Mat-2.4177 Seminar on Case Studies in Operations Research

Tools for analyzing epistemic uncertainties in probabilistic risk analysis

Project plan

Client: VTT (Technical Research Centre of Finland)

Project team:
Janne Laitonen (Project manager)
Henri Losoi
Markus Losoi
Kimi Ylilammi

Version 1	To the team members for comments, 10.2.2013
Version 2	To Holmberg for comments, 15.2.2013
Version 3	Comments included, to the course assistant, opponent team, Toppila and Holmberg, 19.2.2013
Version 4	Comments by Salo and the opponents included, 3.3.2013 Approved by VTT, 22.3.2013

1. Backgrounds

Probabilistic risk analysis (PRA), also called probabilistic safety analysis (PSA) or quantitative risk analysis (QRA), is a systematic methodology for assessing the risks of technical systems and is being widely applied to many sectors e.g. transport, energy, construction, chemical processing, aerospace, military and even project planning. In many of these areas PRA techniques have been adopted to regulatory framework by relevant authorities. PRA has several applications, e.g., identifying system weaknesses for modifications, allocating the recourses for maintenance activities, prioritizing the targets for inspections, or optimizing the test intervals. [1,2]

In PRA, usually three basic questions are asked: What can go wrong, how likely that is, and what are the consequences? This leads to triplet definition of (engineering) risk which is quantified by *scenario, probability and consequence*. In order to assess the failure probability for a system, the failure logic is modeled, e.g., using *fault and event trees*. These trees model the failure and sequence propagation in the system starting from the system components that are the *basic events* in the model (i.e., the leaves of a fault tree). The events that may cause the system to fail are called *initiating events* (e.g., loss of electric power) which may result from various *hazards,* e.g., fires, floods, harsh weather conditions, or seismic activity. The solution of the model can be presented by *minimal cut sets* which are the minimal combinations of events that lead to system failure (so-called *TOP-event*). After solving the Boolean logic for the TOP-event, the probabilities and frequencies of the basic and initiating events are used for quantifying the probability (or frequency) of the system failure, e.g., the frequency estimate for a nuclear accident. In addition, the importance of the basic events can then be analyzed using various *risk importance measures* such as Fussell-Vesely and Birnbaum.

As suggested above, PRA is about studying uncertainties. In general, or at least in Bayesian sense, uncertainty can be presented by *aleatoric and epistemic uncertainty,* i.e., uncertainty due to randomness and due to lack of knowledge, respectively. Aleatoric uncertainty can be quantified, e.g., by a probability estimate and epistemic uncertainty represents the uncertainty on the value of this estimate.

In this project, a method for analyzing the epistemic uncertainties in PRA model parameters is implemented using interval probabilities. The method is based on framework presented by Toppila and Salo [3] where they study prioritization of events under interval-valued probabilities. Their work has been supported by the Nuclear Waste Management Fund provided by the Finnish National Nuclear Power Plant Research Programme SAFIR 2014.

2. Objectives and requirements of the project

The project has been initiated by VTT (Technical Research Centre of Finland) giving the following objectives:

- Get an understanding about epistemic uncertainty and its modeling with interval probabilities.
- Implement an open source Matlab-package compatible with Finnish PRA software FinPSA for analyzing epistemic uncertainties expressed though interval probabilities.
- Develop the software following good software development practices including the following documents: requirements specification, software design specification, testing plan and quality assurance plan. Software validation report including test results must be produced in the end.

The implementation is based on the method presented in [3] thus no selections were needed among different models or methods. In addition, further development of the methodology was given lower importance while high priority was given to high quality documentation and testing. Especially, the quality and completeness of the requirements specification was considered important. The team also has to test how the model complexity affects the calculation time. Due to the extensive documentation required the length of the final report shall be adjusted accordingly and the required documents are considered as a part of the report. All the documents will be public and written in English.

3. Methods

The method for analyzing epistemic uncertainties is based on the framework presented by Toppila and Salo [3], where the uncertainty about the basic event probabilities is expressed by intervals. These intervals indicate the knowledge - or the lack of it - on the true values of the model parameters and contain the belief about the plausible range in which the probabilities may locate. Since there is epistemic uncertainty in the event probabilities, the risk importance measures are impacted by this uncertainty. In the method, dominance relations for the importance measures are established. One event is said to dominate another if its risk importance measure is at least as high for all event probabilities that are within their respective intervals and strictly higher for some probabilities.

As the fundamental objective of the project is to implement a software following good development practices, special attention shall be paid on documentation, updating it and software version control. VTT has provided an SVN repository for the team to be used for the software versioning. Google Drive folder is used for saving and sharing the project documentation, references, testing data etc.

4. Resources and communication

The project team consists of four persons: Janne Laitonen (project manager), Henri Losoi, Markus Losoi, and Kimi Ylilammi. The steering group consists of Jan-Erik Holmberg (VTT), Ahti Salo (Aalto) and Antti Toppila (Aalto). Also the opponent team gives feedback and possible ideas for the project.

The team will have regular meetings decided in agreement with the team or called by the project manager. The team communicates mainly using e-mails, phone calls and sharing documentation in Google Drive. The communication with the project stakeholders is done mainly with e-mails by the project manager. Antti Toppila will get a carbon copy concerning the communication with Jan-Erik Holmberg.

5. Tasks, responsibilities and deliverables

The tasks, main responsibilities and the deliverables for each task are shown below.

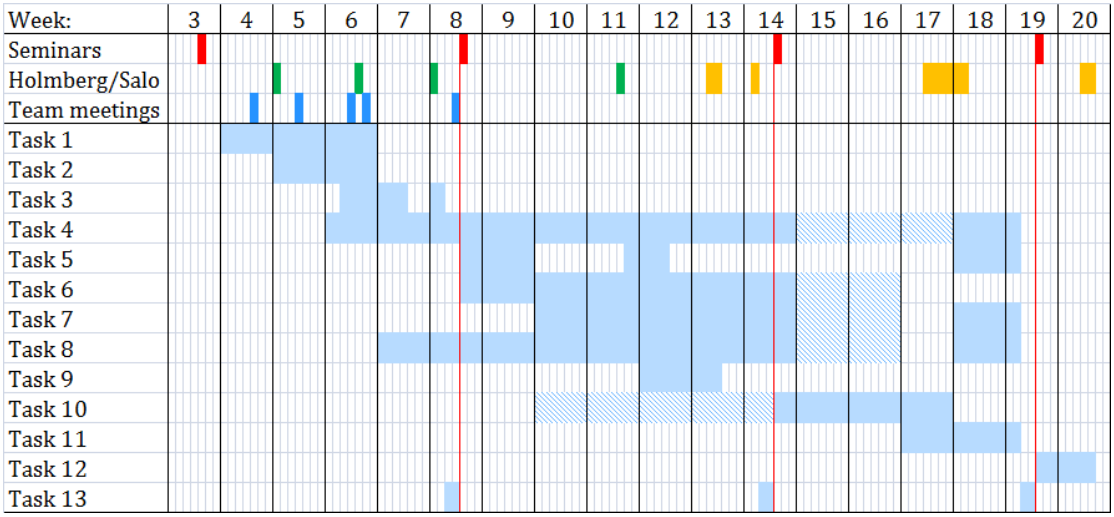
Task 1: Understanding the backgrounds and the theory **Description:** Proper understanding of PRA-methods is required for understanding the problem. Material consists, e.g., [1, 3] as well as risk analysis –course website material. **Responsibility:** All **Deliverables:** No explicit documents required, general understanding is reflected in other documents.
Task 2: Define the objectives of the project **Description:** The goals of the project will be defined in conjunction with Holmberg, Salo and Toppila. **Responsibility:** All **Deliverables:** The required objectives for the project
Task 3: Project plan & presentation **Description:** Planning the tasks, responsibilities and timetable for the project. Writing the report and presentation with required background information. The project plan will be used as a basis for quality assurance plan. **Responsibility:** Janne **Deliverables:** The project plan and seminar slides
Task 4: Requirement specification **Description:** Writing the requirement specification for the implemented method using the template by VTT. The scope of the document will be wider than the implemented Matlab-package or [3]. **Responsibility:** Markus, Kimi assists **Deliverables:** Requirement specification document usable for future development at VTT.

Task 5: Quality assurance plan **Description:** QA-plan that sets the quality requirements for the project. Template by VTT will be used. Project plan will be used as a basis for this document. **Responsibility:** Janne **Deliverables:** QA-plan document
Task 6: Implementation of Matlab-package **Description:** Programming of a Matlab-package that is FinPSA compatible (data from FinPSA can be used as an input) and that can calculate the dominance relations as described in [3]. **Responsibility:** Kimi, Markus assists **Deliverables:** Matlab-package for calculating dominance relations
Task 7: Software design specification **Description:** The implemented software (i.e., Matlab-package) will be documented using the template by VTT. **Responsibility:** Kimi, Markus assists **Deliverables:** Software design document
Task 8: Testing plan **Description:** The testing plan will be written using the template by VTT and the implemented software shall be tested accordingly. **Responsibility:** Henri, Janne assists and provides test data **Deliverables:** Testing plan document
Task 9: Interim report & presentation **Description:** The report will describe the status of the project and an updated project plan. **Responsibility:** Janne **Deliverables:** Interim report and seminar slides
Task 10: Software testing and reporting of the test results **Description:** The implemented software will be tested according to the test plan. First, simple models shall be used but finally more complex cases shall be used to test the calculation time and the limits of the method. **Responsibility:** All **Deliverables:** The test results will be included in the final report
Task 11: Final report & presentation **Description:** Final report that summarizes the project and the results. The documents described above will supplement the report and will be updated at the end. **Responsibility:** All **Deliverables:** Final reports and other documentation, seminar slides
Task 12: Finalization **Description:** The comments received in the final seminar will be included to the final report and the other documents. **Responsibility:** All **Deliverables:** Final versions of the report and the documents
Task 13: Comments for opponent team **Description:** For each seminar the team has to write one page commentary for the opponent team. **Responsibility:** All **Deliverables:** One page commentaries for each seminar

6. Schedule

The important dates and a tentative plan for the schedule are shown below. Agreed meetings with Holmberg or Salo are shown in green and possible future meetings in orange. Tasks 1 to 3 will be accomplished during weeks 4 to 8, tasks 4 to 9 approximately during weeks 9 to 14, and tasks 10 to 13 during weeks 15 to 20. Testing shall be done in parallel with the software coding (shown in light blue in the schedule: task 10, weeks 10 to 14) and it is planned that during weeks 15-17 the testing will concentrate on complex models. Also the documentation will be updated accordingly when the project evolves and new results and ideas are obtained (see, e.g., tasks 4, 6, 7 and 8 during weeks 15 and 16).

15th Feb	Project plan and requirements specification (first draft) to Holmberg for comments
18th Feb	Meeting with Holmberg
20th Feb	Project plan deadline
22nd Feb	Seminar at VTT (project plans)
25th Feb	First test cases ready (i.e. importing data, confidence limits etc.)
15th March	Meeting with Holmberg
Week 13	Possibly a meeting with Holmberg
5th April	Seminar at Nokia (interim reports), Goal: the code solves the cases presented in [3] correctly, after this testing with more complex models.
Week 17	Possibly a meeting with Holmberg
10th May	Seminar in Tallinn (final reports)



7. Project risks

When analyzing the project risks, possible scenarios were identified after which their likelihoods and consequences were considered, thus following the triplet definition of risk. The likelihoods are expressed qualitatively simply by low-medium-high –scale with possible remarks and the consequences by description and qualitative importance (low-medium-high). Finally, risk management measures are considered to mitigate the likelihood and consequence of each scenario. These risks are shown in the table below.

According to our belief, the biggest risks for the project are illnesses (see Scenario 1 in the table below) and too heavy personal workload since the project manager studies part-time and other members have many courses to complete (see Scenario 2). Both of these scenarios can lead to delays and reductions on the quality of the documentation or even partial project failure. Due to the deadline of the course and limited resources the only mitigating measures found were: sharing information and documents by having regular meetings and using SVN repository and Google Drive; prioritization of tasks and possible re-allocation among the team members and possibly omitting the less important tasks; and keeping the required goal in mind to avoid unnecessary burden by widening the scope of the project. Furthermore, it must be recognized that some risks may have cascading features: illness of one member may lead to increasing work load for the others, thus, increasing the likelihood of the second risk.

As for most projects, the team recognizes the fact that some adjustments need to be made for the project plan and schedule as the project evolves. This may be, e.g., due to a risk coming true or due to underestimated time and effort needed for a task. An interesting question is how well the team is able to adapt to these changes which forms one type of a risk. Therefore the project plan and progress is monitored on regular basis and the team will consult Holmberg and Salo if necessary.

Scenario 1: One or several team members catch a cold or fall ill. **Likelihood:** High (already observed twice) **Consequence:** From minor delays to partial project failure depending on the duration, timing and number of members that are unavailable. Importance: from low to high. **Management**: Sharing information and documents, prioritization of tasks. SVN repository is to be used for the software versioning. Google Drive folder is used for saving and sharing the project documentation. Ideas and know-how is shared in regular meetings. If necessary, higher rank tasks are prioritized and the less important are omitted.
Scenario 2: Personal workload is too heavy. **Likelihood:** Medium (the project manager studies part-time and other members have many courses to complete) **Consequence:** Delays and possible reductions on the documentation quality. Importance: medium. **Management:** The group has meetings on regular basis where possible delays and problems can be discussed. If needed, tasks can be re-allocated. The team also considered important not to start widen the scope of the project but to keep the required goal in mind.
Scenario 3: Project requirements are unclear for the team due to several stakeholders, i.e., VTT, Aalto University and Finnish Radiation and Nuclear Safety Authority (STUK). **Likelihood:** Medium **Consequence:** Confusion may lead to delays or different opinions on the success of the project. Importance: medium. **Management:** This ambiguity was discussed with Salo on 8th Feb 2013 and it was confirmed that VTT sets the objectives and requirements for the project. Nevertheless, some ambiguity may still remain and therefore the team will communicate with Holmberg regularly.
Scenario 4: The method is not applicable for solving the dominance relations. **Likelihood:** Low (the method has gone through a journal review process) **Consequence:** The software might be inapplicable but the success and the quality of the requirements specification, test plan etc. are not dependent on that. Moreover, if some deficiencies were to be found in the method it would not lower but more likely increase the value of the project. Importance: Low. **Management:** Due to low likelihood and low importance, no specific mitigation method is planned. If needed though, the problems can be discussed with Holmberg, Toppila and Salo.
Scenario 5: The data for testing is not applicable or available. **Likelihood:** Low (Toppila has already provided the data he used for testing. In addition, Laitonen can use FinPSA-software to produce test-cases.) **Consequence:** If the data for testing is not available, the implemented software cannot be tested extensively. Importance: Medium. **Management:** Since the team already has received some data and results for comparison (paper by Toppila and Salo) this scenario is considered highly unlikely.
Scenario 6: Conflicts among team members. **Likelihood:** Low (Most of the team members know each other beforehand and it seems that the team works well together.) **Consequence:** Decrease in efficiency and motivation possibly leading to delays and reduction on the quality of the documentation. Importance: Medium. **Management:** Fair allocation of tasks among team members. Efforts to keep up positive and optimistic (yet realistic) attitude.

References

[1] Bedford T, Cooke R. Probabilistic Risk Assessment: Foundations and Methods. Cambridge University Press, 2003.

[2] Himanen R, Julin A, Jänkälä K, Holmberg J-E, Virolainen R. Risk-Informed Regulation and Safety Management of Nuclear Power Plants—On the Prevention of Severe Accidents, Risk Analysis, Vol. 32, Issue 11, pp 1978–1993, November 2012.

[3] Toppila A, Salo A. Prioritization of events in fault tree analysis under interval-valued probabilities. Manuscript, 2013.