12

Fault Tree Faults

Larry A. Minton

Robert W. Johnson

presented at the

International Conference on Hazard Identification and Risk Analysis, Human Factors and Human Reliability in Process Safety

January 15, 1992

Orlando, Florida USA

Battelle
. . . Putting Technology To Work

PROCESS RISK MANAGEMENT SERVICES
505 KING AVENUE, COLUMBUS, OHIO 43201 USA

Fault Tree Faults

Larry A. Minton and Robert W. Johnson
Battelle Process Risk Management Services
505 King Avenue, Columbus, Ohio 43201

Abstract

Fault tree analysis (FTA) is a widely recognized tool for the qualitative and quantitative analysis of process hazards. However, FTA is at least as commonly misused as it is applied and used correctly. This paper presents a litany of errors detected in previous fault tree analyses during the course of two consultants' process safety careers, with remedies given for each type of analysis error. Sound principles for constructing and evaluating fault trees are given.

Introduction

Fault tree analysis (FTA) is a powerful tool for analyzing system failures in complex systems. It uses top-down logic to model the significant causes and combinations of causes which would allow an undesired "top" event to occur. It is recognized as a hazard evaluation procedure of value to the chemical process industries, as illustrated by its inclusion as an acceptable method in the Occupational Safety and Health Administration (OSHA) regulations on process safety management, 29 CFR 1910.119.

Fault tree analysis is likely the most misused of the common hazard evaluation procedures. This may be due in part to its attractiveness to inexperienced analysts who become enamored with this rather glamorous tool of the trade. The Boolean algebra expressions which result from the construction of the fault tree model have drawn countless graduate students into the temptation to publish articles on mathematical manipulations and solutions to the model. The complexity of FTA models and quantitative analysis has generally necessitated the use of computerized algorithms, with another host of pitfalls associated with the computerization of the methodology.

This article will not go into detail on fault tree symbols and basic fault tree construction. These are adequately covered in other places, such as in References 1-5. Attention will be focused on common errors and pitfalls which have been observed in the use of fault trees to analyze process hazards. Misapplication of FTA will be addressed first, followed by faulty fault tree construction, faulty quantification, and misuse of FTA results. The topics covered in this paper can be used as a checklist for reviewing fault tree analyses.

Misapplication of Fault Tree Analysis

One of the most common problems with fault tree analysis is not that it is used incorrectly, but that it is used in applications where another analysis tool is much more appropriate. This problem usually comes about because the analyst is told to use FTA, so he proceeds to use it without questioning whether it is the best tool for the job. FTA is probably used too frequently in the process industries, rather than not frequently enough.

Fault Tree Analysis Won't Review the Whole Process. FTA only examines one undesired event at a time, such as a vessel rupture explosion. It only looks at those faults and conditions which can lead up to the specified undesired event. If there are other hazards in the system, such as fire or toxic release hazards, they would not be analyzed by the FTA studying the vessel rupture explosion.

Also, in some cases, the failure of a component may present multiple hazardous consequences. Only the consequence related to the top event is analyzed in the fault tree, with all other consequences being ignored. An example of this may be loss of flow to a part of the process. The consequence of interest is the loss of feed to the process; however, other consequences may exist such as toxic release or fire if the failure mode is line rupture. Other methods such as hazard and operability studies (HAZOPS), failure modes and effects analysis (FMEA), and "What If"/Checklist reviews should be used to identify and evaluate the hazards associated with an entire area or process.

A Complex Analysis Tool is Not Needed on Simple Systems. FTA is not very useful in situations where many single-event failures exist or where any one component failure causes the top event to occur. It is more useful in the analysis of complex, interrelated systems such as intricate control and safety systems which involve redundancy, elaborate logic, or multiple safety layers. FMEA is generally more appropriate to be used where single component failures can be analyzed one at a time. For systems where loss of containment is the major concern, such as a hazardous material release from a piping system, the Loss-of-Containment Checklist (References 5 and 6) has been found to be very useful in identifying containment loss causes. Systems having few initiating events of concern (such as loss of cooling to an exothermic reactor) but many possible outcomes may be better studied using event tree analysis.

Bad Root, Bad Tree. One of the most common FTA faults is a poorly specified top event. Since the top event essentially defines the scope for the entire analysis, it is imperative that the top event be carefully chosen and precisely worded. Some common top event problems are:

- Top event is too general: "Compressor system doesn't work" is too broad and nonspecific to be able to be analyzed effectively.

- Top event is a process deviation and not an accidental event: "Pressure Too High In Reactor" expresses a possibly reversible deviation from normal operating conditions, not an irreversible accidental event. A better top event is "Reactor Rupture Due to Overpressurization." The top event should be the proper subject for a consequence analysis, unless being done for a more limited purpose such as to study the reliability of an individual safety system.

- Top event is a consequence severity and not an accidental event: "Multiple Serious Injuries" or "Major Business Interruption Loss" are statements of consequences. Specific accidental events such as fires, explosions, and hazardous material releases usually make much better subjects for fault tree analyses.

Some further examples of top events are shown in Figure 1. Of course, whether or not a top event is correct for a given study will vary depending on the specific objectives of the study.

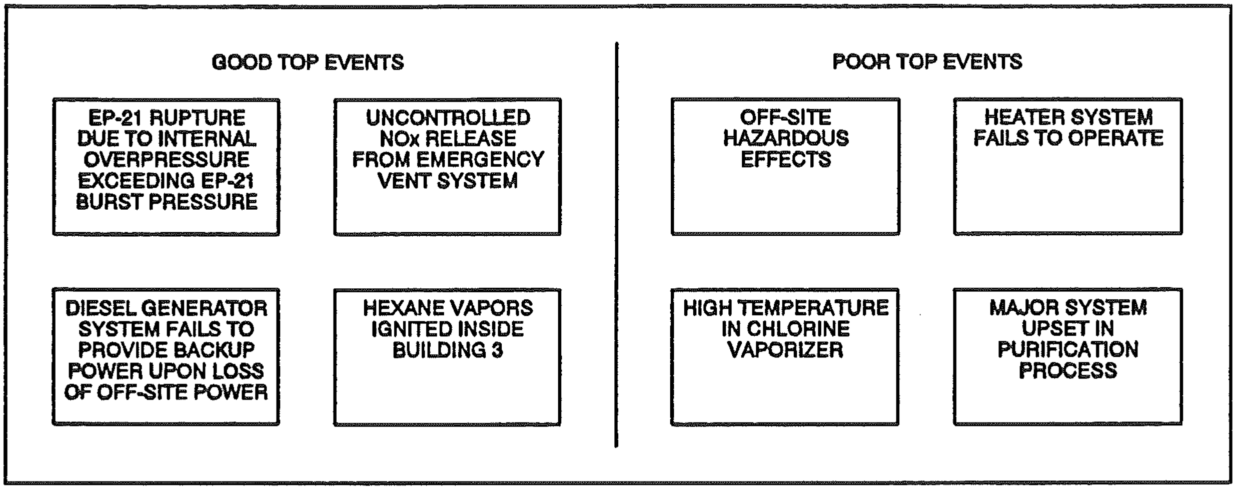

FIGURE 1. TOP EVENT EXAMPLES

More Than Paper is Needed to Build Fault Trees. Due to the detailed nature of FTA, complete and accurate design information is required as basic data for a FTA study. Interdependencies are modeled by the FTA; if these interdependencies are not yet defined, the fault tree model will be of limited use. For this reason, FTA is not appropriate to be used at a preliminary design stage of a project, but rather when both process and instrumentation details are known. Use of FTA on an existing process is even more desirable because of the possibility of having historical failure data available. Operator interactions with the system are also more apparent for operating units than for new designs.

Construction Faults

Problems encountered in fault tree construction generally can be traced to not following some basic rules of fault tree construction: specify the immediate necessary and sufficient causes for each event, determine the correct AND and OR logic to connect the causes, and develop each branch only as far as is needed to model interdependencies and determine event frequencies and probabilities.

Back In Time. Many fault trees combine events together at a single AND or OR gate which, in real life, would occur at different times in an accident event sequence. To avoid this, so as to be less likely to miss important levels of protection which may be built into a system, the fault tree should logically trace the top undesired event backwards in time, past each barrier or safety system, to the initiating events or boundary of the analysis resolution. This can be illustrated by reviewing the general sequence of steps in most process accidents, as presented in Figure 2. An effective FTA starts at an <u>accidental event</u>, works backwards through each <u>protective system</u> encountered in getting to the accident event, and ending at the <u>initiating events</u>. Thus, emergency relief systems should show up near the top of a fault tree studying a bursting vessel explosion event, since they are last-resort protection systems. Lumping of safety systems together at a single AND gate (such as level control failure, operator response to a high level alarm ten minutes after level control failure, automatic shutdown at a high-high level point thirty minutes after level control failure, and mitigation efforts upon detection of organics in the process sewer forty-five minutes after level control failure) does not allow the tracing of the logic back through the accident scenarios.

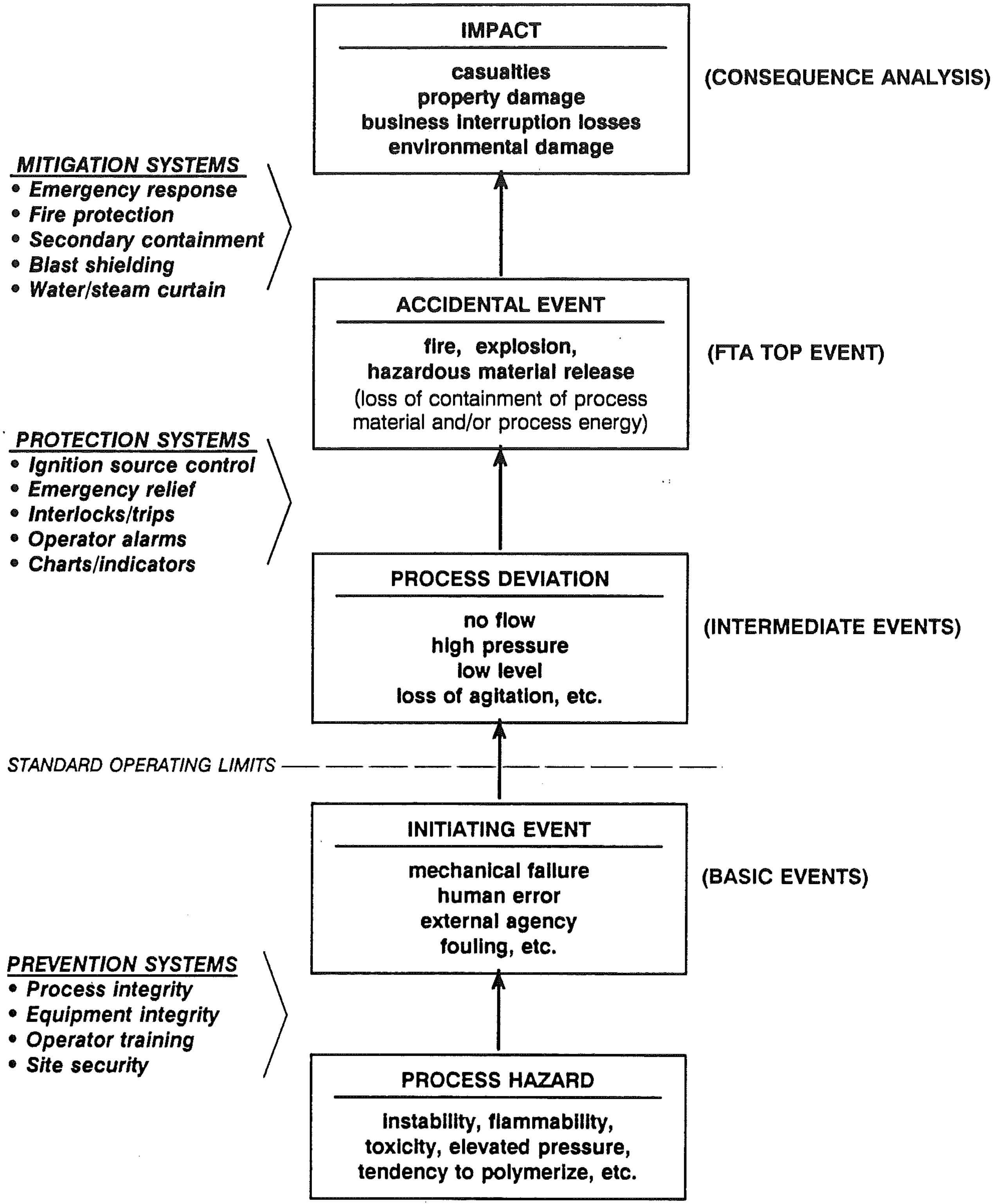

FIGURE 2. ACCIDENT SEQUENCE FROM INITIATING EVENT TO ACCIDENTAL EVENT

Discrimination between Initiators and Enablers. A very helpful distinction in the logical and consistent development of a fault tree model is to separate initiating events from enabling events. Confusing initiators and enablers usually leads to other fault tree construction and calculation problems such as combining frequencies at AND gates.

Initiators are the initiating events of Figure 2 which are the starting point for accident scenarios. Components whose failure or improper action can *initiate* an accident scenario, such as control valves and operator actions, generally control, modify, or influence a process. An initiating event cannot remain uncorrected indefinitely; the situation must be corrected to avoid serious consequences. For example, an incinerator cannot continue to operate normally once combustion air is lost. Either the incinerator will be shut down safely or a process incident will result. Since the duration of the failure is not important for initiating events, the failure frequency is the important parameter for an initiating event.

Enablers, on the other hand, are system conditions or levels of protection which allow, or enable, an accident scenario to proceed. An enabler condition, such as a relief valve having a plugged inlet line, can remain in a failed state for an extended period of time while the process itself operates normally. The enabler condition will only be discovered when the component itself is functionally tested, such as an annual relief valve inspection, or when it is called upon to protect against a process deviation. Failed alarm loops and interlock/trip systems are typical enabler events. Enabler components generally monitor process streams, provide protection against system deviations, or are conditions which must be true for an initiated fault to lead to the top accidental event (such as an operator not being in the control room at the time a critical alarm sounds). Since enabler conditions continue to exist until they are detected and corrected, the failure frequency and duration are both important. What is important to the FTA is the dimensionless *probability* of the component failing to function adequately when "tested" by an initiating event. This probability can be either determined directly or can be calculated by combining the frequency and duration of the enabler condition.

Lost Intermediate Event Levels. All too often, gates are assigned five to ten, or even more, inputs. In determining that all of these inputs were to be assigned to the gate, the analyst had to mentally build a model of the logic under the gate. This information is lost when the logic is collapsed to a single gate. This causes problems in reviewing the model and in ensuring that all failure modes were considered and that the logic of the mental model was correct. The flow of information through the fault tree should be kept consistent, logically building up from the basic events to the top event, even if gates with single inputs are created.

Gate Switching. Probably the most common error in fault tree construction is inadvertently using an AND gate instead of an OR gate or vice versa. The gate type must be checked and re-checked. When hand-drawing fault trees, putting in the gate type should not be deferred until later. Deriving the cutsets for portions of the model will help identify these logic errors.

Logical Bombs. Some aspects of FTA logic model development lend themselves particularly well to logic errors. Among them are:

- Reverse flow paths are easily overlooked. Does the model consider flow diversion through a redundant pump if that pump fails? If multiple suction sources for a pump exist, can a suction valve failing open divert flow from the pump suction? For every line connection, the possibility of flow in each direction must be investigated.

- Some components may need to be included in the model more than once, with different failure modes. For example: a check valve can fail closed, which prevents forward flow, or it can fail open, which allows reverse flow.

- The use of NOT gates should be avoided if at all possible. If they must be used, do not blindly trust that the cutsets generated from the fault tree are correct. Most fault tree analysis programs do not handle NOT gates correctly.

Common Mode Failures are Commonly Missed. At the system level, common mode failures are frequently the dominant contributors. Common mode failures can include external events, support and utility systems, test and maintenance errors, and operator errors. For example, if a reactor has multiple protection systems, but all those systems require plant air to operate, the plant air system failure rate may dominate the top event failure rate.

Omission of Operator Errors. The analyst must identify possible errors of omission and commission by the operator. Include operator errors which could occur during normal operations, test and maintenance operations, and during accident conditions. Ask: "Is the operator suppose to perform any actions? Are the indications to the operator similar to another type of incident? Could the operator react in an inappropriate manner to these indications?"

Too Much Detail in the Model. Some classic fault trees have been created by overzealous analysts who have developed one branch in tremendous detail only to have it be combined at an OR gate with a basic event which is much more likely to occur than the entire overgrown branch. Systems need only be modeled down to the level at which data are available for the basic event frequencies and probabilities, and only down as far as the results are relevant to the intent of the analysis. The diamond symbol can be used very effectively in this regard, indicating systems which are not down to the basic event level but which need not be developed further. An obvious example, to make the point, is not generally needing to trace "loss of cooling water supply" all the way back to the river intake. For an operating unit, the operating personnel usually can give a fairly accurate estimate of how often cooling water is lost to the process unit without going into details of, e.g., how reliable the motors are for the cooling water supply pumps.

Too Little Detail in the Model. Although a less common fault than putting too much detail in a fault tree model, putting too little detail into the model is only warranted for a rough first-cut analysis. An example of not developing the tree enough would be the lumping of all of a complex system's failures, such as all of a compressor's failure modes, together into a single basic event on the fault tree. Components need to be modeled to the level where any failures below that level will result in the same failure state for the component. Also, a specific failure mode such as a coupling failure or a false high output from a level transmitter should always be documented for each basic event and intermediate event in the fault tree model. Failure to do so leads to confusion in the fault tree logic and the use of overly conservative failure rates (e.g., the frequency of all valve failures when only the fail-closed mode is relevant).

Full Names for All Intermediate Events. Fault trees are often seen with no intermediate event names or descriptions, only the logic gate. This practice makes reviewing the fault tree logic very difficult. A gate name of "G18" does not provide any information. A gate name of "NOFLO-F-PMP12A1" is better in that it tells what component is being modeled and what is the failure mode of concern.

Consistency in Event Naming. Be consistent with both the names and descriptions given to basic and intermediate events. An event name should tell the portion of the system and the type of fault condition being modeled. The event description should be such that if the inputs to the event are true, the description is true. For example, a gate name of "NOFLO-F-PMP12A1" tells the analyst that the event being modeled is no flow from pump 12A1.

Sufficiency in Fault Tree Documentation. A fault tree should be as self-documented as possible. One should be able to pull out any branch of the model and understand what is being modeled in that branch. The analyst should try to keep the failure logic associated with a given component or subsystem together as much as possible.

Start-up and Shutdown Sequences. Most fault trees are constructed as if the system never starts up or shuts down. Historically, most process incidents have occurred during, or shortly after, such transient operating modes. It is essential that the startup/shutdown modes be handled separately from the continuous operation of a process. For the continuous operation, the initiating events are usually active operator error, mechanical failure of components, or external agencies. For noncontinuous operations, the frequency of the operation becomes the initiating event.

Define Success. No matter how well a fault tree model is constructed, if the success criteria defined is incorrect, the analysis results will be incorrect. Components and/or systems determined to be important with one set of success criteria can be relatively unimportant with another set. An example of this is knowing whether one or two vent tanks are sufficient to provide overpressure protection. If both tanks are needed, there will be many single-event failures associated with the tanks. If only a single tank is needed, many of these single-event failures will become double-event failures. This will drastically affect the top event frequency. The analyst should take the time to make the calculations necessary to clearly define the success criteria. The selection of improper success criteria is usually easily correctable by minor changes in the fault tree logic.

Computer-Generated Forests. Although the idea of inputting the piping, instrumentation, and control logic of a process into a computer program and having the program generate a fault tree is certainly an attractive concept, attempts at doing so have generated nearly all of the construction faults listed above. In particular:

- Computer-generated fault trees do not have meaningful, if any, intermediate events. This makes checking of the fault tree logic nearly impossible, and it is also not possible to use the fault tree itself as a presentation of results to the decisionmaker.

- Computer-generated fault trees tend to have abbreviated event names without word descriptions. Checking the cutsets for accuracy is thus much more difficult, albeit not impossible.

- FTA is an <u>analysis</u> tool, where the analyst applies his knowledge of systems safety to the process at hand. Having a computer generate the fault tree bypasses an important part of this analysis function. As a result, faults which are unique to the system or subtle in their effects on the system (such as the gradual fouling of a heat exchanger or a slow calibration drift on a process measurement signal) are easily missed. Operator interactions with the system are usually not as fully developed in computer-generated fault trees as in manually generated trees.

- With existing techniques, computer generation of fault trees is not known to be capable of handling reverse flow situations, or any situation where bidirectional flow through a component is possible.

Although the above discussion may appear to indicate a bias on the part of the authors, it can be noted that both of the authors use computerized analysis tools enthusiastically where they are truly appropriate.

Fault Tree Quantification

FTA can be used as only a *qualitative* analysis tool. Its benefits as a qualitative tool are (a) in showing the dependency between component failures, (b) in forcing a deeper understanding of the system, and (c) in finding single-event failures which could lead to the top event. However, the real power of FTA is in its ability to *quantify* the likelihood of the top event by combining the frequencies and probabilities of more basic events and failures. A quantitative analysis can also show which components are more important than other components with respect to the top event of concern. In this regard, quantitative risk analysts like to cite Lord Kelvin's quote, "...when you can measure what you are speaking about, and express it in numbers, you know something about it," as in Reference 5. However, proper quantitative analysis of a fault tree is difficult, as the following illustrate.

Testing and Maintenance Take Time. A component that is out of service one hour per year for test and/or maintenance has an unavailability of at least 1E-04. There is also the probability of failure to restore the component correctly after testing, failure of maintenance to correct identified problems, or maintenance causing new problems. For example, calibration of a sensor may require isolating it from the system. After testing, the operator may forget to reopen the sensor sample line.

Frequency Times Frequency Equals a Nonconservatively Wrong Answer. One of the most common fault tree faults is the confusion of frequencies (e.g., events per year) and probabilities (dimensionless). When two frequencies are combined improperly at an AND gate (i.e., without taking the durations of each event into account), the result is $(\text{events/year})^2$, which is obviously wrong. The reason why such obvious errors get left on a fault tree is that the units are not always put on the frequency-related events. When quantifying a fault tree, the units, if any, should always be included on each and every number. This also helps avoid the confusion of different units of time for event frequencies, such as failures per year and failures per hour. A consistent unit of time should be used throughout the entire analysis if possible.

Imaginary Numbers. Probabilities greater than one are obviously wrong, but nevertheless show up frequently enough to warrant highlighting. Two ways of obtaining probabilities greater than one are easily achievable in FTAs. First, particularly when FTA calculations are done manually, the simplifying assumption of Equation 1 is often used as the means by which the unavailability of a component is calculated:

$$\textit{Unavailability} \approx \lambda \cdot \tau \qquad (1)$$

where λ is the average failure frequency (e.g., failures/year) and τ is the average time to detect and correct the failure (e.g., years). As long as $\lambda \cdot \tau$ is less than 0.1, no significant accuracy is lost by using this approximation. However, if τ is large, such as if a component is never regularly tested, then the product $\lambda \cdot \tau$ could actually be greater than one. The exact equation for unavailability with a constant failure rate is actually

$$Unavailability = 1 - e^{-(\lambda \cdot \tau)}. \tag{2}$$

Although the temptation is just to use this equation and go on with the analysis, the obtaining of an unavailability near one should automatically raise a red flag to indicate that something is wrong with the reliability of the component in question. Either it should be tested more frequently or it should perhaps be removed from service so as not to give a false sense of security concerning the presence of the component in question.

Second, adding more than one near-unity unavailability and/or probability of failure on demand at an OR gate can result in a combined probability greater than one if they are directly added together, as illustrated in Figure 2. This is caught fairly readily with manual FTA calculations, and the gate can just be assigned a value of unity with no significant error. (For the mathematically correct answer, the cross product is subtracted.) However, many computer algorithms solve the FTA model for minimal cutsets before calculating cutset frequencies. This approach does not flag OR gates with probabilities greater than one, since the calculations are not done from gate to gate. Usually, the cumulative error introduced by this problem is relatively small.

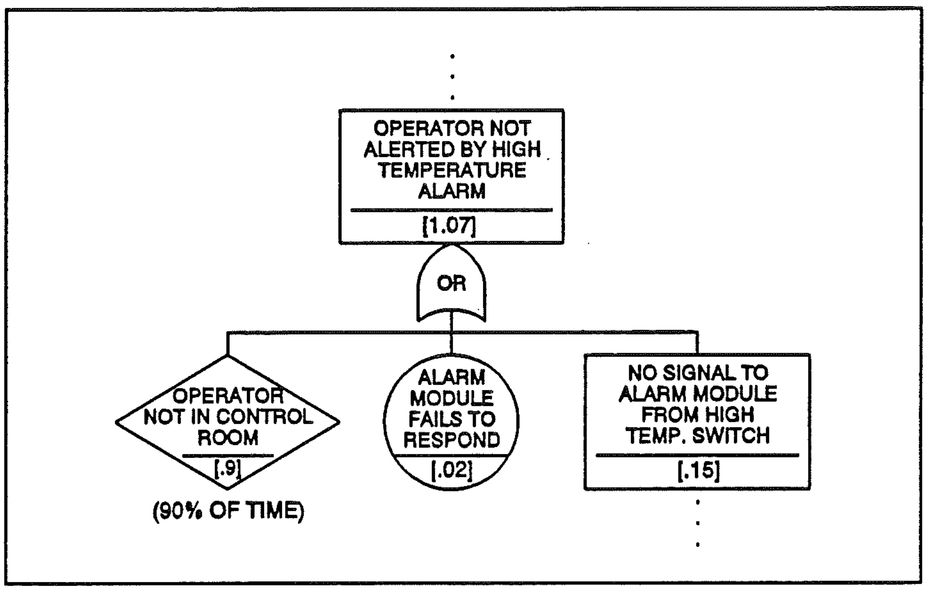

FIGURE 2. CALCULATED PROBABILITY GREATER THAN ONE

Older Than the Universe. Assuming the age of the universe is 13 to 20 billion years, an event that has occurred once in the history of the universe has an average frequency of 5E-11/yr. Any event frequency less than this should certainly be suspect. If the universe was created with the appearance of age about 6,000 years ago, as per Bishop Usher's calculations based on Biblical chronology with no gaps, then this changes to 2E-04/yr. In any case, many significant technology changes, natural disasters, wars and such have assuredly occurred in the last 6,000 years. FTA events with very low predicted frequencies should be carefully reviewed to determine whether they make sense, whether any significant causes have been omitted, and whether any calculational errors exist.

Quantitative Analysis Does Have Its Limits. The quantification of fault trees typically assumes that all component failures occur randomly and their failure rates are independent of the failure rates of other components. Testing and maintenance are the most blatant causes of basic events not being independent. For example, component failure rates typically include unavailability due to maintenance. In a cutset containing redundant components, the quantification assumes the events are independent. Quantification-wise, there is a probability that both components are in maintenance at the same time. This will result in an overly conservative failure frequency for that cutset. Test schedules and preventive maintenance schedules will affect the degree of independence of component unavailabilities.

Misuse of Fault Tree Analysis Results

The primary reason for performing a FTA, besides finding weaknesses in a complex system, is to aid the decisionmaking of those responsible for managing process risks. Thus, it follows that the accurate *presentation* of the FTA and its results are as important as the accuracy of the FTA model itself. The following are some common faults observed with respect to the presentation of FTA results.

Can't See the Forest. The gory details of the fault tree modeling effort should not get in the way of the big picture. Often, the effort involved in just completing the fault tree model and its quantification take all the analyst's time and energy (and budget), and time is not taken to step back and truly analyze the significance of the results.

A Fault Tree is Never Complete. Any fault tree model should not be presented as being totally "complete." For example, another external event could always be added to any model. This is illustrated by an actual incident in which a toilet overflowed on the level above a control room, shorting out much of the electrical equipment in the control room. All kinds of external events are usually not covered in fault tree models, such as fires, floods, earthquakes, lightning, high winds, icing, explosions in adjacent processes, and sabotage. Thus, presentation of FTA results must clearly communicate what the objectives and boundaries of the FTA were, in order to not give a false impression of absolute completeness.

Missing Assumptions Mislead. The making of a multitude of assumptions is inevitable in a FTA of any magnitude, due to incomplete design details at the time of the analysis, limitations placed on the resolution of the analysis, etc. It is important that the decisionmaker who is using the results of the FTA know what the assumptions are upon which the FTA is based. The assumptions cannot generally be all listed on the FTA model itself; it may be necessary to include a list of assumptions in the FTA report, with adequate cross-referencing to be able to tie the assumptions to the events to which they pertain. Also, verification of the assumptions should be an ongoing risk management task. If any assumptions are determined at a later date not to be accurate, the FTA should be revisited to check the effect on the study conclusions.

Vague Uncertainties. Closely related to the analysis assumptions are the uncertainties associated with a quantified fault tree's failure rates and probabilities. The quantitative results of a FTA are generally expressed as a single point estimate of the top event frequency, say one chance in 1000 per year of operation. However, this does not tell the decisionmaker anything about the frequency's degree of uncertainty. Many component failure rates actually have orders of magnitude uncertainty ranges, based on the type of service, maintenance, process conditions, age, etc. The better quantification codes allow the analyst to specify the uncertainty ranges on the individual failure rates.

The codes then calculate the uncertainty range for the top event frequency. This is generally done using Monte Carlo methods, but it is possible to combine uncertainties in cutsets once a distribution is assumed for the failure rate uncertainties. Determining uncertainty ranges and propagating them through a FTA obviously involves a great deal of effort. More commonly, sensitivity studies are conducted on the FTA model to determine what data values are most important to the accuracy of top event frequency.

Express Your Precision Accurately. Presenting quantitative results of a FTA with three-significant-figures precision gives a misleading impression of the true precision of the results. FTA results are, at best, good to within a factor of two to three, depending on the input data available and the completeness of the model. Many studies are only good for order-of-magnitude top event frequency estimates. Quantitative FTA results should only be expressed with a precision comparable to the precision of the analysis and its input data, and in very few cases would this be with more than one significant figure.

Leave Many Bread Crumbs. The most common criticism expressed when FTAs are independently reviewed is that the FTAs are not traceable from the assumptions and data sources to the fault tree results. Documentation of a FTA such that it is easy to follow is difficult and time-consuming, but it must be done for the FTA to be of maximum benefit. This traceability can be accomplished by providing a fault tree index, descriptions of the failure events in a list of the most important cutsets, documentation of all data values used and their sources, and a means of tying the analysis assumptions into the analysis itself. A good practice is to judge whether a person not involved with the FTA could (a) determine how the FTA results were calculated from the fault tree model and (b) trace any part of the FTA back to the basic data and assumptions with a reasonable amount of effort.

Conclusion

Construction and evaluation of a logically correct fault tree is not a simple task. Many companies using FTA require the fault tree to be constructed and quantitatively analyzed, or at least reviewed, by an experienced fault tree analyst. This article may give the reader some indication of why this is the case.

On the other hand, FTA doesn't have to be a black art. Most fault tree faults can be avoided if these key rules are followed:

- Only use FTA where it is the best tool for the job
- Take time to define the FTA scope and select the right top event
- Construct the fault tree by working backwards in time from an accidental event through the safety layers which must be breached to get to the top event
- Keep a clear distinction between frequency branches and probability branches
- Stick to basic AND and OR logic
- Take the effort to construct a tree with fully explained intermediate events

- Only develop each fault tree branch far enough down to get to where historical failure rate data are available
- Check both the fault tree logic and the cutsets for accuracy
- Document all important assumptions and uncertainties
- Be careful how the FTA results are expressed and communicated.

Following these basic, tried-and-true rules will result in more fault-free trees and more valuable analyses.

References

1. W. E. Veseley et al., *Fault Tree Handbook,* NUREG-0492, January 1981.
2. E. J. Henley and H. Kumamoto, *Reliability Engineering and Risk Assessment,* Englewood Cliffs, New Jersey: Prentice-Hall, 1981.
3. F. P. Lees, *Loss Prevention in the Process Industries,* London: Butterworths, 1981.
4. *Guidelines for Hazard Evaluation Procedures,* New York: American Institute of Chemical Engineers-Center for Chemical Process Safety, 1985.
5. *Guidelines for Chemical Process Quantitative Risk Analysis,* New York: American Institute of Chemical Engineers-Center for Chemical Process Safety, 1989.
6. *Guidelines for Vapor Release Mitigation,* Appendix A, New York: American Institute of Chemical Engineers-Center for Chemical Process Safety, 1989.