TREATMENT OF EXPERT OPINION DIVERSITY IN BAYESIAN BELIEF NETWORK MODEL FOR NUCLEAR DIGITAL I&C SAFETY SOFTWARE RELIABILITY ASSESSMENT

Ming Li
U.S. Nuclear Regulatory Commission,
Washington, DC, USA,
Ming.Li@nrc.gov

Hyun Gook Kang*, and Sang Hun Lee
Department of Mechanical, Aerospace, and Nuclear Engineering
Rensselaer Polytechnic Institute,
110 8th St, Troy, NY, USA
kangh6@rpi.edu; lees35@rpi.edu

Seung Jun Lee
School of Mechanical and Nuclear Engineering
Ulsan National Institute of Science and Technology,
50, UNIST-gil, Ulsan, Republic of Korea
sjlee420@unist.ac.kr

Tsong-Lun Chu, Athi Varuttamaseni, and Meng Yue
Brookhaven National Laboratory,
Brookhaven Avenue, Upton, NY, USA
chu@bnl.gov; avarutta@bnl.gov; yuemeng@bnl.gov

Jaehyun Cho
Korea Atomic Energy Research Institute,
111, Daedeok-daero, Daejeon, Republic of Korea
chojh@kaeri.re.kr

Since digital instrumentation and control systems are expected to play an important role in safety systems in nuclear power plants (NPPs), the need to incorporate software failures into NPP probabilistic risk assessments has arisen. In order to estimate the failure probability of safety software in NPP and incorporate it into a PRA model, a Bayesian belief network (BBN) model was developed which estimates the number of defects in the software considering the software development life cycle (SDLC) characteristics. In the model, due to a lack of sufficient safety software operation experience data, expert opinion was instead used to quantify the distributed node probability tables (NPTs) that are tables of random variables whose probabilistic distributions were aggregated from experts' elicitation. In addition, handbook data on U.S. software developments and V&V as well as the testing results of two example nuclear safety software were used to Bayesian update the BBN distributed NPTs in order to reduce the BBN parameter uncertainty from the diverse expert opinion. Based on the estimated NPTs, the number of defects at each SDLC phase is evaluated for the typical digital protection software (50 function points and Medium development, V&V quality). This study is expected to provide insight on several aspects of BBN model quantification for nuclear safety-related software reliability assessment, including the expert opinion elicitation and aggregation, the representation of the node probabilities using probability distribution, and the Bayesian updating of the NPTs using available software development data.

I. INTRODUCTION

The instrumentation and control (I&C) systems in nuclear power plants (NPPs) have recently been replaced with digital systems due to the critical functional advantages that digital systems offer over conventional analog systems. However, digital I&C systems have distinct failure modes and causes, such as software failure, compared to analog systems; therefore, the integration of the NPP digital I&C system risk model into NPP probabilistic risk assessment (PRA) presents special challenges. Software failure can significantly affect the risk of digital protection systems in NPPs [1, 2]; for instance, the failure of reactor protection system (RPS) software may induce the failure of reactor-trip signal generation when a trip condition occurs. Therefore, the reliability of safety graded software in NPP must be quantified to assess the safety of digitalized NPPs.

In order to estimate the failure probability of the NPP safety graded software and incorporate it into a NPP PRA model, a Bayesian belief network (BBN) model was developed which estimates the number of defects in software programs considering the software development life cycle (SDLC) characteristics. In the model, SDLC characteristics such as the quality of software development and verification and validation (V&V) activities, and software-self characteristics such as program size and complexity are represented using a hierarchical structure. In order to estimate software reliability, the node probability tables (NPTs) were used for various BBN node parameters to contain probability information based on the belief relationships between the child and root nodes. In the model, the root nodes are allocated with prior distribution, and the child nodes are allocated with their conditional probability distributions where the NPT represents all possible combinations of the parent states for each child node.

Since the BBN model focuses on NPP safety-related software, NPTs should represent the variability among the class of safety-related software, and their values need to be estimated using safety software operation experience data. In this study, expert elicitation was instead used to cover the quantitative aspects of the model, such as NPTs for the prior and conditional probability distributions. The elicitations were conducted by distributing background material and questionnaires to experts, collecting and analyzing the provided answers, resolving the provided comments [3]. Based on the expert elicitation, distributed node probabilities were used to represent NPT instead of using discretized point estimates originally given by each expert to account for the variability among NPP safety-related software based on the different experts' answers on the BBN model parameter estimates.

Since experts have different levels of experience with nuclear safety-related software, a large diversity of opinions can be observed for some BBN nodes. To reduce the variance of such NPTs derived from the diverse expert opinion, handbook data on U.S. software development and V&V as well as the testing results for two trial nuclear safety software were used for the Bayesian update of the NPTs. Based on the estimated NPTs, the BBN model was applied to quantify the number of defects for a typical digital protection software having the size of 50 function points and having the Medium development and V&V qualities, as a case study.

II. MODEL STRUCTURE

In this study, a BBN framework for NPP safety software reliability quantification is developed that estimates the number of faults in a software program and further converts to the probability of software failure. The model captures NPP safety-related SDLC activity quality indicators and software development information, establishes the quantitative causal relationships between the indicators and the number of remaining defects, and further estimates software failure probability. The BBN model includes (1) identifying software development characteristics; (2) establishing and quantifying the causal relationships between these characteristics; (3) probabilistically aggregating multiple expert inputs; and (4) estimating the number of defects remaining, and the software failure probability using expert opinion.

As shown in Fig. 1, the SDLC consists of five phases (i.e. Requirements, Design, Implementation, Test, and Installation-and-Checkout phases) and the number of software faults remaining at the end of each phase is estimated. The model starts with the defects remaining in the Requirements phase and tracks the number of defects through all five phases of the SDLC. Any remaining defects at the end of one phase are passed on to the next phase and the total number of defects remaining in the software at the end of the last phase is further converted into a software failure probability on-demand.

In the model, the number of defects remaining in each phase is assumed to be dependent on two types of software development activities: development quality and V&V quality. The number of defects remaining in each phase is defined as a function of the development quality and the V&V quality, where the development process adds defects and the V&V process removes defects, at each SDLC phase.

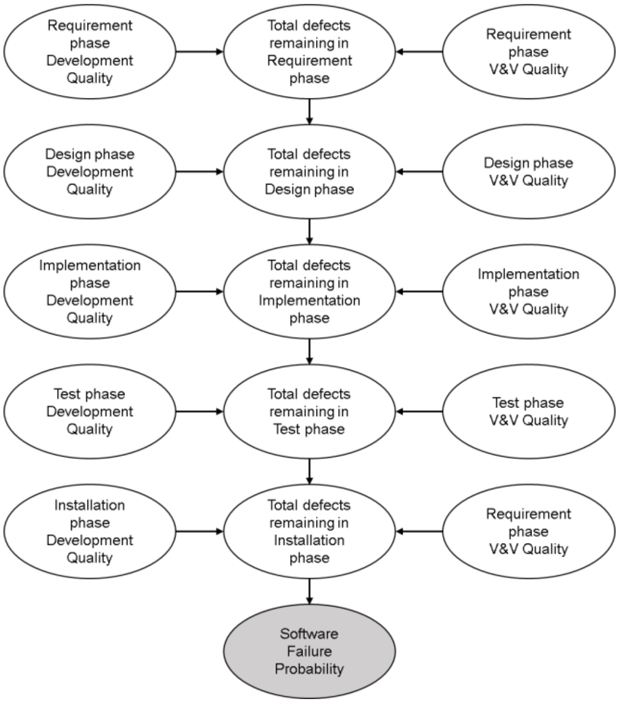

Requirement phase Development Quality
Total defects remaining in Requirement phase
Requirement phase V&V Quality
Design phase Development Quality
Total defects remaining in Design phase
Design phase V&V Quality
Implementation phase Development Quality
Total defects remaining in Implementation phase
Implementation phase V&V Quality
Test phase Development Quality
Total defects remaining in Test phase
Test phase V&V Quality
Installation phase Development Quality
Total defects remaining in Installation phase
Requirement phase V&V Quality
Software Failure Probability

Fig. 1. High-level structure of the Bayesian belief network model for software reliability quantification

As an example of the BBN sub-model, the process of software defect insertion and removal in the Design phase is shown in Fig. 2. The number of residual faults at a specific SDLC phase is determined by (1) the number of defects passed from the previous phase, if it is not the first phase of the SDLC, (2) the number of defects introduced during the current phase, and (3) the number of defects detected and removed by the V&V activities undertaken in the current phase.

In the model, it is assumed that the quality of the software development activities is directly related to the defect density (defects per function point) in the current phase. Other factors affecting the number of faults inserted are the size and complexity of the software. Similarly, the quality of the V&V activities is directly related to the detection probability for defects introduced in the current phase and defects passed from the previous phase. In addition, the number of function points (FPs), which represents the size and complexity of the software is assumed to affect the defect detection probabilities. It is notable that the BBN structures of other SDLC phases can be modeled in a similar manner.

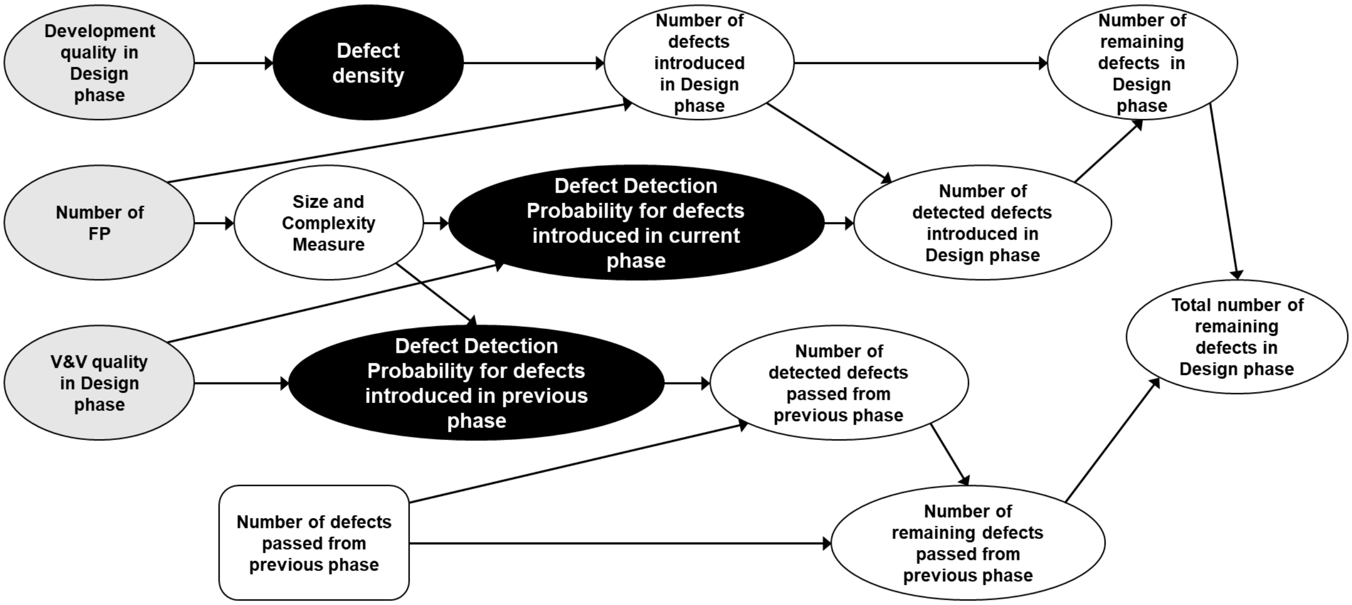

Development quality in Design phase
Defect density
Number of defects introduced in Design phase
Number of remaining defects in Design phase
Number of FP
Size and Complexity Measure
Defect Detection Probability for defects introduced in current phase
Number of detected defects introduced in Design phase
Total number of remaining defects in Design phase
V&V quality in Design phase
Defect Detection Probability for defects introduced in previous phase
Number of detected defects passed from previous phase
Number of defects passed from previous phase
Number of remaining defects passed from previous phase

Fig. 2. Sub-level structure of the Bayesian belief network model for the Design phase

III. QUANTIFICATION OF NODE PROBABILITY TABLE IN BBN MODEL

III.A. Expert Elicitation for BBN NPT Quantification

Since the BBN model focuses on NPP safety-related software, NPTs should represent the variability among the class of safety-related software, and their values need to be estimated. Generally, NPTs are derived through various methods such as direct expert judgment, estimation from safety software operation experience datasets, and representing the relationship between variables in terms of equations [4]. However, due to a lack of sufficient data from the collection of safety-related software, expert elicitation was used to quantify the NPTs for the prior and conditional probability distributions.

The relationship of the expert elicitations performed as a part of BBN model construction and model parameter estimation is shown in Fig. 2. In Fig. 2, black circle nodes denote the NPTs of “Defect density” and “Defect detection probability” nodes that are estimated by expert elicitation. The grey circle nodes denote the root nodes of the model. The development and V&V quality nodes are also estimated by expert elicitation, and later updated using the evidence on attribute nodes obtained from expert elicitation. Similarly, the “Size and Complexity” node was initially estimated in the elicitation and then a constant value can be used when applied to a specific nuclear safety software. The grey rectangle nodes, shown in Fig. 2, are the NPTs of the attribute nodes that were estimated by expert elicitation which can be also replaced with observed evidence from a specific nuclear safety software.

In this study, the elicitations were conducted by distributing background material and questionnaires to experts, collecting and analyzing the provided answers, resolving the provided comments. The BBN parameters were answered as either point estimates or three percentile estimates (5th, 50th, and 95th percentile) by the experts. The nodes elicited by point estimates include (1) prior distribution for the development quality node, (2) prior distribution for the V&V quality node, (3) prior distribution for the function point (FP), and (4) conditional probability distribution for the attributes, given the development and V&V quality. The nodes elicited by the 5th, 50th, and 95th percentile estimates include (1) the number of defects introduced, given the development quality, (2) the probability of defect removal introduced in current phase, given the V&V quality, and (3) the probability of defect removal passed from the previous phase, given the V&V quality.

A potential limitation of using expert opinions to estimate the quantities of NPTs in the BBN model comes from the disparity in the experts' opinions. That is, multiple experts may provide widely diverse answers, which should be treated in an integrated manner when estimating the NPTs in order to account for BBN parameter uncertainty representing the variability among the population of safety-related software. These variety of the experts' opinions must be modeled in a probabilistic manner; therefore, in this study, the variance associated with specified NPTs was represented as the tables of probabilistic distributions (distributed node probabilities) based on experts' answers on BBN nodes instead of discretized estimates to account for BBN parameter uncertainty. In order words, continuous univariate distributions, which gave the best fit for the empirical distribution sampled from the expert opinions based on Akaike information criterion (AIC) and Bayesian information criterion (BIC) measures [5], were used to define the NPTs of each BBN node; thus, represent the uncertainty associated with the NPTs subject to expert elicitation.

III.B. Bayesian update of the BBN NPTs using Evidence data

Considering that experts have different levels of knowledge and experience with nuclear safety-related software, a large variance in the NPT values can be caused by a significant diversity in the experts' NPT elicitation. One of the key features of the developed BBN model is that when the evidence for a BBN node is observed from the literature or the field experience data, the NPTs can be updated based on the Bayesian update method considering the conjugate prior family of probability distributions [6].

As an application of the Bayesian update method to update the NPT values, the parameters of the distribution representing the NPTs of defect density and defect detection probability at current phase were updated using reference data [7] and the Integrated Digital Protection System-Reactor Protection System (IDiPS-RPS) [8] and Loop Operating Control System (LOCS) [9] anomaly reports. Note that the NPTs of other BBN nodes in the model can be updated in a similar manner for any other observed evidence.

III.B.1. Bayesian update of the NPTs using evidence from reference data

The handbook data on the U.S. software development and V&V experience for the software defect potentials and the defect removal efficiency [7] were utilized to Bayesian update the NPTs estimated from the expert elicitation. In the analysis, it was assumed that the mean defect density was estimated in the expert elicitation process, and the software defect potentials in the handbook data was treated as observations, specifically Poisson likelihood mean. For simplicity, the number of inserted defects was estimated from the mean values of the Poisson process.

Table I shows important quality metrics such as defect potentials and defect removal efficiency of various software applications from the handbook data [7]. In this study, it was assumed that the "Capability maturity model (CMM) Level 5 joined with Six Sigma" represents *High* development or V&V quality, the "CMM Level 4" represents *Medium* development or V&V quality, and the "Spiral" represents *Low* development or V&V quality. Since the information was not available regarding how many cases were analyzed to derive the defect potentials and their removal efficiency in the handbook data, the number of observations was assumed to be one for simplicity.

The defect potential in Table I refers to the sum of possible errors in the software from five separate sources: errors in requirements, errors in design, errors in source code, errors in user documentation, and errors associated with bad fixes or secondary errors introduced while fixing a primary error.

Table II lists the overall distribution of software errors among the various categories of origin points across many industry segments from the handbook data [7]. For the Bayesian update of the defect density NPT in the BBN model, defect density evidence in each phase of the SDLC was derived as the product of the defect potential and defect origin percentage. Since the defect origin in the installation phase was not reported, the software defect origin percent for the Installation-and-Checkout phase was assumed to be 0%. The code bugs and bad fix bugs were assumed to be defects from the Implementation and Test phases, respectively. The document bugs were not considered in this study.

Based on the handbook data evidence, a Poisson distribution was then used to represent the likelihood for the defect density node (conjugate prior) derived from expert elicitation represented as a Gamma distribution.

TABLE I. Software types and defect characteristics

Software type	Defect potentials per function point	Defect removal efficiency
CMM5 + Six-sigma	4.80	98.00%
CMM4	6.00	93.00%
Spiral	6.50	85.00%

TABLE II. Software defect origin allocations

Types of bugs	Defect origin percentage
Requirements bugs	10%
Design bugs	25%
Code bugs	40%
Document bugs	15%
Bad fix bugs	10%
Total	100%

Regarding the distributed NPT for "Defect detection probability for defects introduced in current phase" node, the defect removal efficiency reported in the handbook data shown in Table I was used as the evidence for Bayesian update. Here, the defect removal efficiency refers to the percentage of defects removed before the delivery of the software to its users.

In this study, the defect removal efficiency reported in the reference was assumed to be the defect detection probability at each SDLC phase. In addition, the software defect removal efficiency in the reference data was treated as an observation, specifically the probability (p) of Bernoulli likelihood because the expert elicitation on the defect detection probability corresponds to the distribution of p. Therefore, Bernoulli distribution was used to represent the likelihood for the "Defect detection probability for defects introduced in current phase" node (conjugate prior), which was represented with a Beta distribution derived from expert elicitation.

III.B.2. Bayesian update of the NPTs using evidence from anomaly report data

In addition to the handbook data, limited V&V and testing results available for the development of two application systems, namely (1) the LOCS of the Advanced Test Reactor at Idaho National Laboratory [8] and (2) the prototype IDiPS-RPS developed by KAERI [9], were used for the Bayesian update of the distributed NPT of "Defect density" node derived from expert opinions to further reduce BBN parameter uncertainty. Tables III and IV show the defect estimates reported in the IDiPS-RPS and LOCS anomaly reports, respectively.

In case of the IDiPS-RPS, the number of defects detected at the Test phase was assumed to be the sum of defect estimates reported in the Integration phase and Validation phase. The number of defects in the Installation-and-Checkout phase was not considered since IDiPS-RPS had not yet been installed, thus the "Defect density" node was not updated with IDiPS-RPS data.

In the case of defect estimates for LOCS, the number of anomaly reports was assumed to be the number of defects detected at each SDLC phase. Based on the defect estimates data reported in software development anomaly reports of both systems, the defect density NPT in the BBN model was Bayesian updated considering the conjugate prior family of distributions.

TABLE III. Defect estimates from the IDiPS-RPS anomaly report

Software types	Phase	Defect estimates
BP	Requirements	6
	Design	16
	Implementation	3
BP/CP/ATIP/COM	Integration	4
BP/CP/ATIP/COM	Validation	4

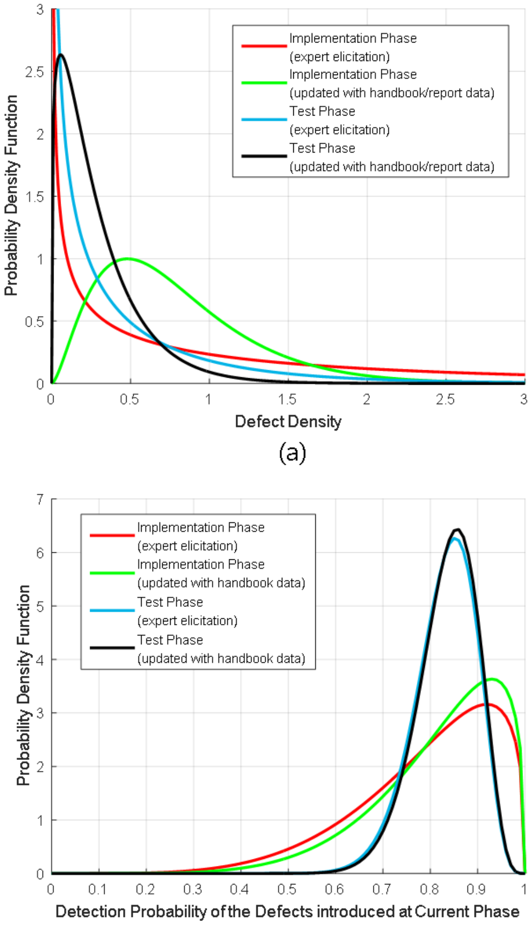
TABLE IV. LOCS defect estimates based on anomaly report

Phase	Defect estimates
Requirements	1
Design	2
Implementation	2
Test	2
Installation-and-Checkout	2
Total	9

III.B.3. Bayesian update result of the BBN NPTs using reference data

In this study, both the handbook data and two application systems' anomaly report data were treated as observations used for the Bayesian update of the NPTs of "Defect density" node to reduce parameter uncertainty from the diverse expert inputs. As a result, the uncertainty regarding defect density NPTs was decreased considerably over all phases, as shown in Fig. 3. Particularly, the defect density in the SDLC phases where a greater diversity in expert estimations was observed showed a larger effect from the Bayesian update using both handbook and anomaly data from the two trial systems. Further, the mean of defect density decreased in all phases. As an example, for the Test and Implementation phases, the mean decreased by 27.48% and 48.32%, and the standard deviation decreased by 53.11% and 77.32%, respectively. Figure 3-(a) shows the updated results for defect density at given High development quality in the Implementation and Test phases.

For the "Defect detection probability for defects introduced in current phase" node, handbook data on the defect removal efficiency of various software applications were used in the Bayesian update to reduce parameter uncertainty. Subsequently, the uncertainty regarding defect detection probability node also decreased over all phases while the mean of defect detection probability was slightly increased. For example, for the Test and Implementation phases, the mean increased by 0.54% and 3.05%, and the standard deviation decreased by 2.52% and 10.30%, respectively. Figure 3-(b) shows the updated result of defect detection probability at current phase at given *Medium* complexity and *High* V&V quality in the Implementation and Test phases using handbook data on U.S. software development experience.



(a)

(b)

Fig. 3. Bayesian-updated NPTs – (a) “Defect Density” NPT at given High development quality, and (b) “Defect detection probability for defects introduced in current phase” NPT at given Medium complexity and High V&V quality for Implementation and Test phases.

IV. BBN MODEL EVALUATION

Based on the NPTs estimated from expert elicitation and Bayesian updated using reference data, the number of defects remaining in a software for a typical NPP safety-related system was estimated using the BBN model. A typical digital protection software contains an input module that reads sensor measurements, an internal processing logic, such as comparison logic, to trigger the actuation unit when the sensor measurements exceed their set points, and an output unit (actuation unit) to produce a trip signal. By following the FP counting rules, one low level external input, one internal logic file, and one external output are counted, and an estimate of 50 FPs was considered to be the rough size of a typical digital protection software in this study. In addition, the states of the attributes of a typical digital protection software were reasonably considered as *Medium* in the BBN model as these software needs to pass the regulatory licensing review.

The model was evaluated using WinBUGS [10], which uses Markov chain Monte Carlo (MCMC) to solve the Bayesian inference problem posed in the model. Tables V and VI shows the evaluation results of the BBN parameters in case of the development and V&V quality in all phases being *Medium* quality given the number of FPs of 50.

TABLE V. BBN model parameter for a typical digital protection system

Phase	Defects introduced in current phase		Detection probability for defects passed from previous phase		Detection probability for defects introduced in current phase	
	Mean	SD	Mean	SD	Mean	SD
Requirement	19.71	35.9	-	-	0.79	0.16
Design	42.61	52.56	0.46	0.26	0.79	0.17
Implementatio n	49.45	56.96	0.48	0.25	0.84	0.15
Test	19.88	35.25	0.70	0.16	0.73	0.14
Installation/Ch eckout	12.63	29.35	0.70	0.19	0.80	0.14

TABLE VI. BBN model evaluation for a typical digital protection system

Phase	Detected defects passed from previous phase		Detected defects introduced in current phase		Defects remaining	
	Mean	SD	Mean	SD	Mean	SD
Requirement	19.71	35.9	-	-	0.79	0.16
Design	42.61	52.56	0.46	0.26	0.79	0.17
Implementatio n	49.45	56.96	0.48	0.25	0.84	0.15
Test	19.88	35.25	0.70	0.16	0.73	0.14
Installation/Ch eckout	12.63	29.35	0.70	0.19	0.80	0.14

V. CONCLUSIONS

In this study, a BBN model was developed which estimates the number of software defects for the NPP digital safety graded systems considering various software development life cycle characteristics. Since there are limited operating experience and data of NPP software-related protection systems, one of the challenges of the BBN model development is to obtain the data required for constructing the BBN nodes and quantifying the NPTs to assess the BBN model. In this study, the quantitative parameters in the BBN model were estimated by aggregating the expert opinions from various fields with hands-on experience in development and V&V of NPP safety graded software systems.

To effectively accommodate the variability of experts' opinion on the quality of NPP safety-related software when evaluating the BBN model, distributed node probabilities were used for the NPT modeling based on the expert elicitation for quantitative BBN nodes. Especially, the variety of the answers from the experts were aggregated and treated in an integrated manner by deriving the best fit probability distributions over the experts' opinions on each BBN node. The distributed NPTs represented as the tables of random variables defined by probability distributions were then used for the quantification of the BBN model.

However, due to the different levels of knowledge and experience among the experts, a diversity in the expert opinions can be observed for some nodes in the BBN model. Therefore, other sources of evidence, such as software development data from literature across many industries and the project data for two trial software developments, were used to Bayesian update the NPTs related to the defect density and defect detection probability nodes estimated from expert elicitation, to reduce the uncertainty caused by these diverse experts' opinions. As a result, parameter uncertainty in estimating the defect density and defect detection probability was significantly decreased in all SDLC phases. Based on the estimated NPTs, the number of defects for a typical digital protection software having the size of 50 FPs and having the Medium development and V&V qualities were quantified using the BBN model as a case study.

This study is expected to provide an insight on several aspects of BBN model quantification for nuclear safety-related software, including (1) the expert opinion elicitation and aggregation, (2) the quantification of the node probabilities using probability distribution, and (3) the Bayesian updating of the NPTs using available software development data. Especially, a framework that can effectively and systematically integrate different kinds of available source information to quantify BBN NPTs and reduce the NPT uncertainties is demonstrated.

Future research is recommended to further reduce the BBN parameter uncertainty regarding the NPT values and to update the BBN model result for software failure probability quantification with less uncertainty. By increasing the number of experts in the elicitation, uncertainty associated with the expert opinions on the BBN parameters is expected to decrease, and so the standard deviation for the number of defects remaining in each phase is expected to considerably decrease. In addition, as more evidence and observations from nuclear safety software operation experiences become available in the future, key parameters in the NPTs can be Bayesian updated to further reduce NPT uncertainties.

ACKNOWLEDGMENTS

This work was supported by Nuclear Research & Development Program of the National Research Foundation of Korea grant, funded by the Korean government, Ministry of Science, ICT & Future Planning (Grant Code: 2016M2B2A9909815, 2015M2A8A4021648).

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the U.S. Government. Neither the U.S. Government nor any agency thereof, nor any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility for any third party's use, or the results of such use, of any information, apparatus, product, or process disclosed in this report, or represents that its use by such third party would not infringe privately owned rights. The views expressed in this paper are not necessarily those of the U.S. Nuclear Regulatory Commission.

REFERENCES

1. H. G. Kang, and T. Y. Sung., "An analysis of safety-critical digital systems for risk-informed design," *Reliability Engineering & System Safety*, **78.3**, 307-314 (2002).
2. H. G. Kang, and S. C. Jang., "A quantitative study on risk issues in safety feature control system design in digitalized nuclear power plant," *Journal of nuclear science and technology*, **45.8**, 850-858 (2008).
3. U.S. NRC, *A BBN Model for the Probability of Software Failure on Demand, Round 2 Expert Opinion Elicitation*, ML16201A141 (2016).
4. C. A. Pollino, and B. T. Hart. "Developing Bayesian network models within a Risk Assessment framework", *Proceedings of the iEMSs Fourth Biennial Meeting: International Congress on Environmental Modelling and Software (iEMSs 2008).* Ottawa, Canada, July 7-10, Vol. 1, p. 372-379, International Environmental Modelling and Software Society (2008).

5. G. Schwarz, "Estimating the dimension of a model." *The annals of statistics*, **6.2**, 461-464, 1978.
6. A. Gelman, et al., *Bayesian data analysis* (3rd ed.), Chapman & Hall/CRC, FL, USA (2013).
7. C. Jones, *Applied software measurement: global analysis of productivity and quality* (3rd ed.), McGraw-Hill Education Group (2008).
8. *Verification and Validation (V&V) Report for 2A Loop Instrumentation and Operating Control System*, PLN-4681, Idaho National Laboratory (2014).
9. H. S. Eom, et al., *Reliability Assessment Method of Reactor Protection System Software by Using V&V based Bayesian Nets*, KAERI/TR-4092/2010, Korea Atomic Energy Research Institute (2010).
10. D. Spiegelhalter, et al., *WinBUGS user manual* (2003).