Analysis on the Important Indicators in Bayesian Belief Network Model for RPS Software Reliability Assessment

2017 International Topical Meeting on Probabilistic Safety Assessment and Analysis (PSA 2017), September 24-28, 2017, Pittsburgh, PA, USA

Sang Hun Lee¹, Tsong-Lun Chu², Athi Varuttamaseni², Meng Yue², Ming Li³, Jaehyun Cho⁴, Seung Jun Lee⁵*, Hyun Gook Kang¹*

¹ Department of Mechanical, Aerospace, and Nuclear Engineering, Rensselaer Polytechnic Institute, Troy, NY, USA
² Brookhaven National Laboratory, Upton, NY, USA
³ U.S. Nuclear Regulatory Commission, Washington, DC, USA
⁴ Korea Atomic Energy Research Institute, Daejeon, ROK
⁵ School of Mechanical and Nuclear Engineering, Ulsan National Institute of Science and Technology, Ulsan, ROK
*Corresponding author: kangh6@rpi.edu, sjlee420@unist.ac.kr
Introduction

Previous work: < BBN Model for Software Reliability Quantification >

Application of BBN Framework on SDLC Optimization

- Contribution analysis of SW quality indicators on SW residual defects
  - Analysis of important SW dev. activities (attributes) on SW SDLC quality
  - Analysis of SDLC Dev./V&V quality on SW residual defects

- Risk-Cost model for optimized software SDLC quality using BBN model
  - 1) Cost for higher SW SDLC development/V&V quality
  - 2) Cost for fixing detected SW defects during SDLC
  - 3) Cost of NPP accident due to SW residual defects

Conclusion
The developed BBN model\cite{1} captures NPP safety-related SDLC quality and software-self characteristics (e.g. software size) in estimating the number of software defects.

- Compared to previous BBN models\cite{2,3}, the proposed model can be applied to analyze important SDLC phases and attributes which largely affects the final SW defects.

In this study, the BBN model was used to analyze important indicators to SW residual faults, and further optimize the SW Dev./V&V qualities considering its risk & cost.
Proposed BBN framework captures the causal relationships among SW Dev. activities (Attributes), SW Dev./V&V quality, and the number of SW defects.

Model can be used to identify which attributes/SDLC phases have larger contribution on the SW defects, and further improve SDLC activities before/after SW deployment.

\( \Rightarrow 1 \) The evidence (**quality of software attributes**) is introduced to BBN model to infer the SW development or V&V quality.

\( \Rightarrow 2 \) The number of defects remaining in each phase is modeled as a function of **Development** and **V&V quality** in each phase.

Figure 2. Overview of the BBN model – Design phase
SW attribute, representing a collection of SW Dev. activities associated with each SDLC phase, serves as an indirect indicator of the SW Dev. and V&V quality.

To identify the attributes as strong or weak indicators for development quality or V&V quality, an indication measure \( I \) is proposed as:

\[
I = P(A_H|Q_H)P(Q_H) + P(A_M|Q_M)P(Q_M) + P(A_L|Q_L)P(Q_L)
\]

Table 1. Conditional probabilities of SW attribute quality given development quality and indication measure \( I \) in Requirement – Development phase

<table>
<thead>
<tr>
<th>Attribute Quality</th>
<th>Development Quality Level</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>High</td>
</tr>
<tr>
<td>System/Software Qualification Test Plan Generation</td>
<td>0.72</td>
</tr>
<tr>
<td>Development of Software Requirements Specifications</td>
<td>0.72</td>
</tr>
<tr>
<td>Development of a Concept Documentation</td>
<td>0.68</td>
</tr>
<tr>
<td>Review and Audit - Requirements Phase</td>
<td>0.59</td>
</tr>
<tr>
<td>Risk Analysis - Requirements Phase</td>
<td>0.54</td>
</tr>
<tr>
<td>Configuration Management - Requirements Phase</td>
<td>0.59</td>
</tr>
</tbody>
</table>

Expert elicitation on conditional NPT of “System/Software Acceptance Test Plan Gen.” node

Figure 3. Attributes nodes for Development quality in Requirement phase
**Contribution analysis of SDLC quality on SW defects**

- SW Dev./V&V quality determines the number of SW defects inserted, and the detection probability of the defects introduced in both current and previous phases.

- In order to effectively reduce the number of defects in safety software, it is important to analyze which SDLC phase has the biggest contribution to final SW defects.

→ **Dev. Quality** in each SDLC phase affects to the number of defects introduced in that phase.

→ **V&V Quality** in each SDLC phase affects to the number of defects detected in that phase.

---

Figure 4. Sub-level structure of the BBN model - Design Phase
Figure 5. High-level structure of the BBN model
To analyze the effect of SW quality on the SW residual defects, an evidence on SDLC Dev./V&V quality was directly inserted to the nodes in BBN model.

- As a case study, an upgrade of Dev./V&V quality from Medium to High is considered.

The result showed that SW Dev./V&V quality in later phases has greater effect on the final defects since defects in early phases can be detected throughout later phases.

**Table 2. Number of defects remaining in the current phase and final residual defects**

<table>
<thead>
<tr>
<th>Phase</th>
<th>Condition**</th>
<th>Number of defects remaining in each phase</th>
<th>Final number of defects remaining</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Mean</td>
<td>△Mean*</td>
</tr>
<tr>
<td>Requirements</td>
<td>Dev. Quality = M → H</td>
<td>3.21</td>
<td>-21.23%</td>
</tr>
<tr>
<td>V&amp;V Quality = M → H</td>
<td></td>
<td>2.60</td>
<td>-36.19%</td>
</tr>
<tr>
<td>Design</td>
<td>Dev. Quality = M → H</td>
<td>8.92</td>
<td>-18.88%</td>
</tr>
<tr>
<td>V&amp;V Quality = M → H</td>
<td></td>
<td>7.46</td>
<td>-32.23%</td>
</tr>
<tr>
<td>Implementation</td>
<td>Dev. Quality = M → H</td>
<td>11.99</td>
<td>-12.55%</td>
</tr>
<tr>
<td>V&amp;V Quality = M → H</td>
<td></td>
<td>8.82</td>
<td>-35.66%</td>
</tr>
<tr>
<td>Test</td>
<td>Dev. Quality = M → H</td>
<td>5.27</td>
<td>-44.19%</td>
</tr>
<tr>
<td>V&amp;V Quality = M → H</td>
<td></td>
<td>4.64</td>
<td>-50.92%</td>
</tr>
<tr>
<td>Installation/Checkout</td>
<td>Dev. Quality = M → H</td>
<td>4.53</td>
<td>-14.94%</td>
</tr>
<tr>
<td>V&amp;V Quality = M → H</td>
<td></td>
<td>1.88</td>
<td>-64.62%</td>
</tr>
</tbody>
</table>

* △Mean: Mean compared to that of Medium Dev./V&V quality in all SDLC phases
** M: Medium quality; H: High quality
Since safety-critical/-related software is developed based on the standard SW SDLC, a systematic framework to address effective software development plan is needed[4].

In the framework, the software quality upgrade cost to reduce software defects along with the plant risk due to software failure (e.g. CDF, LERF) must be considered.
Risk-Cost model for optimized SDLC quality control plan

The objective function ($\xi$) can be used to analyze the efficiency of the SW quality control plan and validate its applicability considering both risk and cost.

- 1) $\xi_a$: Cost of NPP accident due to SW residual defects
- 2) $\xi_f$: Cost for fixing detected SW defects during SDLC
- 3) $\xi_u$: Cost for higher SW SDLC development/V&V quality

$$\xi = \xi_u + \xi_f + \xi_a$$
Estimation of risk and costs related to SW SDLC quality

- Objective function: \( \xi = \xi_u + \xi_f + \xi_a \)

  1) \( \xi_u \): Cost for achieving higher SW quality
     - \( \xi_u = \sum_{SDLC} (N_{att} \times C_{att,u}) \)
     - \( N_{att} \): Number of software attributes,
     - \( C_{att,u} \): Cost of software attribute quality upgrade (Medium → High)

  2) \( \xi_f \): Software defect detection/fixing cost
     - \( \xi_f = \sum_{SDLC} (N_{det,prev} \times C_{fix,prev} + N_{det,curr} \times C_{fix,curr}) \)
     - \( N_{det,prev} \): Number of detected defects passed from previous phases
     - \( N_{det,curr} \): Number of detected defects introduced in current phases
     - \( C_{fix,prev}, C_{fix,curr} \): Cost of fixing detected defects from previous/current SDLC phase

  3) \( \xi_a \): Cost of NPP accident (ATWS due to SW failure)
     - \( \xi_a = R_A \times CDF + C_{ext} \times LERF \)
     - \( CDF, LERF \sim SFP, SFP = N_{final} \times FSD \)
     - \( R_A \): Monetary loss for core damage (electricity loss cost)
     - \( C_{ext} \): External cost of accident
     - \( CDF \): Core Damage frequency due to DPPS SW failure
     - \( SFP \): Software Failure Probability
     - \( N_{final} \): Final SW defects remaining at the last SDLC phase
     - \( FSD \): Fault Size Distribution

*Figure 7. Attributes nodes for Development quality in Requirement phase*

*Figure 8. Sub-level BBN model in Requirement phase*
Estimation of SW defect fixing cost ($\xi_f$)

- By directly inserting the evidence to the SW quality nodes in BBN model, the number of defect detected at each SDLC phase can be assessed.

- Assuming that the detected defects are all fixed during the SW SDLC process, the SW defect fixing cost is estimated based on the number of detected defect, as Fig. 9.

### Table 3. Cost factors for fixing detected SW Defects at each SDLC phase

<table>
<thead>
<tr>
<th>Defect fixing cost (in case study, $C_1 = $1,000)</th>
<th>Detected Defects (current phase)</th>
<th>$C_{\text{fix,curr}}$</th>
<th>Detected Defects (previous phase)</th>
<th>$C_{\text{fix,prev}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Req – Dev</td>
<td>$C_1$</td>
<td>Req – V&amp;V</td>
<td>$1^*C_1$</td>
<td></td>
</tr>
<tr>
<td>Dsg – Dev</td>
<td>$C_1$</td>
<td>Dsg – V&amp;V</td>
<td>$5^*C_1$</td>
<td></td>
</tr>
<tr>
<td>Imp – Dev</td>
<td>$C_1$</td>
<td>Imp – V&amp;V</td>
<td>$10^*C_1$</td>
<td></td>
</tr>
<tr>
<td>Tst – Dev</td>
<td>$C_1$</td>
<td>Tst – V&amp;V</td>
<td>$50^*C_1$</td>
<td></td>
</tr>
<tr>
<td>Ins – Dev</td>
<td>$C_1$</td>
<td>Ins – V&amp;V</td>
<td>$100^*C_1$</td>
<td></td>
</tr>
</tbody>
</table>

### Table 5. Number of Defect detected for various SW SDLC quality improvement plan

<table>
<thead>
<tr>
<th>Attribute upgrade case</th>
<th>Detected Defects (from prev. phase)</th>
<th>Detected Defects (from current phase)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base case (All Medium)</td>
<td>Dsg</td>
<td>1.83</td>
</tr>
<tr>
<td>Req – High</td>
<td>Dsg</td>
<td>0.99</td>
</tr>
<tr>
<td>Dsg – High</td>
<td>Dsg</td>
<td>2.29</td>
</tr>
<tr>
<td>Imp – High</td>
<td>Dsg</td>
<td>1.84</td>
</tr>
<tr>
<td>Tst – High</td>
<td>Dsg</td>
<td>1.84</td>
</tr>
<tr>
<td>Ins – High</td>
<td>Dsg</td>
<td>1.82</td>
</tr>
<tr>
<td>All high</td>
<td>Dsg</td>
<td>1.21</td>
</tr>
</tbody>
</table>

Figure 8. Cost of Software Defect Detection/Fixing
Estimation of risk and costs related to SW SDLC quality

- **Objective function:** \( \xi = \xi_u + \xi_f + \xi_a \)

  1) \( \xi_u \): Cost for achieving higher SW quality
  - \( \xi_u = \sum_{SDLC} (N_{att} \times C_{att,u}) \)
  - \( N_{att} \): Number of software attributes,
  - \( C_{att,u} \): Cost of software attribute quality upgrade (Medium → High)

  2) \( \xi_f \): Software defect detection/fixing cost
  - \( \xi_f = \sum_{SDLC} (N_{det,prev} \times C_{fix,prev} + N_{det,curr} \times C_{fix,curr}) \)
  - \( N_{det,prev} \): Number of detected defects passed from previous phases
  - \( N_{det,curr} \): Number of detected defects introduced in current phases
  - \( C_{fix,prev}, C_{fix,curr} \): Cost of fixing detected defects from previous/current SDLC phase

  3) \( \xi_a \): Cost of NPP accident (due to SW residual faults)
  - \( \xi_a = R_A \times CDF + C_{ext} \times LERF \)
  - \( CDF, LERF \sim SFP \), \( SFP = N_{final} \times FSD \)
  - \( R_A \): Monetary loss for core damage (electricity loss cost)
  - \( C_{ext} \): External cost of accident
  - \( CDF \): Core Damage frequency due to DPPS SW failure
  - \( SFP \): Software Failure Probability
  - \( N_{final} \): Final SW defects remaining at the last SDLC phase
  - \( FSD \): Fault Size Distribution

\[ P(Q_H) \]
\[ P(Q_H) \]
\[ N_{final} \]
Estimation of CDF caused by SW residual defects ($\xi_a$)

- Higher SW Dev./V&V quality leads to a decrease of the number of final SW residual defects, which results in lower SW failure probability from a PRA perspective.

- The number of software defects estimated from the BBN node was converted to the software failure probability, and further reflected in the NPP PSA model.

Figure 9. Software failure in NPP PSA model[7] (resulting in failure of Rx Trip signal generation - ATWS)

Figure 10. Software failure probability for various SW quality improvement plan cases
Total risk-cost for various SDLC quality control plans ($\xi$)

- Assuming the SW quality cost factor to be Table 4, the objective function ($\xi$) was estimated considering both software quality control cost and NPP accident cost.

- In the case study, the result showed that the upgrade of SDLC quality in Test phase reduces the total cost the most compared to the base case (all Medium SDLC quality).

Table 4. Cost factors as a case study

<table>
<thead>
<tr>
<th>Factors</th>
<th>Cost($)</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_1$</td>
<td>$1,000</td>
<td>Fixing cost per detected Defect</td>
</tr>
<tr>
<td>$C_2$</td>
<td>$10,000</td>
<td>Cost of Attribute quality upgrade</td>
</tr>
<tr>
<td>$C_{elec}$</td>
<td>0.06$/kWh</td>
<td>Electricity Cost</td>
</tr>
<tr>
<td>$C_p$</td>
<td>0.9</td>
<td>Capacity Factor</td>
</tr>
<tr>
<td>$C_{ext}$</td>
<td>$1.37E+12</td>
<td>C_{ext} = 17,593ME$ \times M</td>
</tr>
<tr>
<td>$M$</td>
<td>78[8]</td>
<td>Risk aversion factor</td>
</tr>
</tbody>
</table>

$$\xi = \xi_u + \xi_f + \xi_a$$

$$\xi_u = \sum_{SDLC} (N_{att} \times C_{att,u})$$

$$\xi_f = \sum_{SDLC} (N_{det,prev} \times C_{fix,prev} + N_{det,curr} \times C_{fix,curr})$$

$$\xi_a = R_A \times CDF + C_{ext} \times LERF$$

$$R_A = C_p \times (1000 MW) \times (8760 \frac{hrs}{yr}) \times (30 yrs) \times (C_{elec} \frac{\$}{kWh})$$

Figure 11. Total cost ($\xi$) for various SW SDLC quality improvement plan (at $M = 78$)
Considering different public risk aversion towards NPP accident, the objective function ($\xi$) was estimated for various risk aversion factor ($M$) of $20^{[10]} - 150^{[11]}$.

$$\xi_a = R_A \ast CDF + C_{ext} \ast LERF$$

$$R_A = C_p \ast (1000 \text{MW}) \ast \left(8760 \frac{\text{hrs}}{\text{yr}}\right) \ast \left(30 \text{ yrs}\right) \ast \left(C_{elec} \frac{\$}{\text{kWh}}\right)\quad C_{ext} = 17,593MEuro^{[8]} \ast M$$
Conclusion

- In this study, a proposed BBN model was used to identify important SW quality indicators on SW defects and analyze effective SW quality improvement plans.

  - SW attribute at each SDLC phase as strong/weak indicators for SW Dev./V&V quality was identified using indication measure ($I$).

  - As SW Dev./V&V quality affects on the number of final SW defects, the contribution of quality nodes at each SDLC phase was analyzed.

  - A framework for SDLC quality optimization ($\xi = \xi_u + \xi_f + \xi_a$) was proposed considering both the cost from software risk and software quality control costs.

    - The proposed risk-cost model for NPP safety software can be used to optimize the SDLC process before/after its deployment, and quantify the risk from software defects.

- Considerations for future work

  - Use of proposed SW Dev. optimization framework to NPP digital I&C application
    - Use of application-specific evidence to Bayesian update the NPTs
    - Investigation on application-specific SW Dev./V&V quality and cost factors
Thank you for your attention

Q&A
Reference

Appendix I.

Normalized Cost-to-Fix Estimates [5]

<table>
<thead>
<tr>
<th>Source</th>
<th>Requirement</th>
<th>Design</th>
<th>Code</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Boehm, 1981]</td>
<td>1</td>
<td>5</td>
<td>10</td>
<td>50</td>
</tr>
<tr>
<td>[Hoffman, 2001]</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>37</td>
</tr>
<tr>
<td>[Cigital, 2003]</td>
<td>1</td>
<td>3</td>
<td>7</td>
<td>51</td>
</tr>
<tr>
<td>[Rothman, 2000]</td>
<td>-</td>
<td>5</td>
<td>33</td>
<td>75</td>
</tr>
<tr>
<td>[Rothman, 2000] Case B</td>
<td>-</td>
<td>-</td>
<td>10</td>
<td>40</td>
</tr>
<tr>
<td>[Rothman, 2000] Case C</td>
<td>-</td>
<td>-</td>
<td>10</td>
<td>40</td>
</tr>
<tr>
<td>[Rothman, 2000]</td>
<td>1</td>
<td>20</td>
<td>45</td>
<td>250</td>
</tr>
<tr>
<td>[Pavlina, 2003]</td>
<td>1</td>
<td>10</td>
<td>100</td>
<td>1000</td>
</tr>
<tr>
<td>[Mcgibbon, 2003]</td>
<td>-</td>
<td>5</td>
<td>-</td>
<td>50</td>
</tr>
<tr>
<td>Mean</td>
<td>1</td>
<td>7.3</td>
<td>25.6</td>
<td>177</td>
</tr>
<tr>
<td>Median</td>
<td>1</td>
<td>5</td>
<td>10</td>
<td>50.5</td>
</tr>
</tbody>
</table>

Cost ranges of errors per life cycle phase from aerospace industry [5]

<table>
<thead>
<tr>
<th>Phase</th>
<th>All Errors</th>
<th>1 Deviation</th>
<th>2 Deviations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements</td>
<td>$667 - $209,504</td>
<td>$667 - $40,038</td>
<td>$667 - $59,022</td>
</tr>
<tr>
<td>Design</td>
<td>$1,880 - $306,036</td>
<td>$1,880 - $131,104</td>
<td>$1,880 - $192,382</td>
</tr>
<tr>
<td>Build</td>
<td>$54,830 - $1,511,365</td>
<td>$54,830 - $483,694</td>
<td>$54,830 - $483,694</td>
</tr>
<tr>
<td>Test</td>
<td>$50,046 - $12,383,000</td>
<td>$50,046 - $1,941,787</td>
<td>$50,046 - $2,926,000</td>
</tr>
<tr>
<td>Operations</td>
<td>$480,214 - $36,739,000</td>
<td>$480,214 - $4,553,577</td>
<td>$480,214 - $9,401,506</td>
</tr>
</tbody>
</table>
The quality nodes in the BBN model are qualitatively modeled with three states—High, Medium, and Low—with each state representing the overall quality of the activities, defined as follows:

- **High**: State corresponding to the quality of the software development by a high-maturity company rigorously following established standards, and implementing additional measures to significantly improve the quality of the software.

- **Medium**: State representing the quality of a software development in which all required activities for safety-related systems are completed.

- **Low**: State representing the quality of a software development in which required activities for safety-related systems are not completed.
the attribute nodes are modeled with the three states defined as below:

- **High**: In addition to satisfactorily carrying out the required activities, additional activities were undertaken that are expected to significantly improve the quality of the work, and enhance the software’s reliability.

- **Medium**: All required (or equivalent) activities were satisfactorily carried out.

- **Low**: Some of the required activities were not carried out satisfactorily.