FAULT TREE ANALYSIS THROUGH PARALLEL COMPUTING

Reliability and resilient infrastructure workshop

17-18 January 2014 - UNIST

Marco Torbol

(1) Introduction

"Origin of Fault tree analysis"

1962 H Watson and A. B. Mearns invented fault tree analysis (FTA) at the Bells Labs while working on the Minuteman Guidance System

Its use was extended to the entire project.

1970 FTA is used in the nuclear power plant industry

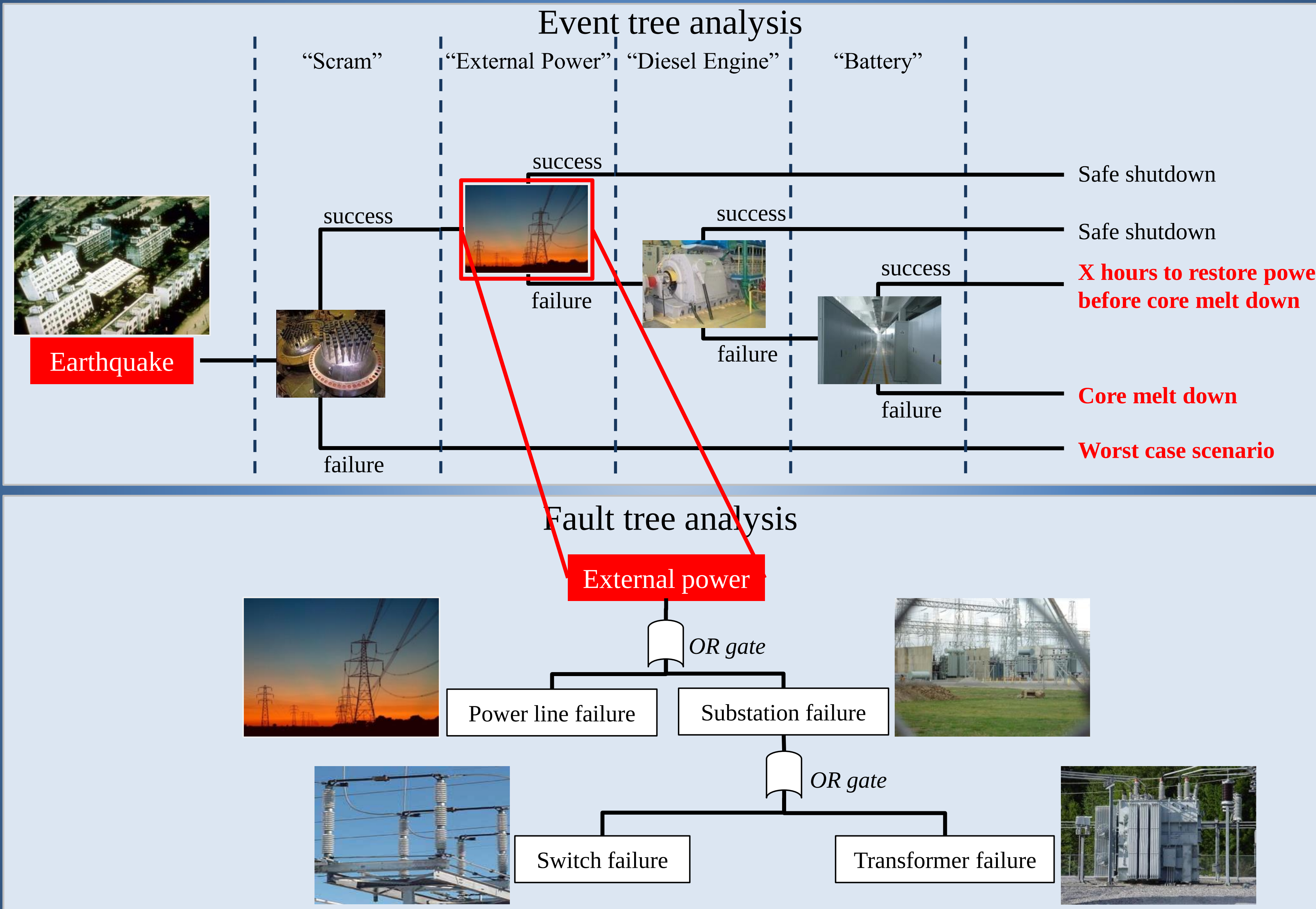
Nowadays FTA is used in a variety of field:
- the chemical industry
- the nuclear industry
- the aerospace industry

"Nuclear industry"

Probabilistic risk assessment (PRA) is used to study the safety, the reliability, the risk, and the possible consequences when operating a nuclear power plant.

Small event tree (ETA) / large fault tree analysis (FTA) is the technique used to assess the risk and consequences of rare events, such as: an earthquake, a tsunami, a loss of coolant accident (LOCA)

(2) Background



(2) Background

Normal Operation ECCS*

1st Backup Power from Utility

2nd Backup Diesel Engine

3rd Backup Battery (6 hr)

No cooling power

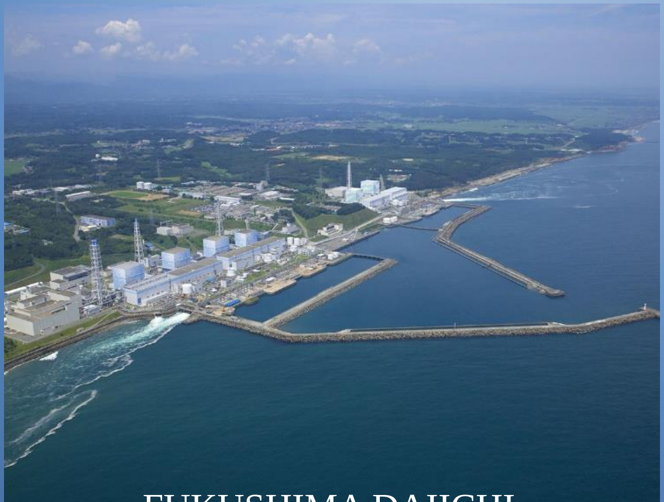
Automatic Shutdown of Reactor

Transmission System is damaged

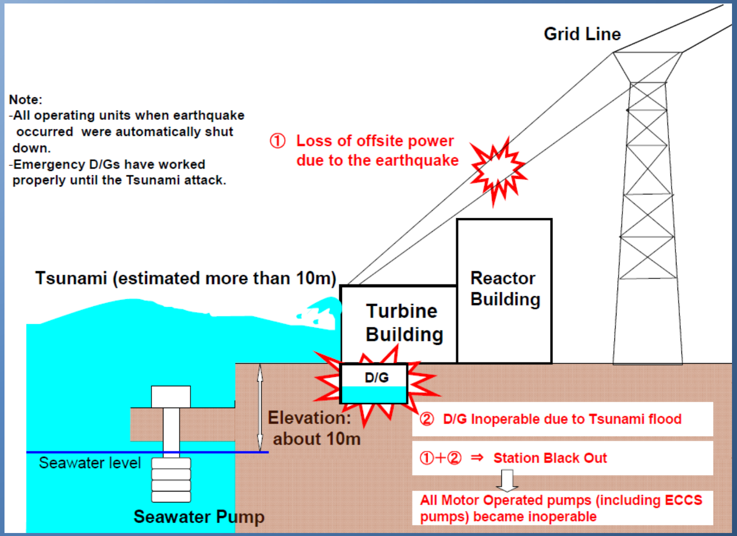
Complete Loss of diesel engine

EARTHQUAKE

TSUNAMI



FUKUSHIMA DAIICHI

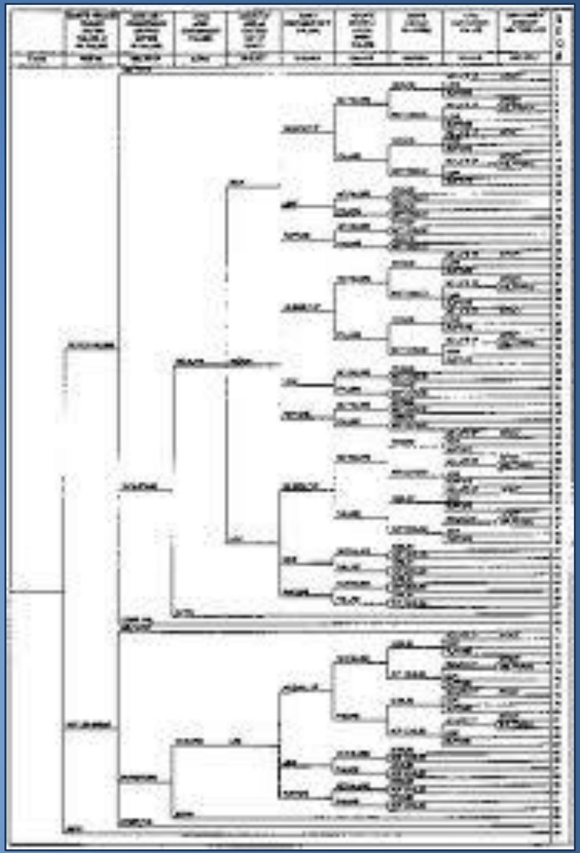


(3) Problem

Problem

- The solution of a large fault tree still takes intensive computational effort and time even with a state of the art CPU.

- The fault tree is composed by thousands of gates and basic events and includes multiple occurring events (MOE), multiple occurring branches (MOB), different gate types, and each gate can have more than two inputs.



Example

The fault tree of an emergency diesel engine is made of more than 5,000 gates

Solution methods

- direct analytical computation
- computation of the cut-sets
- binary decision diagram (BDD)
- Monte Carlo simulation.

(4) Fault tree analysis

Fault tree analysis through parallel computing

Parallel computing is efficient when all cores perform the same amount of operation (load balancing problem)

Example:

Probability of AND gate

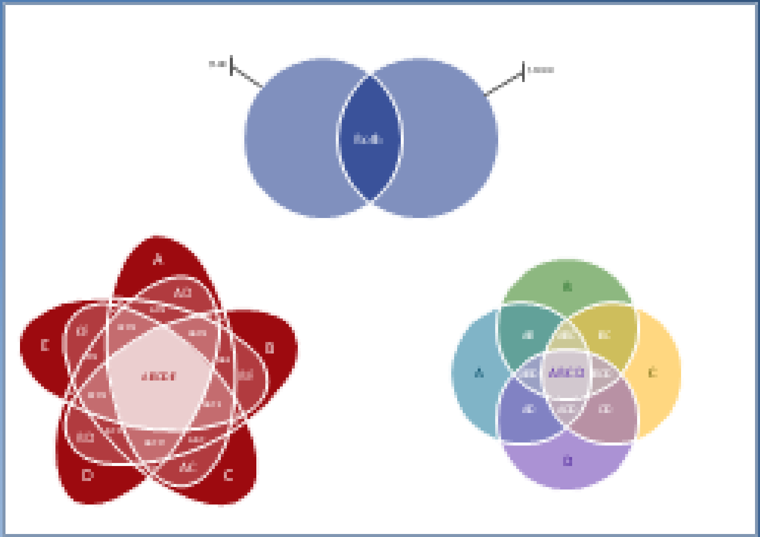
$$P_{g1} = P_{E1} \cdot P_{E2}$$

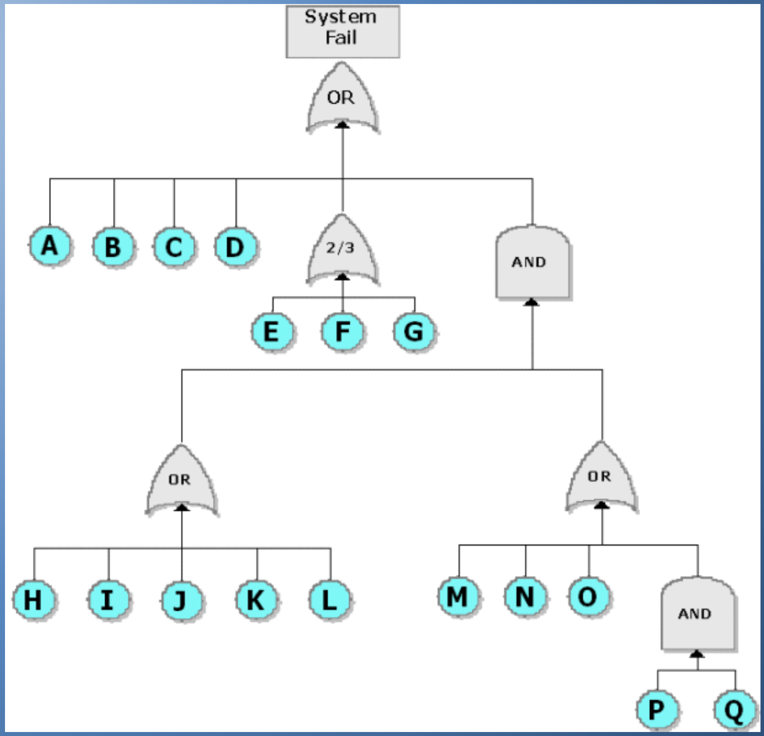
$$P_{g1} = P_{E1} \cdot P_{E2} \cdot P_{E3} \cdot P_{E4}$$

Probability of OR gate

$$P_{G1} = P_{E1} + P_{E2} - (P_{E1} * P_{E2})$$

$$P_{G1} = P_{E1} + P_{E2} + P_{E3} + P_{E4} - (P_{E1} * P_{E2}) - (P_{E2} * P_{E3}) - (P_{E3} * P_{E4}) + (P_{E1} * P_{E2} * P_{E3}) + (P_{E1} * P_{E2} * P_{E4}) + (P_{E1} * P_{E3} * P_{E4}) + (P_{E2} * P_{E3} * P_{E4}) - (P_{E1} * P_{E2} * P_{E3} * P_{E4})$$

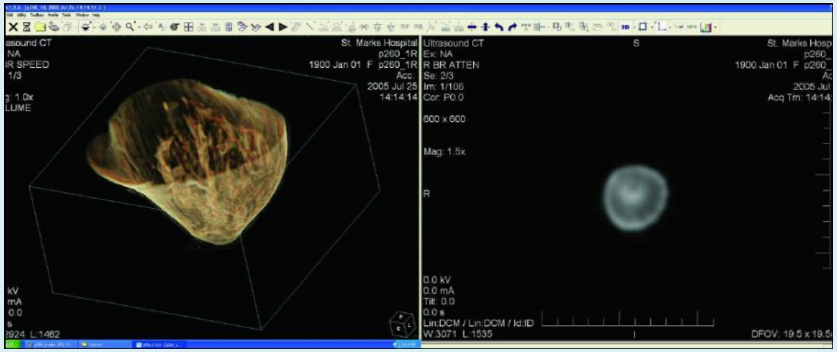




(4) GPGPU Parallel computing

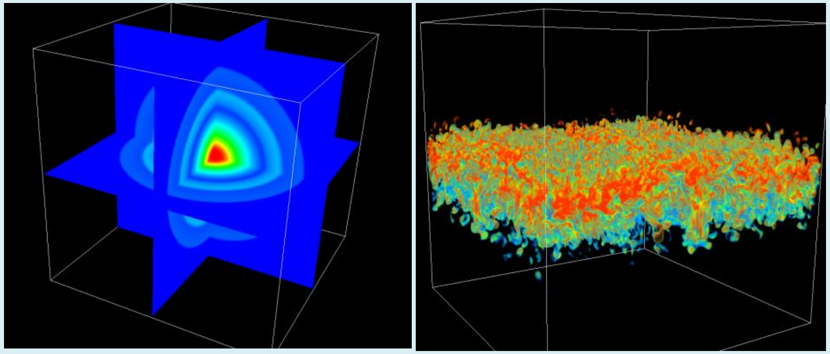
Medical field

Company: TechniScan Medical Systems
Idea: Breast cancer detection using 3D Ultrasound instead of X-ray
Issue: 3D Ultrasound digital image processing was taking days
GPU: 20 minutes

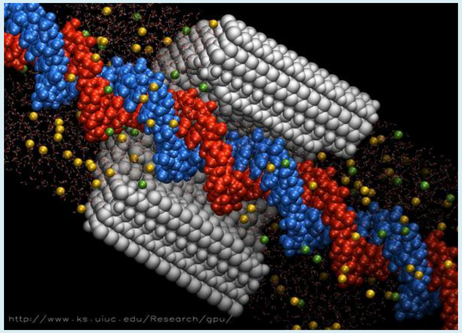


Computational fluid dynamics

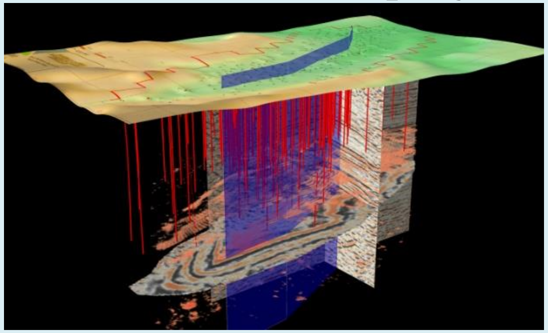
Company: University of Cambridge “many-core group”
Idea: Riemann-problem-based finite volume methods
Advantage: Solution moved from supercomputer cluster to desktop computer.
Even students can run their own simulation without High Performance Cluster



Environmental science



Oil & Gas Company

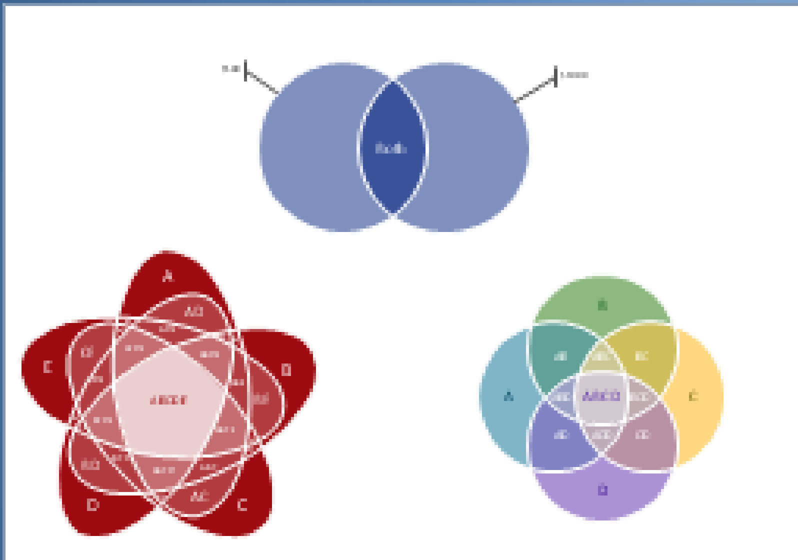


…

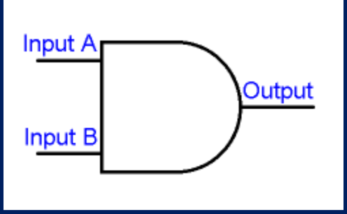
(5) Parallel approach

Gate expansion and shadow gates

Each gate with multiple inputs and that is not an AND gate or an OR gate is reduced to a series of multiple gates that are a combination of OR gates and AND gates



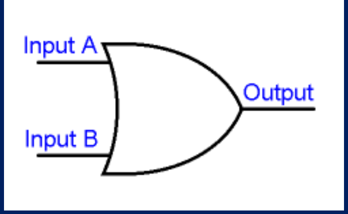
"AND gate"



$$P_{g1} = P_{E1} \cdot P_{E2} \cdot P_{E3} \cdot P_{E4}$$

$$P_{g1} = P_{E1} \cdot P_{g2}$$
$$P_{g2} = P_{E2} \cdot P_{g3}$$
$$P_{g3} = P_{E3} \cdot P_{E4}$$

OR gate



$$P_{G1} = P_{E1} + P_{E2} + P_{E3} + P_{E4} - (P_{E1} * P_{E2}) - (P_{E2} * P_{E3}) - (P_{E3} * P_{E4}) + (P_{E1} * P_{E2} * P_{E3}) + (P_{E1} * P_{E2} * P_{E4}) + (P_{E1} * P_{E3} * P_{E4}) + (P_{E2} * P_{E3} * P_{E4}) - (P_{E1} * P_{E2} * P_{E3} * P_{E4})$$

$$P_{G1} = P_{E1} + P_{g2} - (P_{E1} * P_{g2})$$
$$P_{g2} = P_{E2} + P_{g3} - (P_{E2} * P_{g3})$$
$$P_{g3} = P_{E3} + P_{E4} - (P_{E3} * P_{E4})$$

(6) GPGPU computing

Example:

The Fault Tree Analysis of the emergency diesel engine

ORIGINAL PROBLEM:
- 5,000 gates
- 14,000 basic events

⇩

Gate expansion and shadow gates

⇩

TRANSFORMED PROBLEM:
- 13,999 gates
- 14,000 basic events

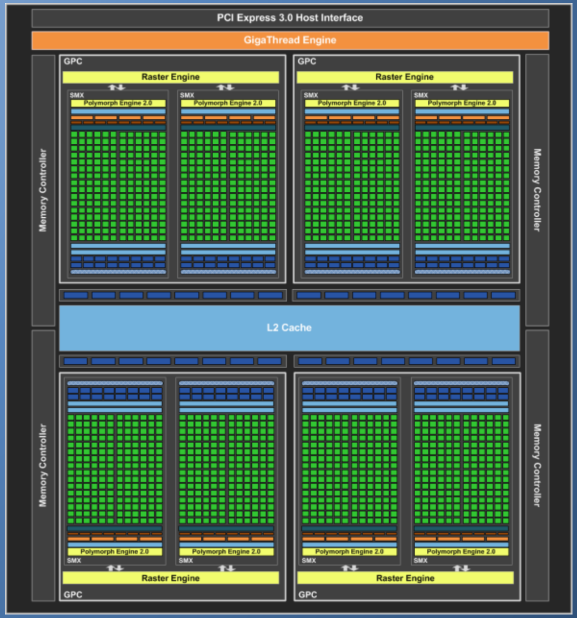
⇩

GPGPU solver

⇩

More than 2 order of magnitude speed up

x 8



Conclusion

- Parallel computing is more efficient than serial computation when dealing with Fault Tree Analysis

- It is beneficial only for large scale problem, i.e. it should be used only after the problem is well known

- The computational speed is high enough that it is possible to carry out:

  - Monte Carlo simulation with uncertain Failure Probability of the basic failure mode
  - Common cause failure analysis
  - Dynamic fault tree analysis

THANK YOU