21st International Congress on Modelling and Simulation, Gold Coast, Australia, 29 Nov to 4 Dec 2015
www.mssanz.org.au/modsim2015

Improvements in Analysing Failure of Defence Systems for Operations Analysis

Tristan Goss [a], Seth Thuraisingham [b], Kym Ide [a] and Kristian Allison [a]

[a] Defence Science and Technology Group, Department of Defence
[b] Consilium Technology Pty Ltd
Email: Tristan.Goss1@Defence.gov.au

Abstract: Fault Tree Analysis (FTA) is an established process for predicting the results of events on systems. Defence Science and Technology (DST) Group apply FTA to analyse the efficacy of helicopter Tactics, Techniques and Procedures (TTPs). In this context FTA allows the consequence of individual damage to be quantified. For example, what is the consequence of damage to the aircraft's left engine, ammunition bay and communications equipment? This provides quantified answers to the question 'does the TTP result in a higher chance of the aircraft completing its mission safely'.

This paper uses standard FTA terminology, consistent with the 'fault tree' analogy. The 'root' of the tree is the desired answer ('will the aircraft complete its mission safely?'), the 'leaves' of the tree are 'basic events' (e.g. the left engine has failed), the 'branches' of the tree are 'intermediary events' depending on multiple basic or intermediary events ('both engines have failed') and the logical gates for combining events to form new events are where the branches of the tree split. Conventionally when drawn as a diagram the 'tree' is usually drawn upside down with the 'root' at the top of the diagram.

Defence application of FTA is different to civilian application of FTA as Defence failure modes which result from adversarial intent differ from civilian failure modes. As Defence needs have developed it was necessary to build a more capable FTA suite, **FAU**lt **N**etwork **A**nalysis (FAUNA). FAUNA introduces NOT, INHIBIT, N of M and TRUTH gates that enable the analysis of more detailed questions and produce results with more impact. Examples of why these new gates are required are provided using the Armed Reconnaissance Helicopter as a case study.

FAUNA is available to DST Group staff on request, and has also been adopted by the Defence Science and Technology Laboratory (Dstl) in the United Kingdom. FAUNA is also available under commercial agreement.

Keywords: *Fault tree, system analysis, Defence, tactics, failure*

1. INTRODUCTION

Fault Tree Analysis (FTA) is a process used to consider the effects of events on systems, and is regularly used in civilian applications. The application of this analysis technique for complex systems with civilian failure modes has been discussed in detail in both the Nuclear Regulatory Commission (NRC) and the National Aeronautics and Space Administration (NASA) Fault tree handbooks developed for their respective applications, Vesely et al (1981) and Stamatelatos et al (2002).

Defence Science and Technology (DST) Group apply FTA to analyse the efficacy of helicopter Tactics, Techniques and Procedures (TTPs). In this context FTA allows the consequence of individual damage to be quantified. For example, what is the consequence of damage to the aircraft's left engine, ammunition bay and communications equipment? This provides quantified answers to the question 'does the TTP result in a higher chance of the aircraft completing its mission safely'.

This paper uses standard FTA terminology, consistent with the 'tree' analogy. The 'root' of the tree is the desired answer ('will the aircraft complete its mission safely?'), the 'leaves' of the tree are 'basic events' (e.g. the left engine has failed), the 'branches' of the tree are 'intermediary events' depending on multiple basic or intermediary events ('both engines have failed') and the logical gates for combining events to form new events are where the branches meet or split. Conventionally when drawn as a diagram the 'tree' is usually drawn upside down with the 'root' at the top of the diagram.

The underlying mathematics used in Australian Defence applications is Bayesian probability, consistent with many fault tree applications and the NASA fault tree handbook (Stamatelatos et al 2002) remains the primary reference text in Weapons and Combat Systems Division fault tree development. In DST Group convention each node represents a failure mode that is binary. Examples of nodes would be "left engine failed" or "hydraulic pressure below 50% threshold". Counter examples would be "fuel temperature >50°C" (a system property, not a failure mode) or "right engine undamaged" (survival, not failure).

Defence application of FTA is different to civilian application of FTA as military failure modes differ from civilian failure modes due to adversarial intent. In the NRC and NASA handbooks the probability of component failure is routinely in the order of 1 in 100 to 1 in 1,000,000 events. These relatively low failure probabilities result from failure due to fatigue, wear or defects. In a defence environment components are exposed to new failure modes, they may fail because they are overloaded or because they are physically damaged by weapons. These active effects can create probabilities of component failure from the order of 1 in 10 to 1 in 1 (certainty).

FTA analysis packages designed for the civilian market are readily available, Ruijtersy and Stoelingay (2015) provide an overview of the capability and limitations of five existing packages. However as Defence needs developed, it was necessary to build an FTA suite unique to DST Group's requirements, **FAU**lt **N**etwork **A**nalysis (FAUNA). FAUNA introduces improvements to previous DST Group FTA programs including introduction of INHIBIT, NOT, N of M and TRUTH gates. Examples of why these new gates are required are provided using the Armed Reconnaissance Helicopter as a case study.

2. THE 'INHIBIT' GATE

In DST Group's implementation of FTA the 'leaf nodes' of the fault tree represent ways for physical components to fail (the computer is hit by a bullet, so it fails). However failure of the system may depend on both system properties (such as airspeed and temperature) and the physical component failure. The INHIBIT gate is designed to predict failure on the basis of both system properties and leaf nodes, and is the only gate to do so. Not all FTA methods make a distinction between properties and events, but the reason we do at DST Group is that the properties and failure mechanisms change in different ways.

Properties will change many times over the course of an analysis, for example fuel temperature increases as a function of time over a simulated mission. Physical failure modes may change, but are more likely to remain constant for a known physical effect. In addition to the frequency of change, the value of the nodes is set by different mechanisms in the larger simulation. Failure modes are set by 'damage algorithms' while properties are set by the constructive simulation.

System properties do not have to be physical, in addition to physical properties like temperature the result may also change on the basis of scenario information. For example a gate could be designed to give different

results depending on the phase of the mission. Damage that would be critical in early stages of a mission may be acceptable in later stages, damage might be less critical when the aircraft is flying back to base.

As seen in Figure 1 the INHIBIT gate takes system properties and nodes and uses these as input to calculate the chance of an intermediate event. In this example the ARH carries a fuel tank which contains a mixture of fuel, fuel vapour and air called ullage. The potential for a fuel fire to occur depends on a system property "ullage temperature" as well as a direct failure mode such as a spark or an incendiary round. The nodes representing the failure modes feed into the gate while the system properties (ullage temperature above a threshold temperature) may inhibit the output. Despite the spark there are certain temperature ranges where the fuel fire will not occur.

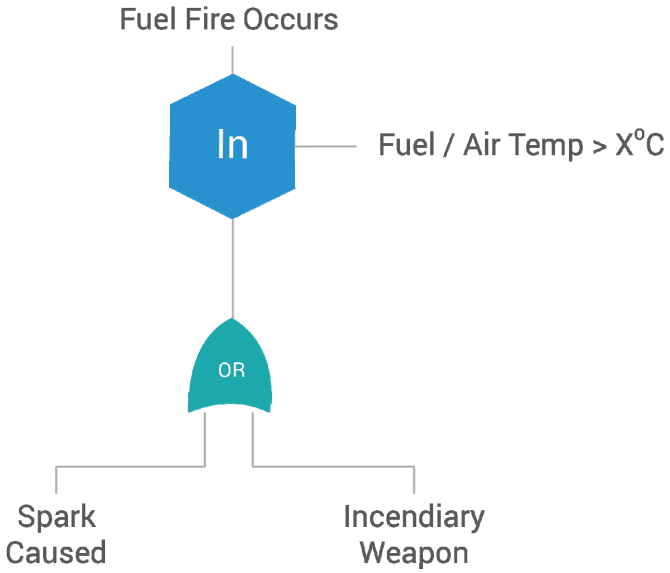

Figure 1. INHIBIT Gate

3. THE 'NOT' GATE

Leaf nodes are always written in terms of component failure not in terms of component survival, which can cause problems if we want to answer certain questions. While AND and OR gates are generally sufficient for finding the overall chance of failure, they cannot determine the chance of a certain failure configuration. In the tactical domain for ARH we use this to phrase questions of the form 'in a damaged state, how capable is the aircraft for certain missions?' or 'where can I introduce redundancy to improve system performance?'

For example, consider two possible ARH missions:

- Reconnaissance Patrol, where the focus is on sensing
- Fighting Patrol, where the focus is on harassing or destroying enemy forces

In simple terms, we can imagine that given different types of damage, an aircraft might be incapable of performing one or both of these missions. Damage to the sensors might preclude reconnaissance, while damage to the weapon systems might preclude a fighting patrol. Use of the NOT gate allows us to build trees that answer questions in the form:

- What is the chance that the aircraft can perform a Fighting Patrol but NOT a Reconnaissance Patrol?
- What is the chance that the aircraft can perform a Reconnaissance Patrol but NOT a Fighting Patrol?

These questions are important for accurately determining the capabilities of ADF air assets in a complex, contested, tactical environment. If we discover that certain scenarios are likely to lead to the disabling of the

sensing systems but not the weapon systems, this information can be used as justification for improving the survivability of those systems.

4. THE 'N OF M' GATE

Designers often have two main methods of reducing system vulnerability. Each component can be made more resilient (less likely to fail) or components can be made redundant in backup systems. Redundant systems result in larger fault trees where the overall system behaviour is less obvious. The core idea of redundancy is that the system can survive a known number of failures out of a pool of total components. We write this as 'N of M' because the test is whether 'N' number of 'M' components will fail, where N and M are both integers.

Consider a communications system containing has 4 radio units and assume the system will fail if 3 of the units fail. Using AND, OR and NOT gates, it would be possible to construct an expansive fault tree for every combination of three out of four failures. This results in a large tree shown as Figure 2.

In the expanded table note that NOT gates are used to create unique combinations of nodes. Without the NOT gates it would not be possible to create branches that determine if three and only three component fail. By using the NOT gate we can check the survival of the fourth component, making the combinations unique. Without the NOT gates we would run the risk of combining dependent events which would result in an invalid tree as the OR gate assumes independence of events.

Compare Figure 2 to Figure 3 where an 'N of M' gate is used. Not only has the substitution increased the clarity of the tree by clearly defining the purpose of the structure, but it has also saved analyst time.

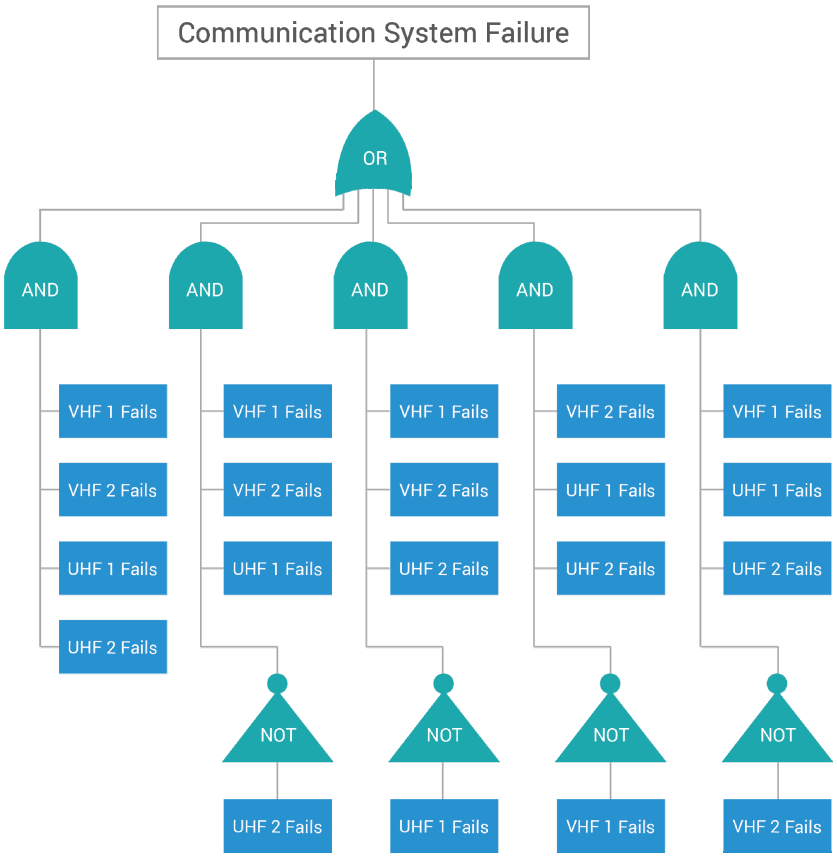

Figure 2. The 3 of 4 gate structure expressed using only AND, OR and NOT gates

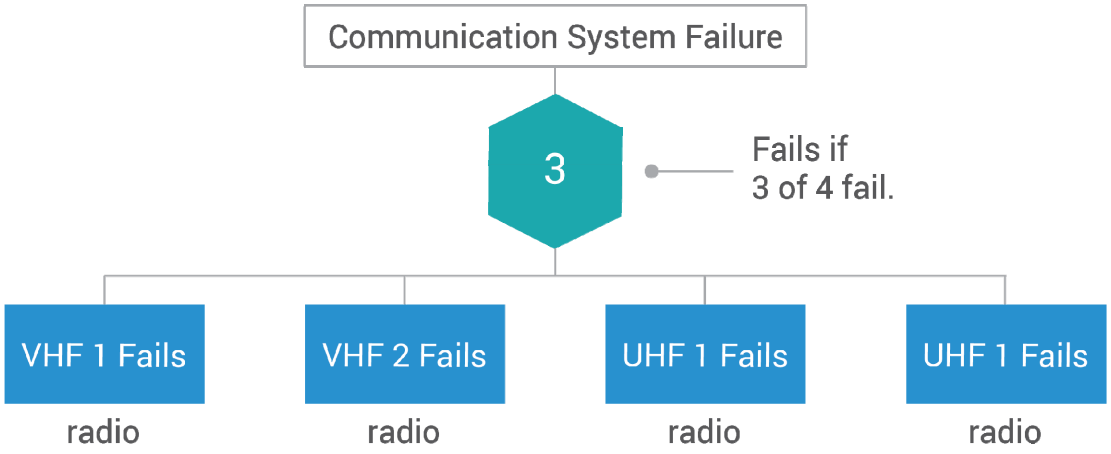

Figure 3. N of M gate (3 of 4)

In this case even a small, relatively simple example was cumbersome to expand using AND, OR and NOT gates. We might wish to consider a problem related to aircraft frames, where to the failure case might be '5 out of 30', which completely impractical to express this way.

5. THE 'TRUTH' GATE

TRUTH gates provide a compact way of describing system behaviour when it is not possible to assign simpler forms of failure logic, such as AND and OR. In aircraft systems this type of gate would be used for systems containing non-mutually exclusive failure modes. For example, consider the engine system defined in Figure 4.

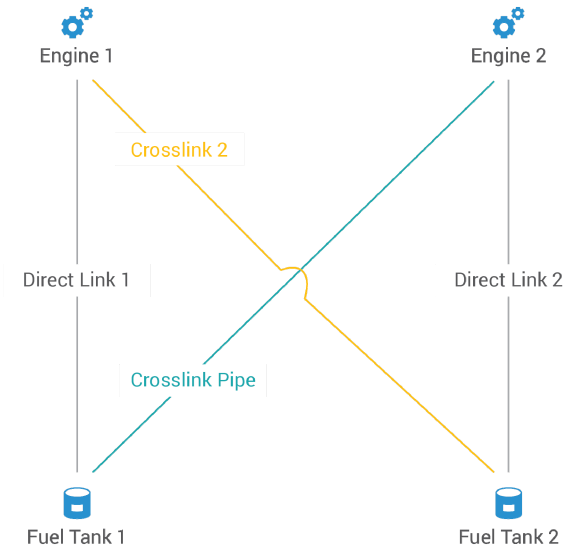

Figure 4. Cross Linked Engine System

The system has two engines, two fuel tanks and four fuel lines. The system will work as long as at least one engine is working and can access fuel (through either a crosslink or the direct link). By inspection there are several obvious ways the system can fail:

- All four pipes fail,
- Both the engines fail, or
- Both the fuel tanks fail

There are also some less obvious ways for the system to fail. The system will also fail if Engine 1, Direct link 2 and Crosslink 1 fail. This is because Engine 1 has failed and Engine 2 cannot access fuel. There are many other possible combinations of failure.

The most important thing about these failures is that they are not mutually exclusive, the pipes failing does not preclude the fuel tanks from failing independently. The difficulty here is that we capture the failure once and only once. If we are not careful we could combine dependent events which will give us the wrong answer. Additionally because the tree can get complex, it may not be immediately noticeable that we have made this error.

A straightforward way to implement a solution is the TRUTH gate. The TRUTH gate represents every possible unique combination of the inputs. Because each combination is unique and mutually exclusive each combination can be assessed in isolation before being independently combined.

Similar to the N of M gate, the TRUTH gate does not provide capability that was not possible with AND, OR and NOT. However, to accurately model the example with AND, OR and NOT gates would require over 2048 nodes connected via 1281 gates. While possible, this is clearly not practical or efficient. The TRUTH table approach introduces 1 gate and the assessment of 256 combinations, which can be assessed quickly through truth table analysis and logical substitution.

6. DISCUSSION AND CONCLUSION

While there is commonality with civilian techniques for fault analysis, FAUNA addresses unique requirements and challenges of a military failure analysis tool. FAUNA is available to DST Group staff on request, and has also been adopted by the Defence Science and Technology Laboratory (Dstl) in the United Kingdom. FAUNA has contributed to an increase in capability in analyzing aircraft vulnerability for DST Group. FAUNA is also available under commercial agreement and the techniques established remain applicable to civilian areas such as infrastructure, healthcare and finance FTA.

REFERENCES

Ruijtersy, E and Stoelingay M (2015) Fault Tree Analysis: A survey of the state-of-the-art in modeling, analysis and tools, *Computer Science Review*, 15–16: 29-62

Stamatelatos, M. Vesely W.E, Dugan, J. Fragola, J. Minarick, J. and Railsback, J. (2002) Fault Tree Handbook with Aerospace Applications v1.1, National Aeronautics and Space Administration (NASA)

Tkalcevic, F.J, Burman, N.M. (1992) GRAFTED – Graphical Fault Tree Editor, Materials Research Laboratory, Publication No. MRL-GD-0043, Defence Science and Technology Group

Vesely, W.E, Goldberg, F.F. Roberts, N.H. and Haasl, D.F. (1981) Fault Tree Handbook, United States Nuclear Regulatory Commission