Probabilistic Model Checking with PRISM

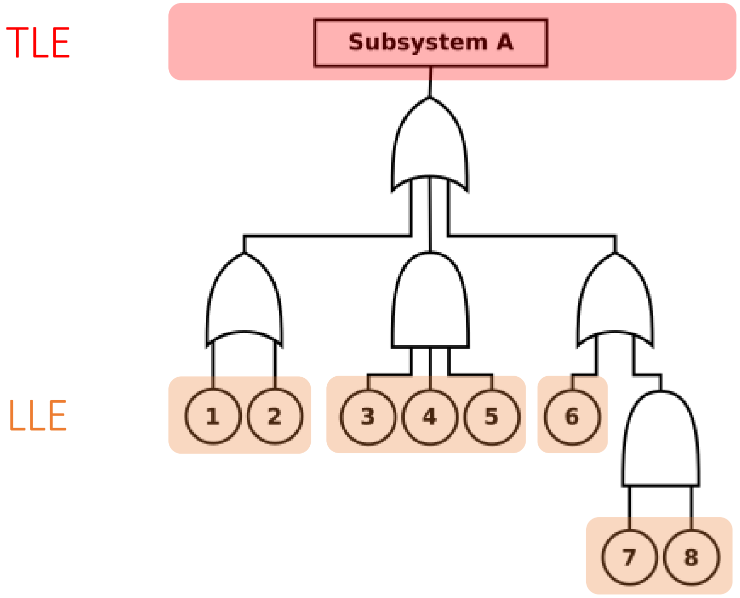
Production Line

Patrick SARDINHA

Fault Trees

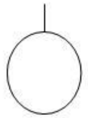
Fault Trees (FT) graphically represent possible combinations of events (Low Levels Events) leading to a predefined undesirable event (Top Level Event)

Representation:

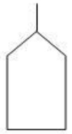


Graphic symbols

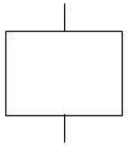
Events:



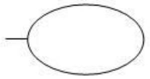
Basic event: failure in a system component



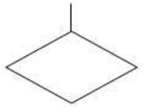
External event: expected to occur



Intermediate event: events occurring at the exit of a door



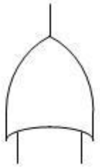
Conditioning event: an event with conditions



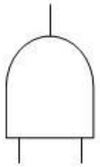
Undeveloped event: an event with insufficient information

Gates:

Describe the relationship between input and output events.



OR gate: the output occurs if any input occurs



AND gate: the output occurs only if all inputs occur

Fault Tree Analysis (FTA)

Method used to evaluate the risks of a system and allows to:

Understand how a system can fail

Know how to reduce the risks

Visualize the event rates of an accident

FTA is often performed by transforming the FT into a Boolean function which is used for simulation …

… but this methodology has a lot of constraints (time/resources)

A new formal Probabilistic FTA methodology

Efficient Probabilistic Fault Tree Analysis of Safety Critical Systems via Probabilistic Model Checking

Marwan Ammar, Ghaith Bany Hamad, Otmane Ait Mohamed, Yvon Savaria

A new formal Probabilistic FTA methodology

The idea is as follows:

1. Model the system (composed of components) and specify event parameters
2. Synthesize the system fault tree
3. Model the behavior of each FT gate as a probabilistic automaton (PA)
4. Generate a formal model of the fault tree with the parallel composition of the PA (PRISM)
5. Analyze the model to evaluate the maximum probability of Top Level Event (TLE)

System description

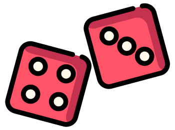
We have a **production chain** made up of:

Machines that extract resources

Machines that transform resources

We have different types of **disruptive primary events**:

Technical failures on machines with a certain probability



Non-deterministic quantities of extracted resources

…

And others

Resource extraction

We have different kinds of machine:

	**Burner mining drill**
Inputs	Raw minerals (from a source)
Outputs	Minerals
Basic event	- Can break down - May be affected by an external event - The input quantity may vary

	**Offshore pump**
Inputs	Water (from a source)
Outputs	Water
Basic event	- Can break down - May be affected by an external event - The input quantity may vary

Resource transformation

We have different kinds of machine:

	Boiler
Inputs	Fuel & water
Outputs	Steam
Basic event	- Can break down - May be affected by an external event - Fuel not supplied - Water not supplied

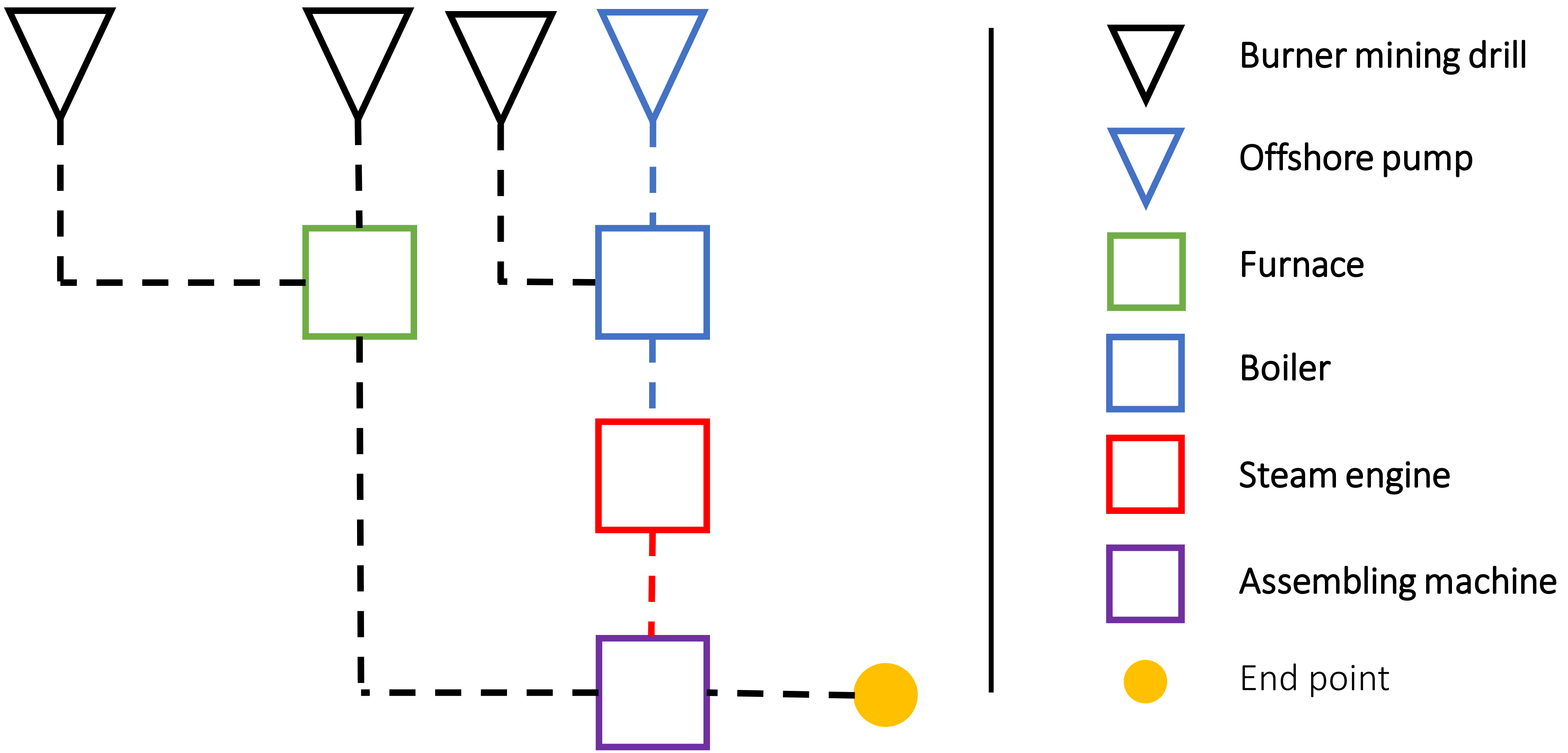
	Steam engine
Inputs	Steam
Outputs	Electricity
Basic event	- Can break down - May be affected by an external event - Steam not supplied

Resource transformation

	**Furnace**
Inputs	Fuel & minerals
Outputs	Processed minerals
Basic event	- Can break down - May be affected by an external event - Fuel not supplied - Minerals not supplied

	**Assembling machine**
Inputs	Electricity & Processed minerals
Outputs	Final product
Basic event	- Can break down - May be affected by an external event - Electricity not supplied - Processed minerals not supplied

Production line: Example



Principal idea

Estimate and compare the probability that faults from different low-level events cause a system failure

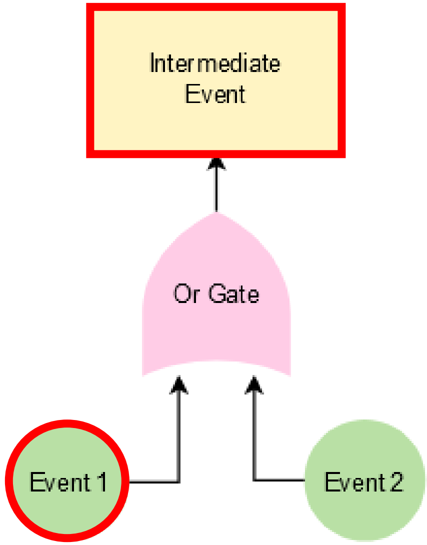
System Failure
Undesirable event

Top Level Event (TLE)

Some mechanisms

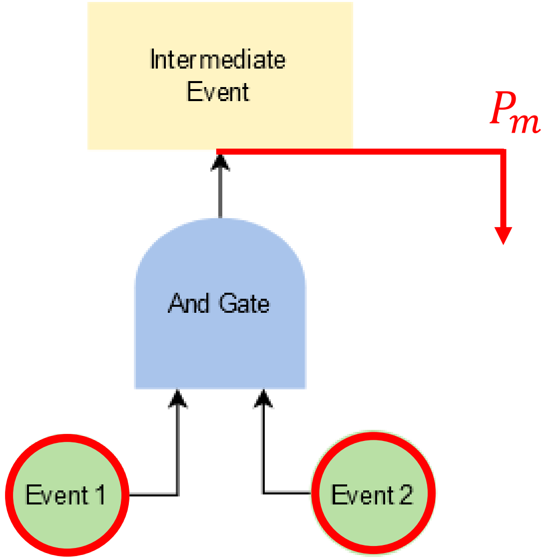
<u>Fault propagation</u>:

An error in a node at a lower level of the tree can propagate to a higher level

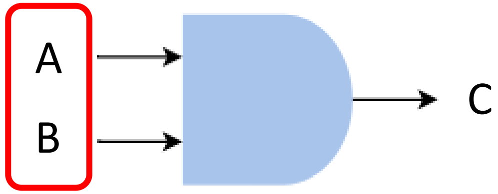


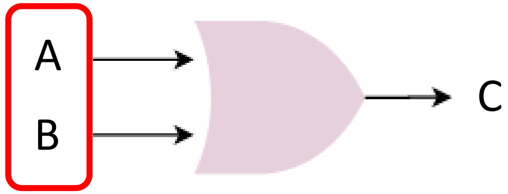
<u>Fault masking</u>:

Adds a probability of fault mitigation inside the gates

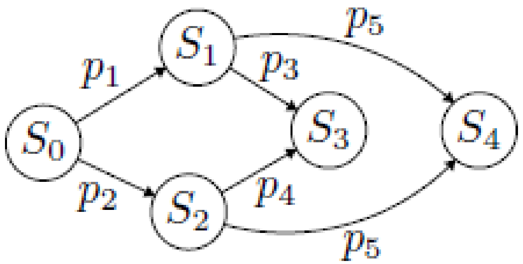


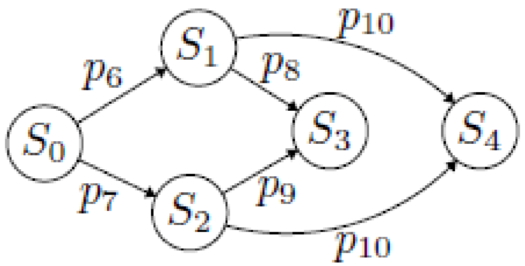
FT gates



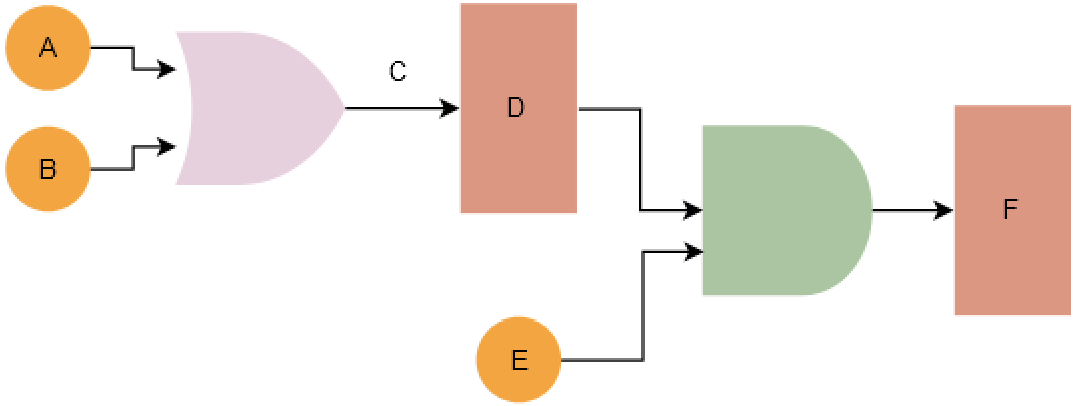


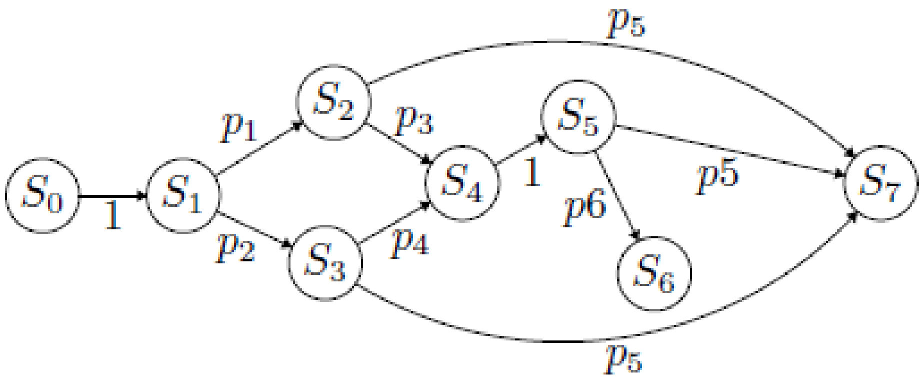
All door entrances represent events with a probability of being triggered





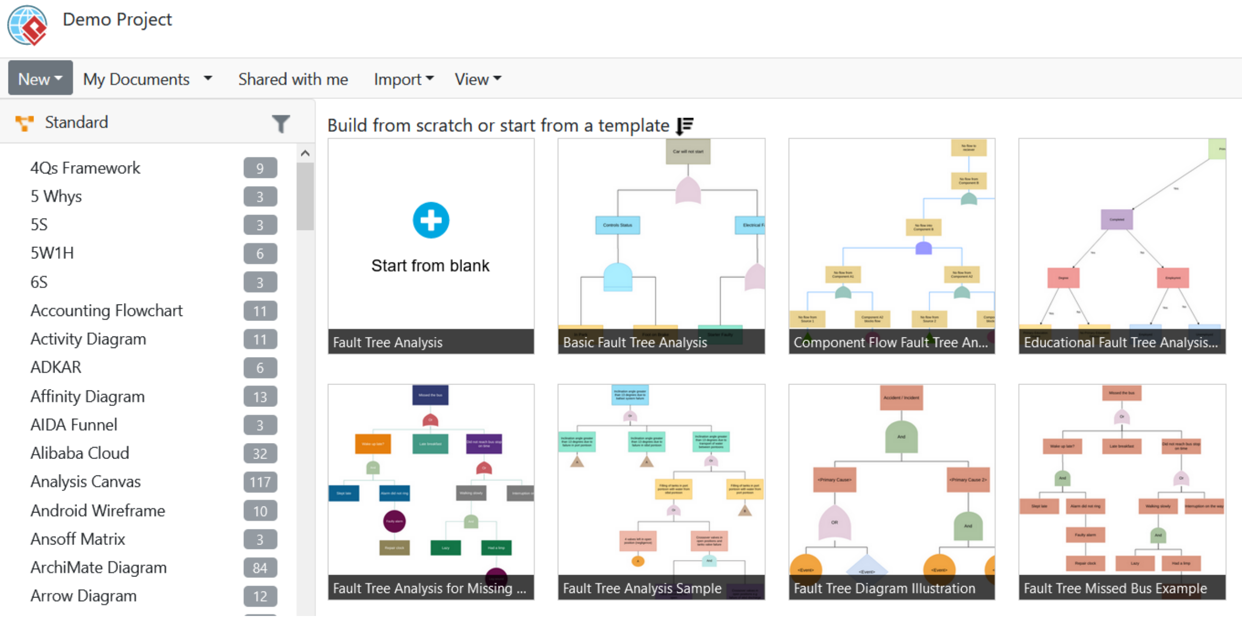
Example: Door combination



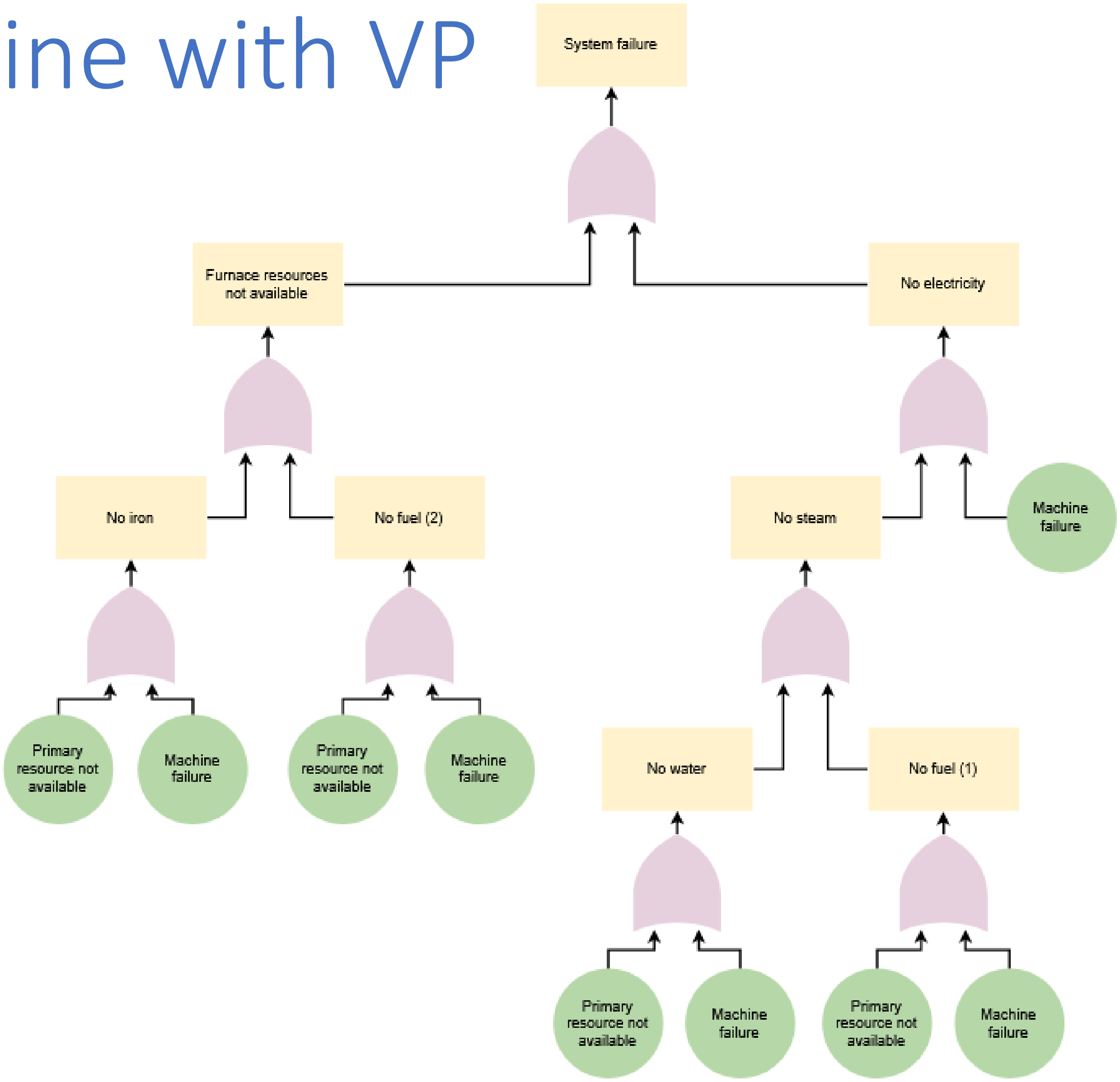


Tool

VisualParadigm Online



Prod. line with VP



Production line fault tree

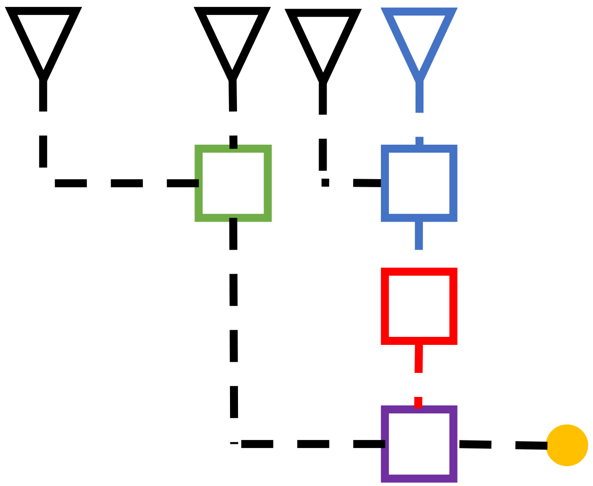
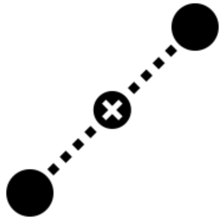


Diagram of the production line

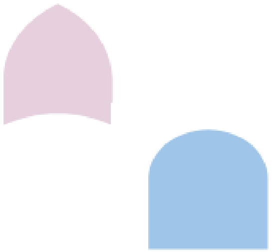
Order on evaluation of gates

Depending on the order in which the gates are evaluated, the PA of the TF will be different

We assume that:



Events on different gates have no connection

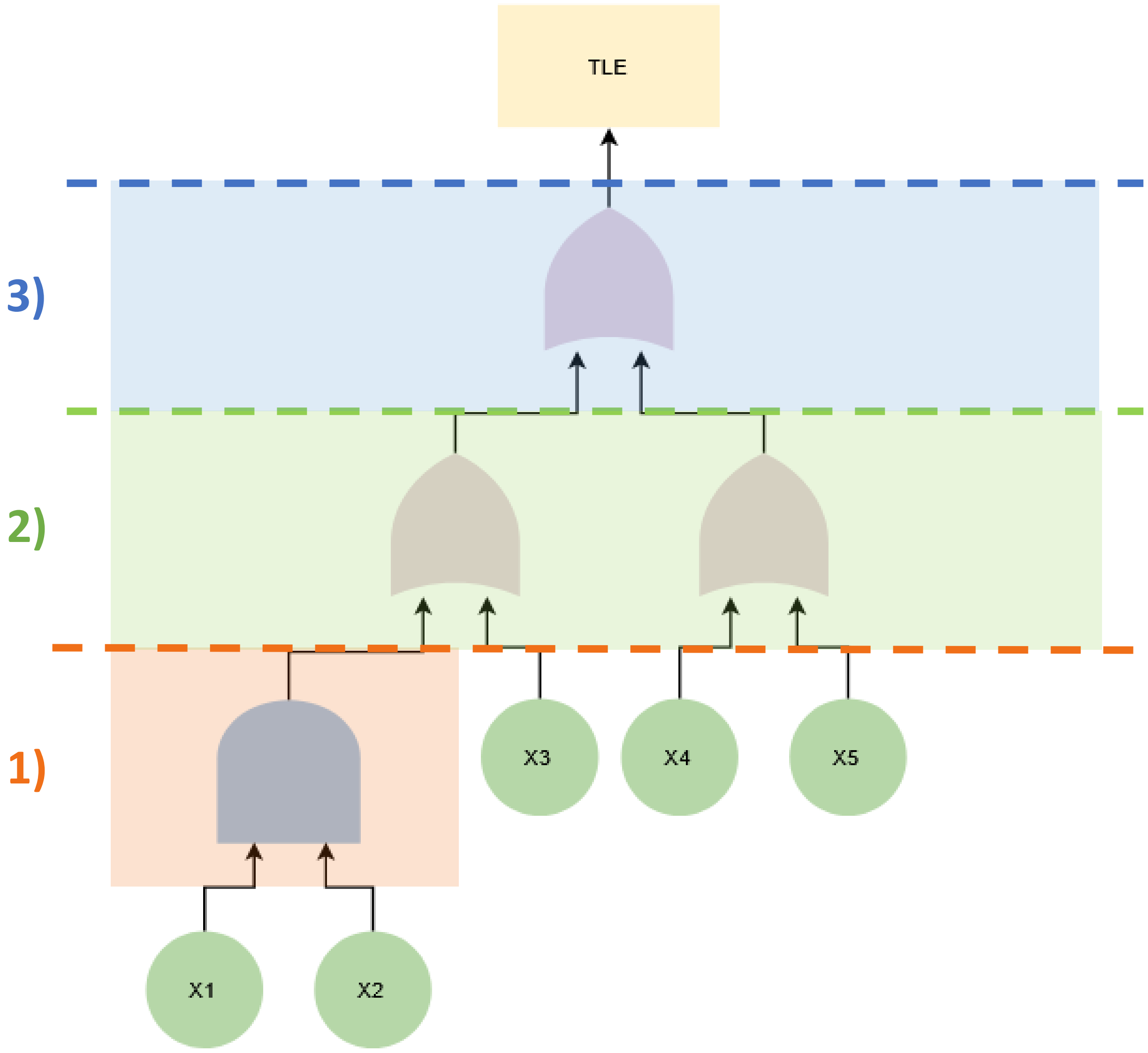


When evaluating a door, we consider all the inputs

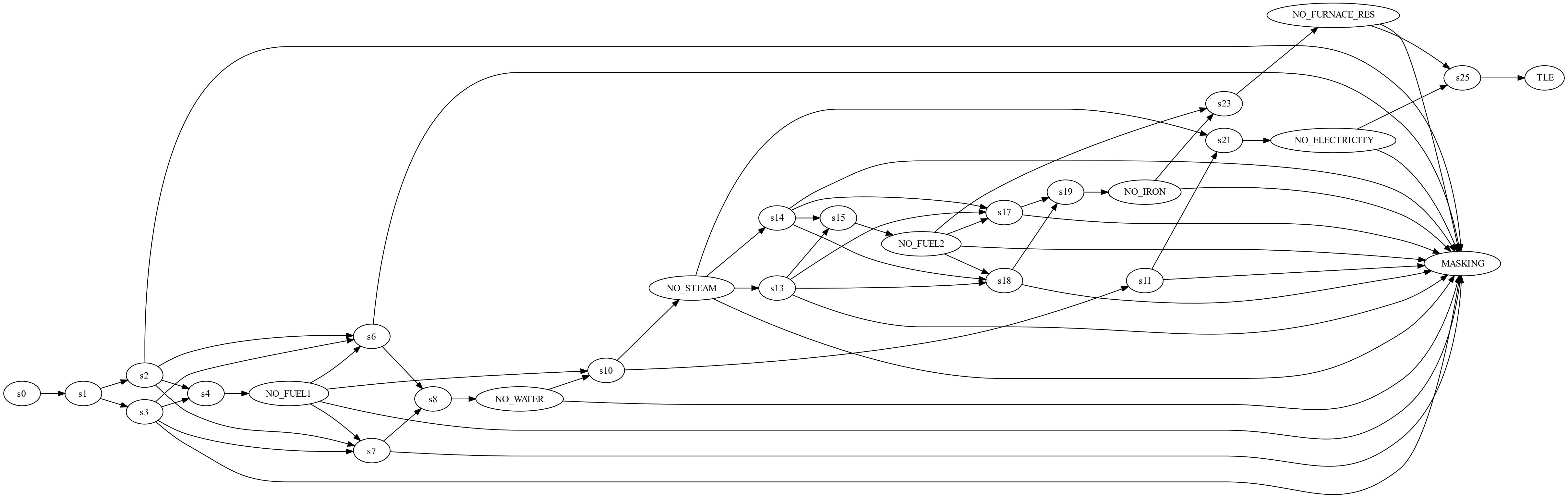
Layer by layer

The idea is therefore to first evaluate all the doors of a layer …

.. and then move to a higher layer



Corresponding PA



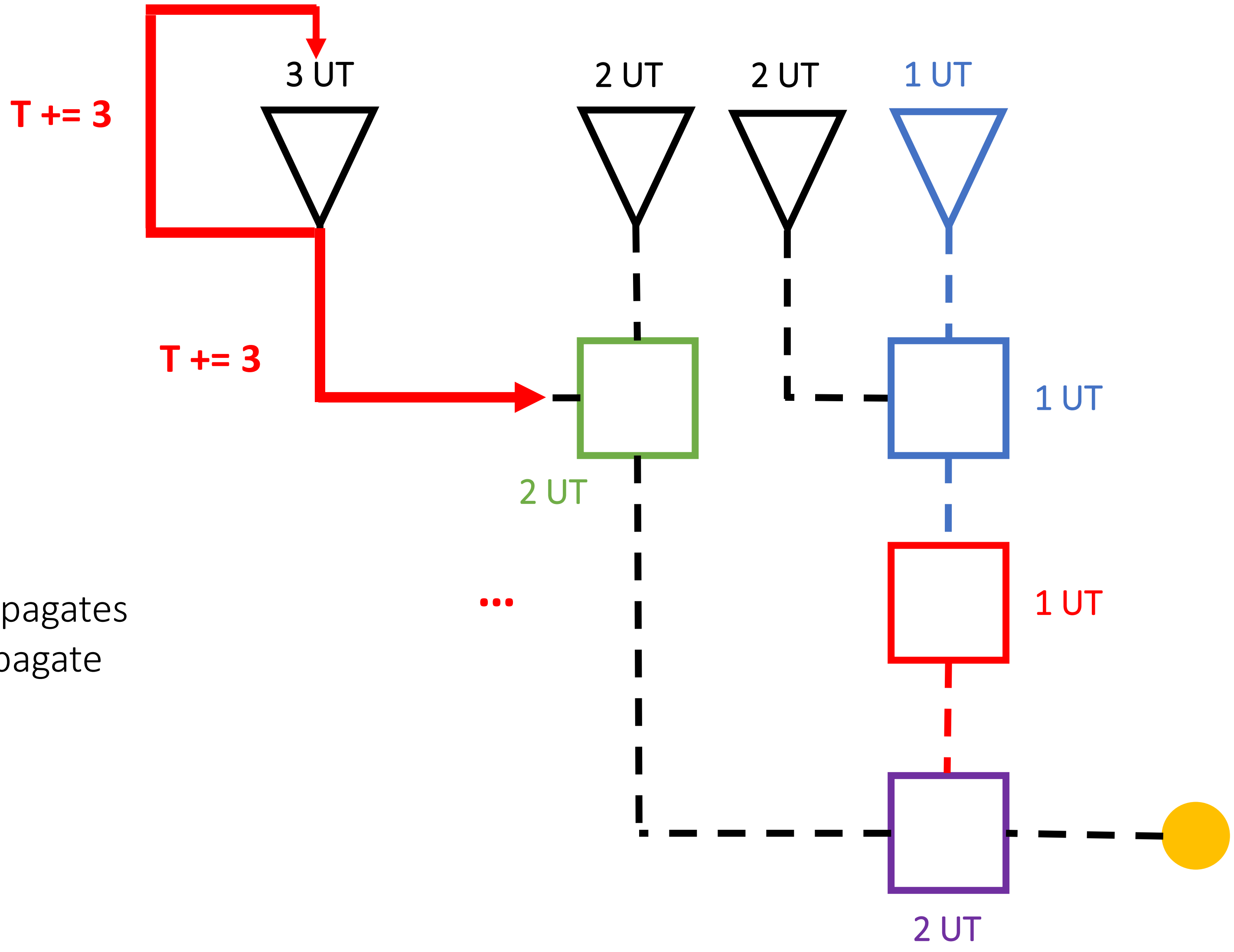
Properties to check with our propagation model

Estimate and compare the probability that a fault from a low-level events cause a system failure **before $\boldsymbol{X_t}$ time units**

$$P_{max} = ?\ [\ (F\ X_i = 1)\ \&\ (F\ TLE = 1)\ \&\ (F\ T < X_t)]$$

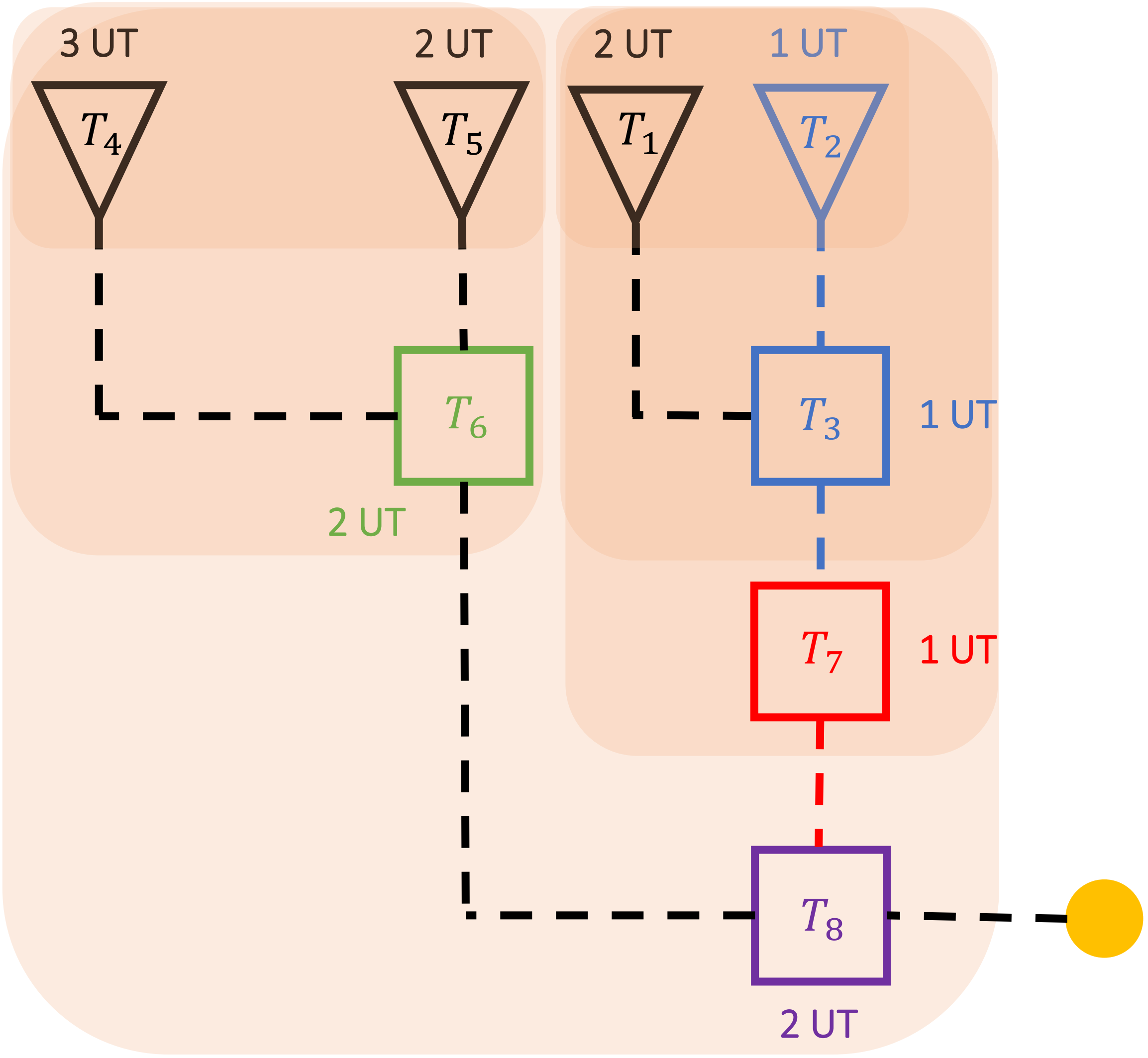
The maximum probability that event X_i will trigger, the fault is propagated to the TLE and T is $< X_t$?

Add time value to each machine



Time increment when a fault propagates through a gate or does not propagate

Parallelism between machines



number of loop

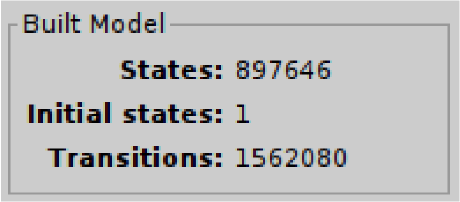
$$T_1 = \left\{ T_1 + 2 * v_1 \right.$$

$$T_2 = \left\{ T_2 + 1 * v_2 \right.$$

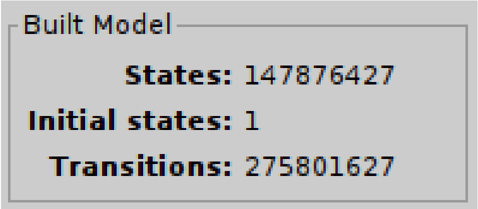
$$T_3 = \left\{ \begin{array}{l} \max(T_1, T_2) + 1 \\ T_1, T_2 = \max(T_1, T_2) + 1 \end{array} \right.$$

Limitation of implementation

Our model is a bit complex and is made up of many states …



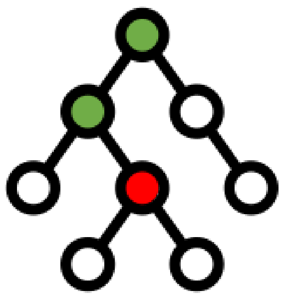
$T_f < 10$



$T_f < 20$

Reduce the complexity

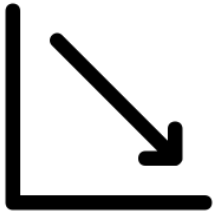
To reduce the complexity of the code, the idea is to recover the intermediate results of the subsystems



This prevents when a fault is masked in a machine that the whole branch of the tree is reassessed

Benefit

This modification leads to a saving of time and space concerning the calculations.



To calculate a single property, we go from **8 - 10 minutes** to **16 - 25 seconds**

Results: $T_{f1} < 10$

Bottom Event	Probability trigger
Primary res. (X_1)	0.32
Machine failure (X_2)	0.05
Primary res. (X_3)	0.15
Machine failure (X_4)	0.20
Primary res. (X_5)	0.32
Machine failure (X_6)	0.05
Primary res. (X_7)	0.10
Machine failure (X_8)	0.05
Machine failure (X_9)	0.25

Fault propagation prob: 90%

Fault masking prob: 10%

$$P = ?\left[(F\ X_1 = 1)\ \&\ (F\ TLE = 1)\ \&\ (T_{f1} < 10)\right] = 0.54066383$$

$$P = ?\left[(F\ X_2 = 1)\ \&\ (F\ TLE = 1)\ \&\ (T_{f1} < 10)\right] = 0.08447872$$

$$P = ?\left[(F\ X_3 = 1)\ \&\ (F\ TLE = 1)\ \&\ (T_{f1} < 10)\right] = 0.26791824$$

$$P = ?\left[(F\ X_4 = 1)\ \&\ (F\ TLE = 1)\ \&\ (T_{f1} < 10)\right] = 0.35722432$$

$$P = ?\left[(F\ X_5 = 1)\ \&\ (F\ TLE = 1)\ \&\ (T_{f1} < 10)\right] = 0.54066383$$

$$P = ?\left[(F\ X_6 = 1)\ \&\ (F\ TLE = 1)\ \&\ (T_{f1} < 10)\right] = 0.08447872$$

$$P = ?\left[(F\ X_7 = 1)\ \&\ (F\ TLE = 1)\ \&\ (T_{f1} < 10)\right] = 0.41676170$$

$$P = ?\left[(F\ X_8 = 1)\ \&\ (F\ TLE = 1)\ \&\ (T_{f1} < 10)\right] = 0.20838085$$

$$P = ?\left[(F\ X_9 = 1)\ \&\ (F\ TLE = 1)\ \&\ (T_{f1} < 10)\right] = 0.12502851$$

Results: $T_{f2} < 20$

Bottom Event	Probability trigger
Primary res. (X_1)	0.32
Machine failure (X_2)	0.05
Primary res. (X_3)	0.15
Machine failure (X_4)	0.20
Primary res. (X_5)	0.32
Machine failure (X_6)	0.05
Primary res. (X_7)	0.10
Machine failure (X_8)	0.05
Machine failure (X_9)	0.25

Fault propagation prob: 90%

Fault masking prob: 10%

$$P=?\left[(F\ X_1=1)\ \&\ (F\ TLE=1)\ \&\ \left(T_{f2}<20\right)\right]=0.83347989$$

$$P=?\left[(F\ X_2=1)\ \&\ (F\ TLE=1)\ \&\ \left(T_{f2}<20\right)\right]=0.13023123$$

$$P=?\left[(F\ X_3=1)\ \&\ (F\ TLE=1)\ \&\ \left(T_{f2}<20\right)\right]=0.41301905$$

$$P=?\left[(F\ X_4=1)\ \&\ (F\ TLE=1)\ \&\ \left(T_{f2}<20\right)\right]=0.55069207$$

$$P=?\left[(F\ X_5=1)\ \&\ (F\ TLE=1)\ \&\ \left(T_{f2}<20\right)\right]=0.83347989$$

$$P=?\left[(F\ X_6=1)\ \&\ (F\ TLE=1)\ \&\ \left(T_{f2}<20\right)\right]=0.13023123$$

$$P=?\left[(F\ X_7=1)\ \&\ (F\ TLE=1)\ \&\ \left(T_{f2}<20\right)\right]=0.64247408$$

$$P=?\left[(F\ X_8=1)\ \&\ (F\ TLE=1)\ \&\ \left(T_{f2}<20\right)\right]=0.32123704$$

$$P=?\left[(F\ X_9=1)\ \&\ (F\ TLE=1)\ \&\ \left(T_{f2}<20\right)\right]=0.19274222$$

Results analysis

We can observe that the probability that a fault propagates and reaches the top of the tree is much higher for $T_2 > T_1$
which implies that the longer the production time, the more likely a technical failure will occur

The same machines with the same properties at the same level (of the FT) **give similar results**

The **number of time units** associated with each machine will influence the results

A particular event will be **more impacted** by events that are **more or less directly linked to it** (same door, sub-part of the tree, etc.)

Conclusion

We were able to apply a new method for the analysis of fault trees and adapt it for our production chain

The property studied allows us to assess the chances of a production line breaking down before reaching a given execution time

Another property to study could be the availability of a machine in a day and know the time it spends downtime due to a failure

References - Github

https://github.com/sardinhapatrick/PMC_PRISM

References

PRISM: https://www.prismmodelchecker.org/

Factorio wiki: https://wiki.factorio.com/Main_Page

Modélisation et simulation de flux de production:

Franck Fontanili, *Intégration d'outils de simulation et d'optimisation pour le pilotage d'une ligne d'assemblage multiproduit à transfert asynchrone,* Partie IV, page 87-133

References

FTA: https://en.wikipedia.org/wiki/Fault_tree_analysis

FTA via PMC: M. Ammar, G. B. Hamad, O. A. Mohamed and Y. Savaria, "*Efficient probabilistic fault tree analysis of safety critical systems via probabilistic model checking* "

https://ieeexplore.ieee.org/abstract/document/7880373/metrics#metrics

SML: https://fr.wikipedia.org/wiki/Systems_Modeling_Language

VP: https://online.visual-paradigm.com/fr/diagrams/features/fault-tree-analysis-software/

MDP: https://en.wikipedia.org/wiki/Markov_decision_process

References

FMTs: M. Ammar, G. B. Hamad and O. Ait Mohamed, "Probabilistic High-Level Estimation of Vulnerability and Fault Mitigation of Critical Systems Using Fault-Mitigation Trees (FMTs)," *2019 IEEE Latin American Test Symposium (LATS)*, Santiago, Chile, 2019, pp. 1-6

https://ieeexplore.ieee.org/document/8704589

FTA via PMC (case study): M. Ammar, K. A. Hoque and O. A. Mohamed, "Formal analysis of fault tree using probabilistic model checking: A solar array case study," *2016 Annual IEEE Systems Conference (SysCon)*, Orlando, FL, USA, 2016, pp. 1-6

https://ieeexplore.ieee.org/abstract/document/7490556

Finite State Machine Designer: https://www.cs.unc.edu/~otternes/comp455/fsm_designer/

Probabilistic Model Checking with PRISM

Production Line

Patrick SARDINHA