Fault tree analysis

Fault Tree Analysis

Fault tree analysis (FTA) is a failure analysis in which an undesired state of a system is analyzed using boolean logic to combine a series of lower-level events. This analysis method is mainly used in the field of safety engineering to quantitatively determine the probability of a safety hazard.

An Overview of Basic Concepts

This quick subject guide provides an overview of the basic concepts in Fault Tree Analysis (FTA, system analysis) as it applies to system reliability and a directory of some other resources on the subject.

History of Fault Tree Analysis (FTA)

Fault Tree Analysis (FTA) is another technique for reliability and safety analysis. Bell Telephone Laboratories developed the concept in 1962 for the U. S. Air Force for use with the Minuteman system. It was later adopted and extensively applied by the Boeing Company. Fault tree analysis is one of many symbolic " analytical logic techniques" found in operations research and in system reliability. Other techniques include Reliability Block Diagrams (RBDs).

Fault Tree Analysis (FTA) was originally developed in 1962 at Bell Laboratories by H. A. Watson, under a U. S. Air Force Ballistics Systems Division contract to evaluate the Minuteman I Intercontinental Ballistic Missile (ICBM) Launch Control System. Following the first published use of FTA in the 1962 Minuteman I Launch Control Safety Study, Boeing and AVCO expanded use of FTA to the entire Minuteman II system in 1963-1964. FTA received extensive coverage at a 1965 System Safety Symposium in Seattle sponsored by Boeing and the University of Washington. Boeing began using

FTA for civil aircraft design around 1966. In 1970, the U. S. Federal Aviation Administration (FAA) published a change to 14 CFR 25. 1309 airworthiness regulations for transport aircraft in the Federal Register at 35 FR 5665 (1970-04-08). This change adopted failure probability criteria for aircraft systems and equipment and led to widespread use of FTA in civil aviation.

Within the nuclear power industry, the U. S. Nuclear Regulatory Commission began using probabilistic risk assessment (PRA) methods including FTA in 1975, and significantly expanded PRA research following the 1979 incident at Three Mile Island. This eventually led to the 1981 publication of the NRC Fault Tree Handbook NUREG-0492, and mandatory use of PRA under the NRC's regulatory authority.

Fault Tree Analysis (FTA) attempts to model and analyze failure processes of engineering and biological systems. FTA is basically composed of logic diagrams that display the state of the system and is constructed using graphical design techniques. Originally, engineers were responsible for the development of Fault Tree Analysis, as a deep knowledge of the system under analysis is required.

Often, FTA is defined as another part, or technique, of reliability engineering. Although both model the same major aspect, they have arisen from two different perspectives. Reliability engineering was, for the most part, developed by mathematicians, while FTA, as stated above, was developed by engineers.

Fault Tree Analysis usually involves events from hardware wear out, material failure or malfunctions or combinations of deterministic contributions to the

event stemming from assigning a hardware/system failure rate to branches or cut sets. Typically failure rates are carefully derived from substantiated historical data such as mean time between failure of the components, unit, subsystem or function. Predictor data may be assigned. Assigning a software failure rate is elusive and not possible. Since software is a vital contributor and inclusive of the system operation it is assumed the software will function normally as intended. There is no such thing as a software fault tree unless considered in the system context. Software is an instruction set to the hardware or overall system for correct operation. Since basic software events do not fail in the physical sense, attempting to predict manifestation of software faults or coding errors with any reliability or accuracy is impossible, unless assumptions are made. Predicting and assigning human error rates is not the primary intent of a fault tree analysis, but may be attempted to gain some knowledge of what happens with improper human input or intervention at the wrong time.

FTA can be used as a valuable design tool, can identify potential accidents, and can eliminate costly design changes. It can also be used as a diagnostic tool, predicting the most likely system failure in a system breakdown. FTA is used in safety engineering and in all major fields of engineering. More on Fault Tree Diagram (FTD)

Fault tree diagrams (or negative analytical trees) are logic block diagrams that display the state of a system (top event) in terms of the states of its components (basic events). Like reliability block diagrams (RBDs), fault tree diagrams are also a graphical design technique, and as such provide an alternative to methodology to RBDs.

An FTD is built top-down and in term of events rather than blocks. It uses a graphic " model" of the pathways within a system that can lead to a foreseeable, undesirable loss event (or a failure). The pathways interconnect contributory events and conditions, using standard logic symbols (AND, OR etc). The basic constructs in a fault tree diagram are gates and events, where the events have an identical meaning as a block in an RBD and the gates are the conditions.

Fault Trees and Reliability Block Diagrams

The most fundamental difference between FTDs and RBDs is that in an RBD one is working in the " success space", and thus looks at system successes combinations, while in a fault tree one works in the " failure space" and looks at system failure combinations. Traditionally, fault trees have been used to access fixed probabilities (i. e. each event that comprises the tree has a fixed probability of occurring) while RBDs may have included time-varying distributions for the success (reliability equation) and other properties, such as repair/restoration distributions.

Drawing Fault Trees: Gates and Events

Fault trees are built using gates and events (blocks). The two most commonly used gates in a fault tree are the AND and OR gates. As an example, consider two events (or blocks) comprising a Top Event (or a system). If occurrence of either event causes the top event to occur, then these events (blocks) are connected using an OR gate. Alternatively, if both events need to occur to cause the top event to occur, they are connected by an AND gate. As a visualization example, consider the simple case of a system comprised of two components, A and B, and where a failure of either

component causes system failure. The system RBD is made up of two blocks in series (see RBD configurations), as shown next:

The fault tree diagram for this system includes two basic events connected to an OR gate (which is the " Top Event"). For the " Top Event" to occur, either A or B must happen. In other words, failure of A OR B causes the system to fail.

Relationships Between Fault Trees and RBDs

In general (and with some specific exceptions), a fault tree can be easily converted to an RBD. However, it is generally more difficult to convert an RBD into a fault tree, especially if one allows for highly complex configurations. The following table shows gate symbols commonly used in fault tree diagrams and describes their relationship to an RBD. (The term " Classic Fault Tree" refers to the definitions as used in the Fault Tree Handbook (NUREG-0492) by the U. S. Nuclear Regulatory Commission).

Methodology

FTA methodology is described in several industry and government standards, including NRC NUREG-0492 for the nuclear power industry, an aerospace-oriented revision to NUREG-0492 for use by NASA, SAE ARP4761 for civil aerospace, MIL-HDBK-338 for military systemsfor military systems. IEC standard IEC61025 is intended for cross-industry use and has been adopted as European Norme EN61025.

Since no system is perfect, dealing with a subsystem fault is a necessity, and any working system eventually will have a fault in some place. However, the probability for a complete or partial success is greater than the probability of

a complete failure or partial failure. Assembling a FTA is thus not as tedious as assembling a success tree which can turn out to be very time consuming.

Because assembling a FTA can be a costly and cumbersome experience, the perfect method is to consider subsystems. In this way dealing with smaller systems can assure less error work probability, less system analysis. Afterward, the subsystems integrate to form the well analyzed big system.

An undesired effect is taken as the root ('top event') of a tree of logic. There should be only one Top Event and all concerns must tree down from it. Then, each situation that could cause that effect is added to the tree as a series of logic expressions. When fault trees are labeled with actual numbers about failure probabilities (which are often in practice unavailable because of the expense of testing), computer programs can calculate failure probabilities from fault trees.

The Tree is usually written out using conventional logic gate symbols. The route through a tree between an event and an initiator in the tree is called a Cut Set. The shortest credible way through the tree from fault to initiating event is called a Minimal Cut Set.

Some industries use both Fault Trees and Event Trees. An Event Tree starts from an undesired initiator (loss of critical supply, component failure etc.) and follows possible further system events through to a series of final consequences. As each new event is considered, a new node on the tree is added with a split of probabilities of taking either branch. The probabilities of a range of 'top events' arising from the initial event can then be seen.

Classic programs include the Electric Power Research Institute's (EPRI) CAFTA software, which is used by many of the US nuclear power plants and by a majority of US and international aerospace manufacturers, and the Idaho National Laboratory's SAPHIRE, which is used by the U. S. Government to evaluate the safety and reliability of nuclear reactors, the Space Shuttle, and the International Space Station. Outside the US, the software RiskSpectrum is a popular tool for Fault Tree and Event Tree analysis and is licensed for use at almost half of the worlds nuclear power plants for Probabilistic Safety Assessment.

Analysis

Many different approaches can be used to model a FTA, but the most common and popular way can be summarized in a few steps. Remember that a fault tree is used to analyze a single fault event, and that one and only one event can be analyzed during a single fault tree. Even though the " fault" may vary dramatically, a FTA follows the same procedure for an event, be it a delay of 0. 25 msec for the generation of electrical power, or the random, unintended launch of an ICBM.

FTA analysis involves five steps:

Define the undesired event to study

Definition of the undesired event can be very hard to catch, although some of the events are very easy and obvious to observe. An engineer with a wide knowledge of the design of the system or a system analyst with an engineering background is the best person who can help define and number

the undesired events. Undesired events are used then to make the FTA, one event for one FTA; no two events will be used to make one FTA.

Obtain an understanding of the system

Once the undesired event is selected, all causes with probabilities of affecting the undesired event of 0 or more are studied and analyzed. Getting exact numbers for the probabilities leading to the event is usually impossible for the reason that it may be very costly and time consuming to do so. Computer software is used to study probabilities; this may lead to less costly system analysis.

System analysts can help with understanding the overall system. System designers have full knowledge of the system and this knowledge is very important for not missing any cause affecting the undesired event. For the selected event all causes are then numbered and sequenced in the order of occurrence and then are used for the next step which is drawing or constructing the fault tree.

Construct the fault tree

After selecting the undesired event and having analyzed the system so that we know all the causing effects (and if possible their probabilities) we can now construct the fault tree. Fault tree is based on AND and OR gates which define the major characteristics of the fault tree.

Evaluate the fault tree

After the fault tree has been assembled for a specific undesired event, it is evaluated and analyzed for any possible improvement or in other words study the risk management and find ways for system improvement. This step is as an introduction for the final step which will be to control the hazards identified. In short, in this step we identify all possible hazards affecting in a direct or indirect way the system.

Control the hazards identified

This step is very specific and differs largely from one system to another, but the main point will always be that after identifying the hazards all possible methods are pursued to decrease the probability of occurrence.

Comparison With Other Analytical Methods

FTA is a deductive, top-down method aimed at analyzing the effects of initiating faults and events on a complex system. This contrasts with Failure Mode and Effects Analysis (FMEA), which is an inductive, bottom-up analysis method aimed at analyzing the effects of single component or function failures on equipment or subysystems. FTA is very good at showing how resistant a system is to single or multiple initiating faults. It is not good at finding all possible initiating faults. FMEA is good at exhaustively cataloging initiating faults, and identifying their local effects. It is not good at examining multiple failures or their effects at a system level. FTA considers external events, FMEA does not. In civil aerospace the usual practice is to perform both FTA and FMEA, with a Failure Mode Effects Summary (FMES) as the interface between FMEA and FTA.

Alternatives to FTA include Dependence Diagram (DD), also known as Reliability Block Diagram (RBD) and Markov Analysis. A Dependence Diagram is equivalent to a Success Tree Analysis (STA), the logical inverse of an FTA, and depicts the system using paths instead of gates. DD and STA produce probability of success (i. e., avoiding a top event) rather than probability of a top event.

References

1. Ericson, Clifton (1999). " Fault Tree Analysis - A History" (pdf). Proceedings of the 17th International Systems Safety Conference. http://www. fault-tree. net/papers/ericson-fta-history. pdf. Retrieved 2010-01-17.
2. Rechard, Robert P. (1999). " Historical Relationship Between Performance Assessment for Radioactive Waste Disposal and Other Types of Risk Assessment in the United States" (pdf). Risk Analysis (Springer Netherlands) 19 (5): 763-807. doi: 10. 1023/A: 1007058325258. SAND99-1147J. http://www. osti. gov/bridge/servlets/purl/759847-JsFRIG/webviewable/. Retrieved 2010-01-22.
3. Winter, Mathias (1995). " Software Fault Tree Analysis of an Automated Control System Device Written in ADA" (pdf). Master's Thesis (Monterey, CA: Naval Postgraduate School). ADA303377. http://handle. dtic. mil/100. 2/ADA303377. Retrieved 2010-01-17.
4. Benner, Ludwig (1975). " Accident Theory and Accident Investigation". Proceedings of the Society of Air Safety Investigators Annual Seminar. http://www. iprr. org/papers/75iasiatheory. html. Retrieved 2010-01-17.

5. DeLong, Thomas (1970). " A Fault Tree Manual" (pdf). Master's Thesis (Texas A&M University). AD739001. http://www. dtic. mil/cgi-bin/GetTRDoc? AD= AD739001&Location= U2&doc= GetTRDoc. pdf. Retrieved 2010-03-09.
6. Eckberg, C. R. (1964). Fault Tree Analysis Program Plan. Seattle, WA: The Boeing Company. D2-30207-1. http://www. dtic. mil/srch/doc? collection= t3&id= AD0299561. Retrieved 2010-01-17.
7. Begley, T. F.; Cummings (1968). Fault Tree for Safety. RAC. ADD874448. http://www. dtic. mil/srch/doc? collection= t3&id= ADD874448. Retrieved 2010-01-17.
8. Hixenbaugh, A. F. (1968). Fault Tree for Safety. Seattle, WA: The Boeing Company. D6-53604. http://www. dtic. mil/srch/doc? collection= t3&id= AD0847015. Retrieved 2010-01-17.
9. Acharya, Sarbes; et. al. (1990) (pdf). Severe Accident Risks: An Assessment for Five U. S. Nuclear Power Plants. Wasthington, DC: U. S. Nuclear Regulatory Commission. NUREG-1150. http://www. nrc. gov/reading-rm/doc-collections/nuregs/staff/sr1150/v1/sr1150v1-intro-and-part-1. pdf. Retrieved 2010-01-17.
10. Vesely, W. E.; et. al. (1981) (pdf). Fault Tree Handbook. Nuclear Regulatory Commission. NUREG-0492. http://www. nrc. gov/reading-rm/doc-collections/nuregs/staff/sr0492/sr0492. pdf. Retrieved 2010-01-17.
11. Vesely, William; et. al. (2002) (pdf). Fault Tree Handbook with Aerospace Applications. National Aeronautics and Space Administration. http://www. hq. nasa. gov/office/codeq/doctree/fthb. pdf. Retrieved 2010-01-17.

12. " 7. 9 Fault Tree Analysis" (pdf). Electronic Reliability Design Handbook. B. U. S. Department of Defense. 1998. MIL-HDBK-338B. http://www. everyspec. com/MIL-HDBK/MIL-HDBK+(0300+-+0499)/download. php? spec= MIL-HDBK-338B. 015041. pdf. Retrieved 2010-01-17.

13. Fault Tree Analysis. Edition 2. 0. International Electrotechnical Commission. 2006. IEC61025. ISBN2-8318-8918-9.

14. Long, Allen (pdf), Beauty & the Beast - Use and Abuse of Fault Tree as a Tool, fault-tree. net, http://www. fault-tree. net/papers/long-beauty-and-beast. pdf, retrieved 16 January 2010.