Business from technology

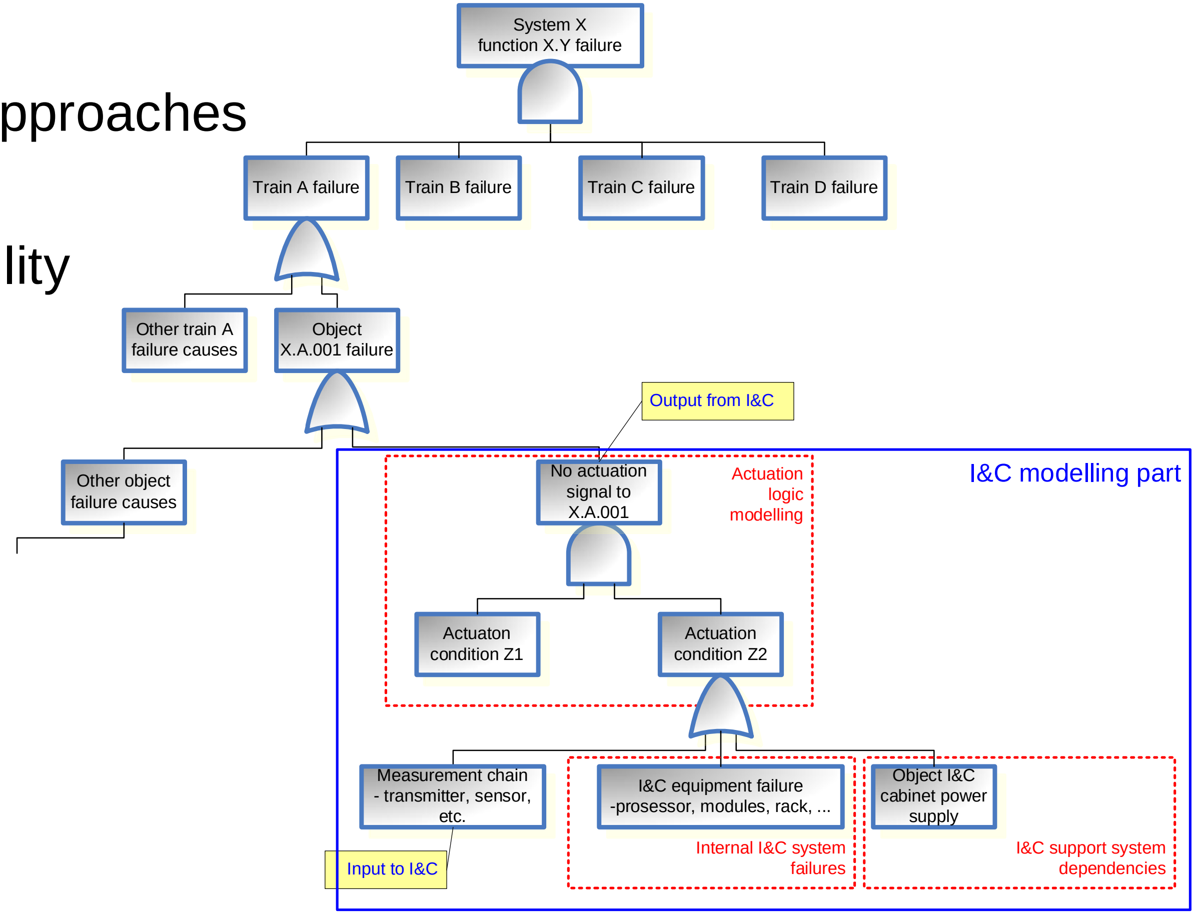
Literature review

Guidelines for reliability analysis of digital systems in PSA context

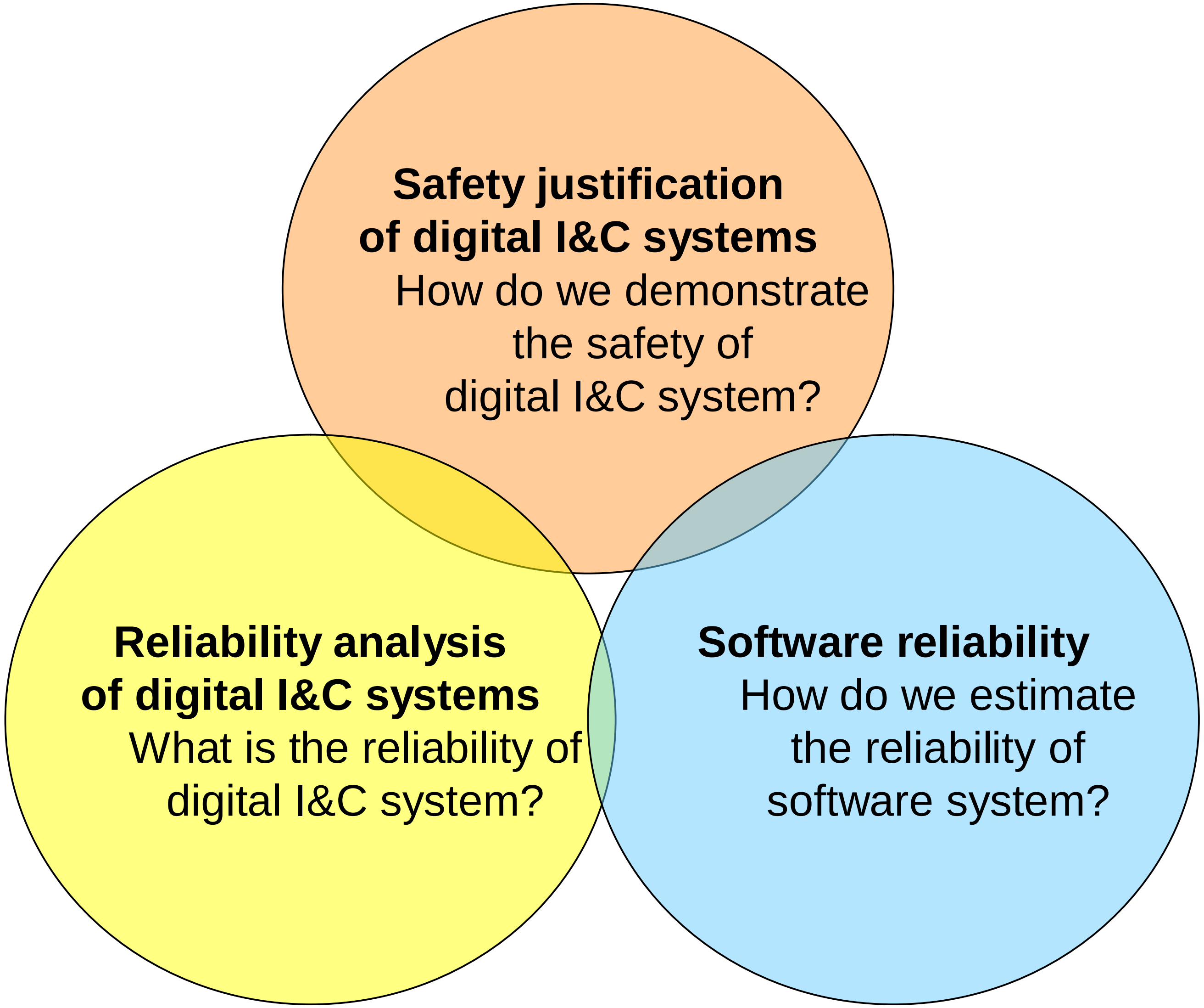
Kim Björkman, Jan-Erik Holmberg
VTT Technical Research Centre of Finland

Contents

- Modelling digital I&C in PSA
- Dynamic reliability modelling approaches
- Assessment of software reliability


System X function X.Y failure
Train A failure
Train B failure
Train C failure
Train D failure
Other train A failure causes
Object X.A.001 failure
Output from I&C
Other object failure causes
I&C modelling part
No actuation signal to X.A.001
Actuation logic modelling
Actuaton condition Z1
Actuation condition Z2
Measurement chain - transmitter, sensor, etc.
I&C equipment failure -prosessor, modules, rack, ...
Object I&C cabinet power supply
Input to I&C
Internal I&C system failures
I&C support system dependencies

Interrelated questions



Safety justification of digital I&C systems

How do we demonstrate the safety of digital I&C system?

Reliability analysis of digital I&C systems

What is the reliability of digital I&C system?

Software reliability

How do we estimate the reliability of software system?

Current practice in PSA

- Computer-based systems are analyzed mostly simply and conventionally
  - Failure mode and effects analysis
  - Fault tree model
    - Basic events: CPU failure, application failure, common cause failure between identical components
    - It is not clear between which system parts CCFs should be postulated
    - It is not clear which failure modes should be postulated
- Primary goal is to model dependencies
  - As long as there is enough diversity in safety systems, computer-based systems will not cause problem

Challenges of the reliability analysis of I&C functions

- The structure of safety I&C (actuation logic sequences) is clear
  - dependencies between actuators can be identified with a reasonable effort
- Operational I&C (control functions) is more difficult to model
  - do we need to take it into account in PSA?
  - on the other plant availability is important, too
  - can we use simulator as an analysis tool?
- Which interactions of digital systems are considered?
- HRA is always a challenge
  - operator actions
  - maintenance and testing
  - modifications
- Power supply dependencies are a challenge too
  - e.g. the effects of voltage peak on automation or effects of fire and smoke?

Challenges of the reliability analysis of I&C equipment

Which failure modes are assumed?

- failure to function, spurious actuation, other?

Which failure types are assumed?

- ”processor failure”, ”application failure” or more specified?

To what degree failure detection and fail-safe behaviour are accounted?

For which components and failure modes common cause failures are postulated?

Which reliability model is assumed (e.g. simply per time or per demand)?

Where reliability data is taken from?

- software testing data
- operational experience
- justification of expert judgments
- Use of formal methods and software analysis tools
  - what can we conclude from such analyses from the reliability point of view?

Traditional reliability analysis methods for digital systems

- Event Tree/Fault Tree method and the Markov method
- Traditional methods are useful in the modelling digital I&C but also limitations are present
- Event tree-fault tree approach does not explicitly treat the timing of events in accident sequences. Interactions with plant processes are implicitly and approximately considered.
- The construction of Markov models can be a laborious, time-consuming manual process, and the resulting transition matrix can be extremely large
- Case study: DFWCS was modelled with Markov method [NUREG/CR-6997]
  - order of component failures is important
  - proposed approach feasible
  - integration with a PRA based on the ET/FT method may not be a trivial task

Improvable areas

- Methods for defining and identifying failure modes and effects of digital systems.
- Methods and parameter data for modeling self-diagnostics, reconfiguration, and surveillance
- Better data for hardware failures of digital components
- Better data for CCFs of digital components.
- Methods for estimating the risk from software faults in both application and support software.
- Methods for modeling software CCF across system boundaries (e.g., due to common support software).
- Methods for considering modeling uncertainties
- Methods for human reliability analysis

Failure classification & FMEA

- FMEA is a well-known method used to identify failure modes of a system and their effects or consequences on the system. A few guidance documents for performing an FMEA are available, e.g. IEEE 352 and IEC 60812
- Specific guidance about how to perform FMEA of digital systems appears to be lacking
- No generic or standard list of failure modes of digital systems/components
- FMEA by itself may not be a sufficient tool to determine how specific component-level failure modes affect digital systems
  - More sophisticated tools, such as simulation tools, can be used to analyze the interactions between the components of a digital system and the effects of one or more failures

Dynamic reliability modelling approaches

- Generally, dynamic methodologies provide a much more accurate representation of probabilistic system evolution in time than the fault tree/event tree approach.
- However, the dynamic models are on a trial stage and usually it is a difficult task to integrate dynamic models to existing PSAs
- E.g., DFM, Markov-CCMT, Petri Nets, Bayesian methodologies

Dynamic reliability modelling approaches

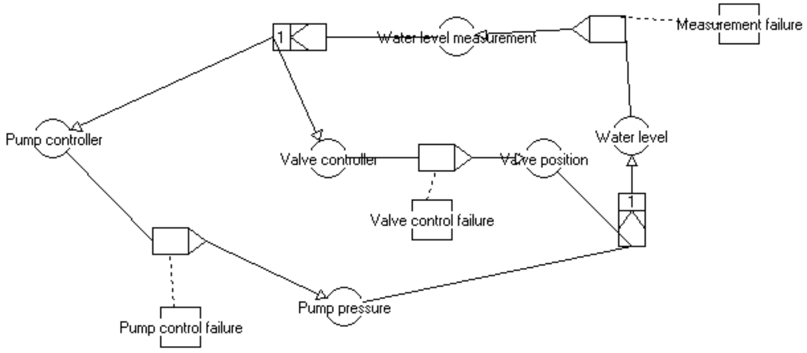
Dynamic flowgraph methodology

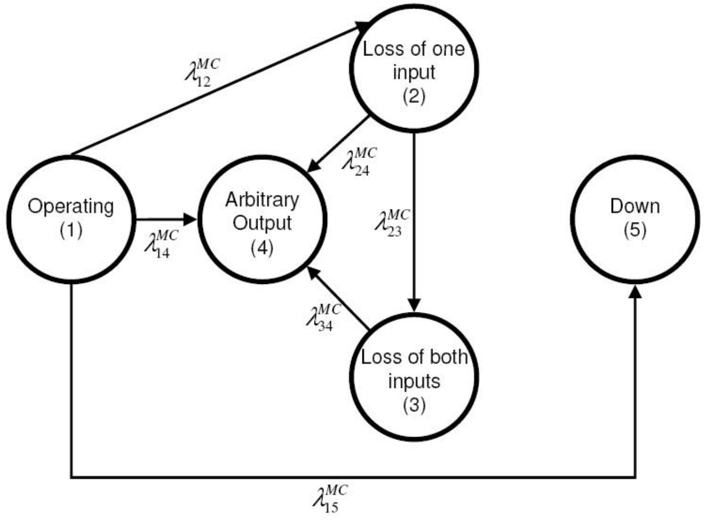
- Based on directed graphs
- For modeling and analyzing the behavior and interaction of software and hardware within an embedded system.
- Continuous variables have to be discretized
- The number of time steps that can be analyzed in deductive mode is limited by computational constraints.

Markov/CCMT

- Combines the traditional Markov methodology with cell to cell mapping
- Enables to represent couplings between failure events, originated from dynamic interactions between:
  - the digital I&C system and the controlled process
  - among the different components of the I&C system
- Construction of a full Markov/CCMT model may not be computationally feasible

Dynamic reliability modelling approaches





- The construction of a DFM or a Markov/CCMT model requires extensive technical knowledge
- Both methodologies are more complex and more difficult to apply in effective fashion compared to typical PSA ET/FT paradigm
- The integration of Markov/CCMT and DFM results with PSA can be fairly straightforward

Software reliability analysis approaches

- Reliability growth models
- Test based models
- Subjective (Bayesian) methods
- “Rule based” methods
- Software metric based methods

Software reliability analysis approaches

Reliability growth models

- These models are based on the sequence of times between observed and repaired failures
- The reliability growth models are in general based on two types of data:
  - debugging data
  - experience data from real operation
- A single system is followed chronologically with recording of times to failure, and that the faults are corrected

Test based models

- Testing means to execute a program with selected data and check the answer against an ‘oracle’
- A simple estimate of software reliability could be based on program testing, by dividing the number of failed tests with the number of executed tests
- Input data profile used during the test should correspond to the input profile during real operation

Software reliability analysis approaches

Subjective (Bayesian) methods

- Many factors, that are important to software reliability, cannot be put directly into a mathematical reliability formula
- Bayesian methods can be used to elicit information from these evidences to make reliability estimates based on expert judgments
- Could be used as prior distribution in Bayes' formula, to produce a posterior distribution after testing
- Bayesian Belief Networks (BBN) can be used to combine evidences from different information sources for a quantitative assessment of this belief

Software metric based methods

- Based on the reliability assessment on objective measures on relevant documents
- E.g., lines of code, number of decisions, number of program paths, path coverage in test, internal complexity measures, complexity in connections to other systems
- A correlation is assumed between these measures and the likelihood that the systems contain faults
- This correlation is difficult to assess

“Rule based” methods

- A set of requirements that must be fulfilled for a system to reach a certain safety level
- To the different safety levels one may assign probabilistic requirements
- Safety Integrity Levels (SIL) in IEC-61508
- Standards specify a set of criteria the program and program development must fulfill to be accepted at that level
- Can be used as design guidelines or as quantitative reliability targets

SIL	**Demand Mode of Operation** (average probability to perform its design function on demand)	**Continuous / High Demand mode of Operation** (probability of dangerous failure per hour)
1	$10^{-2} \leq p < 10^{-1}$	$10^{-6} \leq p < 10^{-5}$
2	$10^{-3} \leq p < 10^{-2}$	$10^{-7} \leq p < 10^{-6}$
3	$10^{-4} \leq p < 10^{-3}$	$10^{-8} \leq p < 10^{-7}$
4	$10^{-5} \leq p < 10^{-4}$	$10^{-9} \leq p < 10^{-8}$

Conclusions
Software reliability

- Software failures are in general mainly caused by systematic (i.e. design specification or modification) faults), and not on random errors
    - Difficult to give a more precise definition of the “uncertainty” concept with respect to software reliability
    - The uncertainty is epistemic, i.e. due to subjective judgment and experts’ lack of knowledge
- The software based systems cannot easily be decomposed into components, and the interdependence of the components cannot easily be identified and modelled
- It is difficult to apply software reliability models in PSA context
    - software reliability models relies on assumptions and statistical data which are not valid for software products implemented at the plant
    - systematic use of expert judgment is the only possibility to assess the *software* reliability?

Conclusions
PSA modelling of digital I&C systems

- Software-based safety functions and components should be included in PSA in a way or another
- The basic question: “What is the probability that a safety system or a function fails when demanded” is fully feasible and well formed question for all components or systems, independently on the technology on which the systems are based
- Dynamic reliability analysis models may be attractive approaches to some systems but do not solve the problem of software reliability
    - conventional FT-approach is sufficient for RPS kind of functions?
- The exact values of failure probabilities are not as important as the proper description of the impact of software-based system to the dependence between the safety functions and to the structure of accident sequences

Literature

OECD/NEA: NEA/CSNI/R, 2002, NEA/SEN/SIN/WGRISK(2007)1, NEA/CSNI/R(2009)18

NUREG/CR -6848, -6901, -6942, -6962, -6985, -6997

NUREG/GR -0019, -0020

Conference papers: SAFECOMP, ANS Topical Meeting NPIC&HMIT, ISSRE, PSAM, PSA, ESREL

Journal articles: Reliability Engineering and System Safety, IEEE Trans. on Systems, Man and Cybernetics, IEEE Trans. on Software Engineering, IEEE Trans. on Reliability

Standards: IAEA, IEC, IEEE, ...

Books on software reliability