Fault Tree Analysis (FTA)

Kim R. Fowler
KSU ECE
February 2013
Purpose for FTA

- In the face of potential failures, determine if design must change to improve:
  - Reliability
  - Safety
  - Operation

- Secondary purposes:
  - educate designers to potential problems
  - perform root cause analysis when a fault occurs
Basic Description

- Determines sources, or root causes, of potential faults
- Qualitative and quantitative
  - Graphical, top-down approach
  - Uses Boolean algebra, logic, and probability
  - Can handle multiple failures
  - Can support probabilistic risk assessment
- Part of system design hazard analysis type (SD-HAT)
Goals of FTA

- Assess system safety
  - Top-down analysis focused on system design
  - Identifies potential root causes of failures
  - Provides a basis for reducing safety risks
  - Documentation of safety considerations
- What does it tell developer? – help find potential risks during design
- What does it tell regulator? – designers used a measure of discipline and rigor

February 2013
History of FTA

- Developed at Bell Labs for the guidance system of the U.S. Minuteman missile during the 1960s
- Used by
  - Boeing for Minuteman Weapon System
  - Regularly used by:
    - Commercial aircraft industry
    - Nuclear power industry
FTA Answers these Questions

- What are the root causes of failures?
- What are the combinations and probabilities of causal factors in undesired events?
- What are the mechanisms and fault paths of undesired events?
FTA Symbols

Event Symbols
- Basic
- Initiating
- Undeveloped
- Intermediate
- Conditioning

Gate (Combinatorial) Symbols
- OR
- Exclusive OR
- AND
- Priority AND
- Inhibit

Transfer Symbols
- In
- Out
FTA Symbolic Event Meanings

- Basic – primary, initiating fault or failure
- Initiating – normal event starting sequence
- Undeveloped – external failure propagating into system
- Intermediate – explanation of operation in fault path
FTA Simple Logic

OR gate – output occurs if one or more inputs occur

\[ P_{\text{out}} = P_A + P_B + P_C - (P_A P_B + P_A P_C + P_B P_C) + (P_{ABC}) \]

[PLEASE NOTE the even cross-terms in probability subtract while the odd cross-terms in probability sum]

AND gate – output occurs only if all inputs occur together

\[ P_{\text{out}} = P_A \cdot P_B \cdot P_C \]
FTA Exclusive and Inhibit Logic

**Exclusive OR gate** — output occurs *ONLY* if one input occurs but *NOT* any others. May need to attach a condition statement to the gate to explain the rationale.

\[ P_{\text{out}} = P_A + P_B + P_C - 2(P_A P_B + P_A P_C + P_B P_C) + 4(P_{ABC}) \]

*PLEASE NOTE* the even cross-terms in probability subtract while the odd cross-terms in probability sum and that cross-terms have even coefficients.

**Priority AND gate** — output occurs *ONLY* if all inputs occur together but *ONLY* in sequence, first A, then B, then C. May need to attach a condition statement to the gate to explain the rationale.

\[ P_{\text{out}} = P_A \cdot P_B \cdot P_C / N! \]

*PLEASE NOTE* this expression is only true for equal rates of faults, that is \( ?_A = ?_B = ?_C \) and \( N = \) number of inputs to the gate, 3 in this example; otherwise the probability derivation becomes more complex.

**Inhibit gate** — output occurs only if input occurs and the attached condition is satisfied. This is a particular instantiation of a 2-input AND gate.

\[ P_{\text{out}} = P_{\text{input}} \cdot P_{\text{condition}} \]
FTA Methodology

1. Understand the system design and operation.
2. Describe the problem and the course of the event.
3. Define and record the scope of the problem and analysis.
4. Build fault path with logic gates and connections.
5. Generate cut sets and probabilities for the fault.
6. Does the model accurately reflect the design?
7. Make any changes to model if validation warrants them.
8. Store and archive the results.
Step 1: Define the System

- Collect design
  - Requirements
  - Source Code
  - Models
  - Schematics
- Layout concept of operations or CONOPs
- Understand the system behavior
Step 2: Define Undesired Event

- Identify the final outcome of the undesired event
- Identify sub-events that lead to final event
- Begin to structure the connections
- - but - -
- Do Step 3 before completing structure of connections
Step 3: Establish Rules

- Define analysis ground rules boundaries
- Concepts that you can (should) use:
  - I-N-S:
    - “What is immediate (I), necessary (N), and sufficient (S) to cause the event?”
    - Helps focus on event chain
    - Helps analyst from jumping ahead
  - SS-SC: “What is the source of the fault?”
    - If component failure – classify as SC (state-of
Step 3: (continued)

- P-S-C: (Ericson, Fig. 11.8, p. 194)
  - “What are the primary (P), secondary (S), and command (C) causes of the event?”
  - Helps focus on specific causal factors

- SS-SC:
  - If component failure – classify as SC (state-of-the-component) fault
  - If not component failure – classify as SS (state-of-the-system) fault
  - If fault is SC, then event ORs P-S-C inputs
  - If fault is SS, then develop event further with using I-N-S logic
Step 4: Building Tree

- Repetitive process
- Ericson, Fig. 11.9, p. 195
- At each level determine
  - Cause
  - Effect
  - Logical combination using logic symbols
- Construction rules (see Ericson, pp. 196 – 197), these are almost self-evident but still good, disciplined techniques
Step 5: Establish Cut Sets

- Cut set – critical path(s) of sub-event combinations that cause the undesirable final state event
- Ericsson provides in-depth mathematical treatment of cut sets and probabilities on pp. 199 – 206
- Often, mere inspection will reveal the weak links that indicate the most important cut set(s) that lead to the event
EXAMPLE OF INCUBATOR
ISOLETTE
Example - Incubator Isolette

Step 1: Define the System

- For simplicity, use the previous diagram as the system model
- Recognize several different subsystems:
  - Controls
  - Display
  - Heater with closed loop thermal sensor
  - Airflow fan and ductwork
  - Independent thermal safety interlock
  - Medical staff operating controls and display
  - Patient receiving output (warmed air)
Step 2: Define Undesired Event

- Undesired event: “Air is not warmed.”
- Sub-events:
  - Operations error
  - Heater fault or failure
  - Air handling system fault or failure
  - Thermal safety system fault or failure
Step 3: Analysis Ground Rules

- Understand process concepts:
  - I-N-S
  - P-S-C
  - SS-SC
Step 4: Construct Fault Tree

- (from Step 2, collect events) These are SS faults, so OR them together
- Proceed to next level
  - Determine underlying events
  - Apply process concepts:
    - I-N-S
    - P-S-C
    - SS-SC
    - Connect them together with logical linkages
- Repeat process for lower levels
Steps 5-7: Find Fault Paths

- Inspect paths for possible faults
- Generate the cut sets
  - (for simplicity in this introduction, we are using inspection)
  - Ericson gives detailed instructions for
    - automating the selection of cut sets
    - calculating probabilities of occurrence
Ex. - Isolette Warm Air Fault, Collecting Event and Sub-events

- Undesirable event: Air is not warmed
- Sub-events:
  - Operational mistake
  - Heater subsystem failure
  - Airflow damper/control failure blocks air from heating element
  - Thermosafety switch failure
Ex. - Isolette Warm Air Fault, Develop Fault Paths for Sub-events, Part 1
Ex. - Isolette Warm Air Fault, Develop Fault Paths for Sub-events, Part 2
Develop path for sub-event, note that there are two lower levels here, with priority inputs in the logic.
Ex. - Isolette Warm Air Fault, Part 4: Final Version of Fault Tree

- Air is not warmed
  - Operational mistake
    - Does not observe mistake
      - Maliciously sets low temperature
      - Incorrectly sets low temperature
        - Ignorantly sets low temperature
  - Heater subsystem failure
    - Heater controls fail
      - Heater element fails
      - Power or supply fails
      - Thermal sensor fails
  - Airflow damper/control failure blocks air from heating element
    - Damper mechanism fails
      - Fan or air pump fails
      - Airflow controls fail
      - Power or supply fails

Develop path and condition for sub-event, note that this sub-event has a conditional inhibit controlling it

- Thermosafety switch failure
  - Heater subsystem operates correctly
  - Switch fails open
Ex. - What do you do now?

- For design purposes:
  - Review each path
    - Can you eliminate that path?
    - If not, can it be made more fault resistant?
  - Does fault tree represent the scope of possible paths (and reasonable – a meteor falling out of the sky and hitting it is not)?

- For root cause analysis:
  - Does the evidence point to any fault path?
  - If so, fix the problem.
  - If not, revise the diagram.
CLASS EXERCISES - PROBLEM #1
Step 1: Define the System (done)

- For simplicity, use the previous diagram as the system model
- Recognize several different subsystems (done already)
Step 2: Define Undesired Event

- Undesired event: “No airflow.”
- Sub-events:
  - Operations error
  - Air handling system fault or failure
- Eliminate sub-events and subsystems that do not interact or control the air handling system:
  - Heater fault or failure
  - Thermal safety system fault or failure
Step 3: Analysis Ground Rules

- Understand process concepts:
  - I-N-S
  - P-S-C
  - SS-SC
Step 4: Construct Fault Tree

- These are SS faults, so OR them together
- Proceed to next level
  - Determine underlying events - Operations
    - Assume that medical staff does not directly control airflow from interface panel
    - Blocking air inlet
      - Malicious
      - Isolette inlet up against wall or obstruction
      - ______________________(hint – ignorance)
Step 4: (continued)

- Determine underlying events – air handling
  - ____________________________ (hint – fan)
  - ____________________________ (hint – what directs airflow?)
  - ____________________________ (hint – problem with control signal)
  - ____________________________ (hint – electrical current into subsystem)

- Apply process concepts
- Connect them together with logical linkages
Exercise - Isolette Airflow Fault

No airflow

Operational mistake

maliciously blocks air inlet

blocks air inlet

air inlet against wall or obstruction

Airflow subsystem
Ex. - What do you do now?

- For design purposes:
  - Review each path
    - Can you eliminate that path?
    - If not, can it be made more fault resistant?
  - Does fault tree represent the scope of possible paths (and reasonable – a meteor falling out of the sky and hitting it is not)?

- For root cause analysis:
  - Does the evidence point to any fault path?
  - If so, fix the problem.
  - If not, revise the diagram.
Solution - Isolette Airflow Fault

No airflow

Operational mistake
- maliciously blocks air inlet
- air inlet against wall or obstruction
- misguided or ignorant blocking of air inlet

Airflow subsystem including fan, valves and dampers, power, and control

Mechanical or electrical fault within fan
Mechanical or electrical fault within valves or dampers
Electrical fault within power supply or connection
Electrical fault within control panel or connection
CLASS EXERCISES - PROBLEM #2
Step 1: Define the System (done)

- For simplicity, use the previous diagram as the system model
- Recognize several different subsystems (done already)
Step 2: Define Undesired Event

- Undesired event: “Failure alarm sounds.”
- Sub-events:
  - Operations error
  - Air handling system fault or failure
  - Heater fault or failure
  - Thermal safety system fault or failure
  - Diagnostic subsystem fault or failure
Step 3: Analysis Ground Rules

- Understand process concepts:
  - I-N-S
  - P-S-C
  - SS-SC
Step 4: Construct Fault Tree

- These are SS faults, so OR them together
- Proceed to next level down:
  - Determine operation faults or failures
    - ____________________________
    - ____________________________
    - ____________________________
    - ____________________________
    - ____________________________
Step 4: (continued)

- Determine heater subsystem faults or failures
  - 
  - 
  - 
  - 
  - 

- Determine air handling subsystem faults
  - 
  - 
  - 
  - 
  - 
Step 4: (continued)

- Determine thermosafety switch faults
  - _______________________________
  - _______________________________
  - _______________________________
  - _______________________________

- Determine alarm subsystem faults
  - _______________________________
  - _______________________________
  - _______________________________
  - _______________________________
Step 4: (continued)

- Apply process concepts
- Connect them together with logical linkages
Exercise - Isolette Alarm Sounds

- Failure alarm sounds
- Operational mistake
- Heater subsystem failure
- Diagnostic/alarm subsystem failure
- Airflow damper/control failure
- Themosafety switch failure
Ex. - What do you do now?

■ For design purposes:
  ■ Review each path
    ■ Can you eliminate that path?
    ■ If not, can it be made more fault resistant?
  ■ Does fault tree represent the scope of possible paths (and reasonable – a meteor falling out of the sky and hitting it is not)?

■ For root cause analysis:
  ■ Does the evidence point to any fault path?
  ■ If so, fix the problem.
  ■ If not, revise the diagram.
Solution - Isolette Alarm Sounds

- Failure alarm sounds
  - Operational mistake
    - Does not observe mistake
      - Maliciously sets low temperature
      - Incorrectly sets low temperature
  - Heater subsystem failure
    - Heater controls fail
    - Thermal sensor fails
    - Heater element fails
    - Power or supply fails
  - Diagnostic/alarm subsystem failure
    - Alarm circuit fails
    - Power or supply fails
  - Airflow damper/control failure
    - Damper mechanism fails
    - Airflow controls fail
    - Fan or air pump fails
    - Power or supply fails
  - Thermosafety switch failure
    - Switch fails open
    - Heater subsystem operates correctly
From satellite imaging systems, blank screen on ground support equipment.

FINAL EXAMPLE
Example FTA (from aerospace)
FINAL THOUGHTS ON FTA
FTA Advantages

- Structured and rigorous
- Easily understood via visual format
- Combines hardware, software, environment, and human operations
- Can do probability assessment
- Commercial software available
FTA Disadvantages

- Can be very time consuming
- Limitations
  - Almost impossible to model:
    - timing and scheduling
    - intermittent faults or injected noise
  - Does not identify hazards unrelated to failure
  - Limited examination of software
- Requires system/product expertise
Parting Comments

- FTA should be used in combination with other analytical tools, not as sole tool for hazard analysis
- FTA only models fault paths, not all events
- This introduction did not cover all the probability assessments or the processes for cut sets
Reference

- Based on MIL. STD. 882.