Massachusetts Institute of Technology

Discussion of Some FAQ Submitted by Workshop Attendees

Nancy Leveson

Relationship between STPA and MIL-STD-882

- MIL-STD-882 does not specify techniques to use
  - Tasks (e.g., hazard identification, hazard analysis)
  - Risk assessment: **impossible**
  - LoR (for software): **meaningless**

- STPA satisfies almost all the tasks (which the other techniques do not)

MIL-STD-882	(1969)	(just hardware)
MIL-STD-882A	(1977)	(just hardware)
MIL-STD-882B	(1984)	Introduced special tasks for software HA
MIL-STD-882C	**(1993)**	**Integrated hardware and software tasks**
MIL-STD-882D	(2000)	minimal, just a risk assessment
MIL-STD-882E	(2012)	added tasks back, took out S/W HA and put in LoR

- See my white paper on PSAS website with details of how STPA satisfies 882E

Bottom Line

- To just satisfy standards, do whatever it tell you to do

- To make sure system is safe
  - Use 882 tasks and STPA to satisfy them (see white paper)

What if I need risk number?

- Why need it? Is a bad number worse than none?

- HA generates scenarios
  - Cannot just add quantification to STPA scenarios
  - Traditional methods leave out design errors, software, etc. for which there are no probabilities

- PRA never been shown to work, almost surely does not
  - Risk matrix (misses catastrophic scenarios)
  - PRA scientific studies?
  - Cognitive biases (confirmation bias, etc.)

    People are TERRIBLE at estimating risk.
  - Anecdotes: Accident reports, drone, rail

- Use something other than likelihood
  - Mitigatability?
  - Controllability?

- Use in prioritizing findings

Comparisons with traditional HA methods

- Linear chain-of-failure event causality models
  - Misses systemic factors and “why”

- FMEA (FFMEA):
  - Reliability (not safety, even FMECA)
  - Failures only
  - Useless results for human error, software
  - Very expensive (forward analysis)

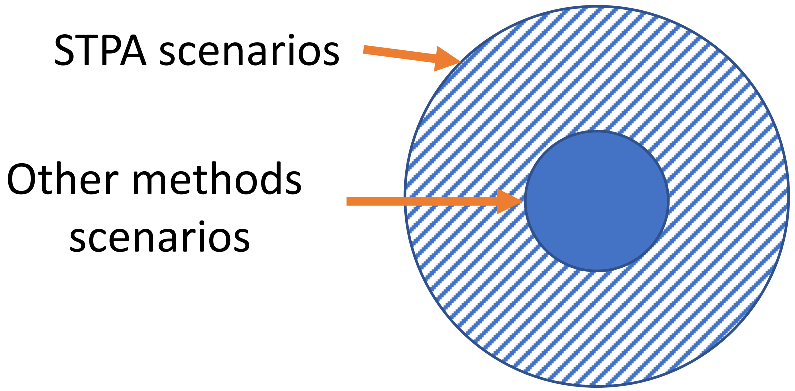
- HAZOP
  - At least has a model to work on
  - Still chain of events (failures => deviations)
  - Doesn’t work for software or complex systems (decompositional, linear)

- ETA (needs probabilities, fixed events/failures, oversimplifies)
  - Leaves out most everything

Comparisons with traditional HA methods

- FTA
  - No model of system
  - Failure events only
  - Need failure probabilities
  - Not useful for systems with software and humans

- STPA
  - Integrates humans, software, management, social aspects into analysis
  - Identifies only hazardous scenarios (backward analysis) so efficient

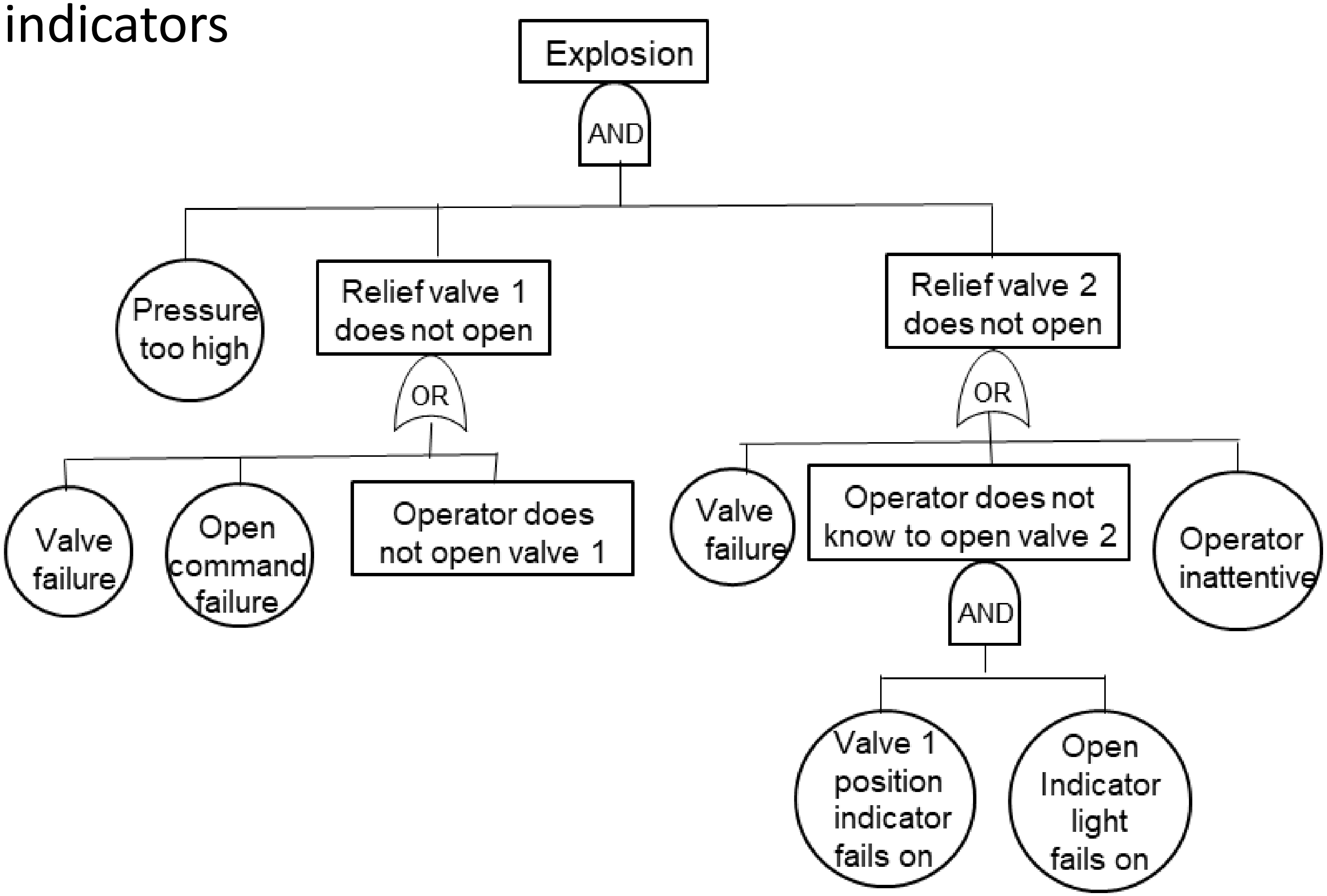


STPA scenarios

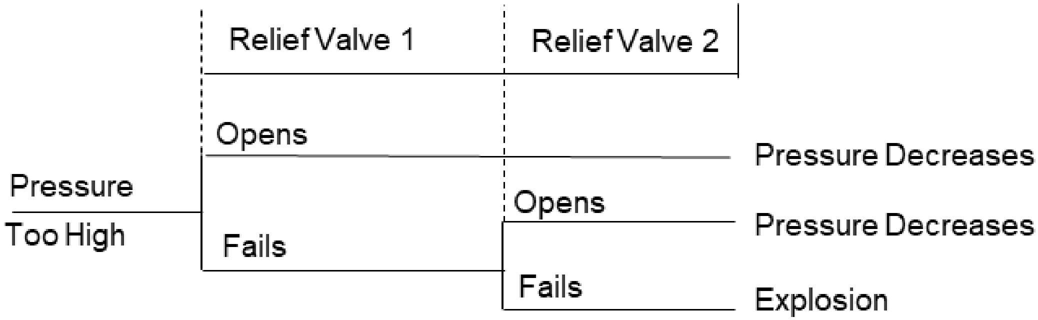
Other methods scenarios

Example Comparison FTA and STPA

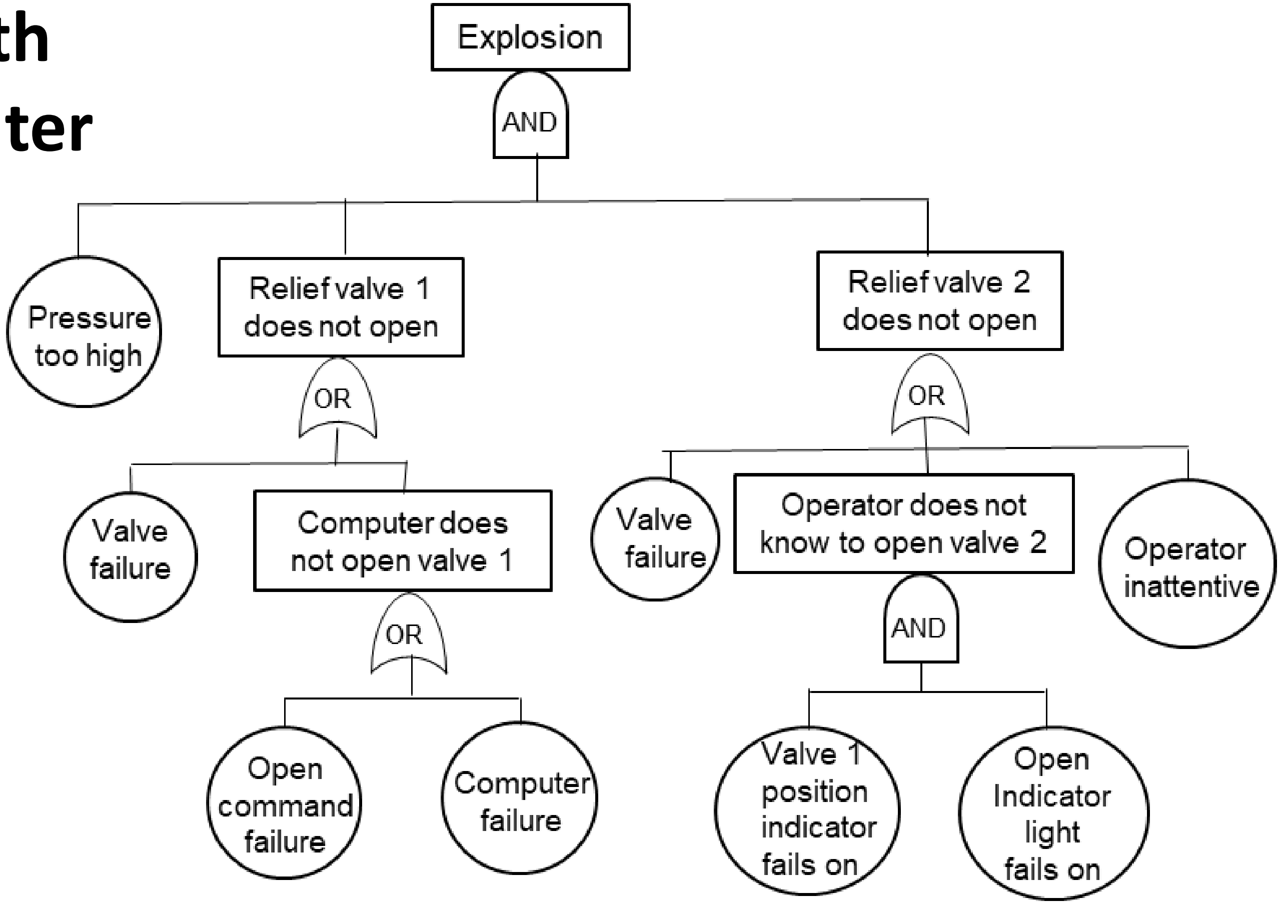
- Defense system, hazard is overpressurization
- Two relief valves added, one is backup in case other one fails
- Two indicators


Explosion
AND
Pressure too high
Relief valve 1 does not open
OR
Valve failure
Open command failure
Operator does not open valve 1
Relief valve 2 does not open
OR
Valve failure
Operator does not know to open valve 2
AND
Valve 1 position indicator fails on
Open Indicator light fails on
Operator inattentive

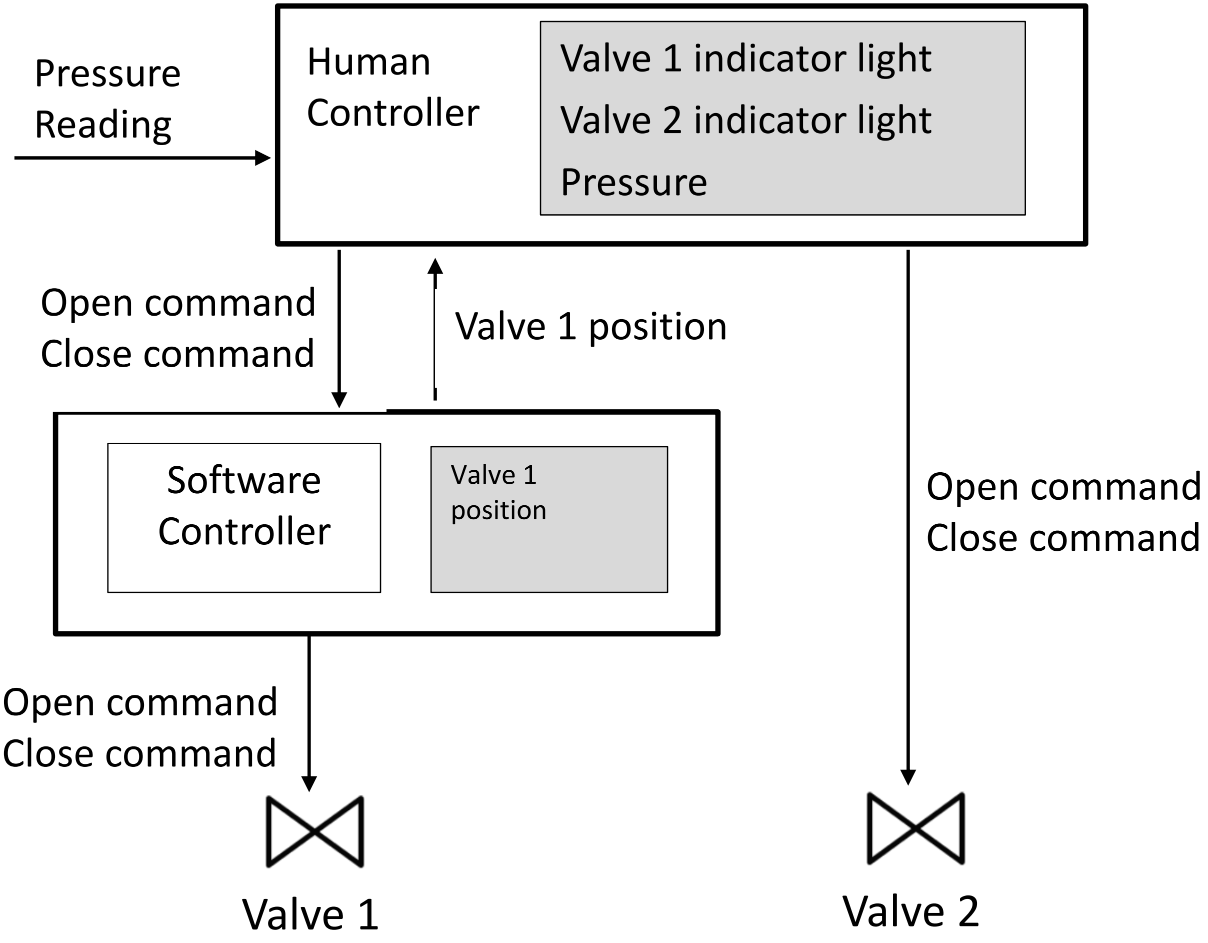
ETA



FTA with Computer


Explosion
AND
Pressure too high
Relief valve 1 does not open
OR
Valve failure
Computer does not open valve 1
OR
Open command failure
Computer failure
Relief valve 2 does not open
OR
Valve failure
Operator does not know to open valve 2
AND
Valve 1 position indicator fails on
Open Indicator light fails on
Operator inattentive

STPA Control Model (model of system design)



Fault tree is not a model of the system, it is the result of the analysis (failure events). Added pressure reading.

Some STPA unsafe control actions for human

- OP-UCA-1: Operator does not send command to software to open V1 (and also does not open V-2) when pressure is too high.
- OP-UCA-2: Operator sends command to software to open V1 when pressure too high but software does not open V1 and operator does not open V2.
- OP-UCA-3: Operator sends command to software to open V1 but too late after "*pressure too high*" condition occurs.
- OP-UCA-4: Operator sends open V2 command too late
- OP-UCA-5: Operator thinks pressure is no longer too high and closes V2 when pressure is still too high.

Some STPA UCAs for Computer

- S-UCA-1: Software does not issue open Valve 1 command when operator commands it. Sends feedback that indicates valve is open.
- S-UCA-2: Issues open Valve 1 command but valve fails. Sends feedback to say valve is open.
- S-UCA-3: Issues open Valve 1 command but too late or there is a delay in command getting to the valve. Software sends feedback that Valve 1 is open.

Example scenarios from STPA not in FTA

UCA: Operator does not send command to computer to open V1

a. Operator distracted by other emergencies or duties, related or not to the pressure, and does not read pressure monitor or does not read it in time.

b. Operator misreads pressure monitor because of human factors issue in control panel design.

c. Pressure monitor failure, wrong value provided (e.g, calibration issue somewhere in pressure sensing or some other failure in plant) or delay in displaying of value. Operator does not know to open relief valves (process model incorrect). Explosion occurs.

Other Scenarios

d. Operator sends command to software to open Valve. Software sends open valve command but valve fails or command is not received or not received in time because of failure or delay in communication line. Software does not know valve has not opened. Process model updated when sends command to valve to open. Process model says valve has opened so computer sends message back to operator that Valve 1 has opened. Operator therefore does not open Valve 2. Explosion.

e. Same as “d” but computer sends command too late because receives it too late from operator (communication delay). Software does not know valve has failed and sends message back to operator that valve 1 has opened. Operator therefore does not open Valve 2. Explosion.

And so on (lots more scenarios)

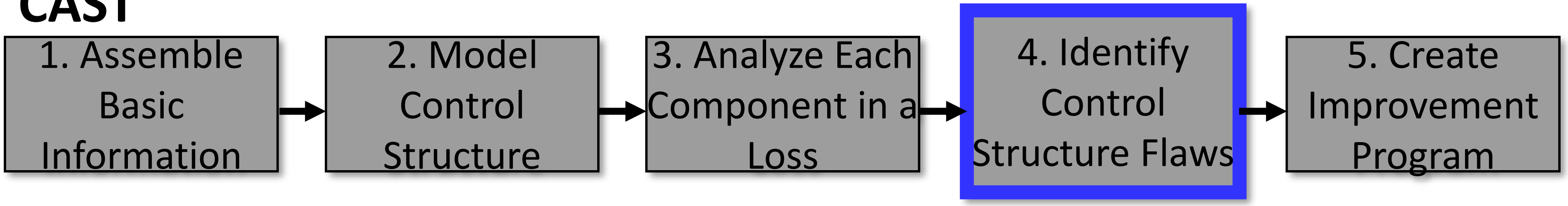
How Know Complete?

- Is this ever possible in most engineering analyses?
- There are always “unknowns unknowns” or inadequate/flawed assumptions

STPA for Organizational Factors?

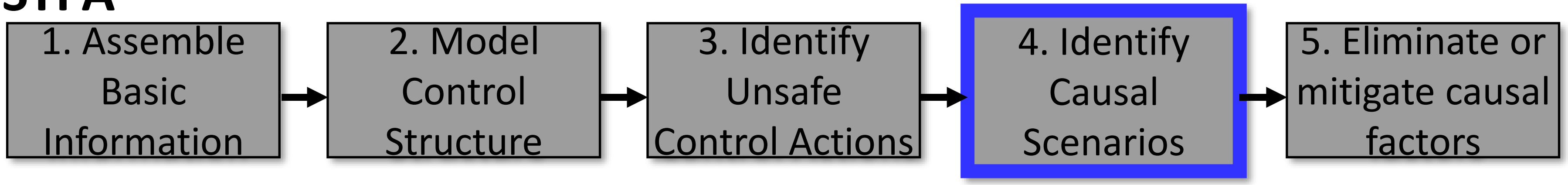
- Communication and Coordination
- Safety Culture
- Safety Information System
- Economic and Environmental Factors
- Changes and Dynamics over Time
- Ill-Defined Responsibility Authority Accountability
- Mental Model Synchronization and Update
- Management of Change
- Safety Management System
- Economics and Resources

CAST



1. Assemble Basic Information → 2. Model Control Structure → 3. Analyze Each Component in a Loss → 4. Identify Control Structure Flaws → 5. Create Improvement Program

STPA



1. Assemble Basic Information → 2. Model Control Structure → 3. Identify Unsafe Control Actions → 4. Identify Causal Scenarios → 5. Eliminate or mitigate causal factors