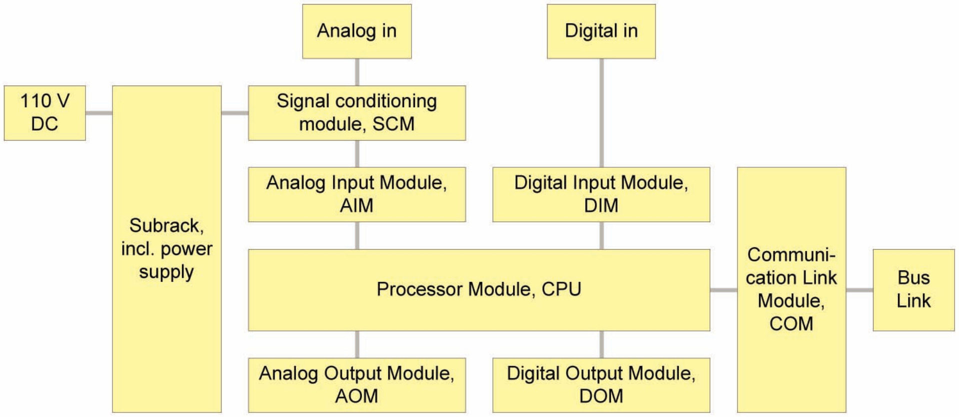
Analog in
Digital in
110 V DC
Subrack, incl. power supply
Signal conditioning module, SCM
Analog Input Module, AIM
Digital Input Module, DIM
Processor Module, CPU
Communi-cation Link Module, COM
Bus Link
Analog Output Module, AOM
Digital Output Module, DOM

Fig. 2. Example of Modules Included in a Computerized I&C Unit.

functional representation that comprises the main outcomes of digital I&C experts' safety and dependability assessments that can be shared with PRA experts and incorporated in a PRA model. The purpose of the Extended compact model is to form a connection between the probabilistic assessment at plant level and the deterministic assessment at I&C level, by a step by step approach. The idea is to "descend" from PRA to critical parameters identification, and to "ascend" from deterministic assessment of factors contributing to I&C safety to its representation in a PRA.

Failure modes and effects analysis (FMEA) is a well-known method for identifying failure modes of a system and their effects or consequences on the system. A few guidance documents for performing a FMEA are available, e.g. [14], but there are no specific guidelines on how to perform FMEA for digital systems. The absence of failure classification is a major issue in the representation of failure modes and mechanisms of digital I&C systems. A preliminary survey on failure modes and failure mechanisms in digital components and systems is presented in [15].

FMEA by itself may not be a sufficient tool to determine how specific component-level failure modes affect digital systems [16]. Therefore, it could be useful to utilize more sophisticated tools, such as simulation tools, to analyze the interactions between the components of a digital system and the effects of one or more failures. A systematic FMEA approach is proposed in [17] for creating reliability models for digital instrumentation and control systems.

3.2.1 Dynamic Reliability Modelling Approaches

There exists several dynamic reliability approaches, for instance, Dynamic Flowgraph Methodology (DFM) [18–20], Markov/CCMT (cell-to-cell mapping technique) [21, 22], Petri Nets [23], Bayesian approaches [24–26], test-based approaches [27], Boolean logic Driven Markov Process (BDMP)[28], and black box approaches [29, 30]. DFM and Markov/CCMT were ranked as the two top dynamic reliability modelling approaches with the most positive features and least negative features [27].

DFM is based on directed graphs for modeling and analyzing the behavior and interaction of software and hardware within an embedded system [18]. Dynamic flowgraphs can predict future failures and integrate hardware and software components. However, extensive technical knowledge is required for the creation of a DFM model. Continuous variables have to be discretized, which is a trade of between model accuracy and complexity and analysis time. The number of time steps that can be analyzed in deductive mode is limited by computational constraints.

The Markov/CCMT approach combines the traditional Markov methodology with cell to cell mapping. The approach enables to represent possible couplings between failure events, originated from dynamic interactions between the digital I&C system and the controlled process, and among the different components of the I&C system [21]. Construction of a full Markov/CCMT model may not be computationally feasible if the analyzed system contains a large number of states. It requires a substantially larger amount of technical knowledge compared to that needed for a traditional ET/FT analysis.

A benchmark implementation of a digital feedwater control system modelled with the two methodologies is discussed in [21]. A brief comparison between the results obtained with the two dynamic methodologies and results computed for the same system with traditional PRA methods is discussed in [11]. The integration of the results obtained with the dynamic model is fairly straightforward, if the basic events identified by the dynamic models do not also appear as basic events elsewhere in the standard PRA models.

Model checking [31] is a computer aided automatic verification technique for formally verifying the correct functioning of a system design model against its formal specification. Model checking is not directly applicable for reliability assessment of digitalized I&C systems. An approach, that combines a safety assessment methodology (fault tree analysis) and a formal methodology (model checking) to provide formal, automated and qualitative assistance to informal and quantitative safety assessmentis presented in [32]. An application of model checking and fault tree analysis for the safety analysis of an embedded system is described in [33]. The use of model checking for fault coverage analysis has been proposed in [34, 35]. Also efficient symbolic techniques for probabilistic model checking have been developed, e.g. [36].

3.2.2 Software Reliability Modelling

Software failures are in general mainly caused by systematic (i.e. design specification or modification) faults, and not by random errors. Software based systems cannot easily be decomposed into components, and the interdependence of the components cannot easily be identified and modelled. Applying software reliability models in the PRA context is hence not a trivial matter.

Software reliability models usually rely on assumptions and statistical data collected from non-nuclear domain and therefore may not be directly applicable for software products implemented in nuclear power plants. More important than the exact values of failure probabilities are the proper descriptions of the impact that software-based systems has on the dependence between the safety functions and the structure of accident sequences. Conventional FT-approach is, on the other hand, considered sufficient for the modelling of RPS like functions.

In spite of the unsolved issue of addressing software failures there seems be a consensus regarding some philosophical aspects of software failures and their use in developing a probabilistic model. The basic question: "What is the probability that a safety system or a function fails when demanded" is a fully feasible and well-formed question for all components or systems independently of the technology on which the systems are based [37]. A similar conclusion was made in the Workshop on Philosophical Basis for Incorporating Software Failures in a Probabilistic Risk Assessment [38]. As part of the open discussion, the panelists unanimously agreed that:

- software fails
- the occurrence of software failures can be treated probabilistically
- it is meaningful to use software failure rates and probabilities
- software failure rates and probabilities can be included in reliability models of digital systems.

For the quantification of software failure rates and probabilities there are several general approaches, e.g., reliability growth methods, Bayesian belief network (BBN) methods, test based methods, rule based methods [37] and software metrics based methods [39, 40]. These methods are reviewed in [41].

Reliability growth models are based on the sequence of times between observed and repaired failures [37]. The models calculate the reliability and the current failure rate. Additionally, the reliability growth models can predict the time to next failure and required time to remove all faults.

The BBN methodology has been adapted to software safety assessment [42, 43] and the methodology can be considered as promising. One of the main drawbacks is that a different BBN has to be built for each software development environment. This problem may be solved by using generalized BBN templates which are not restricted to a specific development environment [44].

In test based methods a program is executed with selected data and the answer is checked against an 'oracle'. A reliability measure can be generated, by running a number of tests and measuring the number of failures. Test-based reliability models assume that the input data profile used during the test corresponds to the input profile during real operation. Unfortunately, this correspondence cannot often be guaranteed.

To assess software risk contribution, [45, 46] presents an application of Context-based Software Risk Model (CSRM). CSRM allows assessing the contribution of software and software-intensive digital systems to overall system risk in a way that can be integrated with the PRA format used by NASA described in [47]. PRA techniques for modelling digital I&C system software reliability focusing in the modelling of digital system software common-cause failures (CCF), and features of I&C systems that minimize potential CCF is described in [48].

3.3 Reliability Data for Digital I&C Systems

3.3.1 Hardware Reliability Data

Usually, hardware failure data is provided by the vendor of the equipment. This is standard requirement in the contract between the utility and the vendor. The data provided by the supplier sets the limit for the detail of the PRA, i.e., it is not feasible to model in more detail due to lack of reliability data. Two kinds of failure data may provided by vendors: 1) based on operating experience, 2) based on a part counting method followed by a standard like Siemens SN 29500 [49] or generic data bases such as the reliability prediction database the Military Handbook for "Reliability Prediction of Electronic Equipment" (MIL-HDBK-217) [50]. MIL-HDBK-217 contains failure rate models for the various part types used in electronic systems, such as integrated circuits, transistors, diodes, resistors, capacitors, relays, switches, and connectors. These failure rate models are based on mathematical models derived from empirical field failure rates that are gathered for different parts and systems. Those models respect ambient conditions, level of stress, and type of applications.

Failure data is typically provided in terms of failure rate (1/time unit). From the PRA modelling point of view it is necessary to distinguish between detected and latent failures, which depends on the failure detection features of the I&C units. The judgement of the share of detected vs. latent failure rates needs to be provided by the vendor.

A second important reliability parameter needed for PRA is CCF failure rates. CCF parameters are sometimes derived from some generic values, but as an alternative IEC 61508-6 [51] has been used, e.g., in [8].

3.3.2 Software Reliability Data

Sophisticated software reliability estimation methods presented in the academic literature are not applied in real industrial PRAs (see chapter 3.2.3 for the software reliability modelling methods). Instead, the numbers are some kind of engineering judgments for which justifications may be hard to find.The engineering judgement approaches can be divided into the following categories depending on the argumentation and evidence they use [52]:

- screening out approach
- screening value approach
- expert judgement approach
- operating experience approach.

The reliability model used for software failures is practically always the simple "probability of failure per demand", denoted here by the parameter *q*.

3.3.2.1 Screening Out Approach

Screening out approach means that software failures are screened out from the model. The main arguments to omit software are that 1) the contribution of software failures is insignificant or that 2) no practical method to assess the probability of software failure (systematic failure).

One approach is to model software failures but not to define reliability values. The impact of software failures are assessed through sensitivity approaches. This approach has been utilized, for instance, in Ringhals 2 [53]. In [9], values 0, 1E-4 and 1E-3 for *q* were used in sensitivity analyses as software failure probabilities to analyse the impact of software failures on the system unavailability and the plant risk.

3.3.2.2 Screening Value Approach

Screening value approach means that some reliability number, like q = 1E-4, is chosen without detailed assessment of the reliability, and it is claimed that this is a conservative number for a software CCF. The screening value is taken from a reference like IEC 61226 [54]. Accordingly, the reference [55] states that reliability claims "q < 1E-4" for a single software based system important to safety shall be treated with extreme caution. This derives partly due to the fact that demonstrating lower probabilities, e.g., by statistical testing is very laborious.

3.3.2.3 Expert Judgement Approach

Expert judgement approach relies on the assessment of the features of the software system which are assumed to have correlation with the reliability. The two questions are 1) which features should be considered and 2) what is the correlation between the features and the reliability. This kind of approaches are used extensively in PRA, e.g., in human reliability analysis. Such models are difficult to validate.

In a case study on quantitative reliability estimation of a software-based motor protection relay, Bayesian networks were used to combine evidence from expert judgment and operational experience [43].

In [56] it was assumed that the contribution from software failure to total failure probability is 10% of the hardware failure probabilities. The rationale to this was that there are two well recognized aspects of software reliability: 1) the contribution of software failures to total failure of a digital system is smaller compared to exclusive failure of hardware, 2) there is a threat of software related common cause failures for a group of identical and redundant components. The second aspect was addressed by selecting a suitable value for β in the beta-factor CCF model. Value $\beta = 0.03$ was given, including CCFs due to hardware and software.

SIL-value (safety integrity level of IEC 61508 [57]) approach is also an example of an expert judgement approach, where the reliability target implied by the SIL is interpreted as the unavailability of the item.

3.3.2.4 Operating Experience Approach

Operating experience approach means an assessment based on operational data. In reality, operating experience approach is like the expert judgement approach since operational data need to be interpreted in some way to be used for reliability estimation. Especially if the reliability estimation is not carried out explicitly using well-defined data and reliability models.

In the PRA study of the Swedish NPP Ringhals 1, the contribution of software CCF to the unavailability of a safety system was assessed based on operational experience [8]. The operational experience of over 60 similar systems showed no CCF caused by platform properties and thus the contribution of platform CCF was estimated at 1E-8. Two events could be considered as CCF, which lead to an unavailability of safety I&C systems as 1E-6. This value was applied for redundant I&C units.

In [48], reasonable estimates for the relative contribution of software to digital system reliability software CCF probabilities were developed based on operational experience and engineering judgment. The CCF of operating system software was estimated as 1E-7 based on data gathered from dozens of plants during a time period of more than 10 years. For the application software, the CCF probability was estimated as 1E-5 for each function group. The SIL-4 targets were used as a general guide in the estimate. Additionally, it is suggested that if multiple application software CCFs appeared in same cut set the dependency between the two CCFs should be assessed. One way to take this into consideration is to assume a beta factor between the two software CCF events. Values $0.001 < \beta < 0.1$ were recommended, depending on the similarity of the software.

3.3.2.5 Conclusions on Software Reliability

Generally, only common cause failures are modelled in PRA. One reason for this is that there has not been a methodology available to correctly describe and incorporate software failures into a fault tree model. The only reliability model which is applied is constant unavailability (q) and this is used to represent the probability of CCF per demand. Spurious actuations due to software failures are not modelled or no need to consider software failure caused spurious actuations has been concluded.

Software CCF is usually understood as the application software CCF or its meaning has not been specified. Software CCF is generally modelled between processors performing redundant functions, having the same application software and on the same platform. One of the exceptions is the design phase PRA made for the automation renewal of the Loviisa NPP, where four different levels of software failures are considered: 1) single failure, 2) CCF of a single automation system, 3) CCF of programmed systems with same platforms and or software, and 4) CCF of programmed systems with different platforms and or software [52].

With regard to the reliability numbers used in PRA, it is difficult to trace back where they come from — even in the case of using operating experience. The references indicate the sort of engineering judgement but lacks supporting argumentation. To overcome the shortcomings of the present approaches for software failure rate estimation, an analytical approach is provided in [58].

4. WGRISK TASK GROUP DIGREL

4.1 Background

In 2007, the OECD/NEA CSNI directed the Working Group on Risk Assessment (WGRisk) to set up a task group to coordinate an activity in the digital system reliability field. One of the recommendations of this activity was to develop a taxonomy of failure modes of digital components for the purposes of PRA [1]. This resulted in a follow-up task group called DIGREL. An activity focused on development of a common taxonomy of failure modes was seen as an important step towards standardised digital I&C reliability assessment techniques for PRA. Needs from PRA will guide the work, meaning e.g. that I&C system and its failures are studied from their functional significance point of view. The taxonomy will be the basis of future modelling and quantification efforts. It will also help define a structure for data collection and to review PRA studies.

The DIGREL task has taken advantage from ongoing R&D activities, actual PRA applications as well as analyses of operating experience related to digital systems in the OECD/NEA member countries. The scope of the taxonomy includes both protection and control systems of a nuclear power plant, though primary focus is on protection systems. The taxonomy is divided into hardware and software related failure modes, for which purpose example taxonomies have been collected. A representative fictive digital protection system example has been developed to be used as a reference in the application and demonstration of the taxonomy. Results presents here should be considered preliminary proposals and not as the Task Group consensus thoughts.

4.2 General Approach for the Failure Modes Taxonomy

Failure modes taxonomy is a framework of describing, classifying and naming failure modes associated with a system. Standard technological equipment of NPP protection systems, like pumps, are either in the running or standby mode. On the opposite, computer based systems are typically always in the running mode – the difference in the modes is that they process different sets of input parameters and consequently solve different branches of algorithms. The need of specific taxonomy establishment is hence obvious.

One of the main uses of digital equipment failure modes taxonomy is to support the performance of reliability analyses and to unify the operational experience data collection of digital I&C systems. In PRA, failure modes taxonomy is applied in the systems analysis, including the performance of FMEA and the fault tree modelling. Systems analysis is a combination of top down and bottom up approaches. Fault tree modelling is a top down method starting from the top level failure modes defined for the system. In the system level, the two main failure modes are 1) failed function and 2) spurious function. For the failed function more descriptive definitions may be given such as “no function”, “not sufficient output”, “no state transition”, “broken barrier”, “loss of integrity”, and “masking failure”, depending on the nature of the system. In the fault tree analysis, the system level failure modes are broken down further into sub-system and component level failure modes. The system level failure modes appear thus as fault tree gates in the PRA model, while component level failure modes appear as basic events.

Basically, the same failure modes taxonomy can be applied for components as at the system level (failed function, spurious function), but the definitions are usually more characterising, e.g., “sensor freeze of value”, and are closer related to the failure mechanisms or unavailability causes. The component level failure modes are applied in the performance of the FMEA, which is a bottom-up analysis approach. The analysis follows the list of components of the system and for each component failure modes, failure causes (mechanisms) and associated effects are identified. FMEA precedes the fault tree modelling but it needs the definitions of the system functions and associated failure modes.

In PRA, the definitions for the failure modes and the related level of details in the fault tree modelling can be kept in a high level as long as relevant dependencies are captured and reliability data can be found.

4.3 Requirements for the Failure Modes Taxonomy

The development of a taxonomy is dependent on the overall requirements and prerequisites since they will set boundary conditions e.g. for the needed level of detail of hardware components and for the structure of the failure modes. A different set of requirements may result in a different taxonomy. The following targets for the taxonomy have been defined:

- to support PRA practice, i.e. fulfil PRA requirements/ conditions
- to cover undetected and detected failures
- to capture all critical dependencies and design features
- to be appropriate for safety related systems
- to support definition of failure modes, not mechanisms
- to be based on function view, not component
- to constitute a proper base for specific data gathering
- to support modelling of CCFs at the necessary level.

4.4 Outline of the Failure Modes Taxonomy

4.4.1 Levels of Details

With regard to the analysis and modelling of protection systems, the following levels of details have to be distinguished

1. the entire system
2. a division (or channel)

3 I&C units
4. modules
5. basic components.

A safety system is the entity performing a safety function or part of it. In PRA, RPS is never treated as a black box, but the analysis is always broken down into the protection functions and at least to the divisional level.

The divisions may be of the same or different architectures but in general all perform the same functions. Each division comprises an entity from power supply and physical separation point of view, although some cross-connections of power supply between divisions may be applied for certain components. From the PRA modelling point of view, a usual simplification is to assume a loss of complete division in case of a hazard affecting the division (fire or flooding initiating event). Loss of AC or DC power supply are also division wide functional failures to be considered in PRA.

Each division consists of several I&C units (e.g. APUs and DCVs) and data buses between them (see fig:s 1 and 3). I&C units are installed in cabinets, each of which has a specific power supply route and condition monitoring. Cabinet level is the most detailed level from the power supply and room dependency point of view.

An I&C unit is a computerised system designed to receive input signals, perform computing and send output. It consists of modules such as input module, processing module, communication module and output module (fig. 2). Modules may be further broken down into basic components such as an analog/digital converter, a multiplexer, a microprocessor and its associated components, a demultiplexer, an A/D converter and channels of an I/O module, e.g., depending on the available failure data. Modules and channels are the most detailed level from the hardware functional dependency point of view. Also the software components can be associated with the modules (Table I).

The taxonomy is divided into hardware and software related failure modes, for which purpose example taxonomies have been collected from the member countries.

4.4.2 Example System

A representative fictive digital protection system example has been developed to be used as a reference in the application and demonstration of the taxonomy. The example is similar to what is described in chapter 2, i.e., a four-redundant nuclear plant. The reactor protection system is organised into two separated subsystems, named Subsystem A (or SSA) and Subsystem B (or SSB) (see fig. 3).

The two subsystems are based on the same safety-grade and computer-based I&C platform, but implement functions that are diverse and redundant, so that the failure of a function in one subsystem can, up to a certain point, be backed-up by a functionally diverse function (with different input signals) in the other subsystem. Each subsystem division is composed of several APUs

Table 1. Software Modules in Reactor Protection System

Unit	Software modules
I&C unit	• Operating system • Application specific software • Elementary functions
Data communication unit	• Operating system • Data communication software • Data link configuration

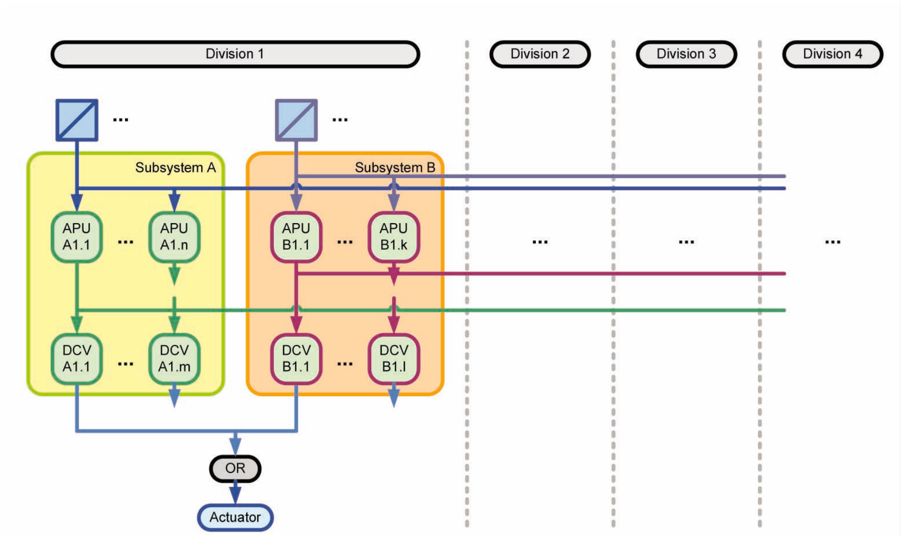

Fig. 3. DIGREL Example System Architecture.

implementing different functions. The APUs do not communicate with one another, and send their outputs only to the DCV (see fig. 3) of their division and subsystem.

4.4.3 Hardware Failure Modes Taxonomy

The hardware taxonomy failure modes can either be based on a function view or a component view. The function view considers component failures with regard to their impact on the function that the component supports, e.g. "loss of function to actuate", while the component view is more descriptive and considers component failures with regard to the manifestation of the failure within the component, e.g. "freeze of value" or "set point corrupted".

In PRA, it is practical to group failure modes with regard to their functional consequence to as high extent as possible, in order to simplify the fault tree analysis. See also [53, 59, 60, 61] for examples of failure modes used in practice. At generic level, the two main failure modes are:

- Loss of function, loss of communication, and no actuation signal when demanded (masking failure)
- Spurious function, and a spurious actuation signal.

Other failure modes, such as erratic output, may be considered where applicable, but in practical PRA applications it may be difficult to consider more ambiguous events than "failure to actuate" or "spurious actuation".

Failure detection is an important aspect of the failure mode. Firstly, failure detection determines the choice of the component reliability model (constant unavailability, monitored, repairable, and periodically tested). Secondly — specifically for I&C systems — failure detection is a relevant attribute from the failure effect point of view. Detected failure may cause a spurious actuation signal or change the voting logic, depending on the design. To accurately model the effect of detected failures may be a laborious task in practice, but failure detection should be analysed and considered at least in FMEA. The following categories of the failure detection are possible:

- Demand (no periodic test detects the failure)
- Periodic test
- Monitoring
- Self-monitoring (online monitoring of the module itself)
- Monitoring by another module
- Self-announced failure.

With regard to the hardware failure modes taxonomy, the module level (see fig. 2) seems to be the most appropriate from the PRA modelling point of view. The module level concurs with the level of detail of general state of the art PRAs and it will make it feasible to perform, maintain and review a PRA of a digital I&C with reasonable resources while capturing critical dependencies. It will also be possible to capture fault tolerant features of the digital system and the impact on the reliability of safety functions.

4.4.4 Software Failure Modes Taxonomy

The software failure modes taxonomy is still an open issue. A set of principally critical failure events associated with software faults can be defined. I&C experts are responsible to judge which of the failure events, being typically common cause failures (CCF), are reasonable to be postulated.

The way of defining software failure modes is rather dissimilar to the hardware ones due to the nature of software [62]. In the DIGREL task, the software failure modes taxonomy has been approached from two perspectives: PRA and software engineering. The PRA perspective follows the functions of the system, e.g., RPS, and considers the critical failure modes of the system.

Knowing the functions of the I&C units, the following functional failures (which are common cause failures) may be considered for the example system type of design (see chapter 4.4.2 and fig. 3):

- Function(s) failure in one subsystem (SSA or SSB)
- Function(s) failure in both subsystems (SSA and SSB)
- Loss of one set of redundant APUs (e.g. APUs A1.1, …, A4.1)
- Loss of multiple sets of redundant APUs in one subsystem only
- Loss of multiple sets of redundant APUs in both subsystems
- Loss of one subsystem only (complete SSA or SSB)
- Loss of one subsystem and of set(s) or redundant APUs in the other subsystem
- Loss of both subsystems (complete SSA and SSB).

In some special cases, some more complex functional failures may be considered.

Software faults may be assumed, in principle, in any software module of an I&C unit (Table I). In order to simplify the analysis, the maximum possible extent of the activation of a single postulated software fault may be assumed. Scopes and types may be restricted when considering measures taken to prevent specific failure mechanisms. Which of these "software basic events" are reasonable to assume and which of them are fully unreasonable to postulate is a judgement task for the software system expert. Based on the list of possible functional failures and the CCF options, we get a set of principally possible basic events associated with software module faults. This approach will be developed further and demonstrated with the example system in the DIGREL task.

5. CONCLUSIONS

The advent of digital I&C systems in nuclear power plants has created new challenges for safety analysis. To assess the risk of nuclear power plant operation and to determine the risk impact of digital systems, there is a need to quantitatively assess the reliability of the digital systems in a justifiable manner. Due to the many unique attributes of digital systems, a number of modelling and data collection challenges exist, and consensus has not yet been reached.

Currently in PRA computer-based systems are mostly analyzed simply and conventionally. The conventional failure mode and effects analysis and failure tree modelling are utilized. As basic events CPU failures, application software failures and CCFs between identical components are modelled. However it is not clear wich failure modes or system parts CCFs should be postulated. The primary goal is to model dependencies.

A clear distinction can be made between the treatment of protection and control systems controlling e.g. the turbine plant. There is a general consensus that protection systems shall be included in PRA, while control systems can be treated in a limited manner.

The survey of literature and PRA shows that software failures are either omitted in PRA or modelled in a very simple way as CCF related to the application software of operating system. It is a difficult basis for the numbers used except the reference to a standard statement that a failure probability 10^{-4} per demand is a limit to reliability claims, which limit is then categorically used as a screening value for software CCF.

Dynamic methodologies can provide a more accurate representation of probabilistic system evolution in time than the FT approach. These methods included unique features that makes them suitable for specific applications, but they do not solve the problem of software reliability.

In the OECD/NEA DIGREL task, the taxonomy will be developed jointly by PRA and I&C experts. An activity focused on the development of a common taxonomy of failure modes is seen as an important step towards standardised digital I&C reliability assessment techniques in PRA. PRA needs will guide the work, meaning e.g. that I&C system and its failures are studied from their functional significance point of view. The taxonomy will be the basis of future modelling and quantification efforts. It will also help define a structure for data collection and to review PRA studies.

The scope of the taxonomy will include both protection and control systems of a nuclear power plant, though primary focus is on protection systems. The taxonomy is divided into hardware and software related failure modes, for which purpose example taxonomies have been collected from the member countries. A representative fictive digital protection system example has been developed to be used as a reference in the application and demonstration of the taxonomy.

With regard to the hardware failure modes taxonomy, the main issue is to define a feasible level of details. Module level, i.e. subcomponents of processing units, seems to be the most appropriate from the PRA modelling point of view. The software failure modes taxonomy is focused on identifying and defining common cause failures which are reasonable to postulate.

ACKNOWLEDGEMENTS

Contributions from the WGRISK/DIGREL task group members are acknowledged. The following organisations form presently (May 2012) the task group, being responsible for planning and organisation of work meetings and preparation of the best practice guidelines: VTT, Finland (leader); Risk Pilot, Sweden; IRSN, France; EDF, France; AREVA, France; GRS, Germany; KAERI, Korea; NRC, USA; Ohio State University, USA; NRI, Czech; JNES, Japan; VEIKI, Hungary; ENEL, Italy; NRG, the Netherlands; RELKO, Slovakia and CSNC, Canada.

The Finnish and Swedish work has been financed by NKS (Nordic nuclear safety research), SAFIR2014 (The Finnish Research Programme on Nuclear Power Plant Safety 2011–2014) and the members of the Nordic PSA Group: Forsmark, Oskarshamn Kraftgrupp, Ringhals AB and Swedish Radiation Safety Authority. NKS conveys its gratitude to all organizations and persons who by means of financial support or contributions in kind have made the work presented in this report possible.

REFERENCES

[1] “Recommendations on assessing digital system reliability in probabilistic risk assessments of nuclear power plants,” NEA/CSNI/R(2009)18, OECD/NEA/CSNI, Paris (2009).

[2] T.L. Chu, G. Martinez-Guridi, M. Yue, J. Lehner and P. Samanta, “Traditional Probabilistic Risk Assessment Methods for Digital Systems”, NUREG/CR-6962, United States Nuclear Regulatory Commission, Washington D.C. (2008).

[3] P. Haapanen, A. Helminen, U. Pulkkinen, ”Quantitative reliability assessment in the safety case of computer-based automation systems,” STUK-YTO-TR 202, STUK, Helsinki (2004).

[4] “Core Knowledge on Instrumentation and Control Systems in Nuclear Power Plants,” IAEA Nuclear Energy Series No. NP-T-3.12, International Atomic Energy Agency, Vienna (2011).

[5] R. Kisner, J. Mullens, T. Wilson, R. Wood, K. Korsah, A. Qualls, M. Muhlheim, D. Holcomb and A. Loebl, “Safety and Non-Safety Communications and Interactions in International Nuclear Power Plants, Guidelines for the Design of Highly Integrated Control Rooms,” ORNL/NRC/LTR-07/05, Oak Ridge Laboratory, Oak Ridge (2007).

[6] “Instrumentation and Controls in Nuclear Power Plants: An Emerging Technologies Update,” NUREG/CR-6992, United States Nuclear Regulatory Commission, Washington D.C. (2009).

[7] “Nuclear power plants. Instrumentation and control important to safety. Classification of instrumentation and control functions,” IEC 61226, International Electrotechnical Commission, Geneva, ed. 3.0 (2009).

[8] S. Authén, E. Wallgren and S. Eriksson, “Development of the Ringhals 1 PSA with Regard to the Implementation of

a Digital Reactor Protection System," *Proc. 10th International Probabilistic Safety Assessment & Management Conference, PSAM 10*, Seattle, Washington, June 7–11, 2010, paper 213.

[9] H.G. Kang and S.-C. Jang, "Issues And Research Status For Static Risk Modeling Of Digitalized Nuclear Power Plants," *Proc. 6th American Nuclear Society International Topical Meeting on Nuclear Plant Instrumentation, Control, and Human-Machine Interface Technologies, NPIC&HMIT 2009*, Knoxville, Tennessee, April 5–9, 2009.

[10] L. Shi, R. Enzinna, S. Yang and S. Blodgett, "Probabilistic Risk Assessments of Digital I&C in Nuclear Power Plant," *Proc. 10th International Probabilistic Safety Assessment & Management Conference, PSAM 10*, Seattle, Washington, June 7–11, 2010, paper 173.

[11] T.L. Chu, M. Yue, G. Martinez-Guridi, K. Mernick, J. Lehner and A. Kuritzky, "Modeling a Digital Feedwater Control System Using Traditional Probabilistic Risk Assessment Methods," NUREG/CR-6997 BNL-NUREG-90315-2009, United States Nuclear Regulatory Commission, Washington D.C. (2009).

[12] D. Blanchard and R. Torok, "Risk Insights Associated with Digital Upgrades," *Proc. 10th International Probabilistic Safety Assessment & Management Conference, PSAM 10*, Seattle, Washington, June 7–11, 2010, paper 453

[13] N. Thuy and G. Deleuze, "A Mixed Approach to Assess the Impact of I&C in PSA," *Proc. 6th American Nuclear Society International Topical Meeting on Nuclear Plant Instrumentation, Control, and Human-Machine Interface Technologies, NPIC&HMIT 2009*, Knoxville, Tennessee, April 5-9, 2009.

[14] "IEEE Guide for General Principles of Reliability Analysis of Nuclear Power Generating Station Safety Systems," IEEE Std. 352, Institute of Electrical and Electronics Engineers, Inc., New York (1987).

[15] S.M. Cetiner, K. Korsah and M.D. Muhlheim, "Survey on Failure Modes and Failure Mechanisms in Digital Components and Systems," *Proc. 6th American Nuclear Society International Topical Meeting on Nuclear Plant Instrumentation, Control, and Human-Machine Interface Technologies, NPIC&HMIT 2009*, Knoxville, Tennessee, April 5–9, 2009.

[16] P. Haapanen and A. Helminen, "Failure mode and effects analysis of software-based automation systems," STUK-YTO-TR 190, STUK, Helsinki (2002).

[17] T.-L. Chu, M. Yue, G. Martinez-Guridi and J. Lehner, "A Generic Failure Modes and Effects Analysis (FMEA) Approach for Reliability Modeling of Digital Instrumentation and Control (I&C) Systems," *Proc. 10th International Probabilistic Safety Assessment & Management Conference, PSAM 10*, Seattle, Washington, June 7–11, 2010, paper 82.

[18] C.J. Garrett, S.B. Guarro and G.E. Apostolakis, "The Dynamic Flowgraph Methodology for Assessing the Dependability of Embedded Software Systems," *IEEE Trans. on Systems, Man and Cybernetics* 25 (1995) 824–840.

[19] C.J. Garrett and G.E. Apostolakis, "Automated hazard analysis of digital control systems," *Reliability Engineering and System Safety*, 77(2002) 1–17.

[20] M. Yau, S. Guarro and G. Apostolakis, "Demonstration of the Dynamic Flowgraph Methodology using the Titan II Space Launch Vehicle Digital Flight Control System," *Reliability Engineering and System Safety* 49 (1995) 335–353.

[21] T. Aldemir, S. Guarro, J. Kirschenbaum, D. Mandelli, L.A. Mangan, P. Bucci, M. Yau, B. Johnson, C. Elks, E. Ekici, M.P. Stovsky, D.W. Miller, X. Sun, S.A. Arndt, Q. Nguyen and J. Dion, "A Benchmark Implementation of Two Dynamic Methodologies for the Reliability Modeling of Digital Instrumentation and Control Systems," NUREG /CR-6985, United States Nuclear Regulatory Commission, Washington D.C. (2009).

[22] P. Bucci, J. Kirschenbaum, L.A. Mangan, T. Aldemir, C. Smith and T. Wood, "Construction of event-tree/fault-tree models from a Markov approach to dynamic system reliability," *Reliability Engineering and System Safety* 93 (2008) 1616–1627.

[23] P.E. Labeau, C. Smidts and S. Swaminathan, "Dynamic reliability: towards an integrated platform for probabilistic risk assessment," *Reliability Engineering and System Safety* 68 (2000) 219–254.

[24] J. Pearl, *Probabilistic reasoning in intelligent systems: Networks of plausible inference*, Morgan Kaufmann Publishers, San Mateo, CA (1988).

[25] O. Doguc and J.E. Ramirez-Marquez, "A generic method for estimating system reliability using Bayesian networks," *Reliability Engineering & System Safety* 94(2009) 542–550.

[26] D.L. Kelly and C.L. Smith, "Bayesian inference in probabilistic risk assessment — The current state of the art," *Reliability Engineering & System Safety* 94(2009) 628–643.

[27] T. Aldemir, D.W. Miller, M.P. Stovsky, J. Kirschenbaum, P. Bucci, A.W. Fentiman and L.T. Mangan, "Current State of Reliability Modeling Methodologies for Digital Systems and Their Acceptance Criteria for Nuclear Power Plant Assessments," NUREG/CR-6901, United States Nuclear Regulatory Commission, Washington D.C. (2006).

[28] M. Bouissou, "Boolean logic driven Markov processes: A powerful new formalism for specifying and solving very large Markov models," *Proc. 6th International Conference on Probabilistic Safety Assessment and Management*, San Juan, Puerto Rico, USA, June 23–28, 2002.

[29] J.D. Musa and K. Okumoto, "A Logarithmic Poisson Execution Time Model for Software Reliability Measurement," *Proc. 7th International Conference on Software Engineering*, Orlando, FL, March 26-29, 1984, pp. 230-238.

[30] N.F. Schneidewind and T.W. Keller, "Applying Reliability Models to the Space Shuttle," *IEEE Software*, 28–33, (1992).

[31] E.M. Clarke, Jr., O. Grumberg and D.A. Peled, *Model Checking*, The MIT Press, Massachusetts Institute of technology, Cambridge, MA (2000).

[32] K.Y. Koh and P.H. Seong, "SACS2: A Dynamic and Formal Approach to Safety Analysis for Complex Safety Critical Systems," *Proc. 6th American Nuclear Society International Topical Meeting on Nuclear Plant Instrumentation, Control, and Human-Machine Interface Technologies, NPIC&HMIT 2009*, Knoxville, Tennessee, April 5–9, 2009.

[33] F. Ortmeier, G. Schellhorn, A. Thums, W. Reif, B. Hering and H. Trappschuh, "Safety analysis of the height control system for the Elbtunnel," *Reliability Engineering & System Safety*, 81(203) 259-268.

[34] M. Bozzano and A. Villafiorita, "The FSAP/NuSMV-SA Safety Analysis Platform," *International Journal on*

Software Tools for Technology Transfer, 9(2007) 5–24.

[35] S. Bingham and J. Lach, "Exhaustive Integrated Circuit Fault Coverage Analysis Using Formal Methods," *Proc. 6th American Nuclear Society International Topical Meeting on Nuclear Plant Instrumentation, Control, and Human-Machine Interface Technologies, NPIC&HMIT 2009*, Knoxville, Tennessee, April 5–9, 2009.

[36] M. Kwiatkowska, G. Norman and D. Parker, "PRISM: Probabilistic Model Checking for Performance and Reliability Analysis," *ACM SIGMETRICS Performance Evaluation Review* 36(2009) 40–45.

[37] G. Dahll, B. Liwång and U. Pulkkinen, "Software-Based System Reliability," Technical Note, NEA/SEN/SIN/WGRISK(2007)1, Working Group on Risk Assessment (WGRISK) of the Nuclear Energy Agency, Paris (2007).

[38] T.-L. Chu, G. Martinez-Guridi and M. Yue, "Workshop on Philosophical Basis for Incorporating Software Failures in a Probabilistic Risk Assessment," BNL-90571-2009-IR, Brookhaven National Laboratory (2009).

[39] C. Smidts and M. Li, "Software Engineering Measures for Predicting Software Reliability in Safety Critical Digital Systems," NUREG/GR-0019, United States Nuclear Regulatory Commission, Washington D.C. (2000).

[40] C. Smidts and M. Li, "Preliminary Validation of a Methodology for Assessing Software Quality," NUREG/CR-6848, U.S.NRC, Washington D.C. (2004).

[41] T.-L. Chu, M. Yue, G. Martinez-Guridi and J. Lehner, "Review of Quantitative Software Reliability Methods," BNL-94047-2010, Brookhaven National Laboratory (2010).

[42] A. Helminen, "Reliability estimation of safety-critical software-based systems using Bayesian networks," STUK-YTO-TR 178, STUK, Helsinki (2001).

[43] A. Helminen and U. Pulkkinen, "Reliability assessment using Bayesian network. Case study on quantative estimation of a software-based motor protection relay," STUK-YTO-TR 198, STUK, Helsinki (2003).

[44] H.-S. Eom, G.-Y. Park, H.-G., Kag and S.-C. Jang, "Reliability Assessment Of A Safety-Critical Software By Using Generalized Bayesian Nets," *Proc. 6th American Nuclear Society International Topical Meeting on Nuclear Plant Instrumentation, Control, and Human-Machine Interface Technologies, NPIC&HMIT 2009*, Knoxville, Tennessee, April 5-9, 2009.

[45] M. Yau and S. Guarro, "Application of Context-based Software Risk Model (CSRM) to Assess Software Risk Contribution in Constellation Project PRAs," *Proc. 10th International Probabilistic Safety Assessment & Management Conference, PSAM 10*, Seattle, Washington, June 7-11, 2010, paper 186

[46] S. Guarro, "Risk-Informed Safety Assurance and Probabilistic Assessment of Mission-Critical Software-Intensive Systems," NASA Technical Paper AR 07-01; JSC-CN-19704, ASCA, Inc., Redondo Beach, CA (2007).

[47] W. Vesely, M. Stamatelatos, J. Dugan, J. Fragola, J. Minarick III and J. Railsback, J., "Fault Tree Handbook with Aerospace Applications," NASA, Washington D.C. (2002).

[48] B. Enzinna, L. Shi and S. Yang, "Software Common-Cause Failure Probability Assessment," *Proc. 6th American Nuclear Society International Topical Meeting on Nuclear Plant Instrumentation, Control, and Human-Machine Interface Technologies, NPIC&HMIT 2009*, Knoxville, Tennessee, April 5–9, 2009.

[49] "Failure Rates of Components," SN 29500. Siemens AG, CT SR SI, Otto-Hahn-Ring 6, D-81739 Munich, Germany.

[50] "Reliability Prediction of Electronic Equipment, Notice 2" MIL-HDBK-217F(2), US Department of Defense, Washington D.C. (1995).

[51] "Functional safety of electrical/electronic/programmable electronic safety-related systems. Part 6: Guidelines on the application of IEC 61508:2 and IEC 61508:3, IEC 61508-6, International Electrotechnical Commission, Geneva (2000).

[52] K. Björkman, O. Bäckström, J.-E. Holmberg. "Use of IEC 61508 in Nuclear Applications Regarding Software Reliability — Pre-study," VTT-R-09293-11, VTT, Espoo (2012).

[53] S. Authén, K. Björkman, J.-E. Holmberg and J. Larsson, "Guidelines for reliability analysis of digital systems in PSA context — Phase 1 Status Report," NKS-230 Nordic nuclear safety research, Roskilde (2010).

[54] "Nuclear power plants – Instrumentation and control systems important to safety – Classification of instrumentation and control functions," IEC 61226. Second edition. International Electrotechnical Commission, Geneva (2005).

[55] "Licensing of safety critical software for nuclear reactors – Common position of seven European nuclear regulators and authorized technical support organisations," SSM Report 2010:01, SSM, Stockholm (2010).

[56] P.V. Varde, J.G. Choi, D.Y. Lee and J.B. Han, "Reliability Analysis of Protection System of Advanced Pressurized Water Reactor-APR 1400," KAERI/TR-2468/2003, Korea Atomic Energy Research Institute, (2003).

[57] "Function Safety of Electrical/Electronic/Programmable Safety-Related Systems, Part 1: General requirements," IEC 61508-1, International Electrotechnical Commission, Geneva (2010).

[58] "Estimating Failure Rates in Highly Reliable Digital Systems." EPRI TR-1021077, Electric Power Research Institute, Inc., Palo Alto, CA (2010). Limited distribution.

[59] S. Authén, J. Gustafsson and J.-E. Holmberg, "Guidelines for reliability analysis of digital systems in PSA context — Phase 2 Status Report," NKS-261 Nordic nuclear safety research, Roskilde (2012).

[60] T.-L. Chu and M. Yue, "A Comparison of Taxonomies of Digital System Failure Modes," *Proc. 11th International Probabilistic Safety Assessment & Management Conference, PSAM 11*, Helsinki, June 25–29, 2012.

[61] Proceedings of the DIGREL seminar "Development of best practice guidelines on failure modes taxonomy for reliability assessment of digital I&C systems for PSA", October 25, 2011, VTT-M-07989-11, VTT, Espoo (2011).

[62] J. Sedlak, "Software critical for safety in reliability models," *Proc. European Safety and Reliability (ESREL) Conference, ESREL 2009*, Prague, September 7–10, 2009.