Embedding event sequence diagrams within the format

T. Friedlhuber, M. HIBTI & A. Rauzy
Outline

Introduction

PSA and event sequence diagrams

Clarity and transparency of event trees

Conclusions and perspectives
Evaluate the frequency of an undesired event and identify the scenarios that lead to...
The general methodology scheme of a PSA model
Event sequence diagrams are the “source code” of the model representation

- Event sequence diagrams
  (S. Swaminathan and C. Smidts)

- Visualisation of the scenarios in a compact form
- Highlight mitigations within safety functions blocs
- Allow to introduce constraints and assumptions that are not compatible with the static framework
- May be translated to other non static computation engines
The practice of PSA and model construction

» Simplified ESD an output of sequence analysis
The practice of PSA and model construction

- Simplified ESD an output of sequence analysis
- In practice PSA users update their models on the basis of event tree structures
- More complex to review and modify event trees than event sequence diagrams
- May be not coherent with the main sequence assumptions
- May be confusing for other applications of the models (dynamic . . . )
Make event trees easy to understand

- A good/optimized event tree is
  - Easy to read
    - compact, informative and clear
  - Suitable for quantification purposes
  - Suitable for overall risk assessment but also for applications
  - Coherent with the original event sequence diagram
Consistency, completeness and clarity

- Event sequence diagram to Event tree

- Clear semantics for ESD and ET for consistency and completeness
  - Verify that all the sequences of the ETS are relevant
  - Verify that all the scenarios of the ESD are represented in the ET
Open PSA initiative for next generation PSA

Introduce dynamic results in event tree sequences

- Use other formalisms as an alternative to event sequence diagrams
  - Boolean logic Driven Marov Processes
  - Monte-Carlo simulation . . .
- May solve issues regarding
  - Recoveries
    - System recoveries
    - Initiators
  - Time dependency conditional events
Conclusion and perspectives

- Event sequence diagrams as source code for event trees for comfortable visualisation and review
- Make the model clearer
- Make possible (semi)-automated generation
  - With the suitable level of granularity (ET per safety function, or global ET, ...)
  - Appropriate with algorithmic considerations
- Allow integration of input from dynamic models