

SANDIA REPORT

SAND97-2695 • UC-706

Unlimited Release

Printed November 1997

An Investigation of New Mathematical Structures for Safety Analysis

J. Arlin Cooper, Timothy J. Ross

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000.

Approved for public release; further dissemination unlimited.



Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

NOTICE: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Prices available from (615) 576-8401, FTS 626-8401

Available to the public from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Rd
Springfield, VA 22161

NTIS price codes
Printed copy: A04
Microfiche copy: A01

SAND97-2695
Unlimited Release
October 1997

Distribution
Category UC 706

An Investigation of New Mathematical Structures for Safety Analysis

J. Arlin Cooper
System Studies Department
Sandia National Laboratories
Albuquerque, NM 87185-0490
acooper@sandia.gov

Timothy J. Ross
University of New Mexico
Civil Engineering Department
Albuquerque, NM 87131
ross@unm.edu

Abstract

In probabilistic safety analyses, the input data and logical combinations of the data can involve considerable subjectivity. Subjectivity is not usually well described mathematically. It is important that mathematical processing of subjective information be done in as meaningful a manner as possible, especially for assessing the safety of high consequence operations. We have begun an investigation of some innovative mathematical structures that are potentially useful for processing subjective information in such safety analyses, and have developed some practical illustrations of the utility of the approaches. We have addressed active and passive safety systems, independent and dependent inputs, fault trees and event trees, and crisp and distributed logic. Our long-range intent is to provide analysts with a comprehensive selection of mathematically based tools, so that the best match possible is available for a particular system or portions of systems.

The recent successes of fuzzy logic and fuzzy and hybrid mathematics in portraying subjectivity is a reminder that a selection made from the most applicable mathematical tools is more important than forced adaptation of conventional tools. In this paper, we consider new approaches that enhance conventional and fuzzy PSA by improved handling of subjectivity. The most significant of the mathematical structures we have investigated (from a standpoint of safety analysis applications) will be described, and the general types of applications will be outlined.

Some of the mathematical structures investigated are based on various logical norms, Ordered Weighted Averaging, implication operations, Gamma operators, sigmoid operators, inference networks, failure "race" comparators, min/max ordinate logic, threshold logic, conditional possibility, Frechet-based dependence operators, and constrained mathematics.

Acknowledgments

DOE Laboratory Directed Research and Development (LDRD) funds supported most of the work on this project. A large number of people acted as reviewers, idea-exchangers, and contributors, including George Klir, State University of New York, Binghamton, Scott Ferson, Applied Biomathematics, Don Wunsch, Texas Tech, and Dave Carlson and Michael Bohn, Sandia National Laboratories. The COSMET software development was expertly done by Bob Roginski, Sandia National Laboratories, who also offered considerable support interacting on the concepts.

TABLE OF CONTENTS

1. INTRODUCTION.....	1
2. CONTEST TOOL.....	4
3. MIN/MAX INTERVAL LOGIC TOOL.....	4
4. MIN/MAX ORDINATE TOOL.....	4
5. FUZZY “AND” TOOL.....	5
6. FUZZY “OR” TOOL.....	5
7. LUKASIEWICZ NORMS.....	6
8. COMPOSITION OPERATOR.....	6
9. WEIGHTED SUM TOOL.....	6
10. FUZZY SIMILARITY RELATIONS.....	7
11. ACTIVE SAFETY SYSTEM APPROACH.....	8
12. PASSIVE SAFETY SYSTEM APPROACH.....	9
13. USE OF CONSTRAINED MATHEMATICS.....	10
14. SOFTWARE.....	15
CONCLUSIONS.....	16
REFERENCES.....	17

LIST OF FIGURES

1. A Solution for Tensile Failure of Series-Parallel Chain Links.....	5
2. A Schematic of the Example Fault Tree.....	8
3. Contest Tool Plot.....	16
4. Weighted Sum Example Problem.....	16
5. Extreme Minmax Example Problem.....	16
6. Link Minmax Example Problem.....	16

An Investigation of New Mathematical Structures for Safety Analysis

J. Arlin Cooper
Sandia National Laboratories*
Albuquerque, NM 87185-0490
acooper@sandia.gov

Timothy J. Ross
University of New Mexico
Civil Engineering Department
Albuquerque, NM 87131
ross@unm.edu

ABSTRACT

In probabilistic safety analyses, the input data and logical combinations of the data can involve considerable subjectivity. Subjectivity is not usually well described mathematically. It is important that mathematical processing of subjective information be done in as meaningful a manner as possible, especially for assessing the safety of high consequence operations. We have begun an investigation of some innovative mathematical structures that are potentially useful for processing subjective information in such safety analyses, and have developed some practical illustrations of the utility of the approaches. We have addressed active and passive safety systems, independent and dependent inputs, fault trees and event trees, and crisp and distributed logic. Our long-range intent is to provide analysts with a comprehensive selection of mathematically based tools, so that the best match possible is available for a particular system or portions of systems.

The recent successes of fuzzy logic and fuzzy and hybrid mathematics in portraying subjectivity is a reminder that a selection made from the most applicable mathematical tools is more important than forced adaptation of conventional tools. In this paper, we consider new approaches that enhance conventional and fuzzy PSA by improved handling of subjectivity. The most significant of the mathematical structures we have investigated (from a standpoint of safety analysis applications) will be described, and the general types of applications will be outlined.

Some of the mathematical structures investigated are based on various logical norms, Ordered Weighted Averaging, implication operations, Gamma operators, sigmoid operators, inference networks, failure “race” comparators, min/max ordinate logic, threshold logic, conditional possibility, Frechet-based dependence operators, and constrained mathematics.

1. INTRODUCTION

Failure is a nearly unavoidable phenomenon with complex technological systems and products. When one considers that perhaps there can be various degrees of degradation between complete failure and no failure, and that many systems contain both random and ambiguous kinds of description, it becomes a natural step to consider the utility of fuzzy or possibilistic methodologies in describing this new paradigm. Fuzzy and possibilistic methodologies can serve as a complementary approach to probabilistic methods. The use of fuzzy methodology to system failure engineering can be traced back to the work by Kaufmann [1]. He introduced the notion of a component *possibility* as a reliability index to supplement that of component *probability*. Then, in the 1980's significant strides in the use of fuzzy logic in system failure engineering started to appear in the areas of human reliability, hardware reliability, software reliability, and structural reliability [2].

* Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000.

One aim in this paper is to suggest that the general paradigm of logical models for undesired outcomes, conventionally described by fault trees and event trees from the probabilistic

perspective, can be supplemented considerably to extend the current approach to include various gate logics, to represent probability and possibility as membership functions on probabilities and possibilities, and to propagate fuzziness and dependence in the logic using a variety of mathematical structures to supplement the standard Boolean logic.

Probabilistic safety analyses (PSAs) have been performed for many years applied to a variety of situations. The most conventional (and convenient) form of PSA is to prescribe the probability of occurrence for particular “events” that could contribute to failure, and to combine such events in a crisp logic structure (e.g., a fault tree, or an event tree). Probabilistic calculus or various forms of Monte Carlo analysis are capable of accounting for stochastic variability and correlation. However, this basic PSA methodology does not accurately portray some situations, most significantly when the inputs contain a significant degree of subjectivity, or are dependent, or the models are inexact [3, 4].

Because of the development of more complex, higher consequence systems, and because of businesses’ and society’s increasing reluctance to accept overly optimistic safety analyses, it is now becoming important to improve analysis approaches. Furthermore, almost all safety analyses depend on significant subjectivity, both in determining inputs and in determining models for processing the inputs. Not only is this factor not treated directly in conventional analyses, but its role is rarely indicated to the recipient of the analyses. For these reasons, it is important to consider new innovative approaches that may enhance conventional PSA by improved handling of complex dependence interactions, “soft” failures, and subjectivity factors.

In this paper, techniques based on new mathematical structures will be described, including continuous (e.g., fuzzy) logic operations, various “norms” for logic combination, and new methods for deriving output information for combinational logic (and time-dependent) processing of inputs. These can serve as a supplementary approach to probabilistic methods

The new methodologies we have developed are intended to optimize the use of experimental data, expert elicitation, and mathematical models. Some of these have now been incorporated in

software in order to provide safety analysts with improved automated tools.

The essence of fuzzy and possibilistic logics is in the capability to model imprecision and vagueness by assigning membership functions to the variables. Membership functions (*mfs*) differ from probability distribution functions (*pdfs*) in that, the *pdfs* describe the uncertainty in the future states of the variables while *mfs* describe the vagueness in the definition of the state itself or the imprecision in the measurement or the representation of the variable used to characterize the future state. A principal difference in classical probability models and fuzzy models is the adherence to the law of additivity. Probability models require a strict adherence, implying that a value assigned to a variable cannot be assigned to a different set of variables. Fuzzy and possibilistic logics, allow a diverse classification, thus enabling partial membership in various sets (this can be useful in modeling rare events, wherein data available are very limited or the engineering knowledge content is low).

Due to intra-component dependence and the rare or otherwise unknown failure rates, it is appropriate to model some systems by fuzzy set or possibilistic theory. For example, in the case of a capacitor which ceases to function under high temperatures, the capacitor can deteriorate or have built in imperfections which cause the component not to behave the way it was designed to operate as temperature increases. Instead of assigning singleton values of probability to each basic event in a logic structure, it is sometimes more appropriate to define the component failure probabilities, and eventually the system failure probability, by *mf* or possibilistic functions. Intra-component dependencies may arise due to the aggregation of various sub-components, each of which are represented in the logic structure as single components. The extension of this concept to systems and inter-component dependence is natural.

With these preliminary ideas considered, we can articulate a broader, more general paradigm for the assessment of uncertainty in logical models of failure. The current prevalent approach makes use of a Boolean logic and a “tree” structure, (1) where the gates in the fault trees or event trees essentially use the classical notion of inference, (2) where the basic event probabilities are either

single values, or are represented by probability density functions, and (3) where the gate logics are the classical probability norms. We extend this current approach to a more comprehensive approach (1) where the gates in the trees use any of a variety of inference schemes to provide confidence in the data available for the basic event failures, (2) where the basic event probabilities are characterized by fuzzy or possibilistic mfs, and (3) where the gate logics are expressed by any number of various logical norms. The advantages of this approach are: we can model more than just random uncertainty in the tree, we can assess more general gate logics, we can embed notions of time variability and component dependencies into the tree, and we can characterize, in numerical terms, the strength of our confidence in the final estimates of failure probability for the tree.

We began our work by conducting a literature search for mathematical structures that had been considered for various applications, and devoted considerable time on developing some of our own origination. We identified twelve general structures, which are:

1. A "Contest" tool, which allows two or more fuzzy entities to compete for success (or failure). The mathematical result is the fuzzy possibility of succeeding for each entity.
2. A "Weighted Sum" tool, which allows weighted accumulation of possibilistic factors such as safety program attribute metrics and comparison to a fuzzy goal.
3. Fuzzy "anding," where the satisfaction of the "and" logic has a membership, rather than a crisp yes or no or a probability.
4. Fuzzy "oring," where the satisfaction of the "or" logic has a membership, rather than a crisp yes or no or a probability.
5. Min/Max interval length logic, where the interval bounds on a range of possible abscissa values can be "stretched" along the abscissa.
6. Min/Max ordinate logic, where for each logic component the smallest ordinate value of a collection at each abscissa point determines the possibilistic result or the maximum value determines the possibilistic result.
7. Implication operators, where the implication does *not* represent probabilities of failure; but rather represents what is happening implicitly in the logic at a particular gate.
8. Lukasiewicz norms, which for the probabilities of AND gates are closer to zero and the OR gate probabilities closer to one than the probabilistic norms. This feature is especially interesting in systems where the basic event probabilities are close to zero or one.
9. Composition operators, which are another mechanism to propagate fuzzy probabilities through a logic tree. The norms are used to aggregate probability information at the basic level in a logic tree. The propagation of this aggregated value will then be propagated to the next higher level using a composition operator. This propagated value, however, will be associated with a confidence value that will come from the nature of the implication used, as discussed. This idea is also powerful when an analyst wishes to model dynamic features or intra-component dependence issues with a composition.
10. OWA (ordered-weighted-average) [5, 6] operators provide an ordering of the probabilities of events at a branch of a logic tree that are multiplied by weights to produce the ordered operator. This operator expresses the top-level probability of that particular branch of a tree. The linear operator used is a summation, because the parameters are discrete and countable. If the linear operator summation were replaced by a linear integral, the effect is that the branches of the tree become continuous and uncountable. Second, the weights become a continuous function, instead of a series of delta functions (discrete valued quantities). Third, the mathematical constraint that the discrete weights sum to unity becomes a constraint that the integral of the weighting function is equal to unity. Fourth, the probabilities become probability density functions. This can

lead to continuous models for fault trees and event trees.

11. Multiobjective decision problems have been investigated as possible analogs to the logical constructs of fault and event trees. In these kinds of problems there is involved the selection of one alternative a_j , from a universe of alternatives A given a collection, or set, say $\{O\}$, of criteria or objectives that are important to the decision maker. We want to evaluate how well each alternative, or choice, satisfies each objective, and we wish to combine the weighted objectives into an overall decision function in a plausible way.
12. Fuzzy similarity relations investigated include two methods to propagate fuzzy information within a fault tree—a double composition method and a normative method. An augmentation of the normative method will be illustrated.

These mathematical processes were all evaluated as potential safety analysis tools. Some promising applications were identified and will be outlined in this paper. Finally, the interrelationship of the applicable tools into a single safety assessment tool was considered. Many of the results were implemented in software routines, which will be described. Some of the approaches considered will be addressed in more detail.

2. CONTEST TOOL

The “contest” tool can be used for quantitative or qualitative information. The input abscissa is a linear measure of “stress,” which could be temperature, pressure, acceleration, etc. The user defines the stressor and the units (e.g., degrees Centigrade) and a fuzzy number about where failure occurs for two or more entities. There could be a “family” of these contests entered to allow for some parameter like the direction from which the stressor source comes, rate of increase of temperature, etc. Each one implies a possibilistic subtraction, e.g., $S_i - W_i$, where W_i is one possibilistic number and S_i is the other possibilistic number. Then we can calculate an mf for the result. The maximum of these results is an mf for the result of each pairwise contest. Another aspect of this tool is that the stressor may be bounded (a possibilistic entity). In this

case, the stress and both response characteristics of every pair interact together to give a possibility of failure. This is done by computing the ordinate failure possibility at each abscissa point of the stress mf.

3. MIN/MAX INTERVAL LOGIC TOOL

In some risk analysis problems, reasonable risk is estimated by discounting certain events that are not considered credible. These can either contribute to failure or act to prevent failure. In probabilistic risk assessment, these are possibilistic functions with probabilistic abscissas and level-of-presumption ordinates. A useful strategy is to take the overall event mf and show an upper and lower extension for including low-credibility threats and excluding low-credibility preventive events. The maximum abscissa value at each level of presumption for the cause functions gives the upper extension, and the minimum abscissa value at each level of presumption for the prevent functions gives the lower extension. This allows demonstrating that situations could be even worse (or better) if slightly less credible occurrences happen, as well as specifying the implications of discounting these occurrences.

4. MIN/MAX ORDINATE TOOL

Consider a “sum of cutsets” Boolean logic equation. Each input has a stressor for an abscissa and a failure possibility for an ordinate. These functions increase monotonically from zero possibility to one possibility as stress increases. An “and-like” ($\&$) operation is done by considering each level of presumption for the smallest ordinate value and an “or-like” (\vee) operation is done by considering each level of presumption for the largest ordinate value.

$$A \& B = \min[A(x), B(x)] \quad (1)$$

$$A \vee B = \max[A(x), B(x)] \quad (2)$$

The user equation for combining the terms includes $\&$ functions and/or \vee functions. For the $\&$ function, the minimum of all the operand ordinates is computed at each abscissa value. For the \vee function, the maximum of all the ordinate values is computed at each abscissa value. This function is called the intermediate function, $I(x)$.

There must also be a (user-defined) stress function, showing a values (point, square, triangle, or trapezoid) for the possibility of levels of applied stress. Now we apply this possibility-of-stress function to the intermediate function to obtain the tool output. This is done by computing the ordinate membership values of response for each level of stress. The spectrum of possibility of failure thereby portrayed can also be associated with stressor membership, for example by portraying a horizontal bar graph of possibilities with the central part corresponding to the maximum level of stressor membership portrayed with more plot intensity.

$$F(x) = \int S(x)I(x)dx \quad (3)$$

where the ordinate of $I(x)$ over the abscissa range of $S(x)$ gives the mf of $F(x)$. This concept is illustrated in Figure 1 using a chain link analogy. In electronics safety analysis, for example, weakest-link failures are common in series components like shift register elements, and strongest-chain support strength can be illustrated by redundant processing components.

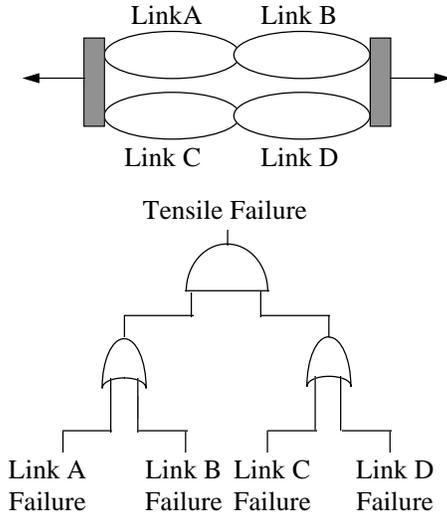


Figure 1. A Solution for Tensile Failure of Series-Parallel Chain Links

5. FUZZY “AND” TOOL

One way of expressing a crisp-logic “and” function is:

$$f = 1 \text{ iff } \sum_{i=1}^n x_i = n \quad (4)$$

where the x_i are inputs and f is the output. Since fuzzy numbers may approach one, a softer transition from 0 to one can be made using a “sigmoid” function [5]. For example:

$$f = \frac{x(1 - e^{-x/n})}{n(1 - 1/e)} \quad (5)$$

6. FUZZY “OR” TOOL

One way of expressing a crisp-logic “or” function is:

$$f = 1 \text{ iff } \sum_{i=1}^n x_i > 0 \quad (6)$$

A softer transition can be made using, for example:

$$f = \frac{1 - e^{-10x}}{1 - e^{-10n}} \quad (7)$$

7. LUKASIEWICZ NORMS

The combination and propagation of the membership functions through a fault tree can be accomplished by one of several prescribed normative functions, *norms*. In the mathematics of logic, functions called *T-norms* are used for the intersection of two or more events, while so-called *S-norms* are used for the union of two or more events. Common non-fuzzy, non-probabilistic T-norms can be found in the literature. The Lukasiewicz norm is:

$$T(x, y) = \max(0, x + y - 1) \quad (8)$$

S-norms (also called *T-conorms*) are used to define the union of two or more events. The Lukasiewicz norm is:

$$S(x, y) = \min(1, x + y) \quad (9)$$

The Lukasiewicz norms have differences in their capacities to consider other than the classical cases. The Lukasiewicz norms for the probabilities of and gates are closer to zero and the or gate probabilities closer to one than the probabilistic norms. This feature is especially critical in systems where the basic event probabilities are close to zero or one, as is also the case for probabilistic norms.

8. COMPOSITION OPERATOR

In addition to the T-norms and S-norms, the composition operator can be another mechanism to propagate uncertain probabilities through a logic tree. The symbolic form of the composition operation is:

$$A = B \circ R \quad (10)$$

where A is the result to be calculated at the next level in a tree, B is the value of the probability assimilated from among the basic events using either a logical intersection or logical union, and R is the form of inference used at a particular gate. The symbol “o” is the composition symbol. A list of the various composition operators is available in the literature [7].

The norms are used to aggregated probability information at the basic level in a logic tree. The propagation of this aggregated value is then propagated to the next higher level using a composition operator. This propagated value, however, will be associated with a confidence value that will come from the nature of the implication used. This idea is also powerful when an analyst wishes to model dynamic features or intra-component dependence issues with a composition.

9. WEIGHTED SUM TOOL

An interesting question in a safety assessment is how to provide metrics to measure a safety program against a set of goals, which include: 1) meet (or exceed) any quantitative safety requirements (e.g., $P(\text{disaster}) \ll 10^{-6}$ per credible situation), and 2) have in place sufficient

qualitative checks and balances to assure that safety is maintained for all operations, maintenance, testing, and changes that may occur during system life. For the quantitative measure, the ordinate associated with the uncertainty function can be thought of as probabilistic (e.g., a pdf) or “possibilistic” (e.g., level of presumption, or degree of possibility, or mf). For the qualitative measure, the abscissa value can be an expert engineering judgment of value, and the ordinate can be membership or possibility of the value.

The “weighted sum” tool is one way to approach the qualitative measure. For example, each user-chosen attribute can be judged by persons expert in the process to have value between 0 and 10 with a value whose abscissa and ordinate describe a fuzzy number (e.g., trapezoid, triangle, square function) with no abscissa values exceeding 10 or going below zero. The membership ordinate is between 0 and 1. The user-chosen weights could be point numbers that can be distributed among the attributes so that the sum is exactly 1. Another user input would be the “goal” for the overall value (e.g., “about 8”), which can be a subjective entry (trapezoid, triangle, square).

The test of the outcome is to compare (by fuzzy subtraction) the weighted sum with the goal. The resultant possibilistic function shows graphically how well the goal is met.

10. FUZZY SIMILARITY RELATIONS

Two new methods to propagate fuzzy information within a fault tree—a double composition method and a normative method—have been developed. We will concentrate here on an augmentation of the normative method. Suppose we have the yet-undefined composition:

$$B = A \circ R \quad (11)$$

where A can be an array of values for the basic elements of a particular intermediate event in a fault tree. For example, this array A could contain the point values used to estimate the probabilities of failure of each basic event, and the array could also contain some information describing the confidence the analyst has in these probability values. Without loss of generality,

the elements of this array A could also be fuzzy sets. The array B is also defined on probabilities, and this represents the set of results for the intermediate events in a logic tree. Generally, however, we do not know the contents of the array B, because this is a calculated quantity in the logic tree as we progress up the tree from basic events eventually to the top event in the tree. The elements of R represent the dependencies among the basic events, or components, contained in the intermediate event. In *active* safety systems, R can be formed from an extensive data base of failure statistics on hardware components, or data on human responses to well-defined situations.

On the other hand, for *passive* safety systems, R can be formed from first principles about basic physical relations or from human knowledge about the passive safety system.

What remains in the development of the proposed fuzzy methodology is the definition of what operations will be performed in the composition given as Equation (11). We introduce the notation,

$$B = \mathfrak{L} (A \circ R) = \mathfrak{L} (A') \quad (12)$$

where A' simply represents the composition operation, and where the symbol \mathfrak{L} is a two-dimensional operator which has the form,

$$\mathfrak{L} = \begin{bmatrix} L_1 \\ L_2 \end{bmatrix} \quad (13)$$

where L_1 is an implication operator used to propagate truth values through the fault tree, and L_2 is a T-norm or S-norm used to propagate probabilities through the fault tree. Both operators L_1 and L_2 can take on many different forms based on the logic in the gate in the fault tree. For example, if the fault tree is of the standard form (Boolean logic with probabilistic gates), then operator L_1 is a classical implication and operator L_2 is a probabilistic T-norm for an and gate and a probabilistic S-norm for an or gate within the fault tree.

In the implementation of this new fuzzy methodology, we define some constraints. The operator \mathfrak{L} in Equation (12) is a 2×1 matrix of operators (i.e., L_1 and L_2), the array A is a $2 \times n$ array of values, where the first row of A contains

the truth values for the basic elements in the intermediate event and the second row of A contains the probabilities of failure (or membership functions) for each of the basic elements, the relation R is an $n \times n$ array containing values which represent the degree of dependency between the basic elements, and n is the number of basic elements in a given intermediate event in the fault tree.

In illustrating this methodology, the following example problem is used. The fault tree in Figure 2 shows a simple generic model for safety where the basic events A, B, I, T, and F are symbols for: an accident, A, failure given the accident of three safety subsystems I, T, and F, or bypass of all three subsystems, B. We will show a procedure to assess the impact of mutual dependence, e.g., transcending the assumption that basic event F is also independent of the basic events I and T. The illustration of dependence makes use of a relational matrix, R, which contains the dependency information for the three basic events, I, T and F.

The development of the relational matrix, R, is approached from two perspectives. Each of these two perspectives illustrates a different way of formulating the dependence fuzzy relation, R, which is of central interest in this methodology. The first presumes that the components in the first intermediate event (IE1) represented in Figure 2 are pieces of an active safety system, and that an extensive data base of numerical information is available on the failure dependent statistics of the various components. In this first approach, we will use a numerical method called the cosine amplitude method to produce the elements of R from a numerical data set. The second approach presumes that the components I, T, and F shown in Figure 2 are components in a passive safety system, and that the content in R can be formulated from rules which describe the dependence relationship among the three basic events.

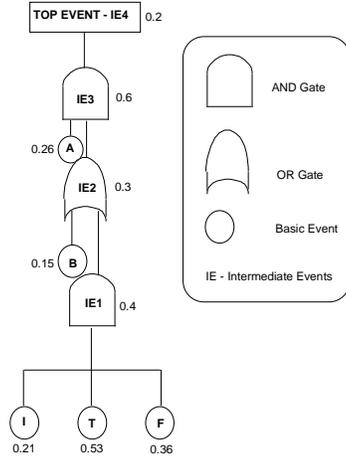


Fig. 2. A Schematic of the Example Fault Tree

11. ACTIVE SAFETY SYSTEM APPROACH

The method used here, for a situation where numerical failure information exists, is the cosine amplitude method [7]. As with all similarity relations, this similarity metric makes use of a collection of data samples; n data samples in particular. If these data samples are collected, they form a data array, X ,

$$X = \{x_1, x_2, \dots, x_n\} \quad (14)$$

Each of the elements, x_i , in the data array X is itself a vector of length m , that is

$$x_i = \{x_{i1}, x_{i2}, \dots, x_{im}\} \quad (15)$$

Hence, each of the data samples can be thought of as a point in m -dimensional space, where each point needs m coordinates for a complete description. Each element of a relation, r_{ij} , results through a pairwise comparison of two data samples, say x_i and x_j , where the strength of the relationship between data sample x_i and data sample x_j is given by the membership value expressing that strength, that is, $r_{ij} = \mu_R(x_i, x_j)$. The relation matrix will be of size $n \times n$ and, as will be the case for all similarity relations, the matrix will be reflexive and symmetric. The cosine amplitude method calculates r_{ij} in the following manner, and guarantees, as do all similarity methods, that $0 \leq r_{ij} \leq 1$.

$$r_{ij} = \frac{\left| \sum_{k=1}^m x_{ik} x_{jk} \right|}{\sqrt{\left(\sum_{k=1}^m x_{ik}^2 \right) \left(\sum_{k=1}^m x_{jk}^2 \right)}}, \text{ where}$$

$$i, j = 1, 2, \dots, n \quad (16)$$

A dependence relation, R (from a constructed example), is,

$$R = \begin{bmatrix} 1.00 & 0.50 & 0.87 \\ 0.50 & 1.00 & 0.50 \\ 0.87 & 0.50 & 1.00 \end{bmatrix} \quad (17)$$

As a point of comparison, if the three components are completely independent of one another, R becomes the identity matrix.

With the dependence relation, R , now developed, Equation 12 can be used to determine the truth values and probabilities propagated from the basic events I , T , and F up the fault tree in Figure 2 to the intermediate event $IE1$. If the composition in Equation 12 uses the max-min method, then the intermediate result, A' , is given as:

$$A' = \begin{bmatrix} 0.6 & 0.8 & 0.7 \\ 0.21 & 0.53 & 0.36 \end{bmatrix} \circ \begin{bmatrix} 1 & 0.50 & 0.87 \\ 0.50 & 1 & 0.50 \\ 0.87 & 0.50 & 1 \end{bmatrix} = \begin{bmatrix} 0.7 & 0.8 & 0.7 \\ 0.50 & 0.53 & 0.50 \end{bmatrix} \quad (18)$$

Now, the operators, \otimes , from Equation 13 can be performed on the matrix A' , with L_1 operating on the first row of A' and L_2 operating on the second row of A' . The first row (labeled 1) in Table 1, shows results in conducting this operation on the basic elements of intermediate event $IE1$. The various results are due to the use of different methods to conduct operations L_1 and L_2 on the basic elements I , T , and F , as detailed in the footnotes to the table. In addition, the propagation of probabilities and truth values up the entire fault tree (to the top event $IE4$) of Figure 2 is also shown in Table 1 as rows 2, 3 and *Top*. For these rows (2, 3 and *Top*) the calculus used is simply that due to a straightforward application of the methods detailed in the footnotes in the table; i.e., no subjectivity matrix is developed for the higher levels of the tree, and all components in the

upper levels of tree are presumed to be independent (e.g., in row 2 of the fault tree basic event B and intermediate event IE1 are independent). In each row in Table 1 the first number in the data-pairs is the probability of failure at that level of the tree, and the second number in the data-pairs is the truth value.

Table 1. Propagation of Probability and Truth

Level	Gate	L(P, T) ^a	P(P, T) ^b	F(P, T) ^c
1	AND	(0, 0.7)	(0.13, 0.4)	(0.5, 0.4)
2	OR	(0.3, 0.6)	(0.26, 0.7)	(0.15, 0.3)
3	AND	(0, 1)	(0.07, 0.6)	(0.15, 0.3)
TOP		(0, 0.2)	(0.07, 0.4)	(0.15, 0.2)

^a Lukasiewicz norm and implication

^b Probabilistic norm and classical implication

^c Zadeh's norm with Mamdani's implication operator.

12. PASSIVE SAFETY SYSTEM APPROACH

In the second approach, the elements of R are determined through rules that express dependence among the three components of the IE1 subsystem shown in Figure 2. Each of the rules will result in a relation using, for example, the Cartesian product (which uses the pairwise minimum operator on the membership values) between the antecedent and the consequent portions of the rules [7].

Now, in each matrix information relates the dependence between pairs of the components (i.e., between I and T, between T and F, and between I and F). But, for the methodology proposed here, we want to condense the information in each matrix to a single quantity that represents, in an average sense, the information in the matrix, using defuzzification. The defuzzification operation chosen is known as the centroid method. Each matrix can be thought of as a two-dimensional grid, where each Cartesian coordinate within the grid has a weight, which is the membership value. The centroid of this grid represents the weighted average of the entire matrix. We can determine the coordinates of the centroid and the weight (maximum

membership value) at this coordinate. These values for each of the three relational matrices are, for the example:

Coordinates for I - T: (0.25, 0.76);

$$\text{Max } \mu_{I-T} = 0.6$$

Coordinates for T - F: (0.82, 0.58);

$$\text{Max } \mu_{T-F} = 0.6$$

Coordinates for F - I: (0.34, 0.61);

$$\text{Max } \mu_{F-I} = 0.5$$

Hence, if these dependence values are arranged in a matrix, R, the resulting dependency matrix for the three components, I, T, and F, becomes:

$$R = \begin{matrix} & \begin{matrix} I & T & F \end{matrix} \\ \begin{matrix} I \\ T \\ F \end{matrix} & \begin{bmatrix} 1 & 0.6 & 0.5 \\ 0.6 & 1 & 0.6 \\ 0.5 & 0.6 & 1 \end{bmatrix} \end{matrix} \quad (19)$$

This subjectivity matrix, R, is now used in Equation (3) with a max-min form of composition to find the intermediate result, A'.

$$A' = \begin{bmatrix} 0.6 & 0.8 & 0.7 \\ 0.21 & 0.53 & 0.36 \end{bmatrix} \circ \begin{bmatrix} 1 & 0.6 & 0.5 \\ 0.6 & 1 & 0.6 \\ 0.5 & 0.6 & 1 \end{bmatrix} = \begin{bmatrix} 0.5 & 0.8 & 0.7 \\ 0.53 & 0.53 & 0.53 \end{bmatrix} \quad (20)$$

As before in the active safety system example, the operators, £, from Equation (4) can be performed on the matrix A', with L₁ operating on the first row of A' and L₂ operating on the second row of A'. Also, as before, the propagation of probabilities and truth values up the entire fault tree of Figure 2 is shown in Table 2 as rows 2, 3 and Top, where all components in the upper levels of tree are presumed to be independent.

Table 2. Propagation of Probability and Truth

Level	Gate	L(P, T) ^a	P(P, T) ^b	F(P, T) ^c
1	AND	(0, 0.09)	(0.15, 0.5)	(0.53, 0.4)
2	OR	(0.3, 0.6)	(0.28, 0.7)	(0.15, 0.3)
3	AND	(0, 1)	(0.07, 0.6)	(0.15, 0.3)
TOP		(0, 0.2)	(0.07, 0.4)	(0.15, 0.2)

^a Lukasiewicz norm and implication

^b Probabilistic norm and classical implication

^c Zadeh's norm with Mamdani's implication operator.

13. USE OF CONSTRAINED MATHEMATICS

Unconstrained application of the basic rules of uncertainty mathematics, e.g., probabilistic calculus, interval analysis, and fuzzy mathematics, can be inaccurate [Ref. 8]. The effect is that probability distributions, interval bounds, or fuzzy (bounded) numbers can be generated that exceed more accurate, narrower bounds, unless mathematical constraints are enforced. This problem is most easily solved in Monte Carlo or Latin Hypercube Sampling routines by sampling only once for a variable in a logic expression, rather than re-sampling for each appearance of the variable. The situation is more difficult in interval-based analysis techniques, as we will illustrate below, but still solvable.

In order to perform correct constrained mathematics calculations, it is also important that approximations, which are useful in efficient probabilistic evaluation of logical combinations of events, be avoided. We will briefly examine these approximations, in order to set the stage for an accurate constrained mathematical approach.

The deductive or inductive logic for outcomes developed using fault trees and event trees are typically expressed as Boolean algebra equations. When these equations are written disjunctively, they are said to express a logical union of terms that can be either "events" (occurrences of safety importance) or "cutsets" (logical intersections of events). For the following discussion, we will first treat the variables as independent. Then the question of this (difficult to satisfy) condition will be revisited. The calculation of the probability of the "top event" described by the Boolean expression can be computed from the probabilities of the individual events in several ways, and all of these have been used in software products.

1. The "rare event approximation" is performed by computing the probabilistic sum of each "cutset" of a Boolean sum. This approach is only correct if all cutsets are disjoint (an extremely rare situation). It is a

useful approximation only if all of the cutsets have very low probabilities.

2. A canonical Boolean "union of minterms" ("truth-table"-rows for which the function is satisfied) expression appears attractive for probabilistic evaluation, because all of the terms are disjoint, and therefore the probabilities of the events in each minterm can be multiplied to produce a minterm probability, and the minterm probabilities can be directly added. For relatively complex problems, this approach is impractical, because an n -variable problem has 2^n potential minterms. For example, a problem having 100 variables could have an astronomical 10^{30} terms to track.

3. A conventional algorithm for the probability of the union of Boolean sets can be exemplified by the simplest form of the method (for two terms), which is: $P(A \cup B) = P(A) + P(B) - P(A \cap B)$ ¹. If the terms are independent, the expression becomes:

$$P(A \cup B) = P(A) + P(B) - P(A) \times P(B).$$

Additional independent sets can be successively brought into the union, e.g.,: $P(A \cup B \cup C) = P(A \cup B) + P(C) - P(A \cup B) \times P(C)$

. The ability to combine successive inputs with prior results makes this algorithm look attractive for software implementation. However, practical computations seldom involve unions of independent terms. One reason is subtle inherent dependence between the processes leading to events, but a more salient factor is dependence due to variables appearing in multiple disjunctive terms. A simple example is $y = x_1 x_2 \cup x_2 x_3$ (where juxtaposition indicates intersection). For this expression,

$$P(y) = P(x_1) \times P(x_2) + P(x_2) \times P(x_3) - P(x_1) \times P(x_2) \times P(x_3)$$

. The methodology described above, if the dependence were not recognized, would give the incorrect result

$$P(y) = P(x_1) \times P(x_2) + P(x_2) \times P(x_3) - P(x_1) \times P(x_2) \times P(x_3)$$

. A straightforward approach to solve these types of problems correctly using the methodology described above is to methodically track all repetitions of variables. This is generally comparable in complexity to the methodology described in part 2 above.

¹ The notation convention is that \cup represents logical union, \cap represents logical intersection, and + and - indicate ordinary addition and subtraction.

4. An accurate and potentially efficient disjoint set algorithm based on successive application of “Shannon’s decomposition” is called the “Sigma Pi” algorithm [Ref. 9]. It generally allows one to efficiently transform the original Boolean equation into a disjoint set Boolean expression. In some situations, the Sigma Pi algorithm is difficult to implement. Recursive operations are required in order to expand and recombine data, and a decision must be made about whether or not to use “pattern recognition” to eliminate recalculation of identical subfunctions.

5. A relatively new, accurate, and generally efficient disjoint set algorithm is based on a segmented form of Boolean logic processing [Ref. 10]. This algorithm was used to derive the probability results presented in this paper.

An example illustrating the necessity for constrained mathematics is the union of two independent events for which the probabilities are specified by intervals, $P(A) = [0.1, 0.8]^2$ and $P(B) = [0.2, 0.9]$. Using unconstrained operations of interval analysis, one obtains $P(A \cup B) = [-0.42, 1.68]$, both of which are of course impossible for probabilities. The problem is that both interval bounds combine an operand’s lower bound together with its upper bound, which is not possible for a single event. Restricting events to have only one value at a time, one obtains the correct answer, which is $P(A \cup B) = [0.28, 0.98]$.

Although constrained mathematics in general requires a complex algorithm, there are situations in probabilistic evaluation of logic expressions for which it can be done in a straightforward manner. Linking the results in a computer parsing routine is important for software implementation, and this too can be done very efficiently for many functions. Demonstrating these assertions is a major purpose of this paper. One important situation, which we address here, involves the probability evaluation of a Boolean function that is unate or has unate variables. Boolean functions are logical descriptions that can be applied to describe the outcomes of fault trees and event trees, among many other safety analysis applications.

² The first member of the ordered pair is by convention the lower bound; the second is the upper bound.

Unateness [Ref. 11] means that every variable of a Boolean function can be expressed such that each variable appears either complemented or uncomplemented, but both senses are not necessary. Any variable that meets this condition is called a unate variable (positive unate if uncomplemented, negative unate if complemented). Unateness is especially important in logical trees, because an event tree or fault tree having only “ands” and “ors” such that each event affects the probabilistic outcome either positively everywhere it appears or negatively everywhere it appears can be represented by a unate Boolean logic expression. This tree condition is sufficient for unateness.

An algorithm that accounts for constrained mathematics in an expression of probability for the Boolean function outcome is:

For a positive unate variable, the lower bound of the result is a function of the lower bound of each appearance of the variable, and the upper bound of the result is a function of the upper bound of each appearance of the variable. For a negative unate variable, the lower bound of the result is a function of the upper bound of each appearance of the variable, and the upper bound of the result is a function of the lower bound of each appearance of the variable.

This is formalized in the following treatment.

For a Boolean function, $y = f(x_1, x_2, \dots, x_n)$, a variable x_i is positive unate iff

$$y_{i1} = f(x_1, x_2, \dots, x_i = 1, \dots, x_n) \supset f(x_1, x_2, \dots, x_i = 0, \dots, x_n) = y_{i0}$$

The variable x_i is negative unate iff

$$y_{i1} = f(x_1, x_2, \dots, x_i = 1, \dots, x_n) \subset f(x_1, x_2, \dots, x_i = 0, \dots, x_n) = y_{i0}$$

If

$$y_{i1} = f(x_1, x_2, \dots, x_i = 1, \dots, x_n) = f(x_1, x_2, \dots, x_i = 0, \dots, x_n) = y_{i0}$$

x_i is a redundant variable and need not appear in the expression for the function.

For the probability, $P(y)$, that the Boolean function is satisfied, all uncertain variables, x_i , are constrained to have identical values within their uncertainty range for all appearances of the variable in an algebraic expression for $P(y)$. This is true whether the uncertainty is represented by a probability density function, a fuzzy number, an interval, or any other practically meaningful uncertainty measure. Where lower and upper bounds are involved

(e.g., interval analysis and fuzzy mathematics) the bound must be the same for each appearance of the variable in the expression for $P(y)$.

Theorem 1:

a) *The lower bound of $P(y)$ is a function of the lower bound of positive unate variables and the upper bound of $P(y)$ is a function of the upper bound of positive unate variables; b) the upper bound of $P(y)$ is a function of the lower bound of negative unate variables and the lower bound of $P(y)$ is a function of the upper bound of negative unate variables; c) there is no contribution by a variable that is redundant.*

Proof:

a) Consider the equivalent expression for $y = f(x_1, x_2, \dots, x_n) = \sum m_j$, where the sum is a Boolean sum of “minterms,” the minterms m_j are Boolean conjunctions of all the variables, each uncomplemented or complemented (which is analogous to rows in a “truth table”), and the number of minterms in the Boolean expression depends on the number of truth table values for which the function is satisfied. Terming a positive unate variable x_i , there must be some $m(x_i=1)$, due to the positive unateness. If there are any $m(x_i=0)$, there is a corresponding (other variables in identical senses) $m(x_i=1)$ because of the containment relation such that the expression for the minterm is identical except for x_i . The probability contributions of these two minterms are $p_a p(x_i)$ and $p_a [1 - p(x_i)]$, respectively, where p_a is the aggregate of the probabilities of all the other variables (which are common to the two minterms). Since the sum of these two terms is p_a (as we know it must be because of the Boolean identity $x_m x_n + x_m \bar{x}_n = x_m$), there is no contribution from the negative probability, although it may be included in the probability expression because of the inclusion of $m(x_i=0)$. Since the only role of the negative quantity is to cancel its positive counterpart, there can be no inversion of the limits (i.e., the lower bound of the positive unate variable contributes to the lower bound of the result, and the upper bound of the positive unate variable contributes to the upper bound of the result).

b) There must be some $m(x_i=0)$, due to the negative unateness. For each $m(x_i=1)$, if such a

term exists, there is a corresponding (other variables in identical senses) $m(x_i=0)$ because of the containment relation. The probability contributions of these two minterms are $p_a [1 - p(x_i)]$ and $p_a p(x_i)$, respectively, where p_a is the aggregate of the probabilities of all the other variables (which are common to the two minterms). Since the sum of these two terms is p_a (as we know it must be because of the Boolean identity $x_m x_n + x_m \bar{x}_n = x_m$), there is no contribution from the positive probability, although it must be included because of the inclusion of $m(x_i=1)$. Since the only role of the positive quantity is to cancel its negative counterpart, there is inversion of the limits (i.e., the lower bound of the negative unate variable contributes to the upper bound of the result, and the upper bound of the negative unate variable contributes to the lower bound of the result).

c) Since $x_m x_n + x_m \bar{x}_n = x_m$, x_n is redundant and therefore is not needed in the computation of the expression.

This result naturally extends from independent variables to independent functions, as we will demonstrate in a subsequent example. Once unate variables have been processed, the solution for any non-unate variables can be traditional, but greatly simplified because of the removal of unate variables from the problem. Parsing for computer solution involves first determining the unate variables and their bounds, and then calculating the bounds for non-unate variables based on the bounds of the unate variables. Finally, all variable bounds are combined to solve for the bounds of the result. The concepts in the theorem, the processing of non-unate variables, and the parsing order will be illustrated through examples.

Example 1:

Consider the Boolean function (of three independent variables) $y_1 = x_1 \bar{x}_2 \cup \bar{x}_2 x_3$ ³. The first and third variables are positive unate; the second is negative unate. An expression

³ The convention used in these examples is that where the \cup symbol appears, the disjunction is a logical “or,” and the juxtaposition indicates logical intersection; where the $+$ symbol appears, the disjunction is ordinary addition, and the juxtaposition indicates ordinary multiplication. We also work these examples implicitly assuming that all of the variables in the function are independent.

(methodology of Ref. 3) for the probability of y_1 is $P(y_1) = P(x_1)P(\bar{x}_2) + P(\bar{x}_1)P(\bar{x}_2)P(x_3)$. According to the theorem, the constrained mathematics treatment for any expression of the function using lower and upper bounds (e.g., fuzzy mathematics or interval analysis) is:

$$P(y_1)_l = P(x_1)_l P(\bar{x}_2)_l + P(\bar{x}_1)_u P(\bar{x}_2)_l P(x_3)_l$$

$$P(y_1)_u = P(x_1)_u P(\bar{x}_2)_u + P(\bar{x}_1)_l P(\bar{x}_2)_u P(x_3)_u$$

If $P(x_1)$, $P(x_2)$, and $P(x_3)$ are interval numbers $[0.2, 0.4]$, $[0.7, 0.9]$, and $[0.4, 0.6]$ respectively, the result is $P(y_1)=[0.052, 0.228]$.

Example 2:

For the function $y_2 = x_1\bar{x}_3 \cup x_2x_4 \cup \bar{x}_2\bar{x}_3$, x_1 and x_4 are positive unate, x_3 is negative unate, and x_2 is not unate. The probability expression is: $P(y_2) = P(x_2)P(x_4) + P(\bar{x}_2)P(\bar{x}_3) + P(x_1)P(x_2)P(\bar{x}_3)P(\bar{x}_4)$. The bounds solution directly implements the bounds for x_1 , x_3 , and x_4 as: $P(y_2)_l = P(x_2)P(x_4)_l + P(\bar{x}_2)P(\bar{x}_3)_l$ and $+ P(x_1)_l P(x_2)_l P(\bar{x}_3)_l P(\bar{x}_4)_l$ and $P(y_2)_u = P(x_2)P(x_4)_u + P(\bar{x}_2)P(\bar{x}_3)_u$. The $+ P(x_1)_u P(x_2)_u P(\bar{x}_3)_u P(\bar{x}_4)_u$

solution for $P(x_2)$ depends on the sign of the function of the other variables that are produced with $P(x_2)$. Taking a partial derivative with respect to $P(x_2)$:

$$\frac{\partial P(y)}{\partial P(x_2)} = P(x_4) - P(\bar{x}_3) + P(x_1)P(\bar{x}_3)P(\bar{x}_4)$$

Then if $P(x_4)_l - P(\bar{x}_3)_l \geq 0$, $P(x_2)_l$ can be used in the computation for $P(y_2)_l$. If $P(x_4)_l - P(\bar{x}_3)_l \leq 0$, $P(x_2)_u$ can be used in the computation for $P(y_2)_l$. If $P(x_4)_u - P(\bar{x}_3)_u \geq 0$, $P(x_2)_u$ can be used in the computation for $P(y_2)_u$. If $P(x_4)_u - P(\bar{x}_3)_u \leq 0$, $P(x_2)_l$ can be used in the computation for $P(y_2)_u$.

For $P(x_1) = [0.1, 0.3]$, $P(x_2) = [0.7, 0.9]$, $P(x_3) = [0.4, 0.6]$, and $P(x_4) = [0.6, 0.8]$, $P(y_2) = [0.5512, 0.8124]$

Many forms of Boolean expressions that do not meet any unateness criteria can be processed almost as efficiently as those described above. In order to demonstrate this, we will address the "exclusive-or" function (satisfied if an odd number of inputs are satisfied) and its inverse (satisfied if an even number of inputs are satisfied). These have in many respects

characteristics completely opposite to unateness (there are no pairs in exclusive-or functions (or their inverses) such that $m(x_i=0) = m(x_i=1)$ in the terminology of Theorem 1). Since these functions are linear and associative, they can be computed iteratively, which simplifies algorithmic implementation. A general theorem will help illustrate these concepts.

Theorem 2:

For an exclusive-or of n Boolean variables (or Boolean functions), consider two of the variables (or functions), w and z , where w and z can each represent either some x_i or \bar{x}_i , a) if $P(w)_u \leq 0.5$, and $P(z)_u \leq 0.5$, $P(y)_l$ is a function of $P(w)_l$ and $P(z)_l$; and $P(y)_u$ is a function of $P(w)_u$ and $P(z)_u$. b) If $P(w)_u \leq 0.5$, and $P(z)_l \geq 0.5$, $P(y)_l$ is a function of $P(w)_u$ and $P(z)_l$; and $P(y)_u$ is a function of $P(w)_l$ and $P(z)_u$. c) If $P(w)_l \geq 0.5$, and $P(z)_u \leq 0.5$, $P(y)_l$ is a function of $P(w)_l$ and $P(z)_u$; and $P(y)_u$ is a function of $P(w)_u$ and $P(z)_l$. d) If $P(w)_l \geq 0.5$, and $P(z)_l \geq 0.5$, $P(y)_l$ is a function of $P(w)_u$ and $P(z)_u$; and $P(y)_u$ is a function of $P(w)_u$ and $P(z)_u$. e) $P(w)_u \leq 0.5$, $P(z)_l \leq 0.5$, and $P(z)_u \geq 0.5$, $P(y)_l$ is a function of $P(w)_l$ and $P(z)_l$; and $P(y)_u$ is a function of $P(w)_l$ and $P(z)_u$. f) If $P(z)_u \leq 0.5$, $P(w)_l \leq 0.5$, and $P(w)_u \geq 0.5$, $P(y)_l$ is a function of $P(w)_l$ and $P(z)_l$; and $P(y)_u$ is a function of $P(w)_u$ and $P(z)_l$. g) If $P(w)_l \geq 0.5$, $P(z)_l \leq 0.5$ and $P(z)_u \geq 0.5$, $P(y)_l$ is a function of $P(w)_u$ and $P(z)_u$; and $P(y)_u$ is a function of $P(w)_u$ and $P(z)_l$. h) If $P(z)_l \geq 0.5$, $P(w)_l \leq 0.5$ and $P(w)_u \geq 0.5$, $P(y)_l$ is a function of $P(w)_u$ and $P(z)_u$; and $P(y)_u$ is a function of $P(w)_l$ and $P(z)_u$. i) Furthermore, if $P(w)_l \leq 0.5$, $P(z)_l \leq 0.5$, $P(w)_u \geq 0.5$, and $P(z)_u \geq 0.5$, $P(y)_l$ is either a function of $P(w)_l$ and $P(z)_l$ or $P(w)_u$ and $P(z)_u$; and $P(y)_u$ is either a function of $P(w)_l$ and $P(z)_u$ or $P(w)_u$ and $P(z)_l$.

If more than one condition is met, either of the indicated options corresponding to the satisfied conditions can be used. Although this theorem appears cumbersome, it is merely a cataloging of all of the possible combinations, which is straightforward to implement algorithmically. The ease of use will be illustrated by the examples following the proof.

Proof:

Since $P(y) = P(w) + P(z) - 2P(w) \times P(z)$, we can derive $\frac{\partial P(y)}{\partial P(w)} = 1 - 2P(z)$ and

$\frac{\partial P(y)}{\partial P(z)} = 1 - 2P(w)$. In the first case, if

$P(z) \leq 0.5$, the derivative is everywhere nonnegative, so higher values of w lead to higher (or equal) values of $P(y)$ and vice versa. If $P(z) \geq 0.5$, the derivative is everywhere nonpositive, so lower values of w lead to higher (or equal) values of $P(y)$ and vice versa. The situation is identical for the variable z as indicated by the second partial derivative. These conclusions taken in the combinations indicated correspond to the theorem parts a through h. For part i, we apply the partial derivatives in pairs to show that if $P(w) \leq 0.5$ and $P(z) \leq 0.5$, decreasing both w and z leads to lower (or equal) values of $P(y)$. If $P(w) \geq 0.5$ and $P(z) \geq 0.5$, increasing both w and z leads to lower (or equal) values of $P(y)$. If $P(w) \leq 0.5$ and $P(z) \geq 0.5$, decreasing w and increasing z leads to higher (or equal) values of $P(y)$. If $P(w) \geq 0.5$ and $P(z) \leq 0.5$, increasing w and decreasing z leads to higher (or equal) values of $P(y)$.

Example 3:

Consider $y_3 = x_1 \oplus \bar{x}_2 \oplus x_3$, where $P(x_1) = [0.1, 0.2]$, $P(x_2) = [0.6, 0.8]$, and $P(x_3) = [0.6, 0.7]$. First, compute $P(z) = P(x_1 \oplus \bar{x}_2)$. Since both operands are everywhere less than 0.5, the lower bound of $P(z)$ is a function of the lower bounds of $P(x_1)$ and $P(\bar{x}_2)$, and the upper bound of $P(z)$ is a function of the upper bounds of $P(x_1)$ and $P(\bar{x}_2)$. We compute $P(z) = [0.26, 0.44]$. Combining this with $P(x_3)$ (operands both greater than 0.5), $P(y)_l$ is a function of $P(z)_u$ and $P(x_3)_l$, and $P(y)_u$ is a function of $P(z)_l$ and $P(x_3)_u$. The result is $P(y) = [0.512, 0.596]$.

Example 4:

Consider $y_4 = x_1 \oplus x_2 \oplus \bar{x}_3 \oplus x_4$, where $P(x_1) = [0.1, 0.2]$, $P(x_2) = [0.3, 0.6]$, $P(x_3) = [0.2, 0.6]$, and $P(x_4) = [0.6, 0.7]$. First (arbitrarily), compute $P(z_1) = P(x_1 \oplus x_4)$. Since

$P(x_1)_u \leq 0.5$ and $P(x_4)_l \geq 0.5$, $P(z_1)_l$ is a function of $P(x_1)_u$ and $P(x_4)_l$, and $P(z_1)_u$ is a function of $P(x_1)_l$ and $P(x_4)_u$. The first

intermediate result is: $P(z_1) = [0.56, 0.66]$.

Next, compute $P(z_2) = P(z_1 \oplus x_2)$. Since $P(z_1)_l \geq 0.5$, and $P(x_2)$ is not constrained to either less than or equal to 0.5 or greater than or equal to 0.5, $P(z_2)_l$ is a function of $P(z_1)_u$ and $P(x_2)_u$, and $P(z_2)_u$ is a function of $P(z_1)_u$ and $P(x_2)_l$. The second intermediate result is: $P(z_2) = [0.468, 0.564]$. Since neither this result or that of $P(\bar{x}_3)$ is constrained to either less than or equal to 0.5 or greater than or equal to 0.5, we search two choices for both $P(y)_l$ and $P(y)_u$. The final result is: $P(y) = [0.4616, 0.5192]$.

When a function contains an independent exclusive-or subfunction (i.e., the variable in the exclusive-or are not necessary anywhere else in the function), the subfunction can be tested for unateness, and Theorems 1 and 2 can be combined to solve for the bounds in the exclusive-ored variables.

Example 5:

$y_5 = x_1 \bar{x}_2 x_3 x_4 \cup x_1 \bar{x}_2 x_3 \bar{x}_5 \cup x_1 \bar{x}_2 \bar{x}_4 \bar{x}_5$, where $\cup \bar{x}_1 x_2 x_3 x_4 \cup \bar{x}_1 x_2 x_3 \bar{x}_5 \cup \bar{x}_1 x_2 \bar{x}_4 \bar{x}_5$, $x_1 = [0.6, 0.7]$, $x_2 = [0.4, 0.6]$, $x_3 = [0.3, 0.5]$, $x_4 = [0.5, 0.6]$, and $x_5 = [0.1, 0.3]$. The probability is: $P(y) = P(x_1)P(\bar{x}_2)P(x_3)P(x_4) + P(x_1)P(\bar{x}_2)P(\bar{x}_4)P(\bar{x}_5) + P(\bar{x}_1)P(x_2)P(x_3)P(x_4) + P(\bar{x}_1)P(x_2)P(\bar{x}_4)P(\bar{x}_5)$

. When the Boolean function is simplified to $y_5 = x_1 \bar{x}_2 x_3 x_4 \cup x_1 \bar{x}_2 \bar{x}_4 \bar{x}_5 \cup \bar{x}_1 x_2 x_3 x_4 \cup \bar{x}_1 x_2 \bar{x}_4 \bar{x}_5$, it is apparent that x_3 is positive unate, and x_5 is negative unate. When the methodology of Ref. 5 is applied to derive $y_5 = (x_1 \oplus x_2)x_3x_4 + (x_1 \oplus x_2)\bar{x}_4\bar{x}_5$, it can be seen that $x_1 \oplus x_2$ serves the role of an independent unate subfunction. Using the information in the theorems, we can now determine that $P(y)_l$ is a function of $P(x_1)_u$, $P(x_2)_u$, $P(x_3)_l$, and $P(x_5)_u$; and $P(y)_u$ is a function of $P(x_1)_u$, $P(x_2)_l$, $P(x_3)_u$, and $P(x_5)_l$. This leaves only the functionality of the bounds of x_4 to determine. Taking the partial derivative, $\frac{\partial P(y)}{\partial P(x_4)} = P(x_1 \oplus x_2)[P(x_3) - P(\bar{x}_5)]$. Since this derivative is everywhere negative, $P(y)_l$ is a function of $P(x_4)_u$, and $P(y)_u$ is a function of $P(x_4)_l$. The final result is: $P(y) = [0.2116, 0.378]$.

Example 6:

$y_6 = x_1x_2 \cup \bar{x}_1x_3 \cup \bar{x}_1\bar{x}_2$, where $P(x_1) = [0.2, 0.4]$, $P(x_2) = [0.7, 0.8]$, and $P(x_3) = [0.6, 0.8]$. Since x_3 is positive unate, $P(y_6)_l$ is a function of $P(x_3)_l$ and $P(y_6)_u$ is a function of $P(x_3)_u$. Exclusive-or functionality is revealed by re-writing the function [Ref. 3] and its probability as:

$$y_6 = (\bar{x}_1 \oplus x_2) \cup \bar{x}_1x_3 \quad \text{and}$$

$$P(y_6) = P(\bar{x}_1 \oplus x_2) + P(\bar{x}_1)P(x_2)P(x_3),$$

but the exclusive-or function is not independent of the second term. Taking the partial derivatives with respect to x_1 and x_2 :

$$\frac{\partial P(y_6)}{\partial P(x_1)} = P(x_2) - P(\bar{x}_2) - P(x_2)P(x_3) \quad \text{and}$$

$$\frac{\partial P(y_6)}{\partial P(x_2)} = P(x_1) - P(\bar{x}_1) + P(\bar{x}_1)P(x_3).$$

For the

lower bound of $P(y_6)$, using the lower bound of $P(x_3)$, we find that both partial derivatives can be either positive or negative, but that the result must either include both $P(x_1)_l$ and $P(x_2)_u$ or $P(x_1)_u$ and $P(x_2)_l$. The lowest result is obtained using the first pair. For the upper bound of $P(y_6)$, using the upper bound of $P(x_3)$, we find that the first partial derivative is everywhere negative, and the second partial derivative is everywhere positive. This results in $P(y_6) = [0.704, 0.832]$.

The methodology outlined in this section is easily implemented in software, as we have partially done in our COSMET fuzzy mathematics routines [Ref. 10], and the results obtained are far superior to unconstrained operations. One other note of interest is that independent events are frequently assumed, but the assumption is seldom realistic. Some software packages attempt to compensate for this weakness by allowing correlation to be specified. We prefer subjective pairwise dependence measures introduced within the theoretical Frechet dependence bounds, which is also included in the COSMET software [Ref. 10].

14. SOFTWARE

There are five tools (features) that have been implemented in software; also implemented is a mode of combining arbitrarily chosen constituent tools. The software package under development is named COSMET (Coordinated

Objective/Subjective Mathematically Enhanced Tools). Four of the tools will be described here.

The "CONTEST" entry specifies the beginning of a multi-line contest definition statement. This allows the user to determine the possibility of a weaklink device surviving a stronglink device. A Contest tool plot is shown in Figure 3.

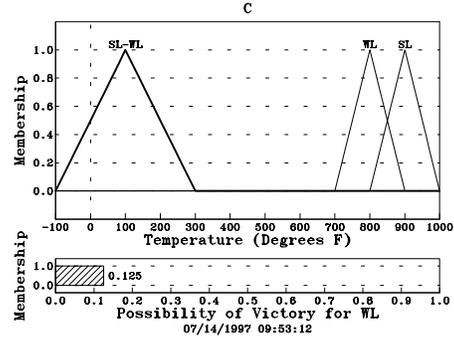


Figure 3. Contest Tool Plot

The "WEIGHTED-SUM" entry specifies the beginning of a multi-line weighted sum definition statement. The weighted sum tool allows the user to qualitatively determine the possibility of achieving a goal that is determined by the combination of two or more related properties. An example generated by this tool is illustrated in Figure 4.

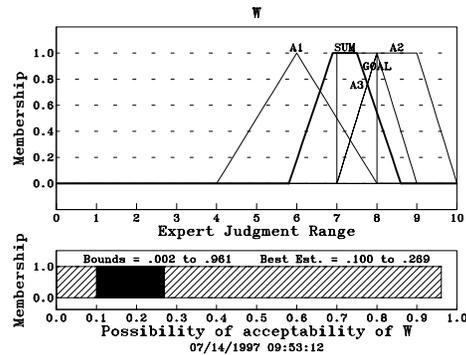


Figure 4. Weighted-Sum Example Problem

The "E-MINMAX" designation specifies the beginning of a multi-line extreme Min/Max definition statement. The extreme Min/Max tool allows the user to determine and display the extremes of the top event of a fault tree by considering the "incredible" inputs. Figure 5 shows the results of an example problem.

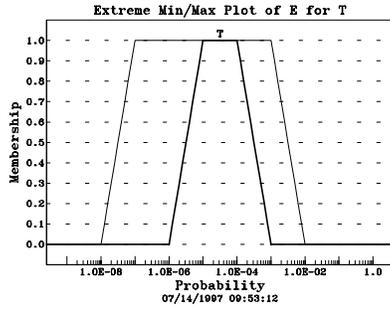


Figure 5. Extreme Minmax Example Problem

The “L-MINMAX” entry specifies the beginning of a multi-line link Min/Max definition statement. The link Min/Max tool allows the user to determine and display the result of two or more responses that are combined according a user-defined equation. An example problem is shown in Figure 6.

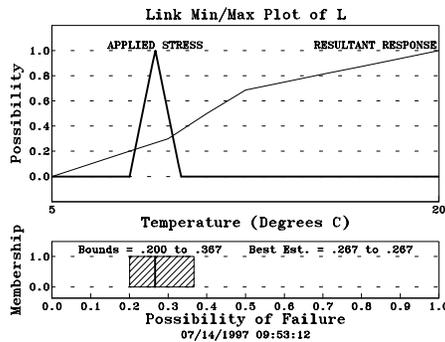


Figure 6. Link Minmax Example Problem

CONCLUSIONS

A general logic paradigm is now possible, which implements general gate logics, which propagates basic event probabilities and basic event truth values, and which incorporates both random and ambiguous information in determining event failure dependencies. A special case of the general paradigm is the classical probabilistic logic analysis with the assumption of independent basic events. The incentive for developing the proposed approach comes from the need to conduct more realistic analysis on safety systems where basic event failure probabilities are formed with substantial contributions from subjective information or general information about first-principle laws of physics.

The general paradigm proposed will accommodate both active and passive safety system analysis. This structure has been generalized to contain various logic norms in the modeling of event intersections or unions. The inference that is implied at every gate in the tree can be represented by any of a number of implication procedures. The conventional logic approach involving probabilistic “and” and “or” norms and the classical implication as the inference mechanism is a *special case* of the proposed general paradigm. The proposed method can be quite powerful, because it provides a framework where numerical failure information can be implemented along with cognitive knowledge of a non-numeric form, for example rule-based information about passive safety features. Moreover, the proposed paradigm is able to model event dependencies implicitly within the implication choices available to the analyst.

REFERENCES

1. Kaufmann, A. (1975) *Introduction to Fuzzy Subsets*, Vol. 1, Academic Press, New York.
2. Cai, K. (1996) Special Issue on Fuzzy Methodology in System Failure, *Fuzzy Sets and Systems*, North-Holland, Vol. 83 (2).
3. Cooper, J. A., “Fuzzy-Algebra Uncertainty Analysis for Abnormal-Environment Safety Assessment,” *Journal of Intelligent and Fuzzy Systems*, Vol. 2, No. 4, 1994.
4. Cooper, J. A., S. Ferson, and L. Ginzburg, “Hybrid Processing of Stochastic and Subjective Uncertainty Data,” *Risk Analysis*, Vol. 16, No. 6, December, 1996.
5. Yager, R. (1988) "On Ordered Weighted Averaging Aggregation Operators in Multicriteria Decisionmaking", *IEEE Transactions Systems, Man, Cybernetics*, Vol 18, No. 1, 183-190.
6. Yager, R., and J. Kapryk, Eds., *The Ordered Weighted Averaging Operators-- Theory and Applications*, Kluwer Academic Publishers, 1997.

7. Ross, T. (1995) *Fuzzy Logic with Engineering Applications*, McGraw-Hill, New York.
8. Klir, G. J. and J. A. Cooper, "On Constrained Fuzzy Arithmetic," *Proceedings of the Fifth IEEE International Conference on Fuzzy Systems*, 1996, pp. 1693-1699.
9. Heger, A. S., J. K. Bhat, D. W. Stack, and D. V. Talbott, "Calculating Exact Top-Event Probabilities Using $\Sigma\Pi$ -Patrec.," *Reliability Engineering and System Safety*, 50 (1995) 253-259.
10. Cooper, J. A., "Theoretical Description of Methodology in PHASER (Probabilistic Hybrid Analytical System Evaluation Routine)," Sandia National Laboratories Report SAND 96-0022, January, 1996.
11. Lewis, R. M. II, and C. L. Coates, *Threshold Logic*, John Wiley & Sons, Inc., 1967.
12. Cooper, J. A., "Orthogonal Expansion Applied to the Design of Threshold-Element Networks," Stanford University Dissertation and Stanford Electronics Laboratories Technical Report 6204-1, SEL-63-123, December, 1963.

Distribution:

C. E. Meyers, 4523 MS 0188
D. D. Carlson, 12333 MS 0405
D. E. Bennett, 12333 MS 0405
R. J. Breeding, 12333 MS 0405
M. A. Dvorack, 12333 MS 0405
M. K. Fuentes, 12333 MS 0405
T. R. Jones, 12333 MS 0405
S. A. Kalemba, 12333 MS 0405
Y. T. Lin, 12333 MS 0405
W. McCulloch, 12333 MS 0405
K. J. Maloney, 12333 MS 0405
K. B. Sobolik, 12333 MS 0405
R. G. Easterling, 5412 MS 0419
D. M. Kunsman, 5417 MS 0423
A. C. Payne, Jr., 5415 MS 0425
W. R. Reynolds, 2103 MS 0427
K. Ortiz, 2167 MS 0481
L. R. Gilliom, 6546, MS 0451
B. K. Cloer, 6531 MS 0451
S. D. Spray, 12331 MS 0490
J. A. Cooper, 12331 MS 0490 (50)
J. P. Hoffman, Jr. , 12331 MS 0490
P. E. D'Antonio, 12324 MS 0491
M. Caldwell, 12324 MS 0491
J. M. Covan, 12324 MS 0491
M. E. Ekman, 12324 MS 0491
D. Isbell, 12324 MS 0491
R. D. Pedersen, 12331 MS 0490
R. E. Smith, 12302 MS 0491
J. L. Tenney, 12333 MS 0491
P. W. Werner, 12324 MS 0491
G. A. Sanders, 5100 MS 0492
J. P. Cates, 12332 MS 0492
D. R. Lewis, 12332 MS 0492
D. H. Loescher, 12332 MS 0492
W. E. Mauldin, 12332 MS 0492
D. R. Olson, 12332 MS 0492
C. G. Shirley, 12332 MS 0492
D. A. Summers, 12332 MS 0492
J. F. Wolcott, 12332 MS 0492
Review and Approval Desk, 12690
for DOE/OSTI MS0619 (2)
J. V. Hancock, 12324 MS 0491

C. A. Trauth, Jr. , 12304 MS 0434
G. C. Novotny, 12334 MS 0434
E. L. Fronczak, 12334 MS 0434
S. B. Humbert, 12334 MS 0434
J. E. Stayton, 12334 MS 0434
F. G. Trussell, 12334 MS 0434
W. C. Nickell, 12300 MS 0428
T. S. Edrington, 12301 MS 0428
P. N. Demmie, 12324 MS 0491
N. R. Ortiz, 6400 MS 0736
D. L. Berry, 6403 MS 0744
R. M. Cranwell, 6613 MS 0746
D. G. Robinson, 6613 MS 0746
A. L. Camp, 6412 MS 0747
S. L. Daniel, 6412 MS 0747
G. D. Wyss, 6412 MS 0747
P. E. Rexroth, 5822 MS 0761
D. E. Ellis, 5500 MS 0766
F. T. Harper, 5514 MS 0767
R. M. Jansma, 6513 MS 0449
K. V. Diegert, 12323 MS 0829
J. M. Sjulín, 12335 MS 0830
Technical Library, 4916 MS 0899 (5)
A. O. Bendure, 7523 MS 0369
R. E. Bair, 1200 MS 0511
Central Technical Files, 8940-2 MS 9018
J. T. Ringland, 8112 MS 9201
J. J. Cashen, 8116 MS 9202

Dr. Scott Ferson
Applied Biomathematics
100 North Country Road
Setauket, NY 11733

Prof. Lev Ginzburg
Ecology and Evolution
State University of New York
Stony Brook, NY 11794