A Study on a Quantitative V&V for Safety-Critical Software

Heung-Seop Eom, Han Seong Son, Hyun-Gook Kang, Seung-Cheol Chang
Korea Atomic Energy Research Institute, ISA Div., P.O.Box 105, Yuseong Daejeon, ehs@kaeri.re.kr

1. Introduction

Verification and Validation (V&V) plays important role in assessing the safety-critical software embedded in the digital systems for a Nuclear Power Plant. A conventional V&V usually adopts a checklist method and its answers are mostly qualitative. There are some limitations to this conventional V&V method.

First, the difficulties in using the checklist method are [1]:

- Even for an acceptable software, some checklist questions will have negative answers. The checklist itself does not help to explain the reasons for drawing an overall positive conclusion in the presence of a few negative answers.
- The checklist does not help decide when enough issues have been examined to achieve a reasonable confidence in the software.
- The checklist method does not support a consideration of different kinds of information, such as software engineering measures.

Second, a difficulty comes from the qualitative form of the answers in the checklist method, which is:

- It is usually hard to know when sufficient evidence has been collected.

Finally a difficulty comes from a human expert's way of combining a great number of diverse evidence and inferring the conclusion, which is:

- Some of this evidence is qualitative and others are quantitative. Both are necessary to evaluate the quality of the software correctly. But, in general, the experts' way of combining the diverse evidence and performing an inference is usually informal and qualitative, which is hard to discuss and will eventually lead to a debate about the conclusion.

Our overall goal is to develop a systematic method that can obtain quantitative information of the software quality from the works of V&V. To achieve this goal and to solve the above-mentioned problems in the current V&V method, we studied a method that can combine qualitative and quantitative evidence, and can infer a conclusion in a formal and a quantitative way by using the benefits of BBN.

2. Bayesian Belief Nets (BBN)

BBN is a formalism for representing and analyzing models involving an uncertainty. Nowadays a number of efficient tools for BBN modeling are available and BBN has become an expanding technology in many areas such as medical, military, financial, and the safety/reliability analysis of complicated systems.

Here are some advantages of BBN in modeling the works of V&V.

- Ability to forecast the target events with some missing or partial data
- Rigorous mathematical semantics for the model
- Intuitive graphical format $\Rightarrow$ easier to understand chains of complex and seemingly contradictory reasoning
- Explicit modeling of 'ignorance' and uncertainty in the estimates, as well as the cause-effect relationships
- Makes explicit those assumptions that were previously hidden – visibility and auditability to the decision making process
- Allow us to employ both subjective probabilities and probabilities based on statistical data in a unified framework, thus in turn they allow us to combine qualitative and quantitative measures in making the inferences.
- Can enable us to make inferences which are much more precise than just using expressions such as "very likely", "unlikely", "slightly increases" ... and so on.

3. BBN based quantitative V&V

We constructed a BBN model which represents the works of V&V in the software requirement phase. The purpose of the model is to assess the quality of the software requirement specification (SRS) in a quantitative way. The basic documents which were used to develop the model are (i) Procedures for V&V [3], (ii) V&V report [4], and (iii) Software Development Plan [5].

The proposed method relies on BBN to combine all the variables relevant to the quality of SRS, and to propagate consistently the impact of these variables on the probabilities of the uncertain outcomes (in this example, the quality of SRS). The variables in our model were mostly identified from V&V procedures [3] and SDP [5]. A summary of the variables is:

- 14 properties of software requirement specification - Accuracy, functionality, reliability, robustness, safety, security, timing, completeness, consistency, correctness, style, traceability, unambiguity, verifiability.
- Questions belonging to the above 14 properties: Each property has a checklist and the checklist has several questions.
- The Quality of the development process, V&V process, and complexity

Fig. 1 is the top-level BBN graph for the assessment of SRS quality. This net shows the abstracted variables of the model.

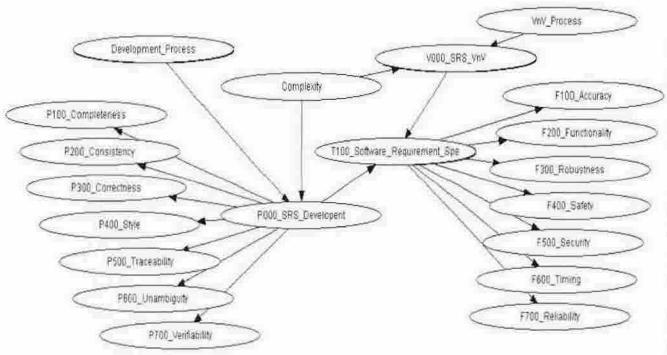

Figure 1. BBN model for requirement phase

The abstracted nodes in Fig. 1 have child nodes and the child nodes were mostly derived from the checklist of the abstracted node. The results of V&V are used as input for the child nodes. Fig. 2 is an example of a sub-net of the node "Consistency" which appeared in Fig. 1.

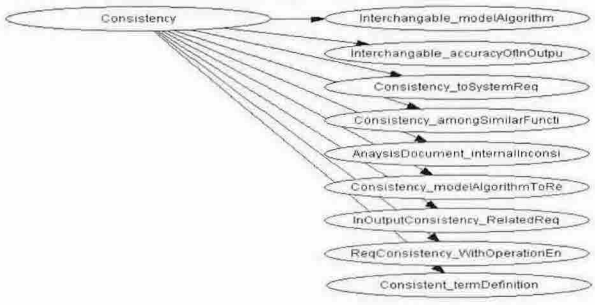

Figure 2. BBN graph of node "Consistency"

A total of 160 nodes and their node probability table (NPT) were developed. The target node of the model is "T100_Software_Requirement_Specificaion" which has two states "acceptable" and "unacceptable". The states of this node mean an acceptable degree of SRS. All the nodes converted from the checklists have two states ("yes" and "no"). The probabilities of the NPT in the model were assigned by a V&V expert's judgment.

The aim of BBN is to calculate the value of the target node based on the observed evidence. Therefore it is necessary to obtain the value of all the observable nodes. These values can be obtained in a quantitative form or a qualitative form. Then the inputs of BBN must be expressed in a quantitative form. Therefore the qualitative evidence needs some conversion from the qualitative form into the quantitative form. V&V experts made the necessary inputs for the BBN model. In assigning a quantitative value for a given question Sherman Kent's rating scale was used.

Finally the quality of SRS can be calculated by using the evidence obtained in the previous step. This evidence is the input of the observable node in the net. The output is the probability distribution of all the nodes in the net.

Since the computation of BBN is a very complicate process one should use a proper computer tool for a realistic BBN model. We used the HUGIN (BBN modeling software) for the calculation.

The calculated probability of "unacceptable" state of the target node in our case study was high. The reason is that the current SRS is still under development and only 30% of all the evidence was positive. The tentative success value of the state "acceptable" in the target node is over 95%, and this value means the completion of V&V activity in the requirement phase.

4. Summary & Conclusion

We studied a systematic method that can obtain the quantitative results of V&V. The method was constructed by utilizing BBN which can combine the qualitative and the quantitative evidence relevant to the quality of the safety-critical software and can infer a conclusion in a formal and a quantitative way. The case study was performed by applying the method for assessing the quality of SRS of the safety-critical software that will be embedded in a reactor protection system. The calculation result of the BBN model showed that its conclusion is mostly similar to those of a V&V expert for a given input data set. The model will provide quantitative information that can assist human experts in various V&V activities.

REFERENCES

[1] Johnson, G., et al, 2000. Bayesian Belief Network Based Review of Software Design Documents, NIPC & HMIT 2000

[2] Gran, B.A. Dahll, G. Estimating dependability of programmable systems using bayesian belief nets, HWR-627, OECD Halden Project, 2000.

[3] HanSeong, S., et al, 2003. V&V Procedure for Software Requirement Specification for Reactor Protection System, KNICS-RPS-(SRS)-SVP121, KAERI KNICS, 2003.

[4] HanSeong, S., 2004. V&V Validation Reports for Software Requirement Specification for Reactor Protection System, KNICS-RPS-(SRS)-SVR121, KAERI KNICS, 2004.

[5] Du-Hwan, K., et al, 2001. Software Development Plan for Engineering Safety Features, KNICS-ESF-SDP101, KAERI KNICS, 2001.