"What's Wrong with the Fault-Tree Linking Approach for Complex PRA Models?"

Executive Summary

Analysts using complex probabilistic risk assessment (PRA) models at nuclear power plants have two approaches from which to choose: Fault Tree Linking (FTL) or Large Event Tree Linking (ETL) methods. The choice of which method is utilized depends primarily on history (which method was used by the analysts who first developed the specific PRA model) or economics (e.g., which method is used by the majority of PRAs in a utility merger). However, users of either method should be aware of the problems and limitations of their approach. In particular, the current generation of PRA analysts and users need to address problems which a previous generation did not have to face due to the growing complexity of current models and the extensive applications to which models are routinely being applied.

The authors of this article clearly have a bias toward, as well as extensive experience with, the ETL method. A number of tools have been built into the RISKMAN® code which implements the ETL method to solve problems that still remain in FTL. This paper discusses a number of key problems that illustrate, "What's Wrong with the FTL Approach?" This paper summarizes these problems, provides an estimate of the significance of the problem, and provides references where some of these problems have been examined in greater detail. Note that for some of these problems, the quantitative level of significance can be estimated, while for others, it is unknowable with the current approach.

Since these problems may impact important risk-informed decisions, the PRA analyst is obligated to examine these problems and identify the extent to which they restrict his or her model.

The Problems with FTL

The "Fault Tree Linking" (FTL) approach, also called "large fault tree-small event tree" approach, is a technique to model and quantify accident sequences often used for the risk and safety assessment of nuclear power plants. In this approach, fault tree models for all systems of interest in a single sequence, or family of sequences, are logically linked for quantification. Frequency truncation is applied to the linked fault trees during Boolean reduction to minimal cutsets. Recovery rules are then often applied to each minimal cutset retained after applying truncation, and the resulting logical combinations then quantified and totaled to obtain the core damage frequency. A similar linked fault tree model is typically developed, recovery rules applied, and then quantified for the large early release frequency.

The FTL approach is subject to some limitations and approximations. Many of these have been investigated in references 1, 2 and 3. These and other issues are identified below.

Event Tree/Fault Tree Construction Issues:

Sequence Representation/ Development: The sequences portrayed in the small event trees used in the FTL approach are defined by a small set of safety functions identified as necessary to prevent core damage or to maintain containment integrity; e.g. reactivity control, high pressure injection, high pressure recirculation, containment isolation. A combination of these safety function states (i.e. success and failure paths of the safety functions) defines one sequence in the small event trees. The combination of safety functions that must all be performed to prevent core damage and to maintain containment integrity may change as a function of the initiating event selected and on earlier events in the specific path through the small event trees. Other approaches to sequence representation and development may also be used. Most practitioners find it useful to add more event tree top events, in addition to the events tracking performance of the safety functions, to provide more clarity and facilitate quantification.

The small event trees used, however, are still too small to list even a small proportion of the symptom based events found in nuclear plant emergency operating procedures. The reviews of these small event tree sequence representations by plant operators, who are most familiar with these procedures, are therefore hindered. Often key accident sequence-related events (e.g. pressurizer relief) are only modeled within the linked fault trees; i.e. a logical step below the event tree top event logic. The temporal representation of sequences is lost when such key events are only modeled within the fault trees.

The small event tree representations of accident sequences generally (except for station blackout models) omit entirely the role of support systems. Again, the support system dependencies are modeled a logical step lower, within the fault trees. Since many emergency operating procedures are keyed to the response of support systems during an accident, this omission makes it impractical to review event tree sequences against the plant emergency operating procedures. The limited usefulness of FTL style event trees is evident by the willingness of FTL analysts to discard the event trees entirely after FTL model construction in favor of a single fault tree top event representing the core damage frequency; i.e. individual sequence frequencies are usually not computed.

Sequence Detail: To simplify fault tree linking models, support system failures may be suppressed for initiators believed *a priori* not to be important. For example, losses of offsite power may only be modeled as the initiating event. Losses of offsite power following other initiators may be neglected. The small event tree sizes make it difficult to sort out recovery from support system failures, and this problem becomes even more acute if multiple support system failures, represented only within the fault tree logic, occur.

Flag Settings: An important aspect of model development using the FTL approach is the setting of model flags. Conceptually, the fault tree linking approach simply involves linking those fault trees associated with each top event along a single event tree path. In truth, other model logic adjustments must be made. Logical flags must be set for each sequence fault tree as a function of the sequence initiator and other conditions in the sequence that determine the specific fault tree success criteria. Logic flag settings may be used to eliminate mutually exclusive events, to incorporate recovery actions, and to account for other sequence specific success options not already included in the baseline fault trees. A review of the fault trees is not complete without also reviewing the settings for such flags for every sequence in which the system fault tree is used. Development and documentation of such flag settings can be difficult and requires a thorough understanding of the model structure and assumptions. Generally, once developed, users strive not to change such settings in model updates for fear of introducing errors.

Sometimes flags with values of 1.0 are used to clarify the path to core damage in the sequence cutsets. This can lead to repeated, otherwise “minimal cutsets”; e.g. if the same basic event failure combination triggers different 1.0 flags, these combinations show up as different cutsets. Event values of 1.0 for such flags are not sufficient to avoid incomplete logical reduction.

Consider a fault tree as identified below:

TOP=G1*C
G1= A + B

The minimal cutsets are {A*C and B*C}. If both events are set to true then the resulting minimal cutset in the modified fault tree is just {C}.

If events are instead set to 1.0, the same minimal cutsets apply but the numerical evaluation is then
P(TOP) =1*P(C) +1*P(C) = 2P(C)

The problem lies in how the cutset probabilities are totaled. The rare event approach assumes basic event probabilities are less than 0.1. The min-cut upper bound approach assumes the cutsets do not share any basic events and therefore are independent.
In this example involving flags, neither is true. Even worse, in this example, A and B always occur together; i.e. they are completely dependent. These approximate methods of totaling do not work for this example and other cases where events are set to 1.0.

Formatted: Line spacing: At least 12 pt, Don't adjust space between Latin and Asian text, Don't adjust space between Asian text and numbers

Formatted: Font: (Default) Helv, 10 pt, Font color: Black

Fault Tree Size: Modern software can easily manipulate and reduce fault trees that can be developed by users. However, reviews of such large fault trees are hampered by the sheer size of the fault trees. Some nuclear plant models may involve 50 to 130 levels of gate logic between the top event (i.e.core damage) and the lowest basic events contributing to the top event (References 1 and 3). Such large fault trees cannot be effectively reviewed by humans. Analysts instead focus their reviews on the minimal cutsets obtained by Boolean reduction during which frequency truncation is applied. This cutset review process cannot easily reveal logic errors or optimistic data assignments resulting in cutset omissions. Experience shows that undiscovered modeling errors may only be revealed, if at all, when equipment out of service evaluations change the relative truncation frequencies, resulting in new lists of minimal cutsets that have not yet been reviewed.

Model Maintenance: It is frequently stated that FTL permits users to simply add basic events and then push the button to requantify the updated model. When performed correctly, and after adjusting the recovery models accordingly, this is true. However, often the fault tree logic is so complicated that analysts not involved in its creation choose to replicate portions of the logic before making the change and only use the replicated logic in the revised model. By making these logic replications, often the same logic is repeated many times in the model with only slight variants. This makes subsequent model changes still more difficult by multiplying the number of places future changes must be made.

Initiator Models: Redundant, normally operating systems (e.g. service water, component cooling water system) require detailed fault trees to evaluate their relatively low initiating event frequencies. Standard fault trees for such initiators can be difficult to construct in ways that accurately account for the true mission times of each component; i.e. accounting for repair of the redundant trains. Also, in FTL models, the top event probabilities so derived are generally represented as single events in the linked fault trees for core damage frequency. This approach certainly simplifies the large linked fault tree models but prevents an accurate assessment of the related basic event importance measures; i.e. the basic event contributions via the system initiator are all rolled together into the one event representing the system initiator. This is unfortunate because the contribution of basic events represented in the system initiator models often is substantially greater than that contributed by the same basic event in response of the system to other initiators.

Boolean Reduction Issues:

NOT Logic: The presence of NOT logic greatly adds to the time and complexities to Boolean reduce the linked fault trees, and is therefore avoided in most complex models. Fault trees that involve NOT logic (i.e. / in the logic equation below) require additional methods for successful Boolean reduction and is generally not possible in standard fault tree analysis. For example, standard fault tree reduction techniques cannot fully Boolean reduce logic such as:

$$(/X)Y + X(/Z) +Y(/Z) = (/X)Y + X(/Z)$$

In practice, the use of NOT logic is avoided, or should be, to the greatest extent practical.

The significance of not applying Boolean reduction rules such as the above for complex, real models employing NOT logic has never been estimated.

Success Terms: Linked fault tree models are developed from functional level, small event trees. The sequences that are mapped to core damage generally contain some success events along the sequence paths. Explicit accounting of these success terms using NOT logic is seldom performed. The success terms can, however, be accounted for on an individual sequence basis by linking only the failed events and using the “delete term” approach; i.e. deleting the sequence minimal cutsets that contain combinations of basic events that otherwise would fail the successful top events along the same sequence. This approach works well only if none of the associated fault trees contain NOT logic.

The delete term approach, however, is also often not used in practice; for example, for risk monitoring models. Instead a single fault tree is constructed for the end state “core damage” by Boolean linking all FTL sequences mapped to core damage. In this linking process, logic for the successful top events is ignored. This approach tries to take advantage of the observation that the union of all core damage sequences may allow some Boolean reduction. For example, if the initiating event is denoted as IE1, and events in the functional event tree are called A, B, C, and D, then two core damage sequences might be represented as:

IE1*A*/B*C,
IE1*A*B.

where /B represents success of B

These two sequences may be logically reduced to

IE1*A*B + IE1*A*C;

The reduced representation then does not require that any successful events be included. However, this example is just a special case. There are other examples where such reduction cannot be achieved. For example, the sequences below cannot be similarly reduced.

IE1*A*/B*C
IE1*A*B*D

Moreover, the small fault trees (A, B, C, and D) may contain sequence specific flag settings so that the fault trees, though they have the same top event name, are not identical anyway. Failing to consider the success terms in the sequence quantification may introduce significant errors in the individual accident sequence frequency. The significance of omitting success terms when computing the overall core damage frequency via a single top event is not established for complex models.

Frequency Truncation: In order to simplify the logic of the LFT models, frequency truncation is applied during minimal cutset reduction. The algorithms used for frequency truncation make comparisons of partial sequence cutset probabilities (i.e. as the fault tree logic is expanded) to the truncation limit to decide whether further fault tree expansion is required or can be neglected. Tests on the accuracy of a real model evaluated from minimal cutsets has shown that individual sequence frequencies may indeed be optimistic; i.e. low by greater than 10% in as many as 25% of the sequences (Reference

1). Usually the software does not keep track of the partial sequence cutset frequency truncated. Attempts to do so have shown that the resulting frequency bounds are too high to be of any use for demonstrating convergence of the quantified result. Even if the total frequency truncated could be summed, it would not be meaningful because many of the partial sequence cutsets truncated are not minimal with respect to core damage or large early release. This problem of trying to estimate the amount truncated and hence the potential for CDF convergence is exacerbated as modeling detail increases. Since risk significant basic events may be defined as those which have a Fussell-Vesely importance greater than just .005 (references 5 and 6), concerns about convergence when using frequency truncation become even more problematic for importance calculations. Clearly the ASME standard requirements for frequency truncation, (i.e. supporting requirement QU-B3) of ≤5% change in CDF for a single decrease in frequency truncation by one order of magnitude (reference 4), is inadequate for demonstrating the convergence of importance measures used in component risk rankings.

Quantification Issues

Recovery Rules: Model approximations are introduced for a variety of reasons, among them the need to avoid the use of NOT logic. Recovery rules are often applied to minimal cutsets retained after Boolean logic reduction in attempts to eliminate these approximations. For example, not all plant accident sequence logic is accounted for in the linked fault trees; e.g. to remove illogical or plant technical specification prevented system alignment combinations. The recovery rules are applied by pattern recognition routines. By this technique, the original minimal cutset frequencies are then either revised (upward or downward) and in some cases the minimal cutsets are deleted entirely. There is no mathematical basis for this approach to cutset recovery; i.e. there is no assurance that the minimal cutsets after recovery are complete or are indeed still minimal. At the very least users must insure that the recovery rules so applied are independent of basic events that do not appear in the revised cutsets.

As an obvious example, consider an initial minimal cutset that involves only one operator action failure event:

IE1*A*HE1,

Its sequence cutset frequency might be revised downward by a second recovery action (HE2) deemed independent of the first action:

IE1*A*HE1*HE2

However, a third action (HE3) that appears in the model but not in the minimal cutset may prevent the recovery action (HE2) if it too was failed; i.e. for the non-minimal cutset

IE1*A*HE1*HE3.

Since the logical dependence between HE1, HE2, and HE3 does not appear in the fault tree logic, HE3 is also missing from the initial minimal cutset,

The full frequency adjustment for the recovery action HE2 should not be applied to the entire frequency of the initial minimal cutset because that would neglect the dependence of HE2 on HE3 for the fraction of time that HE3 also fails.

Depending on whether the recovery rule decreases the initial minimal cutset frequency, this approach to recovery may be non-conservative.

Recovery Rules for Dependencies Between Human Failure Events. Recovery rules are also used to modify the basic event probabilities in the retained minimal cutsets to account for dependencies between events that are not easily built into the fault trees. They are difficult to include because they require the use of NOT logic to distinguish sequence conditions when the dependencies are valid from other sequence conditions (i.e. the same small event tree sequence) when the human failure events may be assumed independent. Such recovery rules are subject to the same lack of a mathematical basis as those previously discussed.

Also, human failure event dependencies are generally accounted for by accepting the Swain and Guttman (NUREG/CR-1278) human error rate dependency model. This model says that the second and subsequent operator action error rates may be dependent on the success or failure of earlier actions along the same sequence. Since the resulting sequence minimal cutsets from a FTL model do not include successful actions, only a portion of the Swain and Guttman dependency model can be implemented. The reduction of dependencies caused by intervening successful actions cannot be accounted for.

Recovery Rules for Exclusive System Alignmewnts. One modeling approximation common to most fault tree models, has to do with the treatment of maintenance events; i.e. the fractions of time in which a plant system train is in maintenance. Often such events are represented as basic events in the system fault trees. Plant technical specifications often preclude redundant trains from being in maintenance with the plant at power; i.e. the plant may be required to shutdown quickly if both trains of the same system are down for maintenance. Recovery rules are often used to exclude minimal cutsets which do include such multiple system train maintenance events.

While this approach is a better approximation it is not exact. The problem lies in representing system maintenance alignments as basic events within the fault trees. This is inappropriate because maintenance events are not true basic events, independent of all other events in the fault tree, yet all fault tree software evaluation tools (including BDD engines) process them as independent events.

As an example, consider a two train system, either train of which may achieve success. We represent maintenance on trains 1 and 2 by the events U1 and U2. Other failure modes of each train are represented by the events X1 and X2. Maintenance on both trains simultaneously is prohibited with the plant at power. The combinations of these events leading to system failure can be represented by a fault tree using NOT logic to exclude the simultaneous maintenance events, or by a fault tree without the NOT logic but later subject to a recovery rule deleting the minimal cutsets of the fault tree that include both maintenance simultaneously.

Let us represent the probabilities of each of these events by the lower case letters; i.e. u1, u2, x1, and x2. The exact algebraic equation for failure of this system written without NOT logic can be written as:

$$\mathrm{EXACT} = (u1)*x2 + (u2)*x1 + (1-u1-u2)*x1*x2$$

The terms in parentheses are the three alignment fractions that the system may be in at the start of the sequence, and the remainder of each term is the conditional probability of failing the system given that alignment. Effectively, then, the system fault tree probability is solved three times, once for each alignment, and the results added to obtain the total system probability for all three possible alignments.

Alternatively, the full fault tree with NOT logic can be solved using prime implicants, minimal cutsets, or BDD solutions.

The resulting fault tree prime implicants are:
$X1*X2*/U1 + X1*X2*/U2 + X1*U2*/U1 + X2*U1*/U2$

The rare event probability total for these prime implicants is then

$$\mathrm{PI\ RE} = x1*x2*(1-u1) + x1*x2*(1-u2) + x1*u2*(1-u1) + x2*u1*(1-u2)$$

The corresponding minimal cutsets are:
$X1*X2 + X1*U2 + X2*U1$

The minimal cutset rare event probability total is then:

$$\mathrm{MCS\ RE} = x1*x2 + x1*u2 + u1*x2$$

The MCUB probability total for these cutsets is:

$$\mathrm{MCS\ MCUB} = 1- (1-x1*x2)*(1-x1*u2)*(1-u1*x2)$$

The BDD solution is more difficult to write down even for this simple example. The results here were obtained by a BDD software tool.

The results are compared in the following table for several combinations of maintenance fractions (i.e. u1, u2) and other train failure probabilities (i.e. x1,x2).

Comparison of Fault Tree Solutions with Excluded Maintenance Alignments

x1,x2	u1,u2	EXACT	PI RE	PI REDUCED	MCS RE	MCS MCUB	BDD

1.00E-03	0.01	2.098E-05	2.18E-05	2.08E-05	2.10E-05	2.10E-05	2.08E-05
1.00E-03	0.05	1.01E-04	9.69E-05	9.60E-05	1.01E-04	1.01E-04	9.59E-05
1.00E-02	0.01	2.98E-04	3.96E-04	2.97E-04	3.00E-04	3.00E-04	2.96E-04
1.00E-02	0.05	1.09E-03	1.14E-03	1.05E-03	1.10E-03	1.10E-03	1.04E-03
1.00E-01	0.1	2.80E-02	3.60E-02	2.70E-02	3.00E-02	2.97E-02	2.61E-02
2.00E-01	0.2	1.04E-01	1.28E-01	9.60E-02	1.20E-01	1.15E-01	8.96E-02

The BDD solution gives the exact top event failure probability of the modeled fault tree with NOT logic. It is seen that the BDD answers are always lower than the exact system failure probability. The amounts lower depends on the event probabilities assumed. The difference between the BDD solution and the EXACT results for ther system failure probability is attributed to the approximate modeling of the maintenance alignments as independent events within a fault tree. Therefore, the use of BDD solutions of the fault tree representation always under predicts the true answers.

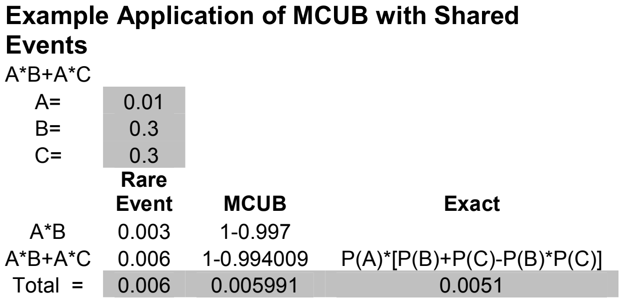
Applying the rare event approximation to the prime implicants for this problem often, but not always, gives the highest answer; i.e. column PI RE. In this problem, the same incomplete Boolean reduction issue noted under the discussion of NOT logic above also applies here for the prime implicants of this problem. No fault tree tool can fully reduced the prime implicants observed in this problem.

However, by expanding the 4 prime implicants to min-term form and then Boolean reducing (as would occur when applying the BDD solution), one can see that the first of the four terms drops out. Once fully reduced, the quantification of the remaining three prime implicants gives a much closer solution to the EXACT expression. This reduced form is titled PI REDUCED in the above table. The results for PI REDUCED always underpredict the EXACT answers as expected.

Rare event or MCUB solutions from minimal cutsets give approximate answers that for this simple example are always the same or higher than the exact solution. When the values are small, the solutions are all very close. If the event failure probabilities are increased the conservatism becomes noticeable even for this simple example. One cannot always count on these approximate solutions always being conservative. The BDD solution clearly shows that the fault tree representation itself of maintenance alignments as independent events is non-conservative.

Cutset Totaling: Two different methods for totaling the frequencies of recovered, minimal cutsets to obtain the core damage frequency or large early release frequency are used; i.e. the rare-event approximation (i.e. first term of the Sylvester-Poincaire expansion) and the Min-Cut-Upper Bound (MCUB) approximation which subtracts the probability of all minimal cutsets being successful from 1.0. Both approaches make the assumption that the minimal cutset frequencies are independent. Though, in reality, they are ***not*** independent because some higher minimal cutsets share the same basic events.

A simple illustration of this is seen in the example below. Two cutsets are used in this example, which share an event A. In this case, the rare-event and MCUB values are reasonably close. However, the shared event A may be factored out and the exact value computed. For the values used in this simple example, the exact answer is 15% smaller than those computed by the rare-event and MCUB approaches. Such shared events can appear frequently in lists of minimal cutsets from complex PRA models. Minimal cutsets involving the same system alignment or maintenance conditions appear frequently in complex PRA model results; i.e. sharing the same basic event representing the alignment. Higher order cutsets (i.e. order 2 or higher), also frequently share the same basic events.



Example Application of MCUB with Shared Events

A*B+A*C

A=	0.01
B=	0.3
C=	0.3

	Rare Event	MCUB	Exact
A*B	0.003	1-0.997	
A*B+A*C	0.006	1-0.994009	P(A)*[P(B)+P(C)-P(B)*P(C)]
Total =	0.006	0.005991	0.0051

The rare-event approximation is known to be excessively conservative, when individual basic event probabilities are large; e.g. for earthquakes. Moreover, even for non-earthquake sequences, the frequency of contributors to core damage frequency computed using the rare event approximation in one complex model with a truncation cutoff of $1x10^{-13}$ was found to be conservative by a factor 5, even when the “delete term” approach for modeling success sequences is applied. The frequencies of some individual sequences were found to be pessimistic by much greater factors (References 1 and 2).

The min-cut-upper bound (MCUB) approach, on the other hand, is derived by computing the joint probability of none of the minimal cutsets occurring, and then subtracting this joint probability from 1.0. This approach prevents the conditional probability of the fault tree top event from exceeding 1.0, thereby limiting the excessive conservatism found when using the rare-event approximation. The MCUB approach can be exact when the minimal cutsets are independent; i.e. no shared basic events. When the minimal cutsets are not independent, the MCUB approach can still give an upper bound to the cutset total which is lower than the rare event approximation. However, there are some practical limitations which may make the MCUB approach inaccurate. These are discussed below.

The MCUB approach is only an upper bound when negation is not present. When the event combinations are derived from fault trees containing NOT Logic, the MCUB result may be non-conservative. Below is an illustrative example of a simple fault tree that contains NOT logic. There are only two terms in the example; i.e. A and (NOTA)*B. Note that since the two terms are mutually exclusive, the rare-event approximation in this case gives the exact answer. As is typical, the computed MCUB value is slightly lower than the rare-event value. However, in this case, the rare-event value is exact so the MCUB approach gives an answer which is non-conservative. Therefore, when NOT logic is used in the fault tree, the MCUB **cannot** be considered a bound on the true answer.

Example Application of MCUB with NOT Logic

A+-A*B	
A=	0.1
B=	0.1

	Rare event & Exact	MCUB	
A	0.1	0.9	(1-.9)
(-A*B)	0.09	0.819	.9*(1-.09)
Total =	0.19	0.181	(1-.819)

In order to simplify and speed up the quantification of fault tree models, some users include the initiating event frequency as a basic event within the linked-fault tree logic for the response of the plant. Indeed, for FTL based risk monitoring models, often all initiators are included as basic events in the linked fault tree logic. This approach eliminates the benefits of the MCUB approach to cutset totaling, as illustrated below.

Two cases are shown, one for high initiator frequencies and the second case for low initiator frequencies. In both cases we postulate two initiators, each of which combines with either event failure A or B to define sequences assumed to result in core damage. Four core damage sequences result. The initiator frequencies and event values for A and B are provided. Relatively high values for conditional events A and B are used to illustrate the points.

In case 1, it's clear that the MCUB approach gives a much lower bound than the rare-event approximation. The problem is that it's too low! In this case the exact answer can be computed. One therefore cannot use the MCUB approach when incorporating multiple initiator frequencies as basic events because the frequency of interest can be greater than 1. What cannot be greater than 1 is the conditional probability of the sequence given that the initiator occurs. This fact is accounted for in the exact equation for the total frequency, but not by the MCUB approach.

Initiators as Basic Events; Case 1: High Initiator Frequencies

Names	**Values**
IE1	1
IE2	1
A	0.75
B	0.75

Seqs.	**Seq. Freq**	**Exact Equation**	**Exact Frequency Total**	**Rare Event**	**MCUB**	**MCUB**
IE1*A	0.75			0.75	0.25	(1-.75)
IE1*B	0.75	IE1*(A+B-A*B)=	0.9375	1.5	0.0625	.25*(1-.75)
IE2*A	0.75			2.25	0.015625	.0625*(1-.75)
IE2*B	0.75	IE2*(A+B-A*B)=	0.9375	3	0.003906	.015625*(1-.75)

Totals=		1.875	3	0.996094	1-.003906

In case 2, much lower initiator frequencies are used. In this case, the MCUB result is again lower than the rare-event approximation but the results are nearly the same. The exact answer is substantially lower. The problem lies in the fact that the MCUB approach only applies to the conditional probability given an initiating event, and not to the sequence cutset as a whole. Cutset totaling schemes should apply the MCUB approach to sequences with the same initiating event, one initiator at a time, and then to add the initiator results directly. Users should insure that the MCUB approach is applied separately for each initiator, the result weighted by the initiator frequency, and only after that are the results summed over all initiators.

When FTL models include the full logic for system initiators directly in the linked fault trees (i.e. not just as a single basic event for each system initiator), this again becomes a problem. The failure of an individual pump to run during the yearly mission time is not by itself an initiator yet all sequence cutsets that begin with such failures should be combined separately using the MCUB scheme before totaling. The authors are unaware of any FTL tool that applies the MCUB approach in this way when system initiator fault trees are linked in with the plant response.

Initiators as Basic Events, Case 2: Low Initiator Frequencies

Names	Values
IE1	1.00E-03
IE2	1.00E-03
A	0.75
B	0.75

Seqs.	Seq. Freq	Exact Equation	Exact Frequency Total	Rare Event	MCUB	MCUB
IE1*A	0.00075			0.0008	0.99925	(1-.00075)
IE1*B	0.00075	IE1*(A+B-A*B)=	0.0009375	0.0015	0.998501	.99925*(1-.00075)
IE2*A	0.00075			0.0023	0.997752	.998501*(-.00075)
IE2*B	0.00075	IE2*(A+B-A*B)=	0.0009375	0.003	0.997003	.997752*(1-.00075)
	Totals=		0.001875	0.003	0.002997	1-0.997003

To address this cutset totaling issue in a more rigorous way, some have proposed to instead use a BDD developed directly from the list of minimal cutsets that survive the frequency truncation and the application of recovery rules. This approach would resolve the cutset totaling issue if applied to groups of cutsets all involving the same initiator. However, it still leaves the frequency truncation issue because all cutsets not in the retained list are still missing. Moreover, the resulting answer would be the lowest possible (i.e. compared to rare-event or MCUB), making any omissions caused by frequency truncation that much more important.

Issues Related to Model Results

Sequence Cutsets: The contributors to core damage frequency from a fault tree linking analysis are presented in the form of basic event level, sequence cutsets. For most fault tree linking models, the sequence cutsets are combined into a single large fault tree to avoid some of the concerns associated with NOT logic and success events along each sequence path. As a result, the sequence cutsets are not associated to any one particular core damage sequence path through the event trees. Moreover, it is often difficult to determine exactly why a combination of basic event failures results in core damage, and therefore if the large FTL model logic is actually correct. Reviewers of the sequence cutsets must have an extensive knowledge of the plant intersystem dependencies and of the fault tree model approximations. Even if an associated event tree sequence path is identified, the small event tree events are defined at too high of a functional level to capture the actual sequence of events, the response of support systems, and the systems guaranteed to be failed by the loss of other systems; i.e. identifying the applicable small event sequence is often not enough to determine why the event combination results in core damage.

Importance Measures: Importance measures can only be computed for the basic events found in those minimal cutsets that are retained following logic reduction, frequency truncation, and application of the recovery rules. In practice, the frequency truncation applied during sequence quantification is at such a high level that for complex models, most (>50%) basic events may be truncated out. One real nuclear plant example retained only 16% of the model basic events after application of a truncation cutoff of $1x10^{-11}$(Reference 1). It cannot be concluded, nor assumed that the basic events truncated out are of low risk significance, because, as noted earlier, convergence of the core damage frequency results cannot be demonstrated with an accuracy of 0.5%; i.e. the most commonly used criterion separating risk significant and not risk significant.

Two tables comparing importance measures computed using Binary Decision Diagrams (i.e. BDD results in exact totaling) are repeated below from reference 1.

We consider first Table 9 of reference 1. The criticality importance measure is equivalent to the Fussell-Vesely importance when computed using minimal cutsets. In the highest ranked 20 basic events by criticality importance, the BDD results are largely higher than the measures computed using minimal cutsets by about 40% to 50%. Only for the very highest ranked basic events, did the BDD measures turn out lower. This suggests that using the FTL quantification approach may underestimate basic event Fussell-Vesely importance measures, especially those near the boundary between high risk and low risk significance; i.e. Fussell-Vesely importance value of .005.

Table 10 from reference 1 ranks the top 20 basic events for the same model by BDD computed risk achievement importance (RAW) measures. The striking finding from this ranking is that 12 out of the 20 highest ranked basic events by RAW were truncated in the minimal cutset approach to quantification. Clearly, when risk significance is measured by RAW values, the potential for omitting significant basic events (i.e. RAW greater than 2, references 5 and 6) because of frequency truncation in the fault-tree linking approach is very high.

Another importance measure is called fractional importance. This importance measure is easy to interpret physically. Fractional importance of an event represents the fraction of core damage frequency that also involves failure of the event. Fractional importance is not the same thing as Fussell-Vesely importance, nor criticality importance. In general, fractional importance is higher than or equal to either of those measures. Consider the event of reactor coolant pump seal return isolation in a PWR station blackout sequence. Failure to isolate would not contribute to core damage, nor to a large early release; i.e. the resulting hole is too small. However, failure to isolate the containment is an important consideration for emergency planning and emergency response. FTL models developed to compute CDF or LERF cannot be used to determine the fractional importance of failing to isolate the RCP seal return line. In fact, the Fussell-Vesely importance to CDF or LERF of failing to isolate the RCP seal return line, if indeed it is even modeled, should be zero! The Boolean reduction process for either CDF or LERF should eliminate the cutsets involving failure to isolate the RCP seal return line, hence they would not contribute to Fussell-Vesely importance. This is just one illustrative example. The fact is that fractional importance is not the same measure as the Fussell-Vesely importance, yet in some applications, it is the higher, fractional importance measure that is of interest.

Another issue is that of determining importance measures for alignments. In FTL models, system maintenance alignment fractions are routinely included as basic events in the logic. When systems are normally aligned with different pumps running, the fractions of time for each alignment are also entered as basic events. The problem is that alignment fractions are not the same thing as failure modes.

One distinct property system alignment fractions have is that the sum of the alignment fractions for a given system must equal 1.0. The problems associated with basic event representations when the values are greater than 0.1, have been discussed above. Further, some importance measures require that their values be set to 1.0 and the core damage frequency then be recalculated ; e.g. for RAW. But if one alignment fraction is set to 1.0, the other alignment fractions must be set to zero so that the system failures are not double counted. This adjustment for RAW importance cannot be made with FTL, because the basic events representing these alignments are assumed independent. Furthermore, often the alignment fractions representing the normal system alignment (i.e. when no maintenance or testing is underway) are omitted from the model; i.e. they are assumed nearly equal to 1.0. Similarly, for risk reduction, when one alignment fraction is set to zero, the others must increase so that the sum of all alignments remains 1.0.

For ETL, these constraints can and are made. The system in question is evaluated once for each alignment and the failure probability for each alignment is weighted by the alignment fraction to obtain the total system failure probability. When computing RAW for a particular alignment, the contribution from the other alignments is zeroed out. When computing risk reduction, the alignment in question is zeroed out and the others increased so that the sum of the alignment fractions remains 1.0.

Table 9. The 20 most important basic events according to their CIF (or equivalently, Fussell-Vesely Importance) for sequence 4 sorted by the second column.

Rank	Criticality Importance Computed Using BDD	Fussell-Vesely Importance Computed Using Cutsets
1	0.605467	0.81045
2	0.441697	0.615231
3	0.288971	0.172044
4	0.141314	0.135865
5	0.137265	0.223926
6	0.137265	0.223926
7	0.137265	0.223926
8	0.0622317	0.041397
9	0.0622317	0.041397
10	0.0622317	0.041397
11	0.0622284	0.041397
12	0.0622284	0.041397
13	0.0622284	0.041397
14	0.0573555	0.0511157
15	0.0573555	0.0511157
16	0.0522112	0.0355426
17	0.052187	0.0355426
18	0.0429735	0.0302301
19	0.0429735	0.0302301
20	0.0429735	0.0302301

Table 10. The 20 most important basic events according to their RAW for sequence 4 sorted by the second column

Rank	RAW Computed Using BDD	RAW Computed Using Cutsets
1	68803.5	92096.8
2	6770.64	1
3	6770.64	1
4	6770.64	1
5	6770.64	1
6	6770.64	1
7	4405.06	1
8	4405.06	1
9	2822.88	1
10	2822.88	1
11	827.259	795.395
12	827.259	288.623
13	827.259	288.625
14	827.259	1
15	827.259	265.348
16	827.259	265.348
17	827.259	265.348
18	762.086	1
19	762.086	1
20	390.821	636.928

Sequence Rankings: The frequency truncation and cutset totaling issues described previously means that individual sequence frequencies for a complex model may be either optimistic or pessimistic (Reference 2). This makes ranking of small event tree core damage sequences dubious at best.

Issues Related to Modeling Detail for Risk-Informed Applications

Common Cause: Some complex fault tree linking models have simplified the common cause failure modeling because a full expansion of the total failure rate into its independent and common cause events, particularly for large order common cause groups, greatly expands the number of basic events required. The US NRC has itself adopted simplified common cause models in its SPAR models for each nuclear power plant. Simplifying the common cause failure treatment limits the frequency truncation issues but makes it difficult to extract risk information from the modeling results.

Seismic Events: An accurate assessment of seismic events requires the combination of seismic failure events and random failures from non-seismic causes. Incorporating the seismic events is difficult because the event failure probabilities are not negligible, and they increase with seismic magnitude. The rare-event approximation and MCUB approaches to cutset totaling break down when event probabilities get large. Moreover, for larger earthquakes, when multiple seismic failures may each lead to core damage, importance measures require extra care. The Fussell-Vesely importance measure can be misleading for seismic events because it overstates the importance of each individual component failure; i.e. due to shadowing effects.

Internal Fires and Low Power and Shutdown Events: Separate models must be developed for each unique fire scenario or low power and shutdown plant operating state. Maintenance unavailabilities are higher (both in frequency and duration) than during power operation. This again violates the rare-event and MCUB assumptions making the totaling of minimal cutset probabilities suspect.

Level 2 Analysis: The containment response to severe accidents (i.e. Level 2 analysis) requires many events in the sequence model, each of which may have substantial probability, especially those modeling phenomenological events. For example, the models developed for the NUREG-1150 projects utilized event trees with more than 100 top events. Fault trees are not a good tool for quantifying such sequence models. The requirement for the use of NOT logic and the large event probabilities violates the approximation assumptions for sequence quantification using linked fault trees. The need to compute separate release category frequencies for use in Level 3 calculations further increases the model complexity.

References:

1. S. Epstein and A. Rauzy, “Can We Trust PRA?” Reliability Engineering and System Safety, 88:195-205, 2005.

2. S. Epstein and A. Rauzy, “Can We Trust PRA? (Take 2)”, PSA’05 – American Nuclear Society Topical Meeting on Probabilistic Safety Analysis, September, San Francisco, California, 2005.

3. S. Epstein, A. Rauzy, and D. Wakefield, “Can We Trust PRA (Take 3)”, PSAM 08, May 15-19, 2006, New Orleans,LA.

4. “ASME Standard for Probabilistic Risk Assessment Level 1 Internal Events at Power, Including LERF”, Addendum B, 3/2005, to be published.

5. NEI 00-04, “10 CFR 50.69, SSC Categorization Guideline, Final Draft”, Nuclear Energy Institute, April, 2004.

6. US NRC, “ An Approach for Determining the Technical Adequacy of Probabilistic Risk Assessment Results for Risk Informed Activities“, Regulatory Guide 1.200 for trial use. December 2003