# Introduction to System Dependability

Kevin Delmas (kevin.delmas@onera.fr)

8 octobre 2024

# Lecture overview

**Goals** provide background to understand how are built dependable systems

- Concepts of dependable systems
- Process used to achieve dependable system
- Dependability Assessment techniques

**Plan**

- Dependability concepts and process (KD)
- Fault tree analysis (KD) and marked lab (KD + TP)
- Specific risk management (KD)
- Model based safety assessment (TP) and marked lab (KD + TP)

**Evaluation**

- A quiz at the end of each lab

Some definitions are mandatory to understand labs (what a surprise)

= slides preparing computer lab

Interactive course ahead

Scan the QR code or connect to menti.com and enter 78720404

# Introduction to System Dependability

## What is a system ?

# What is a system ?

**System**

A system is a set of interacting items, forming an integrated whole

**System**

examples of various complexity : air traffic control, aircraft + pilot, flight-control system, computers, sensors, actuators,. . .

# An example of system

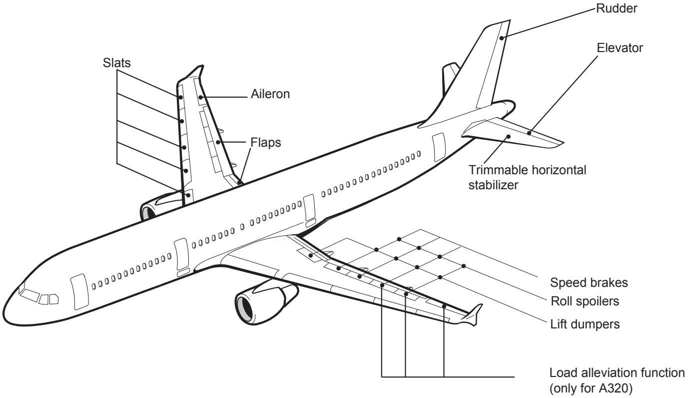



FIGURE – Aircraft actuators

# An example of system

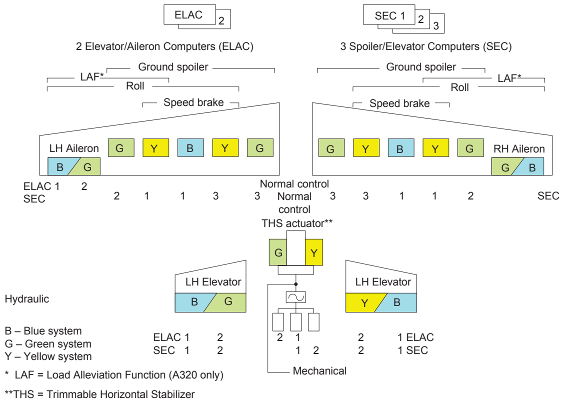



FIGURE – Hydraulic allocation

# An example of system : Hydraulic system

Hydraulic power generation and distribution system made of three sub-systems Green, Yellow and Blue.

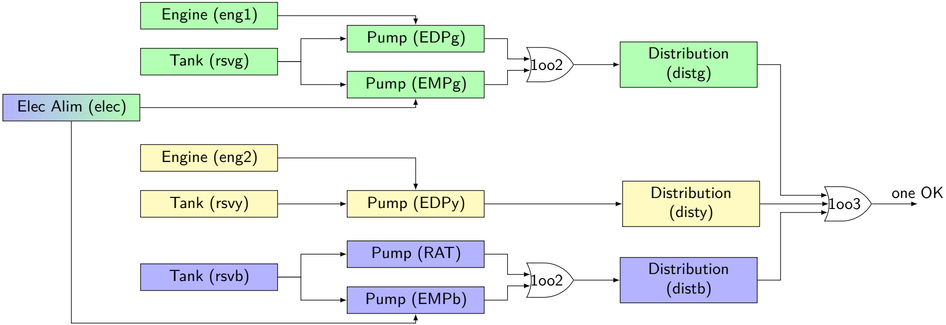



FIGURE – Hydraulic system

# An example of system : Pitot sensor

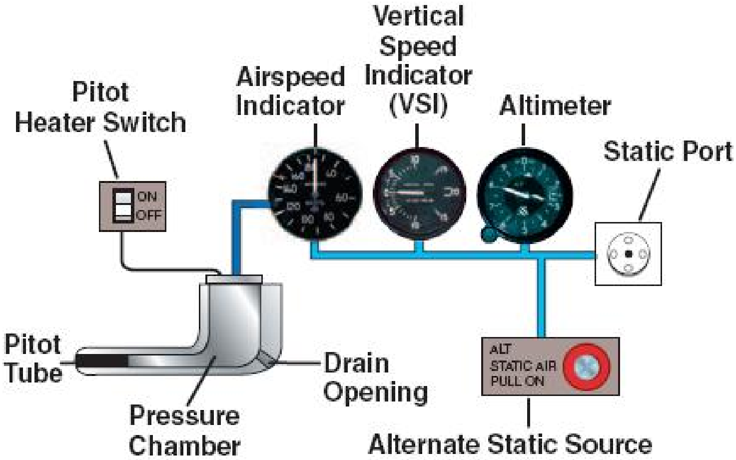
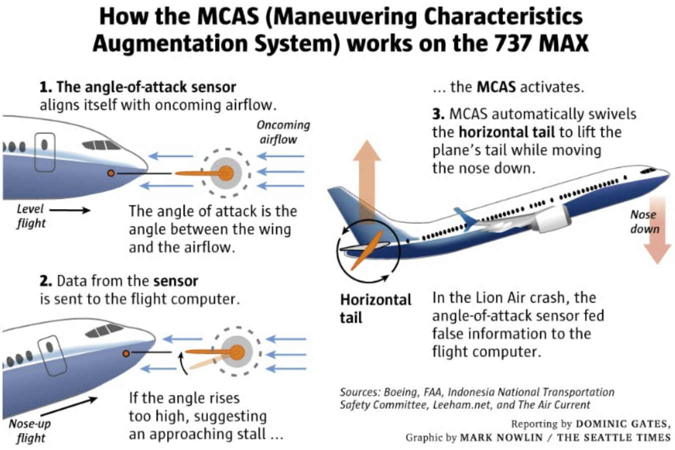



FIGURE – Pitot Static System

# An example of system : MCAS



Introduction to System Dependability

What is dependability ?

# What is dependability ?

## Dependability [ALRL04]

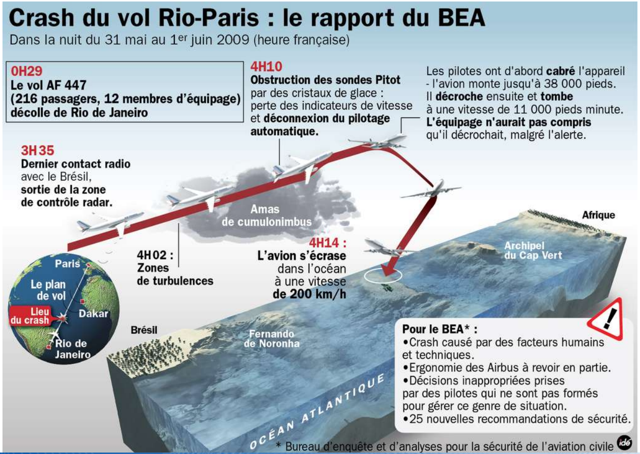
The ability of the system to deliver service that can justifiably be trusted.

Framework to complete the specification beyond the strict definition of what would be expected in a flawless world

- Service specification (and its development and validation)
- Dependability specification (and its development and validation)

# Consequences of flaws : Pitot icing

BEA accident report available here



# Consequences of flaws : erroneous MCAS activation

KNKT accident report available here

Resumed Flight History :

- Unintended trigger of the MCAS (assumed cause erroenous AOA sensor)
- Crew was not able to identify cause of MCAS activation and tried multiple manual overrides
- Crew considered (unusual) that situation not require a landing to nearest airport
- Eventually, final MCAS accivation leads to descente rate above 10000 feet/min

Need to identify and handle the dependability threats

# Dependability concepts

Dependability threats (what can go wrong) :

failure occurrence of the deviation of the delivered service from expectations

- severity : harm of its direct or indirect consequences
- mode : characterization of the way a system/item fails
- consistency : Byzantine failure
- rate : probability of failure per unit of time of items in operation

error Part of the state of the system which may lead to a failure

- latent or detected

fault hypothesized or adjudged cause of an error state

- Dormant or active, internal or external (w.r.t. system boundaries)
- Physical or human (accidental or intentional), in development or operation
- Temporary (transient, intermittent), permanent

Recursive propagation path :

$$\text{fault} \Rightarrow \text{error} \Rightarrow \text{failure} \Rightarrow \ldots$$

# Hydraulic system

**Nominal function** hydraulic power delivery

**Failure** no delivery of hydraulic power

**Failure modes**

- total loss of delivery of hydraulic power (loss of the three lines)
- partial loss of delivery of hydraulic power (loss of one line)

# Behavior under fault

System/items behaviors depend on

- control/observation interface
- internal states (not always distinguishable)
  - nominal functioning modes
  - error states part of the total state of a system/item that may lead to its subsequent failure

# Hydraulic system

**Failure mode** loss of delivery of hydraulic power on one pipe on demand

**Error state** hydraulic pipe broken

**Fault**

- Primary (intrinsic) cause : pipe wearing
- Secondary cause (extrinsic) : pipe received too high pressure fluid

**Observability** Not detectable when not power is demanded (pump off)

Concretely, how to evaluate dependability ?

# Dependability attributes

Dependability assessed using a set of quantitative and qualitative attributes such as :

- **Availability** Readiness of the service
- **Reliability** Continuity of the service
- **Maintainability** Ability to undergo repair
- **Safety** ability to avoid too severe consequences (human, environment)
- **Security** ability to ensure condfidentiality (non disclosure to unauthorized users), integrity (malicious alterations) and availability (no DoS) of the service

# Math corner : Availability

## Availability(A)

Ability of a system S to deliver a correct service at a given time :

$$A(t) = p(S \text{ non faulty at } t)$$

## Availability

In the space domain :

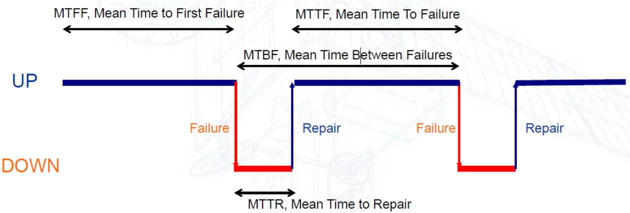
- Launcher : capability to launch at the scheduled time
- Satellite : capability to perform some critical mission phases (*e.g.* orbit insertion, fly-by)

# Math corner : Availability

## Average availability

Proportion of up-time between 0 and t (or over the lifetime)

$$A = MTTF/MTBF$$



# Math corner : Reliability

## Reliability(R)

Ability of a system S to ensure continuity of correct service :

$$R(t) = p(S \text{ non faulty over } [0, t])$$

## Reliability

In the space domain :

- Launcher : reliability characterises the mission success
- Satellite : reliability characterises the lifetime through the probability to have not experienced any fatal failure at t

# Math corner : Safety

## Safety

Ability of a system S to avoid harmful events (human, environement)

## Safety

In the space domain :

- Launcher : explosion, fall-down of large pieces or toxic material
- Satellite :
  - ground operations,
  - in-orbit servicing, docking (e.g., ATV with the International Space Station),
  - end of life, re-entry

# Math corner : Failure rate & Maintainability

## Maintainability(M)

Ability of a system S to undergo modifications and repair

$$M(t) = 1 - p(S \text{ non repaired over } [0, t])$$

## Failure Rate ($\Lambda$)

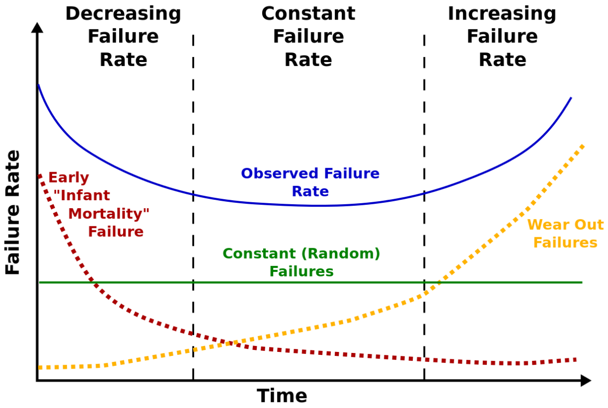
Probability of a system S to fail at $t + dt$ knowing it has not failed over $[0, t]$ :

$$\Lambda(t) = \lim_{dt \to 0} \frac{p(S \text{ fails during } [t, t + dt])}{dt} \frac{1}{R(t)}$$

Relation with R :

$$R(t) = e^{-\int_0^t \Lambda(u) du}$$

# Math corner : Bath curve failure rate




Assume items used during constant failure rate phase

# Math corner : Computation approximation

## Rare failure assumptions

When $\lambda t \sim 0$ (usually $\lambda t < .1$) use Taylor expansion for computations :

$$\overline{R}(t) = 1 - R(t) = 1 - e^{-\lambda t} \underset{0}{\sim} \lambda t$$

## Independence & pessimism assumption

If two components $C_1$ and $C_2$ have independent failures with failure rate $\lambda_1$ and $\lambda_2$

$$\begin{aligned}
p(\text{both fail}) &\underset{\text{independent}}{=} p(C_1 \text{ fails})p(C_2 \text{ fails}) = \lambda_1\lambda_2 t^2 \\
p(\text{one fails}) &= p(C_1 \text{ fails}) + p(C_2 \text{ fails}) - p(\text{both fail}) \\
&\underset{\text{pessimism}}{=} p(C_1 \text{ fails}) + p(C_2 \text{ fails})
\end{aligned}$$

How to ensure dependability ?

# Dependability means

Faults leading to harmful events can be :

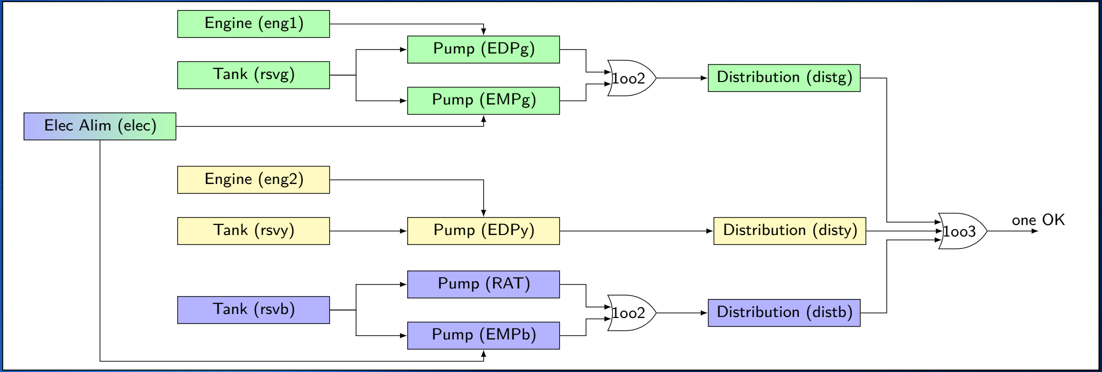
**Prevented** Avoid to introduce fault during the design of the system *e.g.* correct by construction design, rigourous development process

**Tolerated** Deal with the possible errors and failures caused by residual faults *e.g.* architectural tolerance, defensive programming

**Removed** Track and remove faults introduced during the system design *e.g.* formal code verification, specification-oriented test

**Forecasted** Predict the time of the next fault and apply preventive actions to avoid subsequent errors *e.g.* predictive maintenance

# Can you identify a dependability means used to handle failures in the hydraulic system ?



# Fault tolerance by structural redundancy

**Strategy** Implement various element capable of delivering a given (critical) service

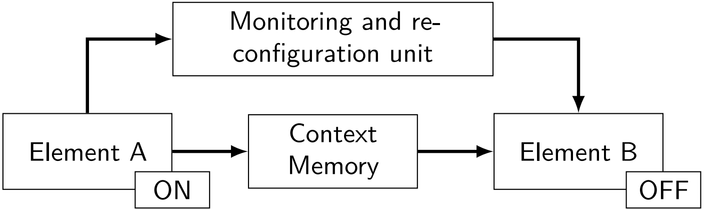
**Selective Redundancy** Provide service out of two elements

- Hot redundancy if both are active
- Warm/Cold redundancy if one of the component is used as a backup

**N-modular redundancy** Duplex, majority voting

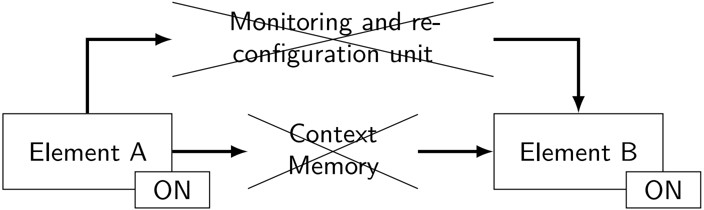
Useful only if indenpendency w.r.t to faults *i.e.* ensure diversification during design

# Cold Redundancy



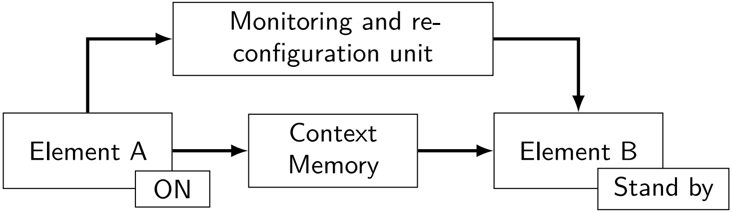

- Most often used for space systems
- Most reliable as the failure rate of an unpowered element is generally significantly lower than of a powered one (about one tenth)

# Hot Redundancy



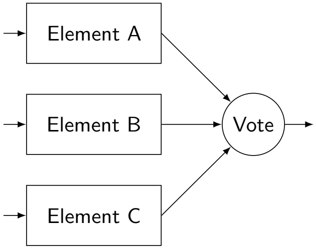

- Need to define output selection process
- Lower long-term reliability
- Useful if the backup cannot be activated in case of failure (*e.g* telecommunication)
- Useful if equipment for which no interruption of service is tolerated (*e.g.* launcher flight control)

# Warm Redundancy



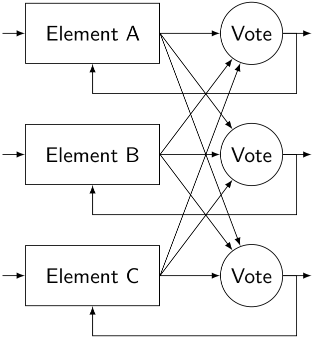

- For equipment with a long start-up time (*e.g.* computers)
- Ensure very short reconfiguration times
- More complex to manage (periodic backup and upload of context, alarm watchdog & reconfiguration)

# N-Modular redundancy




- Ensure service continuity in case of single failure on elements
- Caution, voter can be considered as single point of failure
- Common case/mode faults on elements

# N-Modular N-Voting redundancy




- Ensure service continuity in case of single failure on elements
- Possible element deactivation after desagreement
- Common case/mode faults on elements

# Example of self checking components

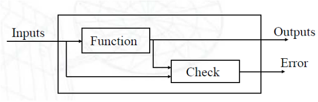



FIGURE – Fail-stop block

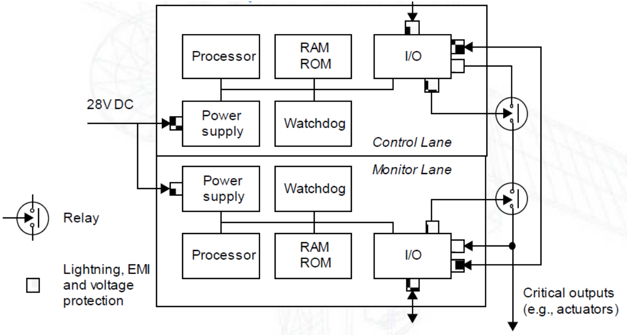



FIGURE – Airbus Command/Monitor (COM/MON) computers

# Combining fault tolerance mechanisms

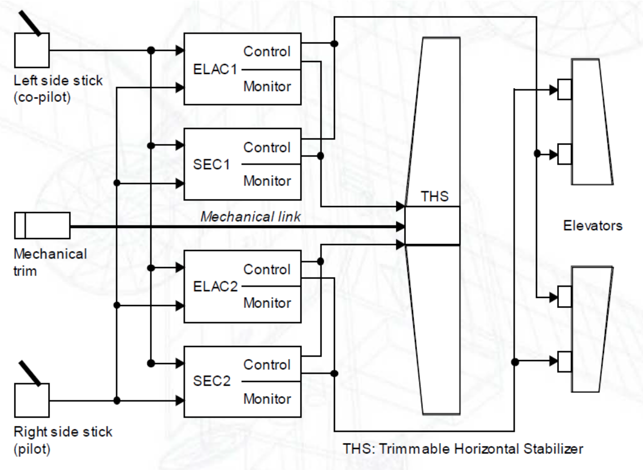



Figure – Aircraft fly-by-wire

OK, but would you take this plane if

$$1 - R_{\text{total loss}}(10^3 h) = 10^{-4},\ 10^{-6}\,?$$

OK, but would you take this plane if $1 - R_{\text{total loss}}(10^3 h) = 10^{-4},\ 10^{-6}$ ?

It depends . . .

# Risk acceptability

The question is :

What happens if ?

# Risk acceptability

The question is :

What happens if hydraulic system fails ?

# Risk acceptability

The question is :

What happens if hydraulic system fails ?

- No power in actuators
- Loss of trajectory control
- Depending on flight phase, injury or death of passengers and/or aircraft crew.

# Risk acceptability

New question :

> Knowing the severity of the failure, what is an acceptable frequency of such failure ?

Another general definition of dependability :

> "ability to avoid service failures that are frequent and more severe than acceptable"

What does service failure, severe, frequent, acceptable mean ?
⇒ Regulatory texts

# Regulatory texts & norms

Regulation For safety-critical systems, regulation are provided as regulatory texts such as :

- Safe use of nuclear technology for peaceful applications, IAEA, 1957
- Peaceful use of outer space, COPUOS, 1958
- Certification specification for large aeroplanes, EASA, 2003
- Certification specification for large rotorcraft, EASA, 2003

Norms & Standards Acceptable means of compliance to the regulatory texts

$\Rightarrow$ sometimes applied by applicant without existing regulation (*e.g.* automotive)

# Overview of standards by domain

**Aeronautics**
- System related : ARP4761, APR4754-A
- Hardware related : DO254
- Software related :DO178-C

**Automotive** ISO26262

**Nuclear** IEC 60880, IAEA DS-431

**Railway** EN 50128, 50126, 50129, 50155, IEC 61508

**Space** ECSS

# Qualification vs Certification

**Qualification** Activities granting a confidence level to an entity (person, organisation or artefact)
$\Rightarrow$ Activities tailored to the context of qualification : item, actors, usage, timeline

**Certification** An assessment body substantiates to an Authority that the engineering process of an applicant ensures regulatory safety objectives through conformance to safety standards

# Actors per domain

| Domain | Applicant | Regulation | Autority | Assessment Body |
|---|---|---|---|---|
| Aeronautics | Manufacturer | Yes | EASA-FAA | EASA-FAA |
| Automotive | Manufacturer | No | No | No |
| Nuclear | Operator | Yes | National agency (*e.g.* ASN) | ASN, IRSN (France) |
| Railway | Manufacturer | Yes | ERA | CERTIFIER, ... |
| Space | Manufacturer | Yes | National agency | CNES (France), NASA/FAA (USA) |

# Integration of the safety

Safety mechanisms can be designed as :

- A dedicated system monitoring and piloting the actual system
  - possible when high-level emergency actions (*e.g.* core shutdown) ensure to reach a safe state
  - classically used in railway and nuclear domains
- A set of component integrated in the system itself
  - mandatory when service interruption is harmful (*e.g.* flight controller)
  - classically used in aeronautics
- A combination of the two (spatial and automotive domain)

# Demonstration of the safety : Means vs objectives

Norms and standard can demonstrate compliance to regulation by :

- Providing high-level objectives (aeronautics, nuclear, space)
  - ⊕ (Quite) Generic and applicable to various context
  - ⊖ Applicant need to provide a compelling demonstration of the compliance to the objective
- Providing specific means and activities (railway, automotive)
  - ⊕ Simplify verification of the compliance
  - ⊖ Tailored to a specific context, need updates for each new technology, system, tools

# Demonstration of the safety : Common philosophy

Across all the applicative domains use the notion of severity/assurance/integrity level

Levels are used to :

- tailor requested objectives and activities
  $\Rightarrow$ risk-driven effort
- identify and avoid failure propagation from "low cofidence" elements (*e.g.* passenger entertainment system) to "high confidence" elements (*e.g.* flight management system)

How these concepts are implemented for large civil aircraft ?

# Risk acceptability for civil aircraft

When considering safety of civil aircraft :

Failure Condition (FC) kind of service failures that :

- has an effect on the aircraft and its occupants, both direct and consequential,
- caused by one or more failures, considering relevant adverse operational or environmental conditions.

Severity Failure Condition is classified in accordance to the severity of its effects as defined .

# Risk acceptability for civil aircraft

| severity class | effects description | acceptable frequency |
|---|---|---|
| catastrophic | prevent continuous safe flight and landing : aircraft loss and loss of crew and passengers | $< 10^{-9}$ per flight hour and no single failure leads to the FC |
| hazardous | large reduction in safety margins or functional capabilities or physical distress or high crew workload or serious or fatal injuries to a relatively small number of passengers | $< 10^{-7}$ per flight hour |

# Risk acceptability for civil aircraft

| severity class | effects description | acceptable frequency |
|---|---|---|
| major | significant reduction in safety margin or functional capabilities or significant increase in crew workload or discomfort to occupants possibly including injuries | $< 10^{-5}$ per flight hour |
| minor | no significant reduction in aircraft safety. | $< 10^{-3}$ per flight hour |
| no safety effect | | |

# Risk acceptability for civil aircraft

## Severity & objectives

"Total loss of hydraulic system " is classified , so

# Risk acceptability for civil aircraft

## Severity & objectives

"Total loss of hydraulic system " is classified Catastrophic, so

- the probability rate of this failure condition shall be less than $10^{-9}$ /FH and
- No single event shall lead to this failure condition

Warnings :

- The regulation is not the same for military aircraft
- The regulation for civil UAV is still in discussion
- A generic agreed classification is an open question for a lot of domains

How to apply these concepts to build a complex dependable system ?

# Process based approach

Main steps :

- Identify dependability requirements
- Specify a system architecture to ensure these properties
- Assess whether the proposed specification fulfills the dependability requirement
- If OK, refine the system design and iterate

Guidelines tuned according to the system kind :

- ISO 26262 [ISO10] for automotive systems
- ECSS Q-ST 40 for space systems
- ARP 4754A [SAE10], ARP 4761 [SAE96] for aeronautic systems

# Dependability & development process

Integrated dependability process in development process ⇒ Avoid late detection of dependability issues

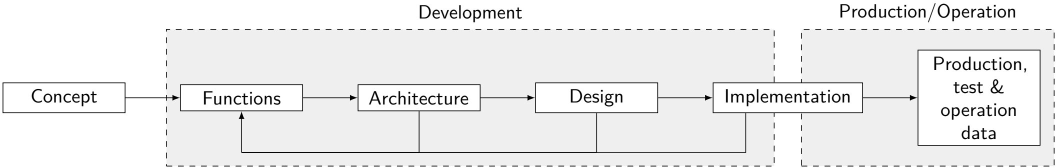



FIGURE – Development life cycle

When should we perform safety activities ?

# Dependability & development process

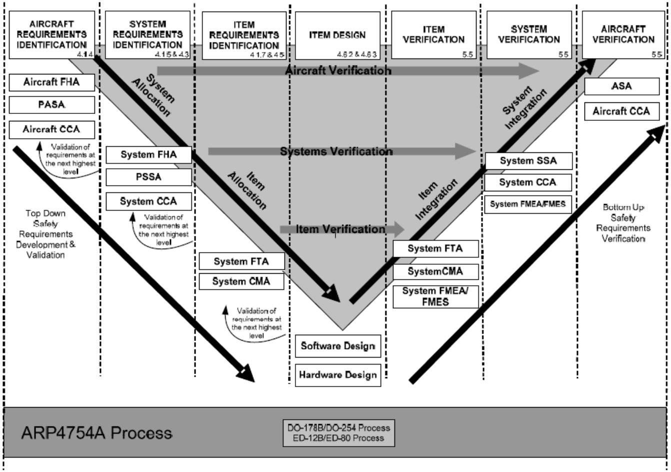
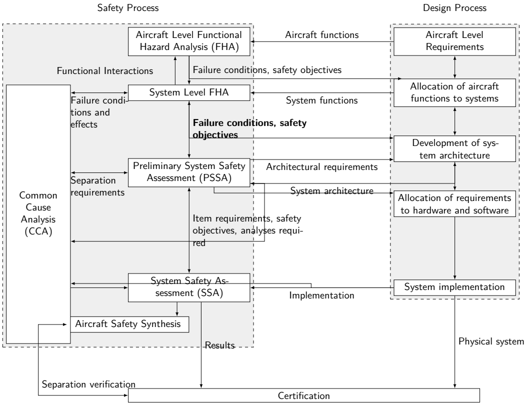


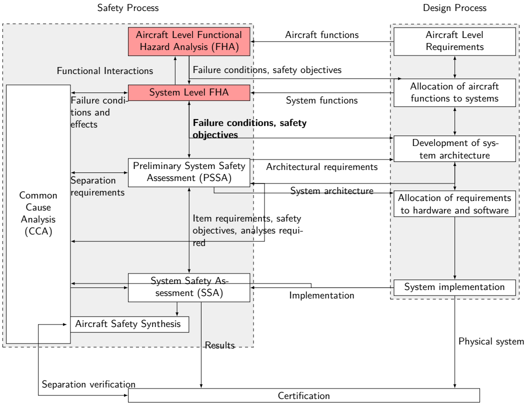

FIGURE 5 - INTERACTION BETWEEN SAFETY AND DEVELOPMENT PROCESSES

# Safety Process (Complete)



When should we identify and classify Failure Conditions ?

# Safety Process (FHA)






ONERA
THE FRENCH AEROSPACE LAB

# Functional Hazard Assessment (FHA)

**Definition** Systematic, comprehensive examination of functions to identify and classify FCs of those functions according to their severity

**Process**

1. identify functions associated with the system under study
2. identify and describe FCs associated with these functions, considering single and multiple failures in normal and degraded environments
3. determine effects of the FC
4. classify FC effects on the aircraft (cat, haz, maj, min, no safety effect)

# Simplified FHA by the example

| System | Function | Failure Mode | Context | Effects | Severity |
|---|---|---|---|---|---|
| Hydraulic system | Generate hydraulic power | Total loss | During cruise | | |

Table – Simplified FHA of Hydraulic system

# Simplified FHA by the example

| System | Function | Failure Mode | Context | Effects | Severity |
|---|---|---|---|---|---|
| Hydraulic system | Generate hydraulic power | Total loss | During cruise | Loss of aircraft controllability | |

Table – Simplified FHA of Hydraulic system

# Simplified FHA by the example

| System | Function | Failure Mode | Context | Effects | Severity |
|---|---|---|---|---|---|
| Hydraulic system | Generate hydraulic power | Total loss | During cruise | Loss of aircraft controllability | Catastrophic |
| | | | Annunciated during taxi | | |

Table – Simplified FHA of Hydraulic system

# Simplified FHA by the example

| System | Function | Failure Mode | Context | Effects | Severity |
|---|---|---|---|---|---|
| Hydraulic system | Generate hydraulic power | Total loss | During cruise | Loss of aircraft controllability | Catastrophic |
| | | | Annunciated during taxi | Evacuation of passengers | Minor |
| | | Partial loss | During cruise | | |

Table – Simplified FHA of Hydraulic system

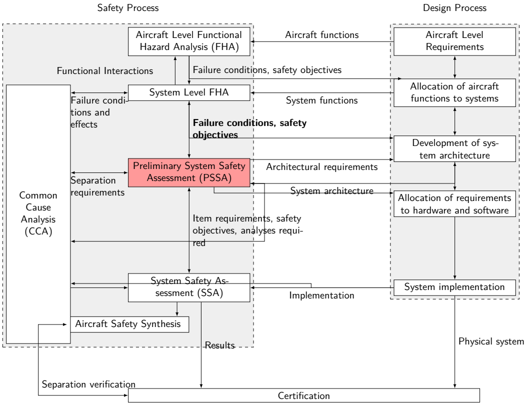
# Simplified FHA by the example

| System | Function | Failure Mode | Context | Effects | Severity |
|---|---|---|---|---|---|
| Hydraulic system | Generate hydraulic power | Total loss | During cruise | Loss of aircraft controllability | Catastrophic |
| | | | Annunciated during taxi | Evacuation of passengers | Minor |
| | | Partial loss | During cruise | Limited controllability of aircraft | Minor |

Table – Simplified FHA of Hydraulic system

When should we check dependability requirements ?

# Safety Process (PSSA)





ONERA THE FRENCH AEROSPACE LAB

How to check dependability requirements ?

$\Rightarrow$ several complementary methods

# Failure Modes and Effects Analysis (FMEA)

**Definition** Inductive analysis of local and global effects of all components failures

**Process** Fill-up for each system component following table.

| Failure Modes and Effects Analysis (FMEA) | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Aircraft :<br>Function :<br>System :<br>Sub-system :<br>Component : | | | | | | | | |
| No | Item | Function | Failure Mode | Failure Cause | Failure Rate | Failure Effects | Recognition failure | Remarks |
| | | | | | | | | |

# Failure Modes and Effects Analysis (FMEA)

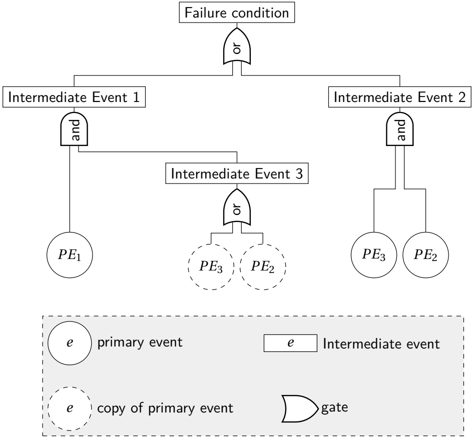
**Definition** Inductive analysis of local and global effects of all components failures

**Process** Fill-up for each system component following table.

| Failure Modes and Effects Analysis (FMEA) | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Aircraft : | | XXX | | | | | | |
| Function : | | Deceleration on ground | | | | | | |
| System : | | Hydraulic Power Generation & Distribution | | | | | | |
| Sub-system : | | Green System | | | | | | |
| Component : | | Pipe | | | | | | |
| No | Item | Function | Failure Mode | Failure Cause | Failure Rate | Failure Effects | Recognition failure | Remarks |
| 1 | Green Pipe | Power distribution | Loss | Aging | $10^{-4}$ | Loss of green system, hydraulic system remains available for aircraft | Warning on pilot display | Select "Green pump off" and turn on power transfert unit |

What is the link between primary events and failure conditions?

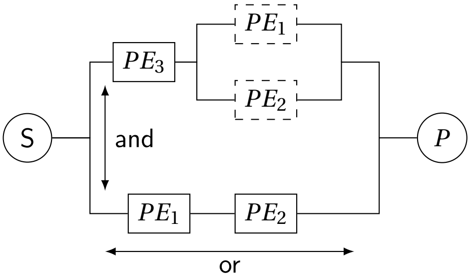
# Failure propagation : The Fault Tree




Legend

# Failure propagation : Reliability Block Diagram

**Alternative notation for fault trees (analogy with serial-parallel electrical circuits)**



How do we use these representations ?

# Build a fault tree

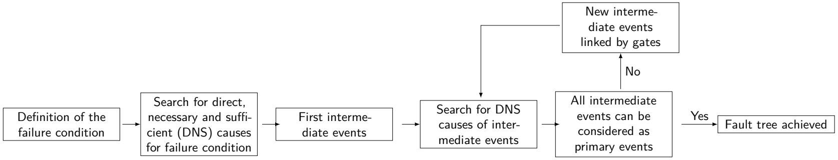



FIGURE – Fault tree construction process

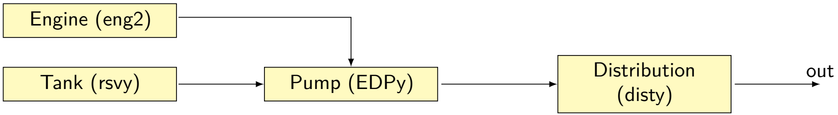



FIGURE – Yellow hydraulic system

# Build a fault tree

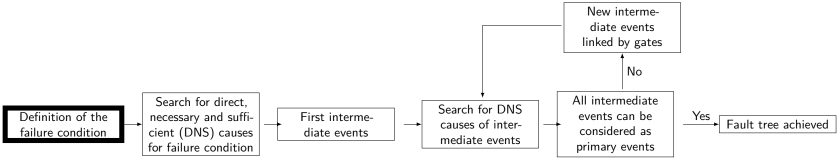



Figure – Fault tree construction process

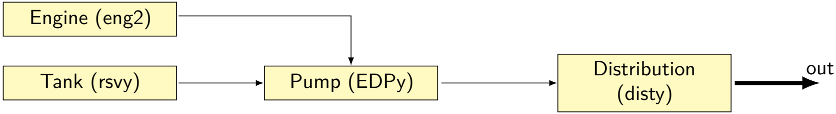



Figure – Yellow hydraulic system

# Build a fault tree

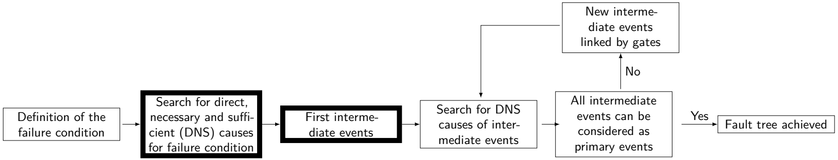



FIGURE – Fault tree construction process

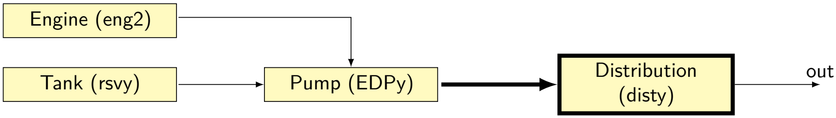



FIGURE – Yellow hydraulic system

# Build a fault tree

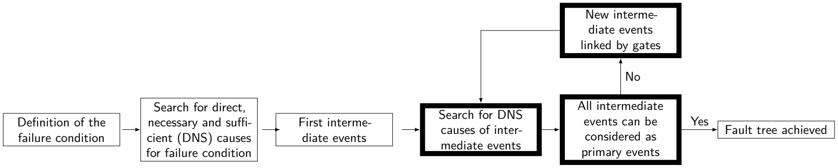



FIGURE – Fault tree construction process

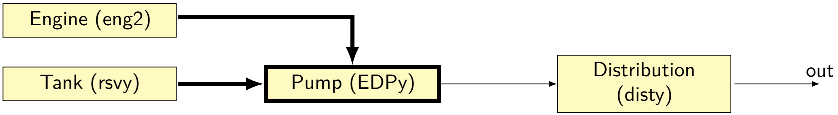



FIGURE – Yellow hydraulic system

# Build a fault tree

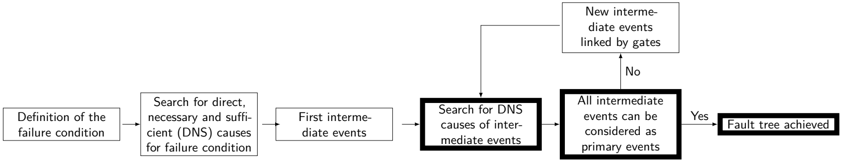



FIGURE – Fault tree construction process

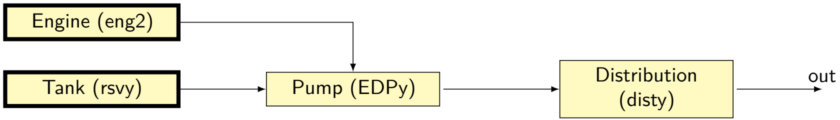
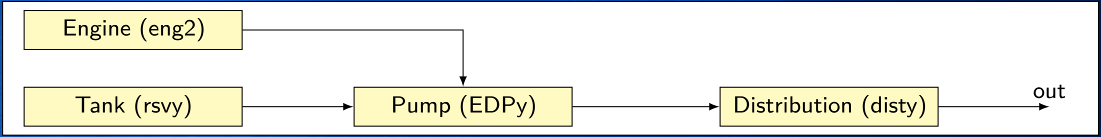


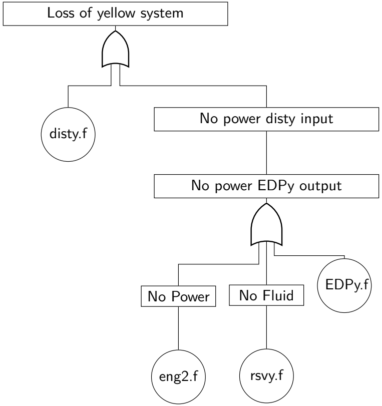

FIGURE – Yellow hydraulic system

According to the previous slides, build the fault tree of

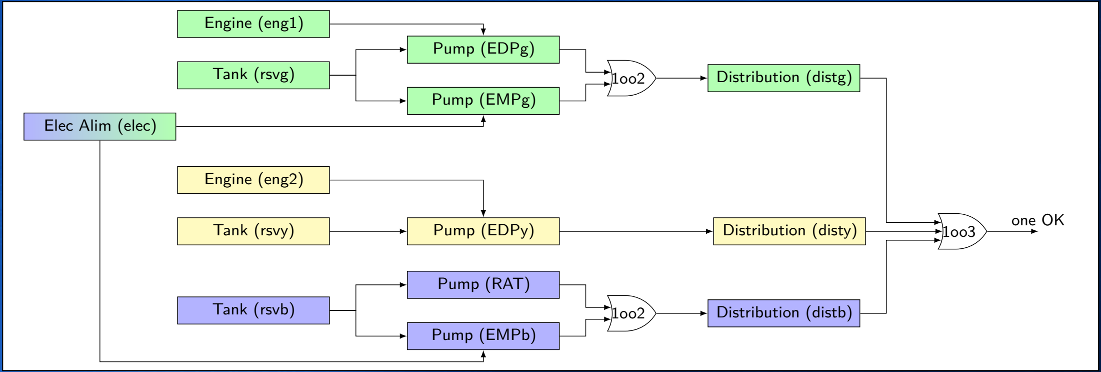
Loss of the yellow system



# Solution


Loss of yellow system
disty.f
No power disty input
No power EDPy output
No Power
No Fluid
EDPy.f
eng2.f
rsvy.f

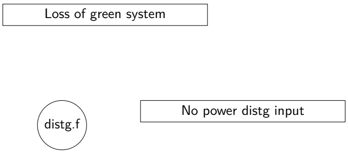
# Try to build the fault tree of Loss of the green system



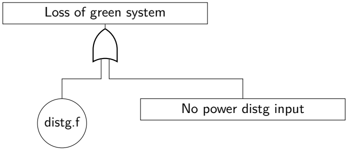
# Solution

Loss of green system

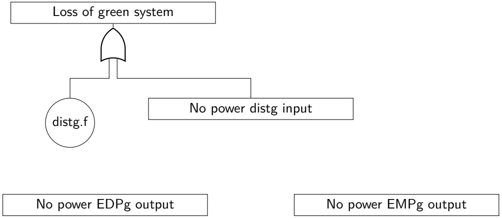
# Solution



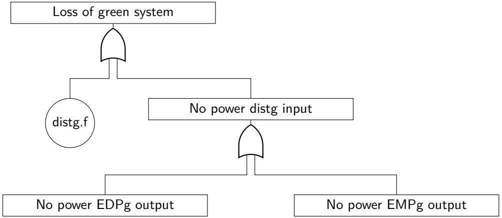
# Solution



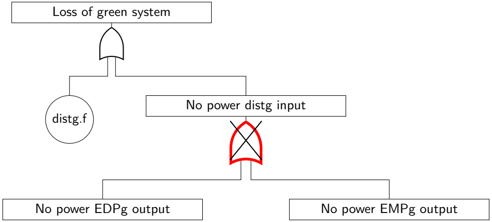
# Solution


Loss of green system
distg.f
No power distg input
No power EDPg output
No power EMPg output

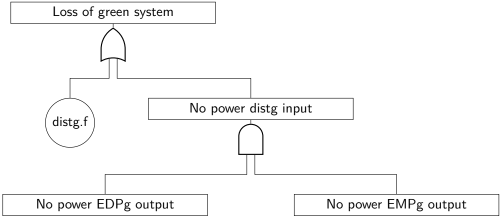
# Solution


Loss of green system
distg.f
No power distg input
No power EDPg output
No power EMPg output

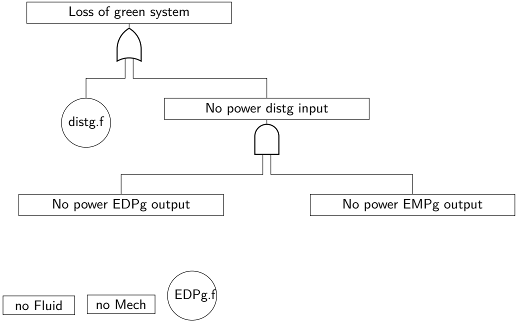
# Solution


Loss of green system
distg.f
No power distg input
No power EDPg output
No power EMPg output

# Solution

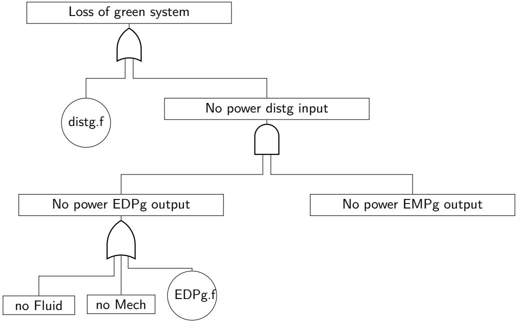

Loss of green system
distg.f
No power distg input
No power EDPg output
No power EMPg output

# Solution



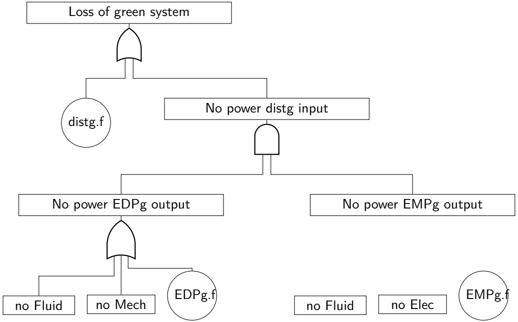
Loss of green system
distg.f
No power distg input
No power EDPg output
No power EMPg output
no Fluid
no Mech
EDPg.f

# Solution



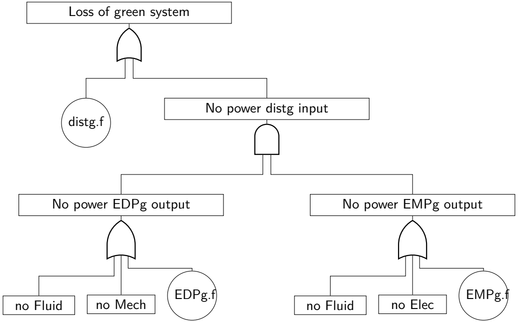
Loss of green system
distg.f
No power distg input
No power EDPg output
No power EMPg output
no Fluid
no Mech
EDPg.f

# Solution



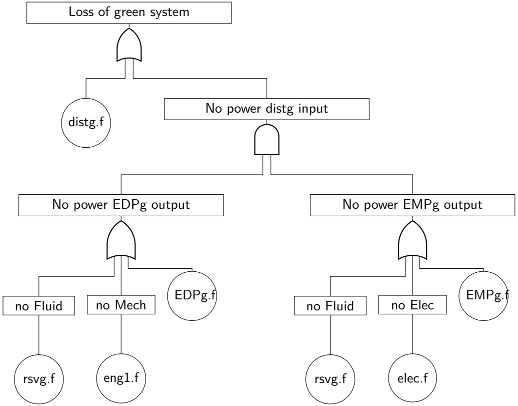
Loss of green system
distg.f
No power distg input
No power EDPg output
No power EMPg output
no Fluid
no Mech
EDPg.f
no Fluid
no Elec
EMPg.f

# Solution



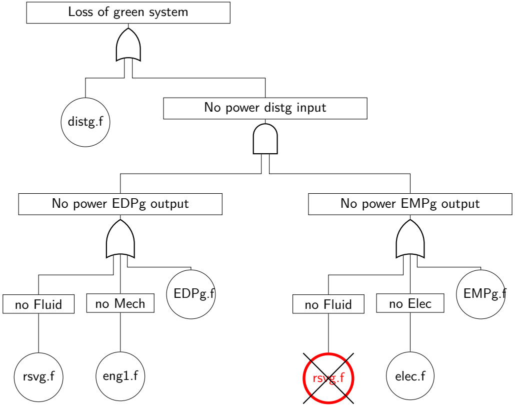
Loss of green system
distg.f
No power distg input
No power EDPg output
No power EMPg output
no Fluid
no Mech
EDPg.f
no Fluid
no Elec
EMPg.f

# Solution

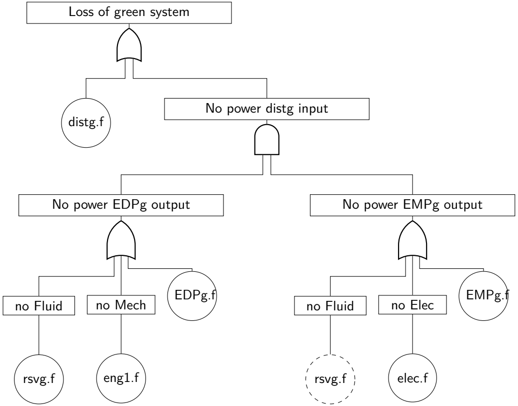


Loss of green system
distg.f
No power distg input
No power EDPg output
No power EMPg output
no Fluid
no Mech
EDPg.f
no Fluid
no Elec
EMPg.f
rsvg.f
eng1.f
rsvg.f
elec.f

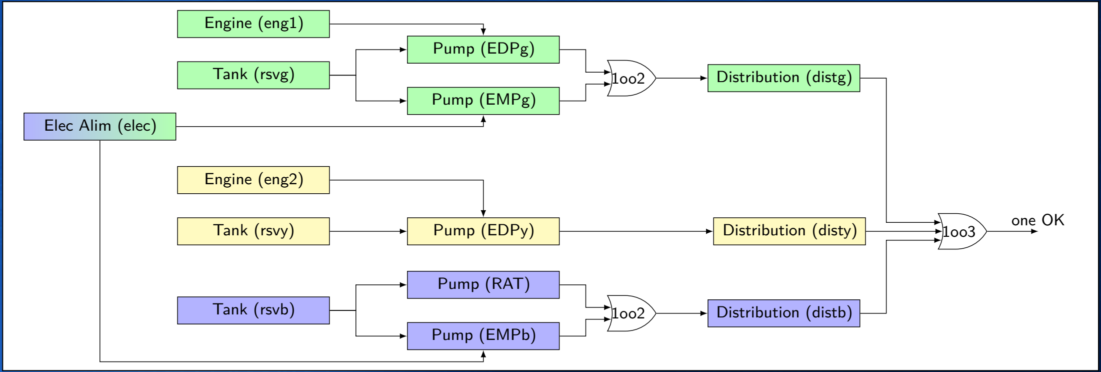
# Solution


Loss of green system
distg.f
No power distg input
No power EDPg output
No power EMPg output
no Fluid
no Mech
EDPg.f
no Fluid
no Elec
EMPg.f
rsvg.f
eng1.f
rsvg.f
elec.f

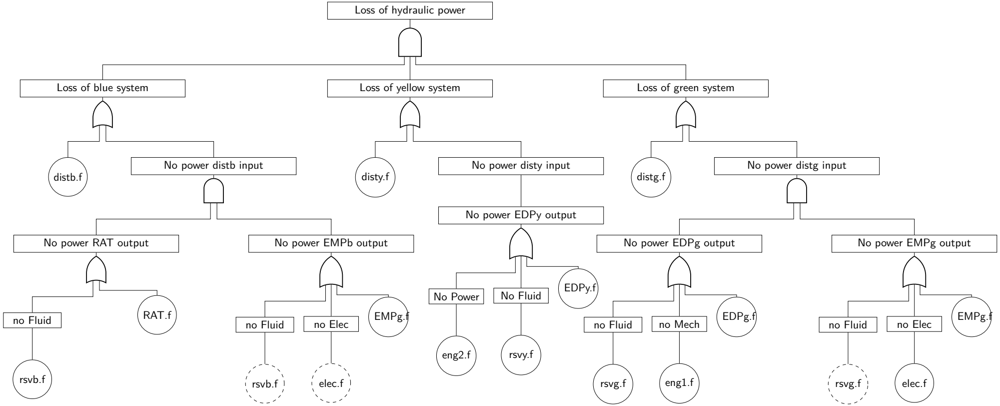
# Solution


Loss of green system
distg.f
No power distg input
No power EDPg output
No power EMPg output
no Fluid
no Mech
EDPg.f
no Fluid
no Elec
EMPg.f
rsvg.f
eng1.f
rsvg.f
elec.f

# Try to build the fault tree of
# Loss of hydraulic power



# Solution


Loss of hydraulic power
Loss of blue system
Loss of yellow system
Loss of green system
distb.f
No power distb input
disty.f
No power disty input
distg.f
No power distg input
No power EDPy output
No power RAT output
No power EMPb output
No power EDPg output
No power EMPg output
no Fluid
RAT.f
no Fluid
no Elec
EMPg.f
No Power
No Fluid
EDPy.f
no Fluid
no Mech
EDPg.f
no Fluid
no Elec
EMPg.f
rsvb.f
rsvb.f
elec.f
eng2.f
rsvy.f
rsvg.f
eng1.f
rsvg.f
elec.f

Now a recap !

# Today's lesson in 30"

- Dependability ⇒ ability to avoid unacceptable failures
- Acceptability defined by regulatory texts
- Dependability integrated trough safety process ⇒ What should we do and when
  - Assess system failures & severity ⇒ FHA
  - Analyse contribution of system's components failures to system failure ⇒ PSSA (FTA, . . .)
  - Quantify dependability with safety indicators $(R, \cdots)$

You understand highlighted terms
⇒ congratulations you've got the idea
Otherwise check out the slides !

How to perform safety assessment out of fault trees ?

Why using propositional logic in safety ?

To find the failure combinations leading to failure conditions

Is propositional logic expressive enough ?

Yes because fault trees are meant to model static systems : failure state does not depend on the order of occurrence of failures

Otherwise $\Rightarrow$ class on dynamic system modeling

# How to define a logic ?

**Syntax**

- Does the sentence belong to the language ?
  Does $a \looparrowright b$ belong to propositional logic ?
- Notions : propositions, connectors, formulae

**Semantics**

- What is the meaning of the sentence ?
  ***if*** *b* ***and*** *c* ***then*** *a* ***and*** *b* ***or not*** *a* ***and*** *c* is *always true* ?
- Notions : formulae valuations, validity, logical consequence

**Example of logic** Propositional logic, First-order logic, Temporal logic

# What can we write ?

$$
\begin{array}{lll}
\varphi ::= & proposition & \text{basic observations (ex :eng1.f)} \\
 & \mid \textbf{not } \varphi & \text{negation (ex :}\textbf{not}\text{ eng1.f)} \\
 & \mid \varphi_1 \textbf{ and } \varphi_2 & conjunction(ex : eng1.f \textbf{ and } eng2.f) \\
 & \mid \varphi_1 \textbf{ or } \varphi_2 & disjunction(ex : eng1.f \textbf{ or } eng2.f) \\
 & \mid \textbf{if } \varphi_1 \textbf{ then } \varphi_2 & implication(ex : \textbf{if } rsvg.f \textbf{ then } green.f) \\
 & \mid \varphi_1 = \varphi_2 & equivalence(ex : rsvg.f = green.f) \\
 & \mid (\varphi) & parenthesis(ex : (eng1.f))
\end{array}
$$

formulae sentences built using $\varphi$ rule

literal $proposition \mid \textbf{not } proposition$

# What does that mean ?

Define a valuation function $\llbracket\varphi\rrbracket \rightarrow \{\mathbf{T},\mathbf{F}\}$

$$
\begin{array}{lcl}
\llbracket proposition \rrbracket & = & v \in \{\mathbf{T},\mathbf{F}\} \\
 & & ex: \llbracket eng1.f \rrbracket = \mathbf{T} \text{ means "eng1 is lost" is true} \\
\llbracket \textbf{not } \varphi \rrbracket & = & \mathbf{T}\ iff\ \llbracket\varphi\rrbracket\ is\ \mathbf{F} \\
\llbracket \varphi_1 \textbf{ and } \varphi_2 \rrbracket & = & \mathbf{T}\ iff\ \llbracket\varphi_1\rrbracket\ is\ \mathbf{T}\ and\ \llbracket\varphi_2\rrbracket\ is\ \mathbf{T} \\
\llbracket \varphi_1 \textbf{ or } \varphi_2 \rrbracket & = & \mathbf{T}\ iff\ \llbracket\varphi_1\rrbracket\ is\ \mathbf{T}\ or\ \llbracket\varphi_2\rrbracket\ is\ \mathbf{T} \\
\llbracket \textbf{if } \varphi_1 \textbf{ then } \varphi_2 \rrbracket & = & \mathbf{T}\ iff\ \llbracket\varphi_1\rrbracket\ is\ \mathbf{F}\ or\ \llbracket\varphi_2\rrbracket\ is\ \mathbf{T} \\
\llbracket \varphi_1 = \varphi_2 \rrbracket & = & \mathbf{T}\ iff\ \llbracket\varphi_1\ \rrbracket and\ \llbracket\varphi_2\rrbracket\ are\ both\ \mathbf{T}\ or\ both\ \mathbf{F} \\
\llbracket (\varphi) \rrbracket & = & \llbracket\varphi\rrbracket
\end{array}
$$

# Satisfiability

## Satisfiability

A formula $\varphi$ is satisfiable iff it exists one valuation V of its propositions such that $[\![\varphi]\!]_V = \mathbf{T}$

## Satisfiability

Let $\varphi =$ eng1.f **and not** eng2.f

$\Rightarrow$ for $V = \{[\![eng1.f]\!] = \mathbf{T}, [\![eng2.f]\!] = \mathbf{F}\}$ we have $[\![\varphi]\!]_V = \mathbf{T}$

$\Rightarrow$ $\varphi$ is satisfiable

# Logical consequence

## Logical consequence

A formula $\varphi_2$ is a logical consequence of $\varphi_1$ iff for all valuation V such that $[\![\varphi_1]\!]_V = \mathbf{T}$ we have $[\![\varphi_2]\!]_V = \mathbf{T}$

## Logical consequence

Let $\varphi_2 =$ eng1.f and $\varphi_1 =$eng1.f **and not** eng2.f.

- $V = \{[\![eng1.f]\!] = \mathbf{T}, [\![eng2.f]\!] = \mathbf{F}\}$ is the only valuation satisfying $\varphi_1$
- $[\![\varphi_2]\!]_V = \mathbf{T}$

$\Rightarrow$ $\varphi_2$ is a logical consequence of $\varphi_1$

# Implicant

## Product

A product is a set of literals that does not contain both a variable and its negation.

## Product

$\{eng1.f,$**not** $eng2.f\}$ is a product

## Implicant

A product P is an implicant of formula $\varphi$ iff $\varphi$ is a logical consequence of P.

## Implicant

$\{eng1.f,$**not** $eng2.f\}$ is an implicant of *eng1.f* ***and not*** *eng2.f*

# Prime implicant

## Prime implicant

An implicant P of $\varphi$ is a prime implicant if there is no implicant P' of $\varphi$ such that P' is strictly included into P.

## Prime implicant

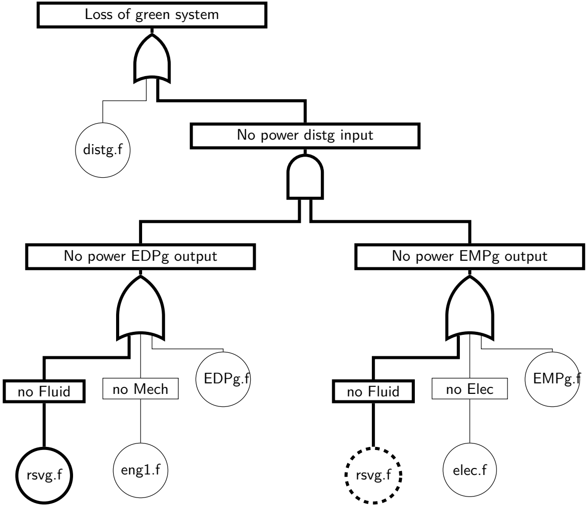
$\{eng1.f, \textbf{not } eng2.f\}$ is a prime implicant of *eng1.f* ***and not*** *eng2.f*

Fault tree ⇔ formula $\varphi$ describing the failure combinations leading to a failure condition

- accident can occur ⇔ $\varphi$ satisfiable
- situations where accident occurs ⇔ implicants of $\varphi$
- causes of the accident ⇔ prime implicants of $\varphi$

1. Is Loss of the green system possible ?
2. If yes, find a combination of failures where Loss of the green system occurs ?
3. Is your combination minimal ?
4. If possible, find prime implicants of size two, three.

# Satisfiability & Implicant



Can we compute automatically satisfiability and prime implicants of $\varphi$

# Shannon Decomposition

**ite operator**

$$\textbf{ite}(v, \varphi_1, \varphi_2) = \textbf{if } v \textbf{ then } \varphi_1 \textbf{ else } \varphi_2$$

**partial valuation** $\varphi|_{v=x}$ is the formula $\varphi$ where all occurrences of the proposition $v$ are replaced by the value $x \in \{\mathbf{T}, \mathbf{F}\}$.

## Shannon Decomposition

Let $\varphi$ be a formula containing a proposition $v$ then the Shannon decomposition on $v$ is :

$$\textbf{ite}(v, \varphi|_{v=\mathbf{T}}, \varphi|_{v=\mathbf{F}})$$

Shannon decomposition is applied recursively on the proposition contained in $\varphi$

# Shannon Decomposition

## Shannon Decomposition

Let $\varphi =$ eng1.f **and not** eng2.f, the step of the decomposition are :

1. Decompose on eng1.f :
   $\varphi|_{eng1.f=\mathbf{T}} = \textbf{not } eng2.f$
   $\varphi|_{eng1.f=\mathbf{F}} = \mathbf{F}$, so
   $\varphi = \textbf{ite}(eng1.f, \textbf{not } eng2.f, \mathbf{F})$
2. Decompose on eng2.f :
   $\textbf{not } eng2.f|_{eng2.f=\mathbf{T}} = \mathbf{F}$
   $\textbf{not } eng2.f|_{eng2.f=\mathbf{F}} = \mathbf{T}$,
   and $\mathbf{F}$ does not depend on $eng2.f$, so
   $\varphi = \textbf{ite}(eng1.f, \textbf{ite}(eng2.f, \mathbf{F}, \mathbf{T}), \mathbf{F})$

# Binary Decision Diagram (BDD)

*What's that ?*

## BDD

A BDD is a directed, oriented and acyclic graph encoding a formula $\varphi$. BDD contains :

- decision nodes labelled by a proposition $v$ own exactly two sons, the low son (resp high son) accessed through "0"(resp "1") edge is the root of the BDD encoding $\varphi|_{v=\mathbf{F}}$ (resp. $\varphi|_{v=\mathbf{T}}$)
- terminal 1 (resp. 0) encoding the formula $\mathbf{T}$ (resp. $\mathbf{F}$)

# Binary Decision Diagram (BDD)

$$\varphi = disty.f \textbf{ or } (EDPy.f \textbf{ or } eng2.f \textbf{ or } rsvgy.f)$$

$$\Downarrow Shannon\ decomposition$$

$$\textbf{ite}(disty.f, \textbf{T}, \qquad )$$

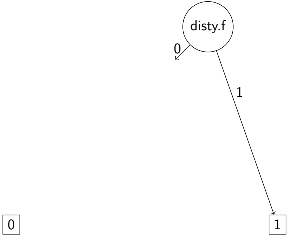



FIGURE – BDD of the loss of yellow system

# Binary Decision Diagram (BDD)

$$\varphi = disty.f \textbf{ or } (EDPy.f \textbf{ or } eng2.f \textbf{ or } rsvgy.f)$$

$$\Downarrow Shannon\ decomposition$$

$$\textbf{ite}(disty.f, \textbf{T}, \textbf{ite}(EDPy.f, \textbf{T}, \qquad ))$$

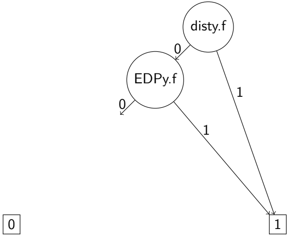



FIGURE – BDD of the loss of yellow system

# Binary Decision Diagram (BDD)

$$\varphi = disty.f \textbf{ or } (EDPy.f \textbf{ or } eng2.f \textbf{ or } rsvgy.f)$$

$$\Downarrow \textit{Shannon decomposition}$$

$$\textbf{ite}(disty.f, \textbf{T}, \textbf{ite}(EDPy.f, \textbf{T}, \textbf{ite}(rsvy.f, \textbf{T}, \qquad )))$$

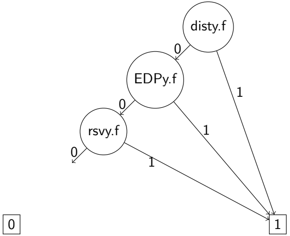



FIGURE – BDD of the loss of yellow system

# Binary Decision Diagram (BDD)

$$\varphi = disty.f \textbf{ or } (EDPy.f \textbf{ or } eng2.f \textbf{ or } rsvgy.f)$$

$$\Downarrow Shannon\ decomposition$$

$$\textbf{ite}(disty.f, \textbf{T}, \textbf{ite}(EDPy.f, \textbf{T}, \textbf{ite}(rsvy.f, \textbf{T}, eng2.f)))$$

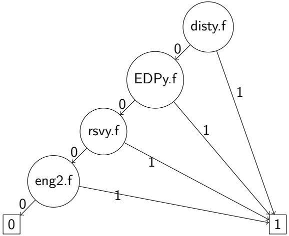



FIGURE – BDD of the loss of yellow system

# Binary Decision Diagram (BDD)

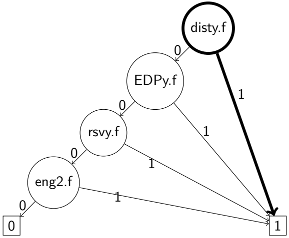



FIGURE – BDD of the loss of yellow system

Paths from root to 1 terminal ⇒ implicants

**Implicant**

Product $\{disty.f\}$ is an implicant of $\varphi$

# Binary Decision Diagram (BDD)

*Why introducing BDD ?*

- compact representation of formulae based on Shannon decomposition
- used to compute prime implicant and probabilities

## Morreale Decomposition Theorem

Let $\varphi = \mathbf{ite}(v, \varphi|_{v=\mathbf{T}}, \varphi|_{v=\mathbf{F}})$ then

$$PI(\varphi) = PI_{-} \cup PI_{\mathbf{T}} \cup PI_{\mathbf{F}}$$

where

$$
\begin{array}{lcl}
PI_{-} & = & PI(\varphi|_{v=\mathbf{T}} \textbf{ and } \varphi|_{v=\mathbf{F}}) \\
PI_{\mathbf{T}} & = & \{\{v\} \cup X | X \in PI(\varphi|_{v=\mathbf{T}}) \textbf{ and } X \notin PI_{-}\} \\
PI_{\mathbf{F}} & = & \{\{\textbf{not } v\} \cup X | X \in PI(\varphi|_{v=\mathbf{F}}) \textbf{ and } X \notin PI_{-}\} \\
PI(\mathbf{F}) & = & \emptyset \\
PI(\mathbf{T}) & = & \{\emptyset\}
\end{array}
$$

# Prime Implicant Computation

## Prime implicant computation

Compute PI of $\varphi = (a$ **and** $b)$ **or** $($**not** $a$ **and** $c)$ :

# Prime Implicant Computation

## Prime implicant computation

Compute PI of $\varphi = (a \textbf{ and } b) \textbf{ or } (\textbf{not } a \textbf{ and } c)$ :

1. $\varphi = \textbf{ite}(a, b, c)$
2. $PI(\varphi|_{a=\mathbf{T}}) = PI(b) = \{\{b\}\}$
3. $PI(\varphi|_{a=\mathbf{F}}) = PI(c) = \{\{c\}\}$
4. $PI_{-} = PI(\varphi|_{a=\mathbf{T}} \textbf{ and } \varphi|_{a=\mathbf{F}}) = PI(b \textbf{ and } c) = \{\{b, c\}\}$
5. $PI(\varphi|_{a=\mathbf{T}}) \cap PI_{-} = \emptyset$ so $PI_{\mathbf{T}} = \{\{a, b\}\}$
6. $PI(\varphi|_{a=\mathbf{F}}) \cap PI_{-} = \emptyset$ so $PI_{\mathbf{F}} = \{\{\textbf{not } a, c\}\}$
7. $PI(\varphi) = \{\{a, b\}, \{\textbf{not } a, c\}, \{b, c\}\}$

What does $\{$**not** $a, c\}$ implicant mean ?

# Minimal cutsets computation

Negative literals in prime implicants

$\Downarrow$

Some components must "work" to trigger the failure condition

$\Downarrow$

**No miracle rule :** Considering that component failure can mitigate the failure condition should be avoided

$\Downarrow$ Pessimistic approach (safe)

Minimal cutsets = Positive part of prime implicants

# Minimal cutsets computation

## Cut sets computation

Let $\varphi = \mathbf{ite}(v, \varphi|_{v=\mathbf{T}}, \varphi|_{v=\mathbf{F}})$ then

$$MCS(\varphi) = MCS_{\mathbf{F}} \cup MCS_{\mathbf{T}}$$

where

$$\begin{aligned}
MCS_{\mathbf{F}} &= \{X \mid X \in MCS(\varphi|_{v=\mathbf{F}})\} \\
MCS_{\mathbf{T}} &= \{\{v\} \cup X \mid X \in MCS(\varphi|_{v=\mathbf{T}}) \text{ and } X \notin MCS_{\mathbf{F}}\} \\
MCS(\mathbf{F}) &= \emptyset \\
MCS(\mathbf{T}) &= \{\emptyset\}
\end{aligned}$$

# Minimal cutsets computation

## Minimal cutsets computation

Compute MCS of $\varphi = (a$ **and** $b)$ **or** $($**not** $a$ **and** $c)$ :

# Minimal cutsets computation

## Minimal cutsets computation

Compute MCS of $\varphi = (a \textbf{ and } b) \textbf{ or } (\textbf{not } a \textbf{ and } c)$ :

1. $\varphi = \textbf{ite}(a, b, c)$
2. $MCS(\varphi|_{a=\mathsf{T}}) = MCS(b) = \{\{b\}\}$
3. $MCS(\varphi|_{a=\mathsf{F}}) = MCS(c) = \{\{c\}\}$
4. $MCS_{\mathsf{F}} = MCS(\varphi|_{a=\mathsf{F}}) = \{\{c\}\}$
5. $MCS(\varphi|_{a=\mathsf{T}}) \cap MCS_{\mathsf{F}} = \emptyset$ so $MCS_{\mathsf{T}} = \{\{a, b\}\}$
6. $MCS(\varphi) = \{\{a, b\}, \{c\}\}$

$$PI(\varphi)=\{\{a,b\},\{\textbf{not } a,c\},\{b,c\}\}$$
$$\Downarrow \text{ Pessimism}$$
$$MCS(\varphi) = \{\{a,b\},\{c\}\}$$

# Probability computation

Option 1 : Approximate computation $MCS$ : minimal cutsets for $FC$, and $p(event)$ probability of failure for primary events :

$$p(FC) = \sum_{cut \in MCS} \prod_{event \in cut} p(event)$$

## Approximate computation

Let MCS=$\{\{a,b\},\{c\}\}$ be the minimal cutsets for FC :

$$p_{approx}(FC) = p(a)p(b) + p(c)$$

# Probability computation

Option 2 : Exact computation Shannon decomposition :

$$\begin{aligned} p(\mathbf{ite}(v, \varphi|_{v=\mathbf{T}}, \varphi|_{v=\mathbf{F}})) &= p(v)p(\varphi|_{v=\mathbf{T}}) + (1 - p(v))p(\varphi|_{v=\mathbf{F}}) \\ p(\mathbf{T}) &= 1 \\ p(\mathbf{F}) &= 0 \end{aligned}$$

## Exact computation

Let $\varphi = \mathbf{ite}(a, b, c)$ be the Shannon decomposition for FC :

$$p(FC) = p(a)p(b) + (1 - p(a))p(c)$$

Pessimism introduced by approximation $\left(p(x) = 10^{-3}\right)$ :

$$\frac{p_{approx}(FC) - p(FC)}{p(FC)} = \frac{p(a)p(c)}{p(a)p(b) + (1 - p(a))p(c)} \simeq .1\%$$

OK but is the hydraulic system is safe or not ?

# Safety objectives (Reminder)

| severity | qualitative requirement | quantitative requirement |
|---|---|---|
| Catastrophic | order $\geq 2$ | $\overline{\Lambda} \leq 10^{-9} / flight\ hour$ |
| Hazardous | order $\geq 1$ | $\overline{\Lambda} \leq 10^{-7} / flight\ hour$ |
| Major | order $\geq 1$ | $\overline{\Lambda} \leq 10^{-5} / flight\ hour$ |
| Minor | order $\geq 1$ | $\overline{\Lambda} \leq 10^{-3} / flight\ hour$ |

TABLE – Acceptability matrix

# Order and Mean failure rate

## Order

The order is the minimal cardinality of MCS

## Order

The order of $MCS = \{\{a, b\}, \{c\}\}$ is 1

## Mean failure rate

Mean failure rate is $\overline{\Lambda}(T) \underset{0}{\sim} \frac{\overline{R(T)}}{T}$

## Mean failure rate

The mean failure rate of $MCS = \{\{a, b\}, \{c\}\}$ at T is $\overline{\Lambda}(T) \underset{0}{\sim} \frac{p(a)p(b)+p(c)}{T}$

# Requirements verification

⚠ We assume that primary events are independent

1. Determine the failure conditions and their severity (from FHA)
2. Build the fault trees for each failure condition
3. Compute the minimal cutsets
4. Qualitative verification : Compute the order and compare it to the required bound
5. Quantitative verification : Compute the probability and compare it to the required bound

## Requirements verification

Check the requirements for yellow system

1. our failure condition "loss of yellow system" is Minor
   $\Rightarrow$ order $\geq$ 1 and $p(FC) \leq 10^{-3}$
2. fault tree (cf slide 76)
3. the minimal cutsets are $MCS = \{\{disty.f\}, \{eng2.f\}, \{EDPy.f\}, \{rsvy.f\}\}$
4. the order is 1 $\Rightarrow$ qualitative requirement OK
5. let assume that $p(event) = 10^{-4}$ and $T = 1$ for all events then :

$$\begin{aligned}\overline{\Lambda}(FC) &= p(disty.f) + p(EDPy.f) + p(eng2.f) + p(rsvy.f) \\ &= 4.10^{-4} \Rightarrow \text{quantitative requirement OK}\end{aligned}$$

Check the hydraulic system considering Loss of the green system is Minor

# Solution

1. our failure condition "loss of green system" is Minor
   $\Rightarrow$ order $\geq$ 1 and $p(FC) \leq 10^{-3}$
2. fault tree (cf slide 78)
3. the minimal cutsets are :

$$MCS = \left\{ \begin{array}{ll} \{distg.f\}, & \{rsvg.f\}, \\ \{EMPg.f, EDPg.f\}, & \{EMPg.f, eng1.f\}, \\ \{elec.f, EDPg.f\}, & \{elec.f, eng1.f\} \end{array} \right\}$$

4. the order is 1 $\Rightarrow$ qualitative requirement OK
5. let assume that $p(event) = 10^{-4}/FH$ for all events then :

$$\begin{array}{rcl} \overline{\Lambda}(FC) & = & 2.10^{-4} + 4.10^{-8} \\ & \simeq & 2.10^{-4} \Rightarrow \text{quantitative requirement OK} \end{array}$$

Now a Recap

# Today's lesson in 30"

Safety assessment process

1. Identify the failure conditions
2. Find the safety objectives (slide 111)
3. If the system is static build the fault tree (slide 74)
4. Compute the order of the cutsets (slide 111)
5. Compute the probability out of minimal cutsets (slide 108)
6. Compare it to the objectives

You understand highlighted terms
$\Rightarrow$ congratulations you've got the idea
Otherwise check out the slides !

# Bibliography I

[ALRL04] Algirdas Avizienis, J-C Laprie, Brian Randell, and Carl Landwehr.
Basic concepts and taxonomy of dependable and secure computing.
*IEEE transactions on dependable and secure computing*, 1(1) :11–33, 2004.

[BDS11] Pierre Bieber, Rémi Delmas, and Christel Seguin.
Dalculus–theory and tool for development assurance level allocation.
In *Computer Safety, Reliability, and Security*, pages 43–56. Springer, 2011.

[ISO10] ISO.
ISO-26262 -Road vehicles – Functional safety, 2010.

[SAE96] SAE.
Aerospace Recommended Practices 4761 - guidelines and methods for conducting the safety assessment process on civil airborne systems and equipment, 1996.

[SAE10] SAE.
Aerospace Recommended Practices 4754a - Development of Civil Aircraft and Systems, 2010.

# Thank you

ONERA

THE FRENCH AEROSPACE LAB

www.onera.fr

# Requirements verification

⚠ We assume that primary events are independent

1. Determine the failure conditions and their severity (from FHA)
2. Build the fault trees for each failure condition
3. Compute the minimal cutsets
4. Qualitative verification : Compute the order and compare it to the required bound
5. Quantitative verification : Compute the probability and compare it to the required bound

What if some primary events are not independent (tire burst, engine burst,...) ?

# Deal with dependencies

What could cause the simultaneous failure of several components ?

- Adversary conditions : overheat, electromagnetic perturbations, ...
- Destruction of a whole zone : engine burst, in-flight fire,...
- But also : implementation common mode (functions depending on the same equipments), specification errors, systematic development errors,...

What are the consequences ?

- Possible violation of safety objective
  $\Rightarrow$ Identify and analyze common mode during the Common Cause Analysis (CCA)

# Deal with dependencies

## Example (Dependencies impact)

Minimal cut $C = \{a, b\}$ for a catastrophic FC, if a and b are not independent (triggered by $d$) :

- ⇒ $C \rightarrow \{d\}$
- ⇒ Order goes from 2 to 1
- ⚠ System does not fulfil requirements

# Deal with dependencies

Event in MCS shall be independent to avoid that their implementation introduces a common mode reducing the size of the MCS under the order requirement.

$\Downarrow$

Define the segregation requirements to ensure independence

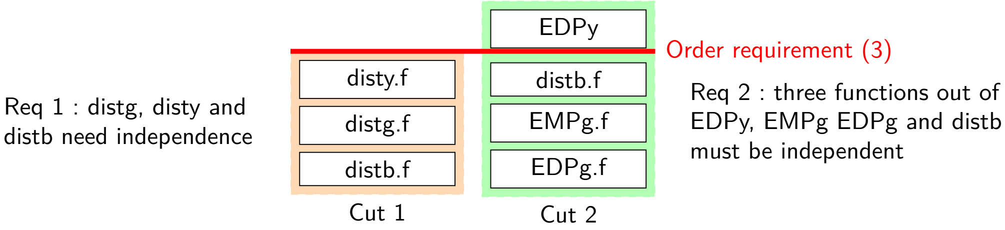



FIGURE – Independence requirements for Total hydraulic system loss

# Deal with dependencies

1. Define the independence groups :
   - Two members of the same group are not independent
   - Two members of different groups are independent

## Example (Independence groups)

Let consider that component can be in three spacial zones, each zone can be completely destroyed by an engine burst, the independent groups are :

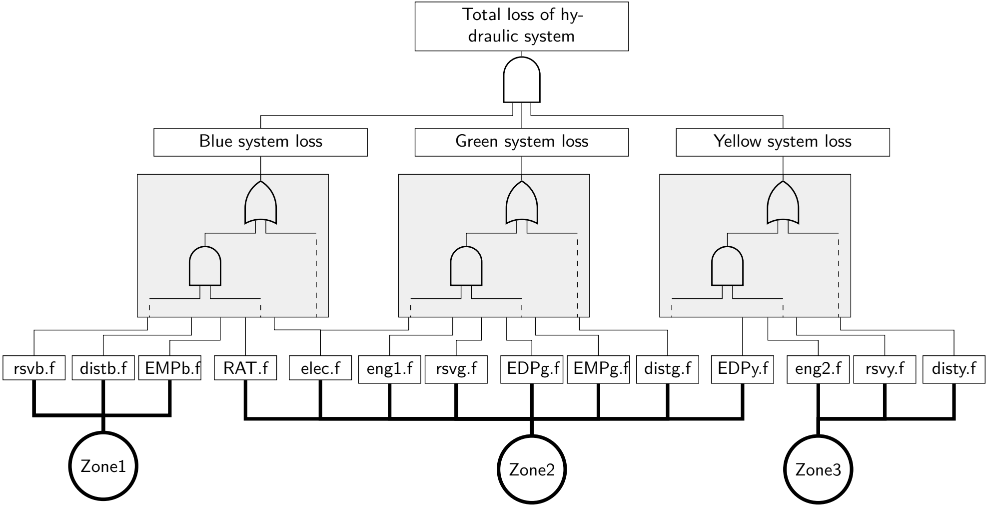
| Zone 1 | Zone 2 | Zone 3 |
|---|---|---|
| rsvb, distb, EMPb | RAT, elec, eng1, rsvg, EDPg, EMPg, distg, EDPy | rsvy, eng2, disty |

# Deal with dependencies

1. Define the independence groups :
   - Two members of the same group are not independent
   - Two members of different groups are independent
2. Modify the fault tree :
   - transform primary event as intermediate events
   - create a primary event per group
   - link intermediate event to the corresponding group
3. Compute the cutsets
4. Check the requirements

Considering the previous independence groups, is the system safe ?

# Deal with dependencies : Example



# Deal with dependencies : Example

$$
\begin{array}{cccc}
\{EDPg.f, & RAT.f, & elec.f, & EDPy.f\} \\
\Downarrow & \Downarrow & \Downarrow & \Downarrow \\
\{Zone2, & Zone2, & Zone2, & Zone2\}
\end{array}
$$

$$\Downarrow Minimisation$$

$$\{Zone2\}$$

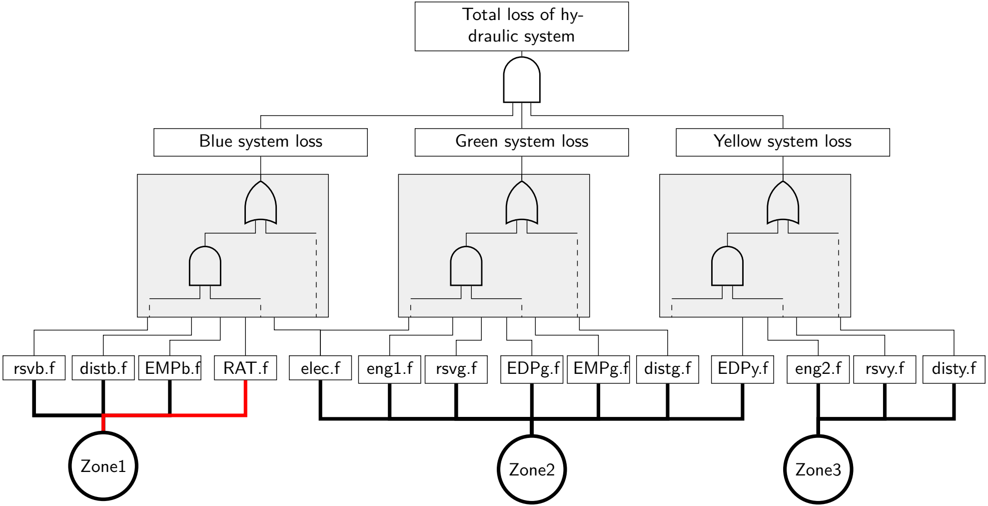
$$\Downarrow$$

Order is 1

$$\Downarrow$$

KO since "Total loss of hydraulic system" is Catastrophic so requirement is 2

# Deal with dependencies : Example



# Deal with dependencies : Example

$$\{\{EDPg.f,\ RAT.f,\ elec.f,\ EDPy.f\},\cdots\}$$
$$\Downarrow Analysis$$
$$\{\{Zone1,\ Zone2\}\}$$

$$\Downarrow$$

Order is 2

$$\Downarrow$$

OK since "Total loss of hydraulic system" is Catastrophic so requirement is 2

# Minimal cutsets computation

What could cause the simultaneous failure of several components ?

- Adversary conditions : overheat, electromagnetic perturbations, ...
- Destruction of a whole zone : engine burst, in-flight fire,...
- But also : implementation common mode (functions depending on the same equipments), specification errors, systematic development errors,...

# Minimal cutsets computation

What could cause the simultaneous failure of several components ?

- Adversary conditions : overheat, electromagnetic perturbations, ... ⇒ Random faults
- Destruction of a whole zone : engine burst, in-flight fire,... ⇒ Random faults
- But also : implementation common mode (functions depending on the same equipments), specification errors, systematic development errors,... ⇒ Systematic faults

Acceptability cannot be based on probability assessment !
⇒ ensure a level of confidence in development correctness

# Minimal cutsets computation

DAL Development Assurance Level (ARP4754) is the level (from E to A) of rigor of development assurance tasks performed on functions and items (software, hardware) whose fault result

Warning :

- DAL can be associated with
  - Functions : FDAL
  - Items : IDAL
- For each DAL level, assurance activities are listed in :
  - ARP4754 for FDAL
  - DO178 (SW) and DO254 (HW) for IDAL

# Assurance Activities Examples

| | Objective | | Applicability | | | |
|---|---|---|---|---|---|---|
| | Description | Ref | A | B | C | D |
| 1 | Software high-level requirements comply with system requirements. | 6.3.1a | I | I | R | R |
| 2 | High-level requirements are accurate and consistent. | 6.3.1b | I | I | R | R |
| 3 | High-level requirements are compatible with target computer. | 6.3.1c | R | R | | |

- High DAL level ⇒ great number of assurance activities
  ⇒ costly
  ⇒ minimize the DAL of software and hardware

# DAL Allocation : Basic Allocation

Based on the severities of the FCs that function fault contributes to.

| Sev(FC) | DAL(FC) |
|---|---|
| CAT | A |
| HAZ | B |
| MAJ | C |
| MIN | D |
| NSE | E |

TABLE – Link between severity and DAL

What does "the severities of the FCs that function fault $f$ contributes to" mean ?

$\Rightarrow$ the severities of the FCs whose MCS contains $f$

# DAL Allocation : Basic Allocation

**Context**
- Let $fc_1$ (resp $fc_2$) be a failure condition of severity HAZ (resp. MAJ)
- Let $MCS_1 = \{\{f_1, f_2, f_4\}, \{f_3\}\}$ and $MCS_2 = \{\{f_1, f_3\}\}$

**Question** What is the basic DAL of $f_1$ ?

# DAL Allocation : Basic Allocation

**Context**
- Let $fc_1$ (resp $fc_2$) be a failure condition of severity HAZ (resp. MAJ)
- Let $MCS_1 = \{\{f_1, f_2, f_4\}, \{f_3\}\}$ and $MCS_2 = \{\{f_1, f_3\}\}$

**Question** What is the basic DAL of $f_1$ ?

**Answer** $f_1$ contained in $MCS_1$ and $MCS_2$ so
$DAL(f_1) = worst(DAL(fc_1), DAL(fc_2)) = DAL(HAZ) = B$

**Question** What is the basic DAL of $f_2$ ?

# DAL Allocation : Basic Allocation

Context
- Let $fc_1$ (resp $fc_2$) be a failure condition of severity HAZ (resp. MAJ)
- Let $MCS_1 = \{\{f_1, f_2, f_4\}, \{f_3\}\}$ and $MCS_2 = \{\{f_1, f_3\}\}$

Question What is the basic DAL of $f_1$ ?

Answer $f_1$ contained in $MCS_1$ and $MCS_2$ so $DAL(f_1) = worst(DAL(fc_1), DAL(fc_2)) = DAL(HAZ) = B$

Question What is the basic DAL of $f_2$ ?

Answer $f_2$ contained only in $MCS_1$ so $DAL(f_2) = worst(DAL(fc_1)) = DAL(HAZ) = B$

# DAL Allocation : Degradation rules

Designer can downgrade the basic DAL $basic$ of a function using independence, the allocation must fulfill the following rules :

Rule 1 $basic$ can be degraded at most by two levels

Rule 2 For all cuts $\{f_1, \cdots, f_n\} \in MCS_{fc}$ where $f_1, \cdots, f_n$ are independent, either :

- Option 1 : it exists $f_i$ such that $DAL(f_i) = basic$
- Option 2 : it exists $f_i, f_j$ such that $DAL(f_i) = DAL(f_j) = basic - 1$

# DAL Allocation : Degradation rules

Suppose $f_1, f_2, f_3$ and $f_4$ are independent and cost : DAL A $= 20$, DAL B $= 15$, DAL C $= 5$, DAL D $= 4$, DAL E $= 0$

| basic DAL | cuts | DAL | | | | Option |
|---|---|---|---|---|---|---|
| | | $f_1$ | $f_2$ | $f_3$ | $f_4$ | |

# DAL Allocation : Degradation rules

Suppose $f_1, f_2, f_3$ and $f_4$ are independent and cost : DAL A $= 20$, DAL B $= 15$, DAL C $= 5$, DAL D $= 4$, DAL E $= 0$

| basic DAL | cuts | DAL | | | | Option |
|---|---|---|---|---|---|---|
| | | $f_1$ | $f_2$ | $f_3$ | $f_4$ | |
| B | $\{f_1, f_2, f_4\}$ | $\geq$ B | $\geq$ D | - | $\geq$ D | 1 |

# DAL Allocation : Degradation rules

Suppose $f_1, f_2, f_3$ and $f_4$ are independent and cost : DAL A $=$ 20, DAL B $=$ 15, DAL C $=$ 5, DAL D $=$ 4, DAL E $=$ 0

| basic DAL | cuts | DAL | | | | Option |
|---|---|---|---|---|---|---|
| | | $f_1$ | $f_2$ | $f_3$ | $f_4$ | |
| B | $\{f_1, f_2, f_4\}$ | $\geq$ B | $\geq$ D | - | $\geq$ D | 1 |
| | $\{f_3\}$ | - | - | $\geq$ B | - | - |

# DAL Allocation : Degradation rules

Suppose $f_1, f_2, f_3$ and $f_4$ are independent and cost : DAL A $=$ 20, DAL B $=$ 15, DAL C $=$ 5, DAL D $=$ 4, DAL E $=$ 0

| basic DAL | cuts | DAL | | | | Option |
|---|---|---|---|---|---|---|
| | | $f_1$ | $f_2$ | $f_3$ | $f_4$ | |
| B | $\{f_1, f_2, f_4\}$ | $\geq$ B | $\geq$ D | - | $\geq$ D | 1 |
| | $\{f_3\}$ | - | - | $\geq$ B | - | - |
| C | $\{f_1, f_3\}$ | $\geq$ C | - | $\geq$ E | - | 1 |

# DAL Allocation : Degradation rules

Suppose $f_1, f_2, f_3$ and $f_4$ are independent and cost : DAL A $= 20$, DAL B $= 15$, DAL C $= 5$, DAL D $= 4$, DAL E $= 0$

| basic DAL | cuts | DAL | | | | Option |
|---|---|---|---|---|---|---|
| | | $f_1$ | $f_2$ | $f_3$ | $f_4$ | |
| B | $\{f_1, f_2, f_4\}$ | $\geq$ B | $\geq$ D | - | $\geq$ D | 1 |
| | $\{f_3\}$ | - | - | $\geq$ B | - | - |
| C | $\{f_1, f_3\}$ | $\geq$ C | - | $\geq$ E | - | 1 |
| Result | | $\geq$ B | $\geq$ D | $\geq$ B | $\geq$ D | |
| Cost | | 38 | | | | |

Is it the cheapest option ?

$\Rightarrow$ Let's try again !

# DAL Allocation : Degradation rules

Suppose $f_1, f_2, f_3$ and $f_4$ are independent and cost : DAL A $= 20$, DAL B $= 15$, DAL C $= 5$, DAL D $= 4$, DAL E $= 0$

| basic DAL | cuts | DAL | | | | Option |
|---|---|---|---|---|---|---|
| | | $f_1$ | $f_2$ | $f_3$ | $f_4$ | |

# DAL Allocation : Degradation rules

Suppose $f_1, f_2, f_3$ and $f_4$ are independent and cost : DAL A $= 20$, DAL B $= 15$, DAL C $= 5$, DAL D $= 4$, DAL E $= 0$

| basic DAL | cuts | DAL | | | | Option |
|---|---|---|---|---|---|---|
| | | $f_1$ | $f_2$ | $f_3$ | $f_4$ | |
| B | $\{f_1, f_2, f_4\}$ | $\geq$ C | $\geq$ C | - | $\geq$ D | 2 |

# DAL Allocation : Degradation rules

Suppose $f_1, f_2, f_3$ and $f_4$ are independent and cost : DAL A $= 20$, DAL B $= 15$, DAL C $= 5$, DAL D $= 4$, DAL E $= 0$

| basic DAL | cuts | DAL | | | | Option |
|---|---|---|---|---|---|---|
| | | $f_1$ | $f_2$ | $f_3$ | $f_4$ | |
| B | $\{f_1, f_2, f_4\}$ | $\geq$ C | $\geq$ C | - | $\geq$ D | 2 |
| | $\{f_3\}$ | - | - | $\geq$ B | - | - |

# DAL Allocation : Degradation rules

Suppose $f_1, f_2, f_3$ and $f_4$ are independent and cost : DAL A $=$ 20, DAL B $=$ 15, DAL C $=$ 5, DAL D $=$ 4, DAL E $=$ 0

| basic DAL | cuts | DAL | | | | Option |
|---|---|---|---|---|---|---|
| | | $f_1$ | $f_2$ | $f_3$ | $f_4$ | |
| B | $\{f_1, f_2, f_4\}$ | $\geq$ C | $\geq$ C | - | $\geq$ D | 2 |
| | $\{f_3\}$ | - | - | $\geq$ B | - | - |
| C | $\{f_1, f_3\}$ | $\geq$ E | - | $\geq$ C | - | 1 |

# DAL Allocation : Degradation rules

Suppose $f_1, f_2, f_3$ and $f_4$ are independent and cost : DAL A $= 20$, DAL B $= 15$, DAL C $= 5$, DAL D $= 4$, DAL E $= 0$

| basic DAL | cuts | DAL | | | | Option |
|---|---|---|---|---|---|---|
| | | $f_1$ | $f_2$ | $f_3$ | $f_4$ | |
| B | $\{f_1, f_2, f_4\}$ | $\geq$ C | $\geq$ C | - | $\geq$ D | 2 |
| | $\{f_3\}$ | - | - | $\geq$ B | - | - |
| C | $\{f_1, f_3\}$ | $\geq$ E | - | $\geq$ C | - | 1 |
| Result | | $\geq$ C | $\geq$ C | $\geq$ B | $\geq$ D | |
| Cost | | 29 | | | | |

Whoopsie, $f_1$ and $f_3$ are not independent

$\Rightarrow$ Any impact on last allocation ?

# DAL Allocation : Degradation rules

$f_1, f_3$ not independent $\Rightarrow$ cannot apply downgradation rules on $\{f_1, f_3\}$.

| basic DAL | cuts | DAL | | | | Option |
|---|---|---|---|---|---|---|
| | | $f_1$ | $f_2$ | $f_3$ | $f_4$ | |

# DAL Allocation : Degradation rules

$f_1, f_3$ not independent $\Rightarrow$ cannot apply downgradation rules on $\{f_1, f_3\}$.

| basic DAL | cuts | DAL | | | | Option |
|---|---|---|---|---|---|---|
| | | $f_1$ | $f_2$ | $f_3$ | $f_4$ | |
| B | $\{f_{1,3}, f_2, f_4\}$ | $\geq$ C | $\geq$ C | - | $\geq$ D | 2 |

# DAL Allocation : Degradation rules

$f_1, f_3$ not independent $\Rightarrow$ cannot apply downgradation rules on $\{f_1, f_3\}$.

| basic DAL | cuts | DAL | | | | Option |
|---|---|---|---|---|---|---|
| | | $f_1$ | $f_2$ | $f_3$ | $f_4$ | |
| B | $\{f_{1,3}, f_2, f_4\}$ | $\geq$ C | $\geq$ C | - | $\geq$ D | 2 |
| | $\{f_3\}$ | - | - | $\geq$ B | - | - |

# DAL Allocation : Degradation rules

$f_1, f_3$ not independent $\Rightarrow$ cannot apply downgradation rules on $\{f_1, f_3\}$.

| basic DAL | cuts | DAL | | | | Option |
|---|---|---|---|---|---|---|
| | | $f_1$ | $f_2$ | $f_3$ | $f_4$ | |
| B | $\{f_{1,3}, f_2, f_4\}$ | ≥ C | ≥ C | - | ≥ D | 2 |
| | $\{f_3\}$ | - | - | ≥ B | - | - |
| C | $\{f_1, f_3\}$ | ≥ C | - | ≥ C | - | - |

# DAL Allocation : Degradation rules

$f_1, f_3$ not independent $\Rightarrow$ cannot apply downgradation rules on $\{f_1, f_3\}$.

| basic DAL | cuts | DAL | | | | Option |
|---|---|---|---|---|---|---|
| | | $f_1$ | $f_2$ | $f_3$ | $f_4$ | |
| B | $\{f_{1,3}, f_2, f_4\}$ | $\geq$ C | $\geq$ C | - | $\geq$ D | 2 |
| | $\{f_3\}$ | - | - | $\geq$ B | - | - |
| C | $\{f_1, f_3\}$ | $\geq$ C | - | $\geq$ C | - | - |
| Result | | $\geq$ C | $\geq$ C | $\geq$ B | $\geq$ D | |
| Cost | | | 29 | | | |

Your turn ! Allocate the DAL of green system

# DAL Allocation : Exercise

Assume FC is Major, all independent except $EMP$ and $eng1$, and DAL cost for $EDP$ and $elec$ is twice the initial cost.

| basic DAL | cuts | DAL | | | | | | Option |
|---|---|---|---|---|---|---|---|---|
| | | $dist$ | $rsv$ | $EMP$ | $EDP$ | $eng1$ | $elec$ | |
| ? | $\{dist\}$ | $\geq ?$ | - | - | - | - | - | ? |
| | $\{rsv\}$ | - | $\geq ?$ | - | - | - | - | ? |
| | $\{EMP, EDP\}$ | - | - | $\geq ?$ | $\geq ?$ | - | - | ? |
| | $\{EMP, eng1\}$ | - | - | $\geq ?$ | - | $\geq ?$ | - | ? |
| | $\{elec, EDP\}$ | - | - | - | $\geq ?$ | - | $\geq ?$ | ? |
| | $\{elec, eng1\}$ | - | - | - | - | $\geq ?$ | $\geq ?$ | ? |
| Result | | $\geq ?$ | $\geq ?$ | $\geq ?$ | $\geq ?$ | $\geq ?$ | $\geq ?$ | |
| Cost | | ? | | | | | | |

# DAL Allocation : Exercise

Assume FC is Major, all independent except $EMP$ and $eng1$, and DAL cost for $EDP$ and $elec$ is twice the initial cost.

| basic DAL | cuts | DAL | | | | | | Option |
|---|---|---|---|---|---|---|---|---|
| | | $dist$ | $rsv$ | $EMP$ | $EDP$ | $eng1$ | $elec$ | |
| C | $\{dist\}$ | ≥ C | - | - | - | - | - | - |
| | $\{rsv\}$ | - | ≥ C | - | - | - | - | - |
| | $\{EMP, EDP\}$ | - | - | ≥ C | ≥ E | - | - | 1 |
| | $\{EMP, eng1\}$ | - | - | ≥ C | - | ≥ C | - | - |
| | $\{elec, EDP\}$ | - | - | - | ≥ D | - | ≥ D | 2 |
| | $\{elec, f_{eng1}\}$ | - | - | - | - | ≥ C | ≥ E | 1 |
| Result | | ≥ C | ≥ C | ≥ C | ≥ D | ≥ C | ≥ D | |
| Cost | | 36 | | | | | | |

It's a lot of rules, is there another way to find an optimal allocation ?

# DAL Allocation : Automatic allocation

DAL, FDAL & IDAL allocation problem is combinatorial problem :

- Real systems : hundreds of FCs & $MCS$ with thousands of cuts !
  $\Rightarrow$ Nearly impossible to find optimal allocation by hand
- Presented rules are simplification of real allocation process (deal with failure modes, ...)
  $\Rightarrow$ Use constraint programming to allocate DAL [BDS11] for instance SAT or IDP).

# DAL Allocation : Automatic allocation

Automatic problem generator needs :

- the MCS of FCs,
- the FC severity,
- a partial or total independence relation,
- a cost function.

Result of the solver :

1. an optimal DAL allocation of function/items,
2. the completed independence relation used to compute the DAL allocation,
3. the downgrading options used.

# DAL Allocation : Ask to IDP

Is the following allocation optimal ? ⇒ Ask to IDP

$$\{dist \mapsto C, srv \mapsto C, EMP \mapsto C, EDP \mapsto D, eng1 \mapsto C, elec \mapsto D\}$$

# DAL Allocation : Ask to IDP

Is the following allocation optimal ? $\Rightarrow$ Ask to IDP $\Rightarrow$ No

$$\{dist \mapsto C, srv \mapsto C, EMP \mapsto C, EDP \mapsto D, eng1 \mapsto C, elec \mapsto D\}$$

| basic DAL | cuts | DAL | | | | | | Option |
|---|---|---|---|---|---|---|---|---|
| | | $dist$ | $rsv$ | $EMP$ | $EDP$ | $eng1$ | $elec$ | |
| C | $\{dist\}$ | ≥ C | - | - | - | - | - | - |
| | $\{rsv\}$ | - | ≥ C | - | - | - | - | - |
| | $\{f_{EMP,eng1}, EDP\}$ | - | - | ≥ C | ≥ E | - | - | 1 |
| | $\{f_{EMP,eng1}\}$ | - | - | ≥ C | - | ≥ C | - | - |
| | $\{elec, EDP\}$ | - | - | - | ≥ C | - | ≥ E | 1 |
| | $\{elec, f_{EMP,eng1}\}$ | - | - | - | - | ≥ C | ≥ E | 1 |
| Result | | ≥ C | ≥ C | ≥ C | ≥ C | ≥ C | ≥ E | |
| Cost | | 30 | | | | | | |

# Tolerance to latent failures

⚠ We assume that all components where initially working properly

1. Determine the failure conditions and their severity (from FHA)
2. Build the fault trees for each failure condition
3. Compute the minimal cutsets
4. Qualitative verification : Compute the order and compare it to the required bound
5. Quantitative verification : Compute the probability and compare it to the required bound

What if some components are already failed ?

# Why considering latent failures ?

Why a component would be initially failed ?

- A safety analysis is performed on a given time interval (*e.g.*, the whole lifetime of the aircraft, for a flight) . . .
- When considering a flight, all components may not be available. . .

What are the consequences ?

- Possible violation of safety objective
  $\Rightarrow$ Identify the minimal list of equipments that must be available.

# Example of latent failure

### Example (Impact of latent failures)

Minimal cut $C = \{a, b\}$ for a catastrophic FC, if $b$ is already failed :

⇒ $C \rightarrow \{a\}$

⇒ Order goes from 2 to 1

⚠ System does not fulfil requirements, so $b$ must be available.

# In the list or not ?

An equipment is in the minimal equipment list if the safety objectives are not met when considering it as failed.

$$\Downarrow$$

Study the contribution of the equipment's failures to the MCS

How to "study the contribution of the equipment's failures to the MCS"?

# Building the minimal equipment list

**Context**
- Let $fc_1$ (resp $fc_2$) be a failure condition of severity CAT (resp. HAZ)
- Let $MCS_1 = \{\{e_1, e_2, e_4\}, \{e_2, e_3\}\}$ and $MCS_2 = \{\{e_1, e_3\}\}$
- Let $T = 10^3$ and $p(t_e \leq T) = 10^{-4}$.

**Question** Is $e_1$ in the minimal equipment list ?

# Building the minimal equipment list

**Context**

- Let $fc_1$ (resp $fc_2$) be a failure condition of severity CAT (resp. HAZ)
- Let $MCS_1 = \{\{e_1, e_2, e_4\}, \{e_2, e_3\}\}$ and $MCS_2 = \{\{e_1, e_3\}\}$
- Let $T = 10^3$ and $p(t_e \leq T) = 10^{-4}$.

**Question** Is $e_1$ in the minimal equipment list ?

**Answer** No because,

No cutset of order 2 (resp. 1) for $fc_1$ (resp. $fc_2$) containing $e_1$ $\Rightarrow$ order requirement still met for $fc_1$ (resp. $fc_2$).

If $p(t_e \leq T) = 1$ then $\overline{\Lambda}_{fc_1} \approx 2.10^{-11}$ (resp. $\overline{\Lambda}_{fc_2} \approx 10^{-7}$) $\Rightarrow$ quantitative requirement still met for $fc_1$ (resp. $fc_2$).

# Building the minimal equipment list

**Context**
- Let $fc_1$ (resp $fc_2$) be a failure condition of severity CAT (resp. HAZ)
- Let $MCS_1 = \{\{e_1, e_2, e_4\}, \{e_2, e_3\}\}$ and $MCS_2 = \{\{e_1, e_3\}\}$
- Let $T = 10^3$ and $p(t_e \leq T) = 10^{-4}$.

**Question** Is $e_1$ in the minimal equipment list ?

**Answer** No because,

No cutset of order 2 (resp. 1) for $fc_1$ (resp. $fc_2$) containing $e_1$ $\Rightarrow$ order requirement still met for $fc_1$ (resp. $fc_2$).

If $p(t_e \leq T) = 1$ then $\overline{\Lambda}_{fc_1} \approx 2.10^{-11}$ (resp. $\overline{\Lambda}_{fc_2} \approx 10^{-7}$) $\Rightarrow$ quantitative requirement still met for $fc_1$ (resp. $fc_2$).

**Question** Is $e_2$ in the minimal equipment list ?

# Building the minimal equipment list

**Context**

- Let $fc_1$ (resp $fc_2$) be a failure condition of severity CAT (resp. HAZ)
- Let $MCS_1 = \{\{e_1, e_2, e_4\}, \{e_2, e_3\}\}$ and $MCS_2 = \{\{e_1, e_3\}\}$
- Let $T = 10^3$ and $p(t_e \leq T) = 10^{-4}$.

**Question** Is $e_1$ in the minimal equipment list ?

**Answer** No because,

No cutset of order 2 (resp. 1) for $fc_1$ (resp. $fc_2$) containing $e_1 \Rightarrow$ order requirement still met for $fc_1$ (resp. $fc_2$).

If $p(t_e \leq T) = 1$ then $\overline{\Lambda}_{fc_1} \approx 2.10^{-11}$ (resp. $\overline{\Lambda}_{fc_2} \approx 10^{-7}$) $\Rightarrow$ quantitative requirement still met for $fc_1$ (resp. $fc_2$).

**Question** Is $e_2$ in the minimal equipment list ?

**Answer** Yes because, $e_2$ is implied in $\{e_2, e_3\} \Rightarrow$ order requirement not met for $fc_1$.

Now a Recap

# Today's lesson in 30"

Deal with dependencies

During design
: Trace independence assumptions during assessment $\Rightarrow$ became requirements during implementation

During verification
: Identify the potential sources of dependencies & integrate them in safety assessment

# Today's lesson in 30"

Emphasis on systematic errors :

- Currently, avoid systematic faults with design assurance level (DAL)
- DAL allocation depends on :
  - severity of functions/items failures,
  - independence between them,
  - cost of DAL related activities.

You understand highlighted terms
$\Rightarrow$ congratulations you've got the idea
Otherwise check out the slides !

Let’s check if you master the basic concepts of safety assessment for simple static systems !

How to select the relevant safety framework ?

# Dysfunctional model

Safety engineer creates **models** of the **failure propagation**

Formalises **contributions** of elementary failures to **feared events**

Derives **scenarios** leading to feared events thanks to a model based on a **formalism**

What a formalism can (or can't) **capture** ?

# Static system

## Definition (Static system)

The order of occurrence of the primary failures **does not** impact the occurrence of the studied feared event(s)

The scenarios leading to the feared event can modelled as **sets** :

- For instance by cutsets or prime implicants
- Can use many methods like Fault trees, Reliability block diagrams, HipHOPS, ...
- Underlying formalism : propositional logic

# Dynamic system

## Definition (Dynamic system)

The order of occurrence of the primary failures impacts the occurrence of the studied feared event(s)

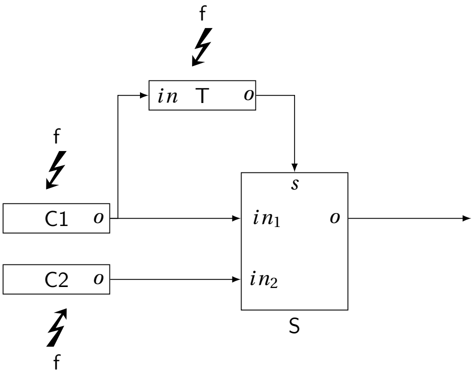
The scenarios leading to the feared event can modelled as **sequences** :

- For instance by minimal sequences or execution traces
- Can use many methods like Bayesian networks, Markov Chains, Petri Nets, ...
- Underlying formalism : State/Transition models

# An example : the auto-check system

Assumptions :

- Data are correct or erroneous
- C1 (resp. C2) can produce erroneous outputs C1.o (resp. C2.o) if occurrence of C1.f (resp. C2.f)
- Test component sends true iff C1 output is correct
- Test can be permanently stuck on the last decision if T.f occurs
- Selector sends in1 if s is true, in2 otherwise
- Feared event is *Erroneous output on S.o*




Is the system dynamic or static ?

# Deal with dynamism

Dynamic system models Either use a formalism dedicated to dynamic systems

- ⊕ Enable fine grain modelling of the failure propagation
- ⊕ Provide more meaningful analysis results
- ⊖ More complex to model and to analyse

Pessimistic model Build a pessimistic static model of your system

- ⊕ Easier to model and to analyse
- ⊖ Ensure that the model is pessimistic not always feasible

# Build a dynamic model of the system : Markov chain

## Definition (Markov chain)

Markov chain is a probabilistic state machine where :

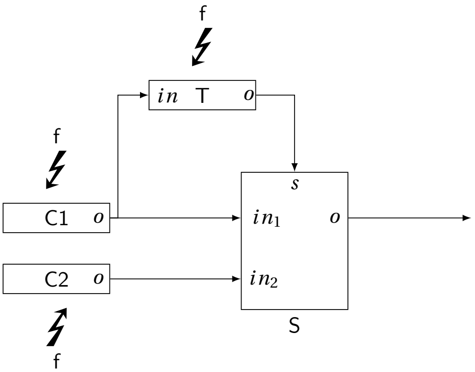
- States models the norminal or error system's states
- Transitions models the evolution of the system's state due to failures or nominal reconfigurations.
- A transition is labelled by a probability (for discrete MC, rate for continuous MC) of firing the transition from the current state.

Warning Applicable only if the system ensure the Markov assumption, i.e. the probability (or rate) of a transition depends only on the current state

# An example : Markov chain for the auto-test system

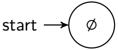
Instructions :

- A node of the chain encode the sequence (or set if the order does not matter) of component failed
- Transition are labeled by the failure rate of the event
- Initially none of the components are failed

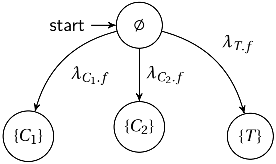



What is the Markov chain of this system ?

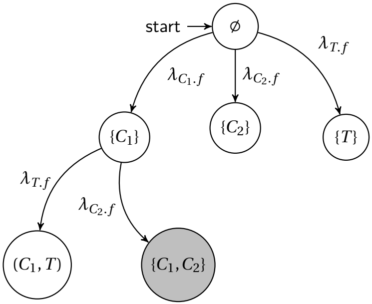
# An example : Markov chain for the auto-test system



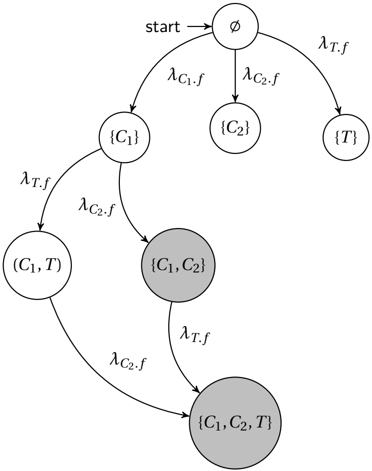
# An example : Markov chain for the auto-test system



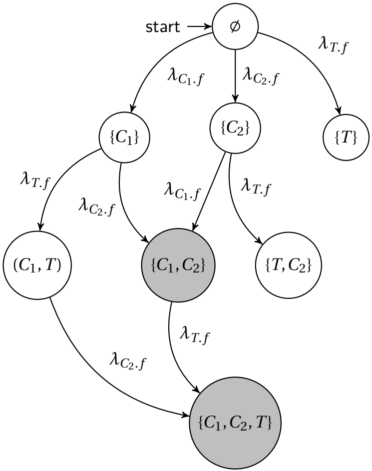
# An example : Markov chain for the auto-test system



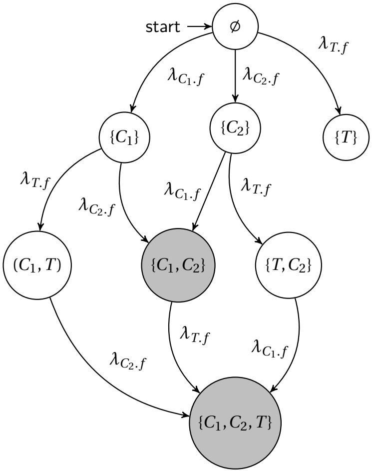
# An example : Markov chain for the auto-test system



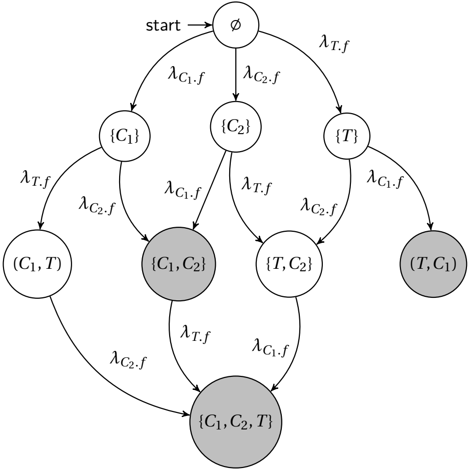
# An example : Markov chain for the auto-test system



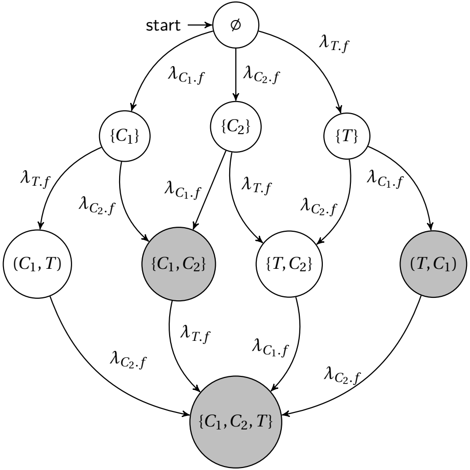
# An example : Markov chain for the auto-test system



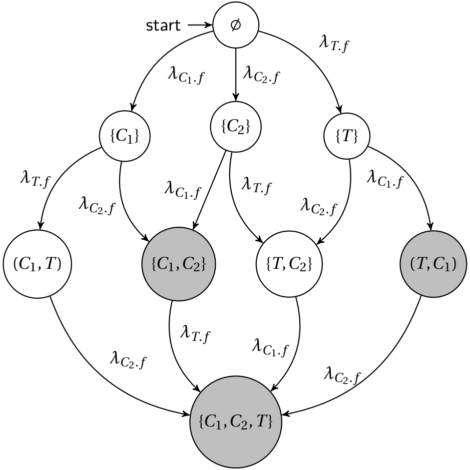
# An example : Markov chain for the auto-test system



# An example : Markov chain for the auto-test system



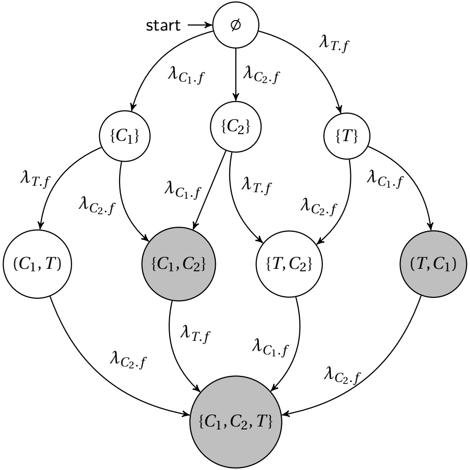
# An example : Markov chain for the auto-test system




Possible analyses :

- Find sequences of events leading to a feared state
- Estimate the probability of a feared event with Monte Carlo method
- Ensure formal properties (with temporal logic)
- Ensure probabilistic properties (with probabilistic model checking)

# An example : Markov chain for the auto-test system




*Minimal Sequences*

(C1.f,C2.f) ; (C2.f,C1.f) ;
(T.f,C1.f)

# Build a pessimistic model of the system

If one want to use a static model then it must ensure that the analysis is conservative

## Definition (Conservative analysis)

If a sequence $(e_1, \dots e_n)$ leads to the failure, in the pessimistic model the set $\{e_1, \dots e_n\}$ leads to the failure.

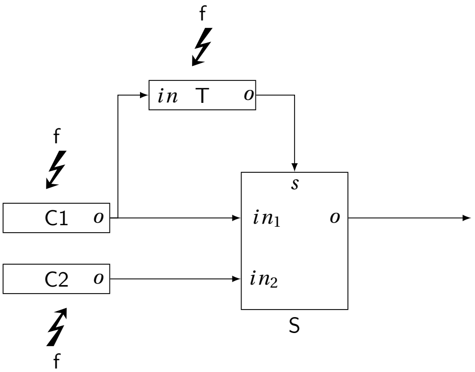
## Example (Test component behavior)

In the auto-test system, assume that if the Test is failed then the selector will send an erroneous value if one of the element is failed.

# An example : Fault tree for the auto-test component

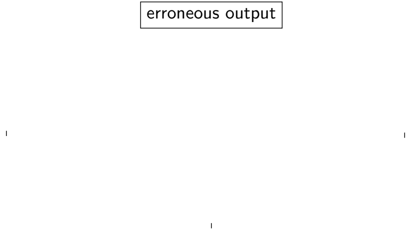
Instructions :

- If the Test is failed then the selector will send an erroneous value if one of the element is failed.

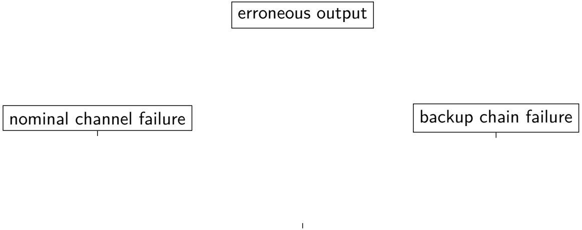



What is the fault tree of this system ?

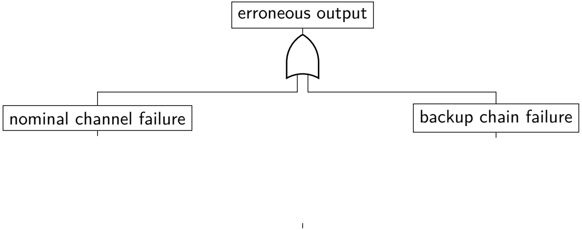
# An example : Fault tree for the auto-test component



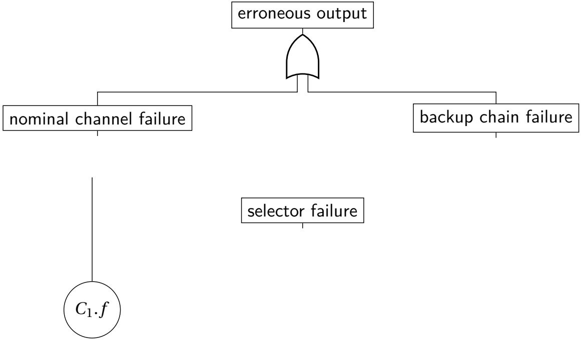
# An example : Fault tree for the auto-test component



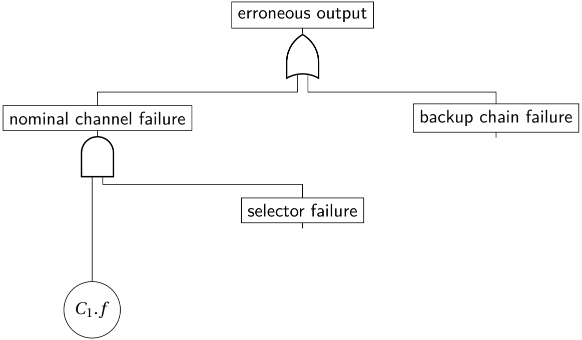
# An example : Fault tree for the auto-test component



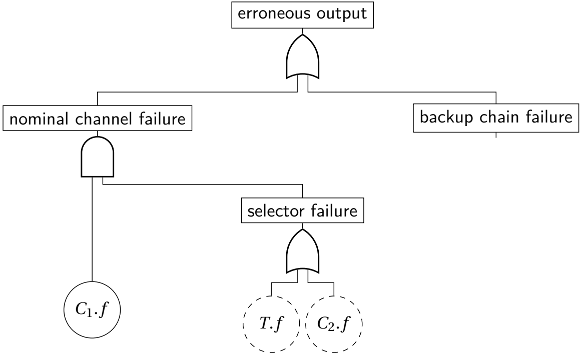
# An example : Fault tree for the auto-test component



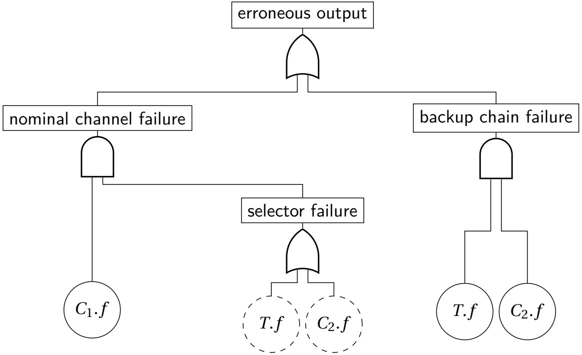
# An example : Fault tree for the auto-test component



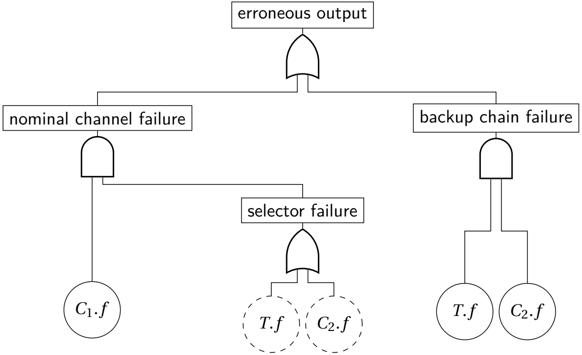
# An example : Fault tree for the auto-test component



# An example : Fault tree for the auto-test component



# An example : Fault tree for the auto-test component




Minimal cutsets

$$\{\{C1.f, C2.f\}; \{C1.f, T.f\}; \{C2.f, T.f\}\}$$

Minimal sequences

$$(C1.f, C2.f); (C2.f, C1.f); (T.f, C1.f)$$

# Limitations of classical formalism

Classic formalism shall highlight some failure propagation paths

- No explicit reference to the global system architecture / nominal behavior
- Potential misunderstanding or inconsistency between safety and design teams

Classical formalism totally relies on expert's analysis

- More and more difficult to be exhaustive for complex systems which integrate of various functions in a same hardware component
- Have reconfigurations of function modes and hardware architectures
- Are strongly interconnected with other systems

# Perspectives

Goals provide

- Formal failure propagation models closer to design models
- Tools to assist construction and automated analysis of complex models

More details in the next lessons

Let's talk about the (your) future!

## What are the new safety challenges ?


What are the new safety challenges ?


Let's have a quick (and non-exhaustive) overview !

# From I to AI

Trend: Huge trend to automate complex tasks preformed by operators (professional or not)

Breakdown: New technologies involving complex sensor fusion or image processing

# From I to AI

Trend Huge trend to automate complex tasks preformed by operators (professional or not)

Breakdown New technologies involving complex sensor fusion or image processing

What are the risks related to the massive adoption of such systems ?

**An Example** Automotive anti-collision system https://youtu.be/ZMFbMV5QNzk?t=81

# Challenge 1 : Trust Me I Am Autonomous

- Classical software correctness demonstrated by :
  1. validation : the specification breakdown is sound, complete and testable (ABS example)
  2. verification : the implementation is compliant to the specification (Offshore example)
- V&V achieved thanks to testing, traceability and formal verification

# Challenge 1 : Trust Me I Am Autonomous

- Classical software correctness demonstrated by :
  1. validation : the specification breakdown is sound, complete and testable (ABS example)
  2. verification : the implementation is compliant to the specification (Offshore example)
- V&V achieved thanks to testing, traceability and formal verification

What is the specification breakdown of an AI-based pedestrian detection system ?
How to provide confidence on safety integrity for critical function based on AI ?

# Challenge 2 : Taking into account new failures

- Safety impact of hardware failure addressed in safety critical systems (redundancy, mutual checks, lock-step)

# Challenge 2 : Taking into account new failures

- Safety impact of hardware failure addressed in safety critical systems (redundancy, mutual checks, lock-step)

What is the safety impact of an hardware failure executing AI-based software ?
Can we detect & manage this failure ?

# Challenge 3 : Safe integration of tomorrow aircrafts

- Various applicative domains can benefit from new aircraft concepts (VTOL, UAV, . . .)
  - Infrastructure inspection (SCNF, ERDF, . . .)
  - Package delivery (Amazon, CDiscount, La Poste, . . .)
  - Flying taxi (Airbus' Vahana project, Boeing, Uber, . . .)

# Challenge 3 : Safe integration of tomorrow aircrafts

- Various applicative domains can benefit from new aircraft concepts (VTOL, UAV, ...)
  - Infrastructure inspection (SCNF, ERDF, ...)
  - Package delivery (Amazon, CDiscount, La Poste, ...)
  - Flying taxi (Airbus' Vahana project, Boeing, Uber, ...)

What are the new risks related to the integration of such aircraft in the flight traffic ?
How to adapt safety analyses to take into account distributed procedures, autonomous avoidance systems ?

# Thank you

ONERA

THE FRENCH AEROSPACE LAB

www.onera.fr