UNIVERSITY OF TWENTE.

Interview Report

A Logic to Reason about Fault Trees

Stefano M. Nicoletti, Moritz Hahn, Marielle Stoelinga

26th October, 2020

Introduction

In the context of the **ERC Funded Project 864075 (CAESAR)** - inside Work Package 5 - we conducted an interview with a domain expert on safety-critical systems, with particular attention to the nuclear domain. Resulting answers from this interview helped in **clarifying** potential **directions** - and **informing** design-related **decisions** - in the development of a logic to reason about **Fault Trees** (**FTs**).

The Interview

The interview with the **Domain Expert (DE)** - whose anonymity is preserved here - was conducted on **October the 26th, 2020** by three **inverviewers (I)**: **Stefano Nicoletti**, **Ernst Moritz Hahn** and **Marielle Stoelinga**. Following, a transcript of the interview. Some of the key passages are highlighted.

Transcript:

I: Do you think that developing a logic for Fault Trees could have an impact on the field you are working into? If so, in which way?

DE: First, I want to clarify that my expertise focuses on the nuclear domain, where models usually are very large, gradually built and updated. There will be analysts with big models, with several thousands gates and events. So, what happens when they want to update them? It would be very useful if they had simple tools that would allow them to query the FT with such simple questions: "Why do I see cut sets that I was not expecting?" "Why am I seeing this event in the cut set when I do not expect this?" Some of the properties we care about go in this direction.

What we have in our tool is the possibility to set states of events, and then run the analysis. That would give us cut sets lists. Another possibility is for us to manually set some failures and propagate them, to see what happens to the tree.

I: What added value could the development of a logic bring in this setting?

DE: One question that we cannot solve elegantly is "Why don't I see a cut set that I was expecting to see?" "Why is some combination of events not a cut set, that I was on the contrary expecting to be?". In general debugging and troubleshooting big trees could be valuable. Generating counterexamples, having something concrete to read and analyze, and then propagate to the tree. More valuable than saying "The properties X and Y would be satisfied".

I: Having a set of events that is not a cut set, would producing the shortest extension to a minimal cut set be interesting from your point of view?

DE: Yes, for sure!

I: Are there also interesting properties involving probabilties from your point of view?

DE: Yes. From the prospective of a commercial tool, you have to keep in mind that we would like to see instruments that are simple, that work and give answers in a reasonable time. Involving probabilites can be useful in this context. This is something that people often use in risk monitoring.

I: Would a way to plan ahead be useful? E.g., suppose that there is maintenance, what does it do to the system? What happens?

DE: Yes, planning ahead is useful and looking at possible scenarios often happens in monitoring. Another area in which this is useful is importance analysis. Understanding how risk increases and decreases factors and sensitivities, for example. If you have an elegant way of expressing some properties with probabilities this could allow for more flexible importance analysis, e.g.,: "I am interested in this subsystem and want to see combinations of events that can increase failure probabilities above some limit". Furthermore, you could assume that some components are not available and compare the risk with this unavailability.

I: Would it be interesting to represent costs in the trees?

DE: Cost representation is interesting and industrially relevant: I do not know of an industrial tool that fully covers this. Of utmost interest: maintenance optimization.

I: Cost of (planned) downtime was also modelled in a previous work with NS. How often inspections need to take place, for example. Would that be relevant?

DE: That would definitely be interesting!

I: Would the analysis of time-related properties be an interesting feature for you to have?

DE: Certainly! For example, I would like to have triggers such that I could start a given component after another one fails. The mandatory analyses that we conduct in the nuclear domain already consider a 24/48h period. This is of interest e.g., to consider the duration of accidents that could extend in time. That is something even we have done some research on. We think it is interesting and relevant but is not in the commercial tools currently in use.

I: Are you also interested to see if these failures occur inside a given time bound e.g., the 24/48h bound you mentioned?

DE: In models we currently use this would be already included, it is an inherent part of the model. It could be interesting to extend or modify this time period.

I: In your work, do you also encounter faults that are caused by malicious agents?

DE: In the nuclear domain there are analyses of terrorist attacks on a plant, for example. Altough they are not in the model I work with: a separate analysis is conducted. Even if they were added to the FTs I would not know about it, due to confidentiality. I have never seen or heard of a co-analysis that integrated, as you are describing it. This is not to say that this does not already happen in defence and is then covered by confidentiality or that this would not be interesting to us directly.

I: How knowledgeable are people using these tools? Would they understand temporal logics in order to specify properties?

DE: They do not necessarily have knowledge of temporal logics. Specifying properties has to be simple e.g.,**:** there have to be templates, which are simple to use and simple to understand, and they have to work nicely.