University of Michigan School of Public Health
The University of Michigan Department of Biostatistics Working Paper Series

Year 2003 *Paper* 19

A Fully Bayesian Approach for Combining Multilevel Failure Information in Fault Tree Quantification and Corresponding Optimal Resource Allocation

M Hamada[*] H. F. Martz[†] C S. Reese[‡]
T. Graves[**] Valen Johnson[††] A. G. Wilson[‡‡]

[*]Los Alamos National Laboratory
[†]Los Alamos National Laboratory
[‡]Bigham Young University
[**]Los Alamos National Laboratory
[††]University of Michigan School of Public Health, valenj@umich.edu
[‡‡]Los Alamos National Laboratory

A Fully Bayesian Approach for Combining Multilevel Failure Information in Fault Tree Quantification and Corresponding Optimal Resource Allocation

M Hamada, H. F. Martz, C S. Reese, T. Graves, Valen Johnson, and A. G. Wilson

Abstract

This paper presents a fully Bayesian approach that simultaneously combines basic event and statistically independent higher event-level failure data in fault tree quantification. Such higher-level data could correspond to train, sub-system or system failure events. The full Bayesian approach also allows the highest-level data that are usually available for existing facilities to be automatically propagated to lower levels. A simple example illustrates the proposed approach. The optimal allocation of resources for collecting additional data from a choice of different level events is also presented. The optimization is achieved using a genetic algorithm.

A Fully Bayesian Approach for Combining Multilevel Failure Information in Fault Tree Quantification and Corresponding Optimal Resource Allocation

M. Hamada, H. F. Martz
Group D-1, MS F600, Los Alamos National Laboratory, Los Alamos, NM 87545

C. S. Reese
Brigham Young University

T. Graves
Los Alamos National Laboratory

V. Johnson
University of Michigan

A. G. Wilson
Los Alamos National Laboratory

September 11, 2003

Abstract

This paper presents a fully Bayesian approach that simultaneously combines basic event and statistically independent higher event-level failure data in fault tree quantification. Such higher-level data could correspond to train, sub-system or system failure events. The full Bayesian approach also allows the highest-level data that are usually available for existing facilities to be automatically propagated to lower levels. A simple example illustrates the proposed approach. The optimal allocation of resources for collecting additional data from a choice of different level events is also presented. The optimization is achieved using a genetic algorithm.

1. Introduction

Vesely et al. [1], the probabilistic risk assessment (PRA) procedures guide (Hickman [2]), and many other textbooks discuss fault tree quantification. Such quantification consists of three steps: (1) determining the basic event probabilities, (2) calculating the minimal cut set probabilities, and (3) determining the system (i.e., the top event) probability using either exact or approximate methods.

It is current and accepted practice in fault tree and accident sequence quantification (as implemented, for example, in the Systems Analysis Programs for Hands-on Integrated Reliability Evaluations [SAPHIRE (Russell et al. [3]) package and the Integrated Reliability and Risk Analysis System [IRRAS [Russell et al. [4]; Vanhorn et al. [5]])]) to use only statistical data and information regarding the basic events. Martz and Almond [6] directly use independent statistical data and information corresponding to higher-level events or gates in the tree. However, normal operation and testing procedures often generate data for many high-level gates corresponding to such events as train, subsystem, and system unavailability, and often even the top event itself. In quantifying the accident sequence frequency for a proposed accident of interest at an existing facility, independent statistical data almost always exist at the highest level; namely, x occurrences of the accident (where x is usually 0) in a given exposure time t or in n demands.

By “independent” we mean that the higher-level data for a system are *not* simultaneously providing collateral information on the basic events comprising that system (which would lead to double counting and thus dependency). In other words, we assume that the higher-level and any basic-event data are *not* the result of the same set of demands or observation period. This is usually the case for any system test that is destructive, such as a missile fired at a target. If the same higher-level data provide basic event-level information, then we can

instead use such data to verify the structure of the fault tree. In particular, any higher-level failure data which is not predicted by the fault tree is an indication that the fault tree model is inadequate.

This paper describes a fully Bayesian approach which can simultaneously combine basic event and independent higher-level failure data and information in fault tree quantification. The obvious advantage is the associated increase in accuracy and precision of the probabilistic results because of the combined use of these data. Note that Martz and Almond [6] only approximates the fully Bayesian approach by utilizing the first two distributional moments.

The fully Bayesian approach can also incorporate independent industry-wide statistical analyses that are sometimes performed on safety systems considered in a PRA. Such analyses represent a source of generic higher-level statistical information for the specific plant under consideration. For example, Grant et al. [7] describe an industry-wide statistical analysis of the safety-related performance of the high-pressure coolant injection (HPCI) system at US commercial boiling water reactor plants for the period 1987-1993.

1.1 Related methods

A number of articles discuss system reliability for systems described by reliability block diagrams which combine both component and independent system-level test data. Mastran [8] and Mastran and Singpurwalla [9] consider an approximate Bayesian approach to the estimation of system reliability in which there exist pass/fail test data at both the component and system for a coherent system of nonidentical components. They use a top-down approach which apportions the posterior system reliability distribution to each component in the form of a component prior distribution consistent with the system configuration. Combining these component priors with the component level data produces component

posterior distributions. Propagating these component posteriors back up to the system level using the system model forms the final system posterior from which the desired inferences are obtained.

Martz, Waller, and Fickas [10] and Martz and Waller [11] develop an approximate Bayesian procedure for estimating system reliability based on a bottom-up approach in which only the means and variances of prior distributions are used, which are then combined with data, and finally propagated upward.

Johnson et al. [12] propose a fully Bayesian approach for system reliability as described by a reliability block diagram. This fully Bayesian approach resolves the upward and downward propagation problem by simultaneously modeling the complete set of system parameters. We generalize their procedure in this paper to fault tree quantification.

When both levels of data exist for the same demands or observation period, the above methods are inapplicable because the data are dependent. For example, a standby system may fail to operate upon demand (a higher-level system failure) which may subsequently be traced to the failure of a particular component in the system (a basic event-level failure). However, the above methods (and the method presented here as well) are still applicable if only one level of data is used. Using the data at the higher-level gate to form an aggregated posterior for the higher-level event produces an *aggregate* analysis. Using the data at the basic event-level to form a disaggregated posterior for the higher-level event produces a *disaggregate* analysis. Usually, the aggregate and disaggregate posteriors will disagree, in which case we say that an *aggregation error* occurs. Very large aggregation errors are grounds for suspicion of the structure of the fault tree model.

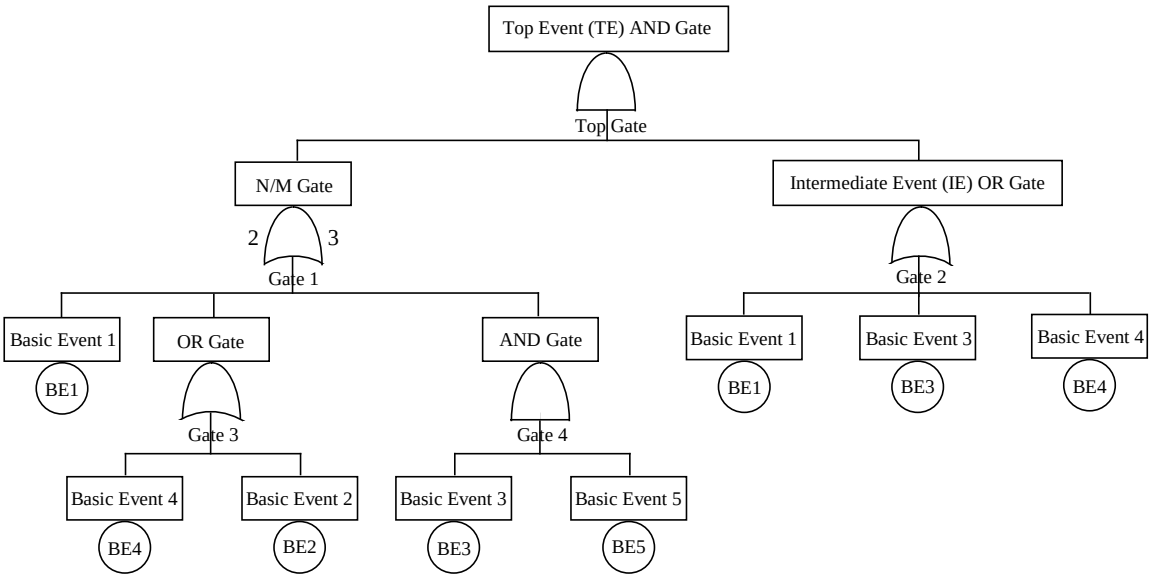
An outline of the paper is as follows. In Section 2, to focus attention, we present an example fault tree. A Bayesian approach for using independent

higher-level failure data in any coherent fault tree is presented in Section 3. Section 4 illustrates the performance of the proposed approach using the fault tree example. Section 5 discusses the problem of allocating additional resources to improve inference of the top event probability. Section 6 concludes with a discussion.

2. Example

Before presenting the full Bayesian approach, consider the following simple fault tree example as depicted in Figure 1 that was used to illustrate the IRRAS fault tree solution and quantification in Appendix A of Russell et al. [3,4]; it consists of AND and OR gates and one 2/3 gate. There are five basic events denoted by BE1 to BE5. One intermediate event denoted by IE is identified as well as the top event TE. Note the difference between a fault tree and a reliability block diagram in which, for our example, a basic event such as BE1 shows up in more than one branch of the fault tree. In this paper, we consider the situation where prior information and/or data are available at the basic, intermediate and top events.

Figure 1: Example Fault Tree



3. A fully Bayesian approach for inference

We assume that the information about the probability of occurrence of each basic event can be summarized using a probability distribution. The proposed method will even properly handle *a priori* state-of-knowledge (SOK) dependence among the basic events; all that is necessary is the specification of their joint distribution. In this paper, we describe prior information for a basic event probability in the form of a beta distribution denoted by Beta(*a,b*). If there are also basic event data available in the form of *x* failures, say, in *n* trials, then the posterior information for the basic event (combining the prior information and data via Bayes' theorem) can be expressed as Beta(*a+x,b+(n-x)*). Thus, we assume that the information available, both prior and data, for a basic event can be described by a beta distribution.

The proposed method also requires that the higher-level event information be expressed as data so that the posterior distribution of the basic event

probability obtained from using multilevel data and information is well defined. Thus, we express the higher-level event information as x failures in n trials, although x and n need not be integers. For example, suppose that we believe a higher-level event probability is 0.05 with only 2 observations; thus, we would set $x = 0.1$ and $n = 2$.

Following Johnson et al. [12], a key feature of the proposed method is that higher-level event probabilities are expressed in terms of basic event probabilities. For fault trees, these expressions can be obtained by determining the higher-level event's minimal cut sets and using the law of total probability also known as the inclusion-exclusion rule. For the example in Figure 1, the top event has five minimal cut sets as follows:

{BE1,BE2}, {BE1,BE4}, {BE1,BE3,BE5}, {BE2,BE3,BE5}, {BE3,BE4,BE5}.

Using the law of total probability, the top event probability expressed in terms of the basic events is

$$\begin{aligned}TE(\mathbf{p}) = {} & p1*p2 + p1*p4 + p1*p3*p5 + p2*p3*p5 + p3*p4*p5 - \\ & p1*p2*p4 - 3*p1*p2*p3*p5 - 3*p1*p3*p4*p5 - p2*p3*p4*p5 \\ & +4*p1*p2*p3*p4*p5,\end{aligned}$$

where p1, …, p5 are the occurrence probabilities for basic events BE1, …, BE5. Similarly, the intermediate event probability can be expressed as

$$IE(\mathbf{p}) = p1 + p3 + p4 - p1*p3 - p1*p4 - p3*p4 + p1*p3*p4.$$

From these expressions of higher-level event probabilities, we see that higher-level event information provides information about basic event probabilities.

Likewise, basic event information provides information about higher-level event probabilities.

As mentioned in the introduction, we rely on a Bayesian approach to inference for the modeling proposed above. Bayesian methods are named for Bayes' theorem

$$(p \mid x) \quad \frac{f(x \mid p) \quad (p)}{f(x \mid p) \quad (p)dp}, \tag{1}$$

where $\pi(p \mid x)$ is called the posterior distribution, and is the conditional distribution of the unknown failure probability p given the observed data. Furthermore, $f(x \mid p)$ is the sampling density (commonly referred to as the likelihood) and (p) represents the prior distribution for p. This prior distribution can be obtained from experts, computer models, engineering or physics theory, or previous studies. If there is no information about p before a study is conducted, a distribution which contains little or no information about p can be substituted, often referred to as a noninformative prior distribution. In our experience, there almost always exists some prior knowledge that can and should be incorporated.

Bayesian methods were relegated to obscurity for a long period of statistical history. The primary reason was that the denominator in (1) was difficult (and sometimes impossible) to calculate. However, Gelfand and Smith [13] introduced computing routines that made computation of the denominator possible through simulation and Monte Carlo integration; Casella and George [14] and Chib and Greenberg [15] provide good introductions to these computing routines. The broad class of modern Bayesian computation was aptly named Markov chain Monte Carlo (MCMC) and Gilks et al. [16] provide a nice review of

the basic elements of MCMC computation. At the heart of most basic Bayesian computation is the complete conditional (or full conditional) distribution which is defined as the conditional distribution of each parameter given all other parameters in the model, and the data. MCMC relies on the fact that sequential simulation from complete conditionals (replacing recently updated parameters successively) converges to the joint posterior distribution of all of the parameters. So, given a starting point, after a certain number of iterations (called the burn-in) the simulated observations will be from the desired joint posterior distribution. Often, simulation from a complete conditional is difficult (or seemingly impossible). The Metropolis-Hastings algorithm (Chib and Greenberg [15]) is a method for simulating from an arbitrary distribution whose form is known up to a constant (as is the case with Bayesian computation). The central idea is that a random variable is generated from any distribution with probability density function $g(\bullet)$, and is accepted with probability

$$\min(1, \frac{g(z|y)h(y)}{g(y|z)h(z)}),$$

where z is the current value of the parameter (say, p) and y is the proposed replacement value of the parameter; here $h(\bullet)$ is probability density function (up to a constant) of the desired arbitrary distribution. As the algorithm proceeds, this distribution converges to the distribution of the actual complete conditional.

This is an amazing result and makes Bayesian computation available for a rich class of problems. One obvious consequence of the above choice is that the realizations will not be independent, but will almost certainly exhibit autocorrelation. In order to remedy this problem, it is often recommended that realizations be skipped and only every third observation, for example, be kept for inference. This process of dropping observations to approximate independence is called “thinning”.

In our case inference is then obtained using Bayes' theorem implemented by Markov chain Monte Carlo (MCMC); that is, we end up with a set of draws from the joint posterior distribution of the basic event probabilities **p**.

The advantage of this fully Bayesian approach is that no approximations are being made (except for the Monte Carlo sampling error which is controlled by taking more samples). The top event posterior distribution is based on all available data and the basic event posterior distributions are updated based on all higher-level data. We will apply the proposed procedure for the simple fault tree example under different scenarios in the next section.

4. Example revisited

We consider several cases to examine the performance of the method as a function of two factors: the strength of the basic event-level data (strong or weak), the strength of the top event-level data (strong or weak). The results for each of these cases are compared and used as a means of assessing the performance of the proposed approach.

In the following tables, BE, TE and IE refer to basic, top and intermediate events, respectively. First the cases considered can be classified by whether there is any information for the events or, in the case of information, whether it is weak or strong. Table 1 describes the cases in these terms.

Table 1: Various cases in terms of none, weak and strong information available

Case	BE1-BE5	TE	IE
1	weak	weak	none
2	weak	strong	none
3	strong	weak	none

4	strong	strong	none
5	none	strong	none
6	strong	none	none
7	none	weak	none
8	weak	none	none
9	weak	weak	weak
10	weak	weak	strong

Recall that the basic event information is described by Beta(*a,b*) and that higher-level event information is described by the equivalent number of event occurrences *x* out of *n* trials. Table 2 describes the 10 cases in these terms.

Table 2: Various cases in terms of beta parameters (*a,b*) and data*(x,n*)

Case	BE1	BE2	BE3	BE4	BE5	TE	IE
1	0.152	0.142	0.129	0.118	0.106	0.163	
	15.092	6.899	4.176	2.821	2.012	162.923	
2	0.152	0.142	0.129	0.118	0.106	5.141	
	15.092	6.899	4.176	2.821	2.012	5140.881	
3	5.086	5.024	4.963	4.901	4.840	0.163	
	503.470	246.180	160.458	117.627	91.954	162.923	
4	5.086	5.024	4.963	4.901	4.840	5.141	
	503.470	246.180	160.458	117.627	91.954	5140.881	
5	0.500	0.500	0.500	0.500	0.500	5.141	
	0.500	0.500	0.500	0.500	0.500	5140.881	

6	5.086	5.024	4.963	4.901	4.840		
	503.470	246.180	160.458	117.627	91.954		
7	0.500	0.500	0.500	0.500	0.500	0.163	
	0.500	0.500	0.500	0.500	0.500	162.923	
8	0.152	0.142	0.129	0.118	0.106		
	15.092	6.899	4.176	2.821	2.012		
9	0.152	0.142	0.129	0.118	0.106	0.163	0.152
	15.092	6.899	4.176	2.821	2.012	162.923	15.244
10	0.152	0.142	0.129	0.118	0.106	0.163	5.086
	15.092	6.899	4.176	2.821	2.012	162.923	508.556

The results for each of the above outlined ten cases are presented in Figure 2. For each case, the posterior distribution summaries of the BE2, BE4, IE and TE events are plotted; the 2.5, 50 and 97.5 percentiles are indicated by red, black and blue lines, respectively. Each plot indicates the effect of including various strengths of data. For example, compare the width of the posterior 95% credible intervals for case 5 (strong TE data) versus case 7 (weak TE data) in which the stronger data have the predictable effect of reducing variability; the same holds for case 6 (strong BE1-5 data) versus case 8 (weak BE1-5 data). Note that having weak BE1-5 data is different than having no BE1-5 (represented by Beta(0.5,0.5)); contrast case 2 with case 5 and case 1 with case 7. The effect of adding weak TE data can depend on the type of BE1-5 data; when there are strong BE1-5 data, there is little effect (see cases 3 and 6). However, when weak TE data is added to weak BE1-5 data, the variability of the BE2, BE4 and IE events has actually increased; here the weak TE data do not exactly reinforce the BE1-5 data so that the resulting posterior from the combined data is wider. Also compare case 1 with case 9 and case 9 with case 10 in which wider posteriors arise when weak IE data or different strong IE data are added. Other examples of the patterns observed above can be seen in cases 1-4 in which weak and strong BE1-5 data and weak and strong TE data are considered in the four possible combinations. Thus, we see that the effect is very different

depending on which level is being examined and what type of data is available at each level. This demonstrates the value of collecting different information at different levels which will be further illustrated with an application of a genetic algorithm for optimizing additional data collection based on an overall budget constraint in the next section.

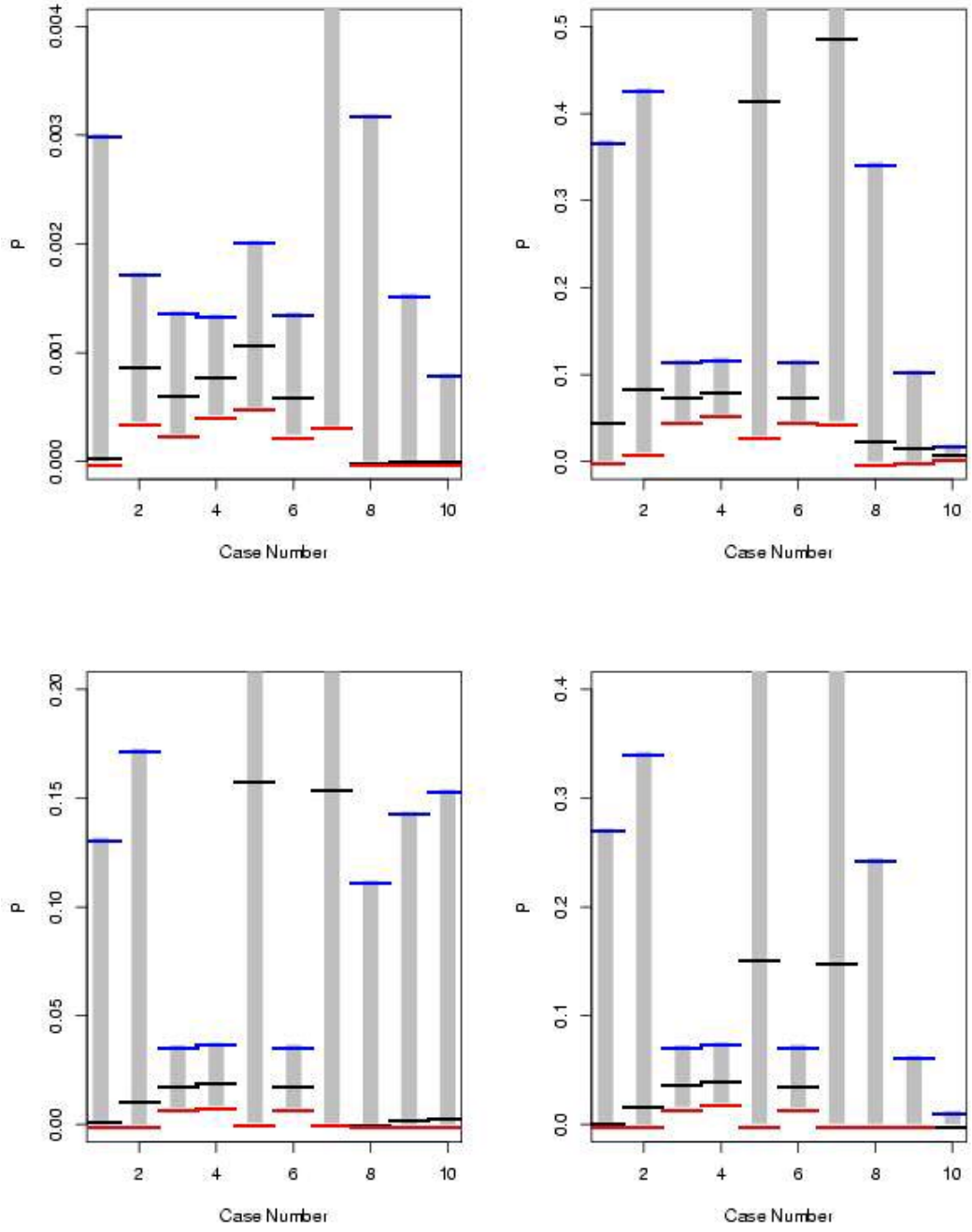


Figure 2: Posterior 95% credible intervals for the probability of events for each of the ten cases. The upper left panel is for the top event (TE) probability, the upper right is for the intermediate event (IE) probability, the lower left is for the basic event BE2 probability, and the lower right is for the basic event BE4 probability. The red line indicates the 2.5 percentile, the black line represents the posterior

median, and the blue line represents the 97.5 percentile. Note that the very long intervals extend beyond the graphs.

5. Optimal resource allocation

In this section, we consider the optimal allocation of additional tests for a fixed budget. In our example, how many tests should be done for each of the seven events? First, we specify a criterion that reflects the information gain in doing additional tests and use a genetic algorithm (GA) (Goldberg [17], Michalewicz [18]) to find the optimal allocation that maximizes the information gain.

We assume that there is a cost for collecting additional event data and that higher-level event data are more costly than basic event data. Consider the following costs as an example which are the costs of a single observation (event):

BE1:	$1
BE2:	$1
BE3:	$1
BE4:	$1
BE5:	$1
TE:	$10
IE:	$3

We define the maximum information gain in terms of the maximum reduction in uncertainty of the top event probability. That is, we consider the maximum reduction in the relative length of the central 90% credible interval from the top event posterior distribution before and after taking additional data. Note that this interval itself has a distribution and we are concerned with the ratio of

the "after" new data and "before" new data posterior lengths. Here we take the 0.75 quantile of this distribution as the criterion we want to minimize.

Briefly, we describe how a GA can be used to find a nearly optimal allocation. A GA operates on a "population" of candidate "solutions" to the optimization problem. Here, each solution is a string of seven sample sizes corresponding to additional tests to be done regarding events BE1-BE5, TE and IE, respectively.

First we construct an initial population of *M* solutions by randomly generating solutions that do not exceed the given fixed budget. Subsequent populations of solutions are obtained by using the genetic operators of crossover and mutation which we will describe next.

The information gain criterion for each of the solutions in the initial population is evaluated and the solutions are ranked from smallest to largest, i.e., the smallest ratio is the best solution in the initial population.

The second (and subsequent) GA generations are now populated using the two genetic operations: crossover and mutation. First consider genetic crossover. Two parent solutions are randomly selected without replacement from the initial population with probability inversely proportional to their rank among the *M* solutions. A new solution is obtained from the parent solutions by randomly picking one of the two parents and taking its sample size and doing this for each of the seven events. The two parents are then returned to the initial population before the next crossover operation is performed. In this way, an additional *M* solutions are constructed using the crossover operator. Note that the solutions are checked to make sure they do not exceed the budget, so that solutions are generated until there are *M* such feasible solutions. The information gain criterion is also evaluated for each of these new solutions.

For each of the initial M solutions, we next apply genetic mutation to each of the event sample sizes. We also incorporate relaxation in the probability that mutation occurs as a function of generation.

It is desired to mutate each factor value with probability that decays exponentially as a function of generation. That is, mutations become less and less likely as the number of generations increases. To accomplish this, at generation g each event sample size is mutated with probability $\exp(-\mu \times g)$ where μ is a user-specified mutation rate parameter.

Given that mutation of an event sample size occurs, we then mutate the value with expectation approximately equal to the current value of the factor and variance that decreases with g. We accomplish this by means of a logit transformation as follows: first compute $z = (y - L) / (U - L)$ where y, L, and U are the current, minimum and maximum sample sizes; L=0, U=floor(budget/cost of event), where *floor* is the largest integer not exceeding its argument. Then calculate $d = \log[z / (1 - z)] + [\text{Uniform}(0, 1) - .5] \times \sigma \times \exp(-\mu \times g)$, where Uniform(0,1) denotes a random draw from a uniform distribution. Here σ is a user-specified parameter that controls the rate at which the variance decreases as a function of g. Finally, compute $u = L + (U + 1 - L) \times \exp(d) / [1 + \exp(d)]$ and the desired mutated sample size is floor(u) which is between L and U. This logit transformation has the properties that the expected value is approximately equal to the current sample size y and the standard deviation decreases with g. Following this mutation procedure, we generate solutions until M additional solutions which do not exceed the budget are obtained. Then the information gain criterion is evaluated for each of these additional M solutions.

We use an “elitist” GA, which means that the population in the next generation consists of the M best solutions from the $3M$ solutions currently being

considered (*M* initial solutions, *M* crossover solutions and *M* mutated solutions). We execute the above GA for *G* generations.

For the allocation problem we consider the optimal allocation for budgets of $100 and $250. For the GA, we use populations of size 25 (*M*=25) and generate 100 generations (*G*=100). We considered case 8 from the previous section in which there were no data at the intermediate and top events. The length of the 90% credible interval for top event probability based on the existing data is 0.00318. To reduce the computational effort 500 draws from the joint posterior distribution of the seven event probabilities based on the current information are taken. For each draw, then numbers of events occurring are drawn from binomial distributions using these event probabilities for the proposed sample sizes. Then the resulting posterior is calculated using MCMC; to keep the computational requirements to a manageable size, we burn-in with 500 draws and then compute the 90% central credible interval for the top event probability based on the next 1000 draws with no thinning. Thus, there are 500 relative lengths (ratio of the new to the old interval). The information gain criterion is taken to be the 0.75 quantile or the 125^{th} largest relative length out of the 500

Now suppose that the budget is $100. What resource allocation yields the most reduction in the 90% credible interval length of the top event probability? Based on a GA as described above, the GA produced the traces presented in Figures 3 and 4 which display the criterion and number of tests allocated, respectively. The information gain criterion starts at 0.69 in generation 1 and decreases to 0.50 in generation 100 with an allocation of 54, 11, 5, 22 and 6 new tests to basic events BE1-BE5, respectively, and no allocation to either of the higher-level events.

Figures 5 and 6 provide the GA trace for the criterion and number of tests allocated for a budget of $250. The information gain criterion starts at 0.45 in generation 1 and decreases to 0.31 in generation 100 with an allocation of 151, 31, 21, 37 and 10 tests to basic events BE1-BE5, respectively, and no allocation to the higher-level events. That is, for the cost structure considered here, the entire test budget is allocated to the basic events.

Figure 3: GA Criterion Trace for $100 Budget

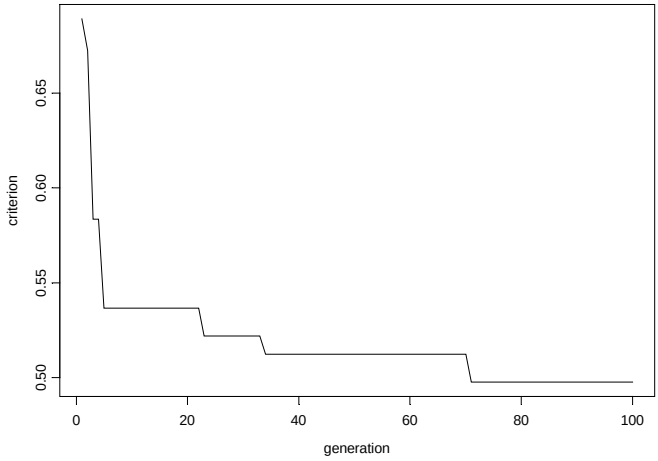


Figure 4: GA Number of Tests Allocation Trace for $100 Budget

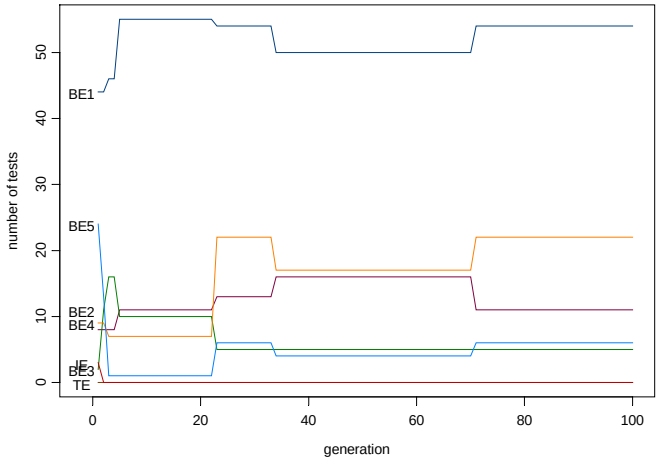


Figure 5: GA Criterion Trace for $250 Budget

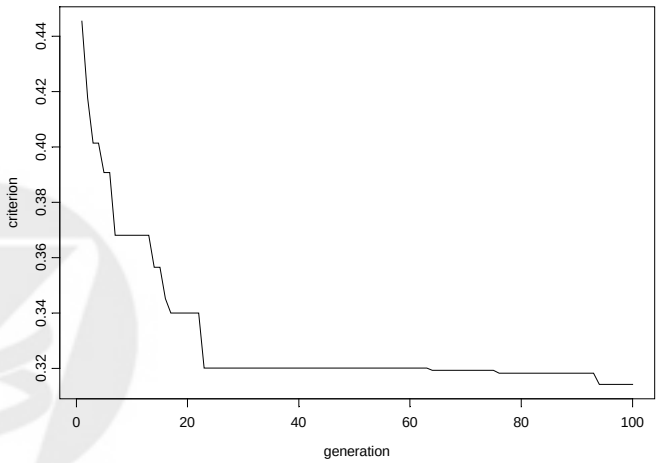
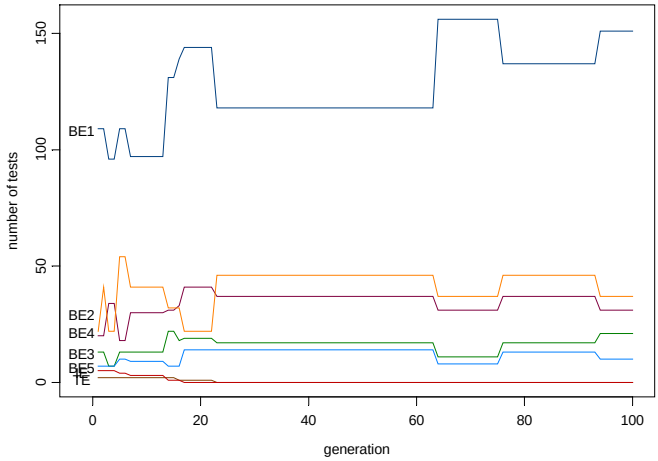


Figure 6: GA Number of Tests Allocation Trace for $250 Budget



6. DISCUSSION

A fully Bayesian methodology has been developed for using multilevel event data in fault tree quantification. The method requires the identification and use of SOK uncertainty distributions for the probabilities of occurrence of the initial basic events. The higher-level event information must be expressed as data. The performance of the methodology was illustrated for a simple example and performs as expected. The combined use of higher-level data is particularly advantageous when the initial basic event data are weak.

Now that multilevel fault tree data can be simultaneously considered and analyzed, the question of how to allocate additional test resources across the fault tree events can be addressed. That is, for a given budget, the allocation

providing the most gain in information can be determined. We demonstrated how a genetic algorithm provides a practical way to accomplish this.

Thus, the fully Bayesian approach is very attractive and easy to use for fault tree analysis. It can naturally handle data at different event levels. Moreover, allocation of additional resources can easily be accomplished.

ACKNOWLEDGMENTS

The authors thank Dee Won for her encouragement of this work.

REFERENCES

[1] Vesely, W. E., Goldberg, F. F., Roberts, N. H., & Haasl, D. F., Fault tree handbook. NUREG-0492, January 1981.

[2] Hickman, J. W., PRA procedures guide: A guide to the performance of probabilistic risk assessments for nuclear power plants, Vol. 1. American Nuclear Society and Institute of Electrical and Electronic Engineers, NUREG/CR-2300, January 1983.

[3] Russell, K. D., Atwood, C. L., Galyean, W. J., Sattison, M. B., & Rasmuson, D. M., Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE) version 5.0, vol. 1 - technical reference manual. EGG-2716 (NUREG/CR-6116), Idaho National Engineering Laboratory, July 1994.

[4] Russell, K. D., Kvarfordt, K. J., Skinner, N. L., Wood, S. T., & Rasmuson, D. M., Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE) version 5.0, vol. 2 - Integrated Reliability and Risk

Analysis System (IRRAS) reference manual. EGG-2716 (NUREG/CR-6116), Idaho National Engineering Laboratory, July 1994.

[5] VanHorn, R. L., Russell, K. D., & Skinner, N. L., Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE) version 5.0, vol. 3 - Integrated Reliability and Risk Analysis System (IRRAS) tutorial manual. EGG-2716 (NUREG/CR-6116), Idaho National Engineering Laboratory, July 1994.

[6] Martz, H. F. & Almond, R. G., Using higher-level failure data in fault tree quantification, Reliab Eng & Syst Safe 1997; 56: 29-42.

[7] Grant, G. M. Roesener, W. S., Hall, D. G., Atwood, C. L., Gentillon, C. D., & Wolf, T. R., High-pressure coolant injection (HPCI) system performance, 1987-1993. INEL-94/0158, Idaho National Engineering Laboratory, February 1995.

[8] Mastran, D. V., Incorporating component and system test data into the same assessment: A Bayesian approach, Op Res 1976; 24: 491-499.

[9] Mastran, D. V. & Singpurwalla, N. D., A Bayesian estimation of the reliability of coherent structures, Op Res 1978; 26: 663-672.

[10] Martz, H. F., Waller, R. A., & Fickas, E. T., Bayesian reliability analysis of series systems of binomial subsystems and components, Technometrics 1988; 30: 143-154.

[11] Martz, H. F. & Waller, R. A., Bayesian reliability analysis of complex series/parallel systems of binomial subsystems and components, Technometrics 1990; 32: 407-416.

[12] Johnson, V., Graves, T. Hamada, M., & Reese, C. S., A hierarchical model for estimating the reliability of complex systems, Proceedings of the 2002 Valencia Meeting, Valencia, Spain, 2003.

[13] Gelfand, A. E. & Smith, A. F. M., Sampling-based approaches to calculating marginal densities, J of the Am Stat Assoc 1990; 85: 398-409.

[14] Casella, G. & George, E. I., Explaining the Gibbs sampler, The Am Statistician 1992; 46; 167-174.

[15] Chib, S. & Greenberg, E., Understanding the Metropolis-Hastings algorithm, The Am Statistician 1995; 49: 327-335.

[16] Gilks, W. R., Richardson, S. & Spiegelhalter, D. J., Markov chain Monte Carlo in practice. London: Chapman & Hall, 1996.

[17] Goldberg, D. E., Genetic algorithms in search, optimization and machine learning. New York: Addison-Wesley, 1989.

[18] Michalewicz, Z., Genetic algorithms + data structures = evolution programs, New York: Springer-Verlag, 1992.