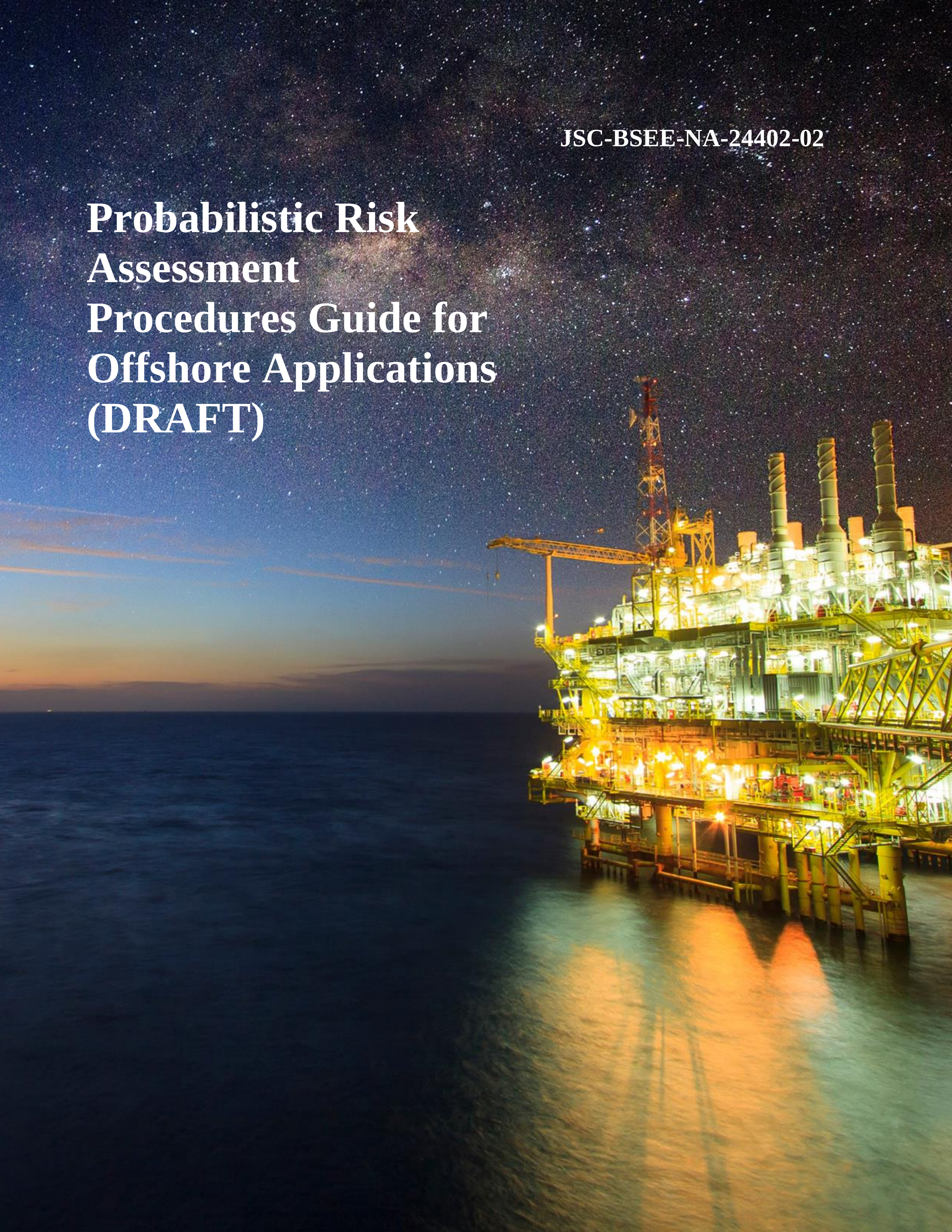
JSC-BSEE-NA-24402-02

Probabilistic Risk Assessment Procedures Guide for Offshore Applications (DRAFT)



DISCLAIMER

This information was prepared as an account of work sponsored by an agency of the U.S. Government. Neither the U.S. Government nor any agency thereof, nor any of their employees, makes any warranty, expressed 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. References herein to any specific commercial product, process, or service by trade name, trade mark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the U.S. Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the U.S. Government or any agency thereof.

Probabilistic Risk Assessment Procedures Guide for Offshore Applications (DRAFT)

Principal Authors:

Robert Cross, NASA Johnson Space Center (JSC)
Robert Youngblood, Idaho National Laboratory (INL)

Other Contributors:

Ronald Boring, INL
Gordon Bower, INL
Bruce Reistle, JSC
Kurt Vedros, INL
Andrew Wolford, Risknology

ACKNOWLEDGEMENTS

The purpose of this guide is to focus on analyses to support the kinds of decisions that need to be made in risk management of offshore facilities; but development of this guide has benefited substantially from numerous earlier developments aimed at other industries, including NASA's "Probabilistic Risk Assessment Procedures Guide for NASA Managers and Practitioners," *NASA's Systems Engineering Handbook*, the ANS/IEEE PRA Procedures Guide, and others. Many of the contributors to those earlier documents are, in some sense, indirect contributors to this guide as well.

The authors gratefully acknowledge numerous useful conversations with Earl Shanks and Mike Worden of the Bureau of Safety and Environmental Enforcement (BSEE), useful review contributions from James Knudsen and Zhegang Ma of INL, and technical editing by Dawn Davidson and Heather Rohrbaugh.

FORWARD WORK

Development of this guide is not yet complete. Work remains to optimize its selection of topics and its treatment of certain issues for use by the offshore industry. This work is ongoing.

CONTENTS

FIGURES

TABLES

ACRONYMS

BGR	Brooks-Gelman-Rubin statistic
ASME	American Society of Mechanical Engineering
BAST	Best Available and Safest Technology
BE	Basic Event
BOP	blow out preventer
BSEE	Bureau of Safety and Environmental Enforcement
BSR	Blind Shear Ram
CCF	common cause failure
DIM	Differential Importance Measure
DPS	Dynamic Positioning System
DSMCS	Dependence-Suspect Minimal Cut Sets
EPRI	Electric Power Research Institute
ESD	event sequence diagram
FMEA	failure modes and effects analysis
F-N	frequency of exceedance
F-V	Fussell-Vesely
HAZOP	hazard and operability study
HEP	Human Error Probability
HFE	human failure event
HRA	Human Reliability Analysis
HSI	Human-System Interaction
IBOP	internal blow out preventer
IE	initiating event
LMRP	lower marine riser package
LOOP	loss of offsite power
MCMC	Markov Chain Monte Carlo
MLD	master logic diagram
MLE	maximum likelihood estimate
MODU	mobile offshore drilling unit
NASA	National Aeronautics and Space Administration
NRC	Nuclear Regulatory Commission
PARLOC	Pipeline and Riser Loss of Containment
PERD	Process Equipment Reliability Database

PRA	probabilistic risk assessment
PSF	Performance Shaping Factor
PVD	population variability distribution
PW	prevention worth
QRA	quantitative risk assessment
RAW	risk achievement worth
ROV	remotely operated vehicle
RRW	risk reduction worth
THERP	Technique for Human Error Rate Prediction

1. INTRODUCTION

1.1 Purpose of this Guide

This guide is intended to assist in the development of probabilistic risk assessment (PRA) of offshore drilling facilities, in order to support decision-making by the Bureau of Safety and Environmental Enforcement (BSEE) and by the industry. This guide is not a policy document, nor does it establish regulatory requirements; rather, it discusses particular modeling techniques that have been found to be useful in a range of applications to decision-making about complex and high-hazard facilities. In order to motivate the approach taken in the remainder of this Guide, this section discusses what kinds of analysis support what kinds of decisions.

The point of departure for development of this guide is the National Aeronautics and Space Administration (NASA) PRA Procedures Guide [1-1], which was itself derived from earlier PRA procedural guidance; but the present development has been informed by numerous other developments from within NASA, as well as work done for the Department of Energy and the Nuclear Regulatory Commission.

1.2 Risk and Risk Management

The term “risk” has many definitions. Most of these definitions are generally consistent with the idea that “risk is uncertainty about the future, viewed through the lens of a value structure (i.e., focusing on outcomes that would be considered adverse).”

In the context of making decisions about complex, high-hazard systems, “risk” is usefully conceived as a set of triplets: failure scenarios, likelihoods of those scenarios, and their actual consequences [1-2]. There are good reasons reasons to focus on these elements rather than focusing on simpler, higher-level quantities such as expected consequences. Risk management involves prevention of (reduction of the frequency of) adverse scenarios (scenarios having undesirable consequences), and promotion of favorable scenarios (scenarios with favorable, or at least benign, outcomes). This requires understanding the elements of adverse scenarios so that they can be prevented, and understanding the elements of successful scenarios so that they can be promoted.

Even if the decision problem is simply to decide whether a facility is deemed adequately safe, the level of assurance (the decision-maker’s confidence) derivable from understanding scenarios far exceeds the level of assurance derivable from an abstract summary of expected consequences.

1.3 Scope of this Guide

Figure 1-1 (adapted from [1-3]) illustrates a general process for safety analysis. The leftmost portion of the figure begins the process with recognition of the decision being supported and an assessment of what technical results are needed to support that decision. The central portion of the figure notionally suggests a range of techniques for safety analysis, going from “qualitative” techniques (thought processes such as Hazard and Operability Studies (HAZOPs), which identify accident potential) to quantitative techniques (modeling processes that generate and quantify scenarios, frequencies, and consequences).

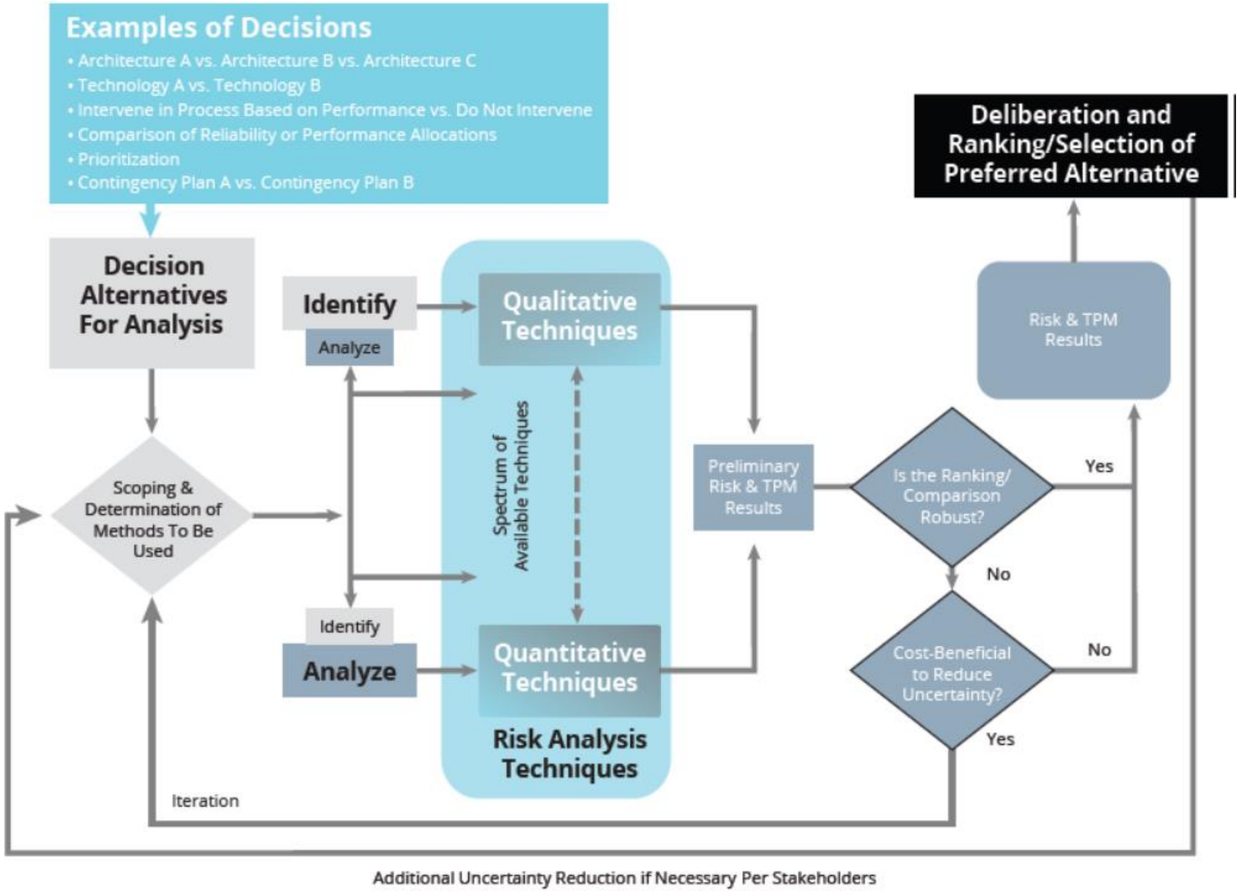

Examples of Decisions
• Architecture A vs. Architecture B vs. Architecture C
• Technology A vs. Technology B
• Intervene in Process Based on Performance vs. Do Not Intervene
• Comparison of Reliability or Performance Allocations
• Prioritization
• Contingency Plan A vs. Contingency Plan B
Deliberation and Ranking/Selection of Preferred Alternative
Decision Alternatives For Analysis
Identify
Analyze
Qualitative Techniques
Risk & TPM Results
Spectrum of Available Techniques
Preliminary Risk & TPM Results
Is the Ranking/ Comparison Robust?
Yes
No
Scoping & Determination of Methods To Be Used
Identify
Analyze
Quantitative Techniques
Cost-Beneficial to Reduce Uncertainty?
No
Yes
Risk Analysis Techniques
Iteration
Additional Uncertainty Reduction if Necessary Per Stakeholders

Figure 1-1. Process for safety analysis.

Different situations will call for a different mix of techniques. It is not always clear *a priori* what techniques are appropriate in a given situation; correspondingly, in the rightmost portion of the figure, the current state of knowledge (after analysis done to date) is assessed to determine whether there is a need to loop back and do more analysis (or get more information) in order to support the current decision.

Broadly speaking, quantitative techniques such as fault tree analysis are techniques that lead to (a) an explicit scenario set, (b) quantification of the likelihoods of those scenarios, and (c) analysis of the consequences of those scenarios (in short, analysis of the triplets discussed above). Calling the other techniques qualitative does not mean that they are applied absolutely without regard to probability; in fact, it is extremely difficult to absolutely decouple safety thinking from probability. Rather, the term "qualitative" is shorthand for "thought processes that help us to identify accident potential, without explicitly generating and quantifying a comprehensive scenario set."

This guide is focused on the quantitative end of the above-described analysis spectrum, using selected qualitative techniques as a front end to the quantitative analysis, in order to help us think appropriately about what we need to analyze in more detail.

1.4 Probabilistic Risk Assessment

1.4.1 Probabilistic Risk Assessment—What We Get Out of it, How We Use It

Based on modeling scenarios, frequencies, and consequences, PRA quantifies risk metrics. The term "risk metric" refers to probabilistic performance measures that might appear in a decision model, such as the frequency or probability of adverse consequences of a specific magnitude, or perhaps expected consequences. Figures of merit such as system failure probability can be used as risk metrics, but the phrase "risk metric" ordinarily suggests a higher-level, more consequence-oriented figure of merit, such as "spills of a certain magnitude."

In order to support resource allocation from a risk point of view (for a permittee or regulator), it is necessary to evaluate a comprehensive set of scenarios. The set of scenarios may need to include events that are more severe than those considered during design, and more success paths than were explicitly factored into the design. Additionally, system performance must be evaluated realistically. In order to support resource allocation decisions, the point is not usually to establish a bound on system capability or reliability, but rather to *quantify* capability and reliability (to characterize them realistically). In other words, risk-informed resource allocation requires identification and realistic quantification of all risk-significant scenarios, where "risk-significant" depends on the context of the evaluation.

In all but the simplest cases, decision support requires that uncertainty be addressed. Because risk analysis frequently needs to address severe outcomes of complex scenarios, and because these scenarios are too infrequent for us to be able to calibrate our models from experience, uncertainties may be highly significant. These uncertainties need to be reflected in the decision model, not only because they may influence the decision, but also because it is important to understand which of the uncertainties that strongly affect the decision outcome are potentially reducible through testing or research.

PRA is needed (and the effort is justified) when decisions need to be made that involve high stakes in a complex situation, as in a high-hazard mission with critical functions being performed by complex systems. Intelligent resource allocation depends on a good risk model; even programmatic research decisions need to be informed by a state-of-knowledge risk model. (Allocating resources to research programs needs to be informed by insight into which uncertainties' resolution offers the greatest payback.) Developing a comprehensive scenario set to provide decision-makers with the best informed picture of threats and mitigation opportunities is a special challenge, and systematic methods are needed for development and quantification of such a model. Those methods are the subject of this guide.

1.4.2 Using Probabilistic Risk Assessment in the Formulation of a Risk-Informed Safety Case

The above discussion has been carried out with emphasis on the role of PRA in assessing system adequacy, especially with regard to selection of design features. This sort of application began even before safety goals were widely discussed. Increasingly, risk managers need to argue that system designs satisfy explicit risk thresholds; nowadays, even if there is no absolute regulatory or policy requirement, the promulgation of safety goals and thresholds creates an expectation that goals and thresholds will be addressed in the course of safety-related decision-making. This creates an issue for PRA, because in general, it is impossible to prove that the level of risk associated with a complex, real-world system is below a given decision threshold.

Partly because PRA results cannot be proven, a “Risk-Informed Safety Case” (RISC) [1-4] is desirable. The RISC marshals evidence (tests, analysis, operating experience) and commitments to adhere to specific manufacturing and operating practices in order to assure that PRA assumptions, including the performance and reliability parameters credited in the PRA, are fulfilled. Among the commitments needed to justify confidence in the safety of the system is a commitment to analyze operating experience on an ongoing basis, including near misses, to improve operations, improve the risk models, and build additional confidence in the models’ completeness. This is not the same as proving that the PRA results are correct, but it is the best proxy for safety that can be obtained.

These matters are discussed further in the following sections of this guide. The present discussion is simply to motivate the emphases placed in treatments of the risk analysis techniques addressed in Section 2.

1.4.3 Characterization of Safety Margin

For purposes of making safety decisions (deciding whether it is necessary to modify the design or operating practices, whether the system risk is Lowest Level Practicable (LLP), or whether reasonable assurance of adequate protection is available), it is useful to analyze system performance in terms of margin, and moreover to do this in a risk-informed way. What attributes does a model need, to support a risk-informed assessment of margin? What is meant by risk-informed?

The phrase “risk-informed” originated in US Nuclear Regulatory Commission (NRC) practice. The NRC website [1-5] offers the following definitions related to “Risk-Informed:”

- Risk-Informed Decision-Making: An approach to regulatory decision-making, in which insights from probabilistic risk assessment are considered with other engineering insights.
- Risk-Informed Regulation: An approach to regulation taken by the NRC, which incorporates an assessment of safety significance or relative risk. This approach ensures that the regulatory burden imposed by an individual regulation or process is appropriate to its importance in protecting the health and safety of the public and the environment.

One important consideration is whether a comprehensive scenario set is modelled with a view to quantitative analysis of decision alternatives, as opposed to pass-fail compliance with prescriptive requirements derived from surrogates formulated by engineering judgment (such as a large-break loss of coolant accident, an early focus of Atomic Energy Commission thinking about the regulation of light-water reactors in the United States). If you are not modeling a scenario set in a way that supports saying what’s important and what is not, you are not being risk-informed.

The phrase “risk-informed” is now widely used to describe a certain thought process. It appears to have originated with NRC Chairman Jackson in the early or mid-1990s. During the 1980s and early 1990s, numerous papers were being written on the subject of risk-*based* regulation (emphasis added). The context of those papers was deciding whether regulatory burden could legitimately be reduced (or, in principle, whether it needed to be tightened up), based on risk model results. Often, risk model results suggest that burden can be reduced; but then, as now, there was a lot of opposition to reducing burden significantly, based on PRA as the primary justification. For the traditionalists, “risk-*based”* was a non-starter. Enter Chairman Jackson: our decision-making will not be risk-*based*, but it will be risk-*informed*, meaning that we will use risk information as one of several inputs to a decision process, other inputs being things like “defense in depth” and “safety margins,” and addressing a broad range of issues of diverse kinds, and not just compliance with regulations.

The concept of margin has evolved in recent years. Originally, the general idea was that a system's capacity to withstand expected loads should be designed with some leeway, recognizing that things may be a bit worse than anticipated, and this excess capacity could be specified either in terms of a point value of extra capacity, or a point value of a safety factor. Recent years [1-6, 1-7] have seen increased appreciation of the usefulness of viewing margin probabilistically, as summarized in a fairly recent doctoral thesis [1-7]:

- Safety margin is the difference between a characteristic value of the capacity and a characteristic value of the load.
- While [this measure] provides a first approximation of functional reliability, ranking different systems on safety margins alone can lead to erroneous results. The knowledge of the distance from failure in terms of safety margins is not sufficient to evaluate the risk of a system; *the breadth of the uncertain distribution* [emphasis added] is the other important part of the assessment.

The breadth of the uncertain distribution is suggested notionally in Figure 1-2.

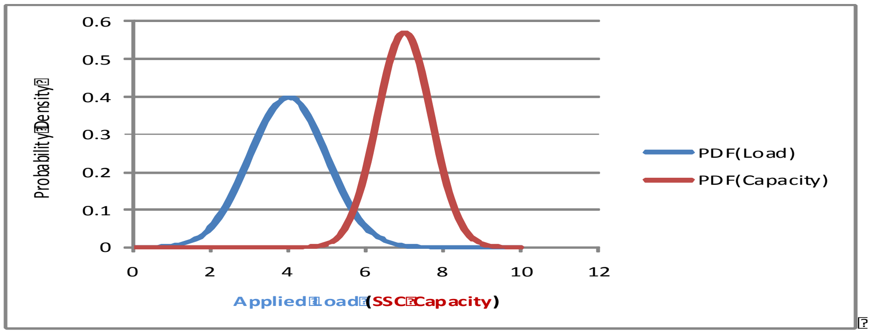


Figure 1-2. Breadth of uncertain distribution.

Figure 1-2 shows an uncertain applied load (such as a pressure) together with the uncertain capacity of a component to survive that load (in this case, the pressure-retaining capability). What matters in the risk analysis is whether the pressure will exceed the actual pressure-retaining capability, and the point of the figure is that if both of these are uncertain, a naïve idea of margin, such as "the distance between the two modes," is inadequate. We need to understand the probability that load exceeds capacity.

Even this load-capacity idea is oversimplified for some purposes, because it is stated above as if the two can be evaluated independently. In some cases, they cannot. Consider a notional example in which high pressure and low pressure-retaining capability are related due to high temperature; in such a case, a calculation based on the simple figure above would underestimate failure probability. In such a case, a more simulation-based approach to risk analysis is necessary; this is discussed in Section 2.

Safety margin characterization is risk-informed if it is based on the following:

- An issue space is formulated, implicitly defining a class of scenarios to be analyzed probabilistically and the figures of merit[a] to be evaluated probabilistically, margin then to be analyzed in terms of those figures of merit in those scenarios.
    - Aleatory[b] variables are identified and assigned appropriate distributions.
    - The state of knowledge within that issue space is delineated in terms of state-of-knowledge probability distributions on uncertain variables, or perhaps probability bounds analysis.[c]
- The scenario set is analyzed in sufficient detail (with sufficient coverage of the issue space) to:
    - Characterize margin in the relevant figures of merit, including the comparison of absolute margin with variability and uncertainty
    - Understand the significance of variability and uncertainty separately
    - Understand the probability of failure (the probabilistic weight of scenarios having zero or negative margin) at least semi quantitatively
    - Understand the main drivers (particular conditions under which margin is high or low) pointing to:
        - Failure modes or initial conditions, control of which would increase margin
        - Information that needs to be obtained in order to reduce uncertainty.

This definition does not address whether the analysis is good or poor; it only addresses whether it is structured to culminate in a probabilistic characterization of margin in a given issue space.

1.4.4 Summary

The essence of "risk-informed" is to create a basis for resource allocation (by permittee and by regulator) that does the best job we know how to do, consistent with our state of knowledge and institutional constraints (such as limitations on the kinds of analysis we can afford). In order to be risk-informed, the analysis must be geared to supporting conclusions about which scenarios are more important than others, and how much more important, and how beneficial (or how justifiable) it would be to add preventive or mitigative measures beyond what is already there. Modeling to support risk-informed decision-making will tend to have the following attributes:

- It will comprehensively analyze representative scenarios within the slice of event space that is probabilistically significant for the decision
- It will make little or no use of bounding (worst-case) arguments, and will instead strive for realism embedded within an honest treatment of uncertainty

[a] Typically, these will be performance metrics in terms of which system success and system failure can be defined.

[b] "Aleatory" uncertainty refers to the variability in outcomes from one trial to the next: the outcome of a roll of honest dice is uncertain, and this uncertainty is aleatory. The term "aleatory" is contrasted with "epistemic," which refers to limitations of our state of knowledge. If we are not sure what fraction of the time a given coin will yield "heads," this is a kind of uncertainty that we could, in principle, reduce by carrying out experiments; this kind of uncertainty is "epistemic." These concepts are discussed in Section 3.

[c] "Probability bounds analysis" [1-8] is the name given to an approach to propagating uncertainty that works with intervals (upper and lower bounds) on the values of the uncertain variables, rather than sampling from explicit probability density functions of those variables.

- It will comprehensively analyze the variability (the aleatory uncertainty) in scenario outcomes.
- It will methodically analyze the implications for the decision of the limitations of the current state of knowledge

This is what we can get out of PRA, and how we use it. The methods and tools discussed in this guide are aimed at accomplishing these things.

1.5 References

1-1 *Probabilistic Risk Assessment Procedures Guide for NASA Managers and Practitioners*, NASA/SP-2011-3421, National Aeronautics and Space Administration, Washington, DC, December 2011.

1-2 Kaplan, S. and B. J. Garrick, "On the Quantitative Definition of Risk," *Risk Analysis*, Vol. 1, No. 1, March 1981.

1-3 *NASA Systems Engineering Handbook*, NASA/SP-2007-6105, Rev. 1, National Aeronautics and Space Administration, 2007, downloadable from the NASA website.

1-4 *NASA System Safety Handbook*, Vol. 1, System Safety Framework and Concepts for Implementation, NASA/SP-2010-580, Version 1.0, National Aeronautics and Space Administration, November 2011 (downloadable from the NASA website).

1-5 See the Glossary on the NRC website, http://www.nrc.gov/reading-rm/basic-ref/glossary.html

1-6 Shaw L., M. L. Shooman, and R. Schatz, "Time-Dependent Stress-Strength Models for Non-Electrical and Electrical Systems," *Proceedings of Annual Reliability and Maintainability Symposium,* pp. 186-197, IEEE Press, New York, 1973.

1-7 Pagani, L., "On the Quantification of Safety Margins," Massachusetts Institute of Technology, 2004.

1-8 *Verification and Validation in Scientific Computing*, W. L. Oberkampf and C. J. Roy, Cambridge University Press, 2010.

2. RISK ANALYSIS TECHNIQUES

2.1 Qualitative Risk Assessment Techniques

As noted in Section 1, the phrase “qualitative risk assessment techniques” is here taken to refer to “thought processes that help us to identify accident potential, without explicitly generating and quantifying a comprehensive scenario set.” The terms “quantitative” and “qualitative” are not perfect descriptors of the distinction that we are trying to make; refraining from using numbers in fault-tree analysis does not eliminate its capability to generate, and even notionally rank, a comprehensive scenario set (e.g., based on the order of the minimal cut sets). By “qualitative,” we mean techniques such as HAZOP, which entail a great deal of thought, but do not typically involve explicit construction of a risk-model representation of a facility.

There are multiple reasons to consider qualitative techniques:

- Sometimes, qualitative techniques are adequate, by themselves, to support the current decision (for example, “design evaluation” can be a category of decision).
- In practice, quantitative techniques need to start out with the insights provided by the qualitative methods: for example, identification and grouping of initiating events, and the development of event tree structure, need to be informed by insights from techniques such as HAZOP.

For the latter reason, discussion of some qualitative techniques is provided in Section 2.2. The present section mentions a few qualitative techniques and suggests how to decide when they are sufficient. However, it is not the purpose of this section to provide detailed procedural guidance on those techniques as stand-alone applications. First, such a discussion would be beyond the scope of this guide; second, abundant material of that kind already exists elsewhere.

Accordingly, the following subsections will mention selected tools with a view to showing how they address the above two considerations: when they suffice for decision-making, and how they fit into quantitative modeling.

2.1.1 Comparison of Selected Qualitative Risk Assessment Techniques

2.1.1.1 Hazard and Operability Study. HAZOPs are performed in a group setting where a facilitator leads a technically diverse group of experts through an exercise to identify hazards related to equipment or operations of a given system in a given operating mode. The design intent in each operating mode needs to have been specified in sufficient detail to support a sensible discussion of system behavior; in particular, nominal values need to have been specified for all important system parameters. The HAZOP discussion is then cued to analyze the system considering deviations of key parameters in one node at a time, based on applying guide words (e.g., high, low) to each parameter (e.g., flow) characterizing each node (e.g., high flow in Node 32, low flow in Node 32). For each such deviation, the group brainstorms possible causes and possible consequences of each cause, and then may consider other factors relevant to the decision context, including possible recommendations for design changes. This discussion implicitly addresses classes of scenarios, identifying them in terms of physical behaviors, many of which could be caused by any of several different component states (good or failed), and some of which could arise even if no components are nominally failed. A notional example of part of a HAZOP table is shown in Table 2-1.

Table 2-1. Example of Hazard and Operability Study.

HAZOP of Drilling Rig's Mud System					
Node	**Deviation**	**Cause**	**Consequence**	**Mitigation**	**Risk Ranking**
1	Low mud weight	Improper materials	Potential underbalance condition leading to well kick	Proper vendor selection for materials Inspection of materials before use Verification of analysis Training	Likelihood - 3 Medium - 3
			…		
		Incorrect mud weight analysis			
		Human error			
2	High mud weight	….		….	….
….	…	…..	….	….	….

2.1.1.2 Failure Modes and Effects Analysis. Failure modes and effects analysis (FMEA) is a component-based technique that breaks down a system into mechanical and electrical components and postulates how each component can fail and the effect the failure has locally and on the overall system. The result is given in tabular form, and documents, for each component, the ways in which it can fail, and the effects of those failures on the system. FMEAs also typically include how the failure can be detected and the mitigations that are in place to prevent or lessen negative effects. An FMEA may be extended to become a Failure Modes, Effects, and Criticality Analysis (FMECA) by adding an evaluation of the likelihood and consequences of each failure mode. An example of part of an FMEA is shown in Table 2-2.

Table 2-2. Example of failure modes and effects analysis.

Component				**Effects**				
System	**Component**	**Identifier**	**Failure Mode**	**Local Effect**	**Next Level Effect**	**System Level Effect**	**Detection Method**	**Mitigation**
Blowout Preventer (BOP)	Blind shear ram (BSR)	BSR	Fails to close	If well kick occurs, and BSR is required to shut in the well, the BSR will not close and seal	If well kick occurs, the annulus will not be sealed	If well kick occurs, formation fluid will rise past the BOP and potentially reach the drill floor	If well kick occurs return mud flow will be high	Remotely operated vehicle (ROV) intervention, lower marine riser package (LMRP) disconnect
BOP	BSR	BSR	Fails to open					
		….	….					

2.1.2 Other Decision Aids

2.1.2.1 Bowtie Diagrams. Bowtie analysis results in a graphical representation of a class of scenarios that helps decision makers to reason appropriately.

- The middle of the bowtie represents a hazardous condition that results when control of a facility is lost (e.g., an underbalanced condition).
- The left-hand side develops causes that can lead to the hazardous condition and the controls in place to prevent its occurrence.
- The controls (including physical barriers) are placed between the cause and hazard showing the failures that must occur for the hazard to occur.
- The right-hand side portrays scenarios ensuing from the occurrence of the hazardous condition, culminating in consequences on the far right. The scenarios on the right are specified in terms of the functions (including physical barriers) that limit or mitigate the consequences potentially resulting from the hazard.
- Each complete left-to-right path through a bowtie is a representation of a hazardous scenario to be considered.

An example of the parts of a bowtie analysis is shown in Figure 2-1.

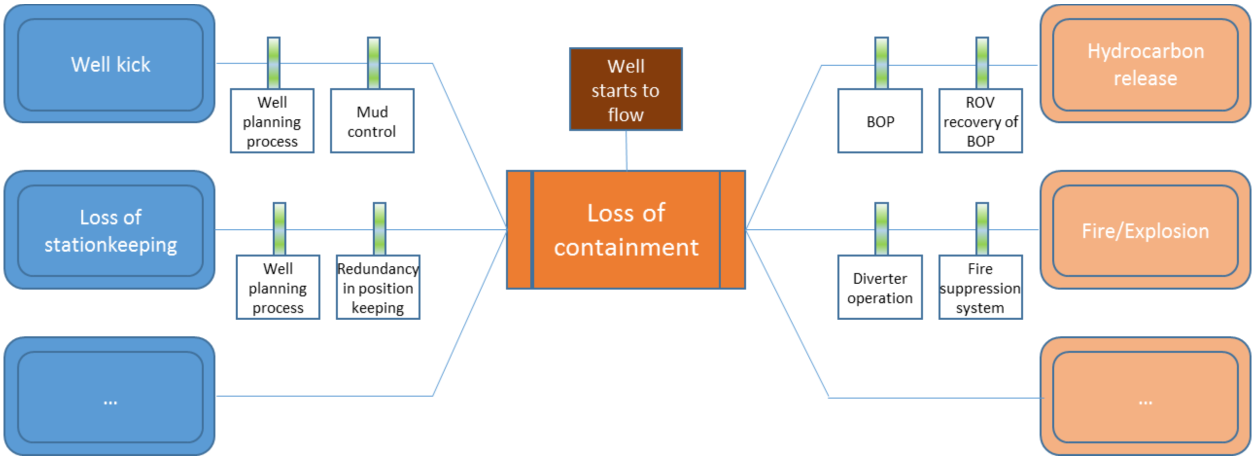


Figure 2-1. Example of bowtie analysis diagram.

2.1.2.2 Risk Matrices. A risk matrix is commonly used to communicate perspective on the significance of particular risks (scenarios, or classes of scenarios having something in common). Notionally, the matrix elements correspond to discrete categories of frequency and consequence, as shown in Figure 2-2; individual scenarios are assigned a likelihood and consequence level and placed on the picture, as illustrated by the numbered circular symbols. Symbol 1 refers to a risk having both low consequences and low likelihood. Its placement in the green region is a way of saying that it is no real threat; either the threat is inherently minimal, or a previously identified real threat has been successfully controlled by prevention or mitigation. Symbol 3 refers to a risk having high likelihood and high consequences, and its placement in the red region is a way of saying that this one needs to be addressed; it may be a showstopper. Symbol 2 refers to a risk that is in between.

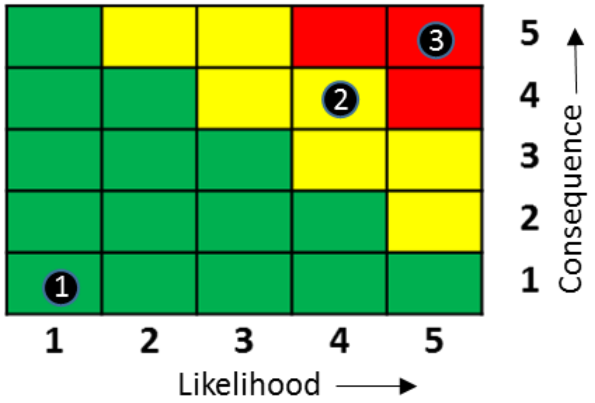


Figure 2-2. Typical qualitative risk matrix.

2.1.3 Strengths and Weaknesses of Qualitative Risk Assessment Techniques

The qualitative risk assessment techniques described above, as well as others not mentioned here, provide a systematic approach to evaluating risk, albeit with different focuses for each method. HAZOP, FMEA, and other methods in this general category promote completeness, which is perhaps the single most critical issue affecting the performance and application of risk modeling. FMEA promotes completeness by considering (in principle) all failure modes for all components; HAZOP does this by considering (in principle) all physical deviations in all nodes. If design intent is properly specified, then anything that would be considered an accident must represent a deviation from design intent; so, by considering all deviations, HAZOP creates at least an opportunity for the group to identify any accident potential that is reasonably foreseeable in the context of any given deviation. Moreover, if a similar system has some operating history, we may have a sense of the likelihood of the deviations identified, and we may have at least some perspective on their consequences.

However, neither FMEA nor HAZOP is particularly well suited to identification or ranking of scenarios that involve multiple failures, or combinations of failure events with off-normal initial conditions. This is not a fault of the thought processes involved; rather, for a system of even moderate complexity, it is impractical for humans to evaluate multiple-failure scenarios without constructing an explicit scenario model and processing that model by computer. Except for very simple systems, it is difficult to determine manually whether a system is single-failure-proof. In principle, FMEA tries to capture the cascading implications of each postulated single failure, but in practice, it is difficult to propagate such implications through the system without some sort of computer aid.

Moreover, it turns out that on-the-fly assessments of probability are unreliable, and it is correspondingly difficult to estimate the likelihood of even moderately complex scenarios. The first large-scale quantitative risk analysis, the Reactor Safety Study, indicated that risk from light-water reactors was dominated not by the sort of postulated pipe rupture event that had dominated safety thought for generations, but rather by events initiated by much more mundane, almost everyday, deviations that are less severe but still challenge safety functions, and need to be dealt with appropriately. The high relative *frequency* of those challenges means that the *reliability of the mitigating systems* must be correspondingly high.

2.1.4 When to Consider Probabilistic Risk Assessment

This guide is not a policy document, nor is it meant to prescribe to facility operators precisely when they need PRA. The present subsection is meant to help management decide what sort of analysis result they need, based on what kind of decision is being made, and what sorts of risks may be in play.

PRA is essentially a high-end risk analysis for supporting certain kinds of decisions. Quite generally, the choice of techniques in a given decision analysis needs to be made in light of the stakes associated with the decision, the complexity involved in analyzing the possible outcomes, the uncertainties, the diversity of stakeholders involved, and perhaps other considerations. By "stakes," we mean the magnitude of the consequences of accidents: fatalities or health effects, adverse environmental effects, significant expense, perhaps other adverse effects on the operating corporation. But high stakes alone may not justify the formulation of a detailed risk model. Selection of a particular course of action may need to be based on strong evidence of low accident likelihood, but if we can get that evidence without a risk model, then we do not need the model.

As an example: if operation of a facility could result in severe safety or environmental consequences and involves new technology or new environments, quantitative risk assessment such as PRA should be considered, because that situation involves high stakes, uncertainty, and (potentially) complexity. Generalizing from that example, questions such as the following can be used to help determine when a PRA should be considered:

- Is the facility design complex?
- Could the consequence of failure of the facility or operation result in higher human or environmental consequences than similar facilities or operations?
- Does the location of the facility or operation magnify the potential consequences of failure? For example, is the location in an area that is fragile, or contain a vulnerable population?
- Have the potential recovery or mitigation measures for the proposed facility or operation been proven in similar environmental situations?
- Has the facility or equipment been used in the proposed type of operation before?
    - How much experience has been gained?
    - What are the outcomes of the use?
- Is the proposed facility or operation in a new or extremely challenging environment?
- Are there any new hazards associated with the facility or operation when compared to facilities or operations performing similar jobs?
- If the facility or operation is being applied in a similar environment with similar consequences to existing facilities or operations, are there any new aspects such as material, equipment layouts, types of equipment, or positioning systems that are untested?

More generally, going back to Figure 1-1, in a situation with high stakes, complexity, uncertainty, and so on, it is unlikely that a qualitative model result will pass the robustness test in the decision diamond on the right of the figure. Correspondingly, the analysts will be directed to loop back through the figure, and choose tools that furnish the results needed to get past the robustness test.

2.2 Quantitative Scenario Modeling: Probabilistic Risk Assessment

When the decision has been made that the qualitative techniques do not offer the assurance necessary to make a decision, quantitative techniques (i.e., PRA) should be considered. The PRA ultimately presents a set of scenarios, frequencies, and associated consequences, developed in such a way as to inform decisions regarding the allocation of resources to accident prevention or mitigation. The implication of the analysis could be a change in design or operational practice, or could be a finding that the design is acceptable as is. Decision support in general requires quantification of uncertainty, and this is understood to be part of modeling and quantification.

For simplicity, the discussion below will be carried out as if the process of PRA model development were a once-through process. But this is not the case. In general, a significant amount of iteration will occur. The process of development is steered by whether the model is adequate for purposes of the decision being supported. Examples of consistency checks include:

- Have we included events that are probabilistically significant relative to the current bottom line (or to other events that we have decided to include)?
- Have we parsed events sufficiently to be able to quantify them accurately?
- Are events parsed down to a level at which we can reasonably treat them as if they were independent?

2.2.1 Elements of a Probabilistic Risk Assessment

This subsection discusses the elements of a PRA logic model. Major elements of the logic are introduced and briefly described; each is then illustrated with respect to simplified examples. The examples emphasize the logic-based (event tree/fault tree) modeling approach; however, some of the concepts described in this section are also applicable to other modeling approaches such as simulation, as discussed in Section 2.3.

A scenario contains an Initiating Event (IE) and, usually, one or more pivotal events leading to an end state. As modeled in most PRAs, an IE is a perturbation that requires some kind of response from the crew or one or more systems. Note that for an IE to occur, there may need to be an associated enabling event(s) that exists (e.g., for a fire IE to occur, there would need to be combustible material present). Then, the scenario end state(s) is defined according to the decisions being supported by the analysis, in terms of the kind and severity of consequences, ranging from completely successful outcomes to losses of various kinds. The pivotal events are then formulated so that specifying which ones occur in a given scenario (and which ones do not occur) is sufficient to determine which end state that scenario goes to. Pivotal events may be system failures, human errors, external events, or other things.

The first major step in logic model development, Step 1, is to determine the boundaries of the analysis. First, based on the goals of the analysis and decisions to be made, what end state(s) are of interest? Examples include:

- Loss of life or injury to personnel
- Damage to the environment
- Damage to, or loss of, equipment or property (including facilities and public properties)
- Unexpected or collateral damage.

Determination of which end states will be analyzed will determine the IEs and critical functions that must be included in the analysis.

In many cases, in addition to the end state(s), the boundaries of the analysis would define what a successful end state would be. For instance, if the end state of interest was an uncontrolled release of hydrocarbons to the environment during exploration drilling, the success state may be defined in different ways depending on the goals of the analysis. If the goal is to evaluate the likelihood of an accident, the success end state may be defined as successful control of the well by the blowout preventer (BOP). If the goal is to evaluate the likelihood of a release as a function of the magnitude of release, considerations beyond the BOP must be taken into account, such as ROV intervention and well capping, and success becomes killing the well rather than successfully isolating the well by the BOP alone.

Step 2 involves determining what perturbations to the process, or IEs, present a challenge that could lead to the end state(s) of interest. There may be many IEs, some of which may be grouped together because the response is the same or very similar (e.g., a well kick due to the swab effect and a well kick due to an unexpected overpressure zone), or the IEs may require different responses (e.g., a well kick due to an unexpected overpressure zone and an inadvertent LMRP disconnect). Determination of the IEs will further determine the critical functions necessary to achieve a successful end state through development of event sequence diagrams (ESDs) that detail the response to the IE. Identification of IEs and ESDs / critical functions are discussed in more detail in Sections 2.2.2 and 2.2.3, respectively.

Step 3 is building the event tree(s) that develops specific accident sequences leading to the end state(s) of interest, and is used, in conjunction with fault trees, to quantify the frequency of each end state. One event tree is usually developed for each IE or group of IEs. The graphical event tree starts with the IE, which is followed by a number of pivotal events determined through the accident progression / critical function assessment in Step 2. Each of the pivotal events has a potential success or failure path (although in some cases, more than a binary state is possible), and are usually ordered as a time sequence of the response to the IE. A detailed discussion of how event trees are built and function is found in Section 2.2.4.

Step 4 involves pivotal event development. This generally involves development of models such as fault trees for each of the pivotal events in the event trees. Fault trees (discussed later in some detail) are models that start with a "top event" that is a failure or condition, and develop ways in which that event can happen, expressed in terms of "basic events." There can be many basic events (the lowest level in the fault tree) and very many combinations of basic events that can cause the top event. The top-event model may comprise systems, human actions, environmental conditions, etc. The basic-event level is where the probabilistic data are used for quantification. Fault tree development and quantification are discussed in Section 2.2.5.

With the development of the event trees and supporting fault trees, the logic model is completed. Quantification requires the development of data to populate the logic model and is discussed in Section 3.

2.2.2 Initiating Event Development

One of the first modeling issues that must be resolved in performing a PRA is the identification of accident scenarios that are related to the analysis goals. This modeling of "what can go wrong?" follows the systematic identification of accident initial causes, called initiators or Initiating Events (IEs), grouping of individual causes into like categories, and subsequent quantification of their likelihood. IEs may lead directly to adverse consequences; or, more typically, additional failures, equipment and/or human, are required in order to reach an adverse consequence.

The identification of initiators can come from a variety of techniques, including those discussed in Section 2.1 (e.g., HAZOPs). Precursor events may also suggest the types and frequencies of applicable upsets. In addition to those, one may deduce IEs through the development of a Master Logic Diagram (MLD).

The MLD is analogous to a fault tree. The top event of an MLD is a type of challenge to facility safety. The top levels are defined by functional events and/or external events (e.g., environment), and successive levels of the MLD are developed until the effect of the failure or event is the same as the block it feeds into. The goal is not only to support identification of a comprehensive set of IEs, but also to group them according to the challenges that they pose (the responses that are required as a result of their occurrences). IEs that are completely equivalent in the challenges that they pose, including their effects on subsequent pivotal events, are equivalent in the risk model.

A useful starting point for identification of IEs in a MLD is a specification of “normal” operation in terms of (a) the nominal values of a suitably chosen set of physical variables and (b) the envelope in this variable space outside of which an IE would be deemed to have occurred. An example of this could be the difference in pressure between the wellbore and formation. There is an expected value for the pressure, and a deviation (increase or decrease) by a certain amount would not be “normal” and could result in a well kick. A comprehensive set of process deviations can then be identified, and causes for each of these can be addressed in a systematic way.

Figure 2-3 shows an example of a MLD that might be used to identify IEs (not exhaustive) related to an uncontrolled release of hydrocarbons during normal drilling operations due to an excessive pressure differential between the wellbore and formation.

For this example, the end state of an excessive delta P between the wellbore and formation is the top block. The next level down would be the sense of the excessive pressure, either high or low. Each of those paths may then be developed into individual events (e.g., swab effect), or categories (e.g., loss of mud column) that may have even lower levels. The level of the MLD in blue represents IEs that are challenges to the end state.

Once an exhaustive list of IEs has been identified, the frequency of the IEs may be quantified. Note that IEs are developed as frequencies because they are on a per-unit-time or per-mission basis. Some IEs may be singular events, such as the frequency of a tropical storm in a particular geographical area and of a specific magnitude that could result in station-keeping challenges. Historical data are available and might be applicable to quantification of the frequency of this IE. Other IEs may be more complex and require further development before quantification can occur. IEs like inadvertent disconnect of the LMRP may require a fault tree to establish the causes or enabling events, which may then be quantified in the fault tree to estimate the IE frequency. Occasionally, some IEs may be conditional. For instance, severe environmental conditions resulting in a drift-off condition may be seasonally and geographically dependent. A severe environment may be more likely in some months (e.g., June–September in the Gulf of Mexico due to the potential formation of tropical cyclones). In this case, a temporary exploration operation may consider the time that is planned for the well, or in the case of a production platform, different initiators with different seasonal frequencies may be used to account for the IE dependencies.

Quantification of an IE frequency is often done using a Bayesian approach, where operational data are evaluated to determine the initiator frequency, including the uncertainty on the frequency. This approach is described in Sections 2.2.5.9 and 3.

2.2.3 Event Sequence Diagrams

Once an exhaustive set of IEs has been established, accident progression, with the elements shown in Figure 2-4, can be modeled using an ESD and/or its derivative, an event tree. Both are used in PRAs to provide organized displays of sequences of system failures or successes, and human errors or successes that can lead to specific end states. A traditional accident progression analysis begins with an ESD, refines it, and then transforms it into an event-tree format. An ESD starts with the premise that some IE has occurred, and then maps out what could occur in the future if particular systems (or humans) fail or succeed in responding appropriately to the IE. The ESD shows event sequences (or pathways) leading to different end states. ESDs are a very useful step in developing logic models: ESDs permit the complex relationships among IEs and subsequent responses to be displayed more readily and understandably than do event-tree models.

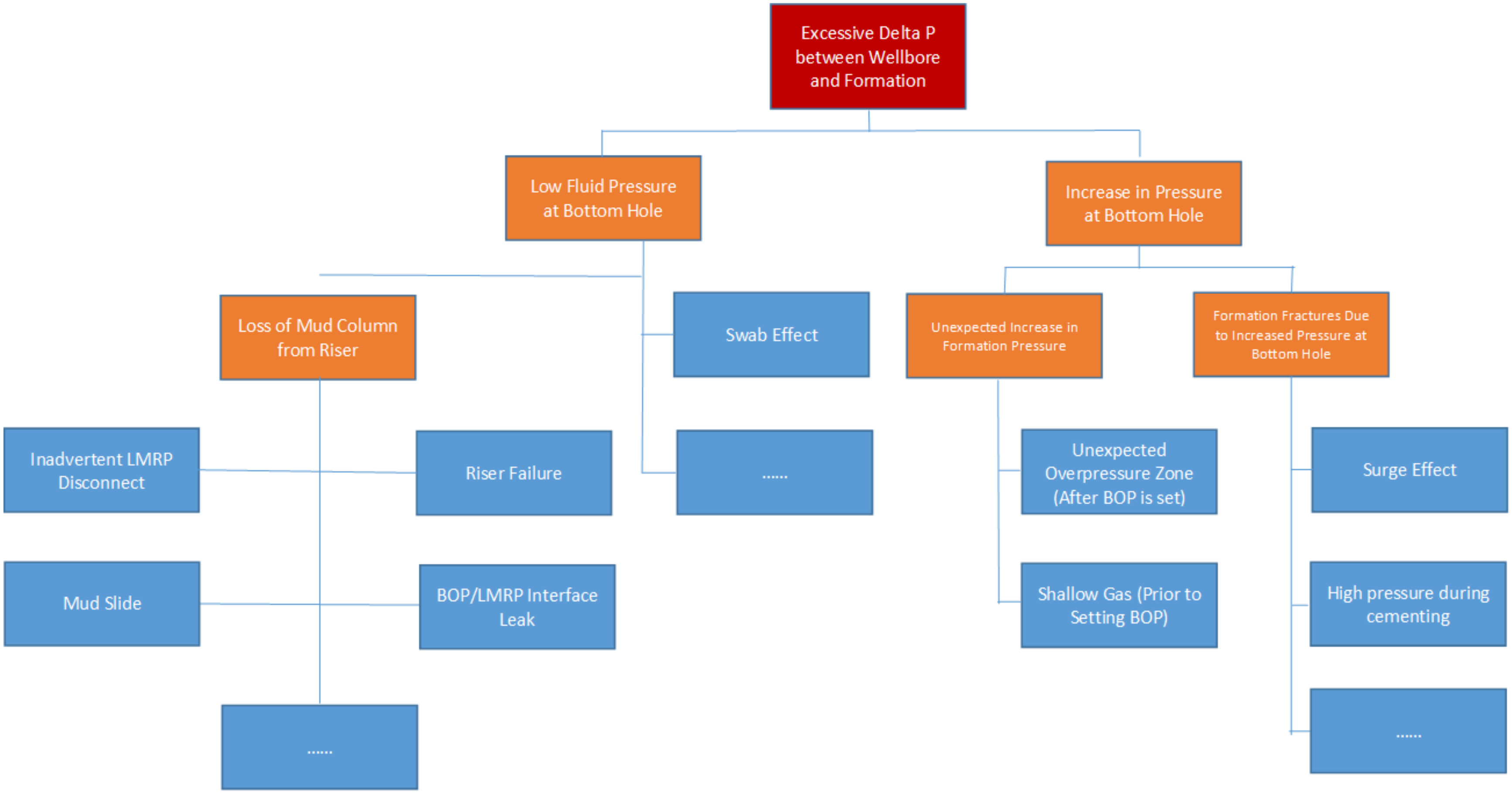

Excessive Delta P between Wellbore and Formation
Low Fluid Pressure at Bottom Hole
Increase in Pressure at Bottom Hole
Loss of Mud Column from Riser
Swab Effect
Unexpected Increase in Formation Pressure
Formation Fractures Due to Increased Pressure at Bottom Hole
Inadvertent LMRP Disconnect
Riser Failure
......
Unexpected Overpressure Zone (After BOP is set)
Surge Effect
Mud Slide
BOP/LMRP Interface Leak
Shallow Gas (Prior to Setting BOP)
High pressure during cementing
......
......

Figure 2-3. Notional MLD related to candidate IEs.

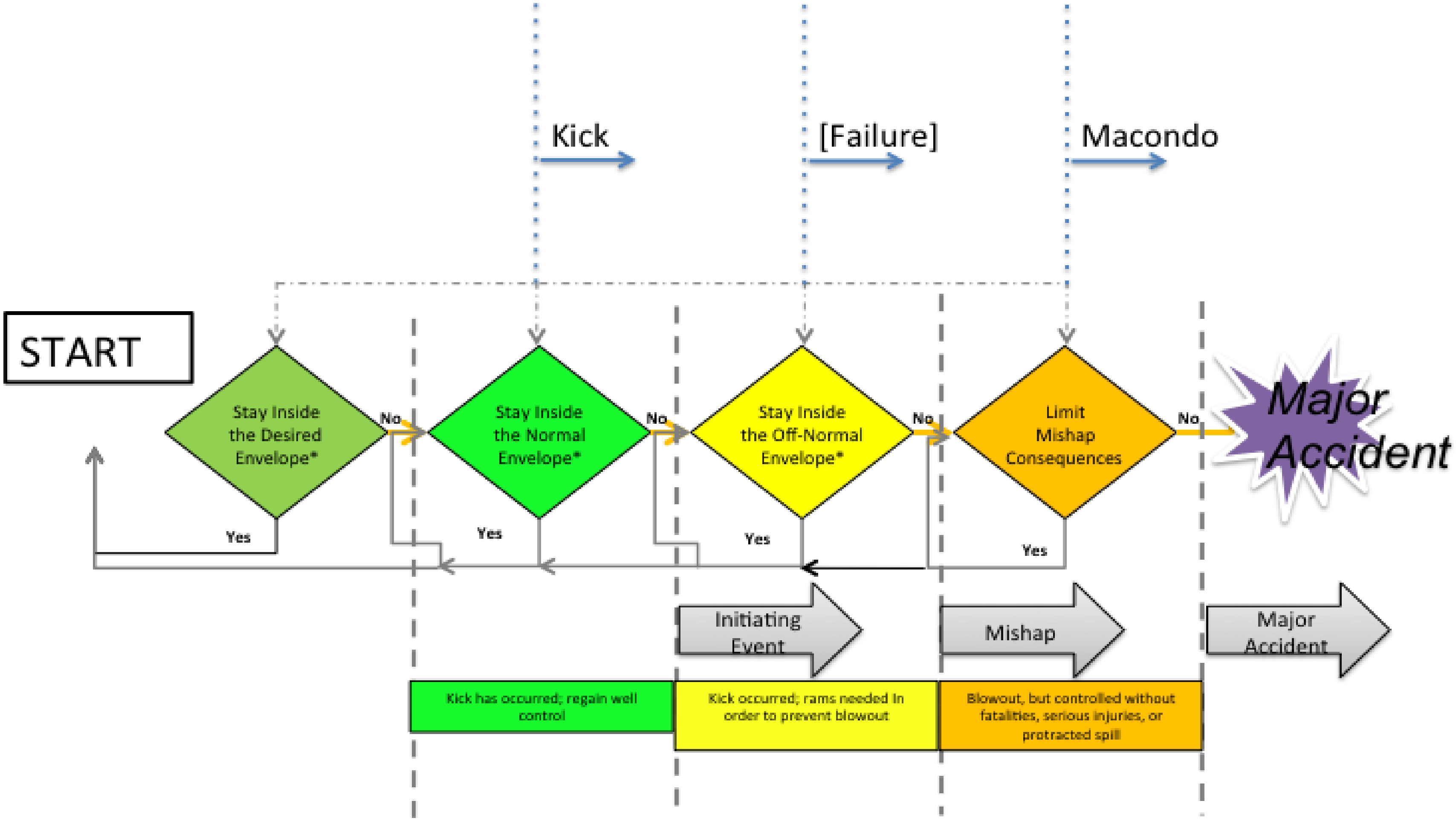

Kick
[Failure]
Macondo
START
Stay Inside the Desired Envelope*
Stay Inside the Normal Envelope*
Stay Inside the Off-Normal Envelope*
Limit Mishap Consequences
Major Accident
No
No
No
No
Yes
Yes
Yes
Yes
Initiating Event
Mishap
Major Accident
Kick has occurred; regain well control
Kick occurred; rams needed in order to prevent blowout
Blowout, but controlled without fatalities, serious injuries, or protracted spill

Figure 2-4. The elements of an accident scenario.

In principle, one ESD is developed for each IE; however, responses to nominally different IEs in terms of system controls or mitigations may be very similar, or even the same. In such a case, a single ESD may be used to represent the accident progression for a set of IEs. The objective is to illustrate all distinct paths from the IE to the end states.

An important attribute of an ESD is its ability to describe and document assumptions used in event trees. An ESD can be very detailed, depicting all sequences considered by the PRA analyst. When simplifying assumptions are used to facilitate event-tree construction or quantification, the ESD may furnish a basis for demonstrating why such assumptions are conservative, or (alternatively) probabilistically justified.

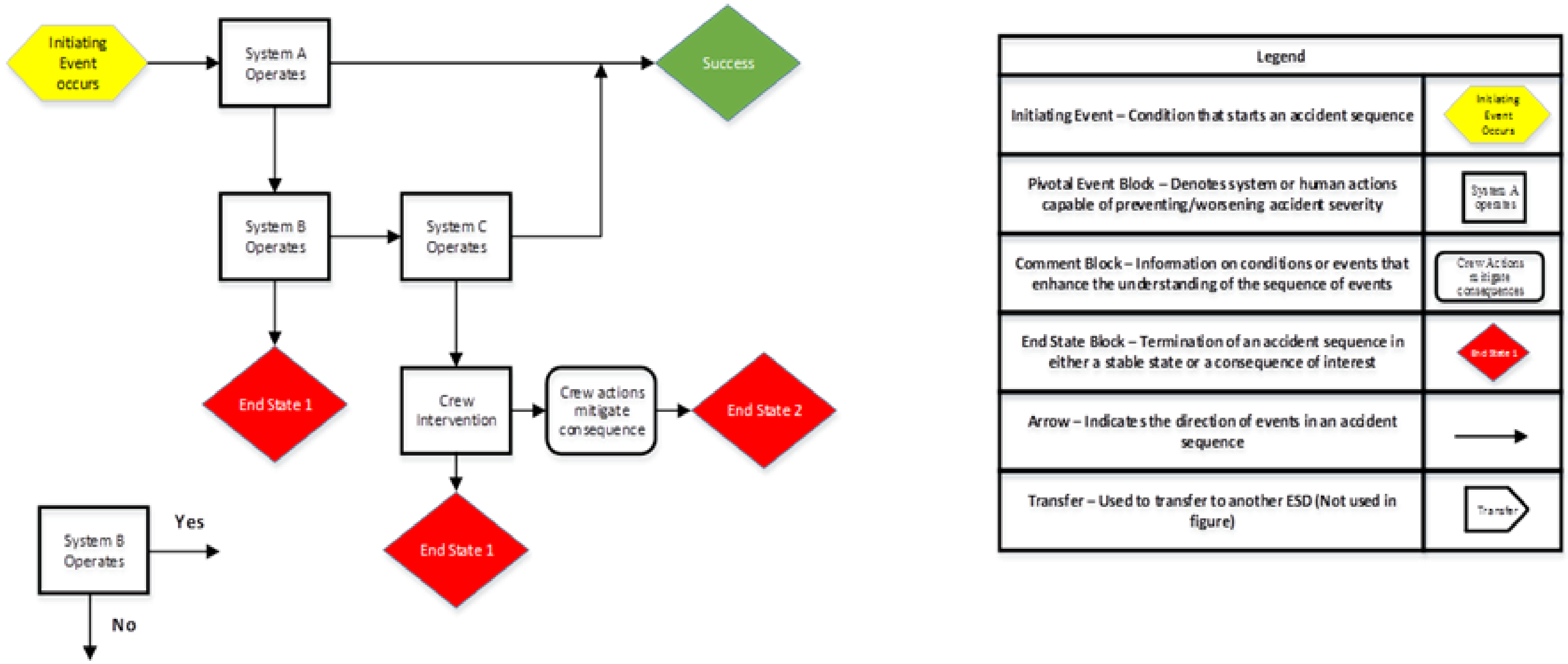
Figure 2-5 depicts a simple ESD and its symbols. The Figure 2-5 ESD begins with an IE that perturbs the function being modeled from a stable state. The initial response to this perturbation is provided by System A, and if System A compensates for the IE, then a successful end state results.

If System A should fail, Systems B and C together can compensate for the IE. According to Figure 2-5, a successful end state ensues if Systems B and C start and operate satisfactorily.

Failure of System B to start or operate results in End State 1. If System B is successful and System C fails to start or operate properly, successful crew intervention can still provide some mitigation for the accident and will result in a different end state (End State 2). If the crew efforts are unsuccessful, End State 1 results.

Figure 2-6 is a more complex (but still relatively simple) ESD developed to evaluate accident sequences related to a well kick from drilling while drilling that results in an environmental release. Five different end state designations are used: WELLCONTAINED and 4 different environmental release end states. The WELLCONTAINED end state represents paths that result in no release or a relatively insignificant environmental release and the condition of the well is stable (i.e., no hydrocarbon flow) and contained. The environmental release end states represent paths where mitigating events have failed to prevent the accident from progressing, resulting in a release to the environment. A different end state designation is used for each path depending on the mitigation used for gaining control of the well (e.g., ROV intervention, and well capping). Even for a given path, the magnitude of the release can vary; for example, the magnitude of the release would depend on the flow rate of the well, and on the time it is flowing to the environment. If a relief well is needed to stop the flow, the resulting time delay will lead to a much larger release than if an ROV can intervene and stop the flow early on in the accident.

In general, an ESD is based on the designer’s intentions. Figure 2-7 illustrates the process of ESD development. Since an ESD is success-oriented, the process begins (Step 1) by identifying the anticipated response to the IE, in this case a well kick while drilling, out to a successful outcome. For this example, the anticipated response is to first properly detect the kick before it reaches the BOP. If that occurs, then mitigating actions can take place to shut in the well. The first responses are shown as comments, stopping the rotation of the drill pipe and positioning the drill string. The mud pumps are then stopped before the annulus is closed by the annular or pipe ram. These first two actions are listed as comments because they affect other steps and will be accounted for in them. The next pivotal event is closing the annular preventer and opening the choke line. This isolates the well except for the path up the drill string if the drill string or another tubular is present in the BOP. With the annular successfully working, the next question would be if the path through the drill string is isolated. The drill string has a float valve that prevents back flow through the pipe when closed. The well is monitored for flow/pressure and if isolation is successful and no flow is present, the well is controlled, and a well kill program may be initiated. In order to keep the diagram simple, the well kill process is not shown here.



Legend	
Initiating Event – Condition that starts an accident sequence	Initiating Event Occurs
Pivotal Event Block – Denotes system or human actions capable of preventing/worsening accident severity	System A operates
Comment Block – Information on conditions or events that enhance the understanding of the sequence of events	Crew Actions mitigate consequences
End State Block – Termination of an accident sequence in either a stable state or a consequence of interest	End State 1
Arrow – Indicates the direction of events in an accident sequence	→
Transfer – Used to transfer to another ESD (Not used in figure)	Transfer

Figure 2-5. Notional ESD.

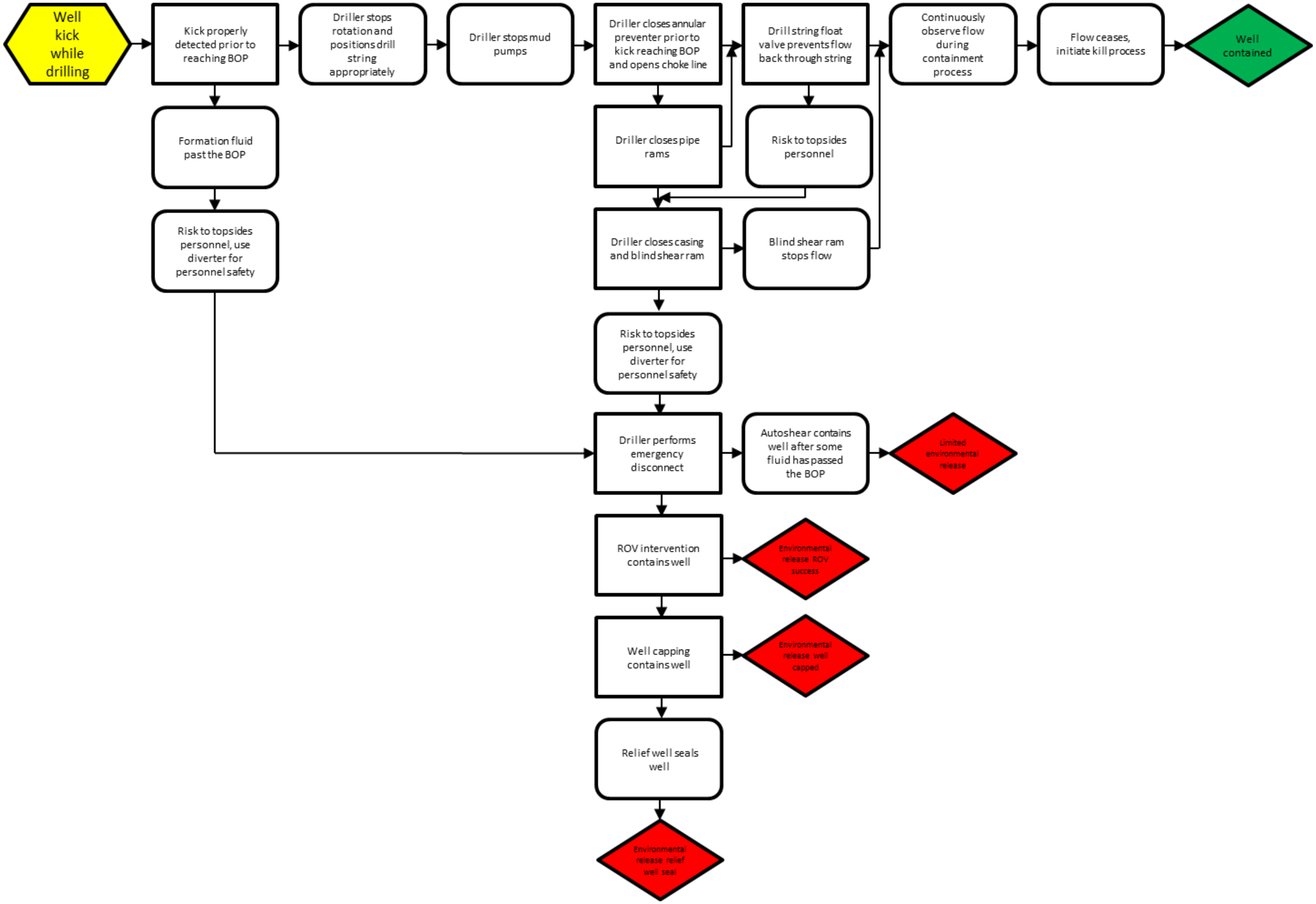

Well kick while drilling
Kick properly detected prior to reaching BOP
Driller stops rotation and positions drill string appropriately
Driller stops mud pumps
Driller closes annular preventer prior to kick reaching BOP and opens choke line
Drill string float valve prevents flow back through string
Continuously observe flow during containment process
Flow ceases, initiate kill process
Well contained
Formation fluid past the BOP
Driller closes pipe rams
Risk to topsides personnel
Risk to topsides personnel, use diverter for personnel safety
Driller closes casing and blind shear ram
Blind shear ram stops flow
Risk to topsides personnel, use diverter for personnel safety
Driller performs emergency disconnect
Autoshear contains well after some fluid has passed the BOP
Limited environmental release
ROV intervention contains well
Environmental release ROV success
Well capping contains well
Environmental release well capped
Relief well seals well
Environmental release relief well seal

Figure 2-6. ESD for a well kick while drilling.

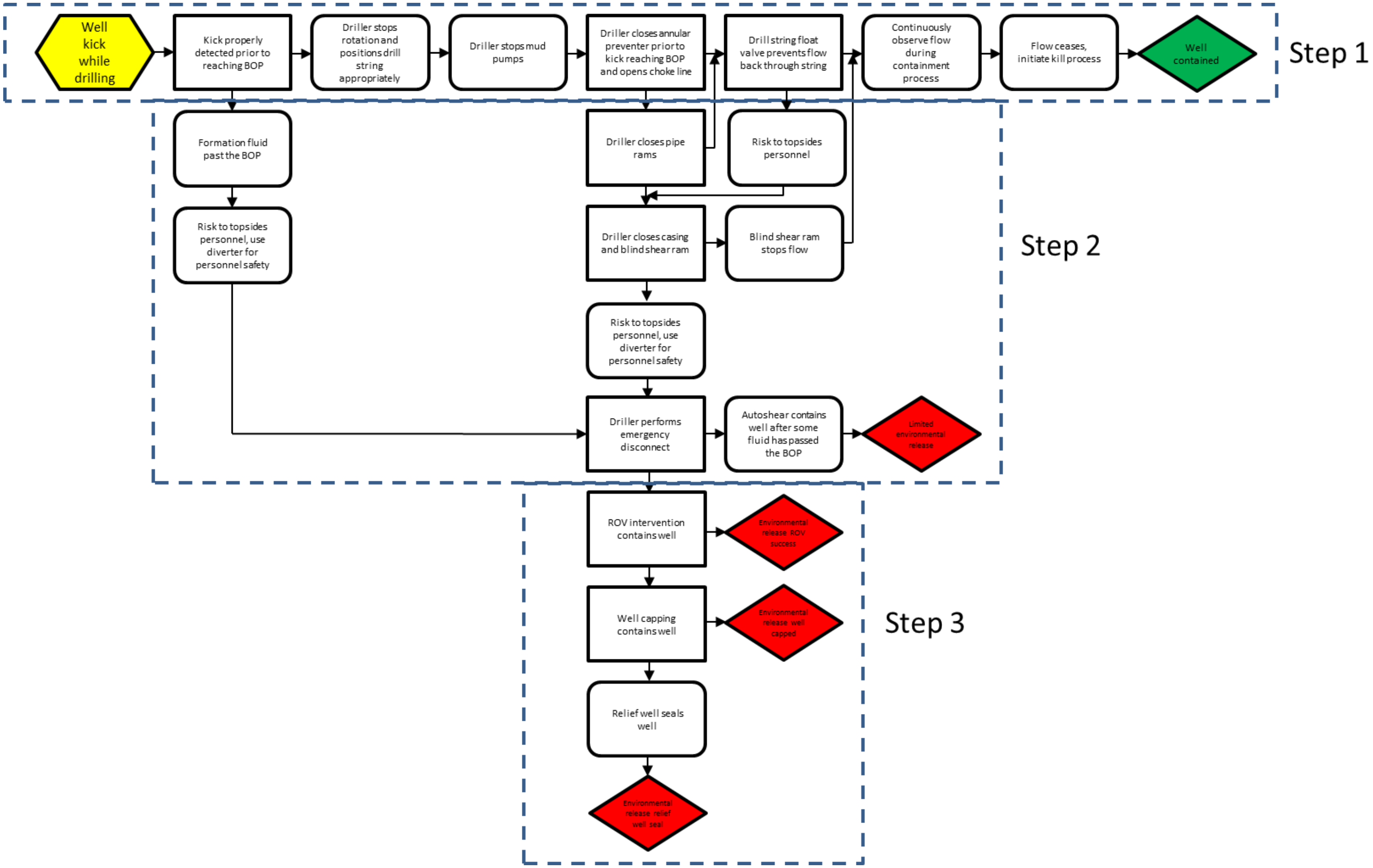

Well kick while drilling
Kick properly detected prior to reaching BOP
Driller stops rotation and positions drill string appropriately
Driller stops mud pumps
Driller closes annular preventer prior to kick reaching BOP and opens choke line
Drill string float valve prevents flow back through string
Continuously observe flow during containment process
Flow ceases, initiate kill process
Well contained
Step 1
Formation fluid past the BOP
Risk to topsides personnel, use diverter for personnel safety
Driller closes pipe rams
Risk to topsides personnel
Driller closes casing and blind shear ram
Blind shear ram stops flow
Risk to topsides personnel, use diverter for personnel safety
Driller performs emergency disconnect
Autoshear contains well after some fluid has passed the BOP
Limited environmental release
Step 2
ROV intervention contains well
Environmental release ROV success
Well capping contains well
Environmental release well capped
Relief well seals well
Environmental release relief well seal
Step 3

Figure 2-7. ESD development steps.

The next step (Step 2 in Figure 2-7) in developing the ESD is to consider what happens when failures occur. On the first block after the IE, if the kick is not detected prior to formation fluid reaching the BOP, then no other barriers exist to prevent the fluid already past the BOP from reaching the rig. A comment was placed in the ESD to show that the diverter may be used, but the diverter is for personnel safety and not for preventing environmental release. The purpose of the ESD is to estimate environmental release, which the diverter does not mitigate, so it is commented for later use if personnel safety is also analyzed.

Once it is determined that that the rig will be impacted, the mitigating actions are assumed to start with an emergency disconnect from the well. This action sets mitigation in motion by operating the casing shear ram and then the blind shear ram (BSR). Successful operation of the BSR is all that would be required to seal the well, as the casing shear ram is assumed not to provide an effective sealing surface. The casing shear ram is operated first, however, in case there is any tubular present in the BOP that would prevent or inhibit closure of the BSR. In developing the sequence of events, it was noted that the casing shear and BSRs may not be effective for all tubulars, and if some specific types of tubulars such as drill collar or tool joints are present, the shear rams will not be able to perform their function. Failures due to nonshearables have the same effect as equipment failures in preventing the shear rams from closing, and will be accounted for in fault trees discussed later. If the BSR works, the end state is that the well is controlled, but a limited release has occurred.

The second failure path is in response to a failure of the annular. In this case, the next step would be to close the pipe rams. The pipe rams will close around the pipe and close the annulus. There are a minimum of two pipe rams available; however, there may be more, and only one has to be successful to shut down flow from the well through the annulus. If the pipe rams fail to contain the well due to a system failure or possibly a tubular that is outside of the design of the pipe ram (e.g., drill collar), formation fluid will continue to travel up the annulus, and operating the shear rams is assumed to be the response. If the pipe rams are successful, then the drill string path is questioned to determine whether that path is isolated or not.

Should both the annular and the pipe rams fail to close the annulus, the shear rams are questioned as in the emergency disconnect sequence, with the difference being in this case that it is a manual action, and not automated like the emergency disconnect sequence. Success of the BSR at this level is assumed to result in a successful containment of the well, with no (or very little) formation fluid getting past the BOP. Success then allows starting the well kill process.

The last pivotal event in the top line is the pivotal event for the isolation of the drill string. If the float valve fails (for simplicity, no credit for a topsides Internal BOP [IBOP] was assumed), formation fluid can reach the rig. In this case, the flow will be significantly less than it would be if the annulus were open to flow, but there is still a risk for personnel on the rig, so the BOP shear rams are questioned. If the BSR is successful, the well kill process can be started.

If the BSR fails to close and seal the well, formation fluid will continue past the BOP and pose a risk to the crew. The next action taken is to perform an emergency disconnect. This action will reattempt to close the shear rams using both the topsides hydraulics and the autoshear function. Success will result in little or no fluid reaching the drill floor, but the rig will be disconnected. This sequence stops here because the well is stable even though the rig is no longer attached.

Step 3 in Figure 2-7 further develops the ESD through the accident management stage out to an environmental release if the emergency disconnect is unsuccessful in closing the BSR. For simplicity, the events considered are only shown as a single block each. At this point in the ESD, the BSR has failed to close or seal, and a release will occur. The next possibility for mitigation would be attempting to manipulate the BOP with an ROV. If this is successful, a release will have occurred, but the magnitude will be somewhat limited due to the relatively short response time. Failure of the ROV (because of BOP condition, ROV failure, etc.) will lead to the next available option, which would be well capping. If well capping is unsuccessful, the only remaining option is a relief well. This is assumed to be successful; however, the release may vary significantly depending on how long the relief well takes to seal the well.

The example ESD developed in Figure 2-7 analyzes end states corresponding to environmental release given that a well kick while drilling has occurred. A well kick may have different causes, such as those shown in Figure 2-3. The ESD can provide a common response in terms of events for similar initiators; however, the probabilities in the ESD may be conditional on the initiator. For instance, if the trip occurs with nothing across the BOP the probability of successfully closing the BSR may be different than if there is drill pipe across the BOP. When quantifying using event trees, these conditions must be accounted for, if we are to accurately estimate the probability of the consequences of interest.

There also may be other IEs that could lead to an environmental release that may have different event sequences. For instance, an inadvertent LMRP disconnect would not have the detection or annular blocks on the top row in Figure 2-7 since the loss of communication with the BOP after the LMRP disconnect negates any actions by the driller. This may therefore require an ESD to be developed specifically to address those scenarios.

2.2.4 Event Trees

Once the accident progression paths are understood, the next step is to build event trees for scenario quantification. An event tree is a graphic that displays scenarios potentially resulting from a specific IE (or a group of functionally similar IEs). Event trees are derivable from ESDs, but event trees are one step closer than ESDs to generation and quantification of scenarios. An event tree distills the pivotal event scenario definitions from the ESD and presents this information in a tree structure that is used to help classify scenarios according to their consequences and perform a quantification of the scenarios. The headings of the event tree are the IE, which is the starting point, the pivotal events showing success or failure of mitigating/ aggravating events, and lastly the end state to bin the consequence of each scenario. Each individual path through the event tree is a "sequence." The event-tree pivotal events are linked to fault trees, and the pivotal event name should match the corresponding fault-tree top-event description. This is because fault trees are tied to the pivotal events and are based on potential failures for that event. An example event tree based on the ESD is shown in Figure 2-8. The simple example in Figure 2-8 shows five sequences on the right-hand side of the event tree with three different end states:

1. SUCCESS
2. ENDSTATE-1
3. ENDSTATE-2.

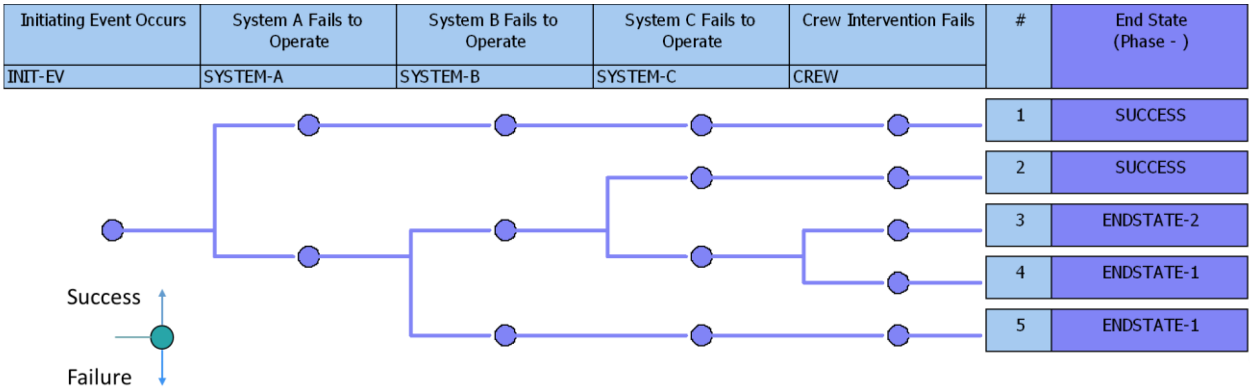


Figure 2-8. Example event tree.

Each sequence resulting in the end state represents a combination of the IE and success and or failures of the pivotal events. For instance, consider Sequence 3 in Figure 2-9.

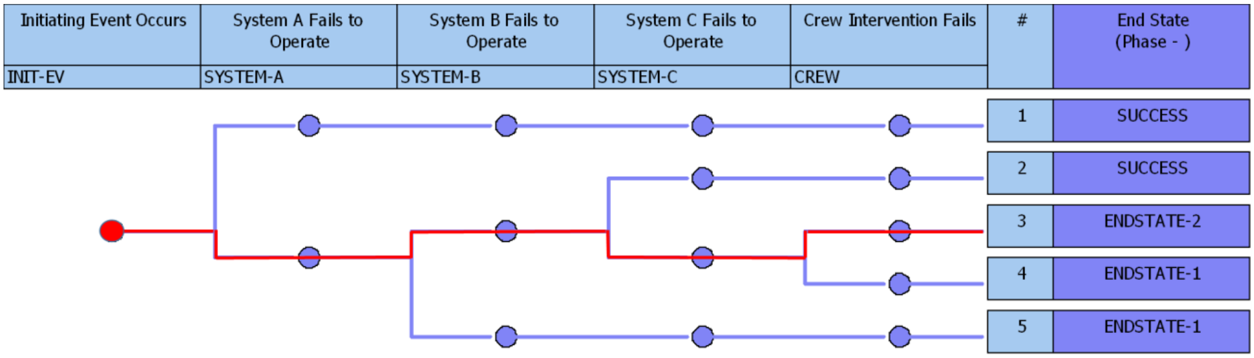


Figure 2-9. Example event-tree sequence.

Sequence 3 starts with the IE and results in ENDSTATE-2. The resulting sequence is a combination of successes and failures of the pivotal events along the path, yielding the expression:

INIT-EV * SYSTEM-A * /SYSTEM-B * SYSTEM-C * /CREW

In the above expression, INIT-EV represents the frequency of the IE, SYSTEM-A and SYSTEM-C represent the probabilities of failure of Systems A and C, respectively, as indicated by the downward step at each of those pivotal events in the event tree. /SYSTEM-B and //CREW represent NOT failure of SYSTEM-B (i.e., success of B) and NOT failure of CREW (i.e., success of Crew), indicated by the upward step in the event tree.

Quantification will be discussed in more detail later. However, for purposes of illustration, the frequency of this event sequence can be quantified, assuming IE and pivotal event values of:

- INIT-EV = 0.10 events per unit time
- SYSTEM-A = 0.02 (failure probability of A given INIT-EV)
- SYSTEM-B = 0.03 (failure probability of B given INIT-EV and failure of A)
- SYSTEM-C = 0.03 (failure probability of C, given INIT-EV, failure of A, and success of B)
- CREW= 0.05 (failure probability of CREW, given INIT-EV, failure of A, success of B, failure of C).

This yields the following as the frequency of occurrence of ENDSTATE-2 (in events per unit time):

0.1* 0.02 * (1-0.03) * 0.03 * (1-0.05) = 5.53E-5 (2-1)

From Figure 2-10, it can be seen that not all pivotal events are questioned in every sequence. Sequence 5 in Figure 2-9 does not question SYSTEM-C or CREW, because once SYSTEM-A and SYSTEM-B have failed, SYSTEM-C and CREW can no longer affect the end state. Dependences like this are typically accounted for when the event tree is developed, so that the resulting sequences are the minimal sets of pivotal events that must occur for that end state to occur. The expression for Sequence 5 then becomes:

INIT-EV * SYSTEM-A * SYSTEM-B.

Substituting the values from above yields:

0.1 * 0.02 * 0.03 = 6.0E-5 (2-2)

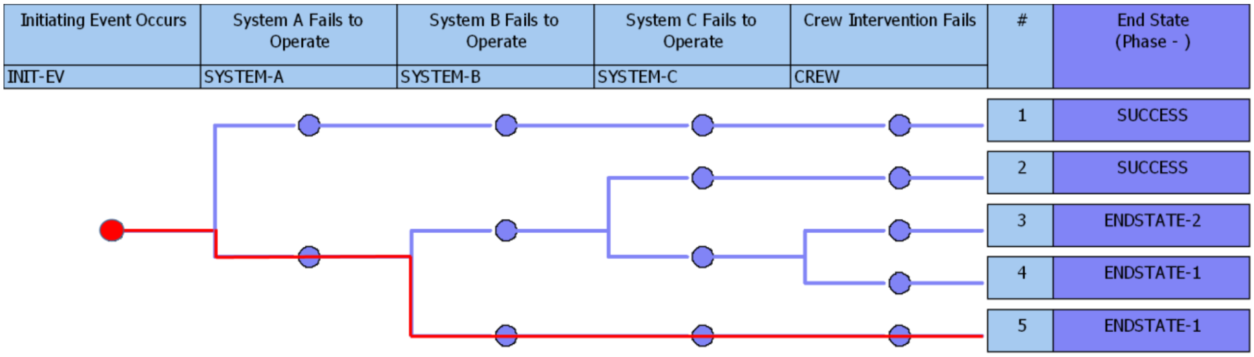


Figure 2-10. Example event-tree sequence where not all pivotal events are questioned.

When the pivotal events are replaced with fault trees, as discussed in Section 2.2.5, it becomes possible to express the event sequences in more detail, namely, in terms of basic events (e.g., component failures) rather than pivotal event names (failures of systems or entire functions). Depending on the size of the fault tree, each event-tree sequence can result in many cut sets or unique contributors that can cause the end state, since pivotal events such as SYSTEM-A may have many different ways to fail (e.g., pump failures, valve failures, and leaks).

2.2.4.1 Event-Tree Development. Developing an event tree usually begins with the ESD. From the ESD in Figure 2-5, four pivotal events were shown:

1. System A operates
2. System B operates
3. System C operates
4. Crew intervention.

The event tree in Figure 2-11 is the start at mapping out the ESD paths from the ESD from the IE to the end states. For clarity, event-tree development often follows the time sequence of events from the ESD. In Figure 2.5, the initial response after the IE is the System A status, so it logically is the second event in the event tree (converted to failure). “System B Fails to Operate” is questioned if “System A Fails to Operate” in the ESD, and if System B is successful, “System C Fails to Operate” is questioned. System B is listed after System A on the event tree because the status of System A must be known before System B is questioned. Similarly, System C is listed after System B because the status of System B must be known. Lastly, Crew Intervention is only questioned if System C is failed, so it must be listed after System C.

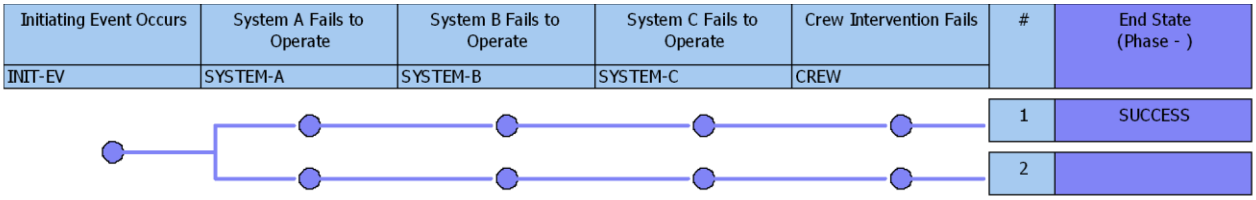


Figure 2-11. Step 1 in building example event tree.

With the top line of the event tree laid out, the next step is to develop the branches for the pivotal events. From the ESD, System A operation directly follows the IE, so it must have a success and failure path as shown in the event tree. If System A is successful in the ESD, then the end state is success. The translation to the event tree in Figure 2.9 shows that none of the other pivotal events after System A operates needs to have a downward branch for failure, since they do not affect the scenario. The end state of the first sequence is labeled SUCCESS as in the ESD as shown in Figure 2-11.

Once System A has failed (downward path on the event tree), the status of System B is questioned. The failure path of System B leads directly to a negative consequence labeled as ENDSTATE-1 (Sequence 4 listed to the left of the End State column) since no other mitigation options are available with both System A and System B failing. If System B is successful, then the status of System C is questioned, with its success (up path) leading to a SUCCESS end state, as shown in Figure 2-12 in Sequence 2.

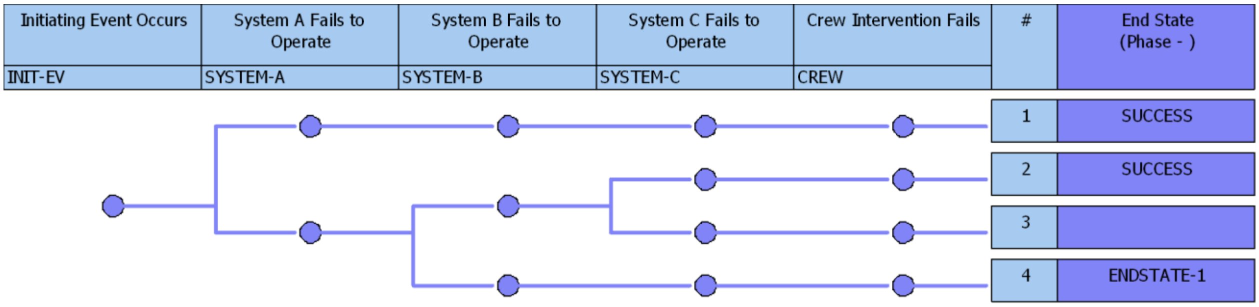


Figure 2-12. Step 2 in building example event tree.

The last step in building the event tree is to fill out the scenarios if System C fails after System A fails and System B is successful. The Crew Intervention event lessens the impact of the consequence per the ESD, and therefore the success path of the Crew Intervention event leads to a second, lesser, negative consequence labeled as ENDSTATE-2 in Figure 2-8. The failure of Crew Intervention leads to the same end state and if Systems A and B or A and C had failed, ENDSTATE-1. The final event tree is shown in Figure 2-8.

The resulting sequences of events for each of the end state in the example (Figure 2-8) are:

- *SUCCESS:*

  Sequence 1: Initiating Event Occurs * /System A fails to operate
  Sequence 2: Initiating Event Occurs * System A fails to operate * /System B fails to operate and /System C fails to operate

- *ENDSTATE-1:*

Sequence 4: Initiating Event Occurs * System A fails to operate * /System B fails to operate * System C fails to operate * Crew intervention fails
Sequence 5: Initiating Event Occurs * System A fails to operate * System B fails to operate

- *ENDSTATE-2:*

  Sequence 3: Initiating Event Occurs * System A fails to operate * /System B fails to operate * System C fails to operate * /Crew intervention fails

The objective is to develop a tractable model for the important paths leading from the IE to the end states. Generally, risk quantification is achieved by developing fault-tree models for the pivotal events in an event tree. This linking between an event tree and fault trees permits a Boolean equation to be derived for each event sequence. Event sequence quantification occurs when reliability data are used to numerically evaluate the corresponding Boolean equation.

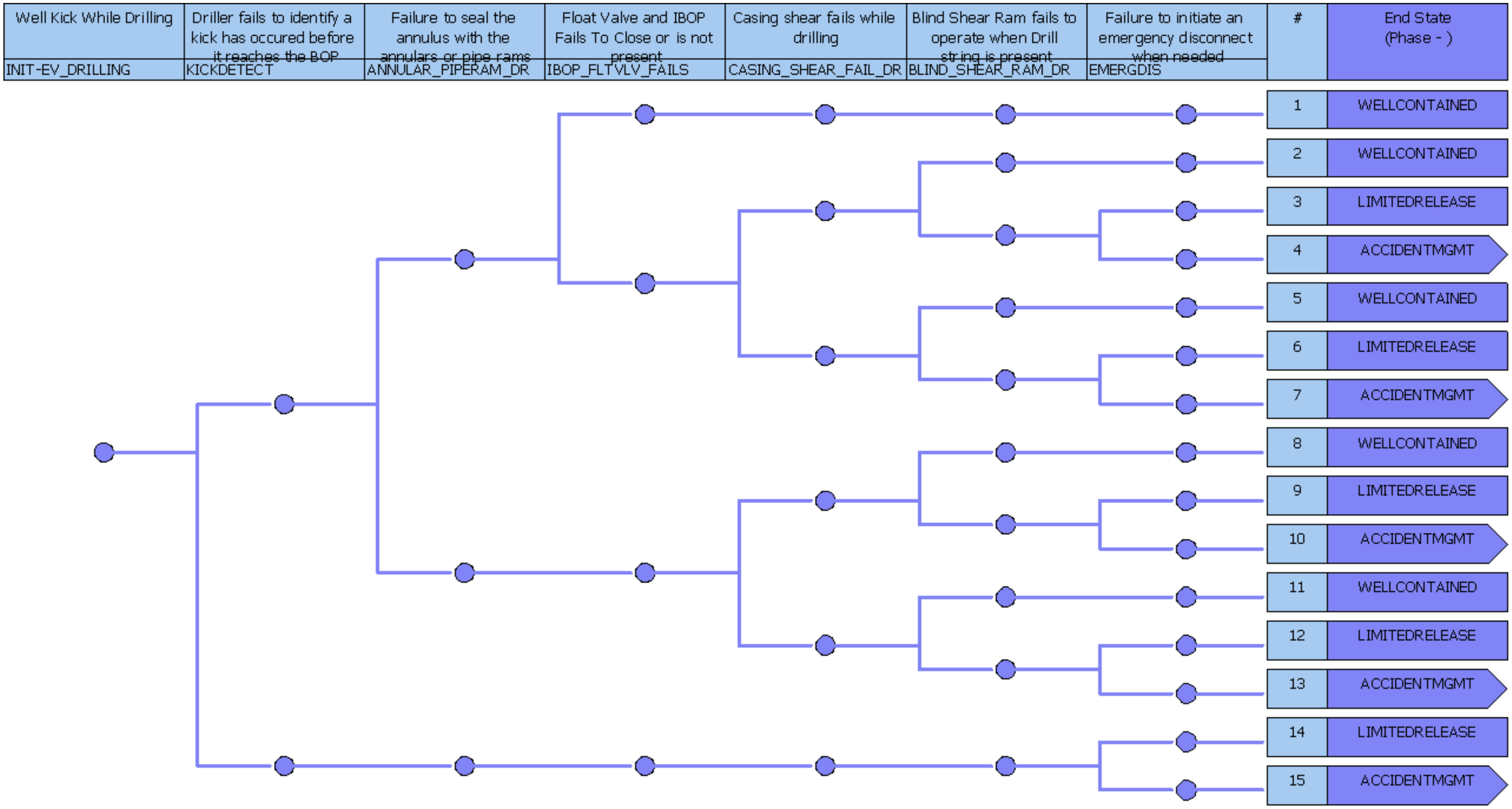


Figure 2-13. Event-tree structure for well kick while drilling.

2.2.4.2 Event-Tree Transfers. Figure 2-13 is a more complicated event tree corresponding to the Figure 2-6 ESD to estimate the frequency of an environmental release in response to a well kick while drilling. In this example, there are many different scenarios modeled. Going back to the original identification of IEs and development of ESDs in Sections 2.2.2 and 2.2.3, it was noted that some IEs may have the same sequence of events and can use the same ESD/event tree, albeit perhaps with different conditional probabilities possibly assigned to the pivotal events. Other IEs may have completely different scenarios that are modeled, or may be partially the same. For initiators with common sequences of events, the common sequences may be best developed in a stand-alone event tree and used as an event-tree transfer. Listed on the right side of Figure 2-13 under the end states is a transfer condition ACCIDENTMGMT shown as the symbol in Figure 2-14.

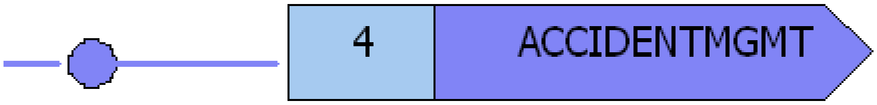


Figure 2-14. Event-tree transfer.

The ACCIDENTMGMT end state is a transfer to a second event tree. A transfer is used generally in the case when there may be common elements to multiple IEs or when an event tree gets very large and has distinct and different sequential processes. The event tree in Figure 2-6 is based on a well during drilling. Other IEs, such as a loss of position, may have different initial responses, but the responses after a failure of the BSR may be the same from the ESD in Figure 2-6. In the case of Figure 2-6, it was decided to use a separate event tree for the ROV intervention through relief well part of the ESD, in order to manage the size of the event tree and because other IEs may have that part in common. Accordingly, the ACCIDENTMGMT event tree is shown in Figure 2-15.

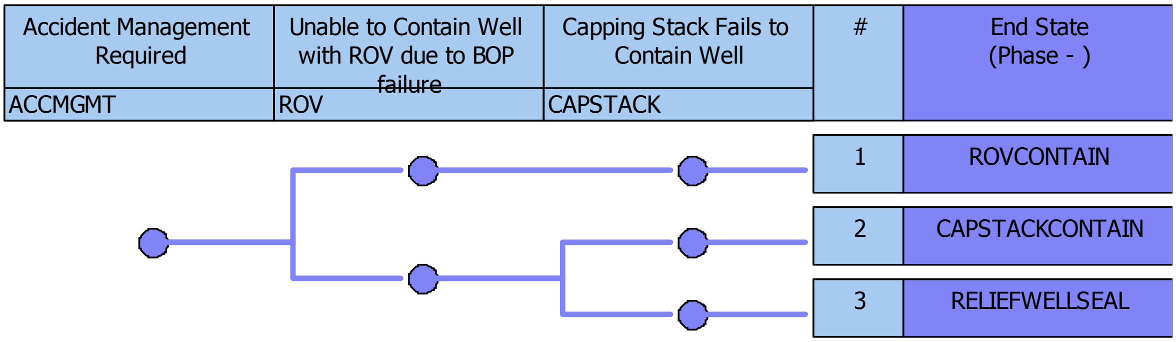

Accident Management Required
ACCMGMT
Unable to Contain Well with ROV due to BOP failure
ROV
Capping Stack Fails to Contain Well
CAPSTACK
#
End State (Phase -)
1
ROVCONTAIN
2
CAPSTACKCONTAIN
3
RELIEFWELLSEAL

Figure 2-15. ACCIDENTMGMT event tree.

The first event, ACCMGMT, is simply the entry point from the previous event tree, and does not show up in event sequences. The sequences of events from the previous event tree will continue in the ACCIDENTMGMT event tree to the end state where the sequences will contain all IE and pivotal event information from the initial event tree and the transfer event tree.

In Figure 2-13, the end states are WELLCONTAINED, which would represent success; where formation fluid has stopped flowing, the well is controlled, and a well kill program may be put in place. This end state is shown in Sequences 1, 2, 5, 8, and 11 in Figure 2-13. The remaining sequences are shown as the LIMITEDRELEASE end state or an event-tree transfer to the ACCIDENTMGMT event tree. These scenarios are associated with a failure of the BSR to contain the well; some formation fluid has risen above the BOP, and will reach the rig floor or environment. The sequences that have their end state transfer to the ACCIDENTMGMT event tree are longer duration events that end in that event tree with one of three end states possible. Each end state in ACCIDENTMGMT is different because the failures are time dependent in that an ROV intervention may happen earlier than well capping while the relief well is much longer term. Therefore, the duration of the releases will vary based on the type of successful intervention, and these are separated into distinct end states as shown in Figure 2-15. Ultimately, if no other well intervention techniques are successful, a relief well will be needed.

The end states discussed do separate out, to some degree, the magnitude of release as a measure of consequences. It may be desired to provide more deterministic estimates showing probabilistically the magnitude of release expected (i.e., barrels of oil released) as a function of probability. This can be done through the use of simulation as described in Section 2.3.

2.2.4.3 Multibranch Event Trees. The pivotal events in the previously described examples are all binary in that they only have paths related to success or failure of the event. In some cases, there may be multiple states or conditions that an event may be in, each with a different probability. Related to the ESD in Figure 2-6, there are different types of tubulars that may be present across the BOP when the shear rams are called on to work, depending on which operation is being performed (e.g., drilling or running casing). Since the casing shear and blind shear have different shearing capability, and there are some tubulars that are nonshearable, it is important to model the type of tubular present in the BOP at the time of shearing in order to achieve model accuracy. This may be done in the fault trees linked to the event tree (discussed in the next section), but fault trees addressing multiple possible configurations or conditions can be complicated. Another way to address multiple conditions is the event-tree structure using a multibranch node in the event tree, as shown in Figure 2-16.

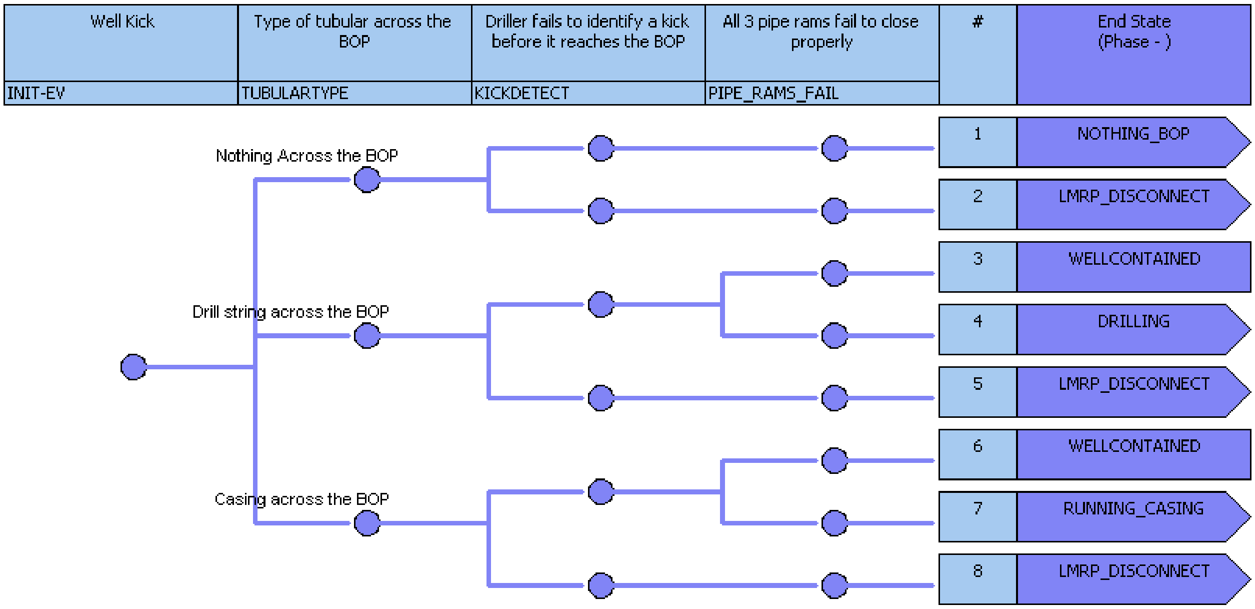


Figure 2-16. Multibranch node in an event tree.

In Figure 2-16, the ESD from Figure 2-7 has been broken up into three operational conditions: nothing across the BOP, drill pipe across the BOP, and casing across the BOP. The first node in the event tree has three branches with each representing one of the conditions. Each branch of the event tree for this event can be assigned a probability of that event being true, and together they would add up to 1.0 since one of the conditions has to be true if the branches cover all possible conditions. Using this approach logically changes the downstream paths if the event tree is fully developed. For instance, if there is nothing across the BOP (top branch of first node) when containment is required (top branch), then the status of the pipe rams do not need to be questioned since they not capable of containment in this condition. A transfer event tree with specific sequences can be developed for each operational condition in the first node and a transfer end state can be used as shown in Figure 2-7.

2.2.5 Fault-Tree Modeling

In many problems of practical interest, estimating pivotal event frequencies actuarially is not possible, even if the events occurred independently, because they do not happen often enough to permit useful statistical analysis. In general, pivotal events are not independent, so even if their frequencies (or probabilities) could be quantified actuarially, those results could not be combined straightforwardly to obtain a sequence frequency. Therefore, such events must be modeled synthetically[d] (i.e., functional failures must be expressed in terms of system failures, system failures in terms of component failures, and component failures in terms of their causes) in sufficient detail that the lowest-level elements of the model can be quantified. Having done that, we then work our way back up to a synthetic estimate of pivotal event probability (or conditional frequency), conditional on its role in each scenario of interest, to finally quantify the top-event frequencies themselves. For PRAs, pivotal events are typically modeled using fault trees. More information, beyond what is in this guide, can be found in [2-1].

2.2.5.1 System Success Criteria. Prior to development of the pivotal event fault trees, success criteria are needed to define satisfactory performance in terms of the function included in the event tree. System success criteria impose operating requirements on the systems needed to successfully perform a particular function. The duration needed for that function determines the system operating time. Once the success criteria for a function have been established, top-event fault-tree logic is established from the Boolean complement of the success criteria (e.g., at least one of two pipe rams must fail to close and seal around the drill pipe when demanded). Success criteria should be clearly defined. All assumptions and supporting information used to define the success criteria should be listed in the documentation (i.e., what is considered to constitute system success needs to be explicitly stated). Some examples of success criteria are:

- The BSR must close and shut in the well on demand
- At least one of two pipe rams must close and seal around the drill pipe when demanded
- At least four of six thrusters (two forward, two aft) must operate to maintain station-keeping under calm (specified) environmental conditions
- At least six of six thrusters must operate to maintain station-keeping under extreme (specified) environmental conditions.

The last two examples show that success criteria may be dependent on external factors and may need to be discretely modeled. In addition, again referring to the last two examples, the conditions may require specific thrusters to be available (e.g., the 4 out of 6 case may require 2 forward and 2 aft thrusters and any such specific details should be included as part of the success criteria). The development of success criteria into logic representations leans heavily on engineering analysis: physical simulation of system behavior in specified conditions, determination of time available for crew actions, and determination of the severity of the consequences associated with scenarios. Behind every logic model is another body of modeling whose results are distilled into the logical relationships pictured in the scenario model. Assignment of system states into “success” or “failure” depends on such modeling, as does classification of scenarios into consequence categories.

[d] The term “synthetic” is used here to refer to modeling a complex event in terms of its contributors. For example, we “synthesize” an estimate of the probability of a complex event by combining estimates of the probabilities of its contributors.

2.2.5.2 Modeling Pivotal Events. Complex pivotal events are usually modeled using fault trees. A fault tree is a picture of a set of logical relationships between more complex (more aggregated) events, such as system-level failures, and more basic (less aggregated) events, such as component-level failures. Fault-tree modeling is applicable not only to modeling of hardware failures, but also other complex event types as well, including descriptions of the circumstances surrounding software response and crew actions.

Pivotal events must be modeled in sufficient detail to support valid quantification of scenarios. As a practical matter, the model must reach a level of detail at which data are available to support quantification of the model's parameters.

Additionally, much of the time, pivotal events are not independent of each other, or of the IEs; the modeling of pivotal events must be carried out in such a way that these dependencies are captured properly. For example, pivotal events corresponding to system failure may have some important underlying causes in common (e.g., support systems). If the purposes of the PRA are to be served—if such underlying causes are to be identified and addressed—it is imperative to capture such dependencies in the scenario model. If pivotal events were known to be independent of each other, their probabilities could be combined multiplicatively, and there would be less reason to analyze them in detail. Because pivotal events often share functional support systems or other dependencies, their modeling in some detail is important.

Functionally, a fault tree is a deductive logic model where a top event, usually a system failure, is postulated, and reverse paths are developed to gradually link this top event with all subsystems, components, software errors, or human actions (in order of decreasing generality) that can contribute to the top event, down to those whose basic probability of failure (or success) is known and can be directly used for quantification. Graphically, a fault tree at its simplest consists of blocks (e.g., rectangles or circles) containing descriptions of failure modes and binary logic gates (e.g., union or intersection) that logically link basic failures through intermediate-level failures to the top event. Figure 2-17 depicts a very simple fault-tree structure.

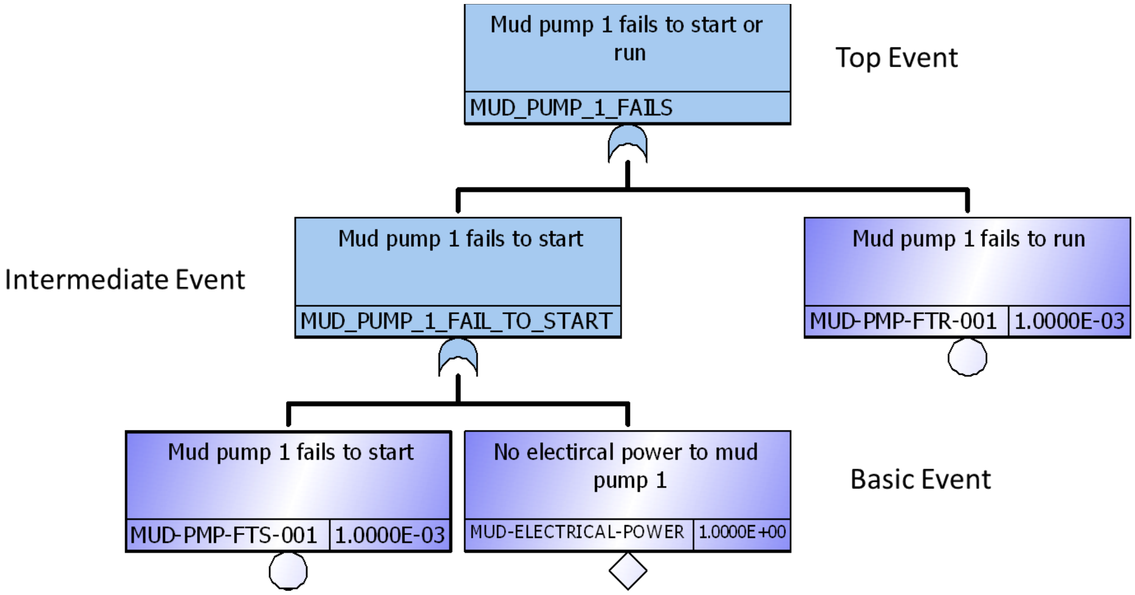


Figure 2-17. Typical fault-tree structure and symbols.

Fault trees are constructed to define all significant failure combinations, called cut sets that lead to the top event. The result of a Boolean reduction of the fault tree results in combinations of failures that are the minimum set(s) required to result in the top event and are called minimal cut sets.

Ultimately, fault trees are graphical representations of Boolean expressions representing the minimal cut sets. For the fault tree in Figure 2-17, there are three minimal cut sets:

1. MUD-PMP-FTR-001
2. MUD-PMP-FTS-001
3. SYSTEM-A-PUMP-PWR.

The corresponding Boolean equation for the fault tree is:

$$SYSTEM\text{-}A = MUD\text{-}PMP\text{-}FTR\text{-}001 \cup MUD\text{-}PMP\text{-}FTS\text{-}001 \cup SYSTEM\text{-}A\text{-}PUMP\text{-}PWR \qquad (2\text{-}3)$$

More detail on minimal cut sets and the Boolean reduction that are the results of the fault tree are shown in Section 4.

2.2.5.3 Fault-Tree Considerations. Developing a fault tree requires several considerations including:

- Identifying the objective and scope of the analysis
- Determining the level of detail
- Setting ground rules and naming conventions.

The objective and scope of the fault tree, in the context of a PRA analysis, is normally defined when the event sequences are being developed by constructing the ESDs/event trees. The critical systems/events required to respond to an IE are assessed by the processes in previous sections, and incorporated as pivotal events in the event tree(s). These pivotal events become top events for the fault trees and should be worded in language specific enough to highlight the failure mode of the event being analyzed based on the success criteria.

Simply labeling the top event as “System A Fails” is generally inadequate as System A may have different failure modes, and the objective of the analysis may only require specific ones to be modeled. If extraneous failures are included in the analysis that do not contribute to the analysis objective, the results of the analysis will be erroneous. For instance, the Emergency Disconnect on a mobile offshore drilling unit (MODU) has several functions, including separation of the LMRP from the BOP and triggering the autoshear function on the BOP. The separation from the well is performed in an emergency situation for personnel safety to allow the MODU to move clear of the well. The intent of the autoshear function is to seal the well and prevent a hydrocarbon release. From the ESD developed in Figure 2-6, the objective of the analysis is to estimate the probability of a hydrocarbon release, so when developing the top event, only the contributors to the failure of the autoshear function of the Emergency Disconnect need to be included, and the top event should be worded with that failure mode.

Defining the scope of the analysis includes understanding the initial configuration/operation of the system being analyzed. The initial state of the system will describe which components are active, which are in a standby state, and any external conditions: for example, if failure of the BOP BSR is being analyzed, it is important to identify what operation is being performed, such as, e.g., running casing. The initial state of the components will determine the applicable failure modes for those components. A pump that is active may fail to operate while a pump in standby may fail to start or fail to operate. In cases where a component is in standby, the analysis may need to account for human error if manual activation is required for the system to start.

The level of detail on the causes resulting in the top event for a PRA analysis should be based on the level at which data is available, the objective of the analysis, and the interdependencies between systems and operations. Data analysis is discussed in Section 3 in detail, but generally data can be found at the major component level (e.g., pumps, valves, electrical busses, etc.) from a variety of sources. Going beyond the level at which data are available may result in an unquantifiable fault tree. The objectives of the analysis must also be considered in determining the resolution of the fault tree. For a fault-tree analysis of a BOP, the analysis could be performed at the yellow/blue pod level; or, if the analyst needs more detail down to the hydraulic component level in order to account for cross-connect ability, then that level may be modeled. Interdependencies, such as cross-connect capability, often drive the analysis down to the component level of detail.

The last consideration is setting up modeling ground rules in order to ensure consistency across the PRA. Establishing a naming convention for fault-tree gates and particularly for basic events is necessary to be able to easily read cut sets and results from the analysis. Cut set naming schemes for the basic events may include:

- The operation being performed (e.g., drilling)
- The system the component belongs to (BOP)
- The subsystem the component belongs to (e.g., yellow pod)
- The component (e.g., shuttle valve)
- The failure mode (e.g., Fails to transfer)
- A unique identifier for the valve (usually from a drawing [e.g., SV01]).

There is usually a character limit to the size of the basic event name, so abbreviations must be used for the above items such as BOP for blowout preventer, YPO for yellow pod, etc. As a minimum, the system, component, failure mode, and a unique identifier should be used when the naming scheme is developed. The overall naming scheme typically has a form like XXX-YYY-ZZZ-DDDDD, where XXX corresponds to the system, YYY is the component, etc. The abbreviations for each are developed before modeling begins with the exception of the unique identifiers. The failure modes for active components should correspond to active failures and not a failed condition. For example, for a valve that is initially open and fails when commanded to close, the best way to express the failure mode is “fails to close” rather than “fails closed.” In the “fails closed” case, it is not clear what the initial condition of the valve is; was the valve open and did not close when commanded? Or was the valve initially closed and failed in that state when commanded to open? Using the active word “to” in “fails to close” implies that the valve is initially open.

A well-thought-out naming scheme for basic events is essential to avoid duplication of names for different events in different fault trees, particularly if multiple analysts are involved. If duplication does exist, the results produced for those events could be erroneous. Examples of typical naming conventions for failure modes and components are provided in Appendix A.

There can be some special events that are adapted to the naming scheme used for components or they may have their own separate scheme. For example, environmental conditions do not have a system or unique identifier associated with them; therefore, these conditions have a separate naming scheme developed for just those types of events.

Gate naming schemes may be more free form since gates are not shown in the results. However, a consistent naming scheme for gates is advisable, in order to ensure that each gate is named uniquely, and to avoid having gates with different logic and the same name in different fault trees.

2.2.5.4 Fault-Tree Symbols. Starting with the top event, the fault tree is developed by deductively determining the cause of the previous fault, continually approaching finer resolution until the limit of resolution is reached. In this fashion, the fault tree is developed from the system end point backward to the failure source. The limit of resolution is reached when fault-tree development below a gate consists only of basic events (i.e., faults that consist of component failures, faults that are not to be further developed, phenomenological events, support system faults that are developed in separate fault trees, software errors, or human actions). The logic of the fault tree is represented by symbols used for fault tree gates and basic events. The most common types of gates are shown in Figure 2-18, along with a description of the logic for each. Appendix B gives a detailed explanation of how each gate is used and quantified. Other gate types such as “NOR” or “INHIBIT” exist, but are rarely used.

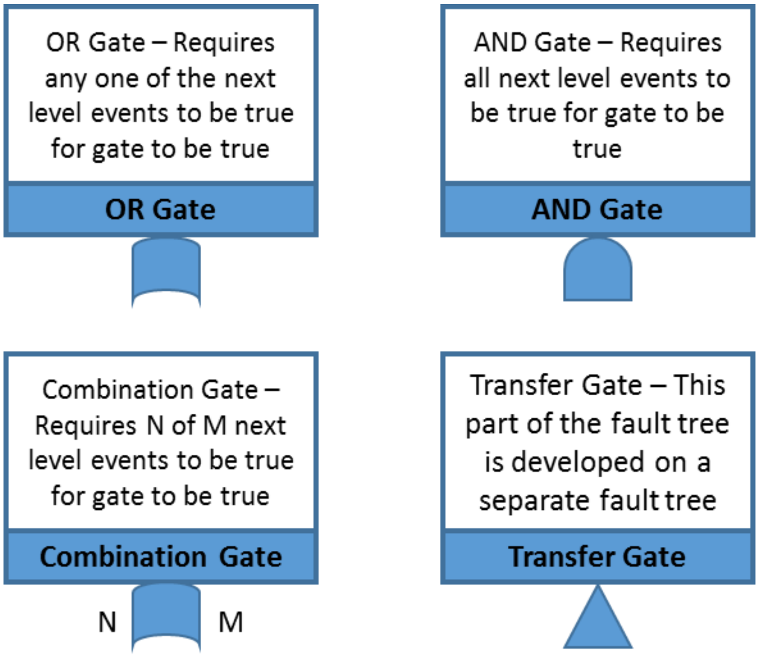

OR Gate – Requires any one of the next level events to be true for gate to be true
OR Gate
AND Gate – Requires all next level events to be true for gate to be true
AND Gate
Combination Gate – Requires N of M next level events to be true for gate to be true
Combination Gate
N
M
Transfer Gate – This part of the fault tree is developed on a separate fault tree
Transfer Gate

Figure 2-18. Commonly used fault-tree gates.

The most common basic event types are shown in Figure 2-19 with a description of what they represent.

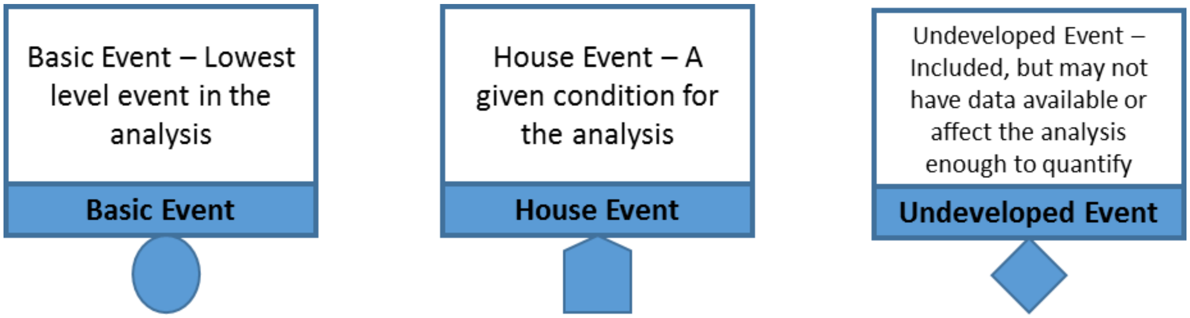

Basic Event – Lowest level event in the analysis
Basic Event
House Event – A given condition for the analysis
House Event
Undeveloped Event – Included, but may not have data available or affect the analysis enough to quantify
Undeveloped Event

Figure 2-19. Commonly used basic-event types.

House events are often used in fault-tree analysis as switches to turn logic on and off or represent a condition. If used as a switch, their probability is usually quantified as unity or zero, they require no reliability data input. House events are also used to indicate conditional dependencies.

2.2.5.5 Simple Fault-Tree Example. Using the steps described above, and going back to the example event tree provided in Figure 2-13, a simplified example of fault-tree construction is developed for the "ANNULAR" event (second event after the initiator in Figure 2-13), which represents the failure of the annular preventer to block the annulus through the BOP. A simplified drawing of a BOP is shown in Figure 2-20.

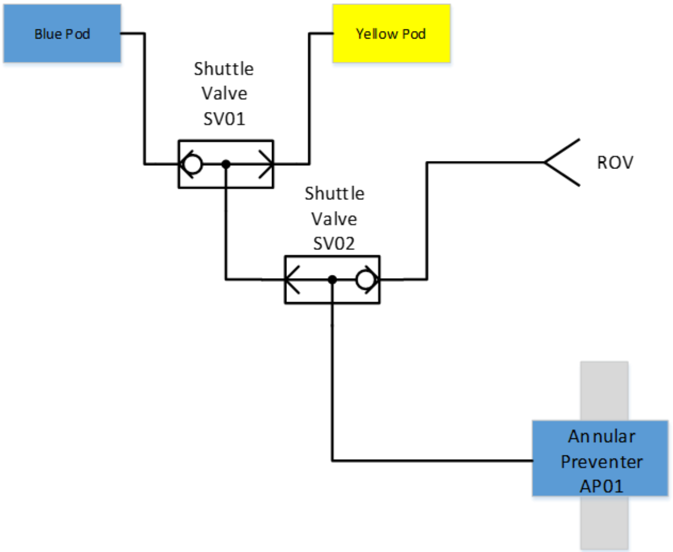


Figure 2-20. Simple BOP schematic.

The top event of the fault tree, as stated in the event tree, is "annular preventer fails to close prior to the kick reaching the BOP, or pressure is beyond the design of the annular." The wording of the top event implies that the initial condition for the annular preventer is "open," and for success of this event, the annular preventer must close in order to prevent flow past the BOP. The simplified *example* diagram in Figure 2-20 shows that both the blue and yellow pods, used for control, are connected to the annular preventer. Either one is adequate to close the annular preventer and, in this example, it is assumed that a crosstie exists. To switch pods, a manual action by the Driller is required.

To develop the next level down in the fault tree, the design and operation is reviewed. In this example, one of the pods must provide hydraulic fluid to the preventer, the preventer itself must close, and the pressure must stay below the design pressure of the annular.

The failure of the annular preventer (BOP-CYL-FTC-AP01) is a singular failure point, so it is included under an "OR" gate, as shown in Figure 2-21. For the purposes of this example, the basic event related to the pressure of the annular has been left as an undeveloped event and is also included under the OR gate (ANNULAROVERPRESSURE). The failure of the pods includes multiple events and combinations of events that must fail to satisfy the top event. Therefore, an intermediate "AND" gate is needed to develop this event (BOTHPODSFAIL) further. The left input (YELLOWPODFAILS) to the AND gate is an intermediate gate for the operating yellow pod, while the blue pod (BLUEPODFAILS) is the standby pod and addressed on the right-hand side of the AND gate. For convenience, selected portions of the blue and yellow pods (gate names: BLUEPODCOMMON, YELLOWPODCOMMON), have been made transfer events to allow these portions of the fault tree to be used with the blind shear, pipe, and casing shear rams. The transfer for each is also shown in Figure 2-21. For each pod, an OR gate is used with the inputs broken down into the pods themselves and the hydraulic paths from the pods. From Figure 2-20, the yellow path is aligned to the annular and the shuttle valves are in position to permit flow, so the only applicable failure mode considered is external valve leakage. Since the flow passes through both valves, both are included (BOP-SHV-LKG-SV01, BOP-SHV-LKG-SV02).

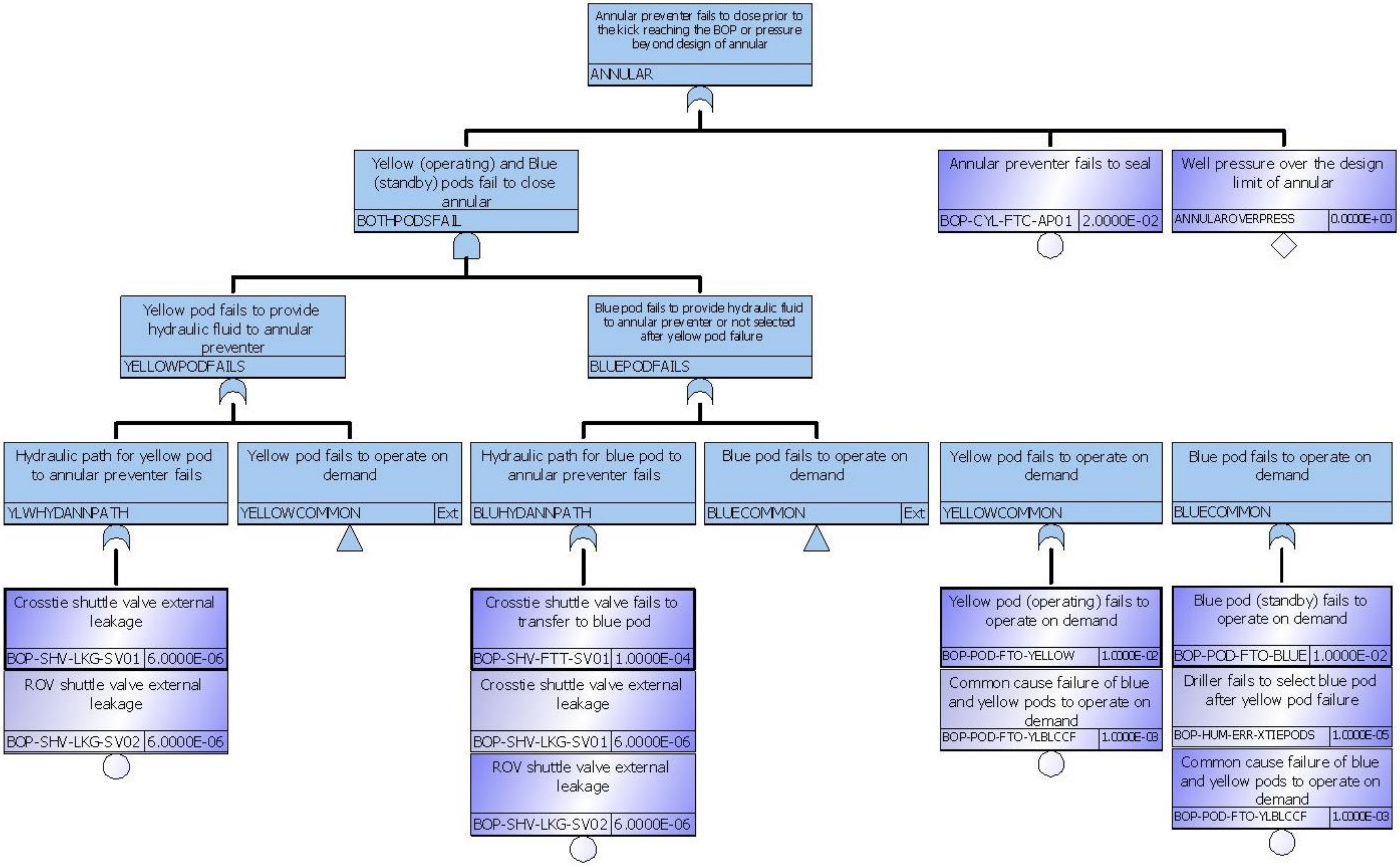

Annular preventer fails to close prior to the kick reaching the BOP or pressure beyond design of annular
ANNULAR
Yellow (operating) and Blue (standby) pods fail to close annular
BOTHPODSFAIL
Annular preventer fails to seal
BOP-CYL-FTC-AP01
2.0000E-02
Well pressure over the design limit of annular
ANNULAROVERPRESS
0.0000E+00
Yellow pod fails to provide hydraulic fluid to annular preventer
YELLOWPODFAILS
Blue pod fails to provide hydraulic fluid to annular preventer or not selected after yellow pod failure
BLUEPODFAILS
Hydraulic path for yellow pod to annular preventer fails
YLWHYDANNPATH
Yellow pod fails to operate on demand
YELLOWCOMMON
Ext
Hydraulic path for blue pod to annular preventer fails
BLUHYDANNPATH
Blue pod fails to operate on demand
BLUECOMMON
Ext
Yellow pod fails to operate on demand
YELLOWCOMMON
Blue pod fails to operate on demand
BLUECOMMON
Crosstie shuttle valve external leakage
BOP-SHV-LKG-SV01
6.0000E-06
ROV shuttle valve external leakage
BOP-SHV-LKG-SV02
6.0000E-06
Crosstie shuttle valve fails to transfer to blue pod
BOP-SHV-FTT-SV01
1.0000E-04
Crosstie shuttle valve external leakage
BOP-SHV-LKG-SV01
6.0000E-06
ROV shuttle valve external leakage
BOP-SHV-LKG-SV02
6.0000E-06
Yellow pod (operating) fails to operate on demand
BOP-POD-FTO-YELLOW
1.0000E-02
Common cause failure of blue and yellow pods to operate on demand
BOP-POD-FTO-YLBLCCF
1.0000E-03
Blue pod (standby) fails to operate on demand
BOP-POD-FTO-BLUE
1.0000E-02
Driller fails to select blue pod after yellow pod failure
BOP-HUM-ERR-XTIEPODS
1.0000E-05
Common cause failure of blue and yellow pods to operate on demand
BOP-POD-FTO-YLBLCCF
1.0000E-03

Figure 2-21. Basic fault tree.

The blue pod side needs to be treated differently because it is in a standby state. Because the pods are manually selected, a basic event for the Driller failing to select the blue pod after the yellow pod fails is added (BOP-HUM-ERR-XTIEPODS). On the hydraulic path intermediate event, a basic event for the crosstie shuttle valve failing to transfer to the correct position is added (BOP-SHV-FTT-SV01). From Figure 2-21, it should be noted that several events are included on both the yellow and blue pods, including the two shuttle valve external leakages and the common-cause failure (CCF) of both the yellow and blue pods. In a fault tree, events or gates may be used in multiple areas and when the fault tree is solved, the cut sets produced will be reduced and will not contain any duplicates.

The result of solving the ANNULAR fault tree in Figure 2-21 is shown in Table 2-3. Using the logic of the fault tree, the inputs are reduced to the "minimal cut sets" that result in the top event. Each minimal cut set is a conjunction of conditions that are sufficient to cause the top event, and are necessary in the sense that if any of the constituent basic events were not true, the top event would not be true. In Table 2-3, Cut Sets 1 through 5 are all single basic events that would result in the top event, while Cut Sets 6, 7, and 8 are double failures. When the basic events are assigned values, a ranked listing can be produced.

Table 2-3. ANNULAR fault-tree minimal cut sets.

No.	Cut Set	Description
1	BOP-SHV-LKG-SV01	Crosstie shuttle valve external leakage
2	BOP-SHV-LKG-SV02	ROV shuttle valve external leakage
3	BOP-CYL-FTC-AP01	Annular preventer fails to seal
4	BOP-POD-FTO-YLBLCCF	CCF of blue and yellow pods
5	ANNULAROVERPRESS	Well pressure over the design limit of annular
6	BOP-POD-FTO-BLUE	Blue pod (standby) fails to run
	BOP-POD-FTO-YELLOW	Yellow pod (operating) fails to run
7	BOP-POD-FTO-YELLOW	Yellow pod (operating) fails to run
	BOP-SHV-FTT-SV01	Crosstie shuttle valve fails to transfer to blue pod
8	BOP-HUM-ERR-XTIEPODS	Driller fails to select blue pod after yellow pod failure
	BOP-POD-FTO-YELLOW	Yellow pod (operating) fails to run

2.2.5.6 Modeling Common Cause. For complex systems with redundancy, CCF of like components can be a major risk contributor. The specifics on how common cause is evaluated and quantified is shown in Section 3. This section discusses the options on how it should be represented in the fault-tree model as basic events.

In Figure 2-21, the basic fault-tree example of the annular preventer, common cause was modeled for the blue/yellow pods designated by the basic event name ending in CCF. In this example, the basic event BOP-POD-FTO-YLBLCCF is included under the intermediate gates for both the yellow pod and the blue pod. Since the yellow and blue pods are under an AND gate, and both are required to fail in order for the top event to be true, this one basic event satisfies that condition. The result of solving the fault tree is shown in Table 2-3, where Cut Set 4 is a single common-cause event. It is logically possible to put common-cause events at or near the top of the fault tree, but repeating a common-cause basic event wherever the effect is appropriate, in this case with the yellow and blue pods, is the preferred method of modeling common cause, because it maintains the relationship of the basic event to the intermediate events. When fault-tree transfers are needed, this practice can be important to ensure accuracy in the model. For systems that have three or more redundant components / systems (e.g., dynamic positioning thrusters), this may lead to multiple common-cause events under each thruster as shown in Figure 2-22 (shown as stacked basic events for simplicity). In Figure 2-22, the Dynamic Positioning System (DPS) Thruster 1 has been filled out with all common-cause terms, and the appropriate ones involving Thruster 1 have also been included for the other three thrusters.

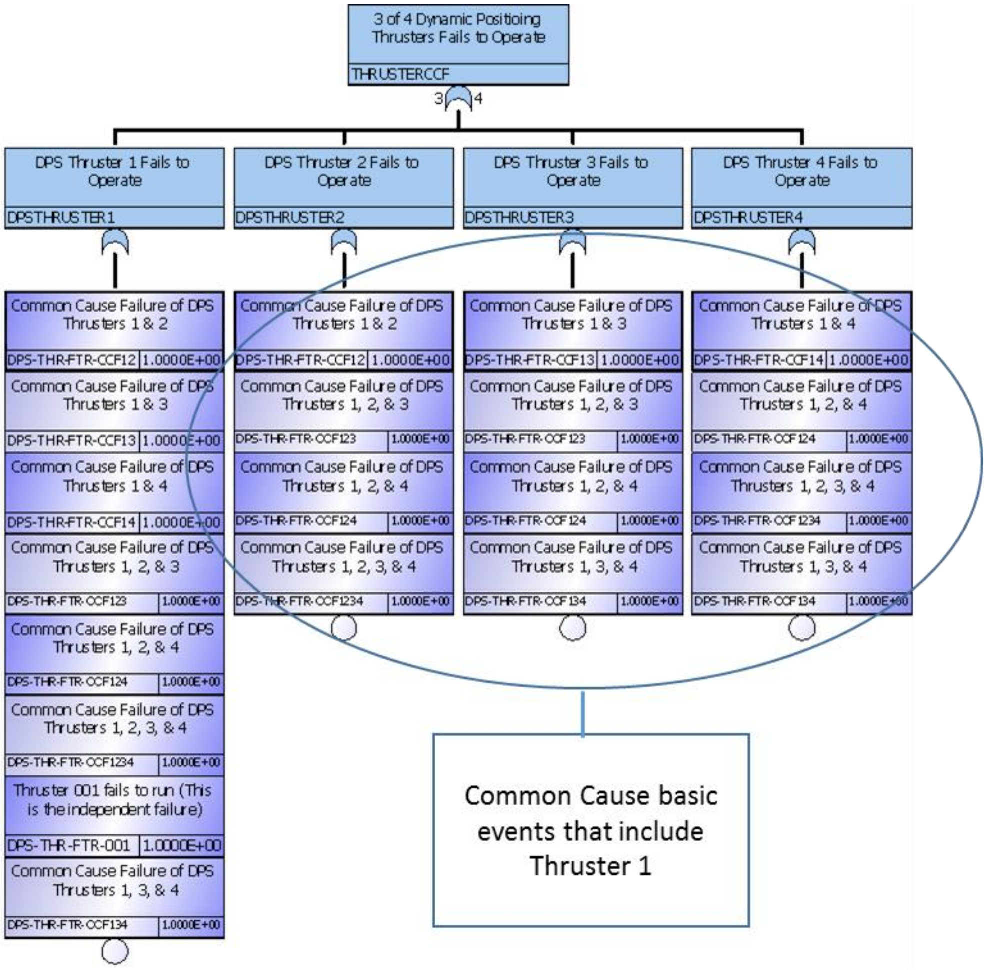

3 of 4 Dynamic Positioing Thrusters Fails to Operate
THRUSTERCCF
3
4
DPS Thruster 1 Fails to Operate
DPSTHRUSTER1
DPS Thruster 2 Fails to Operate
DPSTHRUSTER2
DPS Thruster 3 Fails to Operate
DPSTHRUSTER3
DPS Thruster 4 Fails to Operate
DPSTHRUSTER4
Common Cause Failure of DPS Thrusters 1 & 2
DPS-THR-FTR-CCF12
1.0000E+00
Common Cause Failure of DPS Thrusters 1 & 3
DPS-THR-FTR-CCF13
1.0000E+00
Common Cause Failure of DPS Thrusters 1 & 4
DPS-THR-FTR-CCF14
1.0000E+00
Common Cause Failure of DPS Thrusters 1, 2, & 3
DPS-THR-FTR-CCF123
1.0000E+00
Common Cause Failure of DPS Thrusters 1, 2, & 4
DPS-THR-FTR-CCF124
1.0000E+00
Common Cause Failure of DPS Thrusters 1, 2, 3, & 4
DPS-THR-FTR-CCF1234
1.0000E+00
Thruster 001 fails to run (This is the independent failure)
DPS-THR-FTR-001
1.0000E+00
Common Cause Failure of DPS Thrusters 1, 3, & 4
DPS-THR-FTR-CCF134
1.0000E+00
Common Cause Failure of DPS Thrusters 1 & 2
DPS-THR-FTR-CCF12
1.0000E+00
Common Cause Failure of DPS Thrusters 1, 2, & 3
DPS-THR-FTR-CCF123
1.0000E+00
Common Cause Failure of DPS Thrusters 1, 2, & 4
DPS-THR-FTR-CCF124
1.0000E+00
Common Cause Failure of DPS Thrusters 1, 2, 3, & 4
DPS-THR-FTR-CCF1234
1.0000E+00
Common Cause Failure of DPS Thrusters 1 & 3
DPS-THR-FTR-CCF13
1.0000E+00
Common Cause Failure of DPS Thrusters 1, 2, & 3
DPS-THR-FTR-CCF123
1.0000E+00
Common Cause Failure of DPS Thrusters 1, 2, & 4
DPS-THR-FTR-CCF124
1.0000E+00
Common Cause Failure of DPS Thrusters 1, 3, & 4
DPS-THR-FTR-CCF134
1.0000E+00
Common Cause Failure of DPS Thrusters 1 & 4
DPS-THR-FTR-CCF14
1.0000E+00
Common Cause Failure of DPS Thrusters 1, 2, & 4
DPS-THR-FTR-CCF124
1.0000E+00
Common Cause Failure of DPS Thrusters 1, 2, 3, & 4
DPS-THR-FTR-CCF1234
1.0000E+00
Common Cause Failure of DPS Thrusters 1, 3, & 4
DPS-THR-FTR-CCF134
1.0000E+00
Common Cause basic events that include Thruster 1

Figure 2-22. Common-cause modeling for a three of four system (only complete for Thruster 1).

For highly redundant systems, PRA analysts sometimes include only a global common-cause term, one that accounts for all like components that are failing. This is done for simplicity and can be a good approximation, as the global common-cause terms are often the dominant contributor for CCF. If this approximation is considered appropriate, the single basic event can be included under each system as previously described, or as a single basic event at the same fault-tree level as the AND or N-of-M gate modeling the redundancy.

In some special situations, failure of highly redundant systems or functions may involve specific combinations of equipment that are not symmetric. An example of this may be thruster operation where a vessel may have three forward and three aft thrusters, and the success criterion is that at least one forward and one aft thruster need to be available (i.e., loss of all forward or all aft thrusters causes loss of position). In this case, it may be appropriate to discard the common-cause terms that include both forward and aft thrusters (except for the global term) as they may not be large contributors, while the all-forward-thrusters or all-aft-thrusters terms may affect the results.

2.2.5.7 Modeling Conditionality. The house event, shown in Figure 2-19, is used to show whether a particular condition that affects the analysis is present or not. This is often used as a switch by the analyst to turn a condition on (set the event probability to 1.0) or off (set the event probability to 0.0), in order to see what the effect is on the results. In other cases, there may be a condition that exists a fraction of the time, and the logic that satisfies the top event changes depending on whether the condition is present or not. A case like this was assumed for the BLIND_SHEAR_RAM_DR top event in Figure 2-13. This top event considers that a tool joint may be in the plane of the BSR some fraction of the time, requiring action by the Driller to mitigate if the BSR is to be successful. The condition of a tool joint being present is shown as a house event, since it is not a failure. The simplified fault tree for this event is shown in Figure 2-23.

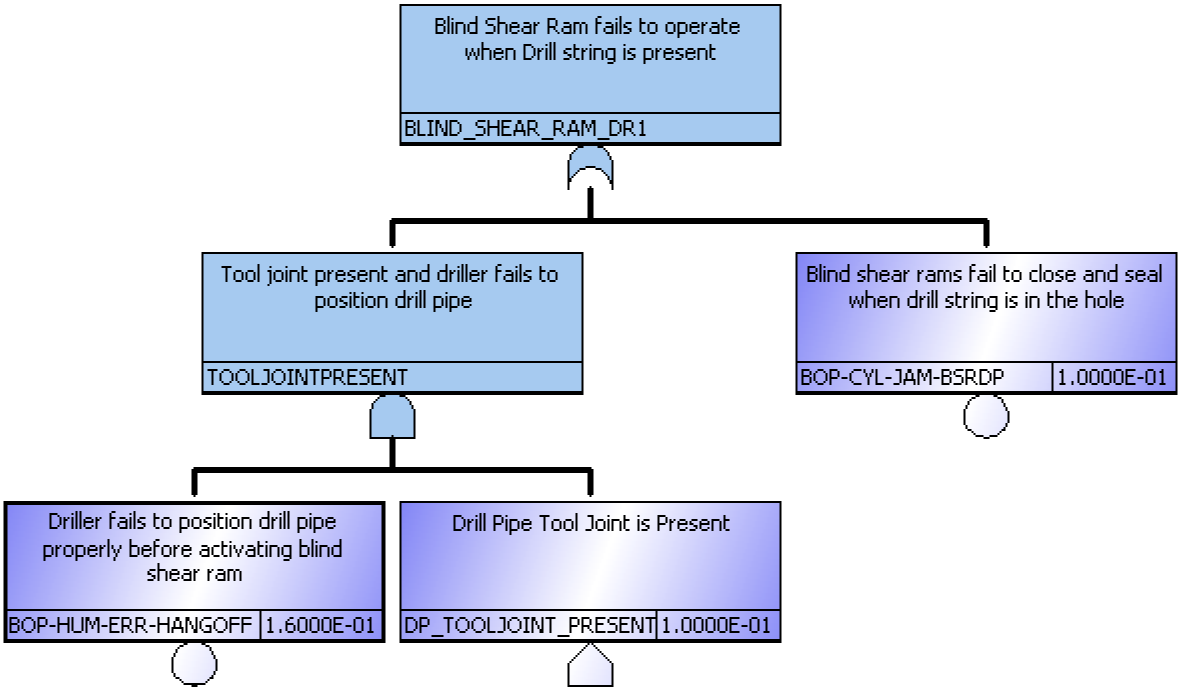

Blind Shear Ram fails to operate when Drill string is present
BLIND_SHEAR_RAM_DR1
Tool joint present and driller fails to position drill pipe
TOOLJOINTPRESENT
Blind shear rams fail to close and seal when drill string is in the hole
BOP-CYL-JAM-BSRDP
1.0000E-01
Driller fails to position drill pipe properly before activating blind shear ram
BOP-HUM-ERR-HANGOFF
1.6000E-01
Drill Pipe Tool Joint is Present
DP_TOOLJOINT_PRESENT
1.0000E-01

Figure 2-23. Modeling conditionality in a fault tree.

As can be seen in Figure 2-23, an intermediate event (AND gate) models the condition that a tool joint is present, and the Driller fails to position the string to allow the drill string to be sheared by the BSR. The failure of the BSR itself, BOP_CYL_JAM_BSRDP, may occur whenever there is drill pipe across the BOP.

When modeling these type of conditions, the analyst must ensure that the dependence is maintained. In this example, the failure of the BSR is always applicable when drill pipe is across the BOP, so it does not need to be conditioned further. In some situations, a different approach is needed. Consider a top event that models a loss of position, as in Figure 2-24. There are different success criteria depending on the weather condition (calm or high weather), as shown by the events requiring a different number of thrusters depending on the weather condition. In this case, the weather condition for each success criterion is modeled to include a house event with the probability of that condition. Because only one or the other weather condition can be applicable at any given time, the sum of the probabilities will be 1.0. If one condition occurs at a very small probability, it may be acceptable to not include the compliment. For example, if high weather occurs with a probability of 0.005, it may acceptable to assume calm weather is 1.0 and not to include the house event if the effect on results would be negligible.

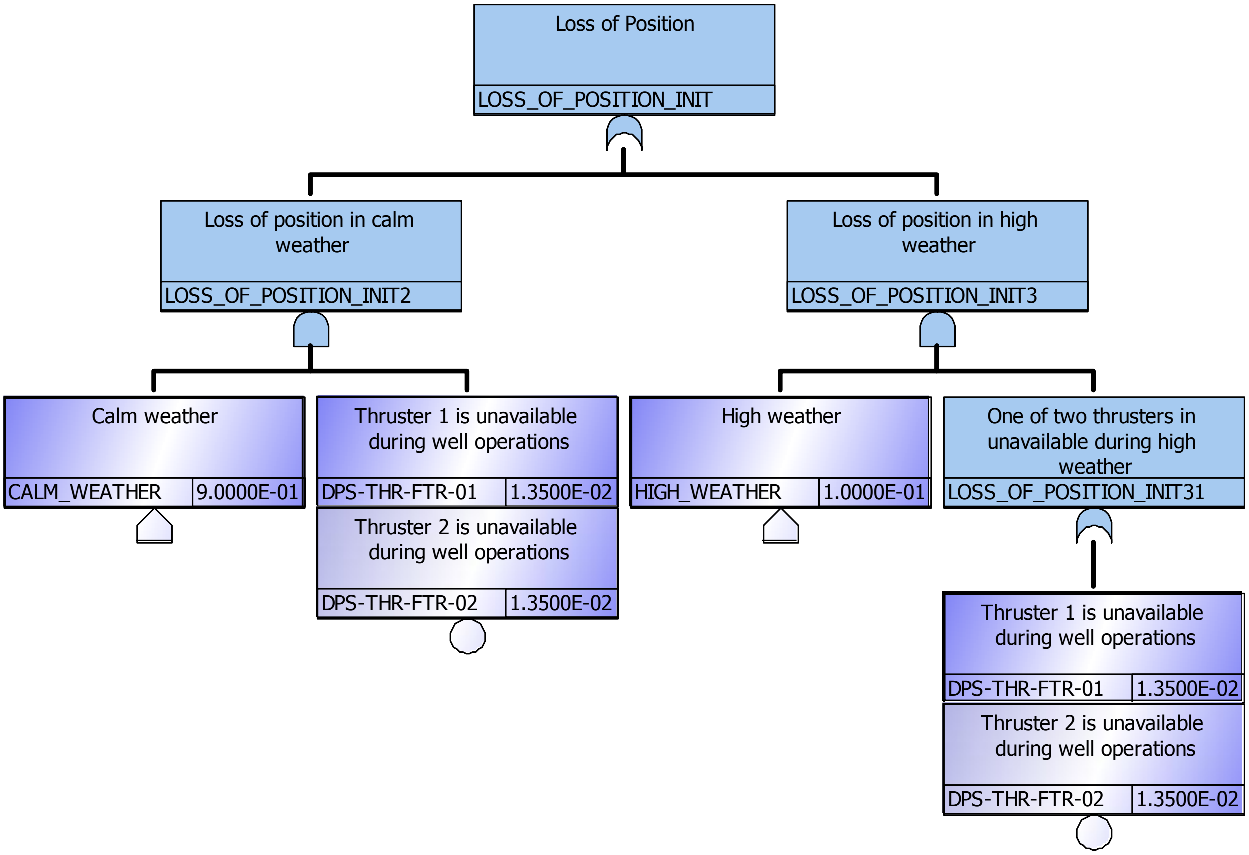

Loss of Position
LOSS_OF_POSITION_INIT
Loss of position in calm weather
LOSS_OF_POSITION_INIT2
Loss of position in high weather
LOSS_OF_POSITION_INIT3
Calm weather
CALM_WEATHER
9.0000E-01
Thruster 1 is unavailable during well operations
DPS-THR-FTR-01
1.3500E-02
Thruster 2 is unavailable during well operations
DPS-THR-FTR-02
1.3500E-02
High weather
HIGH_WEATHER
1.0000E-01
One of two thrusters in unavailable during high weather
LOSS_OF_POSITION_INIT31
Thruster 1 is unavailable during well operations
DPS-THR-FTR-01
1.3500E-02
Thruster 2 is unavailable during well operations
DPS-THR-FTR-02
1.3500E-02

Figure 2-24. Modeling conditionality in a fault tree, loss of position.

2.2.5.8 Modeling Maintenance. Component failure is not the only type of component event that may be used in a PRA. Some equipment is regularly maintained in a preventive maintenance program, or equipment can fail when it is operating and require corrective maintenance. When a component is out of service for maintenance, it can be logically equivalent to the component being failed in a PRA scenario, and this must be accounted for. Likewise, in some cases, scheduled testing of a component may make a component unavailable for its required purpose from a PRA perspective.

Preventive Maintenance/Testing

Most systems with redundancy on a drilling rig will have at least one unit of the system operating under normal conditions while there may be one or more units in standby in case of a failure. Since preventive maintenance and testing is scheduled in advance, unavailability due to these activities would not apply to an operating unit, and would only apply to the unit(s) in standby. Modeling in this case would result in an added basic event for the standby unit with the unavailability determined by the frequency of maintenance/testing and a distribution for the length of the activity. An example of this is shown in Figure 2-25 modeling two diesel generators, with Diesel Generator 1 operating and Diesel Generator 2 in standby. In the figure, basic event EPS_DGN_PMT_002 has been added to account for the preventive maintenance unavailability.

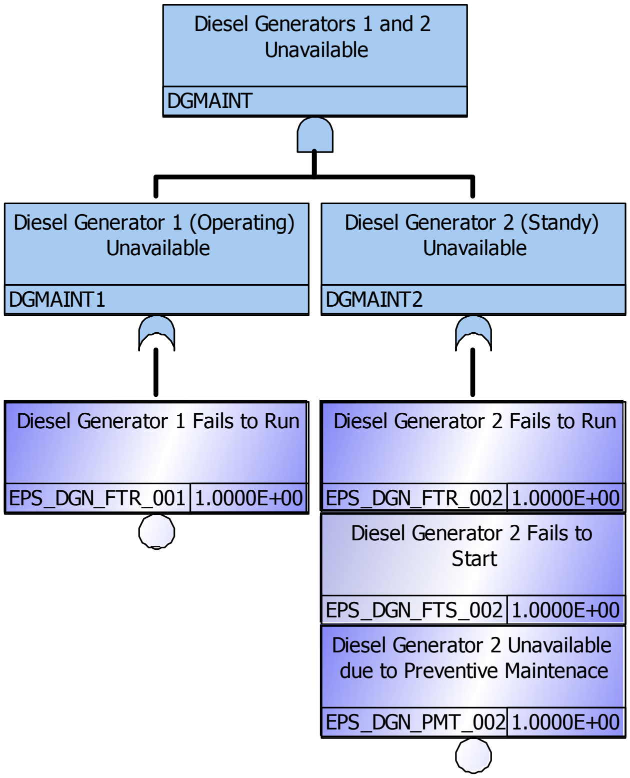

Diesel Generators 1 and 2 Unavailable
DGMAINT
Diesel Generator 1 (Operating) Unavailable
DGMAINT1
Diesel Generator 2 (Standy) Unavailable
DGMAINT2
Diesel Generator 1 Fails to Run
EPS_DGN_FTR_001
1.0000E+00
Diesel Generator 2 Fails to Run
EPS_DGN_FTR_002
1.0000E+00
Diesel Generator 2 Fails to Start
EPS_DGN_FTS_002
1.0000E+00
Diesel Generator 2 Unavailable due to Preventive Maintenace
EPS_DGN_PMT_002
1.0000E+00

Figure 2-25. Preventive maintenance modeling on a diesel generator.

In some cases, there may be a system with redundancy that is all in standby and preventive maintenance may be applicable to all units at any time. This may present a problem in that preventive maintenance would not typically be scheduled on all units at the same time if a system were critical to operations. The fault tree is constructed in a fashion similar to the previous case, as shown in Figure 2-26. One of the cut sets results from solving this fault tree would be EPS_DGN_PMT_001 AND EPS_DGN_PMT_002. On the stated assumptions, this cut set would be spurious if left in the results, if, in reality, preventive maintenance would not be performed on both diesel generators at the same time. In order to deal with this situation, PRA software generally has options allowing the post-processing of results to automatically delete combinations of events that cannot occur.

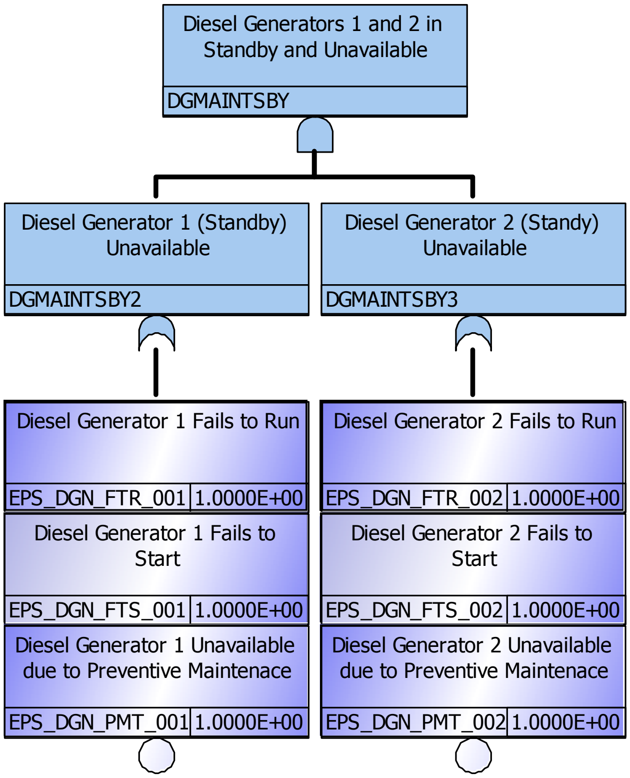

Diesel Generators 1 and 2 in Standby and Unavailable
DGMAINTSBY
Diesel Generator 1 (Standby) Unavailable
DGMAINTSBY2
Diesel Generator 2 (Standy) Unavailable
DGMAINTSBY3
Diesel Generator 1 Fails to Run
EPS_DGN_FTR_001
1.0000E+00
Diesel Generator 1 Fails to Start
EPS_DGN_FTS_001
1.0000E+00
Diesel Generator 1 Unavailable due to Preventive Maintenace
EPS_DGN_PMT_001
1.0000E+00
Diesel Generator 2 Fails to Run
EPS_DGN_FTR_002
1.0000E+00
Diesel Generator 2 Fails to Start
EPS_DGN_FTS_002
1.0000E+00
Diesel Generator 2 Unavailable due to Preventive Maintenace
EPS_DGN_PMT_002
1.0000E+00

Figure 2-26. Preventive maintenance modeling on diesel generators when both are in standby.

Corrective Maintenance

Corrective maintenance is modeled similarly to preventive maintenance, but is treated separately since it is unscheduled and the duration of the maintenance may be different. Figure 2-27 is similar to Figure 2-25 with the corrective maintenance event added.

Appendix C provides more background on unavailability of systems, and shows how it can be estimated using Markov models.

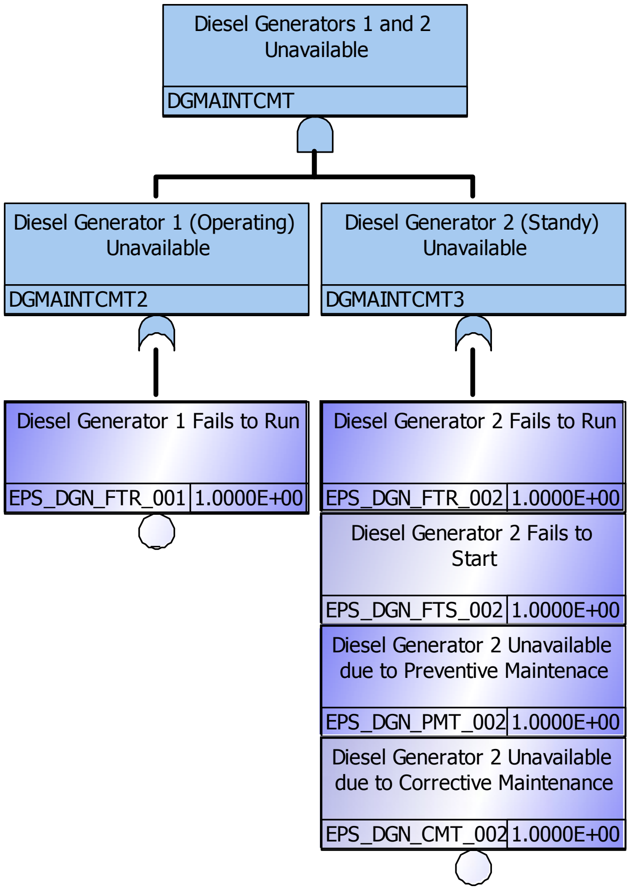

Diesel Generators 1 and 2 Unavailable
DGMAINTCMT
Diesel Generator 1 (Operating) Unavailable
DGMAINTCMT2
Diesel Generator 2 (Standy) Unavailable
DGMAINTCMT3
Diesel Generator 1 Fails to Run
EPS_DGN_FTR_001
1.0000E+00
Diesel Generator 2 Fails to Run
EPS_DGN_FTR_002
1.0000E+00
Diesel Generator 2 Fails to Start
EPS_DGN_FTS_002
1.0000E+00
Diesel Generator 2 Unavailable due to Preventive Maintenace
EPS_DGN_PMT_002
1.0000E+00
Diesel Generator 2 Unavailable due to Corrective Maintenance
EPS_DGN_CMT_002
1.0000E+00

Figure 2-27. Preventive and corrective maintenance modeling on a diesel generator.

2.2.5.9 Modeling Initiating Events. IEs may come from a variety of sources, and may correspondingly require different modeling techniques. Typical categories of IEs that may occur, depending on what is being modeled, are equipment failures, human error, and external events. IEs are developed as frequencies rather than probabilities. For instance, IEs for an exploration well could be expressed as events per well, while for an ongoing production operation, events per year may be more appropriate. Each type of IE is discussed in more detail in the following sections.

Equipment Failure

Equipment failure, in the case of IEs, usually implies that a critical function has been lost, and a sufficiently severe perturbation has occurred that mitigating actions are required. Modeling of equipment failure for an IE is similar to that of mitigating events, but it does have some significant differences. Because the IEs are developed in terms of frequency, the exposure time for failure is usually significantly longer than just responding to an event. For an exploration well, this could typically be on the order of 100 days. Because of the large time window for an IE to occur, equipment may fail and be repaired. Sometime this can occur multiple times as shown in Figure 2-28.

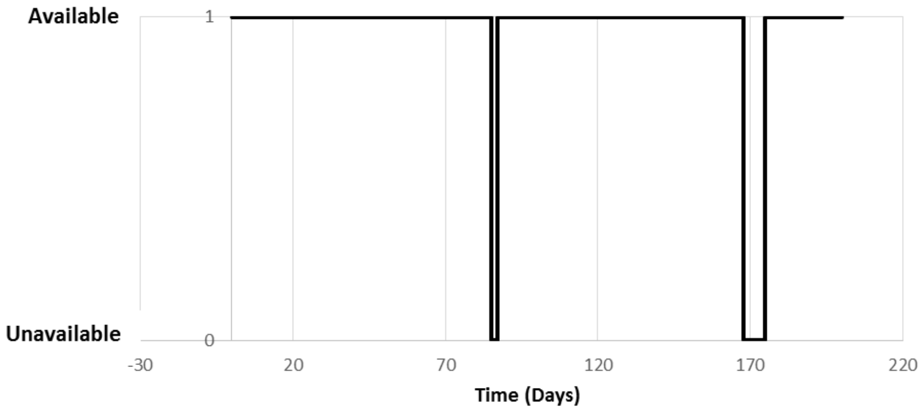


Figure 2-28. Failures with repair.

For systems having redundant components with all of them normally operating, the frequency of loss of that system is the frequency at which enough components fail to violate the success criterion: for example, all (or most, depending on the success criteria) of the redundant components are unavailable at the same time. Because all parts of the system are planned to be operating for the whole mission, any unavailability that occurs is due to random failure (i.e., no planned maintenance will remove the equipment from service). Unavailability, then, is the figure of merit used for redundant-equipment–failure-based IEs. The formula to calculate unavailability of a component is:

$$P = \left(\{\lambda\tau\}/\{1 + \{\lambda\tau\}\}\right) * \left(1 - EXP\left\{-\left(\lambda + \frac{1}{\tau}\right) * Tm\right\}\right) \tag{2-4}$$

where:

P = Unavailability of an operating component during mission time Tm when repair is possible

λ = Mean failure rate of the operating component

τ = Mean repair time for the component.

Both λ and τ have uncertainty distributions associated with them, which must be included in estimating the unavailability of the component. With the unavailability calculated, the modeling can be done in a fault tree as with the mitigating systems. A simple example would be to consider a ship with two thrusters, both normally operating to maintain position, and both required to maintain station-keeping capability. Figure 2-29 shows a simple AND gate with both thrusters and their associated unavailabilities.

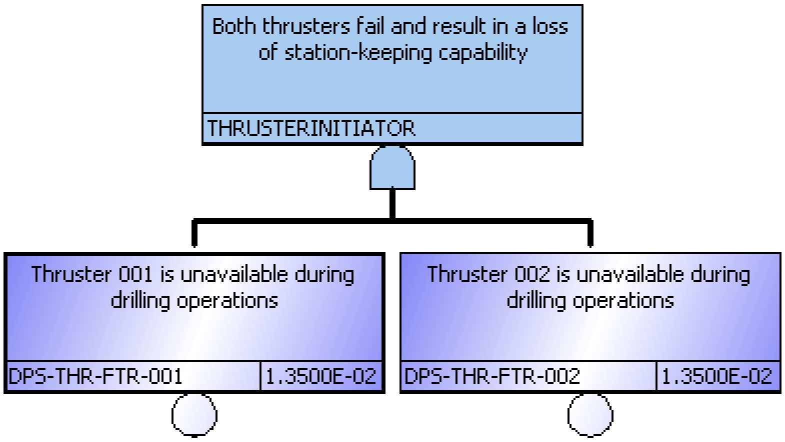

Both thrusters fail and result in a loss of station-keeping capability
THRUSTERINITIATOR
Thruster 001 is unavailable during drilling operations
DPS-THR-FTR-001
1.3500E-02
Thruster 002 is unavailable during drilling operations
DPS-THR-FTR-002
1.3500E-02

Figure 2-29. Loss of station-keeping due to both thrusters being unavailable.

In the case just discussed, it was assumed that a loss of station-keeping capability only occurred if both thrusters were unavailable. Real conditions are often more complicated, and the number of thrusters required may depend on environmental conditions. If conditions were identified where one thruster was adequate to remain on station, then a more accurate model would account for the two conditions where one or both thrusters are actually required, as was shown in Figure 2-24.

For a critical system with redundancy, where one or more components of the system are in a standby condition, a different approach is required. Preventive maintenance may occur on the standby component and must be taken into account along with the random possibilities of corrective maintenance and failure to start and run. The result is logically equivalent to the diesel generator fault tree in Figure 2-27. If the fault tree were developed as an IE, Diesel Generator 1 would have the entire mission time, while the standby diesel generator would have a mission time equal to the average total unavailable time for Diesel Generator 1.

Human Error

Human error can be a source of IEs in routine operations or off nominal conditions, such as severe weather, and is evaluated using human reliability analysis, Evaluating human errors in quantifying IEs is different from evaluating human errors in mitigating events. First, because IEs have the units of frequency, the number of opportunities to have the IE must be accounted for. If a well is being drilled in an area with frequent weather events and the response to weather is to align the ship properly or a loss of position may occur, the frequency of storms requiring the realignment must be accounted for. Other factors such as stress level and amount of time to complete a task factor into human reliability analysis. Human reliability analysis is discussed in more detail in Section 3.

In some cases, it may be possible to estimate the human error frequency for IEs from existing data if the event has occurred before, or possibly from simulator data used for training if it exists.

External Events

Besides equipment failure and human error, external events may also cause IEs. “External events” refers to those events that do not initiate within the systems in the scope of the PRA. For an offshore drilling rig, external events may be man-made or natural. They present a challenge that could result in the consequence being assessed and may involve use of specialized models to assess their unique characteristics. Examples of external events are:

- Severe weather
- Helicopter crashes

- Ship collision
- Mudslides.

Because of the broad range of events, and diversity of data needed for these type of assessments, this guide will only provide a high-level overview of the approach needed to incorporate external events in a PRA.

Many external events may occur with a range of magnitudes (wind speed, wave height, ship collision energy, etc.). The first step is determining whether there is a relationship between frequency and magnitude of an event.

The effect on the facility is reviewed to determine what magnitudes of the external event are important to the PRA. For instance, wind and waves may cause a loss of position, but the wind and wave magnitudes required to cause a loss of position may vary based on the availability of the thrusters. In this case, magnitudes of wind and wave with the potential to cause a loss of position would vary from the case where no thrusters were available (e.g., a blackout condition on the rig), which would require a minimum wind/wave combination, to a case where all thrusters were functioning, which would require a maximum wind/wave combination. There may also be intermediate cases where some loss of thrusters has occurred.

Once the magnitudes of interest are known, an analysis of the frequency of each is performed. Events having very small magnitudes (judged not to be risk drivers) can be screened out of the process as non-contributors. The remaining events and their associated frequencies are treated like other equipment-based events in the PRA model.

Events whose magnitudes are not frequency-dependent, such as helicopter crashes, may still need some analysis of frequency of visits to the rig, and the probability of damaging critical parts of the rig if the PRA consequence involves an effect on rig operations.

2.2.6 Importance of Dependence in Probabilistic Risk Assessment

Significant risk contributors are typically found at the interfaces between components, subsystems, systems, and the surrounding environment. Risk drivers emerge from aspects in which one portion of the design depends on, or interacts with, another portion, or the surrounding environment. Failures arising from dependencies are often difficult to identify and, if neglected in PRA modeling and quantifications, may result in an underestimation of the risk. This section provides an overview of the various types of dependencies typically encountered in a PRA of engineered systems, and discusses how such dependencies can be treated.

2.2.6.1 Definition and Classification of Dependent Events. Two events, A and B, are said to be dependent if $\Pr(A \cap B) \neq \Pr(A)\Pr(B)$. In the presence of dependencies, often, but not always, $\Pr(A \cap B) > \Pr(A)\Pr(B)$. In such a case, if A and B represent failure of a function, the actual probability of failure of both will be higher than the expected probability calculated based on the assumption of independence. In cases where a system provides multiple layers of defense against total system or functional failure, ignoring the effects of dependency can result in underestimation of the probability of failure.

Dependencies can be classified in many different ways. A classification that is useful in relating operational data to reliability characteristics of systems is presented in the following paragraphs [2-2]. In this classification, dependencies are first categorized based on whether they stem from intended functional and physical characteristics of the system, or are due to external factors and unintended characteristics. Therefore, dependence is either intrinsic or extrinsic to the system. The definitions and sub-classifications follow.

Intrinsic. This refers to dependencies where the functional state of one component is affected by the functional state of another. These dependencies normally stem from the way the system is designed to perform its intended function. There are several subclasses of intrinsic dependencies based on the type of influence that components have on each other. These are:

- **Functional Requirement Dependency.** This refers to the case where the functional status of Component A determines the functional requirements of Component B. Possible cases include:
  - B is not needed when A works
  - B is not needed when A fails
  - B is needed when A works
  - B is needed when A fails.

  Functional requirement dependency also includes cases where the load on Component B is increased upon failure of Component A.
- **Functional Input Dependency (or Functional Unavailability).** This is the case where the functional status of Component B depends on the functional status of Component A. An example is the case where Component A must work in order for Component B to work. In other words, Component B is functionally unavailable as long as Component A is not working. An example is the dependence of a motor-driven pump on electric power. Loss of electric power makes the pump functionally unavailable. Once electric power becomes available, the pump will also be operable.
- **Cascade Failure.** This refers to the cases where failure of Component A leads to failure of Component B. For example, an over-current failure of a power supply may cause the failure of components it feeds. In this case, even if the power supply is made operable, the components would still need to be repaired (or perhaps their breakers would need to be reset).

Combinations of the above dependencies identify other types of intrinsic dependencies. An example is the *Shared Equipment Dependency*, when several components are functionally dependent on the same component. For example, if both Component B and Component C are functionally dependent on Component A, then Component B and Component C have a shared equipment dependency.

Extrinsic. This refers to dependencies that are not inherent and intended in the designed functional characteristics of the system. Such dependencies are often physically external to the system. Examples of extrinsic dependencies are:

- **Physical/Environmental.** This category includes dependencies due to environmental factors, including a harsh or abnormal environment. For example, sea states can affect the requirement for the number of thrusters needed to remain on station, or a fire in an engine room may disable multiple diesel generators.
- **Human Interactions.** This is a dependency due to human-machine interaction. An example is an equipment failure due to a maintenance error.

2.2.6.2 Accounting for Dependencies in Probabilistic Risk Assessments. Standard practice is to try to include the intrinsic dependencies in the basic system logic model (e.g., fault trees). For example, functional dependencies arising from the dependence of systems on electric power are included in the logic model by including basic events, which represent component failure modes associated with failures of the electric power supply system. Failures resulting from the failure of another component (cascading failures) are also often modeled explicitly. Sections 2.2.4 and 2.2.5 discuss event tree and fault-tree modeling, and how to incorporate intrinsic dependencies into a PRA model.

Extrinsic dependencies can be treated through modeling of the phenomena and the physical processes involved. Examples are the effects of fire, sea states (wind/wave), etc., in the category of Physical/Environmental dependencies. A key feature of the so-called “external events” is the fact that they can introduce dependencies among PRA basic events. Explicit treatment of the external events such as fire may be a significant portion of a PRA study (see Section 2.3).

2.2.6.3 Modeling Support Systems. Intrinsic dependencies are commonly found in developing a PRA model, in the form of support systems. Support systems are those that provide power, flow, etc., to other systems where critical functions are being performed. They may also support other support systems. From the simple example in Figure 2-17, the lower-right-hand basic event is related to electric power that energizes the pump. The electric power system is a separate system and may provide support for many systems. Since a fault tree is typically detailed down to the component level where data exists, the fault tree in Figure 2-17 would normally have the electric power feeding the pump detailed down to the boundary of the analysis, which on an offshore rig would be the diesel generator(s) (and maybe the fuel and air intake systems) as shown in the simplified fault tree in Figure 2-30.

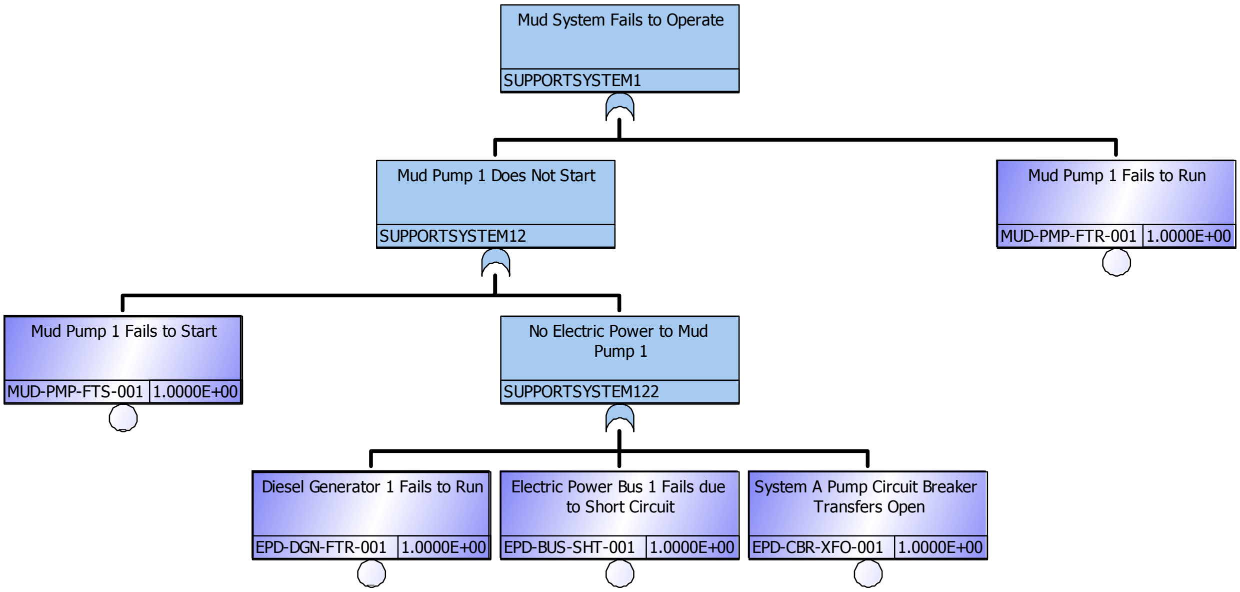

Mud System Fails to Operate
SUPPORTSYSTEM1
Mud Pump 1 Does Not Start
SUPPORTSYSTEM12
Mud Pump 1 Fails to Run
MUD-PMP-FTR-001
1.0000E+00
Mud Pump 1 Fails to Start
MUD-PMP-FTS-001
1.0000E+00
No Electric Power to Mud Pump 1
SUPPORTSYSTEM122
Diesel Generator 1 Fails to Run
EPD-DGN-FTR-001
1.0000E+00
Electric Power Bus 1 Fails due to Short Circuit
EPD-BUS-SHT-001
1.0000E+00
System A Pump Circuit Breaker Transfers Open
EPD-CBR-XFO-001
1.0000E+00

Figure 2-30. Simple fault tree with support system modeled.

From Figure 2-30, the electric power support is shown to be composed of the diesel generator, power bus, and the circuit breaker that feeds Mud Pump 1. The diesel generator and electric power bus would likely feed multiple items, and should be separated so that they can be modeled only once, and that model used where necessary. Figure 2-31 shows the proper way to model this situation. The common parts of the electric power system have been separated under a separate OR gate and generically labeled as power from Diesel Generator 1. If these components are needed in another fault tree, the OR gate can be made a transfer event as shown in Figure 2-32, and that transfer event can be used wherever needed.

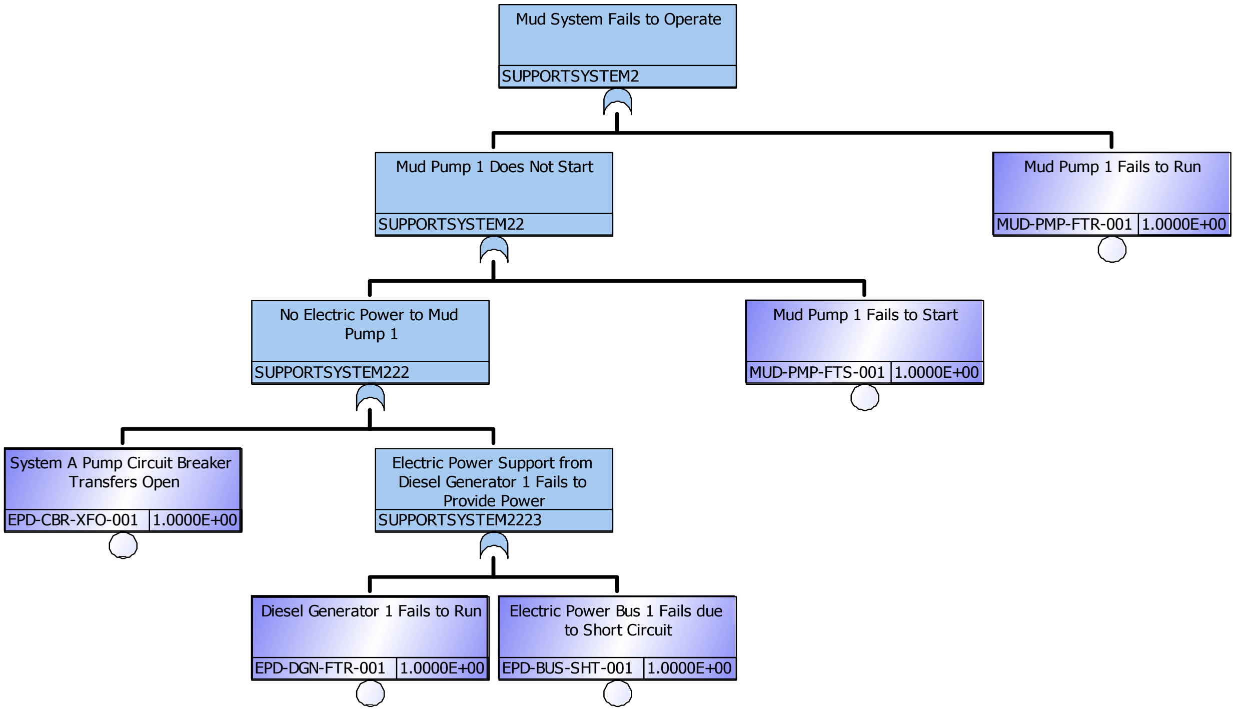

Mud System Fails to Operate
SUPPORTSYSTEM2
Mud Pump 1 Does Not Start
SUPPORTSYSTEM22
Mud Pump 1 Fails to Run
MUD-PMP-FTR-001
1.0000E+00
No Electric Power to Mud Pump 1
SUPPORTSYSTEM222
Mud Pump 1 Fails to Start
MUD-PMP-FTS-001
1.0000E+00
System A Pump Circuit Breaker Transfers Open
EPD-CBR-XFO-001
1.0000E+00
Electric Power Support from Diesel Generator 1 Fails to Provide Power
SUPPORTSYSTEM2223
Diesel Generator 1 Fails to Run
EPD-DGN-FTR-001
1.0000E+00
Electric Power Bus 1 Fails due to Short Circuit
EPD-BUS-SHT-001
1.0000E+00

Figure 2-31. Support system (Diesel Generator 1 and the power bus) modeled so it can be used in multiple fault trees.

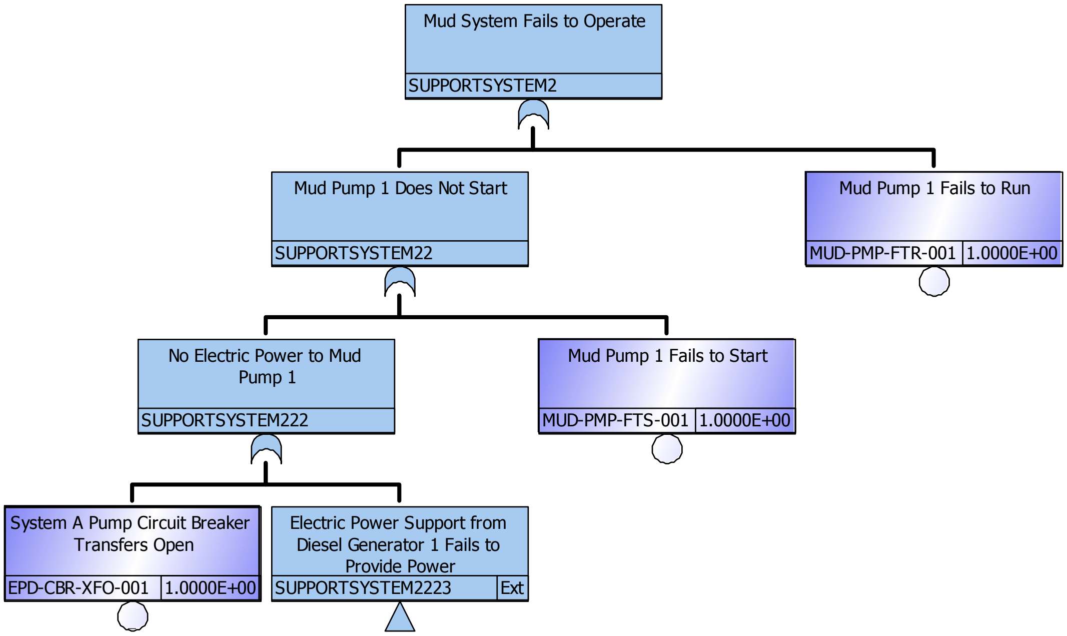

Mud System Fails to Operate
SUPPORTSYSTEM2
Mud Pump 1 Does Not Start
SUPPORTSYSTEM22
Mud Pump 1 Fails to Run
MUD-PMP-FTR-001
1.0000E+00
No Electric Power to Mud Pump 1
SUPPORTSYSTEM222
Mud Pump 1 Fails to Start
MUD-PMP-FTS-001
1.0000E+00
System A Pump Circuit Breaker Transfers Open
EPD-CBR-XFO-001
1.0000E+00
Electric Power Support from Diesel Generator 1 Fails to Provide Power
SUPPORTSYSTEM2223
Ext

Figure 2-32. Power from Diesel Generator 1 modeled as a transfer that will be used in multiple fault trees.

2.2.6.4 Dependency Matrices. Because system dependencies may get complex, PRA analysts may map the system relationships prior to starting a fault tree, in order to ensure that all dependencies are properly accounted for, by developing a system dependency matrix. Figure 2-33 shows a simple block diagram of typical systems on a drill ship. The front-line systems are those that are used to accomplish a primary task such as drilling (mud, drawworks, etc.), and the "support" systems (electric power, cooling, etc.) are those that support the front-line systems, but do not directly perform a front-line function.

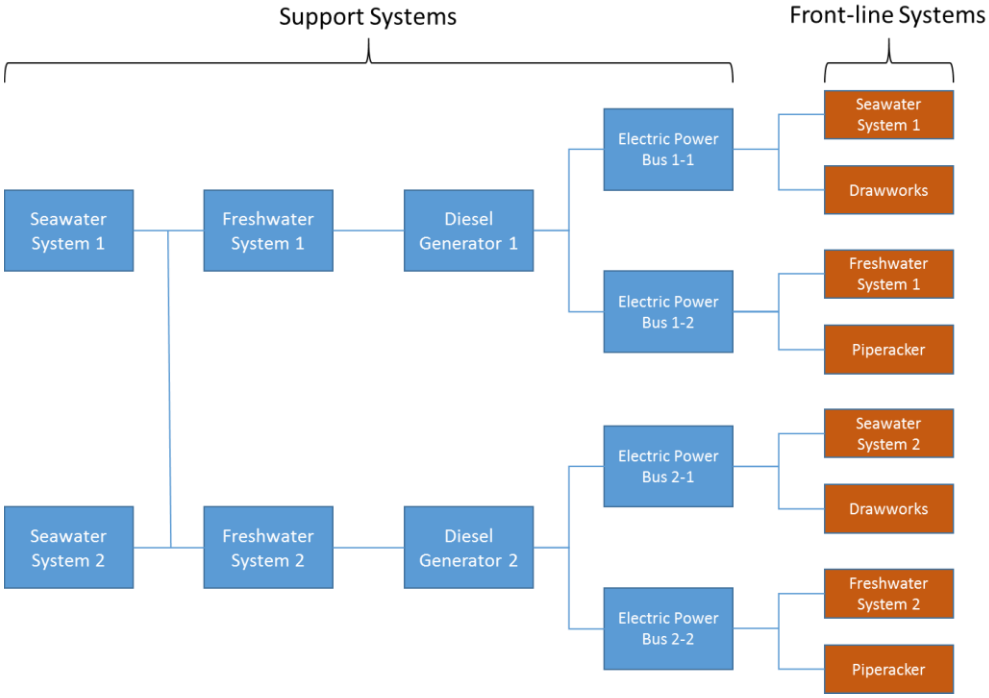


Figure 2-33. Simplified system block diagram.

Knowing the relationships in Figure 2-33, a dependency matrix may be constructed as shown in Figure 2-34. Using the dependency matrix assists the PRA analyst in ensuring the correct support system dependencies are accounted for. The added notes should detail any special situations such as crossties.

Occasionally, "loops" occur in models of support systems. For instance, from Figure 2-34, the seawater system supports the fresh water system, which in turn supports the diesel generator. However, the diesel generator powers the sea water system and the fresh water system; explicitly putting all of this mutual dependency into a logic model creates a "loop," which a logic code considers to be a modeling error (code generally will not produce cut sets from a model containing a loop).

In general, from a modeling perspective, support systems should be modeled in the fault tree the way they support the front-line system being modeled. For a fault tree of the drawworks, based on this example, the first support systems included would be supporting electrical buses, then the diesel generators. The fresh water systems would be a support to the diesel generators, and the sea water systems would support the fresh water systems. In this case, the seawater systems are supported by the same electrical busses as the diesel generators and are already included in the model, so there is no need to create a loop in the fault tree for electrical busses that support the seawater system. The one missing element would be the electric power bus 1-2 and 2-2, which power the fresh water systems but not the drawworks. The specific electric busses 1-2 and 2-2 must be included separately with the fresh water system for this example.

	Sea Water System 1	Sea Water System 2	Fresh Water System 1	Fresh Water System 2	Diesel Generator System 1	Diesel Generator System 2	Electric Power Bus 1-1	Electric Power Bus 1-2	Electric Power Bus 2-1	Electric Power Bus 2-2	Drawworks	Pipe Racker
Sea Water System 1	■		A	A								
Sea Water System 2		■	A	A								
Fresh Water System 1			■		X							
Fresh Water System 2				■		X						
Diesel Generator System 1					■		X	X				
Diesel Generator System 2						■			X	X		
Electric Power Bus 1-1	X						■				B	
Electric Power Bus 1-2			X					■				C
Electric Power Bus 2-1		X							■		B	
Electric Power Bus 2-2				X						■		C

Notes:

A – Seawater Systems 1 and 2 are normally separated but can be crosstied so either System 1 or 2 can be used to cool one or both freshwater systems.

B – The Drawworks can be used from either Electric Power Bus 1-1 or 2-1.

C – The Pipe Racker can be used from either Electric Power Bus 1-2 or 2-2.

Figure 2-34. Example dependency matrix.

2.2.7 Linking Fault Trees and Event Trees

Once the event trees and their associated fault trees have been developed and linked, the qualitative part of the PRA model is completed. Fault trees and event trees are said to be "linked" when the fault trees for pivotal events and the event trees containing those pivotal events are tied together properly in the software being used to evaluate the accident sequence cut sets. The scenarios are formed from the basic events and fault-tree logic combined with the event-tree sequences and end states. The model can now be evaluated qualitatively to review individual scenarios. Table 2-4 shows a sample of the output from a model in terms of cut sets.

Each cut set has the well kick IE, INIT-EV_DRILLING, followed by other basic events whose combined occurrence leads to the end state. Typically, all the events shown are failure basic events; however, success events are sometimes shown and indicated with "/" or other identifiers, depending on the PRA software being used. For computational reasons, success events are often approximated as having unit probability, because in many problems, the values of failure are small enough that "success" actually has essentially unit probability; this approximation can save a lot of computation. When failure probabilities become large, success terms must be computed more accurately in order to provide accurate results. Figure 2-35 shows the path from the second and third cut sets, sequence 15-2, from the table below.

Table 2-4. Sample cut sets from linked fault-tree/event-tree model.

# Total	**Cut Set** Displaying 5 Cut Sets (10551 Original)	**Description**
1	DRILLING: 14	
	INIT-EV_DRILLING	Well Kick While Drilling
	BOP-HUM-ERR-KICKDET	Driller fails to realize a kick has occurred or does not take timely action.
	LIMITEDRELEASE	
2	DRILLING: 15-2	
	INIT-EV_DRILLING	Well Kick While Drilling
	BOP-CYL-JAM-BSRDP	BSR fail to close and seal after emergency disconnect.
	BOP-HUM-ERR-KICKDET	Driller fails to realize a kick has occurred or does not take timely action.
	CAPPINGSTACKCONTAIN	
3	DRILLING: 15-2	
	INIT-EV_DRILLING	Well Kick While Drilling
	BOP-HUM-ERR-HANGOFF	Driller fails to position drill pipe properly before activating BSR.
	BOP-HUM-ERR-KICKDET	Driller fails to realize a kick has occurred or does not take timely action.
	DP_TOOLJOINT_PRESENT	Drill pipe tool joint is present.
	CAPPINGSTACKCONTAIN	
4	DRILLING: 15-3	
	INIT-EV_DRILLING	Well Kick While Drilling
	BOP-CYL-JAM-BSRDP	BSR fail to close and seal after emergency disconnect.
	BOP-HUM-ERR-KICKDET	Driller fails to realize a kick has occurred or does not take timely action.
	CAP_STACK_FAILS	Capping stack is unsuccessful.
	RELIEFWELLSEAL	
5	DRILLING: 15-1	
	INIT-EV_DRILLING	Well Kick While Drilling
	BOP-HUM-ERR-KICKDET	Driller fails to realize a kick has occurred or does not take timely action.
	BOP-PRG-FLO-I02	Subsea manifold pressure regulator I02 fails low (Yellow).
	ROVCONTAIN	

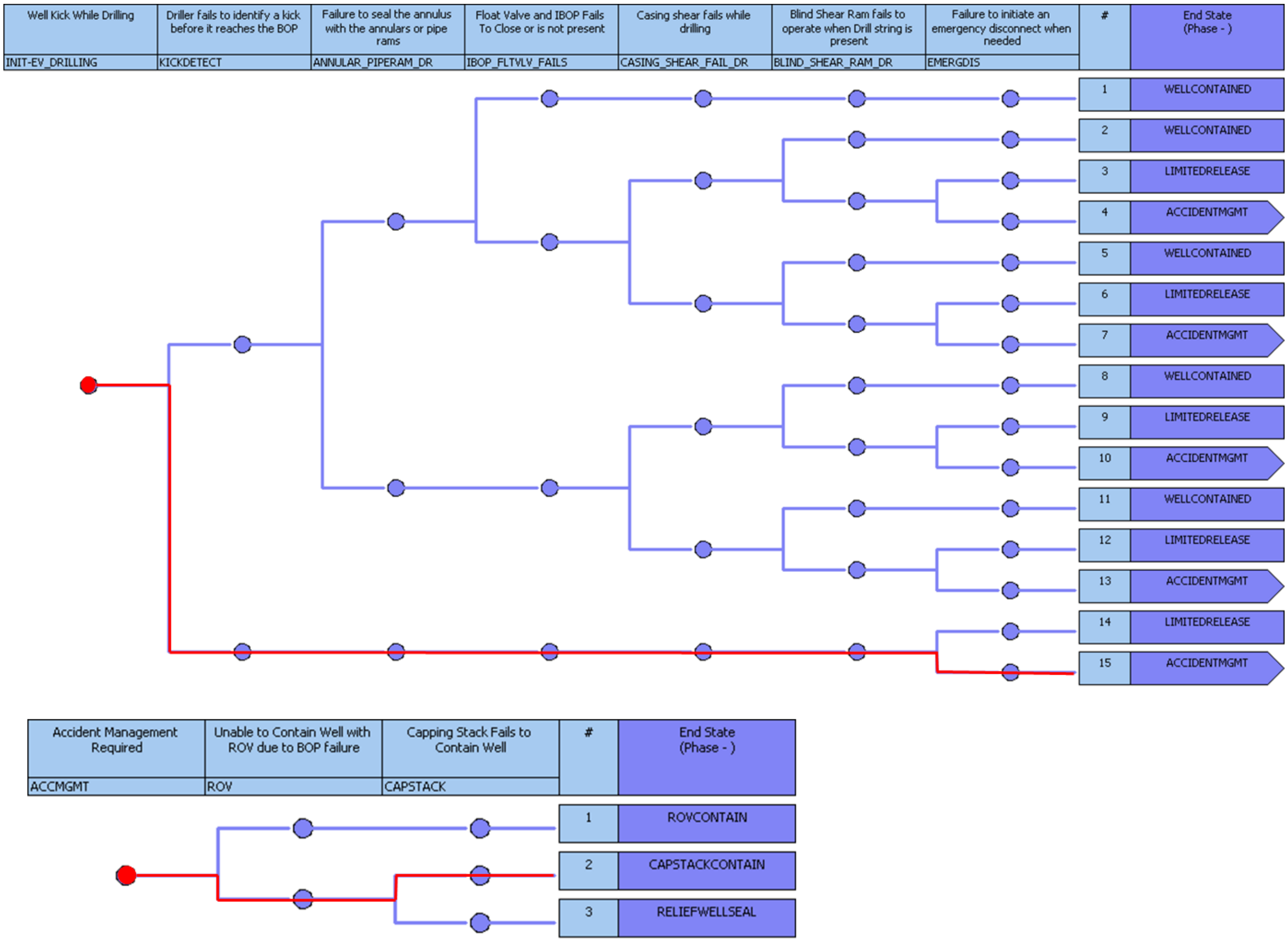


Figure 2-35. Sequence 15-2 for Cut Sets 2 and 3.

The Boolean reduction yields minimal cut sets: combinations of basic events that are each *sufficient* to cause the top event, and *necessary* in the sense that if one event is eliminated from a cut set, the remaining events are no longer sufficient to cause the top event. For instance, if the annular preventer fails due to the CCF of the yellow and blue pods, that common-cause failure will not appear in a minimal cut set together with a failure of the BSR to close or one of the shuttle valve failures. This is because the CCF of the pods has already guaranteed the failure of the BSR, and the failure of the ram or shuttle valve is inconsequential if the CCF has occurred.

The final step in building the PRA model is populating the fault trees with data. Section 3 discusses the various methods needed to establish the most credible data for quantification.

2.3 Simulation

2.3.1 Background

In general, risk modeling is done in order to help a decision-maker appreciate what the future might hold, conditional on the choices available in the decision situation: perhaps a design decision whether to include a proposed safety system, or perhaps an operational decision whether the current facility configuration is safe enough. As stressed in earlier sections, the choice of techniques to use to model the risk depends on the level of rigor needed for the decision at hand, and on the particulars of the situation. But the general idea of rigor does not explicitly reflect all of the considerations that are involved. The technical *ingredients* of the analysis need to be considered. This section discusses modeling choices from the point of view of how hard we need to work, and what considerations we need to include, in order to adequately capture both the phenomenology aspects and the reliability aspects of a given situation.

Section 2.2 has extensively discussed logic modeling. But even in situations where logic models are sufficient for the purposes at hand, they need to be informed by mission success criteria, which are obtained from phenomenological modeling. In addition, at the back end of a logic-based PRA model, consequences of the scenarios may need to be estimated and reported on some form of continuous or discrete scale with a large number of states. Thus, non-binary physical and phenomenological models, such as "casualty expectation models," are also applicable and commonly found in this portion of a PRA.

Moreover, as discussed below, logic models are not always sufficient.

Phenomenological modeling involves physics-based analysis methods used to study and characterize complex, interactive systems where the progression of events is governed by physical processes. Phenomenological modeling techniques are used to complement and extend traditional reliability modeling methods by evaluating events that are driven by physical processes. Fault trees and event trees found in PRAs include IEs that are statistically generated. However, models of the pivotal events that represent the subsequent events following the initiators may need to address phenomenology. This is especially true when the sequence of events and processes depends upon the physical interactions of the system with its current surroundings.

Engineering models based on the fundamental laws of motion, heat transfer, chemical reactions, gas dynamics, structural analysis, and other phenomena can be used to represent the conditions and state of the environment surrounding the system. These physical processes evolve as a function of time and system state, and are defined using mathematical equations that describe fluid flow, wave propagation, structural fatigue, combustion, and so forth. A failure is defined to occur when the system observes physical conditions that violate a subsystem's specific physical limit or capacity.

In phenomenological risk models, the interactions of a complex system are coupled through common physical parameters, and the subsequent responses and consequences are dynamically determined based on the current conditions of the system, its environment, and its design limits. Failure probabilities are then developed by calculating the range of current state conditions, and determining whether they violate a specific design limit or system threshold.

Failure probabilities developed by this approach directly and explicitly connect the existing design definition to the physical failure mechanism, providing design teams with actionable engineering information.

In some cases, a specific type of scenario that lends itself to modeling via physical and phenomenological models may be addressed as a special subject within a PRA, or often is treated as a complementary study.

Summary

All models include both phenomenology and reliability, at least implicitly. Logic models to be used for reliability analysis are structured based on the results of phenomenological analysis (or, if analysis is lacking, expert judgment about the phenomenology); and when scenario phenomenology is simulated, the elements of the scenario that is to be analyzed (including facility configuration, initial conditions, whether components fail, and so on) are specified by the analysts based on some notion of what is either probabilistically significant or, perhaps, what we can take to be a bounding case, depending on the framing of the decision being supported. While both phenomenology and reliability are present in principle, one or the other is typically implicit, and their relative emphasis depends on the application. Therefore, the approach to analysis depends not only on what level of rigor is needed, but also on how the results are influenced by the ingredients of the situation being modeled.

Accordingly, the following sections describe certain strengths and weaknesses of:

- Classical logic modeling
- Discrete-event / event-driven simulation
- Simulation of phenomenology-driven time histories.

Then we discuss current approaches to efficiently combining phenomenology and reliability, when we need to model both aspects with some care. Finally, examples are given of a phenomenology-driven analysis and of a discrete-event analysis.

2.3.1.1 Classical Logic Modeling. In general, logic models work to articulate the various sets of conditions under which some undesired event (some adverse accident consequence) will occur. “Articulate” is used because the logic cannot, by itself, determine when a system or functional failure will occur. Rather, given a body of simulation results for scenario phenomenology,[e] the logic model is structured by the modeler to yield a listing of ALL sets of conditions (system configurations) that correspond to functional failure, and ONLY system configurations that correspond to failure.

As explained in earlier sections, this is done by using logic to keep track of a potentially large set of potentially complicated conditions, usually corresponding to combinations of component states. If a system comprises redundant functional trains, and engineering analysis shows that success of the system is assured if either train functions, then failure must be modeled as a logical “AND” of the failure of each train, failure of each train is modeled as failure of its segments, and failure of each segment as failure of its components, and failure of each component is modeled as failure of the component itself or failure of its various support systems (power, actuation signal, etc.). Incidentally, this level of detail sounds complex, but may not be, even if many components are involved, if the topology of the system is simple. On the other hand, if the topology of the system is complex (e.g., crossties between divisions or between divisions of support systems), the model may be complex in some sense, even if the number of components involved is not large.

The structure of this logic is determined by mission success criteria: statements (based on engineering analysis, e.g., simulation) of what possible system configurations correspond to success. For a fluid system, success may be defined in terms of providing a particular flow at a particular pressure, which implies success of a certain number of pump trains. The above example of a system succeeding if either of its trains functions is an example of a mission success criterion.

Typical Approximations in Classical Logic Modeling (Event-Tree / Fault-Tree Modeling)

Mission Success Criteria

[e] …. or a body of assumptions based on expert judgment of what simulation would say if it were performed …

Ideally, event trees are built with pivotal event headings that collectively serve to accurately determine success or failure of each event sequence, based on the mission success criteria. But in practice, the degree to which this ideal is realized varies considerably. In many models, some event-tree sequences labeled “success” will really be “failure” some fraction of the time, and *vice versa*, owing to such things as variations in initial conditions, event timing (discussed below), and heroic actions that could not be anticipated (e.g., Apollo 13). One could imagine trying to deal with these possibilities by adding pivotal event headings to the event trees in order to introduce the additional distinctions between sets of conditions leading to success or failure. One could distinguish different sets of initial conditions, different timing possibilities, and so on, and develop corresponding multiple versions of fault trees to model pivotal events conditional on different upstream conditions; but doing the work needed to specify detailed success criteria for all these possibilities begins to approach doing the work needed to carry out a fully simulation-based approach.

Discrete Versus Continuous Variables

In theory, logic modeling can address system state variables that are continuous in nature, such as degraded heat transfer, partial blockages of flow paths, and so on. Propositional calculus, as it used to be called, was meant to apply to *propositions*, which can be true-or-false statements about ranges of physical variables or time intervals. However, since the advent of fault-tree analysis, the tendency has been to map component failures onto Boolean variables. Modeling continuous-valued performance variables in logic models is not common practice, partly because for many systems, deviations in those variables are relatively insignificant contributors to assessed risk. For many systems, failure is dominated by failures of active components to change state when they are required to do so, which is natural to capture in Boolean variables having values of either “true” (the pump started up) or “false” (the pump did not start up).

However, in some systems, it may be the case that something like a combination of partial blockage and degraded heat transfer would cause failure of the system. In such a case, trying to express the problem in terms of Boolean variables can be tried, but would result in writing out many combinations of conditions and still being unable to quantify them. Failure could be due to a little blockage together with a significant degradation in heat transfer, or a little more blockage and a slightly less significant degradation in heat transfer, etc. In fact, for a problem like this, we need to explore issue space with a series of simulations, in order to map out a “limit surface” delineating the settings of these variables corresponding to failure. [2-3] This will be discussed further below.

Timing Issues

Another approximation is introduced when we model “success” or “failure” in terms of whether something happens before some designated fixed time. For example, failure may occur in a particular model if a system fails to run continuously for 24 hours, or if emergency power is not restored in 4 hours; but the reality may be that in some situations, failure will occur sooner, and success may occur later, than such a specific cutoff. Treating this sort of situation in logic models is particularly thorny when two timing-related events appear “ANDed” together. Certain types of simulation are much better suited to modeling this kind of situation.

Presumed Independence of Basic Events

Boolean algebra works whether or not the propositions being manipulated are independent of each other. If failure of Pump A and Pump B logically causes system failure, then the truth of this does not depend on whether these two failures are correlated. The *probability* of this situation depends on the correlation, but the logic expression does not.

Although some capability exists in some software implementations (GO) [2-4] to allow for dependency between basic events for purposes of quantification, most fault-tree codes essentially assume independence of basic events in the initial transformation of the logic input into disjunctive normal form (the cut sets). Users of such codes are tacitly expected to model correlations as separate basic events (e.g., common-cause basic events), or perhaps post-process the cut set expression in various ways in order to render a more accurate representation of the scenario set.

For example, consider modeling of recovery actions. In many models of facility safety, it is appropriate to consider the possibility that recovery can occur to prevent the top event, even after a combination of basic events has occurred that will cause the top event if recovery is not successful. *Not* modeling recovery may lead to an unrealistic assessment of top event probability; but in general, it is impractical to put recovery into the model *a priori*, because the probability of its success depends in detail on the specifics of each cut set. However, post-processing the cut sets can model recovery, because each cut set can be analyzed individually, and recovery of each cut set can be quantified conditional on the other basic events in the cut set. Putting a one-size-fits-all recovery event into the logic model *a priori* is likely to be wrong. [2-5]

Another example of correlation occurs if certain events cannot occur together in a scenario. A post-processor can delete terms containing such impossible conjunctions. In analysis of regulated nuclear facilities, a popular application of this idea is to eliminate conjunctions of maintenance actions that, while not physically impossible, are not supposed to occur. Such combinations can be optimistically presumed to have vanishingly low probabilities, so these terms are deleted [2-5].

Actually, this “delete term” capability was formulated in order to avoid the need for “NOT” logic in modeling of accident sequences where it is necessary to model “Failure of System A and NOT Failure of System B.” One way to evaluate such a conjunction is to do exactly what the event definition says: form an expression for Failure of System A and an expression for NOT Failure of System B, and then compute the AND of these two expressions. An alternative way to get a generally good result (approximate but highly satisfactory in many situations) is to evaluate a logic expression for Failure of System A, and then delete from that expression any term that implies Failure of System B. This avoids the need to evaluate the “NOT” of System B failure, and then grind through evaluating the conjunction of that with Failure of System A.

In short, a post-processing algorithm can examine each cut set, and replace particular conjunctions of basic events with different conjunctions of events to account for correlation or common cause, or add basic events corresponding to scenario-specific recovery actions.

Strengths of Classical Logic Modeling

Classical logic modeling is useful for:

- Modeling topologically complex systems. For example, logic modeling is useful for capturing failure modes in systems that depend on support systems in a complicated way. Logic modeling has been known to point out system-level failure modes that are sufficiently obscure to be difficult to comprehend, even in retrospect.
- Finding obscure combinations of component-level failure modes that would cause system failure, but might elude simulation-based approaches. In a complex facility that has, in principle, trillions of combinations of component-level failure modes, logic modeling may struggle to generate all of them, but can generate many combinations that are worth knowing. In particular, logic modeling can identify combinations having nominal probabilities that are sufficiently low that they will not typically show up in a simulation-based approach. For example, if we are simulating histories initiated by a particular event, and some possible histories have conditional probabilities on the order of 1E-4, we will need to simulate multiple tens of thousands of time histories in order to have a decent chance of seeing them; but a logic model should readily identify them.

Classical logic modeling is approximate, of course, but for some purposes, it is adequate. The first plant-scale logic model, WASH-1400 [2-6], needed to be detailed enough to distinguish the risk significance of station blackout events from the risk significance of large loss-of-coolant accidents. These scenarios differ significantly in frequency, and for the purpose of comparing their risk significance, the state of logic modeling practice at that time (mid-1970s) was adequate. For purposes of evaluating operating procedures, it was not adequate. A more recent analysis of the functional failure of "feed and bleed" showed that classical mission success criteria are deficient, and it is conceivable that corresponding plant procedures are deficient as a result.

Weaknesses of Classical Logic Modeling:

- Time dependence is difficult to model accurately. There are many ways in which this arises [2-5].
- Certain system degradations are difficult to map onto Boolean (discrete logic) variables. For certain types of systems, this is a large difficulty [2-3].
- Mission success criteria that are simple are likely to be approximations, and not necessarily good approximations [2-7].

2.3.1.2 Discrete-Event/ Event-Driven Simulation. Suppose that we wish to estimate the reliability of a system whose complexity makes it impractical to compute reliability either directly (for example, by evaluating a closed-form expression for reliability in terms of component reliability models) or using logic models. One common way of estimating complex system reliability is to simulate a large number of time histories, sampling over different instantiations of component failure time; if the sampling is done appropriately, then the reliability can be calculated in terms of the number of time histories in which the system succeeded, and the total number of time histories simulated.

One way of simulating a time history is the following:

1. Initialize the component states and the system clock.
2. Increment the system clock by a small time step *dt.*
3. If the current time exceeds the mission time, report the outcome (e.g., "system success"), and go to Step (1) to begin the next time history.
4. For each component, sample a random number and compare it with λ **dt* to determine whether that component fails in the current time step (λ being that component's failure rate). Propagate the effects of all component state changes through the system configuration.
5. If the system is now failed, return that result, and go to Step (1) to begin a new time history; if not, return to Step (2) to continue the present time history.

Iteration continues until the entire mission has been analyzed, with appropriate tracking of reliability, availability, and performance-related metrics throughout the current time history.

A number of time histories are simulated in this fashion, sufficient to provide adequate statistics for quantification of the metrics being analyzed.

Note that in Step 4, it is possible in principle to condition each component's λ on the time history of the entire system up to that point, including all of the states of the other components. For example, if we are modeling electronics that depend on room cooling, and room cooling has been lost in an earlier time step, the λ of the electronics can be conditioned on abnormally high temperature. However, this approach tends to force the time step to be small, which makes large problems (involving many components and long times) less feasible.

Another way of developing time histories is the following. Instead of determining each component's failure time by marching along the time axis one *dt* at a time, and waiting for a random number generator to decide which time step will yield a transition, sample once per component to determine its failure time directly from its failure time distribution. As illustrated in Figure 2-36, the random-number sampling process for this component furnished the random number 0.7, which yields a failure time of about 51. Carrying out a similar process for all of the aleatory degrees of freedom, and knowing all of the scheduled events (like planned tests) *a priori*, makes it possible to immediately determine the individual variable (component) state transition times in this time history and the system-level state transition times. The reliability/ availability/ performance (RAP) metrics can then be assessed immediately in each time history.

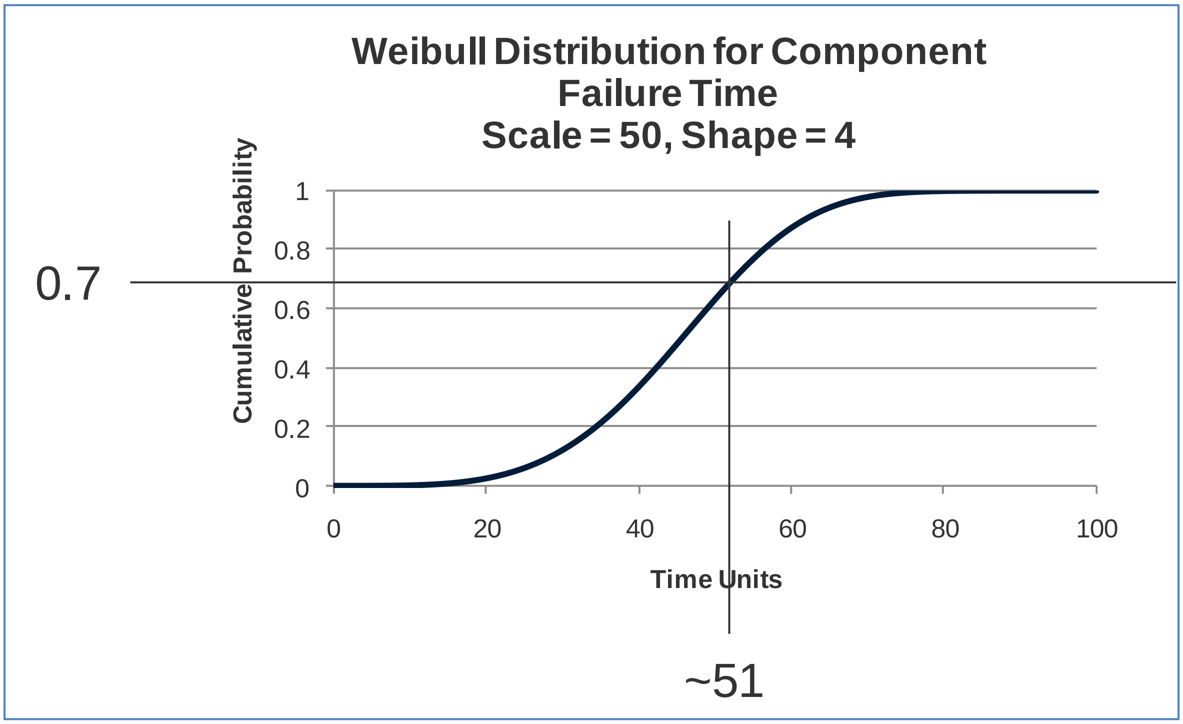


Figure 2-36. Sampling from the cumulative failure time distribution to determine a time-history-specific component failure time.

This latter approach (most commonly called “event-driven” simulation) has been used for generations [2-8] to assess RAP metrics for very large-scale, very complex systems. It is much faster than evaluating RAP metrics over a long mission time by generating stochastic events in each small time interval *dt*. However, in this type of event-driven simulation, dependencies among components or between component behavior and current physical state are either not reflected or are modeled rather selectively. This is because each component’s failure time distribution is written down *a priori*, which does not reflect time-history-specific developments that might influence the component’s behavior in that time history. “Exhaustive” simulation, which generates state transitions (or not) in each time step *dt*, “knows” everything about past history and current state, and can, in principle, model a broad range of influences on the stochastic or deterministic state transitions based on this knowledge. However, this modeling flexibility comes at a high price in terms of execution time.

Strengths of Event-driven Simulation

Simulation of time histories based on sampling component failure times *a priori* can quantify reliability/ availability metrics efficiently, given:

- Failure time distributions
- Schedules for all scheduled events (planned events, such as scheduled maintenance)
- Simple rules relating component states to train/ system/ function success.

Simulation of time histories based on incrementing one *dt* at a time, and modeling failures in that *dt* conditional on current conditions and on the entire scenario-specific history, can reflect all influences on component failure times.

Weaknesses of Event-driven Simulation

Extant approaches for doing this generally treat the component failure times as completely stochastic given their failure time distributions. Scenario-specific influences on component failure time distributions are not reflected.

Exhaustive modeling is prohibitive for large systems with complex influences on component failure times. For smaller *dt*, a larger number of time steps need to be analyzed individually in order to simulate a long time, with a potentially high cost at each time step.

2.3.1.3 Simulation of Phenomenology-driven Time Histories.

In this document, by "phenomenology-driven," we mean scenarios that are specified in terms of IEs, with subsequent evolutions determined entirely by physical laws, and not by discontinuous changes of component states.

For example, consider rupture of a pressurized tank containing toxic or flammable fluid. The fluid is released under pressure and either disperses, with one set of adverse consequences, or ignites, culminating in a different set of adverse consequences; or perhaps a vapor cloud is transported for some distance and then ignites, with yet other consequences. The evolution of these scenarios is described in terms of sets of differential equations or partial differential equations that implement a particular physical model, and initial and boundary conditions; apart from initiation of the scenario, the modeling may not include anything stochastic, other than perhaps a source of ignition in some of the scenarios.

If fire or explosion can cause damage to nearby personnel, equipment, or structures, then it is necessary to characterize that damage, including a range of possible outcomes. For some purposes, it may be adequate to simplify the analysis by assuming that personnel within a certain radius of the site of the fire or explosion will be injured or killed, and systems or structures within another radius of the fire or explosion will be put into a failed state. This would be typical of a bounding analysis, which is satisfactory for some purposes. For other purposes, it may be important to understand which initial conditions realistically lead to adverse outcomes. For this latter purpose, it may be necessary to analyze with a great deal more rigor: running the simulations with finer mesh and smaller time steps, and running many more time histories with different initial conditions, in order to understand where, in the relevant issue space, the risk lies. An analysis like this can easily overwhelm the computational resources available.

The analysis becomes even more difficult if, in addition to analyzing the scenario consequences, the time evolution of the scenario is affected by phenomenology-induced damage to components along the way, especially if there is a potential for failure of mitigating systems, structures, or components, due to harsh environments created by the accident conditions. Real analysis of those effects may call for a multi-physics kind of analysis, either treating multi-physics within a single solver, or somehow tying together analysis tools that each specialize in one of the problem's physics domains.

Phenomenological models may be used to characterize unintended physical interactions among systems and/or between a system and its environment. Developing a strong understanding of system-environment interactions is extremely important because the environment is not a directly controllable system and may deviate unexpectedly. Therefore, system capabilities must be explored within a range of conditions, not just nominal conditions, in order to better understand potential vulnerabilities and the levels of safety margins required. Because phenomenological modeling is based on specific conditions and physical interactions, it can be used to characterize off-nominal system behavior and unsteady physical processes in addition to nominal, steady behavior.

Since the environment and system interactions studied in phenomenological methods are highly dynamic, traditional reliability methods based on statistical means and steady-state behaviors are not adequate to model these situations. Instead, physical modeling is needed to understand how potentially small off-nominal events or deviations can cascade into more serious failures; it is necessary to dynamically track the key conditions of the system over time. The physical models that are used to define the processes, whether they are steady or unsteady behaviors, are mathematical engineering models based on deterministic equations.

Because physics-based models depend on the laws of nature, they are not heavily dependent on expert opinion or failure data developed through traditional reliability techniques or data sources, such as military handbooks, to determine expected outcomes. The mean time to failure cannot be evaluated using handbooks because the system state is constantly evolving and the outcome is dependent on these evolving conditions. Physics-based models also do not specifically address such things as human factors or human error since they are not governed by laws of nature.

For scenarios having the characteristics mentioned above, a state-of-practice simulation of accident phenomenology is needed, along with an appropriate amount of verification and validation of the simulation.

Potential Weaknesses of Phenomenology-Driven Analysis

It may be impractical to scope the analysis to include everything important, because the solver may not be equipped to deal efficiently with all of the multi-physics and multi-chemistry, or with component state changes, which inject physical discontinuities into the analysis.

Some simulations require inordinate amounts of time to specify the simulation inputs, which may require numerous flowpath dimensions and the geometry of the layout.

2.3.1.4 Combining Phenomenology and Reliability

Brute Force

Combination of phenomenology and reliability is often difficult. Simulators are available that are efficient for simulating system phenomenology (more efficient for single-physics than for multi-physics), provided the effects of component failures can be imposed on the time evolution through boundary conditions. Also, simulators are available that are efficient for simulating time histories of systems whose component states can be represented in terms of discrete (logic) variables, provided that those component states are not influenced by phenomenology. But the general state of practice is not yet efficient for simulating combined effects. The desired state of affairs is suggested below in Fig. 2-37. It is difficult to put phenomenology-dependent component state transitions into the phenomenology simulators that are in wide use, and then account for the effects of those component state transitions on the subsequent evolution of the phenomenology. Phenomenology solvers may respond badly if asked to deal with a discontinuity in the system whose phenomenology they are trying to simulate.

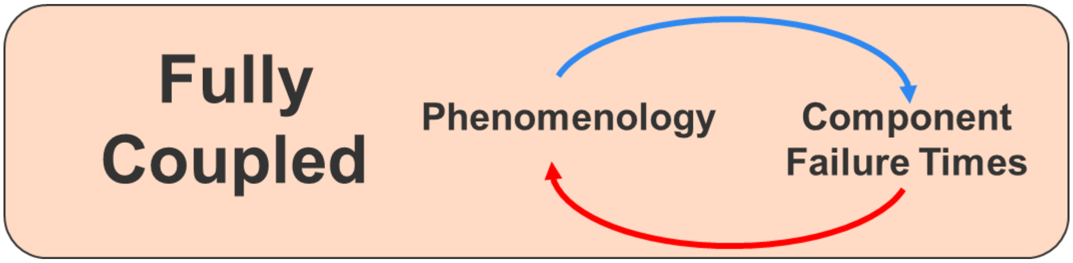


Figure 2-37. Coupling between phenomenology and component failure times.

Heartbeat Model [2-9]

Recently, a step was taken making it more feasible to include, in a single simulator, the effects of multi-physics, the effects of component state changes, and the effects of two-way coupling between component states and phenomenology. Details of the solver have been presented elsewhere. The general idea is that everything is handled by a single solver, which takes time steps that are determined by the most rapidly varying thing in the simulation, and/or impending near-discontinuities. Representative near-discontinuities are things like safety valve lifts, random component failures, or environmentally-induced component failures. These can be near-discontinuities because they can induce very abrupt changes in the physics of the evolving system. Refer to Figure 2-38, which notionally displays the time steps taken by the solver, as a function of what is happening.

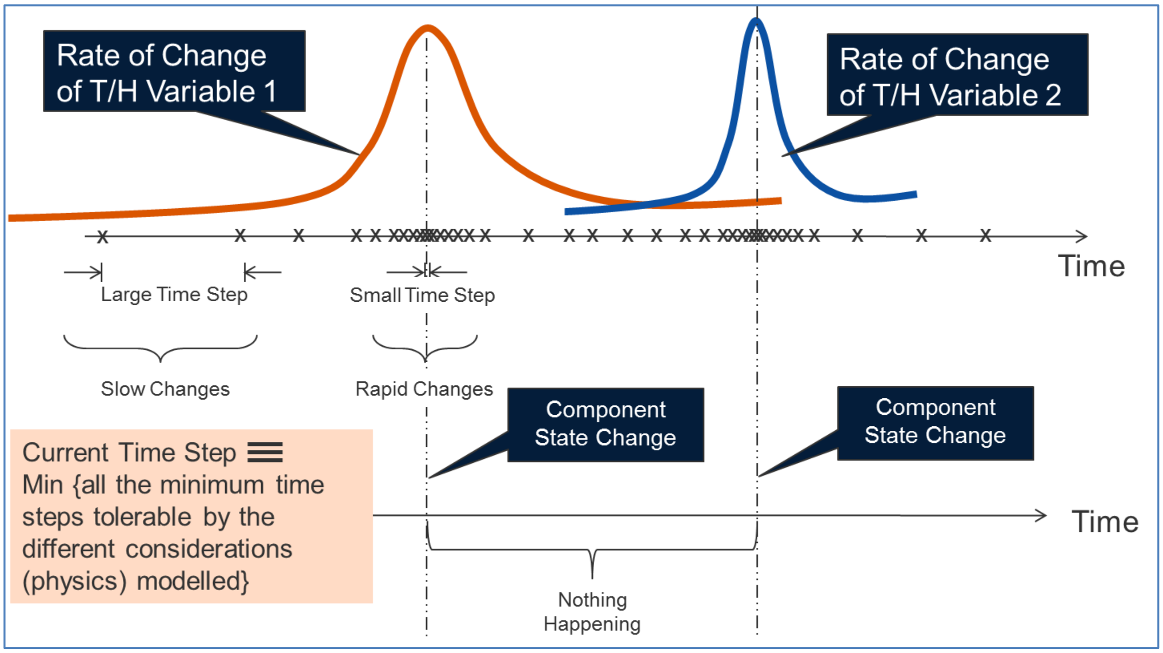

Rate of Change of T/H Variable 1
Rate of Change of T/H Variable 2
Time
Large Time Step
Small Time Step
Slow Changes
Rapid Changes
Component State Change
Component State Change
Current Time Step ≡ Min {all the minimum time steps tolerable by the different considerations (physics) modelled}
Time
Nothing Happening

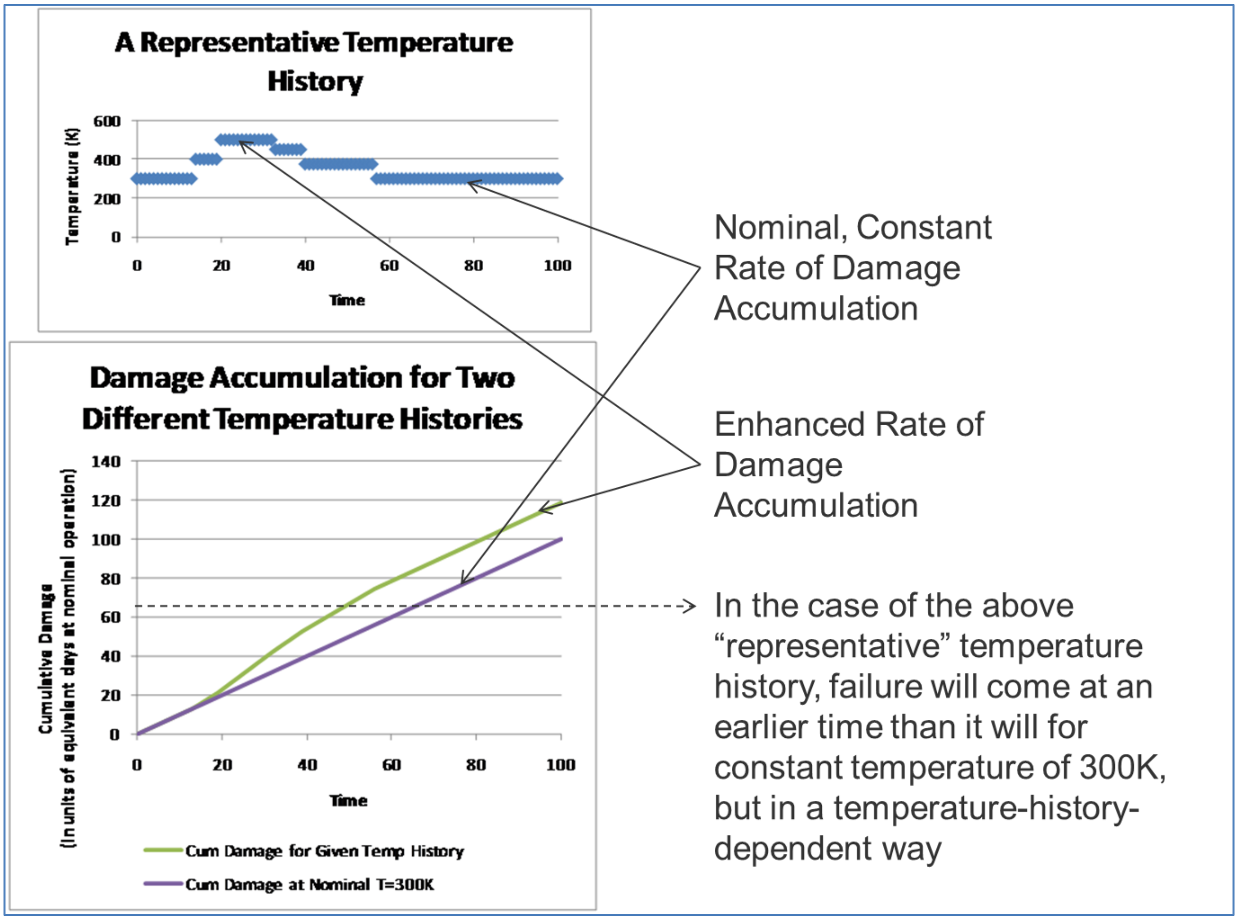
Figure 2-38. Notional time stepping based on activity.

Part of the traditional difficulty of coupling component states to phenomenology has to do with the habit of regarding component failures as stochastic in nature, because reliability formulae superficially resemble formulae for radioactive decay. Arguably, as installed, each component has a specific effective lifetime under nominal operating conditions, due to how it was manufactured. But the actual lifetime of a specific component is unknown *a priori*; what is known about the lifetime is the distribution of failure times that characterizes the population of components from which the component is drawn.[f] This is reminiscent of an old popular idea that each human is born with a certain number of heartbeats, varying from individual to individual, and when an individual's quota is consumed, the individual dies. A lifestyle that induces a high average heart rate will lead to an early death; a lifestyle that induces a low average heart rate will prolong life. The high-average-heart-rate situation is analogous to premature expenditure of those heartbeats leads to early death; and analogously, operating components in a harsh environment (for example, high temperature), which may hasten their demise as well.

Within this picture, we can model a system of components can be modeled by specifying their nominal failure times as initial conditions of each simulated time history; then, within the time history

[f] This is reminiscent of an old popular idea that each human is born with a certain number of heartbeats, varying from individual to individual, and when an individual's quota is consumed, the individual dies.

simulation, we model the *changes* to that destined failure time as the result of harsh environments. An example is illustrated in Figure 2-39.



Nominal, Constant Rate of Damage Accumulation

Enhanced Rate of Damage Accumulation

In the case of the above "representative" temperature history, failure will come at an earlier time than it will for constant temperature of 300K, but in a temperature-history-dependent way

Figure 2-39. Example time history effect on damage.

In Figure 2-39, two time histories are illustrated: one proceeds at nominal temperature, and the other includes a transient increase in temperature. When the temperature increases, damage (or effective component age) accumulates at a greater rate, and the component fails at an earlier time than it would have without the transient. In the figure, the horizontal dotted line at around 63 is the sampled, component-specific, time-history-specific failure threshold that amounts to an initial condition for the time history to be simulated; all else is determined mechanistically.

The above approach was shown to be practical in an *ab initio* development of a thermal-hydraulic simulator [2-9], but that does not mean that any existing simulator can easily implement the idea. The present point is that it is feasible in principle to apply simulation in this way, and for some purposes, it may be necessary to do so.

The Limit Surface [2-10]

In the above subsections, we have tacitly assumed that the goal is to explore an "issue space" of phenomenological possibilities, initial and boundary conditions, and component states by simulating a sufficiently large number of time histories to be able to understand the kinds of conditions likely to lead to adverse consequences, and how likely they are. If we are able to simulate a sufficiently large number of time histories, we will be able to quantify the probability of adverse consequences, and understand what is driving that probability. However, it may be difficult to simulate enough time histories to obtain adequate statistics on the probability of failure, or make useful statements about the conditions that lead to failure, which may limit the ability to prevent failures through proactive risk management.

In order to help work around the difficulty of building up statistics, the “limit surface” idea was proposed around 1990 [2-11]. The point is that instead of sampling the issue space randomly, and doing the implied simulations, one drives the simulator in an adaptive, searching mode to determine the “limit surface”: the surface in the issue space that separates success from failure. Depending on the dimensionality of the problem, the savings in simulation time can be enormous. In exploring an event whose probability is on the order of 1E-4, multiple tens of thousands of runs will get a decent estimate by brute-force Monte Carlo; however, we may be able to determine the limit surface to sufficient accuracy in a few hundred runs, and thereafter trivially calculate the failure probability.

A side benefit of this approach is that the limit surface does not depend on the probability distributions of the underlying parameters, so once the surface is determined, the sensitivity of performance to any perturbation in those distributions can readily be determined. The limit surface is a continuous-variable analog of the minimal cut sets, which likewise do not depend on basic event probability.

2.3.1.5 Summary. In the 1970s and 1980s, logic modeling was coupled to phenomenology simulation, but only rather loosely. A logic modeler needs to begin with a statement of what success and failure mean in terms of component states. There is always some approximation involved in this. A highly precise statement of success of a cooling system will almost surely be wrong a small fraction of the time: when performance is near the threshold, a given flow may be success in some cases, and failure in others. Much of the time, this does not matter, but some of the time, it does.

It has been found that judgment-based mission success criteria can be seriously off, for reasons of system complexity and scenario timing.

For these reasons, some analysts consider that for high-end applications, a much more simulation-based approach to risk analysis should replace logic modeling. However, there are difficulties with this. Reliability can be analyzed well enough for many purposes, and phenomenology can be analyzed with some difficulty; but analyzing them together cannot be said to be the current state of practice. Moreover, as computational progress continues to be made and modeling improves, the more detailed inputs must still be developed to these models, and for some facility types, this development is already an enormous effort.

The above sections have summarized why one would try certain approaches, and have cited recent progress in some areas. For now, managers and principal investigators need to consider carefully the *intended* application (and, arguably, the *potential* applications) of the analysis being undertaken, and formulate the analysis approach accordingly.

2.3.2 Examples

2.3.2.1 Production Facility Fire/Explosion Probabilistic Risk Assessment. Fire and explosion are potentially large risks on an offshore hydrocarbon production facility and can lead to personnel fatalities and/or significant damage. This section describes the basic steps that can be used to incorporate this type of phenomenological event into a PRA.

System Familiarization and Hazard Identification

Production facilities can be large and complex; therefore, the first step in the risk assessment is becoming familiar with the facility. Several common sources of data used for this are:

- Piping and instrumentation diagrams
- Process flow diagrams, including heat and material balances

- Safety Analysis Function Evaluation (SAFE) charts
- General arrangement drawings for fixed platforms and typical drilling support rigs.

Once the basic layout and process of the facility are understood, hazard identification is performed qualitatively. This may already exist in the form of a HAZOP analysis or other qualitative technique. The facility can then be further defined into sections of interest. This can be done by breaking the facility up into isolatable sections of the processing plant, which then allows the scenarios to be defined.

Scenario Definition

To develop the scenarios, an ESD may be useful. Figure 2-40 shows a simple example ESD for a single isolatable section of a facility.

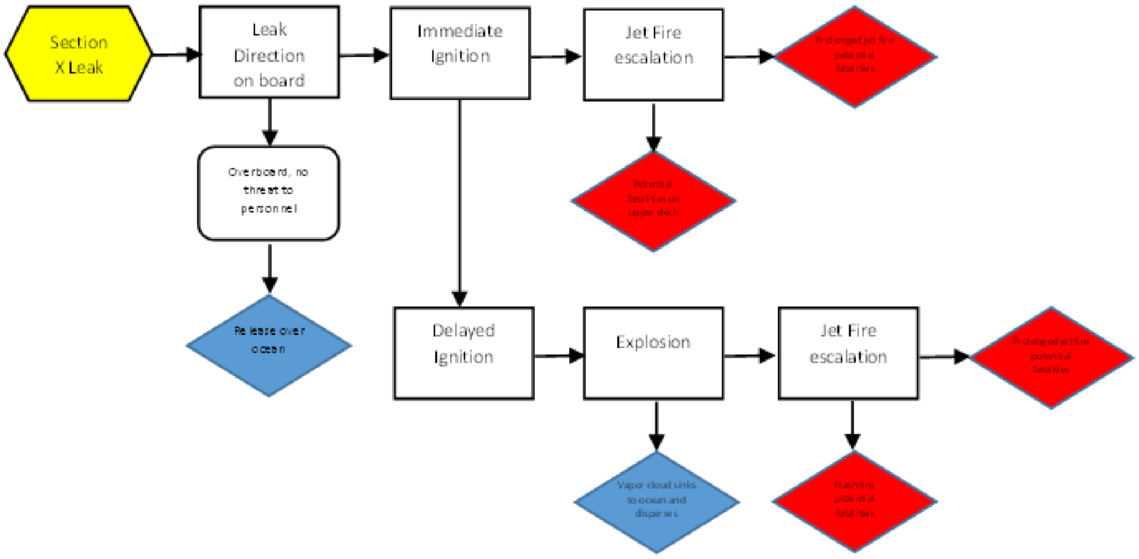

Section X Leak
Leak Direction on board
Immediate Ignition
Jet Fire escalation
Prolonged jet fire potential fatalities
Overboard, no threat to personnel
Potential fatalities on upper deck
Release over ocean
Delayed Ignition
Explosion
Jet Fire escalation
Prolonged jet fire potential fatalities
Vapor cloud sinks to ocean and disperses
Flash fire potential fatalities

Figure 2-40. Example ESD for a process system leak.

Each isolatable section is reviewed for the components included in that section. The failure frequency of these components is the basis for the IE frequency for a hydrocarbon leak, which can lead to a fire or explosion.

Leak size is also important as it affects release rates, flame shape/length, and the duration of a release of a given section of the facility. The initial release rate depends on the on the section pressure and on the size of the hole, and on whether the fluid is gas, liquid, or two-phase. Multiple leaks sizes may need to be analyzed as the consequences may vary. The frequencies for all leak sizes must sum to the total failure frequency for the section. Leak sizes may be obtained from sources such as Lee's *Loss Prevention in the Process Industries* [2-11].

Ignition probabilities, as well as the probability of an explosion if ignition occurs, can be found in sources such as [2-11]. Each section will have an associated hazardous inventory during normal operation and an operating pressure. Using commercially available software such as SAFER, TRACE^{TM}, or PHAST, the section design information along with any other applicable information (e.g., site meteorology) may be used to determine consequence characteristics such as jet flame length, duration, and explosion overpressure as a function of distance.

If no specific information is available to associate a leak with a particular direction, the direction may be taken to be random with each of the cardinal directions and up and down. Leaks with immediate ignition generally have high momentum and will not be significantly changed by wind effects. However, unignited releases should account for average meteorological conditions.

With the probabilities of ignition and explosion developed, the escalation event is evaluated. Escalation may result from failure to isolate a section resulting in a higher mass of hydrocarbons to release, or damage to other pressure vessels from flame impingement or explosion following the initial release. Failure to isolate a section is handled as an equipment failure and treated like other component failures in the PRA. The results of the phenomenological analysis may be used to determine the probability of escalation due to equipment failure following the initial release. Using the general arrangement drawings and output of the phenomenological analysis (e.g., flame length, duration of release, etc.), significant sources of hydrocarbons in the affected area can be analyzed to determine whether they are at risk. Using the duration and heat flux from the consequence analysis, an estimate can be made as to the probability of escalation due to other pressure vessel/equipment failures.

The effect on personnel may be estimated by first determining the probability that personnel will be in an affected area. This can be done by reviewing operating and maintenance practices to determine the frequency and duration of maintenance in specific areas. Some areas, such as a control room, may be crewed at all times, while others may have a very low probability of having personnel in in them. Using the heat flux and overpressure results by location, immediate fatalities may be estimated.

2.3.2.2 Discrete Event Simulation

Section 2.2 described logic modeling, in which scenarios are described using logic (true / false) variables. By their nature, logic modeling techniques discretize the scenario descriptions, and thereby introduce approximations. For example, as illustrated earlier, event-tree end states are typically broadly specified in terms of categories, such as “large” and “limited” breaches of containment, rather than as specific magnitudes of breaches. It is possible to improve significantly on these approximations using discrete-event simulation (e.g., estimate the number of deaths or barrels of oil spilled).

Discrete event simulation modeling is similar to developing an ESD, as discussed in Section 2.2. Time-ordered events are developed and decision blocks are used with probabilities that direct the flow of the simulation. In addition, events can be used to simulate variables such as well flow rates and recovery times. The model is run by performing numerous replications (i.e., thousands or more) using Monte Carlo sampling to obtain the probabilities or values at each decision event in the model, and the outcome of each replication is recorded. Obtaining a sufficient number of replications is important to ensure that all desired events get sampled and all reasonable paths in the model are exercised. For example, if a decision block has a probability of 0.01 (1 in 100), running 100 replications would, on average, only go down that path once. A single data point on a path would fail to show the range of outcomes for that path.

In the example discussed below, the specified outputs are the duration and magnitude of a release based on the example developed in Figure 2-13.

The model was developed as an extension to the event-tree model to estimate the environmental leakage based on the scenarios modeled. Three end states were used to determine what sealed the well and prevented further release; the ROV, well capping, or a relief well. In the case where well capping was used to stop flow, the cut sets were broken down further to try to estimate to determine whether the well cap could be applied to the BOP as it sat (in a vertical position), whether the BOP needed to be adjusted to a more vertical position (if it was tilted [i.e., did not completely disconnect when required]), or whether the well cap had to be applied to the wellhead.

A discrete event simulation model for these conditions was developed, and is shown in Figure 2-41. There are three main types of blocks used in the model. The Assign block provides the value of a parameter such as well flow rate. The Decide block allows the direction of flow to be chosen from multiple potential paths, and the Process block captures the delay time in the successful method to stop the flow.

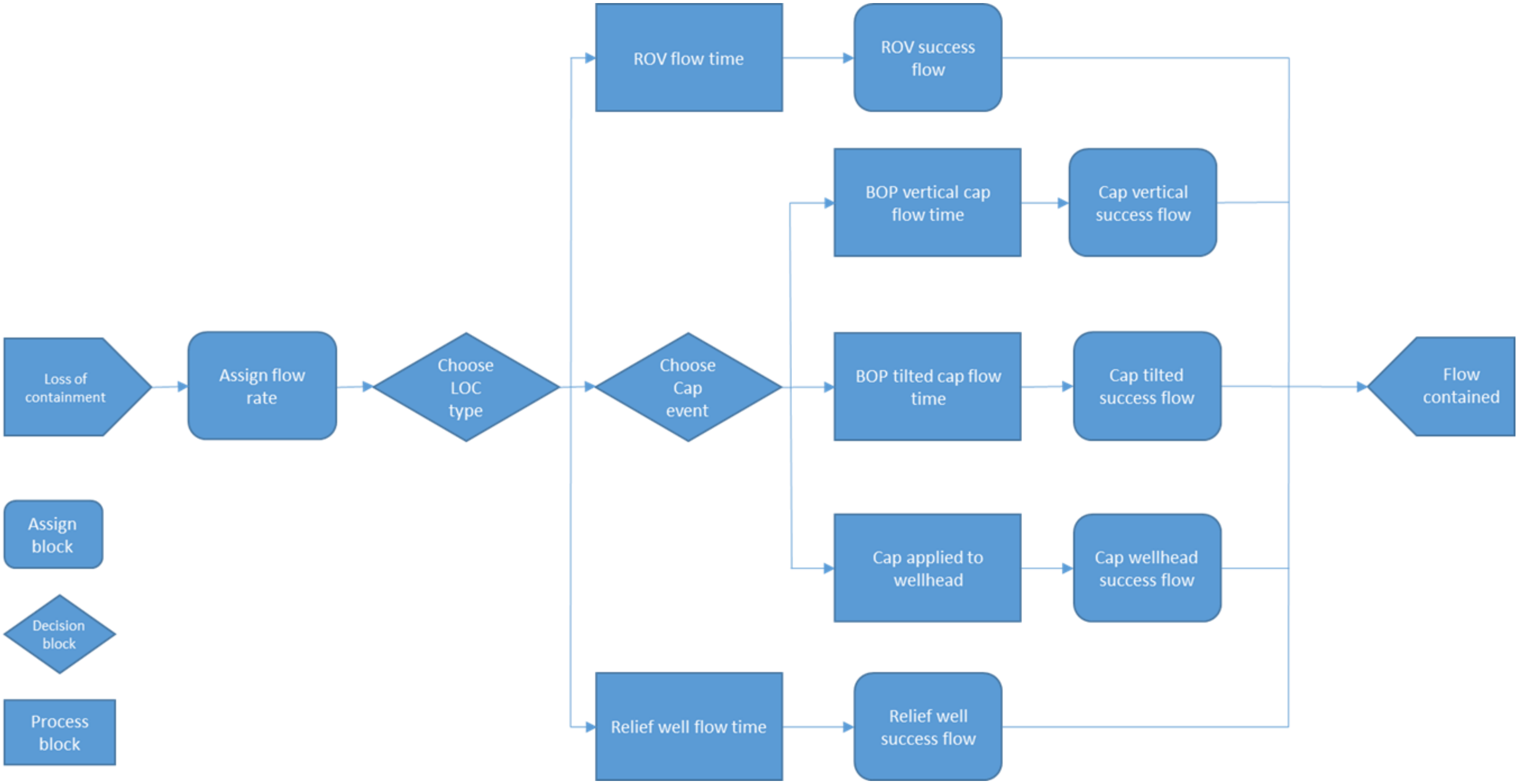


Figure 2-41. Example discrete event simulation model.

The total probability of a large environmental release from the event-tree model is about 5.0E-5 with the probabilities for the three end states shown in Table 2-5.

Table 2-5. End state probabilities for discrete event simulation model.

	Probability	Normalized
P(ROV)	1.64E-06	0.0327
P(WELL CAP)	4.39E-05	0.8793
P(RELIEF WELL)	4.39E-06	0.0879

The events are normalized to allow faster model runs than if the actual probabilities were used. Well capping was broken down further by assuming that the probabilities for a vertical BOP, tilted BOP, and using the wellhead were 0.9, 0.09, and 0.01, respectively.

Distributions for well flow rates and timing of recovery events, such as when an ROV successfully manipulates the BOP, may be added as histograms as shown in Figures 2-42 and 2-43.

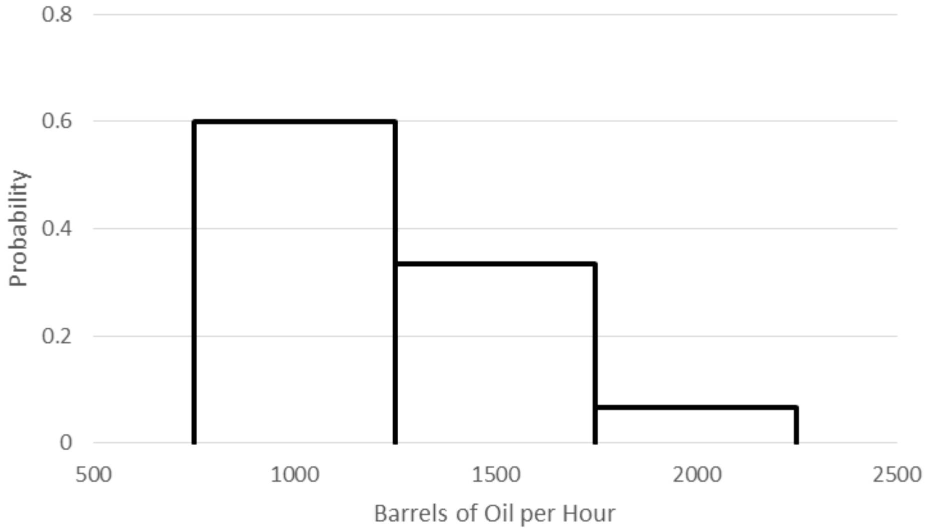


Figure 2-42. Example histogram of probability versus well flow rate.

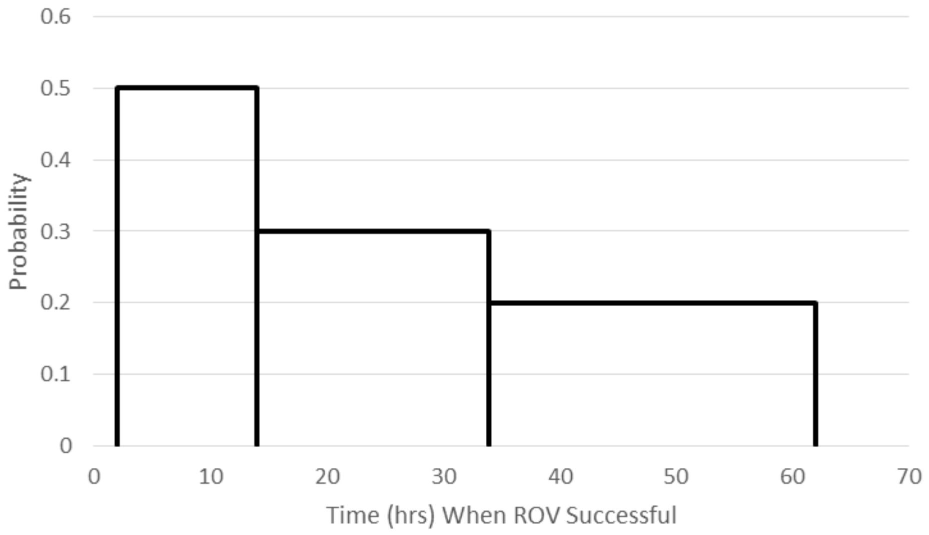


Figure 2-43. Example probability of success versus time for ROV.

Results from the discrete event simulation, shown in Table 2-6, are a sample of results from model simulation replications showing the end event and parameters of interest, in this case the time to shut in the well and the number of barrels of oil released. The replication column shows on which model simulation run the event occurred. The results can be manipulated to develop different kinds of useful products such as frequency of exceedance (F-N) curves discussed in Section 4.

Table 2-6. Discrete event simulation model results.

Replication No.	End Event (what stopped flow)	Duration of Release (hrs)	Barrels of Oil Released
1	CAP with vertical BOP	223	199,660
2	Relief well	3,089	2,474,100
3	CAP with vertical BOP	260	211,600
4	CAP with tilted BOP	519	906,370
5	CAP with vertical BOP	183	313,090
6	CAP with vertical BOP	176	157,030
7	CAP with tilted BOP	367	619,310
8	CAP with vertical BOP	288	251,220
9	CAP with vertical BOP	148	174,060
10	CAP with vertical BOP	164	240,840

2.4 References

2-1 NASA, “Fault tree Handbook with Aerospace Applications,” Version 1.1, August 2002.

2-2 A. Mosleh *et al.*, *Procedures for Treating Common Cause Failures in Safety and Reliability Studies*, U.S. Nuclear Regulatory Commission and Electric Power Research Institute, NUREG/CR-4780, and EPRI NP-5613. Volumes 1 and 2, 1988.

2-3 Lee J. C., and N. J. McCormick, Risk and Safety Analysis of Nuclear Systems, Wiley, 2011.

2-4 EPRI, *GO methodology: prepared for Electric Power Research Institute*, Vol. 1, Overview Manual, EPRI NP-3123-CCM, EPRI, Palo Alto, California, 1983.

2-5 MiTeck, *SAPHIRE Training Manual*, 2010.

2-6 NRC, *The Reactor Safety Study The Introduction of Risk Assessment to the Regulation of Nuclear Reactors*, WASH-1400, NUREG/KM-0010, U.S. Nuclear Regulatory Commission, 1975.

2-7 EPRI, *Technical Framework for Management of Safety Margins—Loss of Main Feedwater Pilot Application*, EPRI Report 1023032, Electric Power Research Institute, 2011.

2-8 NAVSEA, "TIGER User's Manual (Version 8.21)," NAVSEA TE660-AA-MMD-010, 1987.

2-9 R. W. Youngblood, R. R. Nourgaliev, D. L. Kelly, C. L. Smith, and T-N. Dinh, "Heartbeat Model for Component Failure Time in Simulation of Plant Behavior," *ANS PSA 2011 International Topical Meeting on Probabilistic Safety Assessment and Analysis, Wilmington, North Carolina, March 13–17, 2011*, American Nuclear Society, LaGrange Park, Illinois, 2011.

2-10 J. Vandenkieboom, R. Youngblood, J. C. Lee, and W. Kerr, "Reliability Quantification of Advanced Reactor Passive Safety Systems," *Transactions of the American Nuclear Society*, Vol. 76, 296, American Nuclear Society, Inc., La Grange Park, Illinois, 1997.

2-11 Lees, F, Loss Prevention in the Process Industries, Volume 1, 3rd Edition Elsevier, Butterworth, Heinemann, 2005.

3. DATA DEVELOPMENT/ QUANTIFICATION

For purposes of this section, it is assumed that implementation of the processes previously described has developed a scenario set (a collection of scenarios leading to some consequence); their frequencies (or their conditional probabilities) and perhaps their consequences need to be modeled, and these activities need to be carried out with appropriate regard for uncertainty. Those tasks are the subjects of the present section.

In order to claim a low level of risk, it is necessary either to be able to argue some kind of inherent safety (the situation is safe because of physical laws), or to develop a lot of evidence and argument regarding the performance of the engineered systems. In the latter case, the key claims are:

- For the given design built and maintained according to specified engineering standards, and for a stated body of operating procedures and conventions, we have identified the scenarios that lead to the undesirable consequences, and the severity of their consequences.
- We know how to quantify the scenarios:
    - We know what probabilities to assign to basic events, and to combinations of basic events;
        - We have analyzed what levels of reliability performance are achievable, and we know (modulo some uncertainty) what it takes to achieve them.
        - We have analyzed the potential for linkages between the occurrences of various combinations of basic events, and have factored this into our quantification (with uncertainty).
- We commit to the measures needed to attain (and maintain, and assure, and demonstrate on an ongoing basis) the levels of reliability performance credited in the analysis.

Event probabilities are not constants of nature. To assign a low failure probability to an engineered system or component is to take credit for an engineering accomplishment, and the results of the analysis are conditioned on that credit. The reason for reviewing operating experience as part of data development is not that past performance guarantees future performance; operating experience must be reviewed (1) in order to understand how past engineering investments have panned out in past performance, (2) as a sanity check of the numbers that we put into the analysis, and (3) as a guide to the insurance activities to which we need to commit if the risk estimates are to come true.

3.1 Quantification of Individual Scenarios

3.1.1 Quantifying the Probability/ Frequency of Individual Basic Events

Historically, many analyses have implicitly treated PRA input numbers as if they were objectively significant: uncertain, to be sure, but having some objective (albeit unknown) value, analogous to the value of a physical constant, in the sense that it can be looked up. In some cases [3-1, 3-2] it is recognized that basic event probabilities may be, in effect, influenced by operating conditions at the subject facility.

But in some applications, there seems to be a tacit supposition that a PRA result is an attribute of a facility. This interpretation is inappropriate. A more complete interpretation is the following.

Assuming that:

- The scenario model is structurally complete (it addresses all IEs, accounts for dependencies of all types, CCF, etc.)
- The data used to quantify the basic events are relevant (the data are derived from components appropriately similar, with similar service conditions, maintenance practices, and ages),

- Operational practices similar to those to which the performance data pertain will be followed,

we can reasonably hope to achieve performance comparable to the performance achieved by the facilities represented in the data base.

In most cases, the basic event quantifications are not guarantees of future performance: they are simply PRA inputs, implicitly tied to a commitment of sorts (to the regulator) to an investment in achieving the level of reliability performance claimed. We will return to this topic in Section 4 of this guide.

The following paragraphs discuss a range of techniques for basic event quantification: the values for frequency (or, as appropriate, probability) of such things as component failures, IEs, and human failures.

Figure 3-1 shows the context of an existing scenario model to be quantified. The scenario model should have been developed down to a level of detail at which the basic events are largely independent. This makes it possible to directly apply data on how often the basic events occur. The present discussion focuses on events that can be quantified from experience. Common-cause failures and failure events that are driven by scenario-dependent conditions are discussed in other sections.

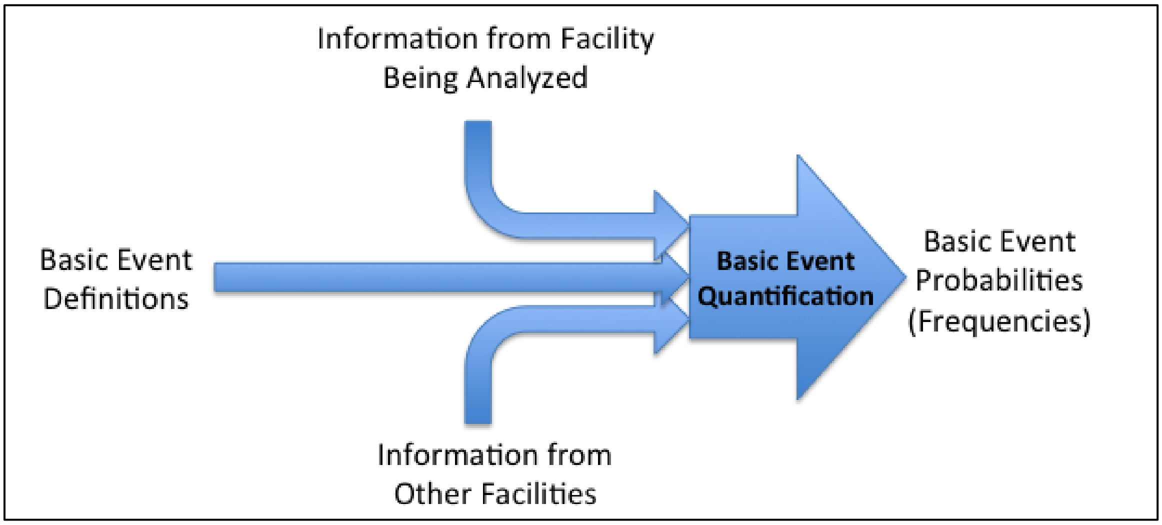

Information from Facility Being Analyzed
Basic Event Definitions
Basic Event Quantification
Basic Event Probabilities (Frequencies)
Information from Other Facilities

Figure 3-1. Sources of information for quantification of basic event likelihood.

The Scenario Context of Basic Events

Normally, the scenario model is developed in such a way that each scenario can be expressed in a narrative way. For example:

> *Initiating event IE1 occurred; the intended response was for Valve A to open, but Valve A failed to open; in that circumstance, Pump C was required to operate, and to continue to operate for at least 6 hours, but it was unavailable at the time. As a result of this chain of events, the top event occurred.*

Per this narrative, in order to model this scenario, the analyst needs to know how often IE1 occurs (the characteristic number of failures in a given time interval), the fraction of demands in which Valve A fails to open after IE1 occurs, and the fraction of time that Pump C is unavailable (under repair, for example) in the facility states within which IE1 can occur. These are only examples, but serve to suggest the kinds of information that we would hope to obtain from operating experience.

3.1.2 Estimating Parameters in Models for Basic Event Probability (Frequency)

The two main phases of developing a PRA database are:

- Information Collection and Classification
- Parameter Estimation.

Typical quantities of interest are:

- IE Frequencies
- Component Failure Frequencies
- Component Test and Maintenance Unavailability
- CCF Probabilities
- Human Error Rates
- Software Failure Probabilities.

Developing a PRA database of parameter estimates involves the following steps:

- Model-Data Correlation (identification of the data needed to correspond to the level of detail in the PRA models, determination of component boundaries, failure modes, and parameters to be estimated, such as failure rates, Mean Time To Repair (MTTR), etc.
- Data Collection (determination of what is needed, such as failure and success data to estimate a failure rate, and where to get it [i.e., identification of data sources, and collection and classification of the data])
- Parameter Estimation (use of statistical methods to develop uncertainty distributions for the model parameters)
- Documentation (how parameter uncertainty distributions were estimated, data sources used, and assumptions made).

Typical PRA parameters and the underlying probability models are summarized in Table 3-1. We do not simply examine experience and directly obtain a number (for a probability or frequency) that we can use in our scenario quantifier; rather, we *model* the probability or frequency in terms of underlying parameters, which we seek to learn from experience. Typically, there is epistemic uncertainty about the values of these underlying parameters, and carrying this uncertainty through the quantification can be important. Parameters for which there is epistemic uncertainty are shown in bold in Table 3-1.

Table 3-1. Typical probability (or frequency) models in PRAs and their parameters.

Basic Event Type	**Commonly Used Models of Basic Event Probability**	**Data Required to Quantify Models**
Initiating event	Poisson model for probability of seeing k events in time t: $Pr(k) = e^{-\lambda t}\frac{(\lambda t)^k}{k!}$ where t: Mission time λ: frequency	Number of events k in time t
Component fails on demand	Constant probability of failure on demand, or q	Number of failure events k in total number of demands N

Table 3-1. (continued).

Basic Event Type	Commonly Used Models of Basic Event Probability	Data Required to Quantify Models
Standby component fails in time, or component changes state between tests (faults revealed on functional test only)	Constant standby failure rate $Q = 1 - \frac{1 - e^{-\lambda_s T_s}}{\lambda_s T_s}$ T_S: Time between tests λ_S : Standby failure rate	Number of events k in total time in standby T
Component in operation fails to run, or component changes state during mission (state of component continuously monitored)	Constant failure rate $U = 1 - e^{-\boldsymbol{\lambda}_0 T_m} \approx \boldsymbol{\lambda}_0 T_m$ T_m: Mission time $\boldsymbol{\lambda}_0$: Operating failure rate Approximation is adequate when $\boldsymbol{\lambda}_0 T_m \ll 1$	Number of events k in total exposure time T (total time standby component is operating, or time the component is on line)
Component unavailable due to test	$Q = \frac{T_{TD}}{T_S}$ T_{TD} : Test duration (only in the case of no override signal) T_S: Time between tests	Average test duration (T_{TD}) and time between tests (T_S)
Component unavailable due to corrective maintenance (fault revealed only at periodic test, or preventive maintenance performed at regular intervals)	$Q = \frac{T_U}{T_T}$ T_U: Total time unavailable while in maintenance (out of service) T_T: Total operating time	Total time out of service due to maintenance acts while system is operational, T_u, and total operating time T_T.
Component unavailable due to unscheduled maintenance (continuously monitored components)	$Q = \frac{\mu T_R}{1 + \mu T_R}$ T_R: Average time of a maintenance outage ["Repair time"]. $\boldsymbol{\mu}$: Maintenance rate	Number of maintenance acts r in time T (to estimate μ)
Standby component that is never tested. Assumed constant failure rate.	$Q = 1 - e^{-\lambda_m T_p}$ T_p : Exposure time to failure λ_m : Standby failure rate.	Number of failures r, in T units of (standby) time
Common-Cause Failure Probability	α_1 through α_m, where *m* is the redundancy level	n_1 through n_m where n_k is the number of CCF events involving k components
Note: Model parameters for which there is epistemic uncertainty are shown in bold in the center column of the table. Data needed to estimate those parameters are listed in the right-hand column. Other model parameters (such as "mission time") are determined by the application.		

Table 3-1 also shows the data needed to estimate the various parameters. The type of data needed varies depending on the type of event, and on how its frequency or probability is modeled. For example, probabilities typically require event counts (e.g., number of failures), and exposure or success data (e.g., total operating time). Other parameters may require only one type of data, such as maintenance/ repair duration for mean repair time distribution, and counts of multiple failures in the case of CCF parameter estimates. CCF parameters are discussed later in this section, while more information regarding maintenance unavailability can be found in Appendix C.

Sources of Information

Ideally, parameters of PRA models of a specific system should be estimated based on operational data of that system. As previously discussed, even past performance of that system does not guarantee future performance, for several reasons; but data from that system must be among the most relevant data available, unless something fundamental has recently changed.

If system-specific data of adequate quantity, quality, or availability are lacking, the analysis has to rely on other sources and types of information. In such cases, surrogate data, generic information, or expert judgment are used directly or in combination with (limited) system-specific data. A survey of generic information sources is given in Appendix D. It bears repeating that in submittals to regulators, the submitter is accountable for the treatment on which the conclusions are based.

Parameter Estimation Methods

Bayesian methods of parameter estimation are widely used in PRA, while classical estimation has found only limited and restricted use in PRA. Accordingly, this section describes only the Bayesian approach to parameter estimation.

Bayesian estimation incorporates information beyond that contained in the data sample; this is part of what makes Bayesian inference different from classical estimation. In practice, Bayesian estimation comprises two main steps. The first step involves using previous information to develop a prior distribution for the parameters of a basic event model, such as a failure rate. The second step of Bayesian estimation involves using additional or new data (e.g., recent performance history) to update the prior distribution, yielding a posterior distribution for the parameters of that basic event model. This step is often referred to as Bayesian updating of the prior distribution. This process is illustrated in Appendix E.

For PRA applications, determining the prior distribution is usually based on generic data, and the new or additional data usually involve system-specific test or operating data. The resulting posterior distribution would then be the system-specific distribution of the parameter. If system-specific data do not exist, the applicability of other data or information would need to be evaluated and used. Refer to Appendices D and F.

Within the standard approach, one formulates explicit state-of-knowledge probability distributions about uncertain variables, both epistemic and aleatory. If these uncertain variables are model inputs, and one has distributions for them, the distribution of the model output(s), or at least the "parameter uncertainty" portion (which, in principle, ought to be evaluated together with other uncertainties) can be inferred during quantification. Given a proper understanding of the uncertainties that affect the analysis, one can proceed to apply the standard machinery of decision-making under uncertainty.

Within the standard Bayesian approach, information is gathered about epistemically uncertain variables (or hypotheses regarding which we are uncertain), including formulation of a prior distribution on the values of those variables (or the probabilities of the various hypotheses being true); those distributions are then updated as new information becomes available, and one's state of knowledge is improved (sometimes). Bayes' so-called theorem states that:

$$p(H_i|E) = p(H_i) \times \frac{p(E|H_i)}{p(E)}, \tag{3-1}$$

where

- H_i represents a hypothesis whose probability is to be updated with new evidence
- $p(H_i)$ is the prior probability of H_i
- E represents a new piece of evidence
- $p(x|y)$ is the conditional probability of x given y
- $p(E)$, the prior probability of the observed evidence, can be written as $p(E) = \sum_i p(E \mid H_i) p(H_i)$.

Hereafter, this is referred to as the update rule. The update rule says that the conditional posterior probability of hypothesis H_i, given new evidence E, is equal to the prior probability of hypothesis H_i, multiplied by the conditional probability of observing E if H_i is true, divided by the total prior probability of observing E, calculated as shown in Eq. 3-1. In essence, new evidence that favors hypothesis H_i more than it favors hypothesis H_j (i.e., $p(E|H_i) > p(E|H_j)$) tends to increase the posterior probability of hypothesis H_i relative to the posterior probability of H_j. In accordance with the update rule, new evidence causes the probabilities of the competing hypotheses to shift towards the implications of the new evidence.

The above paragraph has been worded as if the hypotheses were discrete, but it also applies if the hypotheses are understood to refer to different possible values of a continuous variable. In the latter case, the quantity on the left is understood to be a posterior probability density function of that variable.

The form of the update rule follows easily enough starting with the identity:

$p(a)p(b|a) = p(b)p(a|b)$, for any a, b,

dividing through by p(a), and identifying b with H_i and a with E. The identity, in turn, is easily understood with reference to a Venn diagram (Figure 3-2).

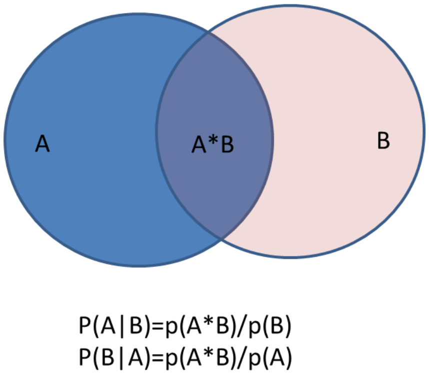


Figure 3-2. Venn diagram.

Examination of a few cases may serve to aid intuition. Suppose the hypothesis (H_i, in the earlier notation) is that an adverse condition is present in a particular system ("A" for "adverse condition is present;" refer to Figure 3-3), and we have gathered evidence E to help determine whether A is true. In Figure 3-3, the Venn diagram on the right illustrates the situation in which A and E do not overlap, so p(E|A) is zero, and the Bayes update rule will yield p(A|E)=0. The Venn diagram on the left illustrates a situation in which we see evidence E only if A is true, and putting the indicated numbers into the update rule will yield p(A|E)=1 (see Figure 3-4). The case in between – partial overlap of A and E – is where the practical applications lie.

Among the important properties of the update rule is that as new evidence is gathered, the process can be iterated; for a given collective body of evidence and a given starting prior and a given likelihood function, the same conclusion will be reached, regardless of how the evidence is parsed and applied in subsets. An illustration of this is shown in Section 1.3 of Appendix E.

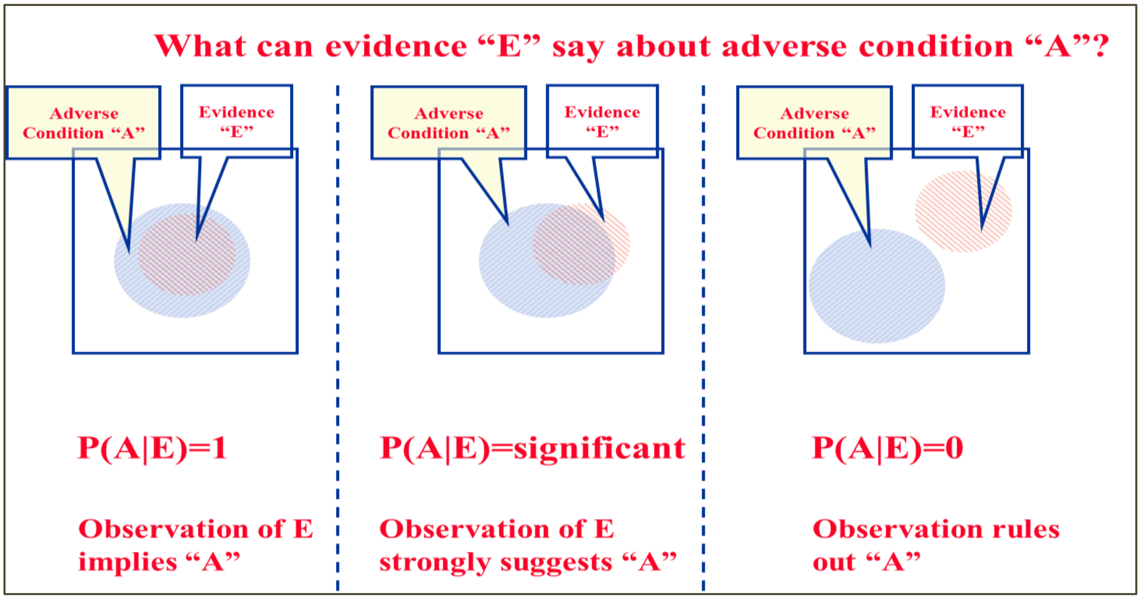
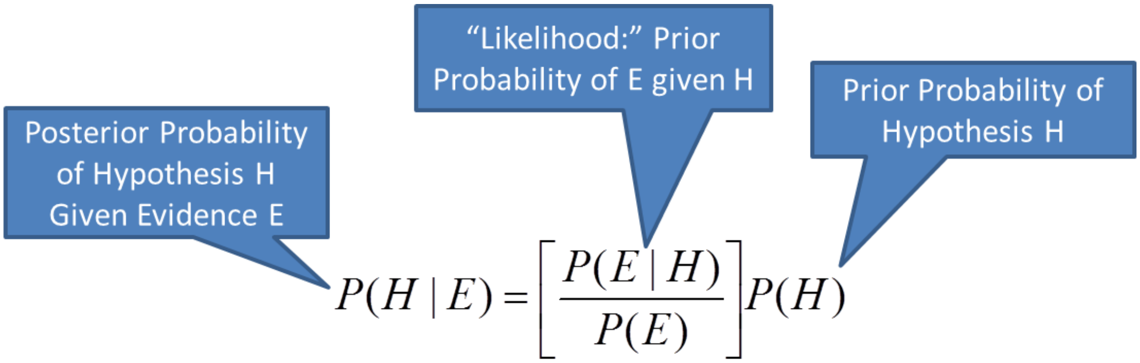

What can evidence "E" say about adverse condition "A"?
Adverse Condition "A"
Evidence "E"
Adverse Condition "A"
Evidence "E"
Adverse Condition "A"
Evidence "E"
P(A|E)=1
Observation of E implies "A"
P(A|E)=significant
Observation of E strongly suggests "A"
P(A|E)=0
Observation rules out "A"

Figure 3-3. Venn diagram illustration.


"Likelihood:" Prior Probability of E given H
Prior Probability of Hypothesis H
Posterior Probability of Hypothesis H Given Evidence E

$$P(H \mid E) = \left[\frac{P(E \mid H)}{P(E)} \right] P(H)$$

Figure 3-4. The update rule.

The preceding statement calls to mind a much stronger claim advanced by Bayesians: that all rational individuals will reach the same conclusion from a given body of evidence.

The current state of knowledge depends not only on the evidence, but also the prior distributions of the variables, and the form of the likelihood function: how the evidence is interpreted in the context of the current application. There is a vast literature on formulating the prior, but, unfortunately, some of it shortchanges the topic of the likelihood function. Further work is needed in this area.

Prior Distributions

Prior distributions can be specified in different forms depending on the type and source of information as well as the nature of the random variable of interest. Functional forms widely used in PRA of engineered systems include:

- Parametric (gamma, lognormal, beta):

  - Gamma or lognormal for rates of events (time-based reliability models)
  - Beta or truncated lognormal for event probabilities per demand
- Numerical (histogram, CDF values/percentiles)
  - Applicable to both time-based and demand-based reliability parameters.

Among the parametric forms, a number of probability distributions are extensively used in risk studies as prior and posterior distributions. These are:

- Lognormal (μ, σ)

$$\pi(x)=\frac{1}{\sqrt{2\pi}\sigma x}e^{-\frac{1}{2}\left(\frac{\ln x-\mu}{\sigma}\right)^2}, 0 < x < \infty,$$

where μ and σ are the parameters of the distribution. The lognormal distribution can be truncated (truncated lognormal) so that the random variable is constrained to be less than a specified upper bound. If this sort of truncation is applied, then the distribution needs to be renormalized.

- Gamma(α, β)

$$\pi(x) = \frac{x^{\alpha-1}\beta^{\alpha}}{\Gamma(\alpha)}e^{-\beta x}\ 0 \le x < \infty$$

where a and b are the parameters of the distribution.

- Beta(α, β)

$$\pi(x) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}x^{\alpha-1}(1-x)^{\beta-1}\ 0 \le x \le 1$$

where α and β are the parameters of the distribution.

Information content of prior distributions can be based on:

- Previous system-specific estimates
- Generic, based on actual data from other (similar) systems
- Generic estimates from reliability sources
- Expert judgment (see discussion in Appendix G)
- Ignorance (i.e., lack of applicable data).

In the above list, the first four situations lead to prior distributions that may reflect considerable uncertainty about the parameters, but nevertheless assign higher probability to some values than to others. In those cases, application of situation-specific information through the update process is supposed to drive the posterior distribution to where it needs to be (or perhaps merely to reduce the uncertainty spread in that distribution). In situations where essentially no *a priori* information exists, attempts are made to formulate a prior reflecting this ignorance. A common approach to this is using a prior distribution that is uniform (constant) over the interval of interest. *Unfortunately, despite generations of work on how best to formulate such a prior, choice of prior distribution remains a research topic. If the current decision is sensitive to the tails of the posterior distribution, extra attention to this issue is warranted.*

Selection of the Likelihood Function

The form of the likelihood function depends on the nature of the assumed Model of the World representing the way the new data/information is generated:

For data generated from a Poisson Process (e.g., counts of failures during operation), the Poisson distribution is the proper likelihood function:

$$\Pr(k|T,\lambda) = \frac{(\lambda T)^k}{k!} e^{-\lambda T} \tag{3-2}$$

which gives the probability of observing k events (e.g., number of failures of a component) in T units of time (e.g., cumulative operating time of the component), given that the rate of occurrence of the event (failure rate) is λ . The maximum likelihood estimate (MLE) of λ is:

$$\lambda_{MLE} = \frac{k}{T}\,. \tag{3-3}$$

It is also possible to combine data from several independent Poisson processes, each having the same rate. This applies to the case where data are collected on different but identical units of equipment to estimate their common failure rate. The failure counting process for each unit is assumed to be a Poisson process. In particular, suppose that the ith Poisson process is observed for time t_i, yielding the observed count k_i. The total number of event occurrences is $k = \sum_i k_i$, where the sum is taken over all of the processes, and the exposure time is $T = \sum_i t_i$. This combined evidence can be used in the likelihood function given above.

For data generated from a Bernoulli Process (e.g., counts of failures on system demands), the Binomial distribution is the proper likelihood function:

$$\Pr(k|N,q) = \binom{N}{k} q^k (1-q)^{N-k} \tag{3-4}$$

which gives the probability of observing k events (e.g., number of failures of a component) in N trials of a component (e.g., total number of tests of the component), given that the probability of failure per trial (failure on demand probability) is q. The MLE of q is:

$$q_{MLE} = \frac{k}{N} \tag{3-5}$$

Analogously to the Poisson processes discussed above, data from independent trials that are known to be exchangeable (they are known to be determined by the same q, because they are the same component or identical components operated similarly) can be pooled: the failures can be summed, $k = \sum_i k_i$, and the demands can be summed, $N = \sum_i n_i$, and the results used in the binomial likelihood formula given above.

These cases are simple ones but are widely used. Likelihood functions for parameters of physical models are discussed in Appendix E.

In some cases, resort must be had to a process that relies on experts to furnish input. This is discussed in Appendix G.

Development of the Posterior Distribution

Using the update rule in its continuous form, the prior probability distribution of a continuous unknown quantity, $\Pr_0(x)$, can be updated to incorporate new evidence E as follows:

$$\Pr(x|E) = \frac{L(E|x)Pr_0(x)}{\int L(E|x)Pr_0(x)dx} \tag{3-6}$$

where Pr(x|E) is the posterior or updated probability distribution of the unknown quantity x given evidence E (occurrence of event E), and L(E|x) is the likelihood function (i.e., probability of the evidence E, assuming that the value of the unknown quantity is x). Illustrative combinations of prior and likelihood functions as well as the form of the resulting posterior distributions are listed in Table 3-2.

Table 3-2. Typical prior and likelihood functions used in PRAs.

Functional Form of Prior	Functional Form of the Likelihood	Resulting Functional Form of the Posterior
Lognormal	Poisson	Numerical
Gamma	Poisson	Gamma
Beta	Binomial	Beta
Truncated Lognormal	Binomial	Numerical

For certain cases in the above table, the posterior has the same functional form as the prior. This occurs when there is a certain similarity between the functional form of the likelihood and that of the prior. For example, the Beta prior is proportional to powers of q multiplying powers of 1-q, as is the binomial distribution used as the likelihood; and as a result, the posterior is likewise a product of powers of q and powers of (1-q). In such a case, the update can be done analytically (in closed form). When a combination of prior and likelihood has this property, the prior is said to be conjugate to the likelihood. In the case of non-conjugate priors (e.g., the case of “lognormal * Poisson => numerical”), resort must be had to numerical integration.

Two commonly used conjugate distributions are listed in Table 3-3. The formulas used to calculate the mean of the resultant posterior in terms of the parameters of prior and likelihood functions are provided.

Table 3-3. Common conjugate priors used in reliability data analysis.

Functional Form of Prior Distribution, Mean Value	Functional Form of Likelihood	Posterior Distribution (same as prior)	Mean of Posterior
Beta (α,β), $\overline{x_{pr\imath or}} = \frac{\alpha}{\alpha + \beta}$	Binomial (k, N)	Beta	$\overline{x_{poster\imath or}} = \frac{\alpha + k}{\alpha + \beta + N}$
Gamma (α,β); $\overline{x_{pr\imath or}} = \frac{\alpha}{\beta}$	Poisson (k, T)	Gamma	$\overline{x_{poster\imath or}} = \frac{\alpha + k}{\beta + T}$

In the case of the conjugate priors listed in the above table, because we can compute the prior and posterior means in closed form, we can see how new data cause the mean to shift. In principle, a prior distribution should reflect a state of knowledge, not a choice made to avoid the need for numerical integration. This has always been true, but is even more emphatically true in light of the very real improvements in computational capability in recent generations. It used to be argued that given a halfway reasonable prior, updates with new data would eventually drive posterior distributions to where they need to be; but in practical applications, where there is not always a surfeit of new data, this ideal is not always realized.

Developing Prior Distributions from Multiple Sources of Generic Information

When multiple sources of generic data are available, the data may not be able to be pooled, and the reliability parameter of interest (e.g., failure rate) will have an inherent variability. The probability distribution representing this variability is known as a population variability distribution of the reliability parameter of interest. Refer to Appendix F for a discussion of this point.

3.2 Common Cause

3.2.1 Common-Cause Definition

Many years ago, it was realized that a significant fraction (up to around 10%) of component failure events involve failures of multiple components due to essentially the same cause; therefore, modeling multiple-failure events as if all failures were independent would seriously underestimate the number of multiple-failure events. Common-cause models are meant to address this issue. A formal definition of CCF is provided below:

> *A common cause failure event is defined as the failure (or unavailable state) of more than one like component due to a shared cause during the system mission. Viewed in this fashion, common cause failures are inseparable from the class of dependent failures and the distinction is mainly based on the level of treatment and choice of modeling approach in reliability analysis.*

Essentially, common-cause failures are dependent failures whose root causes are not explicitly modeled in a PRA. Instead, we address this category of failures by introducing common-cause basic events in the PRA logic models. Components that fail due to a shared cause normally fail in the same functional mode. The term “common mode failure,” which was used in early literature and is still used by some practitioners, is more indicative of the most common symptom of CCFs (i.e., failure of multiple components in the same mode), but it is not a precise term for communicating the important characteristics that describe a CCF event.

The following are examples of CCFs:

- Stress corrosion cracking of multiple subsea bolts due to improper coatings
- Multiple diesel generator failures due to improper maintenance
- Multiple BOP hydraulic valve failures due to contamination.

CCFs may also be viewed as being caused by the presence of two factors: a Root Cause (i.e., the reason for failure of each component that failed in the CCF event), and a Coupling Factor (or factors) that was responsible for the involvement of multiple components. For example, failure of two identical redundant electronic devices due to exposure to excessively high temperatures is not only the result of susceptibility of each of the devices to heat (considered to be the root cause in this example), but also a result of both units being identical, and being exposed to the same harsh environment (coupling factor). Since the use of identical components in redundancy formation is a common strategy to improve system reliability, coupling factors stemming from similarities of the redundant components are often present in such redundant formations, leading to vulnerability to CCF events. Therefore, CCF events of identical redundant components merit special attention in risk and reliability analysis of such systems. The remainder of this section is devoted to methods for modeling the impact of these CCF events.

The process of identifying and modeling CCFs in systems analysis usually involves two important steps:

1. Screening analysis
2. Detailed analysis.

The objectives of the screening analysis are to identify, in a preliminary and conservative manner, the potential vulnerabilities of the system to CCFs, and to identify those groups of components within the system whose CCFs contribute significantly to the system unavailability. The screening step develops the scope and justification for the detailed analysis. The screening analysis typically provides conservative, bounding system unavailabilities due to CCFs. Depending on the objectives of the study and the availability of resources, the analysis may be stopped at the end of this step, recognizing that qualitative results may not accurately represent the actual system vulnerabilities, and that quantitative estimates may be very conservative.

The detailed analysis phase uses the results of the screening step and through several steps involving the detailed logic modeling, parametric representation, and data analysis, develops numerical values for system unavailabilities due to CCF events.

3.2.2 Preliminary Identification of Common-Cause Failure Vulnerabilities (Screening Analysis)

The primary objective of this phase is to identify all important groups of components susceptible to CCF. At this stage, the point is not to miss potentially significant groups, so the screening is typically done in a simple and conservative way. Later analysis will undo any “conservatism” introduced at this stage.

Screening is done in two steps:

- Qualitative screening
- Quantitative screening.

3.2.2.1 Qualitative Screening. At this stage, an initial qualitative analysis of the system is performed to identify the potential vulnerabilities of the system and its components to CCFs. This analysis is aimed at providing a list of components that are believed to be susceptible to CCF. At a later stage, this initial list will be modified on quantitative grounds. In this early stage, deliberate conservatism is justified, because it is important not to discount any potential CCF vulnerability, unless there are immediate and obvious reasons to discard it.

The most efficient approach to identifying common-cause system vulnerabilities is to focus on identifying coupling factors, regardless of defenses that might be in place against some or all categories of CCFs. The result will be a conservative assessment of the system vulnerabilities to CCFs. However, this is consistent with the objective of this stage of the analysis, which is a preliminary, high-level screening.

From the earlier discussion, it is clear that a coupling factor is what distinguishes CCFs from multiple independent failures. Coupling factors are suspected to exist when two or more component failures exhibit similar characteristics, both in the cause and in the actual failure mechanism. Therefore, the analyst should focus on identifying those components of the system that share one or more of the following:

- Same design
- Same hardware
- Same function
- Same installation, maintenance, or operations personnel
- Same procedures

- Same system/component interface
- Same location
- Same environment.

The above list, or a similar one, is a tool to help identify the presence of identical components in the system and most commonly observed coupling factors. It may be supplemented by a system "walk-down" and review of operating experience (e.g., failure event reports). Any group of components that share similarities in one or more of these characteristics is a potential point of vulnerability to CCF. However, depending on the system design, functional requirements, and operating characteristics, a combination of commonalities may be required to create a realistic condition for CCF susceptibility. Such situations should be evaluated on a case-by-case basis before deciding on whether there is a vulnerability. A group of components identified in this process is called a common-cause component group.

Finally, in addition to the above guidelines, it is important for the analyst to review the operating experience to ensure that past failure mechanisms are included with the components selected in the screening process. Later, in the detailed qualitative and quantitative analysis phases, this task is performed in more detail to include the operating experience of the system being analyzed.

3.2.2.2 Quantitative Screening. The qualitative screening step identifies potential vulnerabilities of the system to CCFs. However, detailed modeling and analysis of all potential common-cause vulnerabilities identified in the qualitative screening may still be impractical and beyond the capabilities and resources available to the analyst. Consequently, it is desirable to reduce the size of the problem even further, in order to enable detailed analysis of the most important common-cause system vulnerabilities. Reduction may be achieved by performing a quantitative screening analysis. This step is useful for systems fault-tree analysis, and may be essential for ESD-level analysis, in which exceedingly large numbers of cut sets may be generated in solving the fault-tree logic model.

In performing quantitative screening for CCF candidates, one is actually performing a complete quantitative analysis except that a conservative and simple quantitative model is used. The procedure is as follows:

1. The component-level fault trees are modified to explicitly include a global or maximal CCF event for each component in every common-cause component group. A global common-cause event in a group of components is one in which all members of the group are assumed to fail. A maximal common-cause event is one that represents two or more common-cause basic events (see Figure 3-5). As an example of this step of the procedure, consider a group composed of three mud pumps. According to the procedure, the basic events of the fault tree involving these components (i.e., "Mud pump 1 fails to run," "Mud pump 2 fails to run," and "Mud pump 3 fails to run,") are expanded to include the basic event "CCF to run of mud pumps 1, 2, and 3," which is defined as the concurrent failure of Pumps 1, 2, and 3 due to a common cause, as well as "MUD-PMP-FTR-001," "MUD-PMP-FTR-002," and "MUD-PMP-FTR-003," denoting the independent failure of Pumps 1, 2, and 3, respectively.

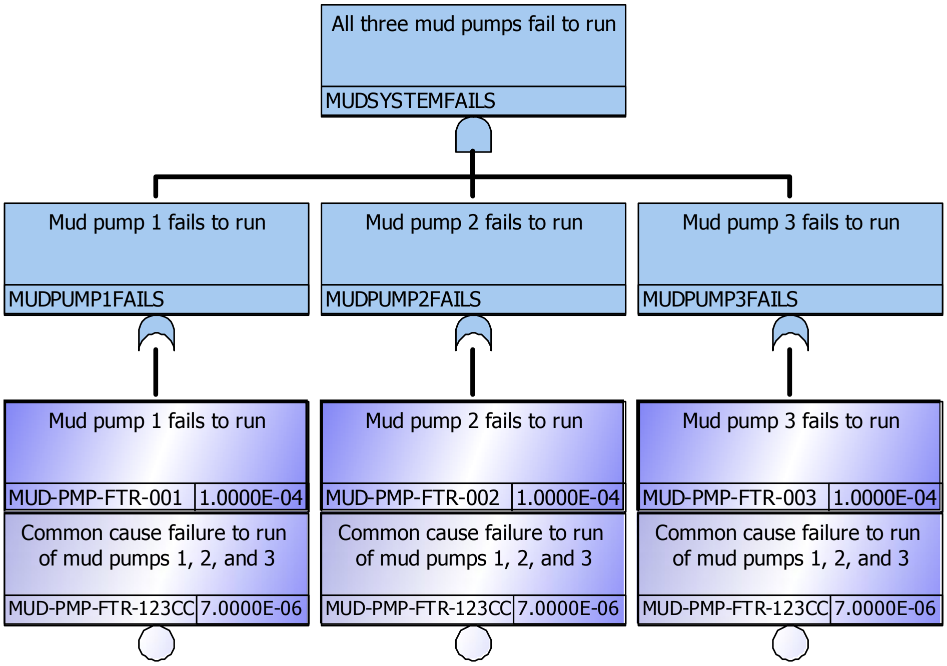


Figure 3-5. Example use of a global common-cause term.

2. Numerical values for the common-cause basic event can be estimated using a simple global parametric model:

   $$\Pr(\text{MUD-PMP-FTR-123CC}) = g\,\Pr(\text{MUD-PMP-FTR-001}) \qquad (3\text{-}7)$$

   Pr(MUD-PMP-FTR-001) is the fail-to-run probability of the component (this would be the same for Pumps 2 and 3). Typical generic values for "g" range between 0.05 and 0.10, but more accurate generic values that consider different logic configuration (k-out-of-n) can also be used. Table 3-4 lists values of the global common-cause factor, g, for dependent k-out-of-n system configurations for success. The basis for these screening values is described in Reference [3-3]. Note that different g values apply depending on whether the components of the system are tested simultaneously (non-staggered) or one at a time at fixed time intervals (staggered). More details on the reasons for the difference are provided in Mosleh *et al.* [3-3].

   The fault trees are now solved to obtain the minimal cut sets for the system or accident sequence. Any resulting cut set involving the independent failure of all three pumps will have an associated cut set involving the CCF of all three pumps. The significance of this process is that, in large system models or event sequences, some truncation of the cut sets on failure probability may be performed to obtain any solution at all; the product of independent failures is often lost in the truncation process due to its small value, while the (numerically larger) common-cause term will survive. The cut sets from the above mud pump example are:

   $$\text{MUD-PMP-FTR-001, MUD-PMP-FTR-002, MUD-PMP-FTR-003} = 1.0\text{E-}12 \qquad (3\text{-}8)$$

   $$\text{MUD-PMP-FTR-123CC} = 7.0\text{E-}6 \qquad (3\text{-}9)$$

   This simple example shows the difference between the independent-failure cut set and the common-cause cut set. Because of its low probability, the independent-failure cut set could easily be truncated from the results, leaving no indication of the significance of this particular combination of failures.

Table 3-4. Screening values of global CCF (g) for different system configurations.

	Values of g	
Success Configuration	**Staggered Testing Scheme**	**Non-staggered Testing Scheme**
1 of 2	0.05	0.10
2 of 2		
1 of 3	0.03	0.08
2 of 3	0.07	0.14
3 of 3		
1 of 4	0.02	0.07
2 of 4	0.04	0.11
3 of 4	0.08	0.19
4 of 4		

Those common-cause component groups that are found to contribute little to system unavailability or event sequence frequency (or which do not survive the probability-based truncation process, even with this quantification) can be dropped from further consideration. Those that are found to contribute significantly to the system unavailability or event sequence frequency are retained and further analyzed using the guidelines for more detailed qualitative and quantitative analysis.

The objective of the initial screening analysis is to identify potential common-cause vulnerabilities and to determine those that are insignificant contributors to system unavailability and to the overall risk, to eliminate the need to analyze them in detail. The analysis can stop at this level if a conservative assessment is acceptable and meets the objectives of the study. Otherwise the component groups that survive the screening process should be analyzed in more detail, according to the Detailed Analysis phase.

3.2.3 Detailed Analysis

Proper treatment of CCFs requires identifying those components that are susceptible to common causes and accounting for their impact on the system reliability. The oldest, and one of the simplest detailed methods for modeling the impact of CCFs, is the Beta-Factor model [3-4].

To illustrate the way the beta-factor model treats CCFs, consider a simple redundancy of two identical components. Each component may be divided into an "independently failing" component and one that is affected by CCFs only. The beta-factor model further assumes that:

Total component failure rate (λ_T) = *Independent failure rate*(λ_I) + *Common cause failure rate*(λ_C)

A factor, β, is then defined as:

$$\beta = \frac{\lambda_C}{\lambda_T} \tag{3-10}$$

$\lambda_C = \beta\lambda_T$ (common cause failure rate)

$\lambda_I = (1-\beta)\lambda_T$ (independent failure rate)

Failure probability of the two-unit parallel system is then calculated as

$$Q_S = (\lambda_I t)^2 + (\lambda_C t) = [(1-\beta)\lambda_T]^2 + \beta\lambda_T t \tag{3-11}$$

where λt is an approximation for the exponential failure probability model.

A point estimate for beta is given by

$$\beta = \frac{2n_2}{n_1+2n_2} \tag{3-12}$$

where:

n_1 = number of independent failure events

n_2 = number of common cause failure events.

Failure events are then used to obtain values of n_1 and n_2 for the specific component of interest. The resulting beta factor value, together with the total failure rate, $\lambda_{T,}$ of the identical redundant components, is then used to calculate the reliability of the function in the presence of CCF events.

For the two-unit parallel system discussed above, consider a failure history where 100 independent, single failures have occurred, and three common-cause events have occurred where both units failed at the same time. The resulting beta would be:

$$\beta = \frac{2(3)}{100+2(3)} = \frac{6}{106} = 0.057 \tag{3-13}$$

If one is using the beta-factor model as a global common-cause term for a system with x parallel units, the point estimate for beta is given by:

$$\beta = \frac{2n_2+3n_3+\cdots xn_x}{n_1+2n_2+3n_3+\cdots xn_x} \tag{3-14}$$

The Beta-Factor model is often useful for simple, dually redundant systems or systems where the global common-cause term of all components failing is driving the risk. For systems where the exact combinations of failures are important, a more comprehensive detailed method, such as the alpha-factor model, is needed. An example would be where a MODU has six thrusters for positioning, and certain combinations (e.g., losing all aft or forward thrusters) can lead to failure coupled with the necessary environmental conditions. The alpha-factor model is presented in Appendix H.

Earlier in this discussion, an example of CCF was mentioned that involved failures of multiple bolts due to issues with their coating. The above models apply most straightforwardly to demands on multiple components that occur more or less simultaneously, and the common cause induces the affected components to fail more or less simultaneously. The simple models given above are not necessarily adequate for events of the multiple-bolt type; more sophisticated models are needed.

3.3 Human Reliability Analysis

Human Reliability Analysis (HRA) is the systematic identification, modeling, and probabilistic quantification of human error. HRA can be qualitative, entailing only identification and modeling of the error; or quantitative, by additionally estimating a Human Error Probability (HEP) for a given task. Human error becomes important when it has consequences such as when it contributes to the failure of a hardware system or when it leads to injury or death. Human error can thereby have significant safety or environmental consequences. HRA seeks to determine the human component of risk and, when so employed, to serve as the basis for preventing and mitigating human errors.

> *Human Reliability Analysis (HRA): a structured approach used to identify potential human failure events and to systematically estimate the probability of those events using data, models, or expert judgment. (ASME RA-Sb-2013)*

HRA was originally developed to support PRA for nuclear weapons assembly work in the 1960s, but was subsequently adapted for use in risk assessment of nuclear power facilities when it was included in WASH-1400, the original PRA framework for nuclear power [3-5]. Following the Three Mile Island meltdown, HRA was formally introduced into the nuclear regulatory framework with the appearance of the Technique for Human Error Rate Predication (THERP) [3-6]. HRA's application in nuclear power dominated much of the development of new methods by the U.S. NRC and the Electric Power Research Institute (EPRI) domestically, with similar efforts internationally. Gradually, HRA methods have been adopted in other safety critical industries such as aerospace, rail and transportation, and oil and gas.

The purpose of this section is to provide a general process on how to perform HRA as part of developing a risk model of offshore drilling facilities. In this context, HRA is the assessment of the reliability and risk impact of the interactions of humans on a system or a function. For situations that involve a large number of Human-System Interactions (HSIs), HRA becomes an important element of PRA to ensure a realistic assessment of the risk. HRA has its own discussion in this guide because it requires special treatment; quantification of individual human events is done in a manner quite different from quantification of hardware basic events. Moreover, dependence between human error basic events is potentially very important and, from a technical point of view, is a topic in itself.

In some industries, HRA has been used for generations. By some counts [3-7], over 60 methods have been developed for performing HRA in those industries. Key attributes of these methods will be reviewed here, and general guidelines will be offered for application by analysts engaged in developing risk models for offshore facilities. But offshore facilities differ in important ways from the facilities for which many of these earlier methods were developed. In particular, operations at those other facility types are much more proceduralized, and their HRA methods make essential use of that fact. Modeling of human performance in scenarios occurring at offshore facilities is necessarily different. This will be discussed below.

Examples of oil industry HSIs include drilling control centers at platforms, onshore operations control centers remotely linked to drilling platforms, and mechanical/electrical personnel during installation, test, operation, and maintenance of equipment. Each of these HSIs involves a technological system—from a digital console in an office environment to mechanical equipment on the platform—and a human user of the system. Just as there is opportunity for hardware failure, there is the opportunity that the human user of the system will commit an activity that causes a negative process outcome. Human reliability analysts, with support from systems analysts, model and quantify the impacts from these HSIs, which are then incorporated as human basic events in the PRA logic models. Although the term "human-system interaction" is generally used, the terms "human interaction," "human action," "human error," and "human failure" have been used in the HRA literature and will also be used in this discussion, particularly when it comes to the quantification of the impacts of those HSIs.

Because there are so many HRA methods, it is beyond the practical scope of this guide to provide a tutorial on all methods. Instead, a general framework is presented in this section, and a method-specific example is provided in Appendix I.

3.3.1 Human Reliability Analysis Process

Figure 3-6 depicts the HRA process as commonly employed in PRA, as adapted from the IEEE-1082 HRA standard [3-8]. The terminology in the figure has been slightly modified to reflect offshore platform applications instead of the nuclear facilities of the original standard. While this process originally reflects HRA in support of nuclear PRA, it easily generalizes to other industries such as oil and gas. Below are explanations for each step of this process.

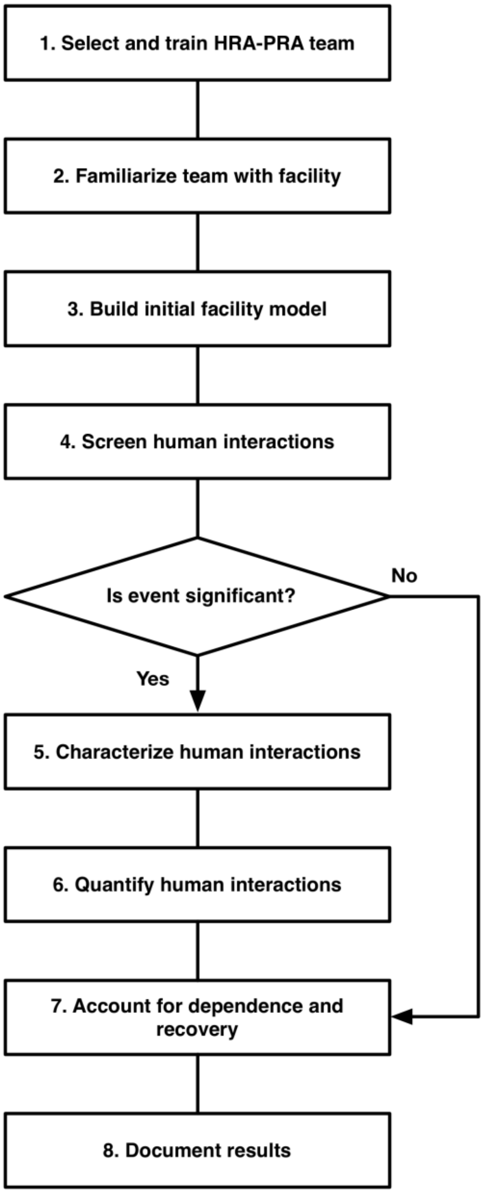

1. Select and train HRA-PRA team
2. Familiarize team with facility
3. Build initial facility model
4. Screen human interactions
Is event significant?
No
Yes
5. Characterize human interactions
6. Quantify human interactions
7. Account for dependence and recovery
8. Document results

Figure 3-6. Standard process for HRA (adapted from IEEE-1082).

1. *Select and train HRA-PRA team.* The analysis team may include PRA, operations, human factors, HRA, engineering, and other expertise. The ideal includes as many of these disciplines as possible, but practically it may only be possible to get one or two areas of expertise. Understanding PRA and HRA and understanding the operational process are the most important skills for the analysis. This step emphasizes the joint teaming of risk analysts in PRA with human reliability analysts. Human factors engineering, which is often used in the design of new systems, may reflect different expertise than HRA. HRA expertise is often centered on operations, which may prove more important from a risk standpoint than is optimizing the design of the human-machine interface.
2. *Familiarize team with facility.* The team should have general knowledge of the HSI and the facility, as well as any details that may be unique to this facility compared to similar facilities. Familiarization may involve briefings with facility experts and walkdowns of the facility with its users. This familiarization process should be done with an eye toward understanding the potential of the human

to impact the safe operations of the facility. The team determines what, how, and why there is potential for human errors, as well as ways in which recovery is possible.

3. *Build initial facility model.* If a PRA model is already in place, this stage validates that model in terms of human interactions. If a new PRA model is being created, this stage provides the necessary input in terms of the human failure event (HFE). An HFE is defined as the basic event involving the human. More practically, the HFE encompasses the human interactions that contribute to a component or system failure.

> *Human Failure Event (HFE): a basic event that represents a failure or unavailability of a component, system, or function that is caused by human inaction, or an inappropriate action. (ASME RA-Sb-2013) [3-9]*

Oil and gas production spans multiple facilities and locations including offshore oil platforms, onshore oil drilling stations, shale gas extraction facilities, and refinery facilities. Due in large part to the remoteness of the facilities and locations, a number of diverse hazards can lead to personnel injury and fatalities, equipment damage, and environmental harm. A representative list of hazards for offshore facilities adapted from the Petroleum Safety Authority of Norway [3-10 and 3-11] can be found in Table 3-5. These hazards characterize the types of activities where the potential for HFEs should be considered and included in the PRA model.

Table 3-5. Representative hazards at offshore oil and gas facilities.

- Non-ignited hydrocarbon leak
- Ignited hydrocarbon leak
- Well incident/ loss of well control
- Fire/explosion in other areas (non-hydrocarbon)
- Ship on collision course
- Drifting objects
- Collision with field related vessel
- Structural damage/stability/mooring/positioning failure
- Leakage from subsea systems/pipelines/risers/flowlines/loading buoy/loading hose
- Damage on subsea systems/pipelines/diving gear caused by fishery equipment
- Evacuation (precautionary/emergency evacuation)
- Helicopter accident
- Man overboard
- Serious injury
- Serious illness/epidemic
- Blackout
- Non-operational control room
- Diving accident
- Release of hydrocarbons
- Loss of control of radioactive source (e.g., radioactive tracer)
- Falling objects.

4. *Screen human interactions.* Once the PRA model includes HFEs, the HFEs should be screened for risk significance. Only those HFEs that have an impact on the overall system risk will receive a

detailed HRA in Steps 5 and 6. If the HFEs do not contribute significantly to overall risk, then they are simply modeled for dependence and recovery in Step 7.

5. *Characterize human interactions.* This step, which may be thought of as qualitative analysis, entails determining the factors that may influence the outcome of the human interaction. Many HRA methods classify the **task type** of human activity that is being performed. For example, detecting an alarm will have a different set of error mechanisms than physically positioning a drill. Cognitive tasks fail in different ways than behavioral action tasks. Beyond the characteristics of particular task types, **Performance Shaping Factors** (PSFs) are used by many HRA methods to identify what may increase or decrease the likelihood of human error for the HFE. Both task types and PSFs help define the context of the human interactions, which defines what types of human errors are possible.

   > *Performance Shaping Factor (PSF): a factor that influences human error probabilities as considered in a PRA's human reliability analysis and includes such items as level of training, quality/availability of procedural guidance, time available to perform an action, etc. (ASME RA-Sb-2013) [3-9]*

6. *Quantify human interactions.* HRA methods are used to estimate the **Human Error Probability** (HEP) and accompanying uncertainty. The HEP and uncertainty distribution may then be inserted into the overall PRA to determine the contribution of the human to the overall risk. A high HEP in some scenarios may serve as a cue to improve the system (e.g., introduce redundant safety systems) or the process (e.g., improve accident training). A high HEP does not necessarily mean a hardware failure, as the HFE is typically only one part of the failure opportunity for a system. There are a number of ways to quantify the HEP, which are overviewed in Appendix I.

7. *Account for dependence and recovery.* **Dependence** is the relationship between two HFEs. In most HRA methods, it is assumed that error begets error—the first instance of human error increases the subsequent chances of error on the next human interaction. Therefore, HRA methods will apply a correction factor to increase the HEP when there is dependence. In this method, **Recovery** is the opportunity to rebound from a human error such as restoring function after a system shutdown. Recovery is usually treated in the modeling of the facility such as a post-error success path in the event tree. In general, not crediting recovery will lead to very high estimates of risk. Where this is true, it is important to be realistic about credit taken; excessive credit will cause the model to bury risk contributors that need to be more widely appreciated.

8. *Document results.* Each HRA method has a specific format for documenting the qualitative and quantitative findings. This final step simply serves as a reminder that all assumptions should be clearly documented to ensure traceability and replicability.

3.4 Expert Elicitation

In research work, if something is currently impractical, one may declare it to be out of scope, and focus effort in areas where progress can be made. However, in performing a risk analysis intended to support an actual decision of some importance, one cannot simply abstain from analyzing certain topics, if the results of such analysis could affect the decisions being supported. For example, if we want to make a decision about siting a facility, we cannot simply choose to neglect things like earthquake risk, just because they are difficult to analyze. We may offer a bounding argument (for example, if we design for an earthquake of a specified magnitude, and can argue convincingly that earthquakes greater than or equal to that magnitude are extremely unlikely, then we may feel sure that earthquake risk is not significant compared to other risks); but we cannot simply declare earthquake risk to be "out of scope," and neglect it. If we abstain from analyzing it, without furnishing some bounding argument to characterize the range of possibilities, then we are furnishing the decision-maker with a result of the form "Here are the results that we obtained, but there are risk contributors that we haven't analyzed that could completely change the decision that one would make, based on results obtained so far. Sorry."

"Expert elicitation" suggests itself when the development of the risk model leads to an area where not enough information is available for immediate purposes, and more is not attainable because of timing constraints or resource constraints. The point is not that the current question is unanswerable in principle, but rather that the only practical thing to do in the immediate term seems[g] to be to make some use of experts in lieu of either a fuller analysis, or perhaps a completely impractical experimental undertaking.

Even if expert elicitation seems to be the only practical thing to do, controversy may result from the application. In order to understand why, it is worth recalling the context in which some risk studies are undertaken. First, as stressed in the introduction to this guide, the need to perform decision analysis itself is driven by the magnitude of the stakes involved (financial, human safety, environmental protection, etc.). Moreover, in the context of *regulatory* decision-making, there is the need to convince not only oneself, but also others, of the essential validity of the risk analysis that informs the decisions.

As with HRA, many methods are available for performing expert elicitation. Appendix G provides a survey of some of those methods that may be useful in the offshore oil and gas industry.

3.5 Quantifying the Scenario Set

3.5.1 "Point" Estimates

Reference is frequently made to point estimates of important quantities such as top-event probability. The term "point estimate" refers to the use of specific numbers for the inputs to the calculation, without immediate regard to uncertainty or variability. (Propagation of uncertainty is discussed below.) Quantification of point estimates is discussed here not because point estimates should be used uncritically in decision-making, but rather because the point estimate is a first step toward a more-complete model quantification, and as such, plays a role even in scenario generation (a point discussed further in Section 4.2) by helping determine the truncation cutoff to be used and allowing a sanity check of the results.

Given the minimal cut sets for a particular top event, presented in sum-of-products form, we can obtain a point estimate of top-event probability (or frequency) by:

- Quantifying each cut set:
  - Multiplying the basic event point estimates of probability or frequency, as appropriate,

    or
  - If the events in the cut set are related in some way, performing a side calculation to quantify the probability of the conjunction,

    and
  - Summing over cut set results to obtain the estimate for the top event.

This result, called the "rare-event approximation," is (in some respects) easy to calculate, and provides a rigorous upper bound on top-event probability: not that the point estimate is guaranteed to be an upper bound on top-event probability, but rather that the exact calculation of the "point estimate" would be less than or equal to this estimate. This follows because, for any two events X and Y (which could be basic events, cut sets, or complex functional failures),

$$P(X+Y) = P(X) + P(Y) - P(X*Y). \qquad (3\text{-}15)$$

So:

[g] In some of the commentary excerpted in Appendix G, an NRC-sponsored analysis was taken to task for resorting to expert opinion when alternatives were arguably available.

$$P(X) + P(Y) \geq P(X + Y). \quad (3\text{-}16)$$

The term P(X*Y) is small in many applications (hence the “rare event” nomenclature), and in those cases, neglect of it is reasonable; moreover, it is often straightforward to check on the magnitude of its effect. In other words, summing the cut set probabilities overestimates the top event, but is frequently reasonable.

A somewhat more involved calculation is the “min cut upper bound,” obtained as:

P(TOP) ~ 1- sum product (1-p(xi)). (3-17)

3.5.2 Propagating Uncertainty through the Scenario Set

Given a way of calculating a point estimate for any setting of the basic event model parameters, we can propagate parameter uncertainty through the model in the same way that we can propagate parameter uncertainty through any model to obtain an uncertainty distribution on its output: we can sample from the joint distribution of the inputs, compute the result of that sample, iterate until the result is deemed to have converged to the point where key metrics can be evaluated (mean, median, mode, key percentiles). In addition to convergence, appropriate correlation of variables and avoidance of distributions with a tail with values greater than 1.0 used for probabilities is important as discussed below.

3.5.2.1 Correlating Variables. The above statement referred to sampling from the joint distribution of all of the variables. If some of the variables are correlated epistemically, it is necessary to reflect this in the calculation. Consider the example of two essentially identical valves in series that are required to close under a certain challenge. They are of common manufacture and are assumed to see the same operating conditions, including test and maintenance practices. Arguably they should have the same failure probability. Since they are in series, and are required to close, failure of this function entails failure of both valves; so the top-event probability will contain a contribution that is proportional to p(X*X), where “X*X” means “failure of both identical valves.” Treating this as if it were equal to p(X)*p(X) underestimates the contribution for possibly several reasons. Temporarily setting aside the issue of CCF, there is an epistemic issue: in general, for any quantity Z,

$$< Z^2 > \geq < Z >^2, \quad (3\text{-}18)$$

so failure to acknowledge this epistemic coupling tends by itself to underestimate the result.

Suppose that the valve failure probabilities in the above example are lognormally distributed, each with an identical mean of 1E-3 and error factor of 5.0. Table 3.6 provides means and error factors for both the case where they are treated independently, and the case where they are correlated.

Table 3-6. Result of correlated and uncorrelated events under an AND gate.

Statistic	Correlated	Uncorrelated
Mean	2.7 E-6	1.0E-6
Error Factor	25.1	9.2

The uncorrelated product results in a much smaller mean and error factor.

3.5.2.2 Truncated Distributions. Consider an event X that is lognormally distributed with a mean of 0.5 and an error factor of 5.0. Clearly this is a large mean and there is a high probability of sampling values greater than one. This is a problem when the samples represent probabilities that are constrained to be between zero and one. In a situation like this, the sampling software will typically either remove samples greater than one or truncate the distribution, effectively cutting the tail off the distribution and renormalizing it (see Figure 3-7). The two methods of truncation appear to be different but are in fact mathematically equivalent.

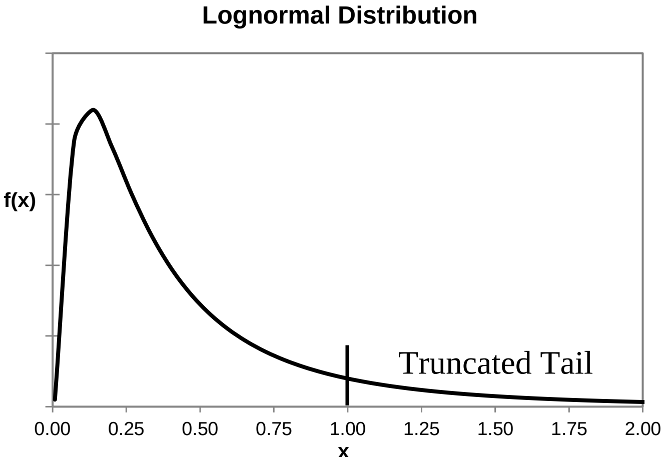


Figure 3-7. Truncated lognormal distribution.

Sampling from *X* and rejecting the values greater than one yields a mean of 0.33, which is noticeably lower than the input mean of 0.50. This is because the larger values are rejected.

Truncating the distribution also yields a mean of 0.33. This is not surprising since this method is mathematically equivalent to the rejection method.

To sample from a right-truncated lognormal distribution at location $b > 0$ [3-11]:

$$x = Loginv\left(Lognorm\left(b, Mean, EF\right) \cdot U, Mean, EF\right) \tag{3-18}$$

where *Loginv* is the inverse of the lognormal distribution, *Lognorm* is the cumulative lognormal distribution, and *U* is a random number between zero and one. In the example above the truncation point is 1.0, the mean is 0.50, and the error factor is 5.0.

Right-truncating a distribution will result in a lower mean than the non-truncated distribution. However, in general it is not good practice to model probabilities with a distribution that will require truncation.

3.6 References

3-1 NUREG-1816, *Independent Verification of the Mitigating Systems Performance Index (MSPI) Results for the Pilot Plants (Final Report)*, U.S. NRC, 2005.

3-2 H. Hamzehee, R. W. Youngblood, *et al*., “Risk-Based Performance Indicators: Results of Phase 1 Development,” NUREG-1753 (U.S. NRC, 2002).

3-3 A. Mosleh *et al*., “Procedures for Treating Common Cause Failures in Safety and Reliability Studies,” U.S. Nuclear Regulatory Commission and Electric Power Research Institute, NUREG/CR-4780, and EPRI NP-5613. Volumes 1 and 2, 1988.

3-4 K. N. Fleming, *A reliability model for common mode failure in redundant safety systems,* General Atomic Report GA-A13284, April 1975.

3-5 Forester, J. A., S. E. Cooper, A. M. Kolaczkowski, D. C. Bley, J. Wreathall, and E. Lois, *An Overview of the Evolution of Human Reliability Analysis in the Context of Probabilistic Risk Assessment, SAND2008-5085*, Sandia National Laboratories, 2009.

3-6 A. D. Swain and H. E. Guttmann, *Handbook of human reliability analysis with emphasis on nuclear power plant applications. Final report. NUREG/CR-1278*, U.S. Nuclear Regulatory Commission, 1983.

3-7 B. J. Bell and J. Holroyd, *Review of human reliability assessment methods,* RR679, Health and Safety Executive, 2009.

3-8 IEEE, *Draft Guide for Incorporating Human Reliability Analysis into Probabilistic Risk Assessments for Nuclear Power Generating Stations and Other Nuclear Facilities*, IEEE-1082/D8, Institute for Electrical and Electronics Engineers, 2017.

3-9 ASME, “Probabilistic Risk Assessment Standard for Advanced Non-LWR Nuclear Power Plants,” RA-Sb-2013, American Nuclear Society, 2013.

3-10 Petroleum Safety Authority Norway, 2013a, *Risikonivå i petroleumsvirksomheten, Hovedrapport; Utviklingstrekk 2012, norsksokkel. Rev. 2, 25.4.2013* (Risk level in the Petroleum activity, Main Report, Development trends 2012, Norwegian Continental Shelf, in Norwegian). Stavanger: Petroleum Safety Authority Norway, 2013.

3-11 Petroleum Safety Authority Norway, 2013b, *Risikonivå i petroleumsvirksomheten, Hovedrapport; Utviklingstrekk 2012, landanlegg. Rev. 2, 25.4.2013* (Risk level in the Petroleum activity, Main Report, Development trends 2012, Onshore Facilities, in Norwegian). Stavanger: Petroleum Safety Authority Norway, 2013.

3-12 A. Law and W. D. Kelton, *Simulation Modeling and Analysis*, McGraw-Hill Inc., 1991.

4. RESULTS: PRESENTATION AND INTERPRETATION

4.1 Risk Analysis Support to a Notional Safety Case

Figure 4-1 shows a notional “claims tree:” a hierarchy of the claims that might be made in a safety case presented to a regulator. The regulation of facilities by the Bureau of Safety and Environmental Enforcement is beyond the scope of this document, but it is nevertheless useful to organize the discussion of certain topics around a figure like this.

In many venues of application of risk models, the models are developed by parties associated with the (proposed) facilities, even though the model results are to be applied in assurance cases put before regulatory decision-makers who are accountable to different parties for different considerations (e.g., regulators may be more accountable for public safety than for facility economics).

The technical content of the present guidance is trying to be useful both to applicants and to regulators. The facility operators (and investors) need to ensure that the facilities are (or will be) safe; the regulator needs assurance that the facility is (or will be) safe. The needs of the two are distinct. The claims tree is aimed specifically at promoting a successful dialogue between applicants and regulators.

The premise of the figure is that a finding has to be made regarding the safety of a specific facility, and this finding needs to be based in part on an analysis. The analysis needs to address certain figures of merit (such as risk metrics) and, potentially, to show that certain other requirements are met (such as requirements on barrier availability and performance). The four major sections of the figure are:

1. Design characterization
2. Analysis of risk (and possibly other metrics), conditional on a particular baseline allocation of performance (e.g., reliability)
   - The analysis satisfies certain process requirements
   - The analysis provides sensitivity and uncertainty information
   - A process exists for identifying, and dealing with, unresolved safety questions.
3. [Optional] A process has been carried out to substantiate a claim that the facility is as safe as reasonably practicable.

   Note: *This figure can be specialized to refer to “Best Available and Safest Technology” (BAST) rather than “Risk Is as Low As Reasonably Practicable.” In either case, process-based arguments to support the respective conclusions are called for.*
4. The performance allocation credited in the analysis is, in fact, feasible. The items considered critical, the associated levels of performance, and the activities needed to make the risk analysis “come true” have been identified and committed to. This includes making reliability allocations come true, barrier availabilities come true, etc., and includes a commitment to analysis of operating experience, looking for deficiencies in the model. Ordinarily, the model used for Item 2 above is also a starting point for this item.

Portions of Items 2 and 4 of the above list are within the scope of the present document. Item 2 notionally covers the safety analysis, and the results that need to be presented, including sensitivity and uncertainty information. Item 4 captures the claims that tie the analysis results to reality, including commitments to scrutinize operating experience as part of an effort aimed at model improvement. Certain aspects of Item 4 are beyond the scope of this document, but an emphasis of this document is that the numbers on which PRA results are based are engineering accomplishments, not constants of nature; submittal of a PRA to a regulator needs to be tied to commitments to make those numbers come true, including identification of model inadequacies revealed by operating experience. This is part of Item 4.

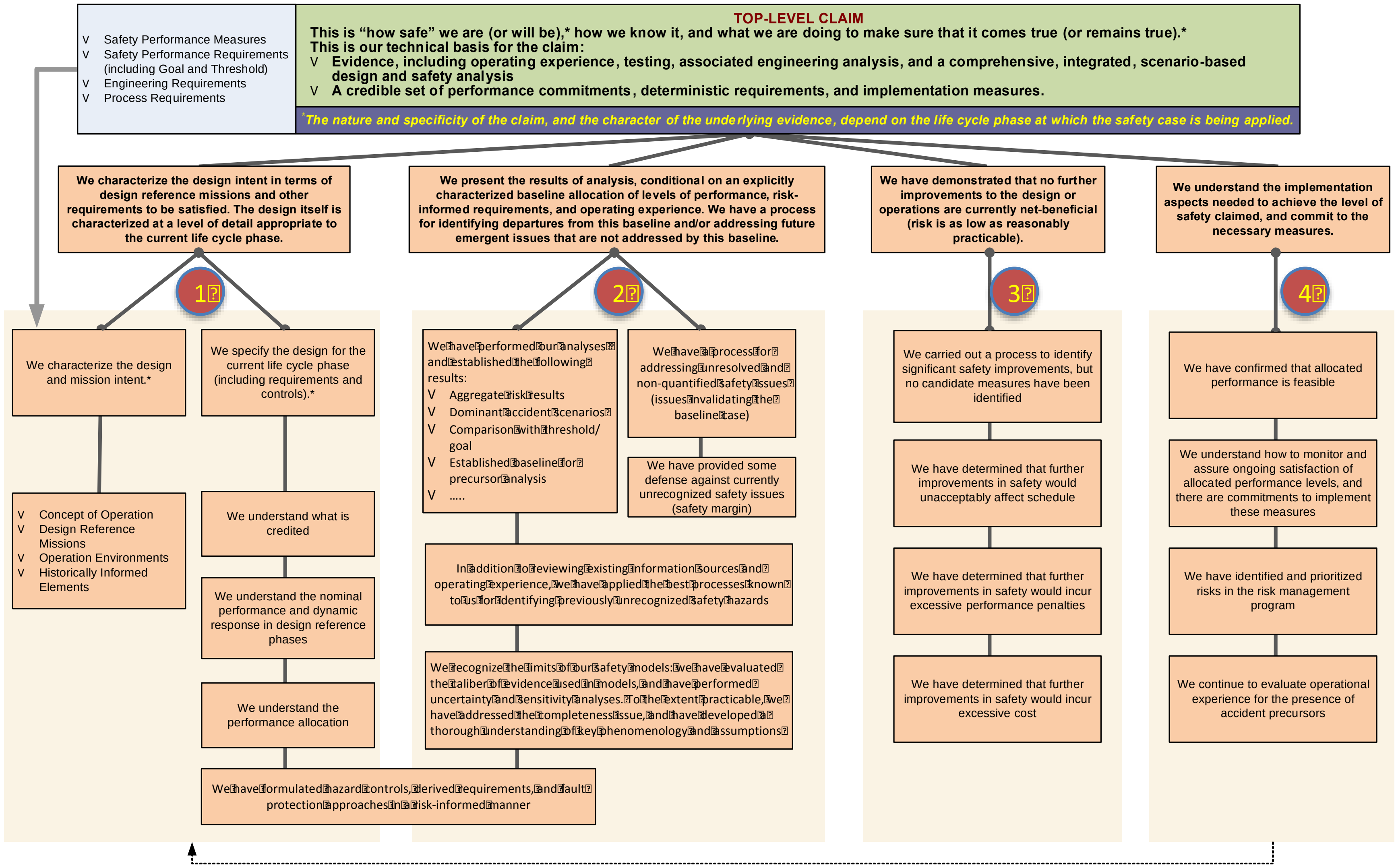

V Safety Performance Measures
V Safety Performance Requirements (including Goal and Threshold)
V Engineering Requirements
V Process Requirements
TOP-LEVEL CLAIM
This is "how safe" we are (or will be),* how we know it, and what we are doing to make sure that it comes true (or remains true).*
This is our technical basis for the claim:
V Evidence, including operating experience, testing, associated engineering analysis, and a comprehensive, integrated, scenario-based design and safety analysis
V A credible set of performance commitments, deterministic requirements, and implementation measures.
*The nature and specificity of the claim, and the character of the underlying evidence, depend on the life cycle phase at which the safety case is being applied.
We characterize the design intent in terms of design reference missions and other requirements to be satisfied. The design itself is characterized at a level of detail appropriate to the current life cycle phase.
We present the results of analysis, conditional on an explicitly characterized baseline allocation of levels of performance, risk-informed requirements, and operating experience. We have a process for identifying departures from this baseline and/or addressing future emergent issues that are not addressed by this baseline.
We have demonstrated that no further improvements to the design or operations are currently net-beneficial (risk is as low as reasonably practicable).
We understand the implementation aspects needed to achieve the level of safety claimed, and commit to the necessary measures.
1
2
3
4
We characterize the design and mission intent.*
We specify the design for the current life cycle phase (including requirements and controls).*
V Concept of Operation
V Design Reference Missions
V Operation Environments
V Historically Informed Elements
We understand what is credited
We understand the nominal performance and dynamic response in design reference phases
We understand the performance allocation
We have formulated hazard controls, derived requirements, and fault protection approaches in a risk-informed manner
We have performed our analyses and established the following results:
V Aggregate risk results
V Dominant accident scenarios
V Comparison with threshold/ goal
V Established baseline for precursor analysis
V .....
We have a process for addressing unresolved and non-quantified safety issues (issues invalidating the baseline case)
We have provided some defense against currently unrecognized safety issues (safety margin)
In addition to reviewing existing information sources and operating experience, we have applied the best processes known to us for identifying previously unrecognized safety hazards
We recognize the limits of our safety models: we have evaluated the caliber of evidence used in models, and have performed uncertainty and sensitivity analyses. To the extent practicable, we have addressed the completeness issue, and have developed a thorough understanding of key phenomenology and assumptions
We carried out a process to identify significant safety improvements, but no candidate measures have been identified
We have determined that further improvements in safety would unacceptably affect schedule
We have determined that further improvements in safety would incur excessive performance penalties
We have determined that further improvements in safety would incur excessive cost
We have confirmed that allocated performance is feasible
We understand how to monitor and assure ongoing satisfaction of allocated performance levels, and there are commitments to implement these measures
We have identified and prioritized risks in the risk management program
We continue to evaluate operational experience for the presence of accident precursors

Figure 4-1. Claims tree.

4.2 Quantifying the Model

Quantification of a PRA model is a simple process; however, steps must be taken to ensure that the output is relatively complete and accurate. The term “relatively” is used here with “complete” because PRA models can have millions of scenarios, and often many are of such a low probability that they do not need to be considered. A PRA model can take a long time to run and produce a very large amount of data, so determining the right level for quantification is an important step to ensure the necessary results are obtained while still being manageable. This applies to both the classical event-tree/ fault-tree models and the discrete event models as discussed below.

4.2.1 Event-Tree/ Fault-Tree Model Quantification

Quantifying an event-tree/ fault-tree model typically involves choosing different options for performing the quantification. Three of the most important are discussed below. It should be noted that the model quantification and results discussed in this section is a simplified example for illustrating the results of PRA, and not an actual analysis of a real facility.

4.2.1.1 Truncation Cutoff. The truncation cutoff used in quantifying an event tree is used to stop the quantification of scenarios (i.e., minimal cut sets) below a user selected value. A large PRA model can have millions or tens of millions of minimal cut sets; evaluating all of them can take a long time to run the model, and result in an excessively large amount output that is hard to manipulate to display results. Typically, lower probability scenarios may be many orders of magnitude lower than the overall result; such scenarios are not significant risk contributors. Most PRA software implements a user-supplied truncation value, discarding cut sets whose probability is below that truncation value. In principle, this is an uncontrolled approximation, but is frequently the practical thing to do, provided that steps are undertaken to understand the effects of the truncation.

The process to determine what the best truncation cutoff is should start at a level the analyst expects would be consistent with the result (this is essentially a guess based on construction of the model), and then vary the truncation cutoff by reducing it an order of magnitude until the results at least converge to a value less with less than a 1% difference between two successive quantifications.

Using the example developed in Section 2 (Figure 2-13), it is possible to see the effect the truncation limit has on the results in Table 4-1 when the end states with releases are assessed.

Table 4-1. Effect of truncation limit on event-tree quantification.

Truncation Cutoff	Number of Cut Sets	Overall Likelihood	End States
1.0E-4	1	3.70E-4	LIMITEDRELEASE
1.0E_5	2	4.07E_4	LIMITEDRELEASE, CAPPINGSTACKCONTAIN
1.0E-6	4	4.17E-4	LIMITEDRELEASE, CAPPINGSTACKCONTAIN, RELIEFWELLSEAL
1.0E-7	8	4.19E-4	LIMITEDRELEASE, ROVCONTAIN, CAPPINGSTACKCONTAIN, RELIEFWELLSEAL
1.0E-8	28	4.199E-4	Same as 1.0E-7 truncation cutoff
1.0E-9	53	4.20e-4	Same as 1.0E-7 truncation cutoff
1.0E-10	120	4.20E-4	Same as 1.0E-7 truncation cutoff
1.0E-12	872	4.20E-4	Same as 1.0E-7 truncation cutoff
1.0E-15	10551	4.20E-4	Same as 1.0E-7 truncation cutoff

When reviewing the results in Table 4-1, two effects can be seen as the truncation cutoff is varied. First, when the truncation value has been reduced to 1.0E-8, the overall likelihood has converged to several decimal places, and the number of cut sets is 28 at that level. When the truncation cutoff is set to 1.0E-15, the model produces 10551 cut sets, and from the table, the majority of them are below 1.0E-12 so do not affect the overall result.

A second consideration should also be given to the end states found in the results. As seen in Table 4-1, the initial quantification of the model only resulted in one of the four end states being evaluated. If the goal of the analysis is based on the overall risk and major contributors, whether end states show up in the result may not make a difference if they do not significantly contribute, but if end states are to be evaluated separately for dominant contributors, the truncation cutoff should also be selected to get representative cut sets from each end state. This is particularly significant if importance measures are evaluated.

4.2.1.2 Solution Method. A second quantification option that can have a significant impact on results is the solution method that is used when results for different pivotal-event fault trees are combined in the event tree sequences. The success path on an event tree can be treated in different ways. Linked fault-tree software typically defaults to a solution that uses a "delete term" function. The delete term function removes invalid cut sets from sequential top events with common basic events. An example of an invalid cut set would arise if, in a particular sequence expression, failure of System A is combined with success of System B, and some of the "failure" cut sets for System A are inconsistent with success of System B. This condition cannot exist, so the cut set is deleted from that sequence expression.

Using "delete term" is reasonable, but it is an approximation: the success of System B is set to a probability of 1.0. Because PRA analyses usually are evaluating rare events, the top-event probabilities are small most of the time, and the approximation of a success path to 1.0 is acceptable. In some cases, a top event may have a relatively large probability (> 0.01), and in this case, choosing a solution method that accounts for the proper success path probability may be required for a sufficiently accurate calculation.

4.2.1.3 Uncertainty. Quantifying the model to obtain the uncertainty distribution is similar to the truncation cutoff issue in that enough iterations of the model must be performed to ensure the mean value has converged. PRA software generally runs quickly so the number of iterations needed to converge is usually not an issue.

In some cases, the mean will actually differ from the point estimate, because in general, basic event distributions are correlated. Correlation of basic events results if the same uncertainty distribution is used for a number of basic events. Each sampling in the uncertainty calculation for the basic events that are correlated uses the same value from the common distribution. This has a tendency to increase the mean value if an AND gate is used as the sampling from the extreme ends of the distribution compound and to stretch the distribution, resulting in a mean value higher than the point estimate.[h]

4.2.2 Discrete Event Simulation Model Quantification

For analyses using discrete event simulation, as discussed in Section 2.3.3, the main option for quantification that must be considered is the number of replications (i.e., the number of passes through the model). Determining a sufficient number of replications is an iterative process that should start by reviewing the paths through the model along with the inputs to estimate the expected number of replications that result in a particular end state. For example, if a review of the model inputs shows that the output should occur with a frequency of about 1E-4, then the number of replications needed to get a single hit on that path would be, on average, 1/1E-4, or 10,000. Because the mean will not have

[h] In general, $<x^2>$ is greater than or equal to $<x>^2$.

converged with a single hit (or a small number of hits), for this example, a reasonable starting point would be 100,000 replications, or 10 times the average needed to get a single hit. Ideally, a sufficient number of replications will be run to obtain a mean within acceptable convergence bounds. If an initial number of replications is not sufficient to establish convergence, then additional replications are required. At some point, time constraints might limit the number of replications, so that the desired convergence is not met. In these cases, the uncertainty due to the limited number of replications should be specified. This uncertainty is often presented as 90% confidence bounds about the mean.

Table 4-2 shows output from the model shown in Section 2.3.3 based on 100, 2,500, and 25,000 replications.

Table 4-2. Discrete model simulation results for different numbers of replications.

			Barrels Leaked			**Time to Effect (Hours)**		
Event	**Count**	**Probability**	**Min**	**Mean**	**Max**	**Min**	**Mean**	**Max**
Cap, Vertical	77	7.70E-01	82,534	257,276	599,490	49	225	470
Relief Well	12	1.20E-01	2,074,500	3,490,658	6,112,600	2,050	2,883	4,496
Cap, Non-Vertical	6	6.00E-02	237,400	614,642	906,370	164	453	564
ROV	4	4.00E-02	2,460	13,162	25,931	2	12	22
Cap, Well Head	1	1.00E-02	562,800	562,800	562,800	475	475	475
Replications	100							
			Barrels Leaked			**Time to Effect (Hours)**		
Event	**Count**	**Probability**	**Min**	**Mean**	**Max**	**Min**	**Mean**	**Max**
Cap, Vertical	1,962	7.85E-01	46,472	277,137	1,062,000	49	225	479
Relief Well	223	8.92E-02	1,683,500	3,668,681	8,168,500	2,015	2,960	6,182
Cap, Non-Vertical	214	8.56E-02	136,480	591,123	1,545,200	107	470	959
ROV	76	3.04E-02	1,886	22,706	104,660	2	17	55
Cap, Well Head	25	1.00E-02	562,800	1,055,688	1,967,600	475	811	1,221
Replications	2,500							
			Barrels Leaked			**Time to Effect (Hours)**		
Event	**Count**	**Probability**	**Min**	**Mean**	**Max**	**Min**	**Mean**	**Max**
Cap, Vertical	19,654	7.86E-01	38,148	279,500	1,062,000	48	227	480
Relief Well	2,246	8.98E-02	1,546,500	3,685,098	13,719,000	2,001	3,007	6,918
Cap, Non-Vertical	2,041	8.16E-02	87,312	577,038	2,037,600	97	467	959
ROV	839	3.36E-02	1,826	26,186	126,050	2	21	62
Cap, Well Head	220	8.80E-03	316,640	1,039,565	2,101,300	321	838	1,313
Replications	25,000							

4.3 Reviewing the Results

The PRA model is typically developed to answer a specific question or questions regarding the risk of a facility or operation, and a range of results are produced and may be reviewed at a variety of different levels (e.g., from system reliabilities to magnitudes of oil released to the environment). Common results evaluated as outputs from a PRA include:

- Total likelihood[i] of various end states
- The relative ranking of each scenario to the total end state likelihood or total risk
- Estimates of scenario consequences (environmental release, damage to property, number of injuries or fatalities, dollar loss, etc.)
- Importance measures
- Display of uncertainties associated with various estimates
- System level reliabilities.

Each of these types of results are discussed in more detail in the following sections with examples based on the environmental release model developed in Section 2.

4.3.1 Overall End State Likelihood and Relative Risk Ranking

The overall objective of performing a PRA is typically to evaluate a design or operation with respect to the risk involved. The purpose could be to ensure the design or process is acceptably safe relative to safety goals or requirements, or to understand if there are any driving vulnerabilities that can be addressed further. The first metrics assessed are usually the overall risk and a relative risk ranking of scenarios. Sample output from the simplified model developed in Section 2 is shown in Tables 4-3 and 4-4.

Table 4-3. End state frequencies for hydrocarbon release events.

Name	Point Estimate	Cut Set Count
Total	4.20E-04	10551
LIMITEDRELEASE	3.70E-04	571
CAPPINGSTACKCONTAIN	4.39E-05	5055
RELIEFWELLSEAL	4.39E-06	2152
ROVCONTAIN	1.64E-06	2773

[i] In this section, the term "likelihood" is used to refer either to probability, frequency, or both.

Table 4-4. Sample PRA model output.

#	Prob/ Freq	Total %	Cut Set	Description
Total	4.19E-04	100	Displaying 10 Cut Sets. (10551 Original)	
1	3.70E-04	88.23		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
		End State	LIMITEDRELEASE	
2	3.70E-05	8.82		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	1.00E-01		BOP-CYL-JAM-BSRDP	BSRs fail to close and seal when drill string is in the hole
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
		End State	CAPPINGSTACKCONTAIN	
3	5.92E-06	1.41		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	1.60E-01		BOP-HUM-ERR-HANGOFF	Driller fails to position drill pipe properly before activating BSR
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
	1.00E-01		DP_TOOLJOINT_PRESENT	Drill Pipe Tool Joint is Present
		End State	CAPPINGSTACKCONTAIN	
4	3.70E-06	0.88		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	1.00E-01		BOP-CYL-JAM-BSRDP	BSRs fail to close and seal when drill string is in the hole
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
	1.00E-01		CAP_STACK_FAILS	Capping Stack unsuccessful
		End State	RELIEFWELLSEAL	
5	9.00E-07	0.21		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
	2.43E-03		BOP-PRG-FLO-I02	Subsea manifold pressure regulator I02 fails low (Yellow)
		End State	ROVCONTAIN	

Table 4-4. (continued).

#	Prob/ Freq	Total %	Cut Set	Description
6	5.92E-07	0.14		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	1.60E-01		BOP-HUM-ERR-HANGOFF	Driller fails to position drill pipe properly before activating BSR
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
	1.00E-01		CAP_STACK_FAILS	Capping Stack unsuccessful
	1.00E-01		DP_TOOLJOINT_PRESENT	Drill Pipe Tool Joint is Present
		End State	RELIEFWELLSEAL	
7	2.22E-07	0.05		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	1.60E-01		BOP-HUM-ERR-HANGOFF	Driller fails to position drill pipe properly before activating BSR
	1.60E-01		BOP_HUM_ERR_IBOP1	Human error - failure to install IBOP
	8.66E-05		BOP_SCV_FTC_FLTVLV1	Float Valve Fails To Close
	1.00E-01		DP_TOOLJOINT_PRESENT	Drill Pipe Tool Joint is Present
		End State	CAPPINGSTACKCONTAIN	
8	1.81E-07	0.04		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	4.90E-04		BOP-HUM-ERR-EMERGDIS	Operator fails to initiate emergency disconnect successfully
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
		End State	ROVCONTAIN	
9	1.74E-07	0.04		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
	4.70E-04		BOP-SHV-LKE-A13	BSR Lock shuttle valve A13 catastrophic leakage
		End State	CAPPINGSTACKCONTAIN	
10	1.74E-07	0.04		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
	4.70E-04		BOP-SHV-LKE-A12	BSR Lock shuttle valve A12 catastrophic leakage
		End State	CAPPINGSTACKCONTAIN	

Table 4-3 shows an overall probability of approximately 4.2E-4/kick, or about 1 in 2400 kicks, for having an environmental release of hydrocarbons during drilling. By examining the different end states, it is shown that by far the highest probability event is a limited release. The long-duration releases represented by the other three end states have a frequency of approximately 5E-5. Of those three, the capping stack represents the highest frequency. Since the ROV is the first chance at intervention and containing the well, one could look at the results and ask why isn't the ROV successful more often?

Table 4-4 lists the Top 10 cut sets that can be individually reviewed for all release end states. For this simplified model, the results show several things that could be of interest. The top two cut sets account for over 95% of the risk, and would be a focus of improvement if the overall risk was considered to be too high (implying that improvement is needed). Both of the first two cut sets include failure to detect the kick or act before the lick reaches the BOP. Further inspection shows that the Cut Set 1 end state is LIMITEDRELEASE while Cut Set 2 is CAPPINGSTACKCONTAIN with potentially significantly higher consequences.

In practice, PRA results are often more "flat" than shown in the results from the simplified model used in this guide, partly because models are developed at a finer level of resolution, so that no single cut set contributes a significant percentage to the total. There may be hundreds or even thousands of cut sets that make up 95% or more of the high-level risk number. In this case, the results may be put in a spreadsheet and manipulated by grouping cut sets related to a particular end state, component, or system to develop insights.

4.3.2 Estimates of Consequences

While the discussion in Section 4.3.1 focused on the frequency of end states, also of interest is the magnitude of the adverse consequences assigned to the various end state(s). Classical PRA models using event trees and fault trees have end states that are subjectively defined (Limited Release, Large Release, etc.). By manipulating the cut sets in the previous section, the frequencies of these end states may be estimated. For many applications, this approach may be acceptable as it shows that frequency of the end state and any scenarios that are major contributors. This information allows actions to be identified that may reduce risk.

In some applications, the actual magnitude of end states may be needed or desired. For instance, there may be governmental requirements on the expected casualty rate from a particular facility or operation. In this case the model must estimate the number of deaths for individual scenarios to develop that result. A discrete event simulation model as discussed in Section 2.3.3 is a method that can be used to perform that analysis. The output from a discrete event simulation is the results related to each replication of the model and can be very large. Table 4-5 shows a sample of typical results.

Table 4-5. Sample results from a discrete event simulation model.

Replication	Event	Time	Barrels
1	CAP_V_LOC Value	223	199,660
2	Relief_LOC Value	3,089	2,474,100
3	CAP_V_LOC Value	260	211,600
4	CAP_NONV_LOC Value	519	906,370
5	CAP_V_LOC Value	183	313,090
6	CAP_V_LOC Value	176	157,030
7	CAP_NONV_LOC Value	367	619,310

The results are then manipulated to be useable by developing plots. Because the inputs are based on distributions (e.g., flow rates), the results must be binned into logical ranges to show results. Figure 4-2 shows the output from the example developed in Section 2.3.3 in terms of probability versus magnitude of release.

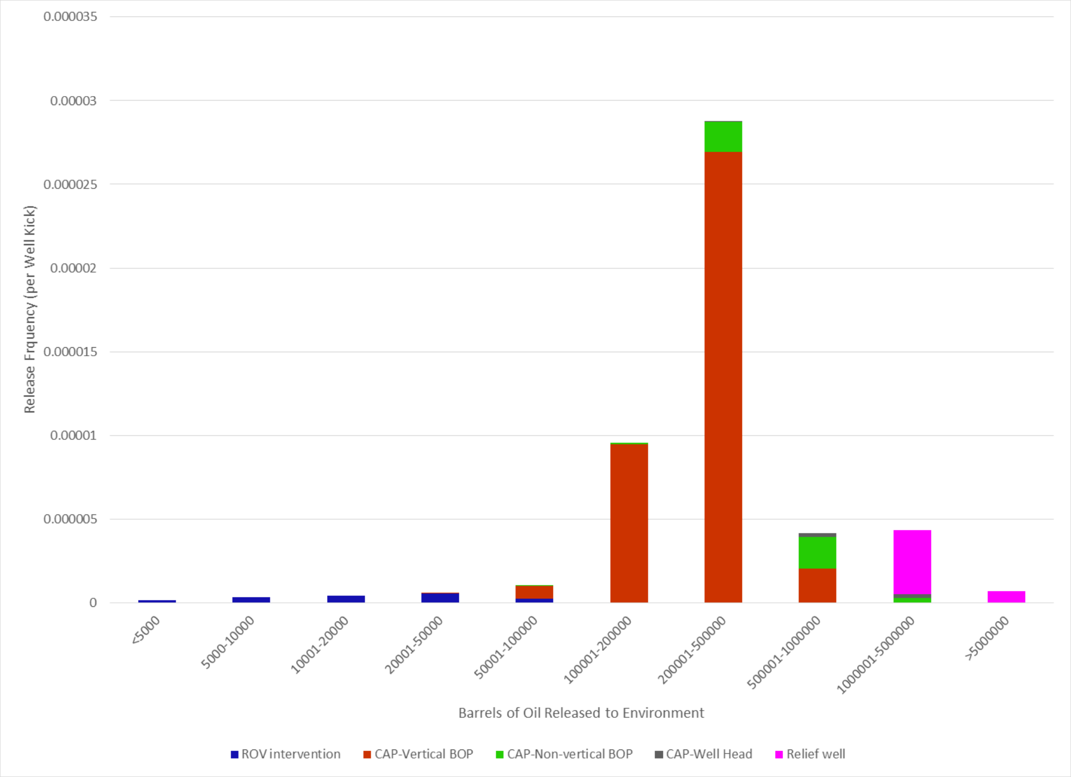


Figure 4-2. Example output from discrete event simulation.

The bins (e.g., 0–100, 100–1000) are chosen after reviewing the results to determine logical groupings.

A special type of graph called an F-N curve can be a valuable tool if the probability of exceeding a particular magnitude of consequence is of interest. This type of plot displays the magnitude of the end state on the x axis and the probability of exceeding it on the y axis.

Table 4-6 shows the data needed to construct an F-N curve. The magnitude is developed by the analyst subjectively based on reviewing the results and assigning the output to bins. Note that each bin has a "greater than" designation. The number of replications assigned to each bin is obtained from the results and then divided by the total number of replications, in this case 25,000. This gives the frequency of exceedance for each bin. The result is a graph as shown in Figure 4-3.

Table 4-6. Frequency of exceedance calculation.

Magnitude (Barrels released)	**Replications**	**Frequency of Exceedance**
>0	25000	4.99E-05
>5000	24929	4.98E-05
>10000	24770	4.95E-05
>20000	24558	4.91E-05
>50000	24252	4.84E-05
>100000	23758	4.75E-05
>200000	18979	3.79E-05
>500000	4579	9.15E-06
>10000000	2496	4.99E-06
>50000000	337	6.73E-07

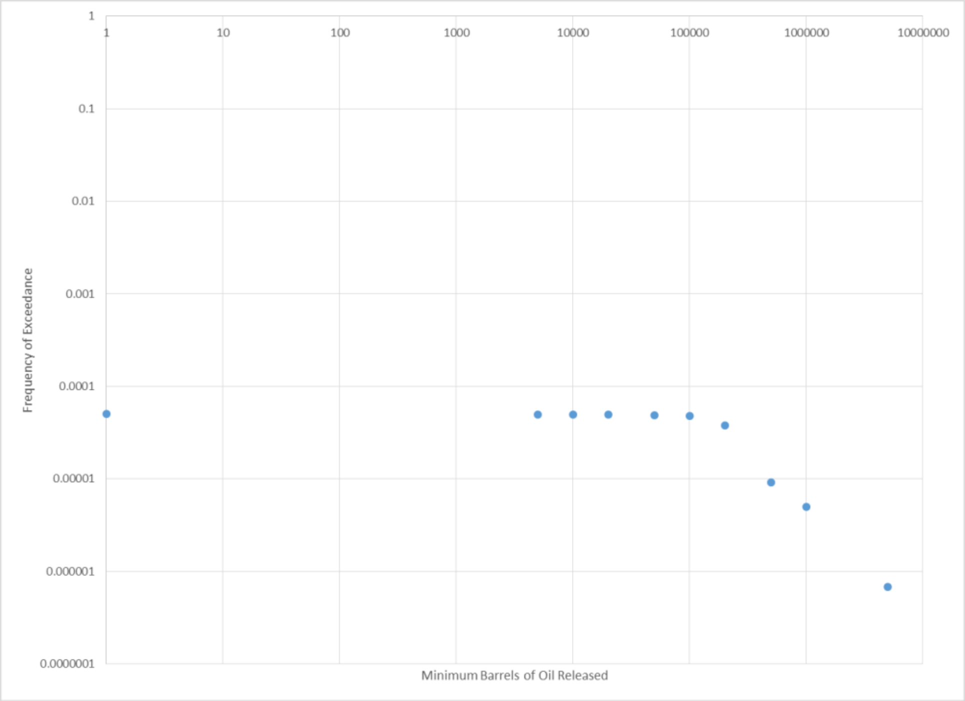


Figure 4-3. Example frequency of exceedance curve.

The frequency-of-exceedance curves are typically plotted on a log-log scale because the data can span orders of magnitude. Looking at the curve from right to left, flat frequency-of-exceedance curves indicate that the failures occurring between points are having a minimal effect on the overall magnitude of the consequence. When the curve has a high slope, and the frequency drops very significantly, this means that it is difficult or impossible (highly infrequent) to exceed the corresponding magnitude of release.

4.3.3 Importance Measures

Ranking of risk scenarios based on their frequencies as discussed in Section 4.3.1 provides limited insight regarding the contribution of individual events such as component failures to the total risk. Scenario ranking provides insights on importance of group of failures, not failure of individual events. An event (say, “component x failure”) can appear in the structure of many low frequency scenarios, yet it may be absent from the definitions of the dominant risk scenarios. If the contribution of low-frequency scenarios to the total risk is comparable to that of a few dominant risk scenarios, then scenario ranking will not capture the risk importance of component x. In order to address this issue and to provide perspective on importance of individual events or parameters of the PRA model, several quantitative importance measures are calculated.

Once the importance measures are calculated, the events or parameters of the risk model can be ranked according to the relative value of the importance measure. This provides some insight into what is, or could be, influencing actual risk. This insight can inform risk-informed decision-making (e.g., allocating resources), or point to the need for risk mitigation efforts, such as redesign of hardware components, the addition of redundancy, etc. However, this sort of insight should not be a sole basis for decision-making.

The quantitative importance measures typically found in PRA software include:

- Fussell-Vesely (F-V)
- Risk achievement worth (RAW)
- Risk reduction worth (RRW)
- Birnbaum.

Another measure, the “Differential” importance measure, is discussed in Appendix J; it reflects the fractional change of a risk metric due to a particular basic event, given a change in a basic event probability. This metric is not typically found in PRA software.

All of the above importance measures are formulated in failure space: they are focused on sets of minimal *cut sets* that involve a specific event or model parameter. An importance measure based on success space (i.e., *path sets*), Prevention Worth (PW), is a single-event measure that can provide insights that are different from those provided by the failure-space measures.

The three most commonly used importance measures (F-V, RAW, and RRW) are discussed below with examples. Detailed information on the derivation of the failure-space-based importance measures are included in Appendix J. Prevention Worth, based in success space, is explained in Appendix K, along with a more comprehensive way of looking at model results, “Prevention analysis.” Instead of looking at basic events one at a time, Prevention analysis answers the question “what *combinations* of basic events should I undertake to prevent, in order to reduce risk in the most cost-effective way?”

4.3.3.1 Fussell-Vesely Importance. The most frequently used importance measure is the F-V importance of basic events. The F-V importance of a given basic event is the fraction of overall risk contributed by the cut sets containing that basic event. This is similar to the scenario risk ranking in Section 4.3.1, but performed at a basic event level. A basic event may show up in many cut sets that are lower frequency than the top scenarios, but the summation of the lower frequency cut sets for that component may show that basic event to be a significant risk because it is included in many scenarios. Since most cut sets are made up of multiple basic events, and the cut set frequency is counted for each basic event F-V importance value, the F-V contributions summed over all basic events will normally be greater than 1.0.

Table 4-7 displays the Top 5 cut sets for hydrocarbon releases from the example model developed in Section 2. The basic events INIT-EV_DRILLING and BOP-HUM-ERR-KICKDET occur in all of the cut sets, and therefore have a F-V importance of 1.0. The Subsea manifold pressure regulator I02 that fails low (BOP-PRG-FLO-I02) is only found in Cut Set 5, which has a cut set value of 9.0E-7. The F-V importance for this event is calculated simply by dividing 9.0E-7 by the total risk, 4.18E-4. The result is 0.22%, which is the same as the cut set contribution, because this basic event is only included in that single cut set. The basic event BOP-CYL-JAM-BSRDP is found in Cut Sets 2 and 4 with cut set values of approximately 3.7E-5 and 3.7E-6, respectively. In this case, the cut set values are added (4.07E-5) and divided by the total risk for a F-V importance of 9.76%. Table 4-8 shows the F-V importance for each of the failure events in Table 4-7.

Table 4-7. Top 5 minimal cut sets example for importance measures, all releases.

#	Prob/Freq	Total %	Cut Set	Description
Total	4.18E-04	100	Displaying 5 Cut Sets. (10551 Original)	
1	3.70E-04	88.6		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
		End State	LIMITEDRELEASE	
2	3.70E-05	8.86		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	1.00E-01		BOP-CYL-JAM-BSRDP	BSRs fail to close and seal when drill string is in the hole
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
		End State	CAPPINGSTACKCONTAIN	
3	5.92E-06	1.42		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	1.60E-01		BOP-HUM-ERR-HANGOFF	Driller fails to position drill pipe properly before activating BSR
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
	1.00E-01		DP_TOOLJOINT_PRESENT	Drill Pipe Tool Joint is Present
		End State	CAPPINGSTACKCONTAIN	
4	3.70E-06	0.89		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	1.00E-01		BOP-CYL-JAM-BSRDP	BSRs fail to close and seal when drill string is in the hole
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
	1.00E-01		CAP_STACK_FAILS	Capping Stack unsuccessful
		End State	RELIEFWELLSEAL	

Table 4-7. (continued).

#	Prob/Freq	Total %	Cut Set	Description
5	9.00E-07	0.22		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
	2.43E-03		BOP-PRG-FLO-I02	Subsea manifold pressure regulator I02 fails low (Yellow)
		End State	ROVCONTAIN	

Table 4-8. F-V importance calculation example.

Basic Event	Description	Cut sets with Basic Event	Total Cut set Value	F-V Importance
INIT-EV_DRILLING	Well Kick While Drilling	1, 2, 3, 4, 5	4.18E-04	1.00E+00
BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action	1, 2, 3, 4, 5	4.18E-04	1.00E+00
BOP-CYL-JAM-BSRDP	BSRs fail to close and seal when drill string is in the hole	2, 4	4.07E-05	9.67E-02
BOP-HUM-ERR-HANGOFF	Driller fails to position drill pipe properly before activating BSR	3	3.70E-06	8.86E-03
DP_TOOLJOINT_PRESENT	Drill Pipe Tool Joint is Present	3	3.70E-06	8.86E-03
CAP_STACK_FAILS	Capping Stack unsuccessful	4	3.70E-6	8.86E-03
BOP-PRG-FLO-I02	Subsea manifold pressure regulator I02 fails low (Yellow)	5	9.00E-07	2.16E-03

The F-V importance is based on basic event contributions. When common cause of a component is modeled using separate common-cause basic events, the F-V importance for the common-cause events is treated separately from the independent failure basic event. The independent failures and CCFs cannot appear in the same cut set since a component cannot fail twice, so the cut sets they represent are mutually exclusive. In this case, the F-V values from the common-cause cut set and the independent-failure cut set must be added to obtain the total for that particular component. The same would be true if there were independent failures of a component with different conditional probabilities, such as the BSR failing with nothing across the BOP, and the BSR failing with drill pipe across the BOP. There may be a different probability (a basic event) for each condition, so the total F-V for the BSR would be calculated based on contributions from all of the cut sets containing one or the other of these basic events. In this case, this can be calculated as the sum of the F-V’s of the two basic events, since no cut set contains both events, and adding the F-Vs will therefore not double-count any contributions. However, in general, the F-V is not “additive” in this way; if we wish to compute a F-V measure for a collection of basic events that *can* appear together in cut sets, we need to separate out the cut sets containing any of those basic events, and compute the F-V of the group, based on the contribution to the top event from all those cut sets.

In the example above, cut sets related to any kind of release were used to obtain the F-V importance. Because of the way in which F-V is defined, it can be referred to any end state or combination of end states; in particular, the F-V importance will change if specific end states are used. Table 4-9 shows the Top 5 cut sets for the CAPPINGSTACKCONTAIN end state. The basic event for the BSR failing to close (BOP-CYL-JAM-BSRDP) is only in Cut Set 1, which gives it a F-V importance for this end state of 85.1%. This is in contrast to the 9.67% in the example where all end states were evaluated together. Both answers are correct, but the context is different.

Table 4-9. Top 5 minimal cut sets example for importance measures, CAPPINGSTACKCONTAIN End State.

#	Prob/Freq	Total %	Cut Set	Description
Total	4.35E-05	100	Displaying 5 Cut Sets.	
1	3.70E-05	85.1		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	1.00E-01		BOP-CYL-JAM-BSRDP	BSRs fail to close and seal when drill string is in the hole
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
		End State	CAPPINGSTACKCONTAIN	
2	5.92E-06	13.6		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	1.60E-01		BOP-HUM-ERR-HANGOFF	Driller fails to position drill pipe properly before activating BSR
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
	1.00E-01		DP_TOOLJOINT_PRESENT	Drill Pipe Tool Joint is Present
		End State	CAPPINGSTACKCONTAIN	
3	2.22E-07	0.51		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	1.60E-01		BOP-HUM-ERR-HANGOFF	Driller fails to position drill pipe properly before activating BSR
	1.60E-01		BOP_HUM_ERR_IBOP1	Human error - failure to install IBOP
	8.66E-05		BOP_SCV_FTC_FLTVLV1	Float Valve Fails To Close
	1.00E-01		DP_TOOLJOINT_PRESENT	Drill Pipe Tool Joint is Present
		End State	CAPPINGSTACKCONTAIN	
4	1.74E-07	0.4		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
	4.70E-04		BOP-SHV-LKE-A13	BSR Lock shuttle valve A13 catastrophic leakage
		End State	CAPPINGSTACKCONTAIN	
5	1.74E-07	0.4		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
	4.70E-04		BOP-SHV-LKE-A12	BSR Lock shuttle valve A12 catastrophic leakage
		End State	CAPPINGSTACKCONTAIN	

These examples only use the first five cut sets from the results. In order to get accurate F-V results, all of the cut sets must be used at the truncation cutoff where the risk has converged.

4.3.3.2 Risk Achievement Worth (RAW). The F-V importance shows the relative contributions of components and basic events to the overall risk, given the probability numbers put into the model. But it cannot be concluded that components and basic events that do not show large contributions are unimportant. It may simply be that as a result of their low presumed failure probabilities, the components and basic events do not contribute much to top-event likelihood. Another way to review results is to use the RAW importance measure. The RAW basically executes a drastic sensitivity study: it assumes the basic event is failed by substituting a value of 1.0 for the basic event probability in all cut sets containing the event, and then recalculating the total risk.[j] The new total risk is divided by the total risk before the substitution to establish a ratio of how much the risk would increase if the basic event were failed.

Using the sample data from Table 4-10 for "Drill pipe tool joint present" (basic event DP_TOOLJOINT_PRESENT), if a value of 1.0 is substituted for the nominal value of 1.0E-1 (in Cut Set 3), the new total risk estimate is 4.71E-4; dividing by the original estimate (4.18E-04) gives a RAW of 1.13, meaning that this condition multiplies baseline risk by 1.13.

The RAW is a particularly good measure for identifying single failure points in the model. An example in Table 4-10 is the driller failing to detect a kick in all five cut sets. If a 1.0 is substituted for that basic event (BOP-HUM-ERR-KICKDET), the RAW is about 2392, which is the inverse of the original total risk estimate (4.18E-4). In other words, failure to detect a kick before it reaches the BOP guarantees some kind of release, whether limited or large depending on what happens as the scenario progresses.

The same caution on the significance of F-V when common cause is present also applies to the RAW. If a component has independent and common-cause failure basic events, the basic event RAW only applies to that type of failure for that basic event. Caution should also be applied to the RAW when selecting a truncation cutoff, because some PRA software calculates RAW based on the cut sets that survive truncation. If the truncation cutoff is set too high, some basic events may not appear in the results, and would have erroneous RAW values. For example, if Table 4-10 was cut off to the first four cut sets, the basic event for the subsea manifold pressure regulator failing (BOP-PRG-FLO-I02) would not be present. With no F-V and no RAW tabulated, one might get the false impression that the regulator has no risk value. If all five cut sets are used, the F-V is small, but the RAW is 1.88, meaning that the risk nearly doubles if the regulator has failed. This would have been missed if a reasonable truncation cutoff was not used.

[j] Ideally, a value of "TRUE" is substituted in the logic model, the top event Boolean expression is re-evaluated and only then requantified.

Table 4-10. RAW examples.

#	Prob/Freq	Total %	Cut Set	Description
Total	4.18E-04	100	Displaying 5 Cut Sets. (10551 Original)	
1	3.70E-04	88.6		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
		End State	LIMITEDRELEASE	
2	3.70E-05	8.86		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	1.00E-01		BOP-CYL-JAM-BSRDP	BSRs fail to close and seal when drill string is in the hole
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
		End State	CAPPINGSTACKCONTAIN	
3	5.92E-06	1.42		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	1.60E-01		BOP-HUM-ERR-HANGOFF	Driller fails to position drill pipe properly before activating BSR
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
	1.00E-01		DP_TOOLJOINT_PRESENT	Drill Pipe Tool Joint is Present
		End State	CAPPINGSTACKCONTAIN	
4	3.70E-06	0.89		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	1.00E-01		BOP-CYL-JAM-BSRDP	BSRs fail to close and seal when drill string is in the hole
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
	1.00E-01		CAP_STACK_FAILS	Capping Stack unsuccessful
		End State	RELIEFWELLSEAL	
5	9.00E-07	0.22		
	1.00E+00		INIT-EV_DRILLING	Well Kick While Drilling
	3.70E-04		BOP-HUM-ERR-KICKDET	Operator fails to realize a kick has occurred or does not take timely action
	2.43E-03		BOP-PRG-FLO-I02	Subsea Manifold Pressure Regulator I02 fails low (Yellow)
		End State	ROVCONTAIN	

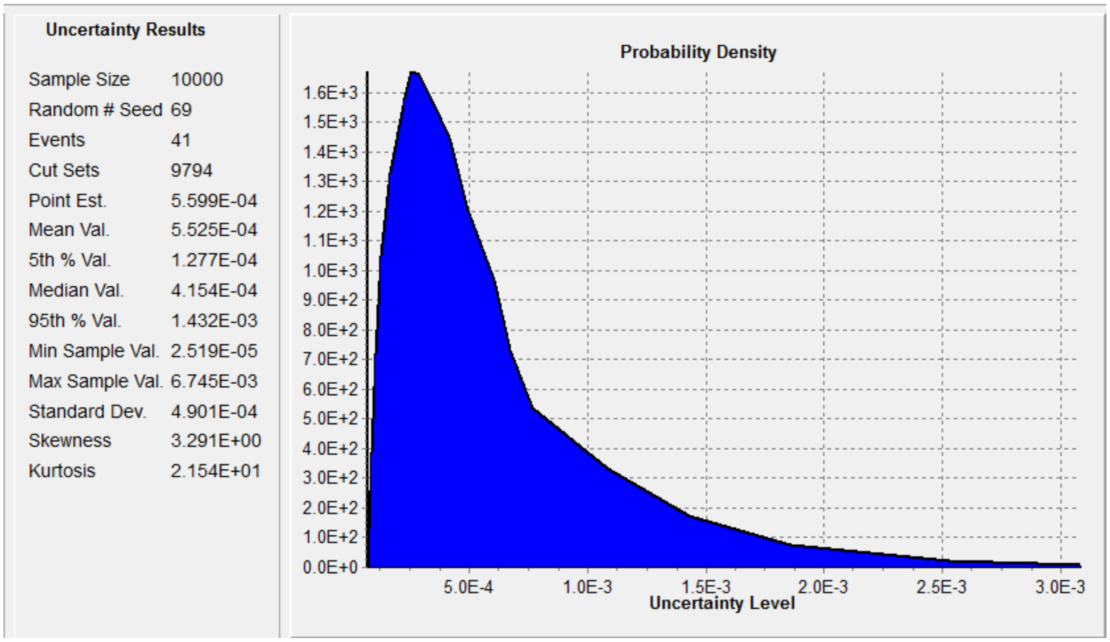
4.3.3.3 Risk Reduction Worth (RRW). RRW is closely related to the F-V importance. Where the F-V importance shows the fractional contribution of a basic event to the total risk, the RRW is a ratio of the total risk if the basic event failure probability is set to 0.0 to the nominal total risk. An easy method for calculating the RRW is:

1.0 / (1.0-F-V importance)

The resulting ratio is the factor by which the risk would be reduced if the failure probability of the basic event was set to 0.0.

4.3.4 Uncertainty

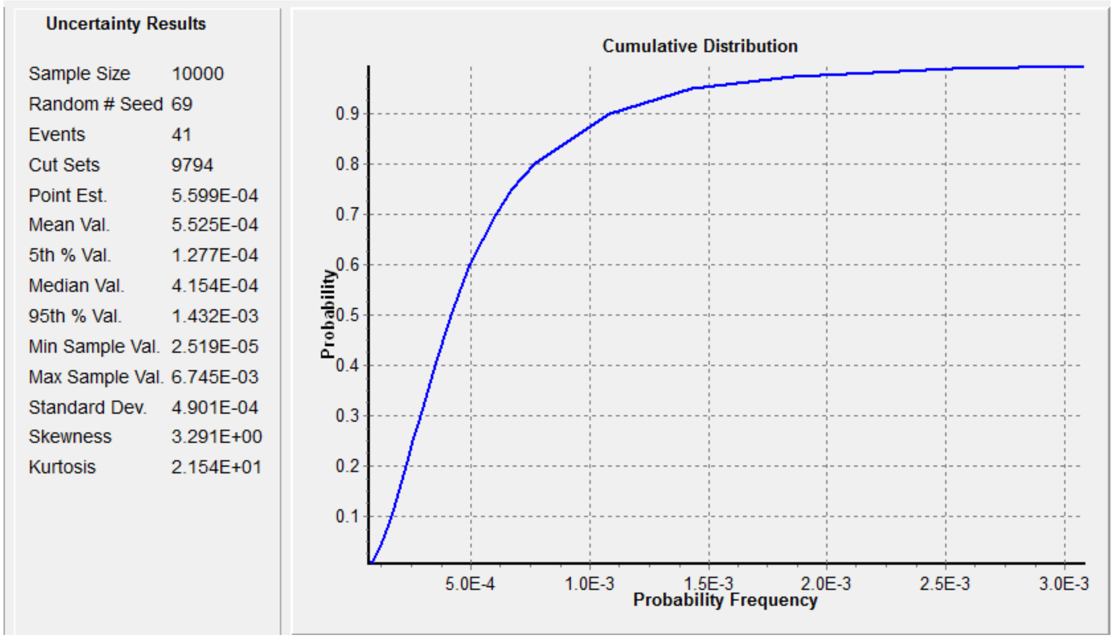
The failure data inputs to a PRA are typically distributions describing the uncertainty around each event being analyzed. These individual uncertainties are used to estimate the uncertainty around the end state(s) of interest in the PRA model. The output from PRA software is typically displayed in two ways: as a probability density curve (Figure 4-4), or a cumulative distribution (Figure 4-5). The probability density represents the relative likelihood (y-axis) for a given probability value (x-axis). The cumulative distribution shows the probability (y-axis) that the end state or event will be less than or equal to the probability value (x-axis).



Uncertainty Results

Sample Size	10000
Random # Seed	69
Events	41
Cut Sets	9794
Point Est.	5.599E-04
Mean Val.	5.525E-04
5th % Val.	1.277E-04
Median Val.	4.154E-04
95th % Val.	1.432E-03
Min Sample Val.	2.519E-05
Max Sample Val.	6.745E-03
Standard Dev.	4.901E-04
Skewness	3.291E+00
Kurtosis	2.154E+01

Figure 4-4. Example probability density function.



Uncertainty Results

Sample Size	10000
Random # Seed	69
Events	41
Cut Sets	9794
Point Est.	5.599E-04
Mean Val.	5.525E-04
5th % Val.	1.277E-04
Median Val.	4.154E-04
95th % Val.	1.432E-03
Min Sample Val.	2.519E-05
Max Sample Val.	6.745E-03
Standard Dev.	4.901E-04
Skewness	3.291E+00
Kurtosis	2.154E+01

Figure 4-5. Example cumulative probability distribution.

The probability density and cumulative distributions are good for describing the uncertainty of a single end state or event. When one is comparing distributions, a chart like that shown in Figure 4-6 can be used that readily displays a comparison of where the mean values lie, as well as the distribution around the means for each point.

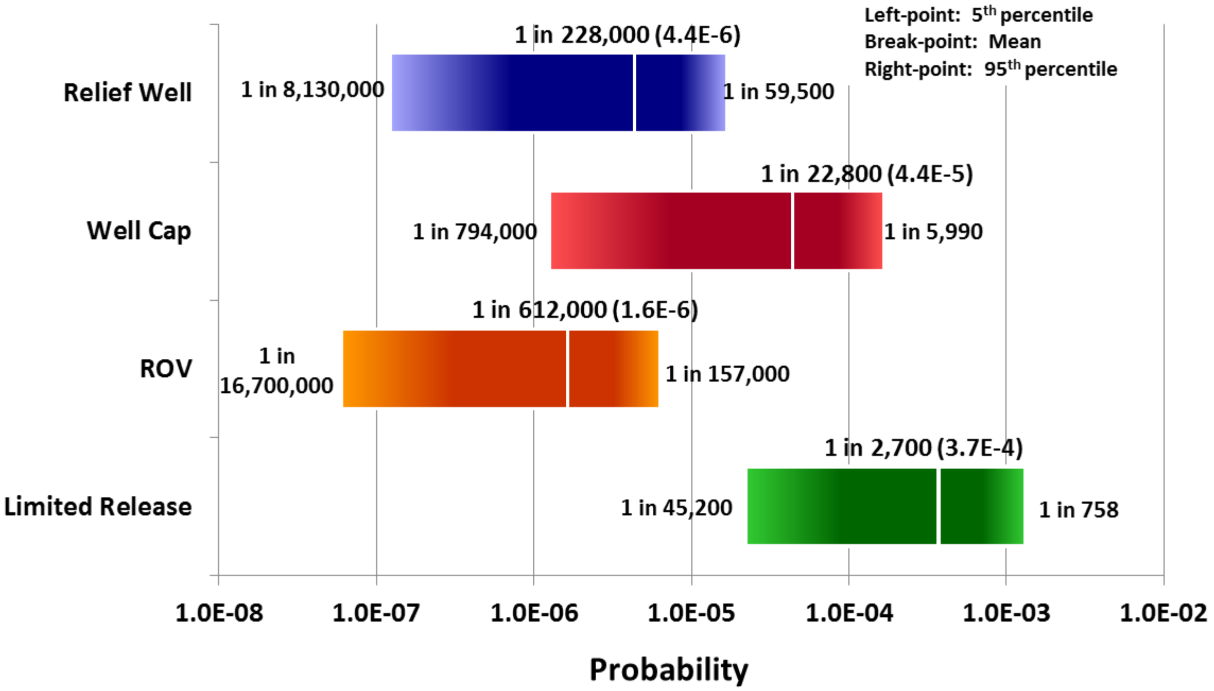


Figure 4-6. Example comparison of end state distributions.

4.3.5 System Level Reliability

A PRA is typically done on a facility with emphasis on a particular end state or states. In developing the model, many systems or functions are analyzed and may be isolated to give insights specifically for those systems and functions. For instance, in the model developed in Section 2, if the analyst wanted to review the causes and contributions to failure of the blind shear to close with drill string across the BOP, the fault-tree results could be used to provide those insights. These are shown in Table 4-11 for the fault tree BLIND_SHEAR_RAM_DR. When reviewing system or function level results from the PRA model, it is important to note the context for which the fault tree was developed. The PRA will have a specific focus and the system analysis may not include all failures of the system.

Table 4-11. Sample fault-tree results for blind shear ram with drill pipe.

#	Prob/Freq	Total %	Cut Set	Description
Total	1.17E-01	100	Displaying 10 Cut Sets. (1348 Original)	
1	1.00E-01 1.00E-01	85.77	BOP-CYL-JAM-BSRDP	BSRs fail to close and seal when drill string is in the hole
2	1.60E-02 1.60E-01 1.00E-01	13.72	BOP-HUM-ERR-HANGOFF DP_TOOLJOINT_PRESENT	Driller fails to position drill pipe properly before activating BSR Drill Pipe Tool Joint is Present
3	4.70E-04 4.70E-04	0.4	BOP-SHV-LKE-A02	BSR High-pressure close shuttle valve A02 catastrophic leak
4	4.70E-04 4.70E-04	0.4	BOP-SHV-LKE-A01	BSR High-pressure close shuttle vlave A01 catastrophic leak
5	4.70E-04 4.70E-04	0.4	BOP-SHV-LKE-A13	BSR Lock shuttle valve A13 catastrophic leakage
6	4.70E-04 4.70E-04	0.4	BOP-SHV-LKE-A12	BSR Lock shuttle valve A12 catastrophic leakage
7	4.54E-04 4.54E-04	0.39	BOP-CYL-JAM-BSL1	Blind shear lock fails to lock or stay locked
8	1.22E-04 1.22E-04	0.1	BOP-PRG-FLO-I0102CCF	CCF of subsea manifold pressure regulators I02 and I01
9	1.31E-05 1.31E-05	0.01	BOP-SVL-FTO-C1213CCF	BSR Lock Solenoid operated valves C12 and C13 CCF to open on demand
10	1.31E-05 1.31E-05	0.01	BOP-SVL-FTO-C0102CCF	BSR High Pressure close Solenoid valves CO1 and CO2 CCF to open on demand

The results in the table are read similar to the end states, except the contributions and importance measures are related to top-event failure instead of an end state.

4.3.6 Sanity Checks of the Results

When the model quantification is completed and results are obtained, the analyst must perform a sanity check to ensure inputs and outputs are appropriate. On a large model, small errors in inputs can make a difference in results.

4.3.6.1 Basic Event Input. A basic check on the input data should be performed at the time of first quantification. A quick review of a basic event listing will reveal any data that has not been input or has a value of 1.0 or 0.0 that was not intended to be. These should be adjusted as necessary.

4.3.6.2 Fault-Tree Linking. Once cut sets are obtained from the first quantification, a review of the basic event names included should be performed to ensure that the correct fault-tree results are being used and the fault trees are linked correctly. Improper fault-tree linking may yield cut sets with fault tree names instead of basic event names. In some PRA software, this can occur when fault trees are used by the event trees at the top-event level instead of using the basic events.

4.3.6.3 Reviewing Results. With the basic event input and fault-tree linking verified, the results should be reviewed to determine if other problems exist. This is a subjective review, based on the analyst's knowledge from building the model. Questions to ask are:

- Does the overall risk make sense?
- Do the top scenarios make sense?
- Are any scenarios that were expected to be risk drivers missing?
- Are the expected symmetries seen (i.e., If two of three pumps are needed for success, are all combinations of two found in the results?)?

If the answer to any of these is less than adequate, the analyst may have to trace through specific scenarios to determine why the expected result is not showing up or higher than expected. The answer may be reasonable, if not, troubleshooting is required to make the appropriate fixes to the model.

4.3.6.4 Sensitivity Studies on Assumptions. When information is lacking, heavy reliance is placed on the analyst's judgment. For example, assumptions made regarding success requirements for pivotal events and for accident progression can significantly affect the PRA results. The effect of such assumptions needs to be investigated by sensitivity analyses. The results of sensitivity analyses should be reported in tabular form and it should include the base-case assumption (the basis for the nominal PRA results), the alternative assumption and its basis, and the change in the numerical results between the base case and the alternative case.

4.4 Can the Model Support the Decision Being Made?

A risk model cannot be perfect; complex risk models contain too many idealizations and abstractions to be literally correct at a high level of detail, even without uncertainties; and in many cases, the uncertainties are significant as well. Is the model good enough to be used in the present decision situation? Or should we do additional work on the model? If the model's results point to one decision alternative with a high degree of confidence – and if the model results are believed – then the work is done. On the other hand, if:

- There is sufficient uncertainty about the model's results to limit our confidence in the present decision, and
- There is a way to reduce that uncertainty, and
- The decision stakes are high enough to justify the additional effort,

then more should be done. However, it is necessary first to understand gain a better understanding of what the risk model is saying.

Risk analysis must accomplish the following:

- Identification of accident scenarios
- Estimation of the likelihood of each scenario
- Evaluation of the consequences of each scenario.

Once this is done, it is necessary to integrate the results into an assurance case, suitable for use by decision-maker(s).

1. The integration includes, among other things, development of best estimates for frequencies and consequences, development of distributions reflecting the uncertainty associated with those estimates, propagation of the uncertainties to obtain final results, and development of appropriate displays to communicate the results with their associated uncertainties. Documentation related to PRA models whose analysis results are used to make critical decisions regarding design, development, manufacturing, and operations that may impact human safety or environmental damage should be reviewed. Specific methods and procedures should be used for assessing and communicating the credibility of PRA model analysis results based on factors such as peer review, input pedigree, uncertainty analysis, results robustness, use history, qualifications of the analysts, and so on.

2. To provide focus for the presentation of results, the results should include identification of system features that are the most important contributors to risk. Insights into relative importance of various features of the system, and the relative importance of various modeling assumptions, may be developed from uncertainty and sensitivity analyses. A discussion of these insights is required to provide the proper interpretation of the “bottom line” conclusions. Such insights should include an appreciation of the overall degree of uncertainty about the results and an understanding of which sources of uncertainty are critical to those results and which are not. In general, many of the insights gained are not strongly affected by the uncertainties. The numerical results need only be accurate enough to allow the decision-maker to distinguish risk-significant elements from those of lesser importance. The level of detail and the style of presentation of risk results depend on the risk assessment objectives. The results section must communicate the project’s motivations and objectives and should be done in a way that clearly establishes the appropriateness of the generated results in meeting the risk assessment objective. For example, if the risk assessment is intended for evaluation of alternative design features as in risk-informed decision-making, the results should be presented in a structure that allows comparison of various design options according to an appropriate ranking scheme.

3. Ultimately the question must be asked: Are the results robust enough to support a decision? If not, what are the soft spots in the analysis (e.g., dominant uncertainties), and what can be done about them?

Appendix A

Example Basic Event Naming Conventions for Fault Trees

As discussed in Section 2.2.5.3, a consistent naming convention for fault-tree basic events is necessary for several reasons. Ultimately, a good naming convention helps in reading and parsing model results in an efficient manner; but even more importantly, the Boolean processing function requires that a given basic event be named consistently in all of the logic models in which it occurs. Serious errors can result if this is not done correctly. If a model development is being carried out by more than one individual, enforcement of this condition and similar conditions is a priority.

Usually the naming convention is in a form similar to:

XXX-YYY-ZZZ-DDDDD

where XXX, YYY, etc. represent identifying attributes to the component and failure mode that may include (as previously discussed):

- The operation being performed (e.g., drilling)
- The system the component belongs to (blow out preventer)
- The subsystem the component belongs to (e.g., yellow pod)
- The component (e.g., shuttle valve)
- The failure mode (e.g., fails to transfer)
- A unique identifier for the valve (usually from a drawing) (e.g., SV837).

The system, component, failure mode, and unique identifier should be included as a minimum, and other fields may be added based on the analysis and the character limitations of the probabilistic risk assessment software being used. Table A-1 and Table A-2 show typical naming conventions for failure modes and components. A three-letter identifier was used for each, but that can vary depending on the analyst’s choice. The number of characters should be related to the number of items to be accounted for in the field. For instance, if the operations being analyzed are drilling, tripping, running casing, and an empty hole, an identifier for operation may only be one letter since only four operations are being considered. Failure modes and components typically have many variations, so allocating three letters allows flexibility for those items as shown in the examples in Tables A-1 and Table A-2. Unique identifiers from drawings or other documents may be variable, so as the last field in the name, it may be desirable to not specify the number of characters for that field.

Table A-1. Example basic event naming convention for failure modes.

Description	Code	Description	Code
Fails to close on demand	FTC	Plugged	PLG
Fails to open on demand	FTO	Short circuit	SHT
Fails to reseat	FRS	Structural failure	STR
Fails to run	FTR	Transfer closed	XFC
Fails to start on demand	FTS	Transfer open	XFO
Fails to transfer on demand	FTT	Fails low	FLO
Jammed	JAM	Degraded	DEG
External leakage	LKE	Fails to operate	FOP
Internal leakage	LKI	Spurious operation	SPO

Table A-2. Example basic event naming convention for components.

Hydraulic/ Pneumatic Components	**Code**	**Instrumentation**	**Code**
Safety relief valve	SRV	Wind sensor	WIS
Hydraulic/ pneumatic cylinder	CYL	Hydroacoustic sensor	HYS
Shuttle valve	SHV	GPS antenna	GPS
Pilot-operated valve	PVL	Gyro compass	GYC
Solenoid-operated valve	SVL	Temperature sensor	TMS
Check valve	SCV	Flow switch	FSW
Accumulator	ACC	Pressure switch	PSW
Reservoir	RES	Level switch	LSW
Pump	PMP	**Fluid System Components**[1]	**Code**
Filter	FLT	Manual valve	VLV
Manual valve	MVL	Motor-operated valve	MOV
Pressure reducer	PRD	Air-operated valve	AOV
Pressure regulator	PRG	Safety relief valve	SRV
Orifice	ORF	Accumulator	ACC
Other	**Code**	Reservoir	RES
Human error	HUM	Pump	PMP
Electrical Components	**Code**	Filter	FLT
Circuit breaker	CBR	Heat exchanger	HEX
Electric power bus	BUS	Orifice	ORF
Relay	RLY	Hydraulic-operated valve	HOV
Battery	BAT	**Mechanical Components**	**Code**
Computer	COM	Diesel generator	DGN
Joystick	JOY	Thruster	THR
Switchboard	SWB	Compressor	CMP

[1] Such as seawater, freshwater, fuel, not hydraulic/pneumatic.

Appendix B

Fault-Tree Gate Logic and Quantification

The primary logic gates used in fault-tree modeling are the OR gate, AND gate, and the COMBINATION gate shown in Figure B-1.

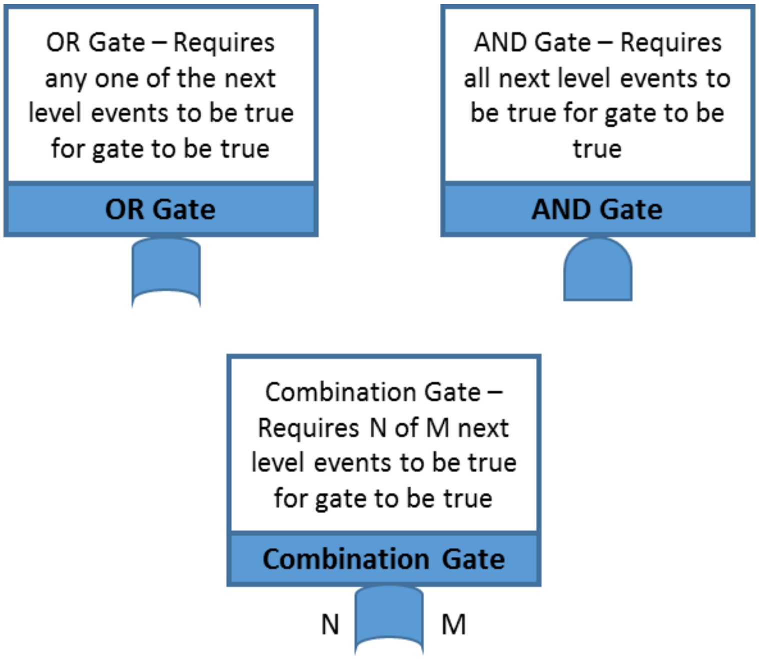

OR Gate – Requires any one of the next level events to be true for gate to be true
OR Gate
AND Gate – Requires all next level events to be true for gate to be true
AND Gate
Combination Gate – Requires N of M next level events to be true for gate to be true
Combination Gate
N
M

Figure B-1. Common fault-tree logic gates.

B-1. OR GATES

The OR gate is used for situations where, if any event under the gate is true (has occurred), then the OR gate will be true (or occur). For instance, if a well kick occurs, the driller has to recognize the well kick and close the annular preventer. For a fault tree, which is developed in failure space, the top event would be "failure to close the annular preventer after a well kick." Since both actions have to occur for success, either one failing (driller failing to recognize the kick, or annular preventer failing to close) would result in the top event being true. The simple OR gate is shown in Figure B-2.

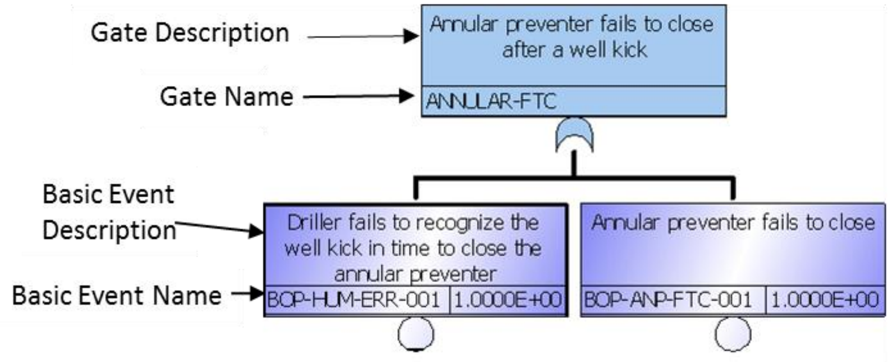

Gate Description
Gate Name
Annular preventer fails to close after a well kick
ANNULAR-FTC
Basic Event Description
Basic Event Name
Driller fails to recognize the well kick in time to close the annular preventer
BOP-HUM-ERR-001
1.0000E+00
Annular preventer fails to close
BOP-ANP-FTC-001
1.0000E+00

Figure B-2. Simple OR gate.

The OR gate ANNULAR-FTC is true if either basic event BOP-HUM-ERR-001 **OR** BOP-ANP-FTC-001 is true.

The output from the OR gate would result in the following cut sets:

BOP-HUM-ERR-001
BOP-ANL-FTC-001.

In Boolean logic, the equation for the results becomes:

ANNULAR-FTC = BOP-HUM-ERR-001 ∪ BOP-ANL-FTC-001

The events in a fault tree are generally considered to be independent (with the exception of common cause). That is, the occurrence of one event does not affect the likelihood of another. Figure B-3 shows a representation of how the events in Figure B-2 are viewed in a Venn diagram if they are independent. As shown in the Venn diagram, if the events are truly independent, either one of them could occur; or, some percentage of the time, both could be true, as represented by the overlap of the two events (labeled “A”). In reality, both would never occur because if the driller fails, the annular preventer will not have a chance to fail, even if a latent failure is present.

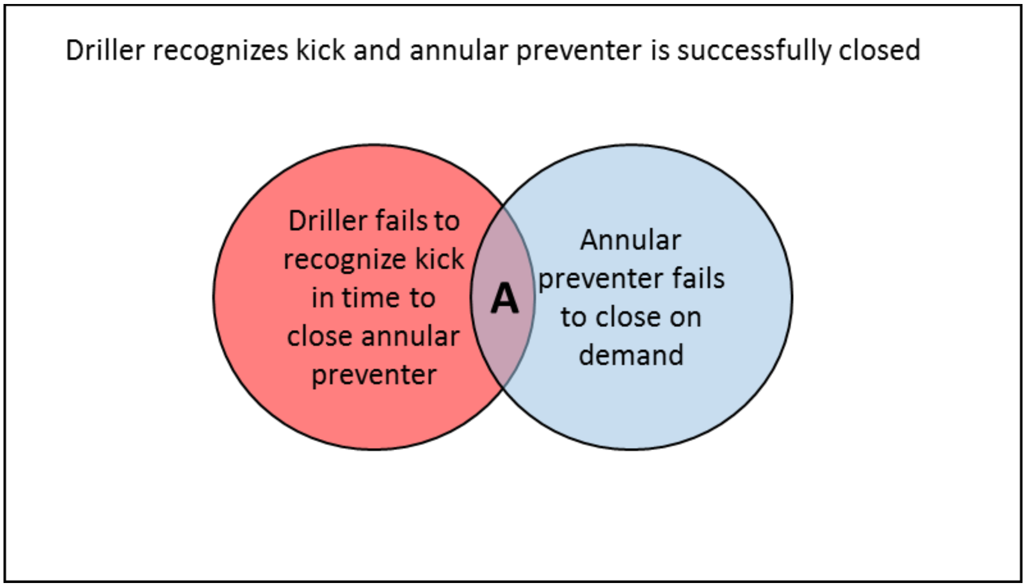

Driller recognizes kick and annular preventer is successfully closed
Driller fails to recognize kick in time to close annular preventer
A
Annular preventer fails to close on demand

Figure B-3. Venn diagram for fault-tree independent events.

Quantification of the OR gate is performed once probabilities are assigned to the basic events. The possibility of the two events being true concurrently must be accounted for if the values of the probabilities are relatively large (i.e., greater than 0.01). In order to do this, the intersection of the two events (“A”) is subtracted from the total in the form:

ANNULAR-FTC = BOP-HUM-ERR-001 + BOP-ANL-FTC-001 - BOP-HUM-ERR-001 * BOP-ANL-FTC-001.

Assigning the probabilities below:

BOP-HUM-ERR-001 = 0.001
BOP-ANL-FTC-001 = 0.001

gives the equation:

ANNULAR-FTC = 0.001 + 0.001 – 0.001 * 0.001
ANNULAR-FTC = 1.999E-3 (B-1)

In this case, the probabilities are small, so the intersection term does not affect the answer significantly. In these cases, the rare event approximation, leaving off the last intersection term, may provide a reasonable answer. If the probabilities were significantly higher, for example 0.5, the equation becomes:

ANNULAR-FTC = 0.5 + 0.5 – 0.5 * 0.5
ANNULAR-FTC = 0.75 (B-2)

In this case, with large probabilities, the answer is significantly affected due to the event independence (the probability of ANNULAR-FTC would be calculated as 1 if it were not corrected for the intersection term).

OR gates may have many inputs, including other gates.

B-2. AND GATES

The AND gate is used for situations where all events under the gate must be true in order for the AND gate to have the value "TRUE." For instance, if a BOP has three pipe rams and closure of any one would stop the well from flowing, the top event for functional failure would be "Failure to close a pipe ram after a well kick." Since any one of the three pipe rams suffices for success, they all must be failed for the top event to be true. The simple AND gate for this situation is shown in Figure B-4.

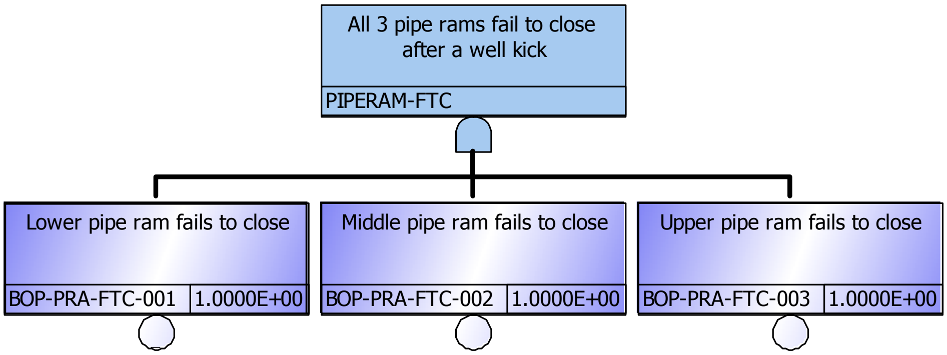

All 3 pipe rams fail to close after a well kick
PIPERAM-FTC
Lower pipe ram fails to close
BOP-PRA-FTC-001
1.0000E+00
Middle pipe ram fails to close
BOP-PRA-FTC-002
1.0000E+00
Upper pipe ram fails to close
BOP-PRA-FTC-003
1.0000E+00

Figure B-4. Simple AND gate.

The AND gate PIPERAM-FTC is true if basic events BOP-PRA-FTC-001 **AND** BOP-PRA-FTC-002 **AND** BOP-PRA-FTC-003 are true.

The output from the AND gate would result in the single cut set:

BOP-PRA-FTC-001* BOP-PRA-FTC-002 * BOP-PRA-FTC-003

In Boolean logic the equation for the results becomes:

PIPERAM-FTC = BOP-PRA-FTC-001 $\cap$ BOP-PRA-FTC-002 $\cap$ BOP-PRA-FTC-003 (B-3)

Figure B-5 shows a representation of how the events in Figure B-4 are viewed in a Venn diagram if they are independent. In the previous case of the OR gate, the area representing failure of the top event was the total shaded area. For the AND gate, the area that satisfies the top-event condition is that where all shaded areas overlap (i.e., intersect), labeled "A" in Figure B-5. Anywhere outside of the area labeled "A," at least one pipe ram has not failed.

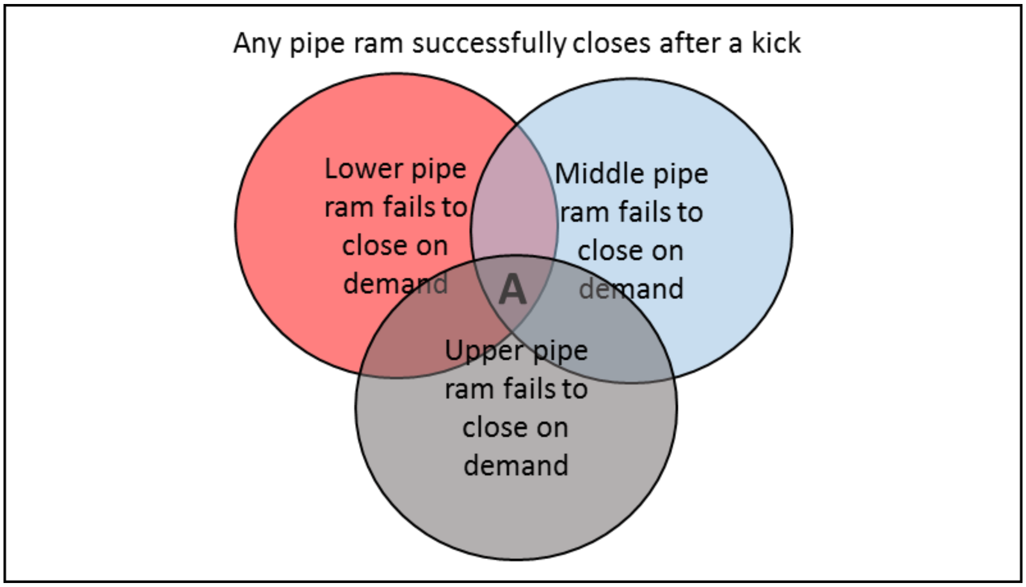

Any pipe ram successfully closes after a kick
Lower pipe ram fails to close on demand
Middle pipe ram fails to close on demand
A
Upper pipe ram fails to close on demand

Figure B-5. Venn diagram for three independent events.

Quantification of the AND gate is performed once probabilities are assigned to the basic events. The equation formed to calculate the probability of area "A" in Figure B-5 is:

PIPERAM-FTC = BOP-PRA-FTC-001 * BOP-PRA-FTC-002 * BOP-PRA-FTC-003 (B-4)

Assigning the probabilities below:

BOP-PRA-FTC-001 = 0.001
BOP-PRA-FTC-002 = 0.001
BOP-PRA-FTC-003 = 0.001

gives the equation:

PIPERAM-FTC = 0.001 * 0.001 * 0.001
ANNULAR-FTC = 1.0E-9 (B-5)

B-3. COMBINATION GATES

The COMBINATION gate is used for situations where M (at least three) events are under the gate and N events (where N is at least two but less than M) must be true in order for the COMBINATION gate to be true or occur. For instance, if a MODU has three thrusters (three is used for simplicity in this example) for station-keeping, and any two operating is enough to maintain position, the top event would be "at least two thrusters fail and station-keeping is lost." Since two of the three thrusters must be operating, if two of the three fail, then the top event will be true. The simple COMBINATION gate for this situation is shown in Figure B-6.

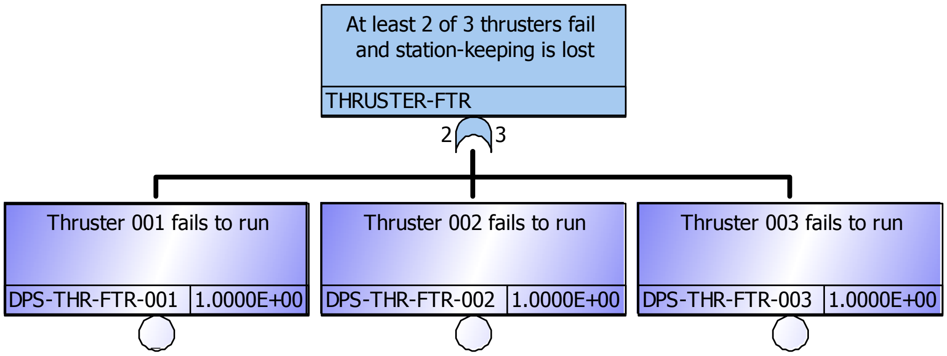

At least 2 of 3 thrusters fail and station-keeping is lost
THRUSTER-FTR
2
3
Thruster 001 fails to run
DPS-THR-FTR-001
1.0000E+00
Thruster 002 fails to run
DPS-THR-FTR-002
1.0000E+00
Thruster 003 fails to run
DPS-THR-FTR-003
1.0000E+00

Figure B-6. Simple COMBINATION Gate.

The COMBINATION gate, THRUSTER-FTO is true if any two of the basic events DPS-THR-FTR-001, DPS-THR-FTR-002, and DPS-THR-FTR-003 are true.

The output from the COMBINATION gate would result in the three cut sets:

DPS-THR-FTR-001 * DPS-THR-FTR-002
DPS-THR-FTR-002 * DPS-THR-FTR-003
DPS-THR-FTR-001 * DPS-THR-FTR-003.

In Boolean logic the equation for the results becomes:

$$\text{THRUSTER-FTO} = \text{DPS-THR-FTR-001} \cap \text{DPS-THR-FTR-002} \cup \text{DPS-THR-FTR-001} \cap \text{DPS-THR-FTR-003} \cup \text{DPS-THR-FTR-002} \cap \text{DPS-THR-FTR-003}. \qquad \text{(B-6)}$$

Figure B-7 shows a representation of how the events in Figure B-6 are viewed in a Venn diagram if they are independent. For the COMBINATION gate, the area that satisfies the top-event condition is that where at least two shaded areas overlap. These areas are labeled “A,” “B,” “C,” and “D” in Figure B-6. Areas “A,” “B,” and “C” are overlaps between two thrusters, and represent the probabilities that each specific combination of two will fail. Area “D” is the overlap of all three thrusters and represents the probability that all three thrusters fail. This area will also satisfy the top event of *at least* two thrusters failing.

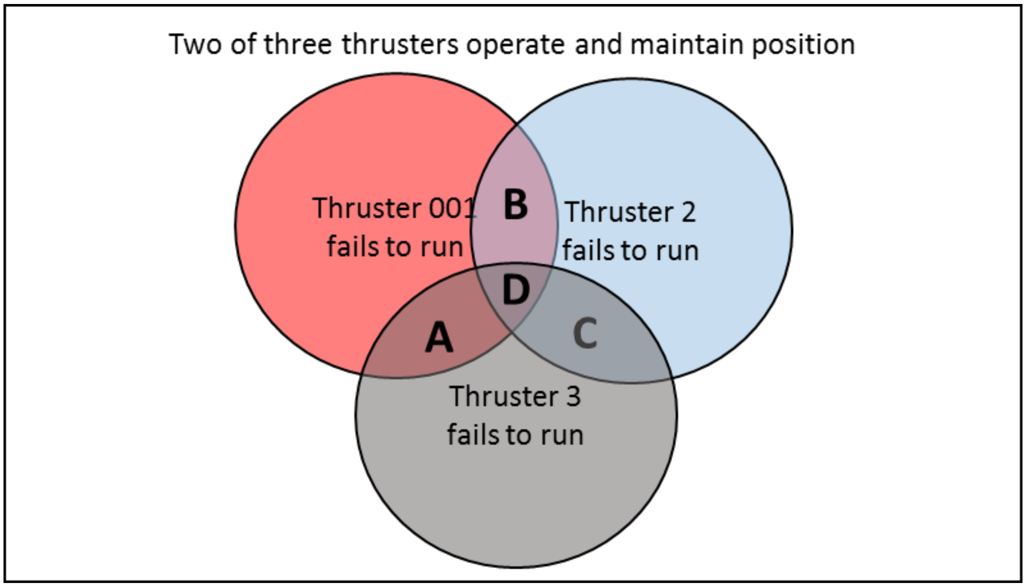

Two of three thrusters operate and maintain position
Thruster 001 fails to run
B
Thruster 2 fails to run
D
A
C
Thruster 3 fails to run

Figure B-7. Venn diagram for three independent events.

Quantification of the COMBINATION gate is performed once probabilities are assigned to the basic events. The equation formed to calculate the probability of Areas “A,” “B,” “C,” and “D” in Figure B-7 is:

THRUSTER-FTO = (DPS-THR-FTR-001 * DPS-THR-FTR-002) + (DPS-THR-FTR-001 * DPS-THR-FTR-003) + (DPS-THR-FTR-002 * DPS-THR-FTR-003) – (2 * DPS-THR-FTR-001 * DPS-THR-FTR-002 * DPS-THR-FTR-003) (B-7)

The first three terms in parentheses in the above equation represent the intersection of each pair of thrusters (Areas “A”,” B”, and “C” in Figure B-7). The Area “D” is included in each intersection term, and would therefore be counted three times if the intersection terms were simply added. The last term in the equation is a correction to account for this over-counting. In this case, the correction is small because the probabilities are small.

Assigning the probabilities below:

DPS-THR-FTR-001 = 0.001
DPS-THR-FTR-002 = 0.001
DPS-THR-FTR-003 = 0.001

gives the equation:

THRUSTER-FTO = 0.001 * 0.001 + 0.001 * 0.001 + 0.001 * 0.001 – 2 * 0.001 * 0.001 * 0.001
THRUSTER-FTO = 2.998E-6 (B-8)

Appendix C

Calculating Frequency, Reliability, and Availability Metrics

This appendix provides a simplified discussion of the basics of quantifying reliability, availability, and frequency of failure metrics for components and systems. It does so using conventional Markov model graphics. This is done in order to clarify how most PRA software uses the component performance information that must be entered in order to quantify the model. Most PRA software does not actually use Markov models, but the standard Markov model representation is a useful reminder of what sort of thing the subject calculations actually do, whether they are based on simulation, solution of Markov models, or hand calculations.

For present purposes, it is assumed that the functions of systems and components are well defined, the failure modes of components and their effects have been identified, the rates of occurrence of these failure modes are quantified in some way, and the system configurations that would be considered "successful" have been defined. For a system having redundancy (more than one way to succeed, despite failed components), this would mean that the number of trains or divisions needed for "success" has been defined; and it is additionally recognized that for different kinds of functional demands, different definitions of "success" might apply.

At any given time, a component that is capable of doing its job is "available" (i.e., it is not out for test or maintenance) and in a "good" state. A very simplified state diagram for a single component is shown in Figure C-1.

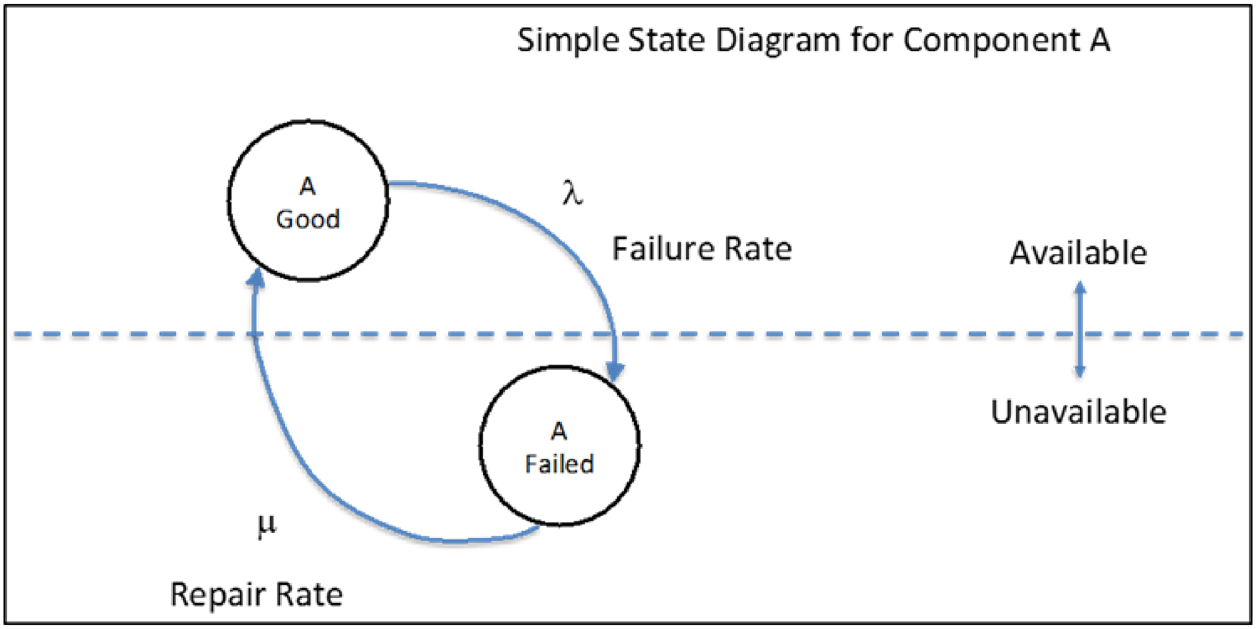


Figure C-1. Simple state diagram for component A.

The circles represent component states, and the arcs represent transitions between states. The arc labeled λ is a “failure”: a transition from “good” to “failed.” This occurs at the failure rate λ, which has the units of “events per unit time.” Analogously, the arc labeled μ corresponds to restoration of the component to “good” status, occurring at the repair rate μ. Within the model underlying this figure, a component is either “good” or “failed,” and the probabilities of these states must therefore sum to 1. If we treat λ and μ as stochastic and constant in time, we can write the following equations for the time rate of change of the probabilities of “up” and “down”:

$$\frac{d(up)}{dt} = -\lambda * p(up) + \mu * p(down),$$

$$\frac{d(down)}{dt} = \lambda * p(up) - \mu * p(down),$$

$$p(up) + p(down) = 1. \qquad \text{(C-1)}$$

This is an extremely simple example of a class of models called “Markov.” A distinguishing feature of these models is that what happens at any given instant depends only on the state of the system at that instant: a Markov model has no memory of what went before. The modeling of “repair” as a purely stochastic phenomenon, occurring independently of how long a component has been down, is a drastic and unrealistic approximation. But it makes this set of equations trivial to solve, and, for some purposes, is a useful starting point. We can do a better job of modeling things like this in discrete event simulation, which is discussed in Section 2.3.

One can solve the above equations straightforwardly. Typically, the initial condition is: at time 0, p(up)=1 and p(down)=0.

This simple model has the property that over sufficient time, it will converge to a condition in which

$$< p(up) >= \frac{\mu}{\mu+\lambda} \; and < p(down) >= \frac{\lambda}{\mu+\lambda}. \qquad \text{(C-2)}$$

This follows from setting the time derivatives to zero, and solving for <p(up)> and <p(down)> using simple algebra. If we were reasoning intuitively, we might argue that the occupancy of the down state is given by the frequency of entering that state (λ), multiplied by the average dwell time in that state ($1/\mu$). This slight difference between this result (1μ) and the above formula results from the need to correct for the availability, discussed below.

Convergence to the steady state is seen in Figure C-2, for illustrative values of λ and μ. The system evolves from its initial condition (p(up)=1) to the steady state given by the above formulas, for the values of λ and μ given on the figure.

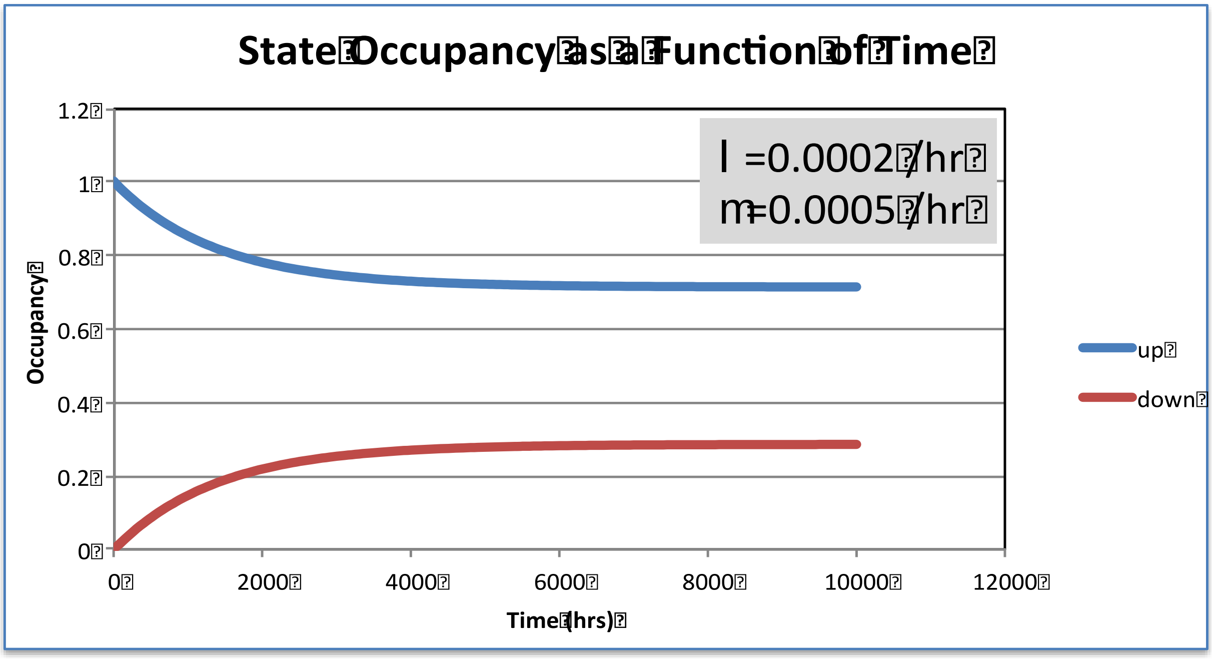


Figure C-2. Steady-state diagram for Component A.

For many components, typical failure rates are on the order of one per many thousands of hours, and typical repair rates are on the order of one per some tens of hours, or less; putting these numbers into the formulae for <p(up)> and <p(down)> yields a number close to 1 for time-averaged availability (<p(up)>), and a small number (equal to 1-availability) for time-averaged *un*availability (<p(down)>).

The rate of failure events actually experienced is not given simply by λ; the component must be “up” in order to be able to fail. The *rate* of failures actually experienced is therefore λ**p(up)*. If *p(up)* is close to unity, then equating the expected rate of failures to λ is a reasonable approximation; but in checking computer calculations, the difference between λ and the observed rate of failures may be observable, if unavailabilities are on the order of a few percent, which can easily be the case.

This point generalizes: the rate at which any arc is traversed is the product of the rate associated with that arc, multiplied by the occupancy (the probability) of the state from which the arc originates.

If a component is known to be “good” at time = 0, then the probability that it is failed at time T is ~ λ*T, for T<<1/λ. *Averaged over this interval*, the probability of being in a failed state is (1/2)*λ*T.

Figure C-3 makes a slightly different point. In Figure C-1, a component was either “good” or “in repair.” In some systems, failure will not immediately be detected, and we need more than a “repair rate” concept to build a model. In Figure C-3, a failed component is placed into repair only when the failure is detected, which could occur either as a result of an actual demand on the system, or as a result of a scheduled test, carried out for the very purpose of detecting a failed state.

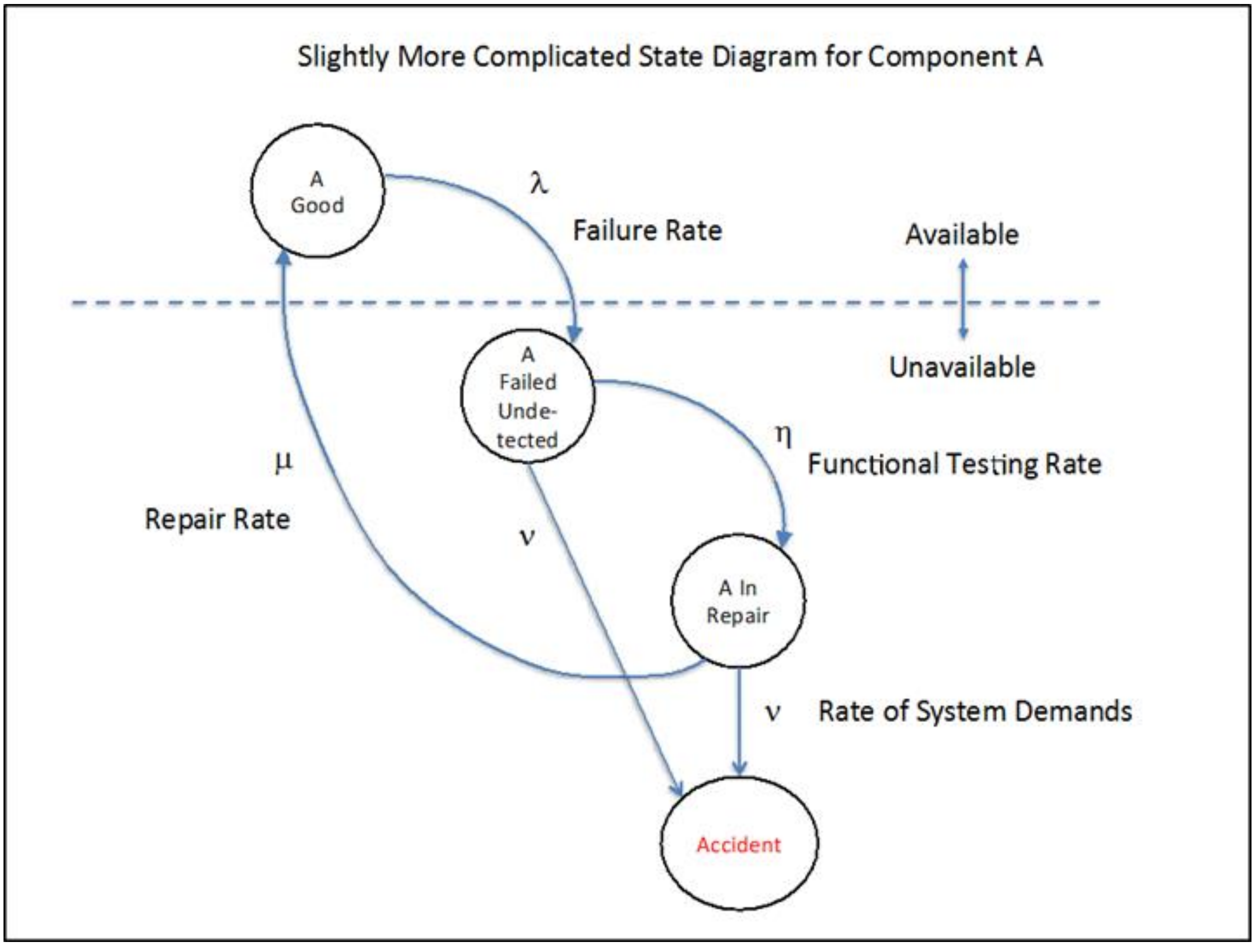


Figure C-3. Slightly more complicated state diagram for Component A.

These figures begin to illustrate a general principle underlying the calculation of complex "rates" (accident frequencies, functional failure frequencies, …). Figure C-4 shows a two-component system with a one-out-of-two success criterion: if the system is demanded and either component works, the system succeeds; if both components are down, the system fails. The accident rate is the rate of demands multiplied by the probability of both components being down. As modeled in Figure C-4, that latter probability depends on the underlying failure rates and repair rates; but as mentioned above, we need a way of detecting component failures (as in Figure C-3) before we initiate repair.

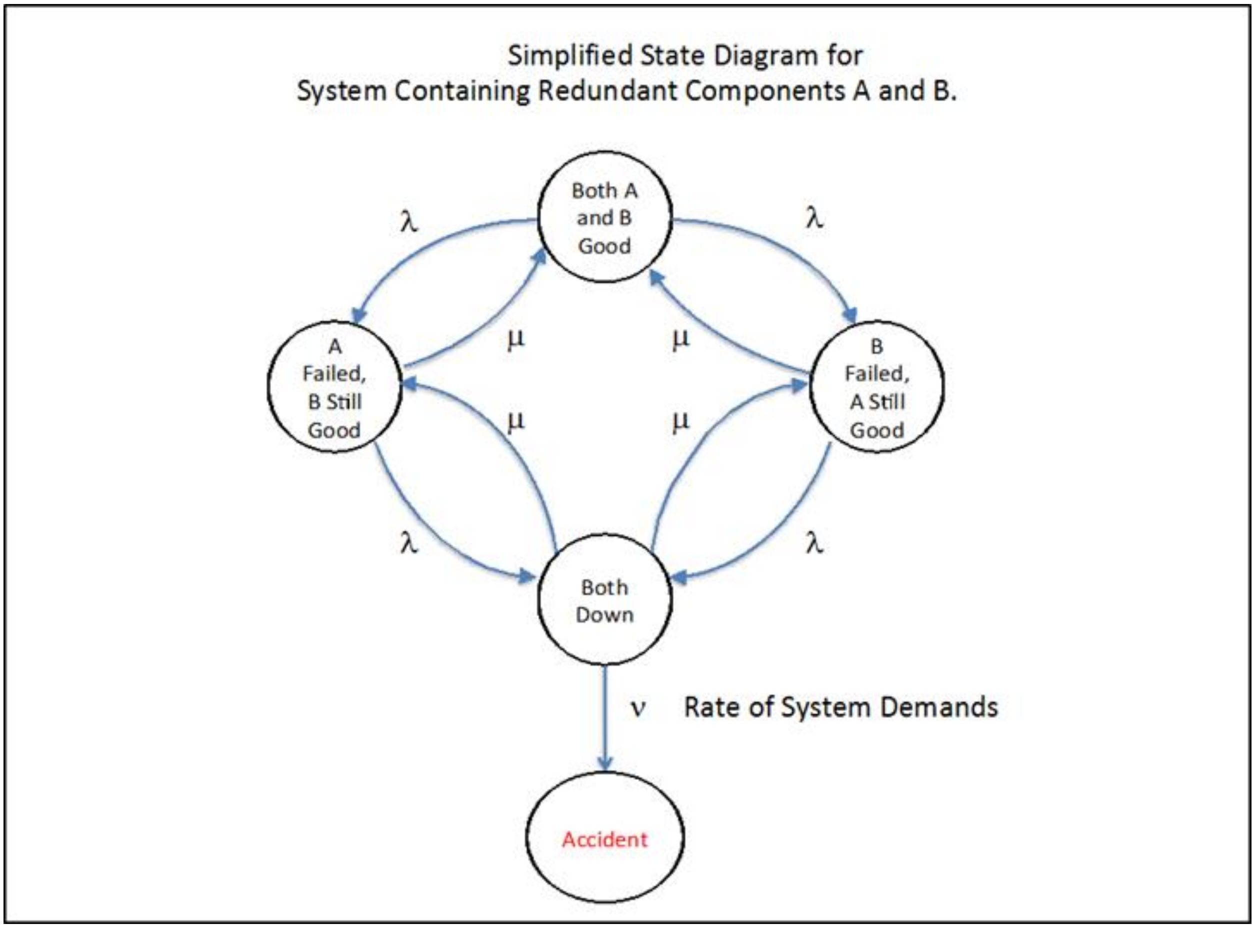


Figure C-4. Simplified state diagram for system containing redundant Components A and B.

If, for some reason, we are interested in the *rate* of system failure, we obtain this by summing the rates of traversing the two arcs into the "both down" state: that is,

$\lambda *$ p(A failed, B still good) + $\lambda *$ p(B failed, A still good).

In all of this, we have assumed that:

- There is no causal linkage between the failures of A and B
- There is no causal linkage between the rate of demands and the failure rates
- All of the rates are constant in time (even the repair rate, and even though this is unlikely to be a realistic description).

“Failure on Demand”

The Reactor Safety Study [C-1] defines “demand probabilities” as:

> *... the probability that the device will fail to operate upon demand for those components that are required to start, change state, or function at the time of the accident [sic]. The demand probabilities, denoted by Q_d, incorporate contributions from failure at demand, failure before demand, as well as failure to continue operation for a sufficient period of time for successful response to the need. When pertinent, the demand data Q_d can be associated with standard cyclic data or can be interpreted as a general unavailability. Human error data can also be associated with demand probabilities (i.e., per action) as discussed in the human evaluation section.*

Not all communities of practice make use of all aspects of this definition. Some argue that if a component is “good” in the instant before a demand, it will (by definition) function during the demand; within this concept, “failures” are either failures in standby, or failures to run, and the occupancies of failed states are quantified accordingly (i.e., in terms of a standby failure rate or a rate of failure to run). Others argue that owing to variability in the stresses imposed by a particular demand, there is a nonzero probability that a nominally “good” component will fail upon the arrival of a demand. Still others would argue for modeling a state between “good” and “bad” (i.e., “degraded”) having a probability of failure on demand that is significant but still less than unity. This is a modeling decision to be evaluated on a case-by-case basis; the present point is that operationally, Q_d is simply the state probability that one multiplies by a “demand” arc to get the frequency of accidents or functional failures, as the case may be.

Although it may seem simple and convenient to lump all causes of component non-performance together, it is conventional to split out maintenance unavailability contributions from actual component failures, because in some applications, operational rules proscribe having multiple components out for maintenance, and the logic model needs to reflect that point: the model should not generate system failure cut sets in which everything is out for maintenance, unless that can, in fact, occur. Sometimes it is necessary to split out failure to start from failure to run, because the consequences are different, or perhaps because common-cause failure considerations are different for the two failure modes, and so on.

C-1. REFERENCES

C-1 “Reactor Safety Study: An Assessment of Accident Risks in U. S. Commercial Nuclear Power Plants,” WASH-1400 (NUREG 75/014), U.S. Nuclear Regulatory Commission, October 1975).

Appendix D

Sources of Failure Rate and Event Data

D-1. BACKGROUND

A fundamental requirement to quantify a risk assessment model is the basic equipment failure rate data. These data are comprised of numerical estimates of failure rate and event data that are used in the model and best represent the failure rate characteristics of the facility. There are several categories of failure that are included in a risk model. These include:

- Loss of containment (leaking or rupture) of equipment that belong to the hydrocarbon containment envelope
- Failure on demand of a component within a safeguard system when required
- External event rate of occurrence for events that challenge the facility to maintain critical safety and environmental integrity functions.

Ideally, parameters of probabilistic risk assessment (PRA) models of a specific system should be estimated based on operational data of that system. The next most representative data is that from the fleet of similar facilities operated by the same entity.

Often, however, the analysis has to rely on a number of sources and types of information if the quantity or availability of system-specific data are insufficient. In such cases surrogate data, generic information, or expert judgment are used directly or in combination with (limited) system-specific data. According to the nature and degree of relevance, data sources may be classified by the following types:

- Historical performance of successes and failures of an identical piece of equipment under identical environmental conditions and stresses that are being analyzed (e.g., direct operational experience).
- Historical performance of successes and failures of an identical piece of equipment under conditions other than those being analyzed (e.g., test data).
- Historical performance of successes and failures of a similar piece of equipment or similar category of equipment under conditions that may or may not be those under analysis (e.g., another program’s test data, or data from handbooks or compilations). General engineering or scientific knowledge about the design, manufacture, and operation of the equipment, or an expert’s experience with the equipment.

D-2. GENERIC DATA SOURCES

Generic data is surrogate or non-specific information related to a class of parts, components, subsystems, or systems. Most generic data sources cover hardware failure rates. All other data categories, particularly human and software failure probabilities, tend to be much more mission-specific, system-specific, or context dependent. As such, generic data either do not exist or need to be significantly modified for use in a PRA.

The international offshore industry has performed risk and reliability assessments for a variety of facilities for over 30 years. Each of these quantitative evaluations tends to increase the general collection of risk and reliability information when this information is stored or published for later use. In addition to the individual quantitative evaluations, various industry entities also manage failure data and incident reporting systems. A selection of offshore industry data collection systems includes:

- Guidelines for Process Equipment Reliability Data with Data Tables [D-1]
- Process Equipment Reliability Database (PERD)
- Failure Rate and Event Data for Use within Risk Assessments (HSE PCAG)
- Failure Frequency Guidance Process Equipment Leak Frequency Data for Use in QRA
- Lees' Loss Prevention in the Process Industries (Third Edition)
- OGP Risk Assessment Data Directory
- OIR/12
- Offshore Reliability Data
- Pipeline and Riser Loss of Containment (PARLOC) Report
- WellMaster Reliability Management System
- Worldwide Offshore Accident Database.

These data sources are presented in this guideline along with content descriptions. These sources are commonly utilized in PRAs conducted for offshore facilities. This list is not exhaustive nor endorsed for use, but simply a compilation of frequently used sources. They are presented in alphabetical order in Tables D-1 through D-11.

Table D-1. Guidelines for Process Equipment Reliability Data with Data Tables.

Name	Description
Sponsor/Author	Center for Chemical Process Safety of the American Institute of Chemical Engineers
Data Types	The level three taxonomy contains 50 component types under the following groups: • Electrical equipment • Instrumentation • Process equipment • Protection Systems.
Description	Failure rate data handbook, multi-industry sources
Number and Type of Records	Book 300 pages, 75 individual failure rate estimate pages
Frequency of Update	None
Time Frame	Prior to 1989
Data Access	Commercial publication
Notes	The PERD handbook is a compilation of data tables based on literature review of estimates from many industries and from proprietary files of previously analyzed and selected information. There is no clear relationship to analysis of individual failure events although the format resembles other handbooks that are based on estimates derived from analysis of event data from equipment populations. The intent of the data is for use in the chemical process industry. Failure rate estimates are given as lower, mean, and upper bound as failure rates or demand failure probabilities, by failure mode.
Reference	ISBN 0-8169-0422-7

Table D-2. PERD.

Name	Description
Sponsor/Author	Center for Chemical Process Safety of the American Institute of Chemical Engineers
Data Types	Relief devices
Description	Event failure database
Number and Type of Records	2,000 relief valve inventory records and over 5,000 proof test event records
Frequency of Update	0
Time Frame	2001–2013
Data Access	Tiered membership scheme. Access to raw event data to allow statistical data analysis for contributing members.
Notes	The PERD database is based on taxonomies developed within the PERD project and is an extension of [D-1]. Relief devices were selected to collect event and test data and implement the database.
Reference	http://www.aiche.org/ccps/resources/process-equipment-reliability-database-perd

Table D-3. Failure Rate and Event Data for Use within Risk Assessments (HSE PCAG).

Name	Description
Sponsor/Author	UK Health and Safety Executive, Hazardous Installations Directorate
Data Types	Categories include mechanical, electrical, bulk transport, and moveable storage. Specific types include vessels, reactors, valves, pumps, hoses and couplings, flanges and gaskets, pipelines, and compressors.
Description	Non-mandatory reference compiled by the agency for assisting their risk assessments
Number and Type of Records	A compilation of many references ranging from proprietary study reports to textbooks comprising 96 pages of data tables and background information
Frequency of Update	0
Time Frame	1972–2012
Data Access	Publication available from HSE website.
Notes	HID CI5 has an established set of failure rates that have been in use for several years in quantitative risk assessments (QRAs) submitted for land use planning cases. The estimates "do not necessarily take account of all factors that could be relevant and significant at particular installations." However, in the absence of site-specific data, the values given here may serve as a starting point for safety reports.
Reference	http://www.hse.gov.uk/landuseplanning/failure-rates.pdf

Table D-4. Failure Frequency Guidance Process Equipment Leak Frequency Data for Use in QRA.

Name	Description
Sponsor/Author	DNVGL
Data Types	• Compressors centrifugal and reciprocating • Filters

	• Flanges • Heat exchangers (air cooled, plate, shell, and tube) • Pig traps • Process piping • Pumps (centrifugal and reciprocating) • Instruments • Valves (actuated and manual) • Pressurized process vessels • Atmospheric storage tanks
Description	A proprietary publication containing guidance and data on process equipment leak frequency for use in a QRA. In this document, a detailed review and comparison is made between the Det Norske Veritas taxonomy and frequency values and the UK HSE Hydrocarbon Release Database taxonomy and frequency values. Additional comparisons are made to guidance developed by Flemish and Dutch governments for the same purpose.
Number and Type of Records	The guide is 40 pages, with 20 pages of data tables presenting leak frequency by equivalent hole size for each equipment type
Frequency of Update	Continuously
Time Frame	2005–2012
Data Access	Proprietary publication available for purchase from DNVGL
Notes	The leak frequency data contained in the guidance document was generated by the LEAK software, which is an application that contains a continuously updated database of leak frequency data and a structured computational capability for leak frequency calculations.
Reference	https://www.dnvgl.com/services/calculate-leak-frequency-data-leak-1759

Table D-5. Lees' Loss Prevention in the Process Industries (Third Edition).

Name	Description
Sponsor/Author	Texas A&M University, Department of Chemical Engineering
Data Types	• Vessels and tanks • Pipework • Heat exchangers • Rotating machinery • Valves • Instruments • Process computers • Relief systems • Fire and gas detection systems • Fire protection systems • Emergency shutdown systems • Utility systems • LNG plants • Leaks • Ignition • Explosion following ignition • Fires • Explosion • Transport

	- External events
Description	A well-known, seminal reference three-volume text compiling the wide range of topics relevant to process safety
Number and Type of Records	Appendix 14 of this reference is titled Failure and Event Data. It compiles 38 pages of reference failure rate data
Frequency of Update	3rd Ed (2005), 2nd Ed (1994), 1st Ed (1979)
Time Frame	Cited references range from 1960–2004
Data Access	Commercial publication
Notes	Failure rate data contained in the book are compilations of many failure rate publications from numerous industries. Failure rate estimates are reproduced from cited publications.
Reference	ISBN 0-7506-7589-3

Table D-6. OGP Risk Assessment Data Directory.

Name	**Description**
Sponsor/Author	International Association of Oil and Gas Producers
Data Types	• Major accidents • Occupational risk • Land transport accident statistics • Aviation transport accident statistics • Water transport accident statistics • Construction risk for offshore units • Process release frequencies • Risers and pipeline release frequencies • Storage incident frequencies • Blowout frequencies • Mechanical lifting failures • Ship/installation collisions • Ignition probabilities • Consequence modelling • Structural risk for offshore installations • Guide to finding and using reliability data for QRA • Vulnerability of humans • Vulnerability of plant/structure • Escape, evacuation and rescue • Human factors in QRA
Description	The Risk Assessment Data Directory is a series of guidance documents that provide data and information for use to improve the quality and consistency of risk assessments with readily available benchmark data. The directory includes references for common incidents analyzed in upstream production operations.
Number and Type of Records	20 individual documents (datasheets)
Frequency of Update	1st Ed (1997), 2nd Ed (2009)
Time Frame	Prior to 2009
Data Access	Commercial publication
Notes	This series of documents was commissioned with the specific goal of defining

	generic data for use in QRAs
Reference	http://www.iogp.org/pubs

Table D-7. OIR/12.

Name	Description
Sponsor/Author	UK Health and Safety Executive
Data Types	Hydrocarbon release event database compiled by the UK Health and Safety Executive, with periodic publications of the analysis of these data in publically available report format.
Description	Event data are required to be submitted under the Reporting of Injuries, Diseases and Dangerous Occurrences Regulations 1995 - (RIDDOR 95). OIR/12 addresses offshore hydrocarbon release events.
Number and Type of Records	585 event records
Frequency of Update	Continuous
Time Frame	2001–2008, previous data deemed inconsistent with current analysis taxonomy and analysis requirements
Data Access	Publication available online
Notes	Data are analyzed by time trend, platform type, platform age, release magnitude, system, cause. Data prior to 2001 are presented with disclaimer.
Reference	http://www.hse.gov.uk/research/rrpdf/rr672.pdf

Table D-8. Offshore Reliability Data.

Name	Description
Sponsor/Author	OREDA, managed, produced and distributed by Veritec, followed by Sintef/Det Norske Veritas
Data Types	Comprehensive topsides and subsea production equipment, safety equipment, and limited onshore exploration and production equipment
Description	OREDA is a project organization sponsored by eight oil and gas companies with worldwide operations. OREDA's main purpose is to collect and exchange reliability data among the participating companies and act as the forum for co-ordination and management of reliability data collection within the oil and gas industry.
Number and Type of Records	Event records from 278 installations, 17,000 equipment items with 39,000 failure and 73,000 maintenance records. The database also includes subsea fields with over 2,000 years operating experience.
Frequency of Update	6th Ed (2015), 5th Ed (2009) 4th Ed (2002), 3rd Ed (1997), 2nd Ed (1992), 1st Ed (1984)
Time Frame	Corresponding with updates
Data Access	Tiered membership scheme. Access to raw event data to allow statistical data analysis for contributing members. Handbook available for purchase.
Notes	All estimates in these handbooks are derived from statistical analysis of event data.
Reference	https://www.oreda.com/

Table D-9. PARLOC Report.

Name	Description
Sponsor/Author	Oil and Gas UK
Data Types	Pipeline and riser
Description	The PARLOC Report is a source of risk assessment data for generic loss of containment frequencies and covers pipelines and risers in the offshore oil and gas industry.
Number and Type of Records	206 incident events, loss of containment and near miss. 10,000 km-yr pipeline, 4,000 riser-yr from the UK sector of North Sea, eastern Irish Sea, West of Shetland
Frequency of Update	1990, 1992, 1994, 1996, 2001 online in 2006/07Hard copy update commenced 2013
Time Frame	1988–current
Data Access	Commercial publication
Notes	Most complete and homogeneous dataset of subsea pipeline and riser incident event data
Reference	http://oilandgasuk.co.uk/parloc.cfm

Table D-10. WellMaster Reliability Management System.

Name	Description
Sponsor/Author	Seven member companies, managed by Exprosoft
Data Types	Subsea (subsurface) equipment types, limited subsea (seabed) equipment, e.g., X-mas Trees.
Description	The world's largest database of reliability data for well and subsea equipment
Number and Type of Records	6,000 wells/40,000 well years
Frequency of Update	Continuous
Time Frame	1986–2016
Data Access	Online access available commercially
Notes	All estimates are derived from statistical analysis of event data using the online application
Reference	https://wellmaster.exprosoft.com

Table D-11. Worldwide Offshore Accident Database.

Name	Description
Sponsor/Author	DNVGL
Data Types	Accident events from global population
Description	Event data including name, type, and operation mode of the unit involved in the accident, date, geographical location, chain of events, causes and consequences, and evacuation details
Number and Type of Records	6,451 accidents occurring among 3,795 operating units

Frequency of Update	Continuous
Time Frame	1986–2016
Data Access	Purchase of data search consultancy or a database subscription. The program is a web application.
Notes	Comprehensive database of offshore accident event data
Reference	https://www.dnvgl.com/services/world-offshore-accident-database-woad-1747

It is important to recognize the perspective of the risk modeler in order to establish requirements on the quality of failure rate and event data to be used in a risk model. Once a complete risk model is constructed and quantified, it is often the case that a large number of individual failure rate and event data input values do not strongly influence the overall calculated level of risk or contribute to insights provided by analyzing risk contributors. This being the case, the requirements for high-fidelity and representative failure rate and event data should vary corresponding to the significance to the calculated risk results. In short, if the failure rate and event data do not significantly influence the results, then we can use lower quality estimates.

D-3. SYSTEM-SPECIFIC DATA COLLECTION AND CLASSIFICATION

System-specific data can be collected from sources such as:

- Maintenance logs
- Test logs
- Operation records.

In the majority of cases, system-specific data are gathered from operation and test records in their “raw” form (i.e., in the form that cannot be directly used in a statistical analysis). Even when data have already been processed (e.g., reduced to counts of failure), care must be exercised to ensure that the data reduction and processing are consistent with QRA modeling requirements, such as having a consistent failure mode classification, and correct count of the total number of tests or actual demands on the system).

In collecting and classifying hardware failure, a systematic method of classification and failure taxonomy is essential. A key element of such taxonomies is a classification of the functional state of components. One such classification system has been offered in D-2. Using a taxonomy implies a knowledge structure used to describe a parent-child relationship (i.e., a hierarchy). Under the guidelines for evaluation of risk and reliability-related data, the taxonomy provides the structure by which data and information elements provide meaning to analysts. Within the risk and reliability community, a variety of taxonomies and associated definitions are used. ISO 14224 [D-3] provides a taxonomy for collecting and processing of equipment failure data in the petroleum industry.

When concerned about the physical *causes* of failures, a set of physics-based causal factors would be required. However, this low level of information is not necessary if the inference being made for a specific component or system is concerned with—in general—failures or successes. If, instead, we wished to infer the probability of failure conditional upon a specific failure mechanism, we would need to have information related to the nature of failure (e.g., the physical causal mechanisms related to specific failures).

In other words, this classification can take place via a failure modes and effects analysis, similar to the functional failure modes and effects analysis. Kumamoto and Henley [D-4] carried this idea one step further when they proposed a formal cause-consequences structure to be stored in an electronic database. In their approach, specific keywords, called modifiers, would be assigned to equipment failures. For example, modifiers for on-off operation included: close, open, on, off, stop, restart, push, pull, and switch. Alternative hierarchy related to system/component/failure modes may look like:

System

└ Component

└ Failure Mode

└ Affected Item

└ Failure Mechanism

└ Failure Cause

With regard to the intended function and in reference to a given performance criterion, a component can be in two states: *available* or *unavailable*. The unavailable state includes two distinct substates: *failed* and *functionally unavailable*, depending on whether the cause of the unavailability is damage to the component or lack of necessary support such as motive power. The state classification also recognizes that even when a component may be capable of performing its function (i.e., it is available), an incipient or degraded condition could exist in that component, or in a supporting component. These failure situations are termed *potentially failed* and *potentially functionally unavailable*, respectively. These concepts have proven useful in many PRA data applications.

Another aspect of reliability data classification is the identification of the failure cause. In the context of the present discussion, the cause of a failure event is a condition or combination of conditions to which a change in the state of a component can be attributed. It is recognized that the description of a failure in terms of a single cause is often too simplistic. A method of classifying causes of failure events is to progressively unravel the layers of contributing factors to identify *how* and *why* the failure occurred. The result is a chain of causal factors and symptoms.

A hierarchy of *parts or items* that make up a component is first recognized, and the functional failure mode of the component is attributed to the failure or functional unavailability of a subset of such parts or items. Next the physical sign or *mechanism* of failure (or functional unavailability) of the affected part(s) or item(s) are listed. Next the *root cause* of the failure mechanism is identified. Root cause is defined as the most basic reason or reasons for the failure mechanism, which if corrected, would prevent reoccurrence. The root cause could be any causal factor, or a combination of various types of causal factors.

D-4. REFERENCES

D-1 *Guidelines for Process Equipment Reliability Data with Data Tables*, Center for Chemical Process Safety, American Institute of Chemical Engineers, New York, February 1989, ISBN 0-8169-0422-7.

D-2 OREDA, “Offshore and Onshore Reliability Data 6th edition,” 2015.

D-3 ISO 14224:2016, *Petroleum, petrochemical and natural gas industries – Collection and exchange of reliability and maintenance data for equipment*, 2016

D-4 Kumamoto H. and E. J. Henley, *Probabilistic Risk Assessment and Management for Engineers and Scientists*, 2nd Edition,” IEEE Press, Piscataway, New Jersey, 1996.

Appendix E

Further Discussion of Bayesian Updating

E-1. SIMPLE EXAMPLES

E-1.1 Updating of Prior for a Poisson Example

In this example, the goal is to estimate an hourly failure rate for a component, assuming that the failures obey a Poisson distribution. We choose a lognormal distribution for the prior, and a Poisson distribution for the likelihood model.[k] The operational data for the component category indicate two failures in 10,000 hours.

Since the prior distribution is lognormal, and the likelihood function is Poisson, and these two are not "conjugate," the posterior distribution must be derived numerically. The prior and posterior distributions are shown in Figure E-1, along with the "maximum likelihood estimate" (MLE). The MLE, the value of the parameter for which the likelihood function P(E|parameter) is maximum, is 2E-4 in this case (failures/hours). Note that the probability density functions are plotted as a function of *log* frequency.

The posterior distribution is shifted from the prior distribution towards the MLE. This is typical.

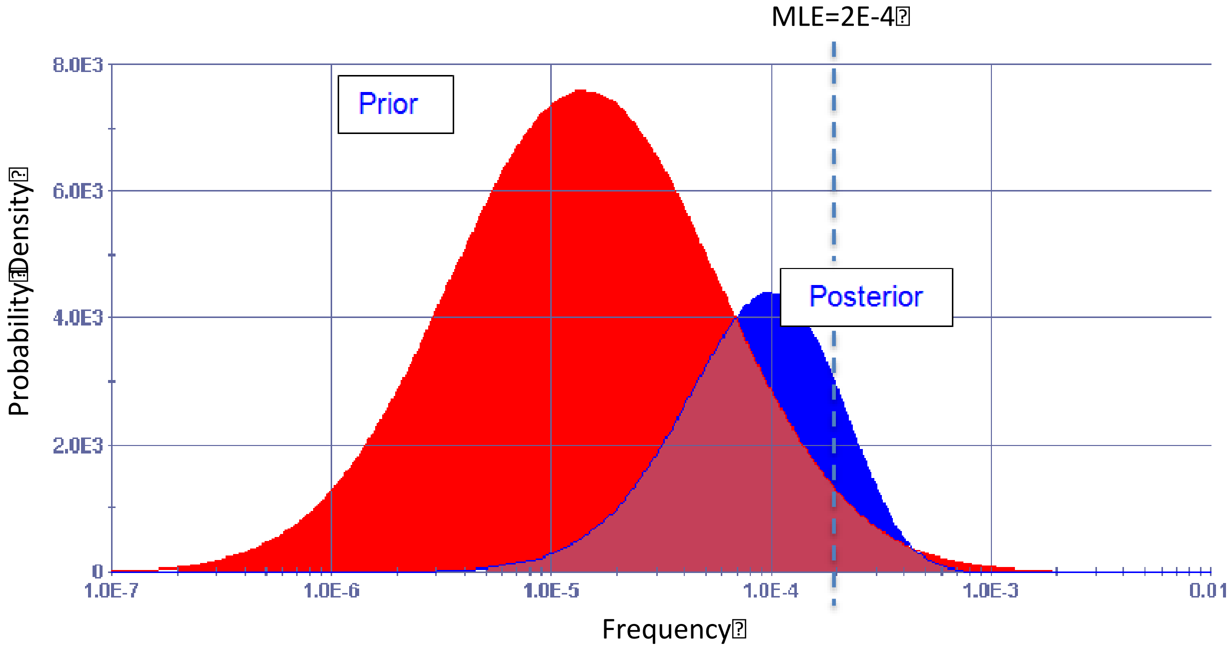


Figure E-1. The prior distribution distributions for the failure rate example.

[k] These distributions are discussed in Section 2.2.1.6.2.

E-1.2 Updating Distribution of Failure-on-Demand Probability

In this example, the goal is to estimate a failure-on-demand probability. We have chosen the prior distribution of a particular component failure probability on demand to be a Beta distribution with mean equal to 1E-4 failures per demand, and standard deviation equal to 7E-5. The operational data for the component category are one failure in 2,000 demands. Our chosen likelihood model is the Binomial distribution, which is conjugate to the Beta prior. Therefore, the posterior distribution is also a Beta distribution. The prior and posterior distributions are shown in Figure E-2, along with the MLE (1/2000=5E-4).

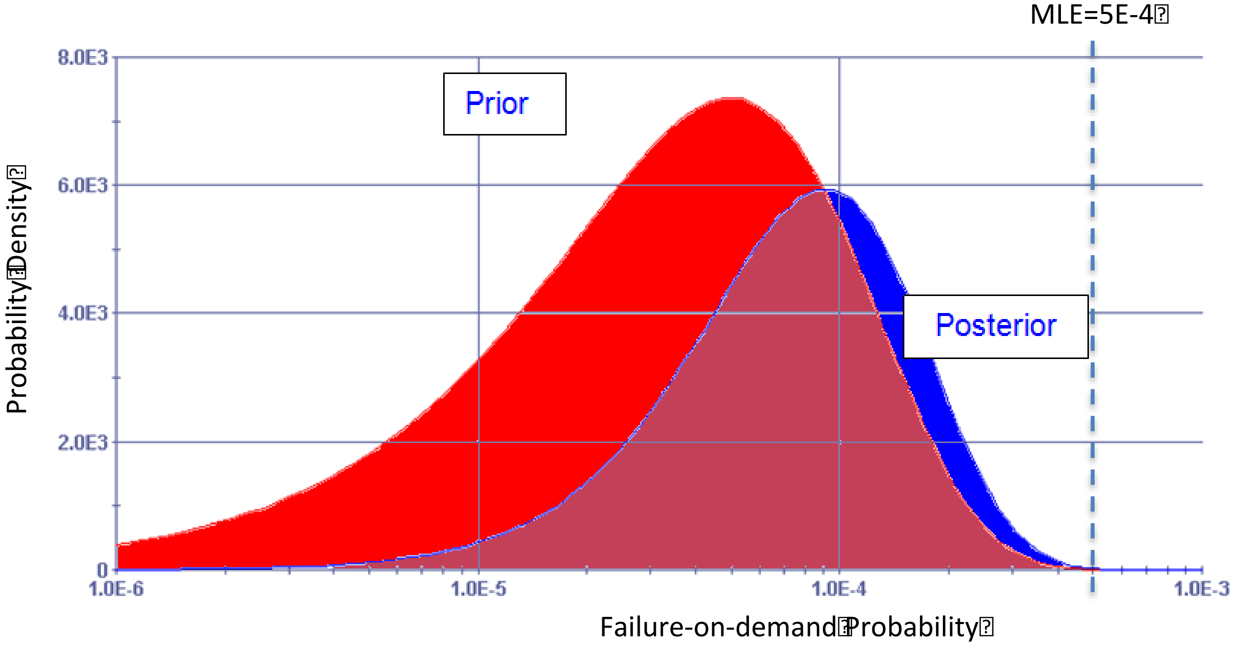


Figure E-2. The prior and posterior distributions for the failure-on-demand example.

Once again, we see the posterior distribution shifted towards the MLE. In this example, however, we also see something else: the MLE is quite unexpected, given the prior. This situation should give us pause, and is discussed in Section E-2.

E-1.3 Sequential Updating

Bayes' Theorem provides a mechanism for updating the state of knowledge when the information is accumulated in pieces. The updating process can be performed sequentially and in stages corresponding to the stages in which various pieces of information become available. If the total amount of information is equivalent to the "sum" of the pieces, then the end result (posterior distribution) is the same regardless of whether it has been obtained in stages (by applying Bayes' Theorem in steps) or in one step (by applying Bayes' Theorem to all the evidence at once).

Example—Updating Failure Rate for a Poisson Process. A component is tested for 1000 hours in one test and 4000 hours in another. During the first test the component does not fail, while in the second test one failure is observed. We are interested in an updated estimate of the component failure rate assuming a Gamma prior distribution with parameters α= 1, β = 500.

Approach 1: Sequential. We first start with prior (Gamma distribution): $\Gamma(x|\alpha=1, \beta=500)$. We also use Poisson as the likelihood function: $\Pr(k_1=0|T_1=1000, \lambda)$, representing the first data set ($k_1 = 0$ in $T_1 = 1000$ hours). The parameters of the resulting Gamma posterior distribution are $\alpha'=\alpha+k_1=1+0=1$, and $\beta'=\beta+T_1=500+1000=1500$ (ref. Section 2.2.6 for a general discussion of the update process).

Next, we use this posterior as the prior distribution in for a new update, using the second data set. The prior is $\Gamma'(l|\alpha'=1, \beta'=1500)$ and the likelihood is again Poisson: $\Pr(k_2=1|T_2=4000, l)$. The parameters of the posterior after the second update are $\alpha''=\alpha'+k2=1+1=2$, and $\beta''=\beta'+T_2=1500+4000=5500$. The posterior mean is given by:

$$\bar{\lambda} = \frac{\alpha''}{\beta''} = \frac{2}{5500} = 3.6E - 4 \text{ failures/hour} \qquad \text{(E-1)}$$

Approach 2: Use all the data at once. The total evidence on the failure history of the component in question is $k=k_1+k_2=0+1=1$, and $T=T_1 +T_2 =1000+4000=5000$. Starting with our prior distribution with parameters $\alpha = 1$, $b = 500$, the above cumulative evidence can be used in one application of Bayes' Theorem with Poisson likelihood: $\Pr(k=1|T_2 =5000, l)$. The parameters of the resulting Gamma posterior distribution are $\alpha'=\alpha+k=1+1=2$, $\beta'=\beta+T=500+5000$, and

$$\bar{\lambda} = \frac{\alpha'}{\beta'} = \frac{2}{5500} = 3.6E - 4 \text{ failures/ hour,} \qquad \text{(E-2)}$$

as for Approach 1. In this case, the equivalence of the two approaches is clear from the functional form: the numerator of the mean is given by the sum of the prior α and the total failures, and similarly for the denominator. But the principle holds generally. Note that the validity of the result depends on the trials all being exchangeable; this point is discussed in Section E-2.

E-2. PRIOR DISTRIBUTIONS, LIKELIHOOD MODELS, AND DATA APPLICABILITY

It is difficult to avoid making choices in the assessment of uncertainty. This section discusses what needs to be considered when those choices are made, taking as a point of departure the situation noted in the demand failure probability update illustrated above.

E-2.1 All Prior Distributions Contain Information

The amount of information that a prior distribution contains can be quantified (in various ways), and the distributions that contain the least amount of information can be found and used if so desired; but there is no such thing as a prior that contains no information. This is not necessarily a bad thing: none of the questions to which Bayesian analysis is applied, even the simplest, can be answered without some information. Classical methods avoid this need by answering different questions, which may or may not be similar enough to the questions we ask for the answers to be useful to us.

Take the simple example of determining whether a coin is fair or two-headed. If we flip a coin and it comes up heads six times in a row, we have collected some evidence against the coin being fair. But we must use our prior belief about the coin to reach a conclusion about whether we believe this coin is fair. If we just obtained the coin at the bank, where it is extremely unlikely to find a forged or misprinted coin mixed in amongst thousands of real coins, we should not be in a hurry to assume the coin is two-headed even after ten consecutive flips. On the other hand, if we found the coin on the floor of a magic shop, we should seriously consider the possibility that someone dropped a joke coin. Without prior probabilities we cannot answer the question “how likely is this to be a fair coin?” All we can say is that it is 64 times easier for a two-headed coin to generate the data we collected than for a fair coin to generate it. A classical statistician would say “there is less than a 5% chance that a fair coin will come up heads six times in a row,” but he will say nothing about the chance that we are holding a two-headed coin in our hands now. A blind devotee of non-informative priors will assign a prior probability one-half to the fair and two-headed possibilities, calculate posterior probabilities of 1/65 for fair, 64/65 for two-headed—and probably make a lot of false accusations of two-headed coins. A reasonable Bayesian will assign a prior probability somewhere between 1/1000 and 1/1000000 to the two-headed possibility if he is at the bank (and not suspicious at all of a coin that comes up heads six times), and something closer to 1/100 if he is at a magic shop (and start seriously considering the possibility of a two-headed coin after several heads in a row come up).

Proper choices of model form and prior should encapsulate all available information that we had about a problem before we started collecting data. This is an easy task if we have a small set of alternatives and easily quantifiable information, but it can be a very hard task for real-world problems—often impossible to do perfectly. The better of a job we do at choosing the prior, the better our final answer will be.

The importance of choosing the correct model—choosing the correct family of distributions to try to fit one’s data to—is often downplayed, but the success of the whole model-fitting enterprise depends on the reasonableness of this model. This is also a convenient and effective way to encapsulate information about the allowable range of the data.

For example, consider the distribution of times between failures of some component (or times between eruptions of a volcano, or some similar problem). If failure appears to be a completely random process, perhaps controlled by some external process, then a constant hazard rate, and exponential distribution of failure times, may be appropriate. If components accumulate damage through use, or pressure builds up during the interval between eruptions, then the hazard function increases with time, and the underlying distribution has a thinner-than-exponential tail. The Weibull distribution is popularly used to model component lifespans because it has a polynomial hazard function, convenient to model rapidly increasing risk of failure as the component exceeds its design lifespan. An nth degree polynomial hazard function corresponds to a distribution with tail thickness proportional to ($e^{-\left(x^{(n+1)}\right)}$.

On the other hand, consider the length of time a car sits in a parking lot. One’s natural reaction to seeing the car sit there for a full day is not “surely the owner is going to be back any second now!” but rather “while originally I thought that car would only be there for a few minutes or hours, I should now entertain the possibility that it will be left here all week or all month”—a situation modeled by a distribution with decreasing hazard function and a very long (decaying more slowly than exponential) tail, such as a lognormal or power-law distribution.

If the variable of interest takes on values only in a certain range, then it usually makes good sense to choose a prior that covers only that same range. When the data are confined to [0,1], the Beta prior is a common choice; when the data are confined to positive numbers, the Gamma, Weibull, and lognormal are common choices. If there is a theoretical reason to expect the data to take a certain form, the model should usually be chosen to match that form: if you are trying to estimate the size of a 100-year flood from annual peak discharges of a river, fitting a Gumbel distribution to the data is probably a better choice than fitting some general-purpose distribution.

Modelers should be wary of choosing a prior that is at odds with the real world. In particular, remember that the normal distribution has support from $-\infty$ to $+\infty$, and if one models lengths, or times, or some other variable that is nonnegative by a normal distribution, the resulting posterior will always assign a nonzero probability to negative values (maybe very small, but still nonzero). The normal distribution is also very thin-tailed: it is extremely hard to make the posterior mean be more than a few standard deviations from the prior mean. This is a feature for some applications like modeling measurement errors that are known to be small compared to the quantity being measured; but this can be a significant flaw if a diffuse prior is desired.

A similar problem arises if one uses a lognormal distribution to model the probability of a rare event, as is commonly done in human reliability analysis. The lognormal is an excellent choice when a prior distribution spanning several orders of magnitude is needed (as when you do not know whether a rare event has probability 0.001 or 0.00001) but the lognormal has support on [0, ∞], not just [0,1], so care must be taken to handle the case when the posterior has substantial mass beyond 1. Truncation (treating all mass beyond 1 as if it were concentrated at 1) is only reasonable if very little mass is beyond 1.

E-2.2 How Much Information does a Prior Contain?

Sometimes it is easy to directly interpret how much information the prior contains relative to the data set. The simplest example is the Beta-Binomial model: a Beta(a,b) prior can be interpreted as providing "the same amount of information as if we had obtained a successes and b failures already." The same kind of interpretation can be applied to a Gamma-Poisson model for failure rates.

A popular intuitive assessment of how much information the prior contains relative to the data is obtained by inspecting the posterior mean: the posterior mean always lies between the prior mean and the mean of the data. For many Bayesian models, the posterior mean can be thought of as a weighted average of these two means. If the posterior mean is, say, three-fourths of the way from the prior mean to the mean of the data, one interprets the data as containing three times as much information as the prior contained. The example of Section E-1.2 is a case where the prior contained more information.

For several of the most popular Bayesian models that use conjugate priors, including the Beta-Binomial, Gamma-Poisson, and normal-normal, the intuitive interpretations in the two paragraphs coincide with each other and can be made rigorous.

Even so-called "non-informative" priors contain information. Three popular families of priors—maximum entropy priors, Jeffreys priors, and reference priors—seek to minimize the information content of the prior, for three different technical definitions of 'information.' The first maximizes the information-theoretic entropy of the prior, subject to some given constraints; the second creates a distribution of the shape of which is invariance under any change of variables; the last maximizes the expected Kullback-Leibler divergence ("information gain") between prior and posterior, given some assumptions about the posterior. Jeffreys and reference priors coincide in one dimension but differ in multidimensional problems.

Revisiting our coin that came up heads six times in a row at the beginning of the section, suppose we start with a Beta(10,10) prior—a fairly strong belief the coin is approximately fair. Our posterior is a Beta(16,10) distribution. Our prior has a mean 1/2, our data have mean 1, our posterior has mean 16/26 (approximately 0.615). The posterior mean moved 23% of the way—6/26ths—from 1/2 toward 1, in agreement with the intuitive assessment that our prior contained 10+10=20 coinflips worth of information, to which we add six more with our new data.

Had we done the same experiment with a Jeffreys or reference prior, Beta(1/2,1/2), our posterior would be a Beta(6 1/2, 1/2) distribution with mean 6.5/7 (approximately 0.929). The posterior mean moved 6/7ths of the way toward the mean of the data, in agreement with the interpretation that a Jeffreys prior provided "as much information as one coinflip, one-half heads and one-half tails."

The situation one might intuitively think of as a "no-information prior"—Beta(0,0)—is in fact a very strange improper prior, with all its mass concentrated at p=0 and p=1.

E-2.3 Bias

MLEs are the "gold standard" of classical statistical estimation because of their desirable properties. Chief among these are that MLEs are asymptotically unbiased and efficient (no other asymptotically unbiased estimate has a smaller variance). Note that being *asymptotically* unbiased does *not* guarantee that an MLE based on finite sample size is unbiased. Some MLEs (like $\bar{x}/n$ for the mean of a normal distribution) are always unbiased, but others -- like the MLE for the variance of a normal distribution, $\sum(x_i - \bar{x})^2/n$, are biased. In this last case it is easy to compute a bias correction -- this is why the usual formula for sample variance is $\sum(x_i - \bar{x})^2/(n-1)$ -- but in many other cases it is not a trivial task to remove the bias. When computing MLEs for complicated models using small data sets, the bias problem may be severe.

MLEs are a special case of a Bayesian point estimate, with a uniform (possibly improper) prior. Bayesian point estimates are almost always biased as a result of the choice of prior: for a Binomial distribution with a uniform (Beta(1,1)) or Jeffreys (Beta(1/2,1/2)) prior, observing *x* successes in *n* trials results in a point estimate of *(x+1)/(n+2)* (uniform) or *(x+1/2)/(n+1)* (Jeffreys), in contrast to the unbiased *x/n*. Generally speaking, the more information in the prior, the more strongly the Bayesian estimate is biased. In a well understood problem, this may be considered a feature, not a flaw: when we have strong prior knowledge we may want our posterior estimate to be only slightly different, and even without strong prior knowledge, we may want to prevent the estimate of a Binomial probability from being unreasonably close to 0 or 1, for instance.

It is important to remember that a Bayesian update never "fails": it always returns an answer. If you ask a question about which you have collected little or no data, the answer it gives is driven entirely by the prior. Especially in cases where it is not obvious how much information the prior contains, or an experimenter uses a standard non-informative prior without thinking about how that will affect his answer, this can lead to surprisingly bad, or at least unexpected, answers.

This is simply a limitation of having sparse data. Careful choice of prior can mitigate this issue but not avoid it entirely. Consider a rare type of accident that is only expected to occur once in 1000 site-years of exposure. No one site is going to have sufficient local experience to independently estimate its accident rate; instead, each site is going to use the nationwide average rate as a prior for a Gamma-Poisson model, and update it with its local experience. How strong of a prior should each plant use?

Suppose one takes a very strong prior like Gamma(10,10000). A plant that has no accidents in 10 years will update this to a Gamma(10,10010) posterior. A plant with two accidents in 10 years—wildly unlikely, if that site's true accident rate is close to once in 1000 years of exposure—updates this to a Gamma(12,10010) prior, and claims that its site-specific accident rate is around once in 800 years. *Using too strong of a prior distribution means that grossly underperforming sites are not called to account for their poor performance.*

Now suppose one takes a very weak prior like Gamma(0.01,10). Now the plant with two accidents in 10 years has a Gamma(2.01,30) posterior, estimates a site-specific accident rate of once in 15 years, and is forced to take corrective action. But a site with no accidents in 10 years has a Gamma(0.01,20) posterior, and, on the basis of only 10 years of experience containing almost no real information—we EXPECT not to see a once-in-1000-years accident in any given 10-year period—now claims its site-specific accident rate has improved to once in 2000 years.

There is *no* prior that can completely avoid both of these two flaws. Any scheme that ensures poorly performing sites are "properly punished" will also "improperly reward" well-performing (or just lucky) sites.

The non-informative priors for the Gamma-Poisson model have shape parameters near 0.5—the intuitive interpretation is "pretend 0.5 accidents happened at each site, in addition to however many were really observed"—as a compromise so that sites with one or more accidents see some kind of significant increase in site-specific estimated rate, while sites with zero accidents do not calculate impossibly rare site-specific rates.

E-2.4 Bayesian Analysis Assumes a Static Underlying Process

Bayesian modeling is rooted in the notion that the observed data are *exchangeable*. Many classical methods are based on the similar but stronger idea that all of the observations are *independently and identically distributed*. This constitutes an assumption that the order in which the data were collected does not matter. If one flips a coin 10 times today, and flips the same coin 10 more times tomorrow, these can be pooled into a set of 20 equally important observations.

This assumption breaks down if the underlying process has changed over the observation period. Estimating the value of real estate based on last year's (or last decade's) sales prices gives poor results if economic conditions have changed. Similarly, using the failure rate of brand-new pumps to estimate the failure rate of broken-in and well-maintained pumps, or using well-maintained pumps to estimate the failure rate of worn-out pumps, has that same issue.

When data are sparse, the temptation to pool data over an unreasonably long time period is strong. Sometimes it is justified: if one is averaging over thousands of pumps nationwide, perhaps it is fair to assume that pumps are constantly wearing out and being replaced, such that the overall distribution of pump ages remains static, even though any one individual pump's behavior may be different next year than last year. This is a difficult assumption to defend. Conditions nationwide may change—in an economic downturn, facilities across the nation may defer maintenance, or a new law may be passed mandating replacement at a certain age—or maybe a large proportion of units entered service at the same time: look at what happened to Social Security when it assumed the ratio of earners to retirees would stay approximately constant forever.

For convenience, we often use models that we know are an over-simplification of the real world. Using several years of old data to create a prior distribution for what we expect to see next year is a very common practice. It is important, when doing so, not to just directly use the distribution of past observations as one's prior, but rather to use a more diffuse prior that takes into account the possibility that conditions are the same now as they were in the past.

E-2.5 Assessing Goodness of Fit

Assessing whether new data are consistent with a proposed model is an important task, but it is not a task for which a single universal method exists. One (extreme) perspective is that if the prior has properly encapsulated everything we know, the posterior should always be correct: that is, one of things the posterior tells us is exactly how much we should change our belief after collecting surprising data. In principle this is true, but in practice, people commonly use less-than-perfect priors, either for mathematical convenience or due to lack of information that would have been desirable while choosing a prior.

Classical tests exist for determining whether a data set appears to have been drawn from a particular distribution, and for determining whether two data sets appear to be drawn from the same distribution or not. These tests, or their Bayesian adaptations, may be suitable for answering some questions of this type.

One particularly valuable classical test consists of fitting two models to the same data, with one model (the "reduced model") a special case of the other (the "full model"). If the larger family of distributions fits the data significantly better than the smaller, embedded, family of distributions does, this is evidence that the reduced model is inadequate for the task at hand.

This is typically done when one wishes to argue for the more complicated model. When a study reports that it has found that family income has a significant effect on academic success “after controlling for gender and race,” it means that it fit a model that explains success by income, gender, and race, and shown that that model is significantly better than a model that explains success only by income.

One might, for instance, assess whether a linear trend is a good fit to a scatter plot, by fitting a quadratic or cubic model to the same data set, and conclude that if the quadratic term of the larger model is statistically significant, then the simple linear model is a poor explanation of the data. Note that a non-significant result does not *prove* the simple model is correct, but it is evidence in favor of that claim.

This type of test can be adapted to almost any problem of interest. The question of data seemingly inconsistent with a prior might be approached in this way by, for instance, fitting both a simple Poisson model with Gamma prior to a set of count data, and an over dispersed Poisson model. If the later model fits much better than the former, one has a basis for arguing that there is something wrong with the first model: either you needed a more complicated model all along, or the prior and the data were not consistent, or something else.

E-2.6 Surprise

An alternative to formally testing goodness of fit (or lack thereof) is assessing whether the data are “surprising,” without considering any particular alternative. This is useful, as a sanity check and to get a feel for one’s data; but developing a firm rule for how surprising data must be before saying “our model is wrong” is not possible without bringing in some outside information (such as showing that another model fits the data better.)

Various people have proposed formal definitions of the notion of ‘surprise’. No one definition has achieved universal acceptance. Bayarri and Berger [E-1] review the options that have been used in the past. The classical p-value has sometimes been interpreted as a measure of surprise [E-2]. Two more recent alternatives are the “s-value” [E-3] and the Kullback-Leibler divergence from the prior to the posterior, which is being vigorously promoted as a “formal Bayesian theory of surprise” [E-4].

This last proposal, grounded in the same mathematics that underlies the reference prior, may be the most likely of these to stand up to the test of time, though the emotionally charged notion of “surprise” is not likely to remain attached to it. The Kullback-Leibler divergence is more often described in drier terms like “bits of information gained” (in the formal Shannon-information sense, not the informal “information contained in n observations” terms used earlier in this appendix).

Returning one last time to our coinflip example, suppose we have a Beta(10,10) prior, and we flip a coin six times. We would be not surprised at all to see three heads and three tails, or 4-2 or 2-4; we might be mildly surprised to see six heads in a row. If we flipped the coin 10 times, we would not be surprised by anything between say 8-2 and 2-8; 9 heads out of 10 would be about as surprising as 6 out of 6; 10 out of 10, more surprising still—about one bit of information more surprising, seeing something that was supposed to be a 50-50 proposition happen an additional time.

If we calculate the Kullback-Leibler divergence between prior *p(x)* and posterior *q(x)*

$$\int q(x)\ \log_2 \frac{q(x)}{p(x)} dx \qquad \text{(E-3)}$$

we see that the divergence between Beta(10,10) and Beta(13,13) is 0.024 bits—almost no surprise at all; between Beta(10,10) and Beta(14,12) is 0.112 bits; between Beta(10,10) and Beta(15,11) is 0.379 bits; and between Beta(10,10) and Beta(16,10) is 0.837 bits. The Kullback-Leibler divergence between Beta(10,10) and Beta(15,15) is 0.054 bits; between Beta(10,10) and Beta(18, 12) is 0.654 bits; between Beta(10,10) and Beta(19,11) is 1.14 bits; and between Beta(10,10) and Beta(20,10) is 1.79 bits.

The question "how surprising is surprising enough to cause us to doubt that we chose an appropriate prior?" still lacks a rigorous answer.

E-3. REFERENCES

E-1 Bayarri, M. J. and James O. Berger, "Measures of Surprise in Bayesian Analysis," Institute of Statistics and Decision Sciences, Durham, North Carolina, 1998.

E-2 Reinhart, A., "Statistics Done Wrong: the Woefully Complete Guide," No Starch Press, March 2015.

E-3 Howard, J. V., "Significance Testing with No Alternative Hypothesis: A Measure of Surprise," Springer Science+Business, 2009.

E-4 Itti, L. and P. F. Baldi, Bayesian Surprise Attracts Human Attention," Advances in Neural Information Processing Systems, Vol. 19," Cambridge MA: MIT Press, 2006.

Appendix F

Population Variability Modeling

Population variability modeling solves the problem of how best to estimate facility-specific risk-model parameters, given that facility-specific data are limited but industry data are more plentiful, albeit most likely inhomogeneous. From a broader perspective, it is also of interest to understand just how variable certain parameters are, and the values over which those parameters can reasonably be expected to range, in a given population of facilities.

Population variability modeling was introduced by Kaplan [F-1] in the 1980s and applied to modeling the frequency of loss of offsite power (LOOP) at nuclear power plants. Most plants lose power very seldom, but the fleet as a whole has several such events per year; the rate varies quite significantly from one plant to another, and plant risk is sensitive to this parameter, so we do not wish to use a generic value derived by pooling all the losses and dividing by the total exposure time. In Kaplan's original work, the population variability distribution (PVD) portrayed the relative fraction of plants having a given LOOP frequency. In plant-specific analysis, this PVD was then used as a prior distribution for LOOP frequency, and updated with plant-specific data, the resulting posterior being then used in probabilistic risk assessment (PRA) as the state-of-knowledge distribution of LOOP frequency for that plant. Note the assumptions being used here: that it makes sense to draw a PVD in the first place for the population that we are trying to work with, that the facility we are interested in can be viewed as a member of this population, recognizing that it is characterized by variation in LOOP frequency, and so on. If we accept the basic ideas, then in using this framework, we end up imputing to our plant a distribution for this parameter that reflects our experience as well as the experience of the operating fleet, in both the central tendency of that parameter for our plant and the epistemic distribution of likely values of that parameter for our plant. If we have very little plant-specific data, our state-of-knowledge curve will look like the PVD curve describing the whole plant population.

Therefore, when presented with a set of data of component performance, the most pressing question that arises is can we pool the data, and if not, can we find a curve that fits the variability of the data sufficiently to quantify a prior distribution that adequately represents the performance of the component population.

F-1. POPULATION ANALYSIS STEPS

Figure F-1 represents the basic steps used for population analysis.

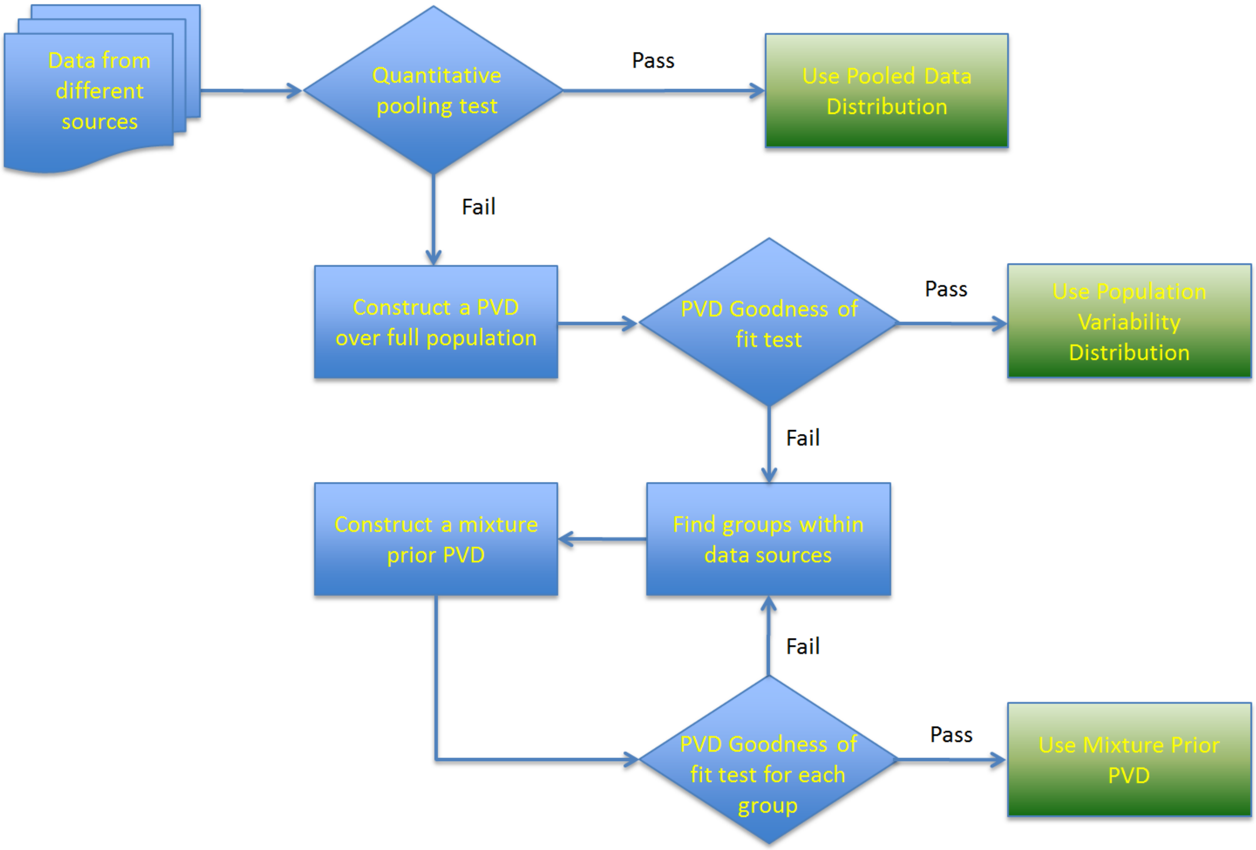

Data from different sources
Quantitative pooling test
Pass
Use Pooled Data Distribution
Fail
Construct a PVD over full population
PVD Goodness of fit test
Pass
Use Population Variability Distribution
Fail
Construct a mixture prior PVD
Find groups within data sources
Fail
PVD Goodness of fit test for each group
Pass
Use Mixture Prior PVD

Figure F-1. Population analysis steps.

It is always best to use a pooled data set if the data set fits a pooled goodness-of-fit test. The first analysis is to take the full population and perform a quantitative pooling test. If the goodness-of-fit test passes for the distribution inferred by the pooled data, then the prior fitting the population is found and can be used to update with the facility component data.

If the pooled data test fails, then a PVD is constructed for the entire population. Hierarchical Bayes is used in this analysis, with the prior parameters used as the first hierarchy (hyperpriors). These hyperprior parameters are started as flat or completely diffuse, ideally chosen to spread out along the entire realm of possibility. Generally, a Markov Chain Monte Carlo (MCMC) program is used with initial guesses for the starting points of the parameters to run the inference until the parameters converge to values that are used as the parameters in the PVD. The parameters may not converge to an answer depending on the degree of variability within the full population data set. If the PVD parameters converge to values, then the results for the PVD inference are used to compare to replicated values for the individual data sets in the same manner as for the pooling test and a goodness-of-fit score is produced. If the goodness-of-fit test passes for this PVD, then the prior fitting the population is found and this PVD can be used to update with the facility component data.

If the full population PVD test fails to find converged values or fails the goodness-of-fit test, then a data source grouping analysis must be performed. This consists of both qualitative and a quantitative analysis. In a qualitative manner, as much information about the data set is determined as possible beyond just the number of failures over time or demands. If data sources are known to be from one manufacturer of component, for instance, that is noted. The same holds for any other pertinent information such as environment used, application, etc. The quantitative part of the analysis is to use a machine learning data mining algorithm to cluster the data into groups by these attributes identified in the qualitative analysis.

The cluster analysis can work on as little as two points of information (covariates) such as failures and time or failures and demands, however, the more covariates that can be provided, the more information can be gleaned from the cluster analysis.

Using the clusters obtained from the grouping analysis an individual PVD analysis is performed on each cluster. If a cluster cannot be fitted with a PVD, then the grouping process can be re-performed with different parameters for the algorithm or by changing the number of covariates (increase or decrease). If clusters are marked as outliers and a PVD cannot be fitted, then a simple Bayesian update on the individual data sources with a Jeffrey's prior will suffice. Once all clusters have a PVD, then they are weighted using a mixture prior for use in updating facility component data. The weights are equivalent to their percentage of the population.

F-1.1 Markov Chain Monte Carlo Sampling

The use of Markov Chain Monte Carlo (MCMC) programs greatly simplifies the calculation of the problems encountered in population variability analysis. Bayesian inference typically involves several integrals in the denominator of the equation and MCMC avoids the need to empirically solve the multidimensional integral. The basic process of MCMC uses a random number to sample directly from the posterior distribution. It then has one "answer." Another random sample is taken, and another, and so forth until an entire set of samples can be used to determine a numerical distribution that represents the posterior distribution.

The basic premise of a Markov chain is that it is constructed such that the chain converges to a joint posterior distribution. The chain uses a sequence of random variables X_0, X_1, X_2, … to sample and create the posterior distribution. The distribution of X_{n+1} only depends on X_n, which is a property of Markov chains. The chain "forgets" its initial state and the next sample builds on the resulting distribution of the prior sample. The Markov function, $f(x_{n+1}|x_n)$, is known as a "transition kernel." Once the distribution is stable from sample to sample (known as "stationary"), samples can be taken to estimate the parameters of interest. Various methods exist to construct the transition kernel. Gibbs sampling, Slice Sampling, and Metropolis-Hastings are a few of the most popular.

F-1.1.1 OpenBUGS and JAGS

OpenBUGS (Open-source Bayesian Updating Using Gibbs Sampling) and JAGS (Just Another Gibbs Sampler), two Bayesian inference MCMC programs, are vetted open-source programs that are good to use for these types of problems. Both programs use the BUGS language and are nearly identical. Any MCMC program capable of Bayesian inference can be used, however, these programs were used for the sample analyses presented here.

The publicly available NASA publication NASA/SP-2009-569, [F-2] Appendix C, provides a tutorial in the basic use of OpenBUGS.

F-1.2 Pooled Data Analysis

The pooled data test uses a non-informative prior to infer parameters over the entire data set. Each data source is duplicated using the pooled data distribution parameters, and compared to the results obtained by updating individually with the non-informed prior. A goodness-of-fit such as a Chi-squared test is applied to determine if the replicated data matches the data using the pooled data distribution.

The Bayesian update rule for a single degree of freedom can be expressed as:

$$\pi_1(\theta|E) = \frac{L(E|\theta)\pi_0(\theta)}{\int L(E|\theta)\pi_0(\theta)d\theta} \qquad \text{(F-1)}$$

where:

E is the evidence

θ is the parameter of interest

$\pi_0(\theta)$ is the prior distribution

$L(E|\theta)$ is the likelihood function

$\pi_1(\theta|E)$ is the posterior distribution (the updated estimate).

The parameter of interest in a pooled data update is the prior distribution $\pi_0(\theta)$ and how well it replicates the posterior distributions of each data source.

Figure F-2 displays the directed acyclic graph of a pooled Bayesian inference of Poisson distributed data (failures over time).

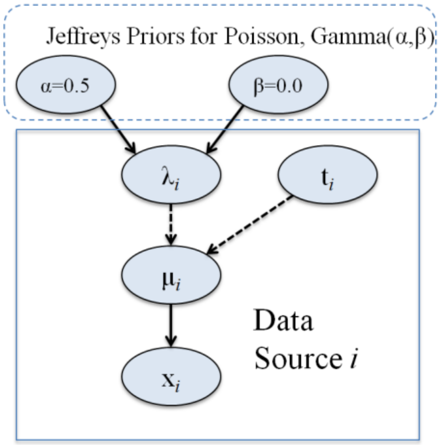


Figure F-2. Bayesian inference for a pooled analysis of failures versus time Poisson analysis.

Parameters α and β are parameters of a Gamma distribution, which is represented by the solid arrows leading to λ. This Gamma distribution starts as a non-informative prior; the Jeffreys prior is commonly used. Updating this model for each data source starting with the Jeffreys prior until the entire population is updated gives a posterior numerical distribution prediction for the entire population. Each data source uses this update to replicate its number of failures. If the replication is nearly or exactly equivalent to the number of failures for the data set, then the goodness-of-fit test passes. If this is the case, then a distribution is fitted to the properties of the numerical distribution (such as mean, percentiles, or standard deviation) and used to update with facility component performance data.

F-1.3 Population Variability Distribution Analysis

It is preferable to pool data when it is appropriate to do so; however, if goodness-of-fit tests prove that the data cannot be pooled, then an attempt is made to model the variability in the sources of the data within the data set. A viable PVD represents the variability from source to source of component data. Unfortunately, there may not exist a simple-looking PVD: the data themselves may belie a simple picture. However, constructing an honest variability distribution is preferable to pooling data that are patently inhomogeneous.

A PVD is a distribution that adequately represents the variance in the data sources of the data set. This is the top level of the hierarchical Bayesian inference required for this type of problem. The PVD distribution is on the input data at one level and the likelihood distribution for individual source outcomes in on a second level, which describes the hierarchy. Recall that the update rule for a single degree of freedom is shown in Equation (F-1).

Another integral is added to the denominator for each degree of freedom, whether that be a second parameter in the likelihood function or a second level of hierarchy.

A Poisson distribution is commonly used when one is examining a rate-based problem (failures experienced over an operating time). The distribution requires inputs for failures (x) and time (t) and will produce a rate (λ). Simply the Poisson is described as "x is distributed as (~) the Poisson of μ," which is equal to λt.

$$x \sim Poisson(\mu), where\ \mu = \lambda t \tag{F-3}$$

Figure F-3 displays the directed acyclic graph of a hierarchical Bayesian inference using Poisson parameters.

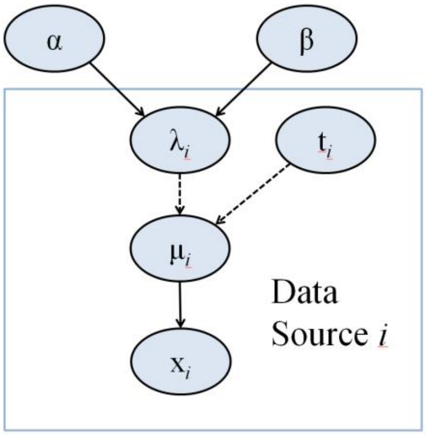


Figure F-3. Hierarchical Bayesian inference using the Poisson parameters.

The hyperpriors, α and β, are parameters of a Gamma distribution, which is represented by the solid arrows leading to λ. This Gamma distribution is the PVD and defines λ for the first hierarchy. The second hierarchy is the Poisson distribution, which provides the posterior for each data source.

Generally, in hierarchical Bayes, if the parameter of interest is denoted as π(θ), then the prior distribution is written as:

$$\pi(\theta) = \int \pi_1(\theta|\varphi)\, \pi_2(\varphi) d\varphi \tag{F-4}$$

where $\pi_1(\theta|\varphi)$ is the first stage prior that represents the population variability in θ for a given value of φ, which is the vector $(\alpha, \beta)^T$.

Further broken down into terms of α and β, the first stage prior is defined as:

$$\pi_1(\lambda) = \iint \pi_0(\lambda|\alpha,\beta)\, \pi_0(\alpha,\beta) d\alpha d\beta \tag{F-5}$$

The hyperpriors, α and β, are not defined as discrete values in the non-pooled inference model. Instead, they are defined as diffuse, or flat values over the breadth of possible values. A Gamma distribution with α and β both equal to zero is a good example of a diffuse prior for use in hierarchical Bayes. Using MCMC, α and β are given starting points and samples are taken until the values converge to the Gamma parameters of the prior for use in the PVD.

F-1.4 Data Source Grouping Analysis

A quantitative tool to help in identifying groups within a population data set is called cluster analysis. Many algorithms have been developed for use in the area of data mining. One such algorithm proposed for use in heterogeneous populations is Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [F-3]. DBSCAN uses the data points in a matrix to determine the groups that are closely packed together (points that have many nearby neighbors) [F-4].

Two parameters are used in DBSCAN, the first is called the Epsilon Neighborhood of a point. This specifies the distance at which to determine if two data points representing a set of covariates are within the same neighborhood.

$$N_{Eps}(p) = \{q \in D | dist(p, q) \leq Eps\} \qquad \text{(F-6)}$$

The second parameter is the Minimum Points, which is the minimum number or points to lie within a neighborhood to determine if the points are within a group.

$$\begin{gathered} p \in N_{Eps}(q) \\ |N_{Eps}(q)| \geq MinPts \end{gathered} \qquad \text{(F-7)}$$

The concept of direct density reachability in the DBSCAN algorithm states that the two points, *p* and *q*, are in the same Epsilon Neighborhood and there are a specified number of points within that neighborhood to call it a cluster. Further, a third point, *o*, is also density reachable with respect to N_{Eps} and *MinPts*. For more information, see [F-3] and [F-4].

DBSCAN is available via many statistical analysis program packages, including the open source and free program R. The data is set into a matrix of covariates. This can be as simple as failures and time or failures and demands. It can also include more covariates such as environmental parameters such as temperature, humidity, manufacturer, etc. However, the data matrix must be entered numerically.

The clusters that DBSCAN provides can be further analyzed in a qualitative manner to determine what caused the data to cluster in that way. Was it an environmental difference? Was it a manufacturer? Was the component used in a different manner? Can data sources be discarded as outliers? These are all questions that cluster analysis can help answer.

An output graph showing the grouped points and outliers is presented in Figure F-4. The data set grouped here consisted of only two covariates, time (years in y-axis) and failures (x-axis). Listing the indices from the matrix for the groups then allows a PVD analysis to be performed for each group and a mixture prior set up for use in updating facility data.

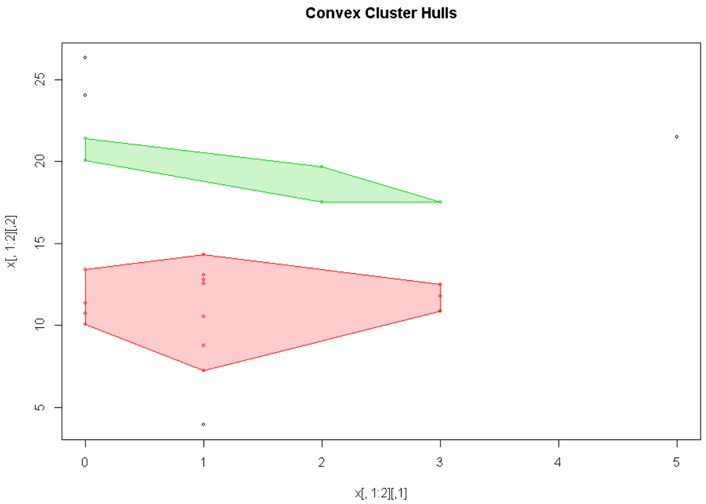


Figure F-4. Convex cluster hulls.

F-1.5 Mixture Priors

A mixture prior uses all the information from the data set in a weighted manner. The prior in the Bayesian inference formula (π_0) is a sum of the component parts that are PVDs of the groups found through cluster analysis.

$$\pi_1(\theta|E) = \frac{L(E|\theta)\pi_0(\theta)}{\int L(E|\theta)\pi_0(\theta)d\theta} \tag{F-8}$$

$$\pi_0 = \sum_n^{i=1} w_i\pi_i + w_{i+1}\pi_{i+1} + \cdots w_n\pi_n \tag{F-9}$$

$$\sum_n^{i=1} w_i + w_{i+1} + \cdots w_n = 1.0 \tag{F-10}$$

where:

w is the weight of the group as a ratio of the entire population. If 10 indices of the matrix out of 100 are in the group, then the weight will be 0.10. Weights must sum to 1.0.

π_0 is the mixture prior distribution for use in the Bayesian inference formula to find the posterior update of the facility component performance.

F-2. EXAMPLES OF POPULATION ANALYSIS

The following examples use rate-based data based on time used and failures experienced. Following these examples are another set of examples using demand-based data.

F-2.1 Example of a Population Pooling Test

Data for a component based on time used and failures experienced are presented in Table F-1. This could be any failure mode for any component. This particular data set is from a textbook example in [F-5].

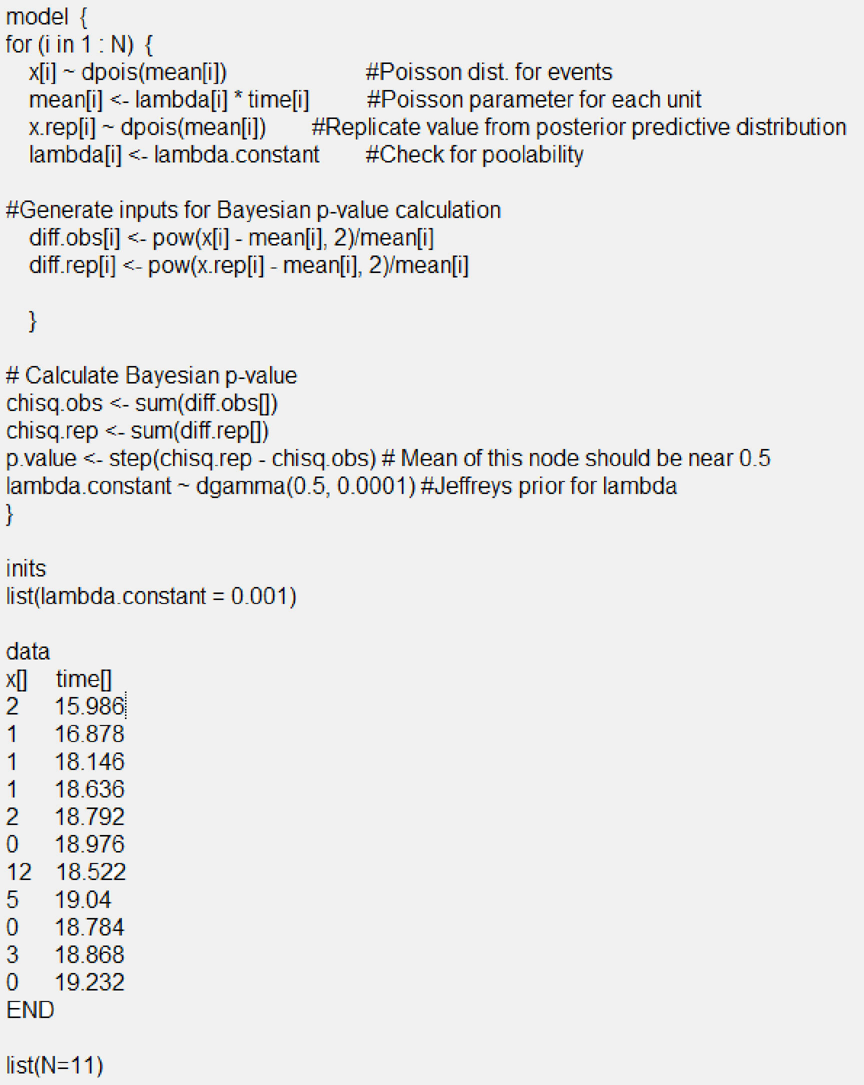
Table F-1. Component failure rate data.

Source	Failures	Exposure Time (years)
1	2	15.986
2	1	16.878
3	1	18.146
4	1	18.636
5	2	18.792
6	0	18.979
7	12	18.522
8	5	19.040
9	0	18.784
10	3	18.868
11	0	19.232

This example uses the data presented in Table F-1 and is for a set of data with failure counts over time. It uses OpenBUGS as the analysis tool. The OpenBUGS script is shown in Figure F-5.

Key parameters in this model are:

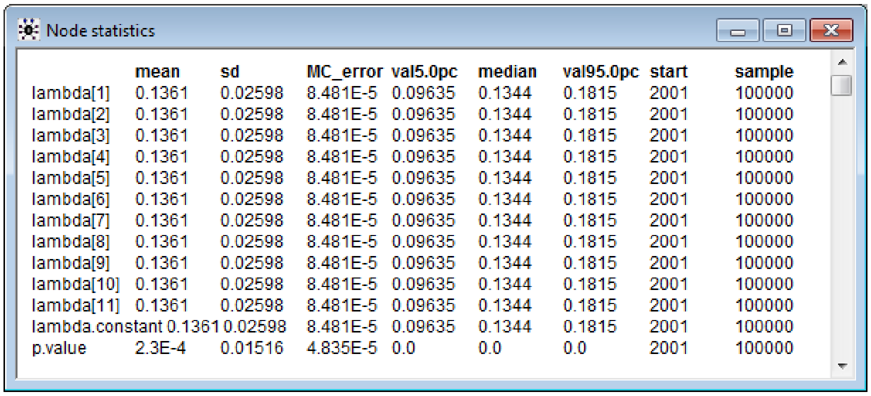
- x: Failures for each source listed in the data
- time: Time in years for each source listed in the data
- mean: The Poisson parameter = lambda * time
- lambda: The rate in failures per year
- lambda.constant: The defined rate which is inferred upon the lambda for each data set in the pooled test
- x.rep: The replicated Poisson result for x for each source to use in the Chi-squared comparison.



```
model {
for (i in 1 : N) {
   x[i] ~ dpois(mean[i])                #Poisson dist. for events
   mean[i] <- lambda[i] * time[i]       #Poisson parameter for each unit
   x.rep[i] ~ dpois(mean[i])     #Replicate value from posterior predictive distribution
   lambda[i] <- lambda.constant         #Check for poolability

#Generate inputs for Bayesian p-value calculation
   diff.obs[i] <- pow(x[i] - mean[i], 2)/mean[i]
   diff.rep[i] <- pow(x.rep[i] - mean[i], 2)/mean[i]

   }

# Calculate Bayesian p-value
chisq.obs <- sum(diff.obs[])
chisq.rep <- sum(diff.rep[])
p.value <- step(chisq.rep - chisq.obs) # Mean of this node should be near 0.5
lambda.constant ~ dgamma(0.5, 0.0001) #Jeffreys prior for lambda
}

inits
list(lambda.constant = 0.001)

data
x[]   time[]
2     15.986
1     16.878
1     18.146
1     18.636
2     18.792
0     18.976
12    18.522
5     19.04
0     18.784
3     18.868
0     19.232
END

list(N=11)
```

Figure F-5. OpenBUGS model to check for poolability of rate-based data.

The results of running the model with 2,000 samples to “burn-in” and converge, with 100,000 samples taken for results are shown in Figure F-6.



```
                mean     sd       MC_error val5.0pc median   val95.0pc start   sample
lambda[1]       0.1361   0.02598  8.481E-5 0.09635  0.1344   0.1815    2001    100000
lambda[2]       0.1361   0.02598  8.481E-5 0.09635  0.1344   0.1815    2001    100000
lambda[3]       0.1361   0.02598  8.481E-5 0.09635  0.1344   0.1815    2001    100000
lambda[4]       0.1361   0.02598  8.481E-5 0.09635  0.1344   0.1815    2001    100000
lambda[5]       0.1361   0.02598  8.481E-5 0.09635  0.1344   0.1815    2001    100000
lambda[6]       0.1361   0.02598  8.481E-5 0.09635  0.1344   0.1815    2001    100000
lambda[7]       0.1361   0.02598  8.481E-5 0.09635  0.1344   0.1815    2001    100000
lambda[8]       0.1361   0.02598  8.481E-5 0.09635  0.1344   0.1815    2001    100000
lambda[9]       0.1361   0.02598  8.481E-5 0.09635  0.1344   0.1815    2001    100000
lambda[10]      0.1361   0.02598  8.481E-5 0.09635  0.1344   0.1815    2001    100000
lambda[11]      0.1361   0.02598  8.481E-5 0.09635  0.1344   0.1815    2001    100000
lambda.constant 0.1361 0.02598    8.481E-5 0.09635  0.1344   0.1815    2001    100000
p.value         2.3E-4   0.01516  4.835E-5 0.0      0.0      0.0       2001    100000
```

Figure F-6. Rate-based pooled model results.

Note that the individual sources are using the constant lambda (lambda.constant) that has defined values for its Gamma distribution based on the Jeffreys prior. The lambda.constant is set with an initial value of 0.001 in the MCMC in order toto initialize the model, but it converges to the 0.1361 per year value based on the inference from the data set. The replicated Poisson result from each source is compared to the pooled result to determine the P-value (p.value), which is the measure of model fit to the data. A perfect P-value would be 0.5, with values closer to zero or one indicating a poor fit. In this case, the P-value of 2.3E-04 (on the bottom left of Figure F-6) indicates that the pooled model is not a good fit to the data.

F-2.2 Example of Full Population Variability Distribution Estimation

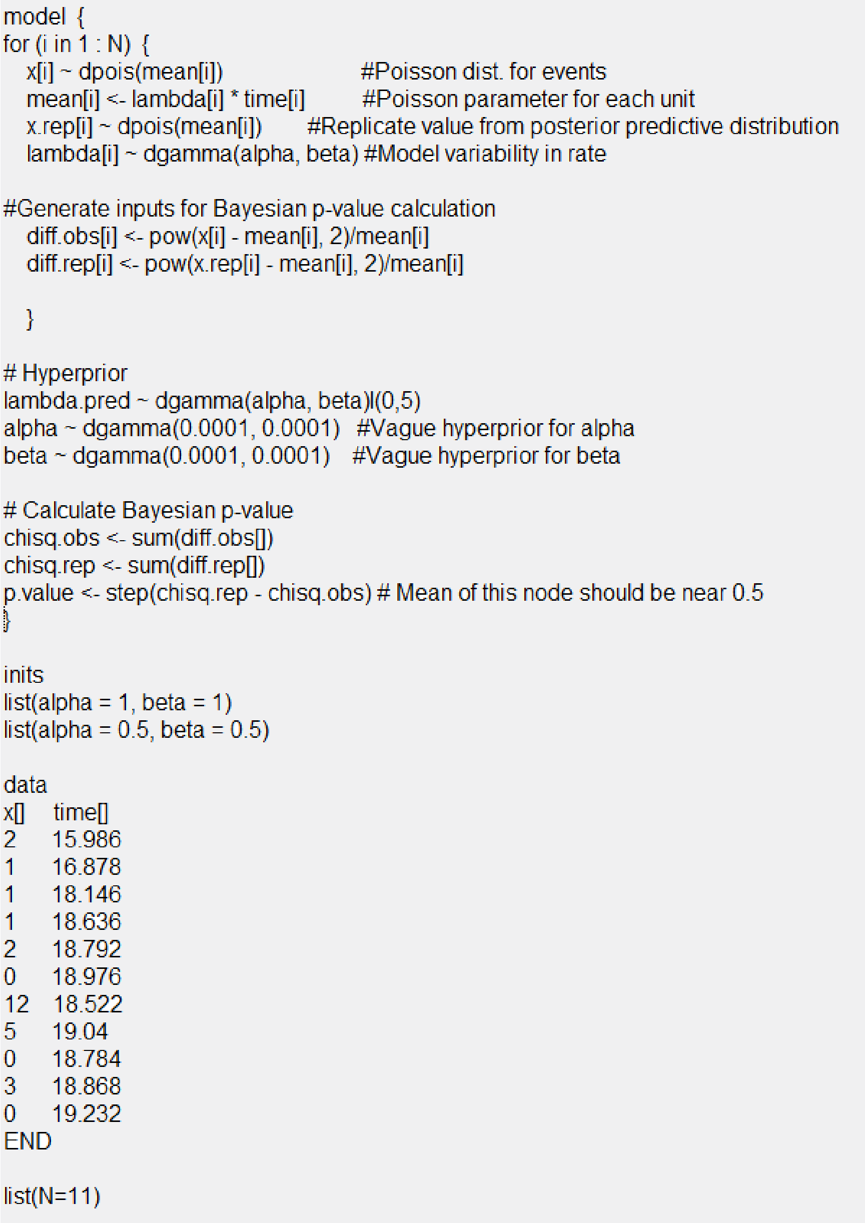
For the data set in Table F-1, the pooling test indicates that the data should not be pooled. The next step is to see if a PVD can be fit to the data by using a hyperprior distribution to represent the variation of data sources as discussed above.

A hyperprior that is commonly used with Poisson distributed data is a Gamma distribution. Other distributions can be utilized, such as lognormal. The reader is directed to [F-2] and [F-5] for further guidance. A Gamma distribution was used for this analysis.

The hyperprior should not influence the model; rather, the model should drive the parameters of the hyperprior distribution to the values that fit the data. For this reason, the parameters of the Gamma distribution are in turn represented hyperpriors of diffuse Gamma values that produce as flat a distribution over the realm of values as possible, given an initial starting point, and then the MCMC uses the Bayesian inference to drive the Gamma parameters to converged values. This example uses the data presented in Table F-1 and is for a set of data with failure counts over time. It uses OpenBUGS as the analysis tool. The OpenBUGS script is presented in Figure F-7.

Key parameters in this model are:

- x: Failures for each source listed in the data
- time: Time in years for each source listed in the data
- mean: The Poisson parameter = lambda * time
- lambda: The rate in failures per year
- lambda.constant: The defined rate which is inferred upon the lambda for each data set in the pooled test
- x.rep: The replicated Poisson result for x for each source to use in the Chi-squared comparison.



```
model {
for (i in 1 : N) {
   x[i] ~ dpois(mean[i])                  #Poisson dist. for events
   mean[i] <- lambda[i] * time[i]         #Poisson parameter for each unit
   x.rep[i] ~ dpois(mean[i])      #Replicate value from posterior predictive distribution
   lambda[i] ~ dgamma(alpha, beta) #Model variability in rate

#Generate inputs for Bayesian p-value calculation
   diff.obs[i] <- pow(x[i] - mean[i], 2)/mean[i]
   diff.rep[i] <- pow(x.rep[i] - mean[i], 2)/mean[i]

   }

# Hyperprior
lambda.pred ~ dgamma(alpha, beta)I(0,5)
alpha ~ dgamma(0.0001, 0.0001)  #Vague hyperprior for alpha
beta ~ dgamma(0.0001, 0.0001)   #Vague hyperprior for beta

# Calculate Bayesian p-value
chisq.obs <- sum(diff.obs[])
chisq.rep <- sum(diff.rep[])
p.value <- step(chisq.rep - chisq.obs) # Mean of this node should be near 0.5
}

inits
list(alpha = 1, beta = 1)
list(alpha = 0.5, beta = 0.5)

data
x[]    time[]
2      15.986
1      16.878
1      18.146
1      18.636
2      18.792
0      18.976
12     18.522
5      19.04
0      18.784
3      18.868
0      19.232
END

list(N=11)
```

Figure F-7. Hierarchical Bayes BUGS Language Model for source population variability in rate-based data.

F-2.2.1 Checking the Model for Convergence

Any MCMC program must converge before having confidence in the samples taken for results. Qualitative checks for convergence include looking at a graph of the histories of the key parameters. Running this model with two initial values for alpha and beta allows the check of convergence in these parameters so that there is confidence in the samples taken for results. In addition to the qualitative checks, a more quantitative test for convergence used in the OpenBUGS program is the Brooks-Gelman-Rubin statistic (BGR). Convergence is represented graphically in BGR by an R-value that is consistently at 1.0 and a B-value and W-value that are equivalent values to each other.

Figure F-8 shows the BGR test results for alpha and beta parameters in the model. Note that the number of iterations is half the numbers of samples since there are two chains compiled in the model. Alpha and beta R-values appear to settle solidly at 1 by 30,000 iterations with the other two parameters tracking together. This particular example takes longer to converge for a "picky" analyst than most do. For models that do not converge, the R-value will typically not settle on 1.0, and will wander significantly away from this value.

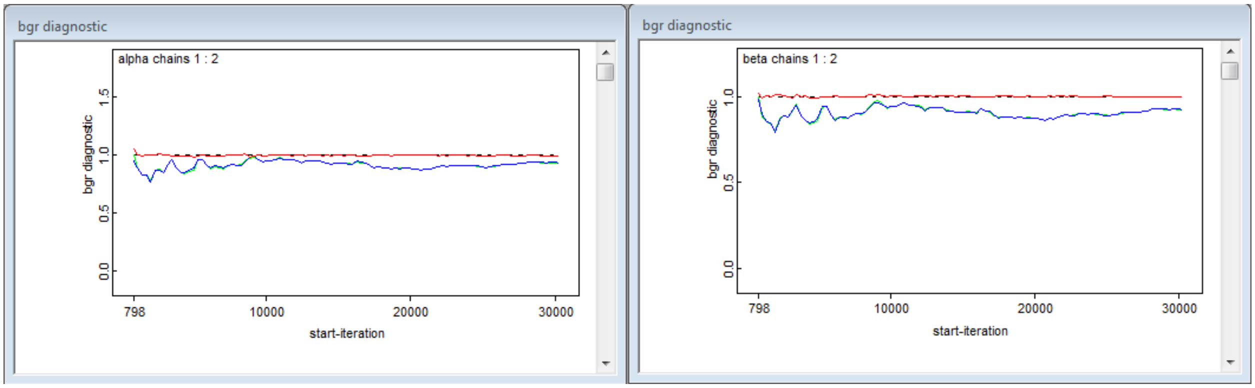


Figure F-8. BGR diagnostic test for convergence of gamma parameters.

To use as samples for the results, 100,000 iterations are run beyond the 60,000 burn-ins. A quick check of the BGR diagnostic for the duration of the sampling, shown in Figure F-9, does not show any significant events to question the validity of the calculations.

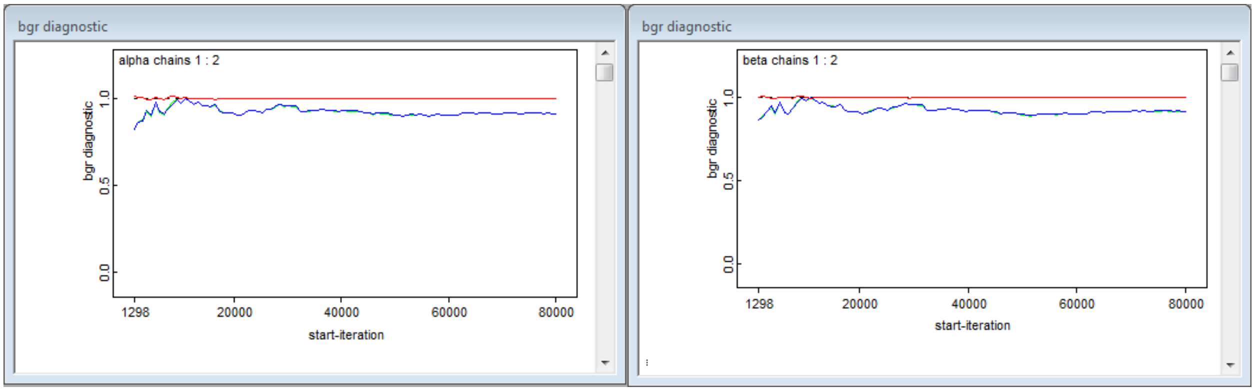
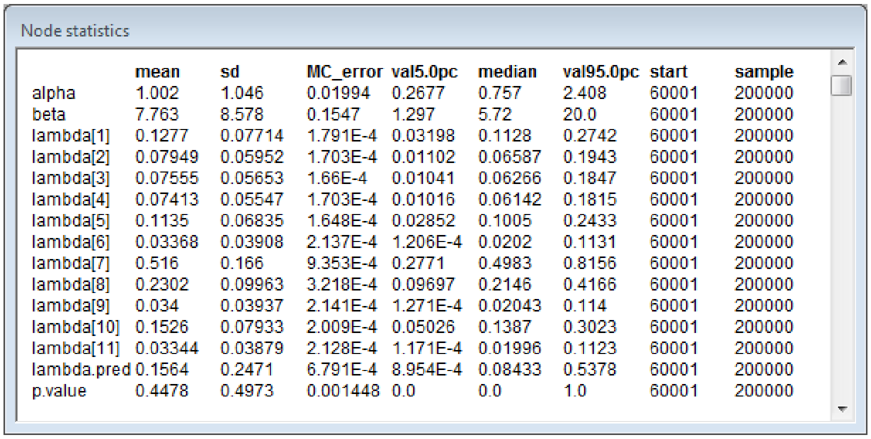


Figure F-9. BGR diagnostic for full sampling of gamma parameters.

F-2.2.2 Results of the Rate-Based Population Variability Analysis

Results of the analysis shown in Figure F-10 provide the following insights:

- The Chi-Square Bayesian P-value goodness-of-fit parameter is at 0.44, which is close to the ideal value of 0.5 and indicates high confidence in the predicted posterior results.
- The predicted posterior distribution that would be used for the PRA failure distribution for this component is a Gamma with alpha = 1.00 and beta = 7.76. Its mean is 1.56E-01 per year with a 5th percentile value of 8.95E-04 and a 95th percentile value of 5.38E-01. The Gamma(1.00, 7.76) distribution would be valid for use to update facility component performance until the next overall population update is performed.
- This prior, used in the Poisson model, provides the estimations of the lambda for the mean, 5th percentile and 95th percentile for each source and a predicted lambda as well.



Node statistics

	mean	sd	MC_error	val5.0pc	median	val95.0pc	start	sample
alpha	1.002	1.046	0.01994	0.2677	0.757	2.408	60001	200000
beta	7.763	8.578	0.1547	1.297	5.72	20.0	60001	200000
lambda[1]	0.1277	0.07714	1.791E-4	0.03198	0.1128	0.2742	60001	200000
lambda[2]	0.07949	0.05952	1.703E-4	0.01102	0.06587	0.1943	60001	200000
lambda[3]	0.07555	0.05653	1.66E-4	0.01041	0.06266	0.1847	60001	200000
lambda[4]	0.07413	0.05547	1.703E-4	0.01016	0.06142	0.1815	60001	200000
lambda[5]	0.1135	0.06835	1.648E-4	0.02852	0.1005	0.2433	60001	200000
lambda[6]	0.03368	0.03908	2.137E-4	1.206E-4	0.0202	0.1131	60001	200000
lambda[7]	0.516	0.166	9.353E-4	0.2771	0.4983	0.8156	60001	200000
lambda[8]	0.2302	0.09963	3.218E-4	0.09697	0.2146	0.4166	60001	200000
lambda[9]	0.034	0.03937	2.141E-4	1.271E-4	0.02043	0.114	60001	200000
lambda[10]	0.1526	0.07933	2.009E-4	0.05026	0.1387	0.3023	60001	200000
lambda[11]	0.03344	0.03879	2.128E-4	1.171E-4	0.01996	0.1123	60001	200000
lambda.pred	0.1564	0.2471	6.791E-4	8.954E-4	0.08433	0.5378	60001	200000
p.value	0.4478	0.4973	0.001448	0.0	0.0	1.0	60001	200000

Figure F-10. Rate-based results with gamma hyperprior.

A comparison of the lambdas and the predicted lambda as presented in Figure F-11 shows where the predicted value lies within the 5th to 95th percentile ranges of the sources.

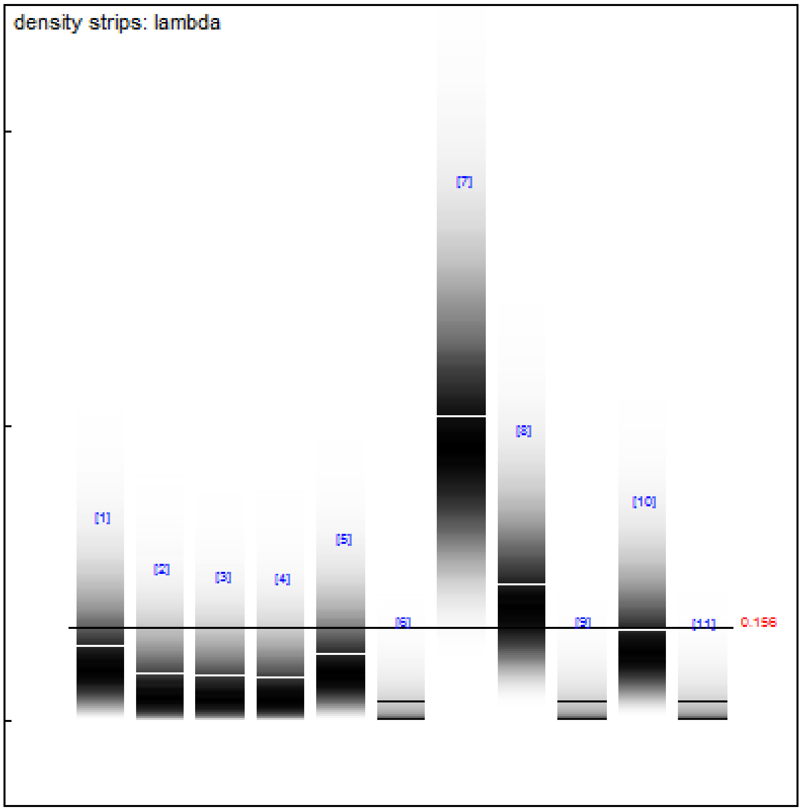


Figure F-11. Comparison of rate-based data source results.

F-2.3 Cluster Analysis Example

Data for a cluster analysis are presented in Table F-2. This failure rate data is from nuclear power plant loss of offsite power (LOOP) records, and deals with a population in which the parameters alpha and beta present difficulty converging to values.

Table F-2. Component failure rate data for a cluster analysis.

Source	Failures	Time (years)
1	1	13.054
2	1	12.77
3	1	7.22
4	1	3.944
5	1	10.548
6	0	10.704
7	0	24.0
8	1	8.76
9	3	11.79
10	2	17.5
11	0	20.03
12	0	13.39
13	5	21.5
14	0	10.075
15	0	26.32
16	1	12.54
17	3	17.5
18	1	14.3
19	3	10.89
20	3	12.5
21	0	21.38
22	2	19.65
23	0	11.34

There are multiple references online for using DBSCAN via R. The first step is to use the k-Nearest Neighbor distance plot to determine the knee in the graph. This is generally the best starting point for the Epsilon (Eps) parameter. It can be seen from Figure F-12 that the knee is approximately at 2.3 NN distance.

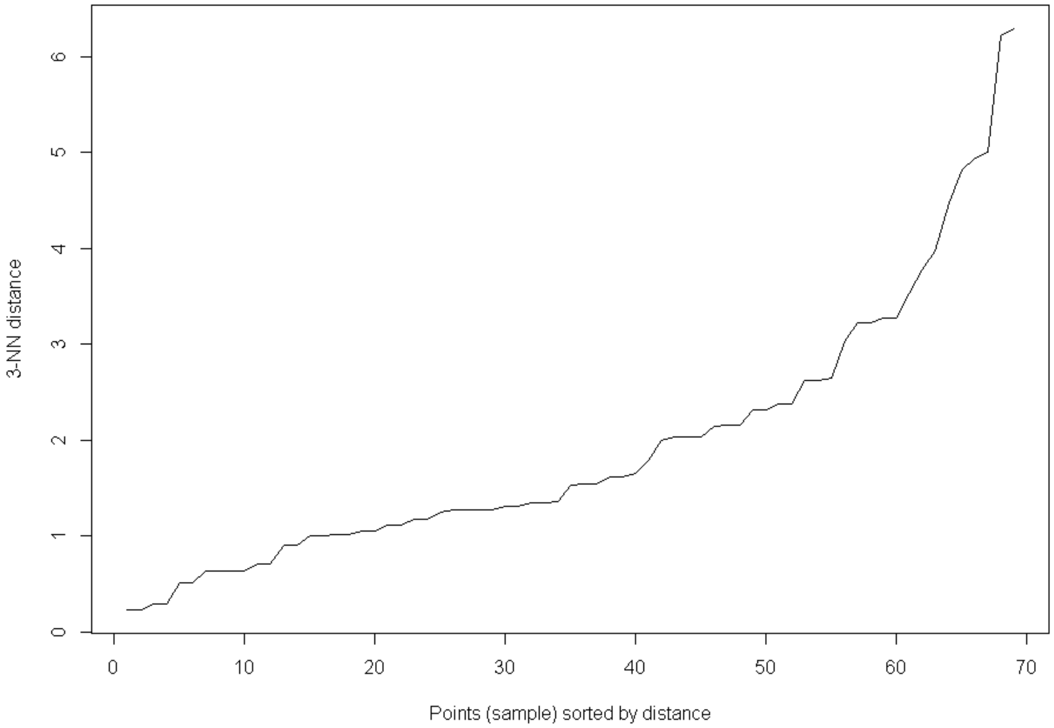


Figure F-12. k-nearest neighbor distance plot.

The next step is to run DBSCAN using the Eps and the Minimum Points (MinPts) parameters. Determining the MinPts parameter is less of a science than the Eps parameter. If one chooses too high of a value for MinPts, the algorithm will not find any clusters; too low of a value and it will find too many clusters. The default for MinPts is 5. For this data set, there are only 23 sources, and using a MinPts of 5 only generates one cluster of 14 and 9 outliers. Using a MinPts value of 3 produces two clusters and 4 outliers: 14 in Cluster 1 and 5 in Cluster 2. This is shown graphically in Figure F-13, with the two clusters and points in the clusters. The x-axis is the failures covariate and the y-axis is the time covariate.

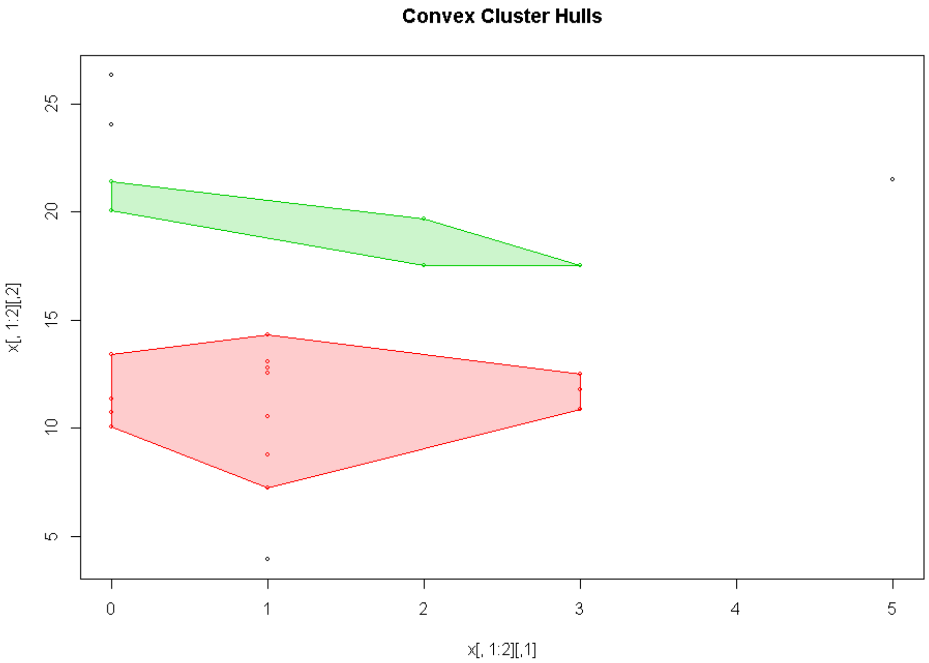


Figure F-13. Convex cluster hulls.

The indices of the data set that belong in each cluster can be extracted in R through the where() command. These are the source numbers in Table F-3.

Table F-3. Clusters with corresponding sources.

Cluster	Data Set Index (Source, from Table G-4)
0 (outliers)	4, 7, 13, 15
1	1, 2, 3, 5, 6, 8, 9, 12, 14, 16, 18, 19, 20, 23
2	10, 11, 17, 21, 22

The clusters can now be run through a PVD analysis to fit individual distributions. If the outliers fail to find a PVD as a group, then they can be re-ran as a cluster analysis as their own data set of four or find an update with a Jeffreys prior to use as individual distributions in the mixture prior.

F-2.4 Use of Mixture Prior Example

This example will use the data presented and grouped by the previous cluster analysis. There are three clusters identified in the cluster analysis, however, one of the clusters is identified as an outlier. Outliers are not related to each other and become their own group when setting up mixture priors. So in essence, there are six clusters to use in setting up the mixture prior.

A weight must be assigned for each cluster. The weight is equivalent to the proportion of the sources in the group to the overall number of sources in the population. Table F-4 summarizes the information.

Table F-4. Cluster weighting for mixture prior example.

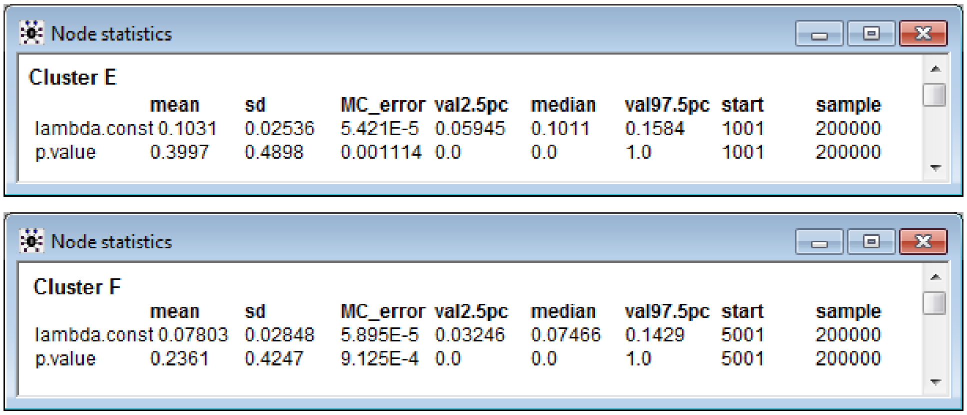
Cluster	Weight	Source	Failures	Time (years)
A	0.0435	4	1	3.944
B	0.0435	7	0	24.0

C	0.0435	13	5	21.5
D	0.0435	15	0	26.32
E	0.6090	1	1	13.054
		2	1	12.77
		3	1	7.22
		5	1	10.548
		6	0	10.704
		8	1	8.76
		9	3	11.79
		12	0	13.39
		14	0	10.075
		16	1	12.54
		18	1	14.3
		19	3	10.89
		20	3	12.5
		23	0	11.34
F	0.2170	10	2	17.5
		11	0	20.03
		17	3	17.5
		21	0	21.38
		22	2	19.65

The next step is to see if the individual clusters are poolable. If they are not, then tighter set of parameters are required when using the data mining in order to produce smaller clusters.

The poolability OpenBUGS model for rate-based data is used as previously described.

The results of the p.values for this analysis as shown in Figure F-14 show that it is reasonable to pool the data for each cluster since neither Cluster E nor Cluster F have a p-value that is close to zero or 1 as was the case when testing the entire population in the prior rate example. Ideal fit would be 0.5, however, the range between 0.2 and 0.8 can be considered acceptable.



```
Cluster E
            mean     sd       MC_error val2.5pc median  val97.5pc start  sample
lambda.const 0.1031  0.02536  5.421E-5 0.05945  0.1011  0.1584    1001   200000
p.value     0.3997   0.4898   0.001114 0.0      0.0     1.0       1001   200000
```

```
Cluster F
            mean     sd       MC_error val2.5pc median  val97.5pc start  sample
lambda.const 0.07803 0.02848  5.895E-5 0.03246  0.07466 0.1429    5001   200000
p.value     0.2361   0.4247   9.125E-4 0.0      0.0     1.0       5001   200000
```

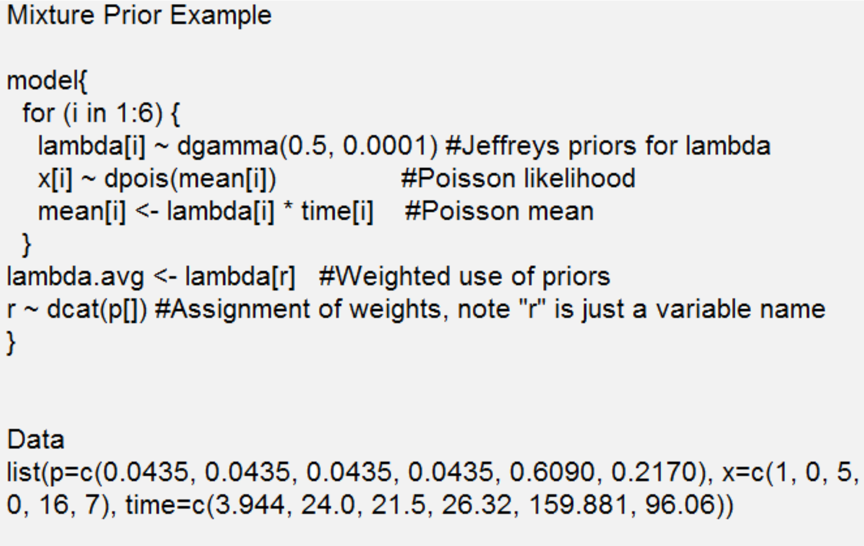
Figure F-14. Pooling result with P values for mixture prior example.

The next step is to pool the data in the clusters with multiple sources by summing the failures and time components. The mixture prior data now consists of that shown in Table F-5.

Table F-5. Clusters with corresponding sources for mixture prior example.

Cluster	Weight	Source	Failures	Time (years)
A	0.0435	4	1	3.944
B	0.0435	7	0	24.0
C	0.0435	13	5	21.5
D	0.0435	15	0	26.32
E	0.6090	1, 2, 3, 5, 6, 8, 9, 12, 14, 16, 18, 19, 20, 23	16	159.881
F	0.2170	10, 11, 17, 21, 22	7	96.06

The model shown in Figure F-15 uses the mixture priors by weighting each cluster's data input through the use of a categorical distribution, of which all components (weights) of the distribution must sum to one.

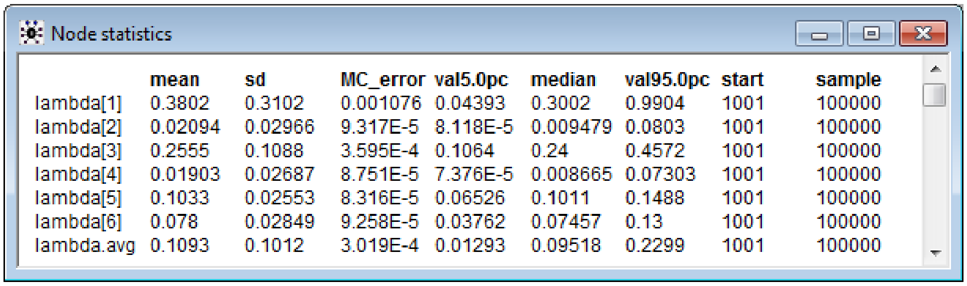


```
Mixture Prior Example

model{
  for (i in 1:6) {
    lambda[i] ~ dgamma(0.5, 0.0001) #Jeffreys priors for lambda
    x[i] ~ dpois(mean[i])               #Poisson likelihood
    mean[i] <- lambda[i] * time[i]    #Poisson mean
  }
lambda.avg <- lambda[r]   #Weighted use of priors
r ~ dcat(p[]) #Assignment of weights, note "r" is just a variable name
}


Data
list(p=c(0.0435, 0.0435, 0.0435, 0.0435, 0.6090, 0.2170), x=c(1, 0, 5,
0, 16, 7), time=c(3.944, 24.0, 21.5, 26.32, 159.881, 96.06))
```

Figure F-15. Mixture prior example.

Note the lack of a requirement to use initial values to start the MCMC. In the case where each sub-population (cluster) is poolable, the program can usually generate its own initial values and burn-in quickly. For this example, 1,000 samples were used.

The results are shown in Figure F-16. Lambda[1] through lambda[6] are the results for each of the clusters. The node "lambda.avg" is a multi-modal distribution of which its mean, percentiles, and/or standard deviation would be used to fit a traditional distribution for use in industry failure data.



```
            mean     sd       MC_error  val5.0pc  median    val95.0pc  start  sample
lambda[1]   0.3802   0.3102   0.001076  0.04393   0.3002    0.9904     1001   100000
lambda[2]   0.02094  0.02966  9.317E-5  8.118E-5  0.009479  0.0803     1001   100000
lambda[3]   0.2555   0.1088   3.595E-4  0.1064    0.24      0.4572     1001   100000
lambda[4]   0.01903  0.02687  8.751E-5  7.376E-5  0.008665  0.07303    1001   100000
lambda[5]   0.1033   0.02553  8.316E-5  0.06526   0.1011    0.1488     1001   100000
lambda[6]   0.078    0.02849  9.258E-5  0.03762   0.07457   0.13       1001   100000
lambda.avg  0.1093   0.1012   3.019E-4  0.01293   0.09518   0.2299     1001   100000
```

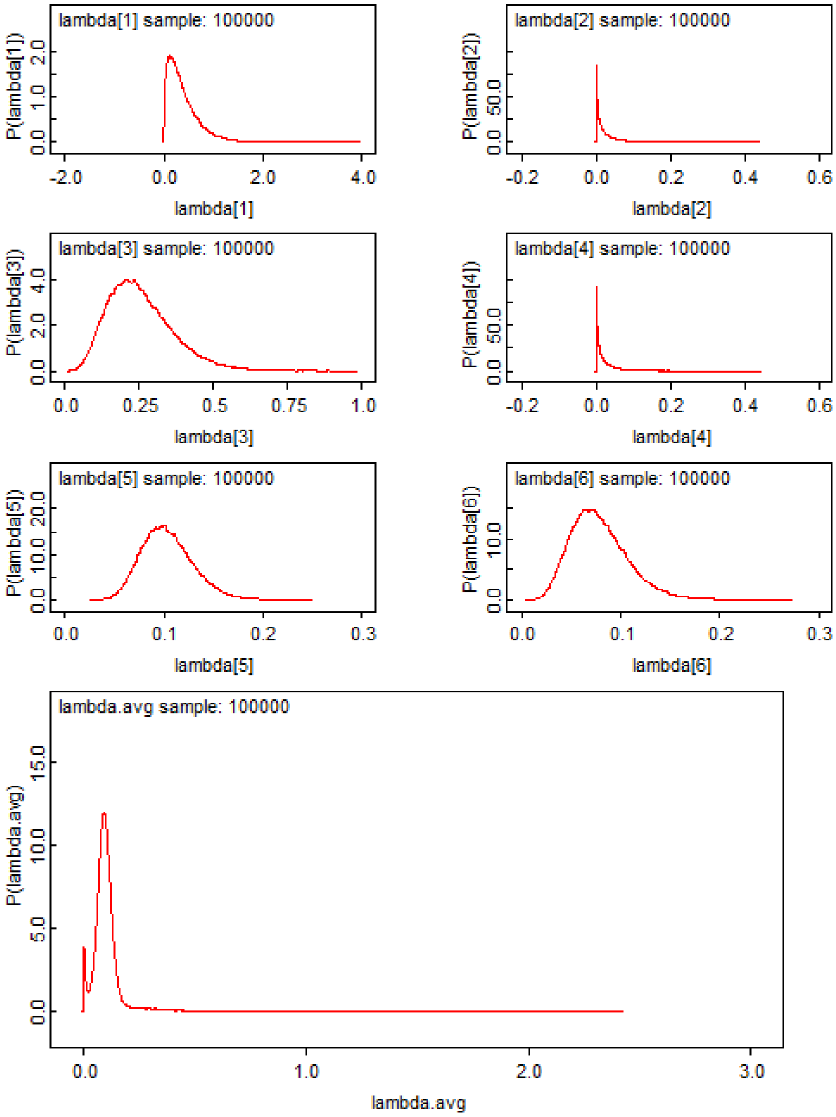


Figure F-16. Convergence test result for mixture prior example.

F-2.5 Demand-Based Component Population Variability Example

Data for a component based on demands and failures experienced are presented in Table F-6. This could be any failure mode for any component and differences in manufacturer or operating conditions should be kept in mind in case the set cannot be pooled and a PVD cannot be fit. This particular data is from a textbook example in [F-5].

Table F-6. Component demand-based failure data.

Source	Failures	Demands
1	0	140
2	0	130
3	0	130
4	1	130
5	2	100
6	3	185
7	3	175
8	4	167
9	5	151
10	10	150

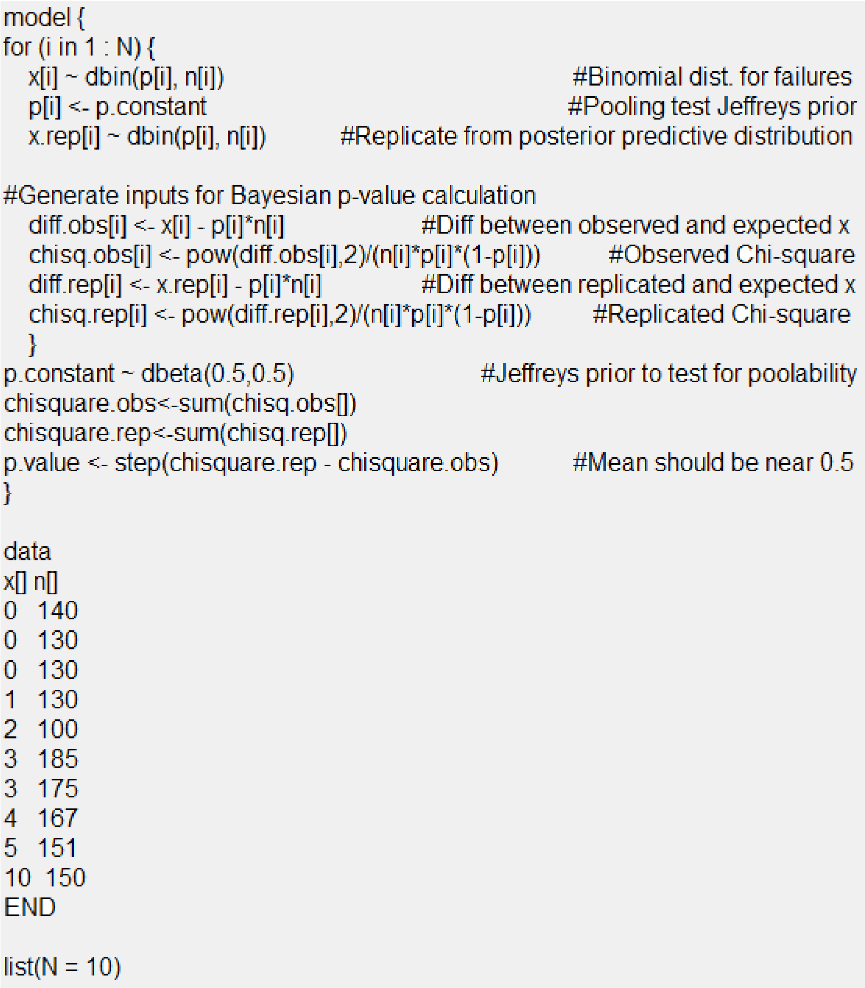
F-2.5.1 Testing for Pool-ability of the Data

A first qualitative look at the data shows what appears to be an outlier in Source 10. This is good to note in case the data cannot be pooled or a PVD cannot be applied.

The first quantitative analysis should be to see if the data can be pooled. An OpenBUGS model to check for pooling applicability is presented in Figure F-17. Note that any MCMC program capable of Bayesian inference will work and that OpenBUGS is used here as an example. The goodness of fit test in this model is a Chi-squared test of the constant probabilities for each source determined by the model versus the replicated results using the posterior predictive distribution which in this case is a Jeffreys prior which adds minimal influence on the data.

Key parameters in this model are:

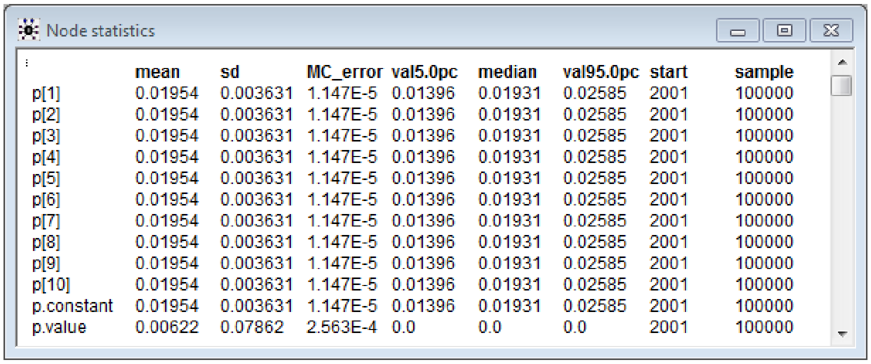
- x: Failures for each source listed in the data
- n: Number of demands for each source listed in the data
- N: Number of sources
- p: The probability of failure per demand
- x.rep: The replicated Binomial result for x from the posterior predictive distribution.



```
model {
for (i in 1 : N) {
   x[i] ~ dbin(p[i], n[i])                                   #Binomial dist. for failures
   p[i] <- p.constant                                        #Pooling test Jeffreys prior
   x.rep[i] ~ dbin(p[i], n[i])        #Replicate from posterior predictive distribution

#Generate inputs for Bayesian p-value calculation
   diff.obs[i] <- x[i] - p[i]*n[i]              #Diff between observed and expected x
   chisq.obs[i] <- pow(diff.obs[i],2)/(n[i]*p[i]*(1-p[i]))         #Observed Chi-square
   diff.rep[i] <- x.rep[i] - p[i]*n[i]          #Diff between replicated and expected x
   chisq.rep[i] <- pow(diff.rep[i],2)/(n[i]*p[i]*(1-p[i]))       #Replicated Chi-square
   }
p.constant ~ dbeta(0.5,0.5)                   #Jeffreys prior to test for poolability
chisquare.obs<-sum(chisq.obs[])
chisquare.rep<-sum(chisq.rep[])
p.value <- step(chisquare.rep - chisquare.obs)          #Mean should be near 0.5
}

data
x[] n[]
0  140
0  130
0  130
1  130
2  100
3  185
3  175
4  167
5  151
10  150
END

list(N = 10)
```

Figure F-17. OpenBUGS model to check for poolability of demand-based data.

The results of running the model with 2,000 samples to "burn-in" and converge, with 100,000 samples taken for results are shown in Figure F-18.



```
            mean      sd        MC_error  val5.0pc  median   val95.0pc  start   sample
p[1]        0.01954   0.003631  1.147E-5  0.01396   0.01931  0.02585    2001    100000
p[2]        0.01954   0.003631  1.147E-5  0.01396   0.01931  0.02585    2001    100000
p[3]        0.01954   0.003631  1.147E-5  0.01396   0.01931  0.02585    2001    100000
p[4]        0.01954   0.003631  1.147E-5  0.01396   0.01931  0.02585    2001    100000
p[5]        0.01954   0.003631  1.147E-5  0.01396   0.01931  0.02585    2001    100000
p[6]        0.01954   0.003631  1.147E-5  0.01396   0.01931  0.02585    2001    100000
p[7]        0.01954   0.003631  1.147E-5  0.01396   0.01931  0.02585    2001    100000
p[8]        0.01954   0.003631  1.147E-5  0.01396   0.01931  0.02585    2001    100000
p[9]        0.01954   0.003631  1.147E-5  0.01396   0.01931  0.02585    2001    100000
p[10]       0.01954   0.003631  1.147E-5  0.01396   0.01931  0.02585    2001    100000
p.constant  0.01954   0.003631  1.147E-5  0.01396   0.01931  0.02585    2001    100000
p.value     0.00622   0.07862   2.563E-4  0.0       0.0      0.0        2001    100000
```

Figure F-18. Pooling result with P values for demand-based example.

Note that the individual sources are using the constant probability which has defined values for its Beta distribution based on the Jeffreys prior. The p.constant does not require an initial set value in this particular model because OpenBUGS is able to generate the initial value on its own. The p.constant converges to the 1.95E-02 failures/demand based on the inference from the data set. The replicated

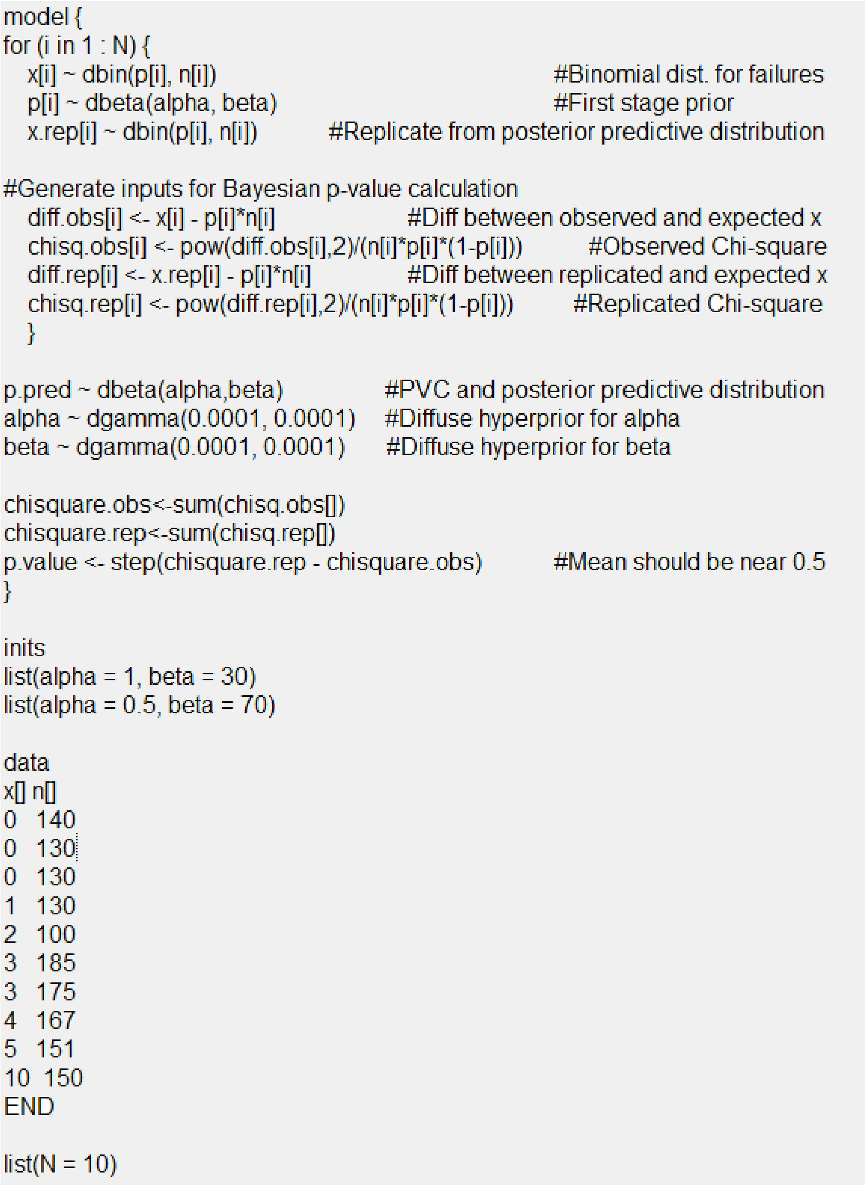
Binomial result from each source is compared to the pooled result to determine the P-value (p.value) as with the previous rate-based example. The P-value of 6.22E-03 indicates that the pooled model is not a good fit to the data.

F-2.5.2 Demand-Based Heirarchical Bayes for a Population Variability Distribution Estimation

Now that it is determined that the data set cannot be pooled, the next step is to use a hyperprior to represent the variability of the data from source to source and check to see if this model can fit the data.

A hyperprior is selected to attempt to fit the data source variability. The most common hyperprior distributions used for a Binomial model are the Beta, but other distributions such as the Lognormal can be used. A Beta distribution is used in this example. The hyperprior should not influence the model; rather the model should drive the parameters of the hyperprior distribution to the values that fit the data. For this reason, the parameters of the Beta distribution are in turn represented by diffuse values, given an initial starting point and then the MCMC uses Bayesian inference to drive the Beta parameters to converged values.

The model in Figure F-19 uses the Beta prior to infer upon the probability in the Binomial distribution in a similar manner that the fixed distribution of the Jeffreys Beta prior was used to test for poolability of the data. The Gamma hyperprior representing the Beta parameters are the PVD that helps to fit the data. Predicted performance from the population is given by the posterior Beta distribution (p.pred) with the parameters alpha and beta. Note that the prior being used here allows the parameters to vary based upon the distributions they are defined by so the MCMC will have to run enough samples to converge both the alpha and the beta parameters.



```
model {
for (i in 1 : N) {
   x[i] ~ dbin(p[i], n[i])                              #Binomial dist. for failures
   p[i] ~ dbeta(alpha, beta)                            #First stage prior
   x.rep[i] ~ dbin(p[i], n[i])         #Replicate from posterior predictive distribution

#Generate inputs for Bayesian p-value calculation
   diff.obs[i] <- x[i] - p[i]*n[i]               #Diff between observed and expected x
   chisq.obs[i] <- pow(diff.obs[i],2)/(n[i]*p[i]*(1-p[i]))         #Observed Chi-square
   diff.rep[i] <- x.rep[i] - p[i]*n[i]           #Diff between replicated and expected x
   chisq.rep[i] <- pow(diff.rep[i],2)/(n[i]*p[i]*(1-p[i]))        #Replicated Chi-square
   }

p.pred ~ dbeta(alpha,beta)              #PVC and posterior predictive distribution
alpha ~ dgamma(0.0001, 0.0001)    #Diffuse hyperprior for alpha
beta ~ dgamma(0.0001, 0.0001)     #Diffuse hyperprior for beta

chisquare.obs<-sum(chisq.obs[])
chisquare.rep<-sum(chisq.rep[])
p.value <- step(chisquare.rep - chisquare.obs)         #Mean should be near 0.5
}

inits
list(alpha = 1, beta = 30)
list(alpha = 0.5, beta = 70)

data
x[] n[]
0   140
0   130
0   130
1   130
2   100
3   185
3   175
4   167
5   151
10  150
END

list(N = 10)
```

Figure F-19. Hierarchical Bayes model for source population variability in demand-based data.

F-2.5.3 Convergence of the Hyperprior Parameters

The BGR diagnostic for the alpha and beta parameters in Figure F-20 shows some wild fluctuations prior to approximately 40,000 iterations (80,000 samples) at which point the R-value settles along the 1.0 line and the other two measures track with each other. Another BGR is performed to make sure that nothing out of the ordinary happened during the sampling for analysis. Figure F-21 shows slight bumps away from exactly 1.0, as viewed around 60,000 iterations of each parameter. This is not out of the ordinary and does not indicate divergence. An example of divergence would be continued behavior such as noted prior to 35,000 iterations.

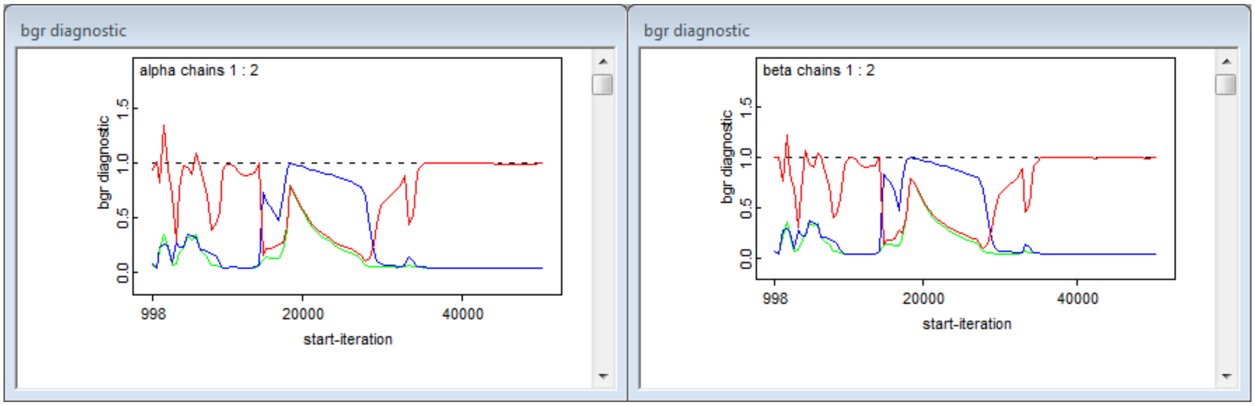


Figure F-20. BGR diagnostic for convergence of beta parameters.

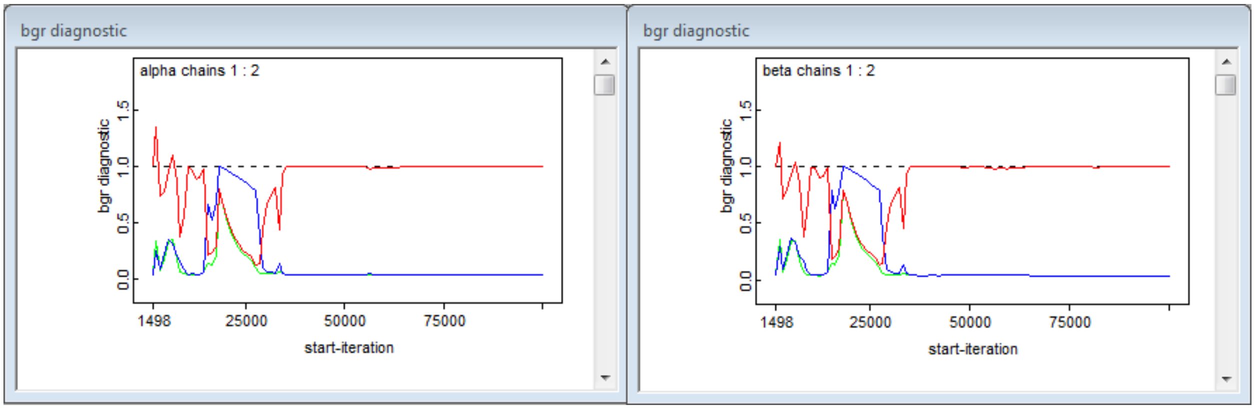
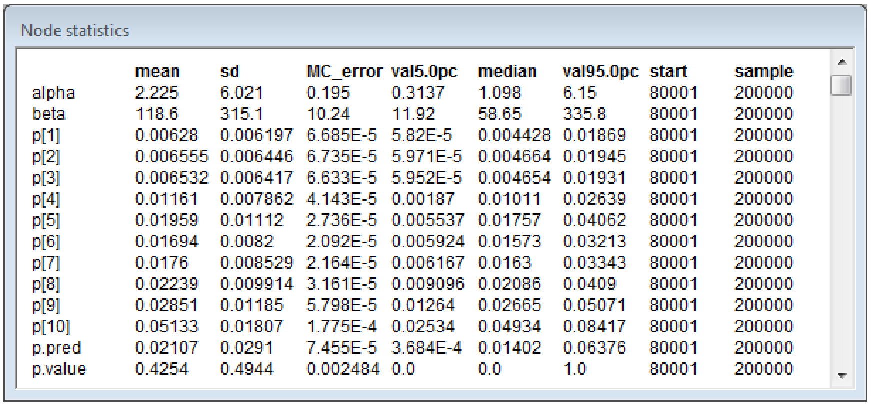


Figure F-21. BGR diagnostic for full sampling of beta parameters.

F-2.5.4 Results of the Demand-based Population Variability Analysis

Results of the analysis are shown in Figure F-22 and provide the following insights:

- The predicted posterior distribution that would be used for PRA failure data for this component is a Beta with alpha = 2.225 and beta = 118.6. Its mean is 2.11E-02 per demand, with a 5th percentile value of 3.68E-04 and a 95th percentile value of 6.38E-02.
- This prior, used in the Binomial model, provides the estimations of the probability of failure per demand for the mean, 5th percentile and 95th percentile for each source and a predicted probability.
- The P-value goodness-of-fit parameter is at 0.43, which is close to the ideal value of 0.5 and indicates high confidence in the predicted posterior distribution results.



Node statistics

	mean	sd	MC_error	val5.0pc	median	val95.0pc	start	sample
alpha	2.225	6.021	0.195	0.3137	1.098	6.15	80001	200000
beta	118.6	315.1	10.24	11.92	58.65	335.8	80001	200000
p[1]	0.00628	0.006197	6.685E-5	5.82E-5	0.004428	0.01869	80001	200000
p[2]	0.006555	0.006446	6.735E-5	5.971E-5	0.004664	0.01945	80001	200000
p[3]	0.006532	0.006417	6.633E-5	5.952E-5	0.004654	0.01931	80001	200000
p[4]	0.01161	0.007862	4.143E-5	0.00187	0.01011	0.02639	80001	200000
p[5]	0.01959	0.01112	2.736E-5	0.005537	0.01757	0.04062	80001	200000
p[6]	0.01694	0.0082	2.092E-5	0.005924	0.01573	0.03213	80001	200000
p[7]	0.0176	0.008529	2.164E-5	0.006167	0.0163	0.03343	80001	200000
p[8]	0.02239	0.009914	3.161E-5	0.009096	0.02086	0.0409	80001	200000
p[9]	0.02851	0.01185	5.798E-5	0.01264	0.02665	0.05071	80001	200000
p[10]	0.05133	0.01807	1.775E-4	0.02534	0.04934	0.08417	80001	200000
p.pred	0.02107	0.0291	7.455E-5	3.684E-4	0.01402	0.06376	80001	200000
p.value	0.4254	0.4944	0.002484	0.0	0.0	1.0	80001	200000

Figure F-22. Demand-based results with beta prior.

A comparison of the probabilities (p) and the predicted probability presented in Figure F-23 shows where the predicted value lies within the 5th to 95th percentile ranges of the sources.

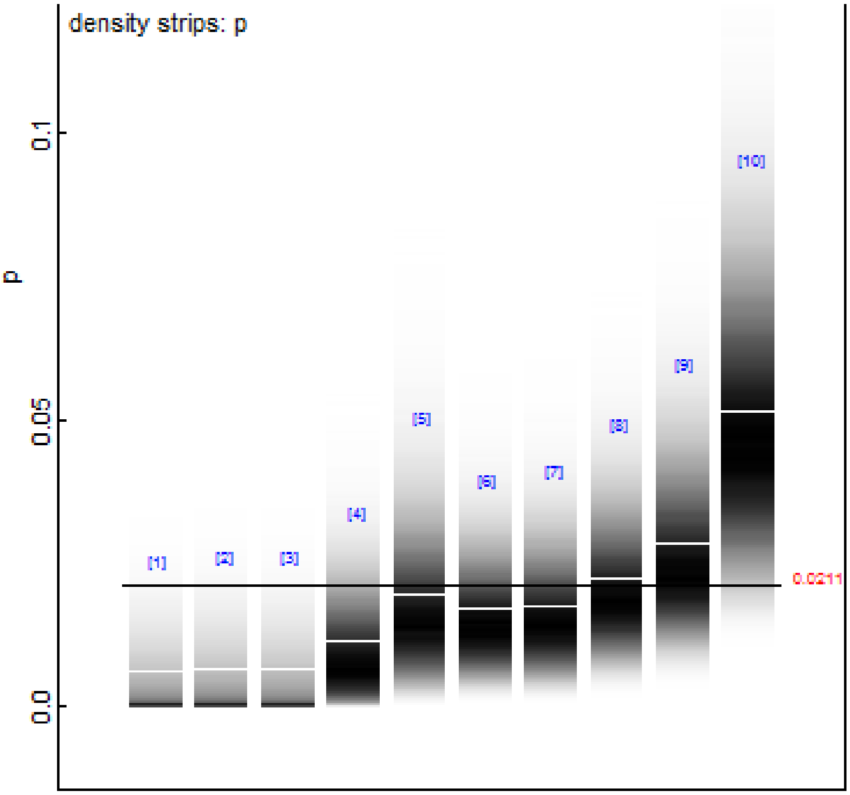


Figure F-23. Comparison of demand-based source results.

F-2.5.5 Pitfalls of MCMC and Selection of Hyperpriors

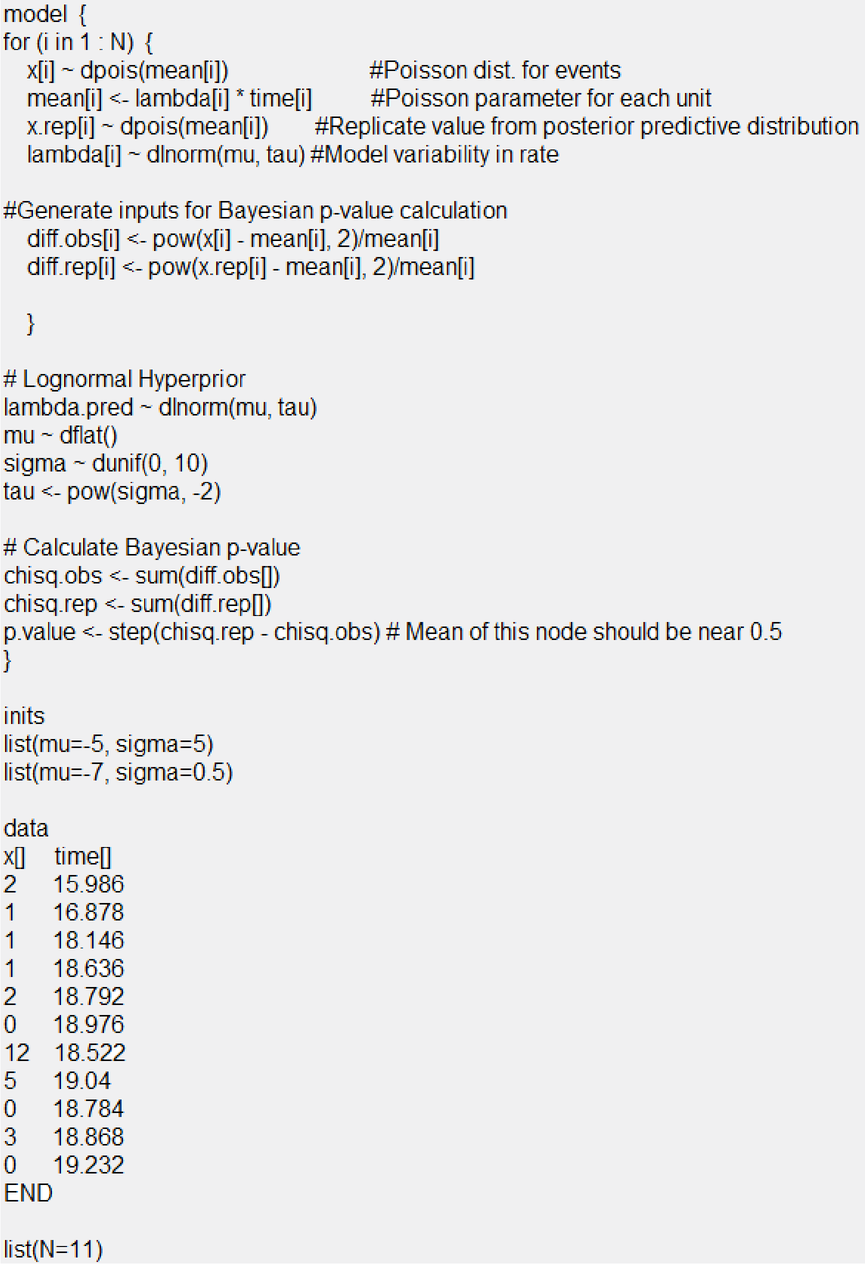
Convergence Issues

In the detailed demand-based data example used above, the alpha and beta variables in the Beta distribution were slow to converge. Generally, it is best to run as many samples as required to attain converged samples for measure. However, if the variables continue to refuse to converge it sometimes helps to reparameterize the distribution in terms such as mean and variance. If secondary parameters are used, then take the samples for measure after they converge and use the secondary parameters to attain the primary ones for use in the PVD. For more information on this topic, see [F-6]. If running more samples and reparameterization does not work, then group analysis must be performed on the population using DBSCAN or another cluster algorithm, individual PVD analysis performed for each group, and a mixture prior set up for use as the current state of knowledge.

Choosing Adequate Hyperpriors

MCMC programs use random “picks” of a simulation across the breadth of the posterior distribution. For distributions with long tails this can present problems where the mean can be in the tail, sometimes even beyond the 95th percentile. Care must be taken by the analyst to choose a prior that will not only cause the posterior to fit the data, but will also produce logical results. An example using the rate-based data set follows.

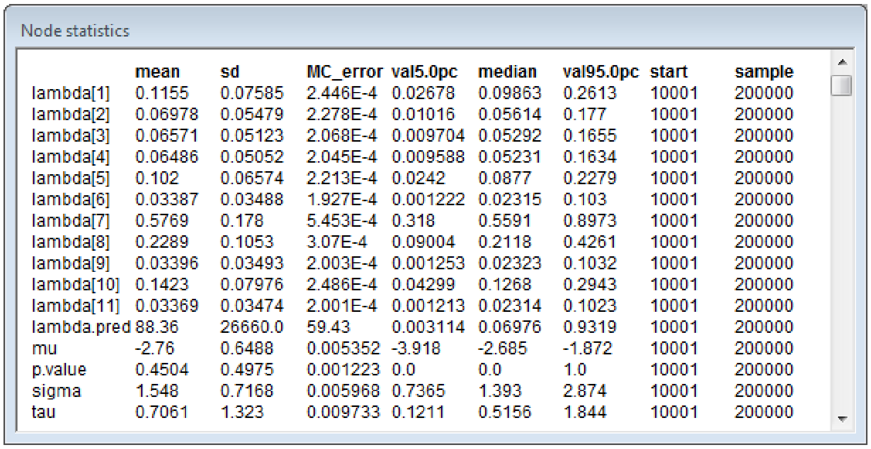
Another prior that is popular to use with Poisson data is the lognormal. An OpenBUGS script using the lognormal as the hyperprior along with the sample data from Table F-1 is shown in Figure F-24.



```
model {
for (i in 1 : N) {
   x[i] ~ dpois(mean[i])                    #Poisson dist. for events
   mean[i] <- lambda[i] * time[i]         #Poisson parameter for each unit
   x.rep[i] ~ dpois(mean[i])       #Replicate value from posterior predictive distribution
   lambda[i] ~ dlnorm(mu, tau) #Model variability in rate

#Generate inputs for Bayesian p-value calculation
   diff.obs[i] <- pow(x[i] - mean[i], 2)/mean[i]
   diff.rep[i] <- pow(x.rep[i] - mean[i], 2)/mean[i]

   }

# Lognormal Hyperprior
lambda.pred ~ dlnorm(mu, tau)
mu ~ dflat()
sigma ~ dunif(0, 10)
tau <- pow(sigma, -2)

# Calculate Bayesian p-value
chisq.obs <- sum(diff.obs[])
chisq.rep <- sum(diff.rep[])
p.value <- step(chisq.rep - chisq.obs) # Mean of this node should be near 0.5
}

inits
list(mu=-5, sigma=5)
list(mu=-7, sigma=0.5)

data
x[]   time[]
2     15.986
1     16.878
1     18.146
1     18.636
2     18.792
0     18.976
12    18.522
5     19.04
0     18.784
3     18.868
0     19.232
END

list(N=11)
```

Figure F-24. Poisson model with lognormal prior.

This model's parameters for the lognormal PVD converge much more quickly. By 10,000 iterations the BGR R-value is solidly at 1.0 and the other two are tracking each other. Running the model for an additional 100,000 iterations gives us 200,000 samples, the same number as was used in the Gamma distribution hyperprior. This produces the results shown in Figure F-25.



	mean	sd	MC_error	val5.0pc	median	val95.0pc	start	sample
lambda[1]	0.1155	0.07585	2.446E-4	0.02678	0.09863	0.2613	10001	200000
lambda[2]	0.06978	0.05479	2.278E-4	0.01016	0.05614	0.177	10001	200000
lambda[3]	0.06571	0.05123	2.068E-4	0.009704	0.05292	0.1655	10001	200000
lambda[4]	0.06486	0.05052	2.045E-4	0.009588	0.05231	0.1634	10001	200000
lambda[5]	0.102	0.06574	2.213E-4	0.0242	0.0877	0.2279	10001	200000
lambda[6]	0.03387	0.03488	1.927E-4	0.001222	0.02315	0.103	10001	200000
lambda[7]	0.5769	0.178	5.453E-4	0.318	0.5591	0.8973	10001	200000
lambda[8]	0.2289	0.1053	3.07E-4	0.09004	0.2118	0.4261	10001	200000
lambda[9]	0.03396	0.03493	2.003E-4	0.001253	0.02323	0.1032	10001	200000
lambda[10]	0.1423	0.07976	2.486E-4	0.04299	0.1268	0.2943	10001	200000
lambda[11]	0.03369	0.03474	2.001E-4	0.001213	0.02314	0.1023	10001	200000
lambda.pred	88.36	26660.0	59.43	0.003114	0.06976	0.9319	10001	200000
mu	-2.76	0.6488	0.005352	-3.918	-2.685	-1.872	10001	200000
p.value	0.4504	0.4975	0.001223	0.0	0.0	1.0	10001	200000
sigma	1.548	0.7168	0.005968	0.7365	1.393	2.874	10001	200000
tau	0.7061	1.323	0.009733	0.1211	0.5156	1.844	10001	200000

Figure F-25. Rate-based results with lognormal hyperprior.

The Bayesian P-value of 0.45 shows that this model replicates the data as well as the Gamma hyperprior model did. A review of the lambdas for the data set shows that the means are close to the ones calculated for the Gamma model. However, a study of the PVD (lambda.pred) shows that the result for the mean (88.36 failures/year) is extreme and well beyond the 95th percentile of 0.93 failures/year. The mean lies in the heavy tail due to the MCMC picking some extreme values in the tail.

This is an anomaly where the goodness-of-fit measure says that the lognormal model replicates the data just as well as the Gamma model, yet the predictive posterior distribution's mean is not logical based on the most extreme case in the data set (lambda[7]) having a 95th percentile result of 0.90 failures/year. This is also intuitively a "wrong" answer because a qualitative look at the data tells us that there is very little chance of 88 failures in a year. Figure F-26 displays the full probability density function and the section near zero using just the predicted μ and σ. This illustrates a sharp peak very close to zero and a long tail.

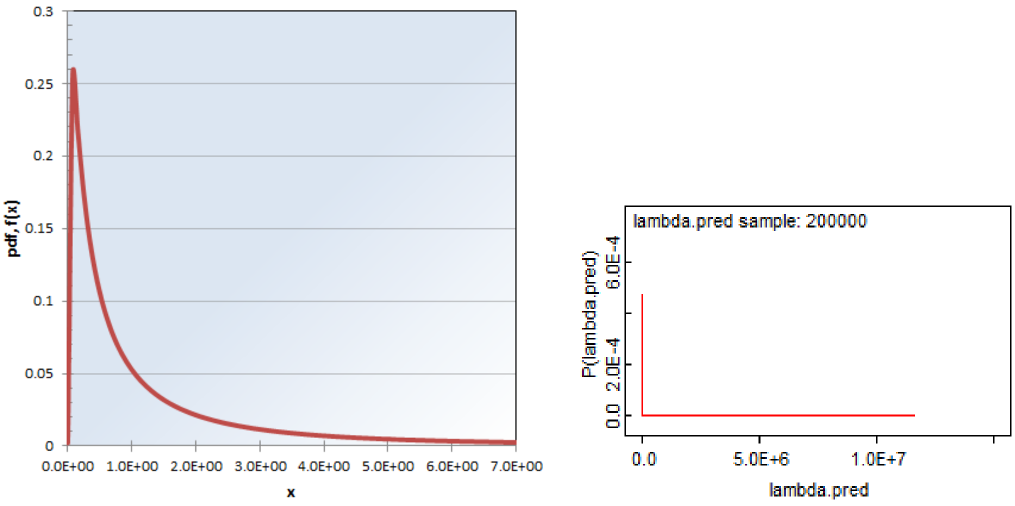
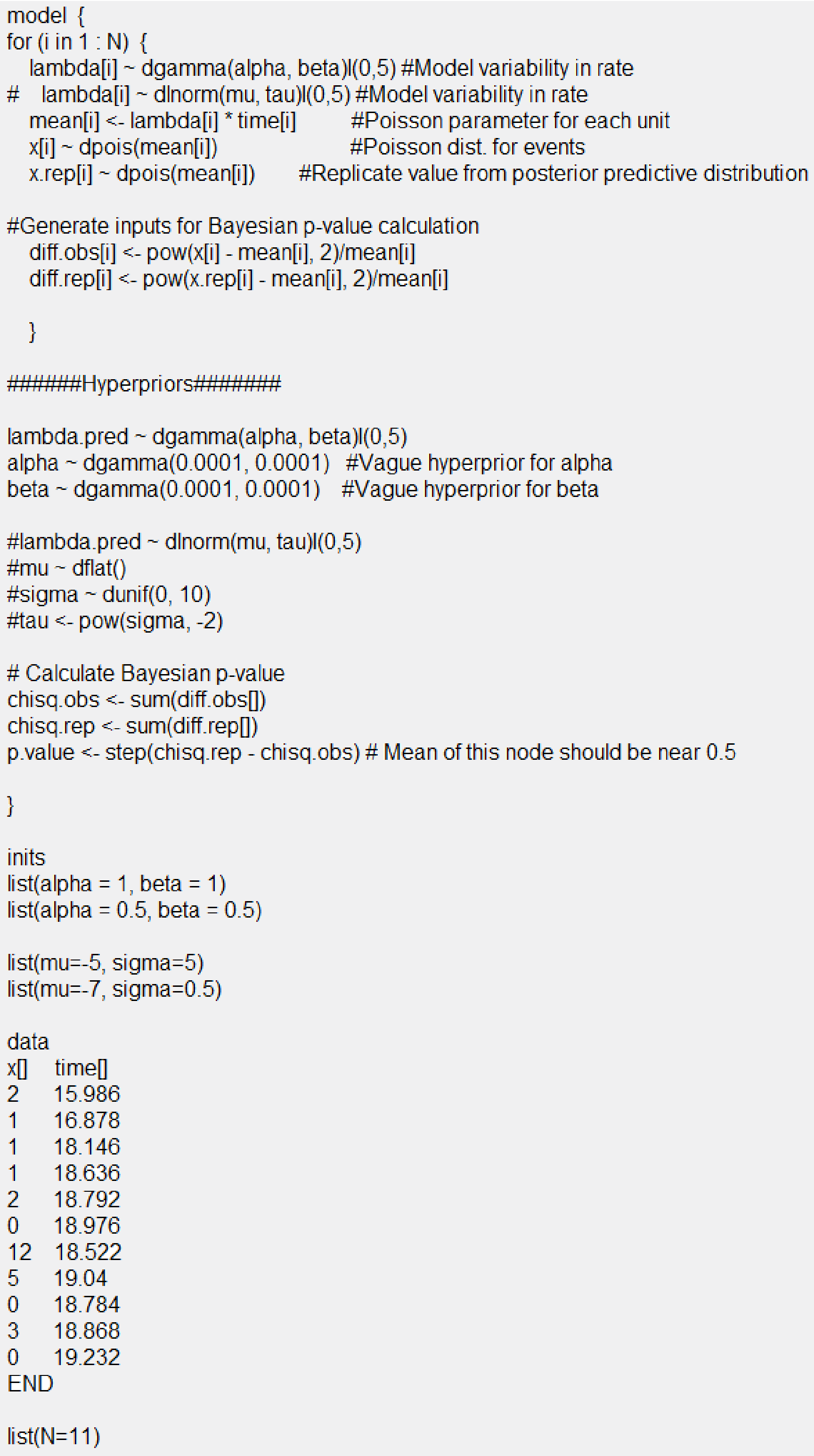


Figure F-26. Probability density function with lognormal hyperprior example.

Further analysis of the two priors can be performed by truncating the Gamma and lognormal priors using the OpenBUGS interval command of “I(x,y)” where x is the lower number and y is the highest number in the results to consider. The OpenBUGS script used for comparison of the two hyperpriors is shown in Figure F-27. Note that the interval command is placed inline and behind the distribution text. The use of “#” comments out the hyperprior not currently in use.



```
model {
for (i in 1 : N) {
   lambda[i] ~ dgamma(alpha, beta)I(0,5) #Model variability in rate
#    lambda[i] ~ dlnorm(mu, tau)I(0,5) #Model variability in rate
   mean[i] <- lambda[i] * time[i]         #Poisson parameter for each unit
   x[i] ~ dpois(mean[i])                  #Poisson dist. for events
   x.rep[i] ~ dpois(mean[i])       #Replicate value from posterior predictive distribution

#Generate inputs for Bayesian p-value calculation
   diff.obs[i] <- pow(x[i] - mean[i], 2)/mean[i]
   diff.rep[i] <- pow(x.rep[i] - mean[i], 2)/mean[i]

   }

######Hyperpriors#######

lambda.pred ~ dgamma(alpha, beta)I(0,5)
alpha ~ dgamma(0.0001, 0.0001)   #Vague hyperprior for alpha
beta ~ dgamma(0.0001, 0.0001)    #Vague hyperprior for beta

#lambda.pred ~ dlnorm(mu, tau)I(0,5)
#mu ~ dflat()
#sigma ~ dunif(0, 10)
#tau <- pow(sigma, -2)

# Calculate Bayesian p-value
chisq.obs <- sum(diff.obs[])
chisq.rep <- sum(diff.rep[])
p.value <- step(chisq.rep - chisq.obs) # Mean of this node should be near 0.5

}

inits
list(alpha = 1, beta = 1)
list(alpha = 0.5, beta = 0.5)

list(mu=-5, sigma=5)
list(mu=-7, sigma=0.5)

data
x[]   time[]
2     15.986
1     16.878
1     18.146
1     18.636
2     18.792
0     18.976
12    18.522
5     19.04
0     18.784
3     18.868
0     19.232
END

list(N=11)
```

Figure F-27. Hyperprior comparison model.

This script was run for truncations from (0,1) to (0,5) to discover the behavior of each of these hyperpriors as more of their complete distribution is used in predicting the rate of the varied population. A chart of the results is presented in Figure F-28.

A few insights from the results:

- The 95th percentile diverges, which shows the effect of the much larger tail of the lognormal.
- The 50th percentile (median) is flat for both, with a small difference between the two. This explains the equally good replication of the data in the Bayesian Chi-squared test.
- The lognormal mean starts correlated with the median in relationship to the Gamma at truncation (0,1), but then it is affected by the tail as the truncation increases, eventually reaching the 88.36 prediction at full use of the distribution.
- The Gamma mean has very little movement between the (0,5) truncation and the use of the full distribution.

The take-away is that even though both hyperpriors "fit" the data in the Bayesian P-value replication of the model, this sort of analysis points out the better of the two priors to use for this particular set of data when used as a PVD for predicting future performance since its mean converges to its full distribution value within an intuitively reasonable numbers of failures per year truncation.

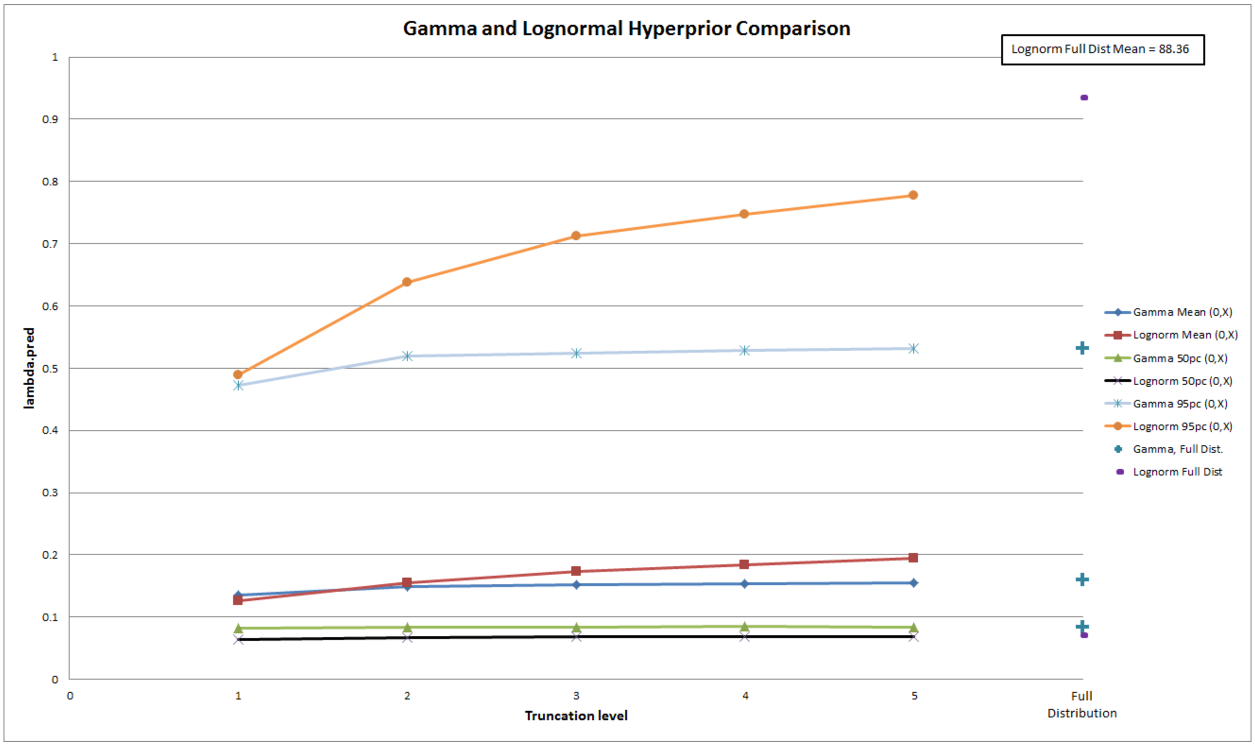


Figure F-28. Hyperpriors comparison.

F-3. PRECISION LIMITATIONS OF NUMERICAL PROGRAMS

Returning to the rate-based example, let us assume that a prediction for hourly failure rates is required, rather than the yearly one. One could simply divide the results by 8760 since that is how many hours there are in a year; however, another analyst might want to perform the MCMC calculations using the yearly data first converted to hourly data. However, beware the precision limits of the MCMC program when performing this sort of analysis (note this limitation holds for many software programs that have to treat small numbers).

The issue here comes from how the mean is attained through the Poisson model:

$P(x) \sim mean^{x * e^{(-mean)}}$

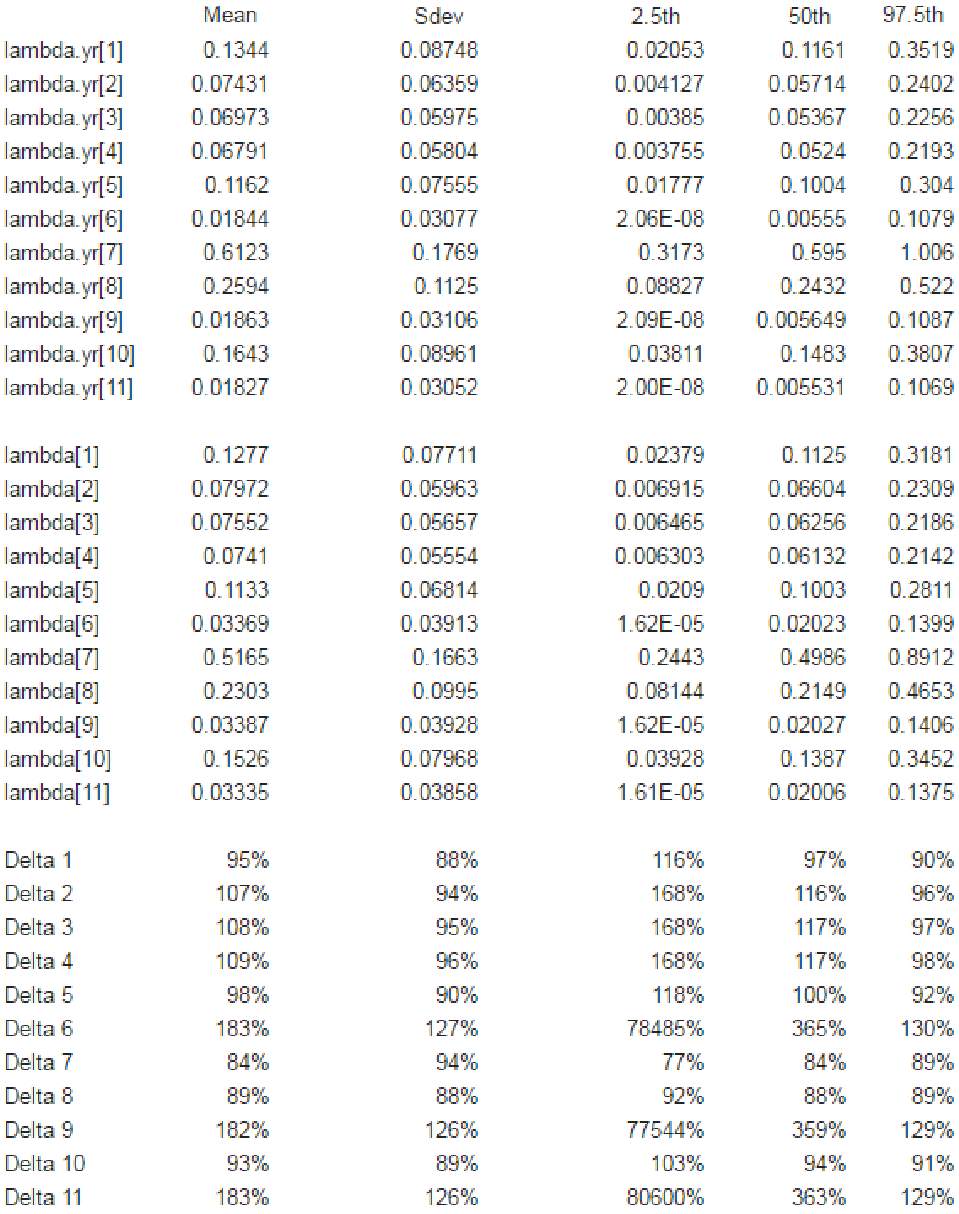
$mean = lambda * t$

$$P(x) \sim (lambda * t)^{x} e^{(-lambda * t)} \quad \text{(F-11)}$$

As parameter “t” becomes larger it is multiplied by the diffuse Gamma in the model and the numerical precision of the MCMC program comes into play.

A comparative analysis was performed using the Gamma prior with yearly data versus using hourly data. The results of these are presented in Figure F-29. The difference (delta) between the two shows up most significantly in the lower tails of zero failure data sources, where the extremely low values for the 2.5th percentile in the E-08 range indicate an issue with the analysis. However, there is approximately a 100% delta across the board between using yearly data versus hourly data. A look at the hourly lambda.pred of 3.904E-05/h (not in the figure) multiplied by 8760 h/y results in a mean of 0.342/y rate, which is 219% greater than the rate determined by using yearly data.

In cases where yearly data are given and an hourly rate is desired, it is best to use the yearly data and convert the results to hourly for use in PRA. If hourly data are given, be aware of limitations of the MCMC program in use and possibly convert the hourly data to yearly for the analysis.



	Mean	Sdev	2.5th	50th	97.5th
lambda.yr[1]	0.1344	0.08748	0.02053	0.1161	0.3519
lambda.yr[2]	0.07431	0.06359	0.004127	0.05714	0.2402
lambda.yr[3]	0.06973	0.05975	0.00385	0.05367	0.2256
lambda.yr[4]	0.06791	0.05804	0.003755	0.0524	0.2193
lambda.yr[5]	0.1162	0.07555	0.01777	0.1004	0.304
lambda.yr[6]	0.01844	0.03077	2.06E-08	0.00555	0.1079
lambda.yr[7]	0.6123	0.1769	0.3173	0.595	1.006
lambda.yr[8]	0.2594	0.1125	0.08827	0.2432	0.522
lambda.yr[9]	0.01863	0.03106	2.09E-08	0.005649	0.1087
lambda.yr[10]	0.1643	0.08961	0.03811	0.1483	0.3807
lambda.yr[11]	0.01827	0.03052	2.00E-08	0.005531	0.1069
lambda[1]	0.1277	0.07711	0.02379	0.1125	0.3181
lambda[2]	0.07972	0.05963	0.006915	0.06604	0.2309
lambda[3]	0.07552	0.05657	0.006465	0.06256	0.2186
lambda[4]	0.0741	0.05554	0.006303	0.06132	0.2142
lambda[5]	0.1133	0.06814	0.0209	0.1003	0.2811
lambda[6]	0.03369	0.03913	1.62E-05	0.02023	0.1399
lambda[7]	0.5165	0.1663	0.2443	0.4986	0.8912
lambda[8]	0.2303	0.0995	0.08144	0.2149	0.4653
lambda[9]	0.03387	0.03928	1.62E-05	0.02027	0.1406
lambda[10]	0.1526	0.07968	0.03928	0.1387	0.3452
lambda[11]	0.03335	0.03858	1.61E-05	0.02006	0.1375
Delta 1	95%	88%	116%	97%	90%
Delta 2	107%	94%	168%	116%	96%
Delta 3	108%	95%	168%	117%	97%
Delta 4	109%	96%	168%	117%	98%
Delta 5	98%	90%	118%	100%	92%
Delta 6	183%	127%	78485%	365%	130%
Delta 7	84%	94%	77%	84%	89%
Delta 8	89%	88%	92%	88%	89%
Delta 9	182%	126%	77544%	359%	129%
Delta 10	93%	89%	103%	94%	91%
Delta 11	183%	126%	80600%	363%	129%

Figure F-29. Comparison of using yearly data versus hourly data in Poisson model.

F-4. REFERENCES

F-1. Kaplan, S., 1983, On a “two-stage” *Bayesian Procedure for Determining Failure rates From Experimental Data*, IEEE Transactions on Power Apparatus and Systems, PAS-102, 1983.

F-2. Dezfuli, H., D. Kelly, C. Smith, K. Vedros, and W. Galyean, 2009, *Bayesian Inference for Probabilistic Risk and Reliability Analysis*, NASA/SP-2009-569, June 2009.

F-3. Modarres, C., E. Droguett, and M. Fuge, 2016, *A Novel Clustering Based Methodology for Overcoming Heterogeneous Populations for Reliability Prediction*, Wiley-Manuscripts, 2016.

F-4. Ester, M., H. P Kriegel, J. Sander, and X. Xu, 1996, *A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise*, Proceedings of Second International Conference on Knowledge Discovery and Data Mining (KDD-96), AAAI Press, 1996.

F-5. Kelly, D. and C. Smith, 2011, *Bayesian Inference for Probabilistic Risk Assessment- A Practitioners Guidebook*, Springer, 2011.

F-6. Kelly, D. and C. Atwood, 2008, *Bayesian Modeling of Population Variability – Practical Guidance and Pitfalls*, PSAM-9, INL/CON-08-14208, May 2008.

Appendix G

Expert Elicitation

G-1. OVERVIEW

G-1.1 Background: Why Ever "Elicit" "Expert Opinion?"

Below, a cursory survey is provided of the evolution of thinking about expert elicitation. The purpose of the survey is to illustrate to analysts how it is possible to go wrong, or to appear to go wrong, and how this experience has shaped the methods for "expert elicitation" that currently seem best. Following that, selected methods are discussed in slightly more detail, illustrating some of the current thinking.

Use of "expert elicitation" will probably always have its detractors, including advocates of methods that are not highlighted in this brief appendix; but the purpose of this appendix is to help investigators who are responsible for risk analyses, not to provide a cookbook that will preclude all controversy. That said, the methods discussed at the end have evolved from some of the misadventures of the 1980s and 1990s, which are worth understanding, even if some newer method arises.

Regardless of the discussions and recommendations offered in this appendix, the risk analysts are responsible for their results, and the decision-makers are accountable for their decisions. The experts are not there to take the blame for bad decisions.

G-1.2 Summary of Recommendations

Expert elicitation is a very large subject, and a comprehensive survey would be much longer than this appendix. The brief historical survey provided below identifies issues that have arisen in expert elicitations, and shows how elicitation methods have evolved to respond to those issues. The methods surveyed range from simple polls to studies involving committees of experts working for months.

Considering the need for objectivity and some rigor, balanced against the availability of resources (time and money), two approaches to expert elicitation that presently suggest themselves for a typical offshore risk analysis are the:

- Cooke method, using performance-based weighting
- Kaplan "expert evidence" method ("weigh evidence, not experts").

The survey provided below is intended to show why those methods proceed in the way that they do.

G-2. SURVEY OF METHODS

In the interest of accuracy, and to try to convey the evolution of the thinking in this area, much of the following is directly quoted from the original sources. Verbatim excerpts from material cited are indented. As usual, ellipses … in quoted material denote areas where material has been left out; material in brackets in quoted material [thus] is not in the original, except where the original is using them to cite references, but has been added here to emphasize some particular point.

The survey is not exhaustive, but is generally chronological. Over the time period covered in the survey, we see an evolution from a largely subjectivist Bayesian posture to a posture that goes to much greater lengths to achieve "objectivity." Any method that can be called "Bayesian" aims at furnishing evidence-driven results, and no method so far works without the involvement of human analysts at some level; but in recent times, much effort has been spent devising methods that reduce the effects of certain limitations of human analysts. The two methods recommended at the end of this appendix are formulated with this in mind.

G-2.1 The Delphi Method

An interesting recent survey by Ayyub [G-1] cites very early work on the "Delphi" method for assessing model parameter values by asking experts for their opinions. The Delphi method somewhat resembles some later methods, in treating expert opinions as if they were experimental observations of the parameter being quantified; but it differs markedly in how those opinions are obtained.

According to Ayyub:

> *The purpose and steps of the Delphi method depend on the nature of use. Primarily the uses can be categorized into (1) technological forecasting, and (2) policy analysis. The technological forecasting relies on a group of experts on a subject matter of interest. The experts should be the most knowledgeable about issues or questions of concern. The issues and/or questions need to be stated by the study facilitators or analysts or a monitoring team, and high degree of consensus is sought from the experts. On the other hand, the policy analysis Delphi method seeks to incorporate the opinions and views of the entire spectrum of stakeholders, and seeks to communicate the spread of opinions to decision-makers. In engineering, we are generally interested in the former type of consensus opinion.*

The present concern is "engineering," but as will be seen later, the methods surveyed here seek to communicate the spread of opinions to decision-makers. In fact, consideration of uncertainty is key to a reasonable decision-making process.

Excerpt from Dalkey [G-2]:

> *Selection of a single advisor in "soft" areas is clearly fraught with danger; on the other hand, committees have certain drawbacks which have been dramatized by a large number of investigations by psychologists and small-group sociologists over the last two decades (1). One major drawback is the influence of the dominant individual. A quite convincing group of studies have shown that the group opinion is likely to be highly influenced, if not determined, by the views of the member of the group who does the most talking, and that there is no significant correlation between success in influencing the group and competence in the problem being discussed.* ***[!!!!]*** *Another difficulty which has not received as much attention in the literature is "noise" – irrelevant or redundant material that obscures the directly relevant material offered by participants. A third difficulty is group pressure that puts a premium on compromise.*
>
> *<u>2. DELPHI PROCEDURES</u>*
>
> *The Delphi procedures have been designed to reduce the effects of these undesirable aspects of group interaction. The procedure has three distinctive characteristics:*
>
> 1. *Anonymity.*

2. *Controlled feedback.*
3. *Statistical "group response."*

By "group response," Dalkey means what others would refer to as the "aggregation" of opinion. Interestingly, Dalkey says that where a number is required, the median of the individual members' estimates is the best of the indices tried in his work. Later, we will see a case where the mean was used, and implicitly criticized; and *neither* index is used in either of the two methods recommended above and in [G-3].

The Delphi method is mentioned here because it is a very early example of method in this area, and because it is still occasionally cited. However, context is also important. Dalkey was working at RAND when the method was formulated. The intended application was "prognostication." This has something in common with regulatory decision-making, or safety decision-making by facility operators, but it also has features that differ. Among the noteworthy differences is the matter of "agency" (*who* is deciding *what* on behalf of *whom*). Prognostication for essentially private applications is not subject to the sort of scrutiny that takes place in debates over facility safety. The analysis must convince not only oneself, but others.

Dalkey was seriously concerned about issues with "committee" dynamics. Methods surveyed later in this appendix take an opposing view, promoting interaction rather than preventing it, and trying to solve "dominant personality" issues by other means. It can also be argued that if you view expert opinions as being analogous to experimental data, as the early workers in this field apparently did, then you want your "measurements" to be "independent." The change in thinking about this over the last half-century is significant.

G-2.2 IEEE/ANS PROBABILISTIC RISK ASSESSMENT PROCEDURES GUIDE TREATMENT OF EXPERT ELICITATION

The first probabilistic risk assessment (PRA) procedures guide, NUREG/CR-2300, [G-3] illustrates the state of practice that was obtained in the early 1980s in nuclear PRA.

Annotated excerpts:

> *5.5.2.2.5 Using Expert Opinion*
>
> *Expert opinion is often used for a prior probability distribution when other information is inadequate. If neither physical nor theoretical models are available and relative frequency is unavailable as well, subjective assessment is the only alternative for obtaining a probability. The practical feasibility of this alternative is supported not only by theoretical foundations that show judgments about uncertain events can be expressed as probabilities but also by practical assessment procedures. Holloway (1979) reviews the basis for these procedures and gives examples for several assessment approaches. The following summary of assessment procedures draws on his book. After this summary, well-known cautions and guidelines for interpreting and reviewing expert opinions are presented to highlight the care and caveats that must accompany the quantitative assessment. However, the user of this guide should be cautioned against the indiscrete [sic] use of the methods described in this section. These techniques and results are not necessarily applicable to PRAs, which often treat extremely small probabilities of various events. More research is needed to determine the direct applicability of these methods and findings to PRAs. The user should be aware that the subjective estimates frequently used in PRAs can have large biases and errors.*

Note the caution expressed in the above guidance. At that time, the authors still found it necessary to cite arguments to the effect that opinions can be processed using the mathematics of probability. They also alluded indirectly to the point that people are not generally good at assessing small probabilities, especially ones associated with events that are beyond common experience. That caution is repeatedly stressed in the rest of the discussion as well, and appears to have influenced the formulation of the Kaplan method, discussed later.

> …
>
> *Assessment Procedures*
>
> *Two approaches to subjective probability assessment are in practical use, either the direct approach or the indirect approach. With the direct approach, the expert is asked to declare the probability number associated with the feeling of uncertainty for the occurrence of an event. With the indirect approach, an expert is asked to choose between a reference assessment lottery and the uncertain feeling (the degree of belief) in an opinion or judgment.* ***Until an expert has shown an ability both to form a knowledgeable opinion and to assess, unaided, a probability for the degree of belief associated with that opinion, the indirect approach is preferred. The well-known difficulties in obtaining useful subjective probability assessments are summarized below in the section entitled "Validity of Expert Opinion."*** *These difficulties are magnified by inexperienced, unaided direct assessments. The references in that section give some experience comparing the two approaches. [Emphasis added]*
>
> *The direct approach has the expert state a number that represents the assessment of the probability. Some studies have shown it possible for people to become better at assessing their own feelings of uncertainty as probabilities (see for example, Stael von Holstein, 1970; Lichtenstein et al. 1977). This improvement in direct assessment comes from specific training and guided practiced discipline rather than by trial and error.* ***A good direct assessment comes from one who is both an experienced expert in what is known about a technical area (as well as how much is not known) and an experienced expert on how to express that judgment with little cognitive bias. This is an uncommon combination of expertise.***
>
> *Assessment lotteries are used in the indirect approach to disclose the subjective probability. This external reference is used as a scale to measure the internal degree of belief an expert holds toward an opinion. Dividing between the expert and the assessors the responsibility to provide both a well-founded, knowledgeable judgment and an accurate representation of that judgment as a probability allows the use of expert opinion in PRAs. Most technical experts are not practiced, good probability assessors of themselves. Using the indirect approach improves the quality of expert opinion over that obtained by unaided, inexperienced direct assessment. Fischhoff et al. (1981) have shown that people qualified as technical experts are by no means qualified as probability assessors of that expertise.*

Following are summary observations on the guidance in NUREG/CR-2300:

- Throughout, the PRA Procedures Guide is relatively non-prescriptive, but it is arguably especially non-prescriptive in its discussion of expert elicitation. It even offers commentary (excerpted above) to the effect that the applicability of "these methods" to "PRAs" is still (at the time of its writing, early 1980s) a research topic. If, in fact, research activity in this area has died down by now (2017), it is arguably not because all of the issues have been settled.

- The point is repeatedly stressed that domain expertise in a particular subject area *does not* translate into competent assessment of probabilities. We do not see this reflected in the implementation of Delphi, but with the passage of time, the methods use more and more formal process for many reasons, including this issue.

G-2.3 Expert Elicitation in NUREG-1150

WASH-1400 [G-4] (ca. 1975; begun under the auspices of the Atomic Energy Commission, and finished under the auspices of the Nuclear Regulatory Commission) is generally considered to be the first plant-scale PRA to have been done. Industry subsequently performed full-scope PRAs, and others performed PRA-like activities; but the next real NRC-sponsored "PRA" was NUREG-1150 [G-5], performed in the late 1980s. This was, among other things, a kind of update to WASH-1400. The relevance of NUREG-1150 to this appendix is that NUREG-1150 invested very significantly in elicitation of expert opinion; results in DRAFT NUREG-1150 were controversial, and were revised in response to comments. Certain essentials of these issues are of present interest, because the methods discussed in more detail later in this appendix are, in part, a reaction to these issues. Accordingly, review commentary on NUREG-1150 is excerpted below; interested readers can still find most of the original NUREG-1150 material online.

From NUREG-1420 [G-6]:

> *3.2.3 Elicitation of Expert Opinion*
>
> *One of the distinctive features of NUREG-1150 was the extensive use of structured, formalized elicitation of expert opinion. ... The process was used to generate input values and distributions for many of the parameters in the study where reliable models and values were not available, e.g., due to the complexity of the phenomena. The procedure to elicit expert opinion used for the first draft of NUREG-1150 and the results obtained with it were extensively criticized by the peer reviews; the entire process was restructured and elicitation was redone for the second draft. ...*
>
> *The elicitation of expert opinion was such an important part of the NUREG-1150 methodology that it is discussed at length in Section 4.4 of this report [NUREG-1420].*
>
> *...*
>
> *4.4 Expert Opinion*
>
> *One of the distinctive features of NUREG-1150 was the extensive use of structured, formalized elicitation of expert opinion. This process provided input values and distributions for many of the parameters in the study for which values were not otherwise available or where the available results were incomplete, highly uncertain, or internally discrepant. ...*
>
> *The expert opinion process involved several steps:*
>
> - *Selection of the expert panels.* *Several expert panels were assembled. An attempt was made to include technical judgments from national laboratories, government, universities, and industry, endeavoring to include a wide range of views. This did not always succeed.*
> - *Training.* *Professionals in the elicitation of expert opinion trained the panel members in that discipline. ... [Recall the commentary in NUREG/CR-2300 regarding training of experts.]*

- *<u>Technical Presentations and Discussions.</u> The objective was to provide the experts with the information and relevant technical literature available on the subjects, and, consequently, to bring all the experts on a panel up to approximately the same technical background and level of understanding.*
- *<u>Elicitation Process.</u> After the training sessions, the experts were given several weeks to review the material, continue discussions, consult other experts, and make additional supporting analyses of their own. In some cases, the groups were reassembled for additional discussions and presentations. Each expert provided his/her opinion on an individual basis in a private session with an individual trained in the elicitation process. The experts were also required to provide detailed documentation of the rationale for their opinions.*
- *<u>Results.</u> The values or distribution functions from the experts were averaged* ***[!!!!]*** *to provide those used in the analysis.*

Expert opinion was selected for the initial draft of NUREG-1150 but this was not the formal, professionally guided process described above, and most of the reviewers of the initial draft were critical of this first attempt at elicitation. Therefore, the elicitation was repeated using this more structured process. ...

Expert opinion elicitation is technically less satisfactory than the use of detailed, validated analytical procedures, or experimental data. ...

- *Formal, professionally structured expert opinion is preferable to the current alternative, according to which the individual PSA analysts make informal judgements which are not always well-documented. ... The reproducibility of the results of expert opinion is a concern.*

  …
- *There is always a question as to who is an expert on a given issue. ...*
- *The training of the experts and their subsequent discussions were valuable in clarifying the focus on the important issues.*

  …
- *Expert opinion may have been relied upon too heavily in some instances. ... It may have been thought that the analysis would have been too time-consuming. It would have been appropriate if possible to have developed these analyses and then to have subjected them to critical review to which expert opinion could have been directed.*

  …
- *The study assigned equal weight factors to the opinions of all experts.* ***[!!!!]*** *Some other methods, which might develop unequal weight factors, were not used.*

Key points from the above are as follows:

- The above discussion still reflects the idea that expert opinions are to be treated as if they are somehow themselves “data.” For example, it is remarked that in at least some cases, the expert opinions were “averaged.” The commentary excerpted above does not take exception to the idea that expert opinions are “data”; it merely notes that “unequal weight factors” might have been used instead of the unweighted averaging process. The methods recommended later in this appendix do not use "averaging." One of the methods pays a great deal of attention to methodically developing "unequal weight factors"; the other does not even ask experts for the answer to the subject question, but rather asks what *evidence* informs their answer to the subject question, and tries to get the group of experts to agree on the body of evidence that informs the range of opinions in the group.

- The NUREG-1420 commentary favors explicit process, and documentation.
- NUREG-1150 appears to have tried to address the commentary quoted previously from NUREG/CR-2300, to the effect that domain experts are not good at assessing probabilities, and that "training" is important.
- It is observed that analysts make judgments all the time, generally within a less formal process than the process used for big-ticket "elicitations." The big-ticket elicitations draw attention, while numerous smaller instances of analyst judgment slip under the radar.
- Even the improved version of NUREG-1150 is criticized for having resorted prematurely to the use of expert opinion.

G-2.4 EPRI/ LLNL Studies of Seismic Hazard

In the late 1980s, the conduct of two studies of seismic hazard, described below, led to a methodological fork in the road. Commentary from NUREG-2117, Rev.1 [G-7]:

> *Perhaps the most dramatic and important revelations regarding the significance of expert assessment methodologies emerged when parallel regional Probabilistic Seismic Hazard Analyses (PSHAs) were conducted for central and eastern U.S. nuclear power plant sites. The Electric Power Research Institute-Seismicity Owners Group (EPRI-SOG, 1988, 1989) and Lawrence Livermore National Laboratory (LLNL) (Bernreuter et al., 1989) studies were both conducted using multiple experts, and both studies were conducted mindful of the importance of uncertainties. However, the processes used to conduct the studies were quite different.*
>
> *...*
>
> *Although the methodological differences used in the two studies were known at the time the studies were conducted, no procedural guidance existed and there was little indication that the differences would have a significant effect on the results. However, a comparison of the calculated hazard results at the 56 common sites in the central and eastern United States (CEUS) showed significant differences between the two studies (summarized in USNRC 2010, NUREG-0933, Generic Issue 194). The DOE and NRC looked into the issue and found that an important contributor to the difference was the seismicity experts' input related to lack of correlation between the recurrence parameter "a" and "b" values. This issue was the driving force behind NRC formal updating of the LLNL results as documented in NUREG-1488 (Sobel, 1994). NUREG-1488 both compares the studies and provides an updated model. Another key concern on the part of the LLNL study was the fact that a single ground motion expert provided assessments that were well outside of the range of assessments provided by the other four ground motion experts on the panel. The "outlier" expert's assessments had a significant impact on the mean estimate that was calculated based on input provided across the panel. The concern regarding the unbalanced impact of outliers on assessments was largely addressed in a followup study by LLNL that took a different approach to uncertainty characterization by developing a*

> *composite ground motion model (Savy et al., 1993). However, the differences between the LLNL and EPRI-SOG hazard estimates remained—particularly in the range of annual frequencies of t10^{-4} to 10^{-6} , which is the range of seismic hazard that typically has the most contribution to seismic risk for nuclear power plants (USNRC 2010, NUREG-0933, Generic Issue 194). Within this range, the LLNL mean hazard results were systematically higher than the EPRI-SOG results.*

This episode was quite well known at the time. Two studies [G-8, G-9] of more or less the same thing got very different results, because the opinions were being treated as data, and because one of the studies included an expert whose opinion was very different from the opinions of the other experts. The fact that the two studies differed was already noteworthy, but even without this difference, it was noteworthy that because of the aggregation method, and the fact that uncertainties in seismic hazard are enormous, one expert's opinion effectively determined one of the studies' outcomes, even though other experts had also participated.

This helped to fuel the development of the Senior Seismic Hazard Analysis Committee (SSHAC) guidelines [G-10], presently considered by many authorities to be a sort of gold standard in the area of expert elicitation. In the interest of chronology, we will next discuss Cooke's approach to performance-based weighting [G-11], followed by Kaplan's "Expert Evidence" idea [G-12], and then briefly describe the SSHAC approach.

G-2.5 Performance-based Weighting

The Cooke method is described perhaps most accessibly in [G-11].

In a sense, the Cooke approach treats the experts as "data," but instead of directly pooling the experts' responses, the assessor uses those data in a much more thoughtful way. Cooke's name for this is "performance-based weighting." In Cooke's parlance, good "performance" of an expert means that in response to calibration questions asking the expert to quantify parameters whose values are unknown to the expert but known to the assessor, that expert usually quotes intervals that (a) contain the right answer, and (b) are narrow enough for the expert to be considered "informative." An expert whose intervals seldom contain the right answer, or whose bounds are so wide that the response is uninformative, scores low in the calibration exercise. So Cooke's approach proceeds through a calibration stage, posing calibration questions whose answers are known to the assessor, and then moves on to questions whose answers are unknown but needed. Cooke's algorithm processes the responses to the unknown questions based on experts' performance on the calibration questions. At the end of the process, it may well be the case that a large pool of experts (say, several tens of experts) has been distilled down to two or three experts whose answers to the questions about the unknown should be given significant weight.

Besides being affected by the choice of experts, this approach is obviously affected by the formulation of the calibration questions. The questions could range over a very wide technical area, and still be useful for sorting out who understands his/her own limitations; but the questions arguably should relate somewhat to the subject domain of the elicitation. This requires some preparatory work by the elicitors.

G-2.6 “Expert Evidence vs. Expert Opinions”

For a simple introduction to the idea, it is difficult to improve on Kaplan’s original abstract [G-12], provided below.

> *‘Expert information’ versus ‘expert opinions.’*
>
> *Another approach to the problem of eliciting/ combining/using expert knowledge in PRA*
>
> *Stan Kaplan*
>
> *In the traditional approach to eliciting expert knowledge for use in risk assessment and decision analysis, the [i^{th}] expert is asked for his opinion about, say, the numerical value of some unknown parameter λ. This opinion is then expressed as a point estimate, λ_i, or a probabilistic estimate, $p_i(\lambda)$. Much attention and debate is then given, in the traditional approach, to methods of weighing and combining the opinions from the individual experts.*
>
> *The present paper advocates another approach in which we ask each expert, instead, for his body of evidence, E_i, relevant to the value of λ. In this way, the approach first arrives at a consensual body of evidence, E- $\{E_i\}$, and second, at a consensual curve $p(\lambda|E)$ that expresses our knowledge about λ based on that body of evidence.*
>
> *The essential difference between this ‘expert information’ approach and the traditional ‘expert opinion’ approaches may be captured in the slogan: ‘Weigh evidence, not experts!’*

In a sense, this idea goes back to the long-recognized, but not always realized, point that domain experts are not necessarily good at the probabilistic aspects of the assessment. According to Kaplan’s summary:

> *The emphasis is on getting a very clear written statement of a ‘total body of evidence’, E_T, which contains the evidence from each expert, and which has been talked over and clarified to the extent that all the experts are willing to agree that E_T constitutes the total evidence of the group relevant to the value of λ.*
>
> *E_T must, therefore, be the basis for any decision in which the value of λ plays a role.*
>
> *What remains, then, is to translate E_T into a probability curve, $p_c(\lambda|E_T)$, against λ. The PRA analyst should take the lead in this translation since that is his business [not the business of the domain experts], but all the experts should agree that this curve expresses the consensus state of knowledge of the group. This means that an individual expert may have a different curve, but, nevertheless, he agrees that the consensus curve is the one that should go to the decision maker with the statement that this represents the combined knowledge of the expert group.*

This latter idea strongly influenced the formulation of the SSHAC process.

This idea was developed well after NUREG/CR-2300, and was designed specifically to avoid the *opinion*-pooling process. Kaplan calls this idea “expert evidence,” as contrasted with “expert opinion.” Instead of asking experts for their answers, one first asks the experts what evidence informs their opinions, and aggregates that evidence. In Kaplan’s own words:

> *With regard to the difficult question: "How should we combine probability distributions from different experts?" my suggestion is that we bypass it. We can bypass it by not asking the experts for their probability distributions. Instead, we ask each expert independently what evidence, information, and experience he/she has, relevant to the question at hand. We write these items of evidence down very carefully, and collect them in a combined list. Then, using the experts as a group, we work over these items, clarifying, coalescing, refining, adding new items that come to mind, and all the time being sure to distinguish the actual evidence (i.e., "what happened") from the experts' interpretation of what happened. At the end of this process we should have a single, agreed upon "consensus body of evidence." Then, together with the group, we process this combined body of evidence, item by item, through Bayes' theorem to arrive finally at a posterior probability curve that expresses our joint state of knowledge about the parameter of interest.*

Kaplan's idea was used by the SSHAC work, described later in this appendix. As indicated earlier, the SSHAC work was initiated as the result of a pair of seismic hazard assessments that used different experts and came up with essentially irreconcilable results, and in which the traditional processes of pooling of expert opinions was deemed to have failed: one of the assessment results was dominated by the input from a single expert, whose opinion was an outlier from the perspective of the group as a whole. The application of the Kaplan idea in the SSHAC process results in a body of evidence that the technically-informed community collectively agrees is the body of evidence that the technically-informed community collectively relies upon. This does not mean that every expert believes the same thing or relies in the same way on the various bits of evidence, but rather that the *group* agrees that the indicated body of evidence is relied on by the *group*.

Note that the general discussion given above does not tell the prospective user how to formulate the likelihood model based on all that evidence. It is the "business" of the "PRA analyst" to know how to do that. Eliciting evidence in the way described above could lead to gathering of a collection of evidence of disparate types and varying applicability, and accommodating all this inhomogeneity is a challenging and important task.

G-2.7 Comparison of Kaplan's "Expert Evidence" Idea with Cooke's Performance-Weighting Approach

Similarities: Both are quintessentially Bayesian. Both acknowledge (as do many, many others) that in real applications, situations often arise in which there is no practical alternative to doing what is called "expert elicitation" (though Kaplan advocates eliciting "expert evidence" rather than answers).

Differences: Cooke describes a method for getting answers out of the experts; the method has an interesting and extensive technical basis, involving calibration of the experts and subsequent weighting of their inputs, based on their informativeness and on their understanding of their own uncertainty. Superficially, in asking experts what they believe, Cooke's method resembles certain of the classical opinion-pooling methods, to which Cooke compares the properties of his method. But Cooke's method spends a significant amount of effort deciding whom to pay attention to: in other words, to the development of what earlier subsections in this appendix called "unequal weights." This is done not by having some sort of super-analyst judgmentally assess the "experts," but rather by measuring their performance in the course of a calibration process.

The Kaplan work cited here specifically suggests NOT asking experts for the answer(s) directly; instead, he advocates an exercise culminating in the collection and assessment of a body of *evidence* bearing on the answers, which the decision-maker/ analyst then processes with Bayesian methods to get the answers. The "processing" that is required of the decision-maker/analyst eventually entails writing down a likelihood function that needs to be based on a relevant model, which may need to be developed.

Besides a model of likelihood, relating the evidence to the various competing hypotheses, one needs a prior distribution. As discussed elsewhere in this guide, at this writing, consensus is still lacking on this matter. Much of the work of Kaplan and his collaborators relies on the idea that using a suitably diffuse prior is the best that one can do, but a large and growing body of opinion takes exception to the idea that even diffuse priors are ultimately satisfactory.

The present authors do not know of a real application of Kaplan's idea, other than the seismic application cited earlier and discussed below (SSHAC). There have apparently been many applications of the Cooke method. It would have been interesting to apply Cooke's performance-weighting approach to the experts in the seismic studies whose irreconcilability catalyzed the development of the SSHAC.

G-2.8 Senior Seismic Hazard Analysis Committee (SSHAC) Process

Unlike the other methods surveyed here, the SSHAC process was developed with a specific application in mind: probabilistic seismic hazard analysis. However, methodologically, it is a high-end approach to the use of experts, and is not inherently limited to seismic hazard analysis. For that reason, it merits inclusion for present purposes.

SSHAC makes use of a group of experts, but does not initially ask them for the answer: it asks them for their evidence, conceptually along the lines given by Kaplan. Ultimately, the "skillful user" turns all that into a body of results, but the significance of that body of results is still grounded in what that body of results means to the group of experts:

> *Regardless of the scale of the PSHA study, the goal remains the same: to represent the center, the body, and the range of technical interpretations that the larger informed technical community would have if they were to conduct the study.*

The SSHAC process can be applied at any of several levels of rigor, and much of the discussion of SSHAC revolves around "Level 4," the most rigorous. Kaplan's suggestion was nowhere near as elaborate as SSHAC Level 4.

As mentioned earlier, the SSHAC process [G-10] was developed to address a need that had been highlighted by an unsatisfactory result from a very significant body of elicitation work done along traditional lines.

Level 4 SSHAC is extremely resource-intensive. To understand why the costs may be justified, consider its context. Assessment of seismic hazard for nuclear plants is important to get right at an early stage of design, and the results have to be convincing to a very diverse spectrum of stakeholders: offsite populations, regulators, plant operators, and insurers. An overly conservative assessment of hazard may imply very large up-front costs; but an imprudent assessment may leave the facility exposed to an excessive risk. Once the facility is built, it may no longer be practical to change the facility based on new information. Unfortunately, site-specific seismic hazard is not practical to assess actuarially; if "risk" is to be considered at all, there is no real alternative to some use of experts. Nor is it practical to devise a risk management scheme relying on performance trending to prevent disaster.

Level 4 SSHAC has been mentioned here not because it is foreseen that an exactly comparable process will be widely used in risk assessment for drilling, but rather to illustrate a high-end methodology. Methodological recommendations are furnished later in this appendix. The excerpt below is from Volume 1 of [G-10]:

> *Seven-Step Process*
>
> *…*
>
> *Based on their NUREG- 1150 experience, Keeney & von Winterfeldt (1991) describe a seven-step process:*
>
> *Step 1 Identification and selection of the technical questions*
>
> *Step 2 Identification and selection of the experts*
>
> *Step 3 Discussion and refinement of the issues*
>
> *Step 4 Training for elicitation*
>
> *Step 5 Group interaction and individual elicitation*
>
> *Step 6 Analysis, aggregation, and resolution of disagreements*
>
> *Step 7 Documentation and communication*
>
> *…*
>
> *Most of the discussion in the literature on multiple-expert applications, e.g., in Otway and von Winterfeldt (1992); Meyer and Booker (1991) and Cooke (1991), can be accommodated by this list of seven steps. In a project similar in spirit to the SSHAC project, DeWispelare and others (DeWispelare, Herre, Miklas and Clemen 1993) implemented an analogous formal expert elicitation process in their Yucca Mountain future-climate study.*

Executive Summary, NUREG-2117:

> *Because adopting a Level 3 or a Level 4 process to conduct a PSHA results in a significant increase in the cost and duration of the study over that required to conduct a Level 1 or Level 2 project, it is important to highlight the potential benefits to be gained by moving to these higher levels. These benefits are associated with the greater levels of regulatory assurance in Level 3 and 4 studies. We define regulatory assurance to mean confidence on the part of the NRC (or other regulator or reviewer) that the data, models, and methods of the larger technical community have been properly considered and that the center, body, and range of technically defensible interpretations have been appropriately represented and documented. In other words, it is increased confidence that the basic objectives of a SSHAC process have been met. We do not use the term "reasonable assurance" because it has a specific definition within the NRC's regulatory framework related to compliance with regulations. Rather, regulatory assurance is a qualitative term that is specific to the confidence that is engendered by the proper execution of a SSHAC process.*

G-3. DISCUSSION

- The term "expert elicitation" dates from several generations ago, when the governing idea was that expert opinions were analogous to experimental "data" (as if asking an expert is something like doing an experiment, with noise affecting the results), and the PRA task was to "elicit" those opinions. Arguably because early studies were not deemed to be sufficiently convincing, the term "elicit" also came to refer to an increasingly formal process, without which an expert elicitation study may not be deemed to be convincing. Kammerer and Ake [G-8] argue that although the term "elicit" is used in the SSHAC reports, the way in which experts are actually used in SSHAC is fundamentally different from the way in which they are used in most previous processes.
- Sometimes, the only practical thing to do is use expert "elicitation." That said, users of risk analysis results also need to be aware that risk analysis involves many choices made by analysts, choices that are not necessarily subjected to quite the same level of process rigor as that entailed by formal elicitation.
- Use of experts in lieu of experiment or more exhaustive analysis should be considered a last resort. "Expert judgment should only be used to identify and quantify the uncertainty that remains after appropriate data collection and analysis activities have been completed." Executive Summary, NUREG-2117
- In general, formal, documented expert elicitation is expensive. It appears that the level of effort implied by the methods surveyed has tended to increase, as the methodological guidance has gone into more detail, and mandated more iteration among the experts. Every time a large group of experts is brought together for a meeting, the costs are significant, and some of the methods surveyed entail multiple meetings.
- From a purely formal point of view, the Level 4 SSHAC process has much to recommend it; it is as data-driven as it can be, given that it is administered by humans, and given that humans then execute the analysis based on the body of evidence identified. But it will likely prove to be too expensive, and too time-consuming, for many analyses of drilling operations.

G-4. RECOMMENDATIONS

- If use of expert judgment is even being considered to deal with a particular issue, this should be because:
  - There is sufficient uncertainty in the analysis team's current state of knowledge of the issue that a robust decision cannot presently be supported
  - There is no other practicable way to deal with the subject issue.
- The intended use of the risk analysis needs to inform both the selection of method, and the level of rigor involved in the execution of the method. The analysis needs to be carried out in a manner that will be convincing to the affected stakeholders. If the stakeholders are all stockholders, one level of rigor is implied; if the set of stakeholders comprises not only stockholders but also the operating crew (safety) and regulatory authorities, who are generally making risk acceptance decisions on behalf of the general public, a higher level of rigor is implied.
- Use of expert judgment has been a research topic for many years, but the field has not settled on a single approach. Two approaches that presently suggest themselves for a typical offshore risk analysis are:
  - the Cooke method, using performance-based weighting

- the Kaplan “expert evidence” method. (It is assumed that a full SSHAC Level 4 assessment will be too expensive for a facility-specific risk analysis, unless both the stakes and the uncertainties associated with the decision are extremely high.)

- Essentially all “elicitation” processes surveyed here, with the possible exception of the Delphi method, need to be carried out (led) by persons having significant normative expertise (expertise in the business of eliciting and using these judgments). This emphatically includes both of the above processes.
- Each of these two approaches deals with the “equal weight” issue that plagued some early applications of expert judgment. The Kaplan method is explicitly evidence-driven, and leads to a group statement about the group’s views of the evidence; the Cooke method is implicitly evidence-driven, in the sense that more evidence-driven experts will tend to have greater weight. Both methods characterize the team’s state of knowledge, giving not only an “answer,” but also characterizing uncertainty.
- Both methods require resources, both to acquire the services of the experts and to support preparation on the part of the analysts. The Kaplan method requires at least one meeting. But either can be executed more easily than a SSHAC Level 4 can be executed.
- The availability of methods such as the above is not a license to go through the steps listed for each method, and use the results uncritically in an important analysis. The decision-maker needs to “own” the residual uncertainties, and if they are still substantial enough to be worth reducing, the decision-maker needs to consider doing more to reduce them.

G-5. REFERENCES

G-1. Ayyub, Bilal, *Methods for Expert-Opinion Elicitation of Probabilities and Consequences for Corps Facilities*, IWR Report -00-R-10 Prepared for U.S. Army Corps of Engineers Institute for Water Resources, Alexandria, VA 22315, December 2000.

G-2. DELPHI, Norman C. Dalkey, Second symposium on Long-Range Forecasting and Planning, Alamogordo, New Mexico, October 11-12, 1967.

G-3. *PRA Procedures Guide: A Guide to the Performance of Probabilistic Risk Assessments for Nuclear Power Plants,* NUREG/CR-2300, Prepared Under the Auspices of the American Nuclear Society and the Institute of Electrical and Electronics Engineers, U.S. Nuclear Regulatory Commission, 1983.

G-4. *Reactor Safety Study: An Assessment of Accident Risks in U. S. Commercial Nuclear Power Plants*, WASH-1400 (NUREG 75/014), U.S. Nuclear Regulatory Commission, October 1975.

G-5. *Severe Accident Risks: An Assessment for Five U. S. Nuclear Power Plants*, NUREG-1150, U.S. Nuclear Regulatory Commission, December 1990.

G-6. *Special Committee Review of the Nuclear Regulatory Commission’s Severe Accident Risks Report (NUREG-1150)*, Herbert J. C. Kouts, George Apostolakis, E. H. Adolf Birkhover, Lars G. Hoegberg, William E. Kastenberg, Leo G. LeSage, Norman C. Rasmussen, Harry J. Geague, and John J. Taylor, NUREG-1420, U.S. Nuclear Regulatory Commission, 1990.

G-7. *Practical Implementation Guidelines for SSHAC Level 3 and Level 4 Hazard Studies*, NUREG-2117, Rev. 1, U.S. Nuclear Regulatory Commission, April 2012.

G-8. EPRI-SOG, 1988, *Seismic Hazard Methodology for the Central and Eastern United States*, EPRI NP-4726A, Rev. 1, Volumes 1-11, Electric Power Research Institute, Palo Alto, California.

G-9. Bernreuter, D. L., J. B. Savy, R. W. Mensing, J. C. Chen, and B. C. Davis, 1989, *Seismic Hazard Characterization of 69 Nuclear Plant Sites East of the Rocky Mountains*, NUREG/CR-5250, Volumes 1–8, U.S. Nuclear Regulatory Commission, Washington, DC.

G-10. *Recommendations for Probabilistic Seismic Hazard Analysis: Guidance on Uncertainty and Use of Experts*, Prepared by Senior Seismic Hazard Assessment committee (R. J. Budnitz, G. Apostolakis, D. M. Boore, L. S. Cluff, K. J. Coppersmith, C. A. Cornell, and P. A. Morris), NUREG/CR-6372, U.S. Nuclear Regulatory Commission, 1997.

G-11. T. Bedford and R. Cooke, Probabilistic Risk Analysis, Cambridge University Press, UK, 2001.

G-12. Kaplan, S., *Expert Information vs. Expert Opinions: Another Approach to the Problem of Eliciting/ Combining/ Using Expert Knowledge in PRA*, Reliability Engineering and System Safety, 25, 61-72, 1992.

Appendix H

Common-Cause Failure

The objective of the detailed common-cause analysis is to identify the potential vulnerabilities of the system being analyzed to the various common-cause failures (CCFs) that can occur, and to incorporate their impact into the system models. This appendix focuses on a detailed approach, called the alpha-factor method, which is commonly used in other industries.

As a first step, the analyst should extend the scope of the qualitative screening analysis and conduct a more thorough qualitative assessment of the system vulnerabilities to CCF events. This detailed analysis focuses on obtaining considerably more system-specific information and can provide the basis and justification for engineering decisions regarding system reliability improvements. In addition, the detailed evaluation of system CCF vulnerabilities provides essential information for a realistic evaluation of operating experience and system-specific data analysis as part of the detailed quantitative analysis. It is assumed that the analyst has already conducted a screening analysis, is armed with the basic understanding of the analysis boundary conditions, and has a preliminary list of the important common-cause component groups.

An effective detailed qualitative analysis involves the following activities:

- Review of operating experience (generic and system-specific)
- Review of system design and operating practices
- Identification of possible causes and coupling factors and applicable system defenses.

The key products of this phase of analysis include a final list of common-cause component groups supported by documented engineering evaluation. This evaluation may be summarized in the form of a set of cause-defense and coupling factor-defense matrices (see [D-1]) developed for each of the common-cause component groups identified in the screening phase. These detailed matrices explicitly account for system-specific defenses, including design features and operational and maintenance policies in place to reduce the likelihood of failure occurrences. The results of the detailed qualitative analysis provide insights about safety improvements that can be pursued to improve the effectiveness of these defenses and reduce the likelihood of CCF events.

Given the results of the screening analyses, a detailed quantitative analysis can be performed even if a detailed qualitative analysis has not been conducted. However, as will be seen later, some of the steps in the detailed quantitative phase, particularly those related to analysis and classification of failure events for CCF probability estimation can benefit significantly from the insights and information obtained as a result of a detailed qualitative analysis.

A detailed quantitative analysis can be achieved through the following steps:

4. Identification of common-cause basic events
5. Development of probabilistic models of common-cause basic events
6. Estimation of common-cause basic event probabilities
7. Incorporation of common-cause basic events into the system fault tree.

The first three steps are discussed in the following sections. The last step, incorporation into the system fault tree, is discussed in Section 2.

H-1. IDENTIFICATION OF COMMON-CAUSE BASIC EVENTS

This step provides the means for accounting for the entire spectrum of CCF impacts in an explicit manner in the logic model. It will also facilitate the fault-tree quantification to obtain top event (system failure) probability.

A common-cause basic event is an event involving failure of a specific set of components due to a common cause. For instance, in a system of three redundant components A, B, and C, the common-cause basic events are C_{AB}, C_{AC}, C_{BC} , and C_{ABC}. The first event is the common-cause event involving components A and B, and the fourth is a CCF event involving all three components. Note that the common-cause basic events are only identified by the impact they have on specific sets of components within the common-cause component groups. Impact in this context is limited to "failed" or "not failed."

The complete set of basic events, including common-cause basic events, involving component A in the three component system is:

A_I	=	Single independent failure of component A. (a basic event)
C_{AB}	=	Failure of components A and B (and not C) from common causes
C_{AC}	=	Failure of components A and C (and not B) from common causes
C_{ABC}	=	Failure of components A, B, and C from common causes.

Component A fails if any of the above events occur. The equivalent Boolean representation of total failure of component A is:

$$A_T = A_I + C_{AB} + C_{AC} + C_{ABC} \qquad \text{(H-1)}$$

H-2. DEVELOPMENT OF PROBABILISTIC MODELS OF COMMON-CAUSE BASIC EVENTS

With the CCF events identified, this section describes the probabilistic models that are commonly used for common-cause basic events. This is done first by utilizing a three-component system example, where two component failures result in the system failing.

Using the rare event approximation, the system failure probability of the two-out-of-three system is given by:

$$\Pr(S) = \Pr(\mathrm{A_I}) \Pr(\mathrm{B_I}) + \Pr(\mathrm{A_I}) \Pr(\mathrm{C_I}) + \Pr(\mathrm{B_I}) \Pr(\mathrm{C_I}) + \Pr(\mathrm{C_{AB}}) + \Pr(\mathrm{C_{AC}}) + \Pr(\mathrm{C_{BC}}) + \Pr(\mathrm{C_{ABC}}) \qquad \text{(H-2)}$$

It is common practice in risk and reliability analysis to assume that the probabilities of similar events involving similar components are the same. This approach takes advantage of the physical symmetries associated with identically redundant components in reducing the number of parameters that need to be quantified. For example, in the above equation it is assumed that:

$$\Pr(A_I) = \Pr(B_I) = \Pr(C_I) = Q_I \qquad \text{(H-3)}$$

$$\Pr(C_{AB}) = \Pr(C_{AC}) = \Pr(C_{BC}) = Q_2 \qquad \text{(H-4)}$$

$$\Pr(C_{ABC}) = Q_3 \qquad \text{(H-5)}$$

In other words, the probability of occurrence of any basic event within a given common-cause component group is assumed to depend only on the number and not on the specific components in that basic event.

With the symmetry assumption, and using the notation just introduced, the system failure probability can be written as:

$$Q_s = 3(Q_1)^2 + 3Q_2 + Q_3 \quad \text{(H-6)}$$

Generalizing, for a system with m components,

$Q_k^m \equiv$ probability of a common-cause basic event involving k specific components in a common-cause component group of size m ($1 \leq k \leq m$).

The model that uses Q_k^m s to calculate system failure probability is called the basic parameter model [D-1].

H-3. ALPHA-FACTOR MODEL

For several practical reasons, it is often more convenient to rewrite Q_k^ms in terms of other more easily quantifiable parameters. For this purpose, a parametric model known as the alphafactor model is recommended [H-1]. Reasons for this choice are that the alpha-factor model:

- Is a multi-parameter model which can handle any redundancy level
- Is based on ratios of failure rates, which makes the assessment of its parameters easier when no statistical data are available
- Has a simpler statistical model
- Produces more accurate point estimates as well as uncertainty distributions compared to other parametric models that have the above properties (e.g., the multiple-Greek-letter model).

The alpha-factor model develops CCF frequencies from a set of failure ratios and the total component failure rate. The parameters of the model are:

$Q_t \equiv$ total failure frequency of each component due to all independent and common- cause events.

$\alpha_k \equiv$ fraction of the total frequency of failure events that occur in the system and involve failure of k components due to a common cause.

Using these parameters, depending on the assumption regarding the way the redundant components of the systems in the database are tested (as part of the data collection effort), the frequency of a common-cause basic event involving failure of k components in a system of m components is given by:

- For a staggered testing scheme, in which components are tested one at a time in fixed intervals:

$$Q_k^m = \frac{1}{\binom{m-1}{k-1}} \alpha_k Q_t \quad \text{(H–7)}$$

- For a non-staggered testing scheme, in which components in a group are tested simultaneously:

$$Q_k^m = \frac{k}{\binom{m-1}{k-1}} \frac{\alpha_k}{\alpha_t} Q_t \quad \text{(H-8)}$$

where the binomial coefficient is given by:

$$\binom{m-1}{k-1} = \frac{(m-1)!}{(k-1)!(m-k)!} \quad \text{(H-9)}$$

and

$$\alpha_t = \sum_{i=1}^{m} k\alpha_i \quad \text{(H-10)}$$

As an example, the probabilities of the basic events of the example three-component system are written as (assuming staggered testing):

$$Q_1^3 = \alpha_1 Q_t \quad \text{(H-11)}$$

$$Q_2^3 = \frac{1}{2}\alpha_2 Q_t \quad \text{(H-12)}$$

$$Q_3^3 = \alpha_3 Q_t \quad \text{(H-13)}$$

Therefore, the system unavailability can now be written as:

$$Q_s = 3(\alpha_1 Q_t)^2 + \frac{3}{2}\alpha_2 Q_t + \alpha_3 Q_t \quad \text{(H-14)}$$

Note that the staggered versus non-staggered assumptions are applicable for parameter estimation as part of the data collection activities. During modeling activities, the typical CCF model to be used will be that of non-staggered testing.

H-4. ESTIMATION OF COMMON-CAUSE-BASIC EVENT PROBABILITIES

The objective of this step is to estimate the common-cause basic event probabilities or parameters of the model used to express these probabilities. Ideally, parameter values are estimated based on actual field experience. The most relevant type of data would be the system-specific data. However, due to the rarity of system-specific common-cause events a search will usually not produce statistically significant data. In almost all cases, parameter estimation will have to include experience from other systems, i.e., generic data. In some cases even the generic data may be unavailable or insufficient. Data might be obtained from various sources including:

- Industry-based generic data
- System-specific data records
- Generically classified CCF event data and parameter estimates (reports and computerized databases).

Only a few industries have developed databases for CCF events. These include nuclear power and, to a lesser extent, aerospace.

The problem of data scarcity can be addressed at least in part by applying a method for extracting information from partially relevant data based on using the impact-vector method and Bayesian techniques [H-1]. This is done through a two-step process:

8. Generic Analysis: Analysis of occurrences of CCFs in various systems in terms of their causes, coupling factors, as well as the level of impact (i.e., the number and nature of component failures observed).
9. System-Specific Analysis: Re-evaluation of the generic data for applicability and relevance to the specific system of interest.

The specific techniques are described in [H-1]. In the following it is assumed that the statistical data needed for the estimation of CCF model parameters are developed by following the referenced procedure or a similar one.

Once the impact vectors for all the events in the database are assessed for the system being analyzed, the number of events in each impact category can be calculated by adding the corresponding elements of the impact vectors. The process results in:

n_k = total number of basic events involving failure of k similar components, k=1,…,m

Event statistics, n_k, are used to develop estimates of CCF model parameters. For example, the parameters of the alpha-factor model can be estimated using the following maximum likelihood estimator (MLE):

$$\hat{\alpha}_k = \frac{n_k}{\sum_{j=1}^{m} n_j} \quad \text{(H-15)}$$

For example, consider a case where the analysis of failure data for a particular two-out-of-three system reveals that of a total of 89 failure events, there were 85 single failures, 3 double failures, and 1 triple failure, due to common cause. Therefore, the statistical data base is {n_1 = 85, n_2 = 3, n_3 = 1}. Based on the equation for the MLE:

$$\alpha_1 = \frac{n_1}{n_1+n_2+n_3} = \frac{85}{89} = 0.955 \quad \text{(H-16)}$$

$$\alpha_2 = \frac{n_2}{n_1+n_2+n_3} = \frac{3}{89} = 0.034 \quad \text{(H-17)}$$

$$\alpha_3 = \frac{n_3}{n_1+n_2+n_3} = \frac{1}{89} = 0.011 \quad \text{(H-18)}$$

Table H-1 provides a set of estimators. The estimators presented in Table H-1 are the MLEs and are presented here for their simplicity. The mean values obtained from probability distribution characterizing uncertainty in the estimated values are more appropriate for point value quantification of system unavailability. Bayesian procedures for developing such uncertainty distributions are presented in [H-1, H-2].

Table H-1 displays two sets of estimators developed based on assuming different testing schemes. Depending on how a given set of redundant components in a system is tested (demanded) in staggered or non-staggered fashion, the total number of challenges that various combinations of components are subjected to is different. This needs to be taken into account in the exposure (or success) part of the statistics used, affecting the form of the estimators. The details of why and how the estimators are affected by testing schedule are provided in [H-1].

Table H-1. Simple point estimators for various CCF parametric models.

Method	Non-Staggered Testing[a]	Staggered Testing[a]	Remarks
Basic parameter	$Q_k^m = \frac{n_k}{\binom{m}{k} N_D}$ $k = 1, \ldots m$	$Q_k^m = \frac{n_k}{m\binom{m}{k} N_D}$ $k = 1, \ldots m$	For time-based *m* failure rates, replace system demands (N_D) with total system exposure time T.
Alpha-factor	$\alpha_k^m = \frac{n_k}{\sum_{j=1}^{m} n_j}$ $k = 1, \ldots m$	Same as non-staggered case	—

a. N_D is the total number of tests or demands on a system of m components.

H-5. GENERIC PARAMETER ESTIMATES

For cases where no data are available to estimate CCF model parameters, generic estimates based on parameter values developed for other components, systems, and applications may be used as screening values. The average value of these data points is B=0.1 (corresponding to an alpha-factor of 0.05 for a two-component system). However, values for specific components range about this mean by a factor of approximately 2.

These values are in fact quite typical and are also observed in CCF data collection efforts in some other industries

H-6. TREATMENT OF UNCERTAINTIES

Estimation of model parameters involves uncertainties that need to be identified and quantified. A broad classification of the types and sources of uncertainty and potential variabilities in the parameter estimates is as follows:

10. Uncertainty in statistical inference based on limited sample size.
11. Uncertainty due to estimation model assumptions. Some of the most important assumptions are:
    A. Assumption about applicable testing scheme (i.e., staggered *versus* non-staggered testing methods).
    B. Assumption of homogeneity of the data generated through specializing generic data to a specific system.
12. Uncertainty in data gathering and database development. These include:
    A. Uncertainty because of lack of sufficient information in the event reports, including incompleteness of data sources with respect to number of failure events, number of system demands, and operating hours.
    B. Uncertainty in translating event characteristics to numerical parameters for impact vector assessment (creation of generic database).
    C. Uncertainty in determining the applicability of an event to a specific system design and operational characteristics (specializing generic database for system-specific application).

The role of uncertainty analysis is to produce an epistemic probability distribution of the CCF frequency of interest in a particular application, covering all relevant sources of uncertainty from the above list. Clearly, some of the sources or types of uncertainty may be inapplicable, depending on the intended use of the CCF parameter and the form and content of the available database. Also, methods for handling various types of uncertainty vary in complexity and accuracy. A comprehensive coverage of the methods for assessing uncertainty distribution for the parameters of various CCF models is provided in [D-1].

H-7. REFERENCES

H-1. Mosleh, A., et al., *Procedures for Treating Common Cause Failures in Safety and Reliability Studies*, U.S. Nuclear Regulatory Commission and Electric Power Research Institute, NUREG/CR-4780, and EPRI NP-5613. Volumes 1 and 2, 1988.

H-2. Dezfuli, H., et al., “Bayesian Inference for NASA Probabilistic Risk and Reliability Analysis,” NASA-SP-2009-569, 2009.

Appendix I

Human Reliability

I-1. HUMAN RELIABILITY ANALYSIS METHODS

I-1.1 General Overview of Quantification Approaches

Human reliability analysis (HRA) methods serve the twofold purpose: to classify the sources of errors qualitatively, and to estimate the human error probability (HEP). Error classification serves as the basis for quantification. Of the roughly 60 HRA methods created, most are centered on quantification [I-1]. Boring [I-2] proposed the following ways of classifying HRA quantification methods:

- *Scenario Matching Methods:* This approach, used by the original HRA method, the technique for human error rate prediction (THERP) [I-3], entails matching the human failure event (HFE) to the best fitting example scenario in a lookup table and using the HEP associated with that template event as the basis for quantification. See Table I-1(a).
- *Decision-Tree Methods:* Methods like the cause-based decision tree (CBDT) [I-4] follow a decision tree (similar to an event tree), which guides the quantification along a number of predefined analysis decision points. See Table I-1(b).
- *Performance Shaping Factor (PSF) Adjustment Methods:* In these methods, exemplified by approaches like the standardized plant analysis risk-HRA (SPAR-H) method [I-5], the PSFs serve as multipliers on nominal error rates. For example, a PSF with a negative influence would serve to increase the HEP over a nominal or default error rate. A list of PSFs and associated multipliers is provided by the method. See Table I-1(c).
- *Expert Estimation Methods:* In these approaches, subject matter experts including risk analysts will estimate the likelihood of the HFEs. A technique for human error analysis (ATHEANA) [I-6] uses a structured expert estimation approach to arrive at HEPs. Such approaches often provide anchor values for quantification to assist subject matter experts in producing the relevant HEP, but the specific method used to derive the HEP and the factors that may influence the quantification are largely left to the subject matter experts. Because expert estimation methods typically do not specify how to decompose the factors shaping the quantification but rather look at the HFE as a whole, they are often referred to as holistic approaches [I-7]. See Table I-1(d).

The wide availability of HRA methods may leave the analyst overwhelmed at which methods to select for which applications. Recent method comparisons exist for nuclear (e.g., [I-1, I-8, I-9]), and they provide helpful benchmarks in considering the advantages and disadvantages of each method. For example, the National Aeronautics and Space Administration's (NASA's) method [I-9] serves as a helpful template for downselecting HRA methods. Across multiple selection criteria, NASA selected four primary HRA methods to be used individually or in combination. Table I-2 lists the four methods selected by NASA and a summary of their primary strengths and weaknesses in a generalized form (e.g., without consideration of specific NASA domain applications). While this downselection is helpful, it does not necessarily represent optimal methods with respect to offshore oil applications.

Table I-1. Examples of common HRA quantification approaches.

(a) Scenario matching lookup table from THERP [I-3], which provides the HEP and the error factor (EF) for uncertainty.	(b) Decision tree from CBDT [I-4] provides HEPs for the event-tree end states.

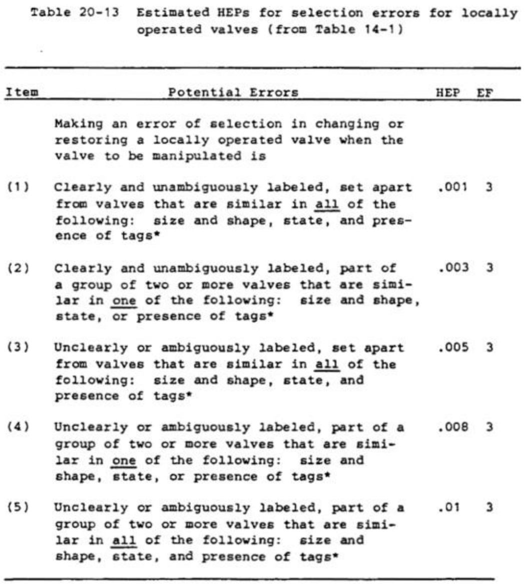


Table 20-13 Estimated HEPs for selection errors for locally operated valves (from Table 14-1)

Item	Potential Errors	HEP	EF
	Making an error of selection in changing or restoring a locally operated valve when the valve to be manipulated is		
(1)	Clearly and unambiguously labeled, set apart from valves that are similar in all of the following: size and shape, state, and presence of tags*	.001	3
(2)	Clearly and unambiguously labeled, part of a group of two or more valves that are similar in one of the following: size and shape, state, or presence of tags*	.003	3
(3)	Unclearly or ambiguously labeled, set apart from valves that are similar in all of the following: size and shape, state, and presence of tags*	.005	3
(4)	Unclearly or ambiguously labeled, part of a group of two or more valves that are similar in one of the following: size and shape, state, or presence of tags*	.008	3
(5)	Unclearly or ambiguously labeled, part of a group of two or more valves that are similar in all of the following: size and shape, state, and presence of tags*	.01	3

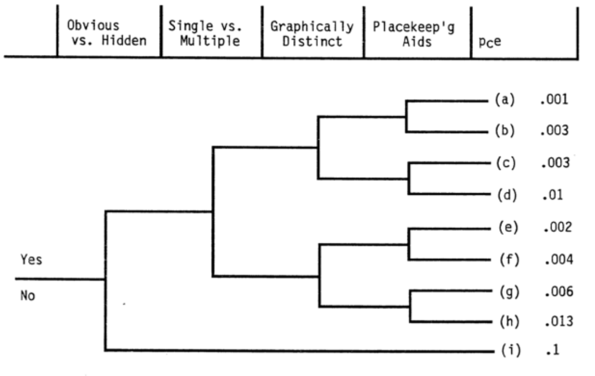
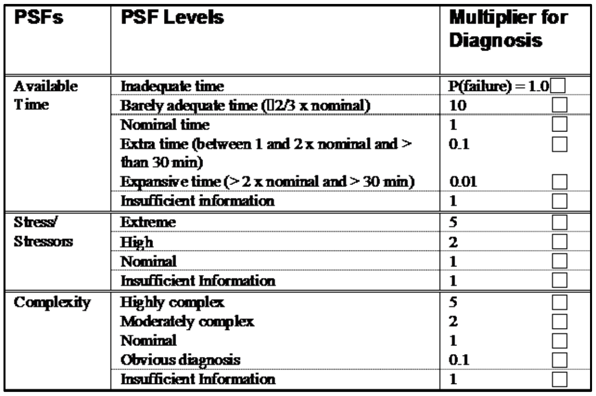


Figure 4-6. Decision tree Representation of P_{Ce}, Skip a Step in Procedure

(c) Example PSF multipliers on the nominal HEP (0.001) for diagnosis tasks in SPAR-H [I-5].

(d) Example anchor HEPs for expert elicitation in ATHEANA [I-6].



PSFs	PSF Levels	Multiplier for Diagnosis
Available Time	Inadequate time	P(failure) = 1.0 □
	Barely adequate time (≈2/3 x nominal)	10 □
	Nominal time	1 □
	Extra time (between 1 and 2 x nominal and > than 30 min)	0.1 □
	Expansive time (> 2 x nominal and > 30 min)	0.01 □
	Insufficient information	1 □
Stress/ Stressors	Extreme	5 □
	High	2 □
	Nominal	1 □
	Insufficient Information	1 □
Complexity	Highly complex	5 □
	Moderately complex	2 □
	Nominal	1 □
	Obvious diagnosis	0.1 □
	Insufficient Information	1 □

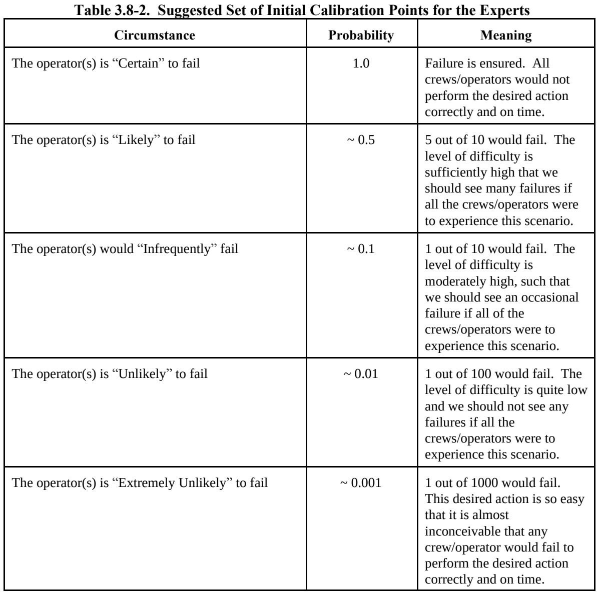


Table 3.8-2. Suggested Set of Initial Calibration Points for the Experts

Circumstance	Probability	Meaning
The operator(s) is "Certain" to fail	1.0	Failure is ensured. All crews/operators would not perform the desired action correctly and on time.
The operator(s) is "Likely" to fail	~ 0.5	5 out of 10 would fail. The level of difficulty is sufficiently high that we should see many failures if all the crews/operators were to experience this scenario.
The operator(s) would "Infrequently" fail	~ 0.1	1 out of 10 would fail. The level of difficulty is moderately high, such that we should see an occasional failure if all of the crews/operators were to experience this scenario.
The operator(s) is "Unlikely" to fail	~ 0.01	1 out of 100 would fail. The level of difficulty is quite low and we should not see any failures if all the crews/operators were to experience this scenario.
The operator(s) is "Extremely Unlikely" to fail	~ 0.001	1 out of 1000 would fail. This desired action is so easy that it is almost inconceivable that any crew/operator would fail to perform the desired action correctly and on time.

Table I-2. The four HRA methods selected for NASA use.

Method	Approach	Strengths	Weaknesses
THERP	Lookup table	Widely used original HRA method. THERP specifies a complete process model for HRA. It has good coverage of errors related to human actions.	Little coverage of cognitive factors. Method may have limited generalizability beyond the nuclear-specific human interactions in the lookup tables.
CREAM[a]	Task types (lookup table) and PSF	Good coverage of cognitive factors and detailed task	Method is complex in practice (e.g., involving many steps for

	multipliers	decomposition approach for qualitative insights into errors.	basic quantification) and tends to produce similar HEPs.
NARA[b]	Task types (lookup table) and PSF multipliers	Good use of human factors literature as data source to validate HEPs for task types.	The task types are aligned to nuclear power plant operations, and specialized variants need to be developed for air traffic control and rail domains. The method remains proprietary.
SPAR-H	PSF multipliers	Simplified method that can be used without extensive HRA background. PSFs allow generalizability beyond predefined task types.	Quantification-only approach that assumes HFEs defined in the probabilistic risk assessment (PRA). PSF multipliers are not calibrated to non-nuclear techniques.

a. Cognitive reliability error analysis method [I-10]

b. Nuclear action reliability assessment [I-11]

I-1.2 HRA Methods for Oil and Gas

Two HRA methods have been developed specifically for oil and gas, and they are briefly noted below.

I-1.2.1 Barrier and Operational Risk Analysis

Despite information suggesting major accident sequences may be attributed to several risk influencing factors classified as technical, human, operational and organizational, the majority of quantitative risk analyses of offshore oil and gas production platforms has been directed at technical safety systems. The barrier and operational risk analysis (BORA) of hydrocarbon releases (BORA-Release) is a method for carrying out the qualitative and quantitative risk analysis of platform specific hydrocarbon release frequency. In finer detail, the method assesses the effect of risk reducing measures and risk increasing changes within operations. BORA affords the ability to analyze both the effect of safety barriers put in place to impede the release of hydrocarbons as well as how platform specific conditions such as the aforementioned technical, human, operational and organizational factors influence the performance of the barrier [I-12]. Analysis of hydrocarbon release risk via the BORA method is executed with the use of barrier block diagram/event trees, fault trees, and risk influence diagrams.

The BORA-Release method is made up of eight steps:

1. Development of a basic risk model including release scenarios
2. Modeling for the performance of safety barriers
3. Assignment of industry average probabilities/frequencies and risk quantification based on these probabilities/frequencies
4. Development of risk influence diagrams
5. Scoring of risk influencing factors
6. Weighting of risk influencing factors
7. Adjustment of industry average probabilities/frequencies

8. Recalculation of the risk in order to determine the platform specific risk related to hydrocarbon release.

Many of these steps overlap the basic HRA process model described previously. BORA focuses on the breakdown of barriers designed as part of defense in depth to prevent accidents in oil and gas production facilities. These barriers, however, may omit many of the HFEs that can precipitate accidents at the facility. HRAs centered on barriers may overlook important precursors to many types of accidents. Additionally, BORA's emphasis on prevention of accidents may limit some of its application as a risk analytic tool for as-built systems and processes.

I-1.2.2 Petro-HRA

The Norwegian Research Council and that Norwegian state oil company, Statoil, have recently sponsored development of an HRA method to aid human factors analysts in completing HRAs for oil and gas applications. The approach, named the Petro-HRA method [I-13], features seven steps that mirror much of what is outlined in [I-14]:

1. Scenario definition
2. Qualitative data collection
3. Task analysis
4. Human error identification
5. Human error modeling
6. Human error quantification
7. Human error reduction.

Quantification in the Petro-HRA method is SPAR-H, offering some refinement to PSFs and multipliers to make them more oil and gas industry specific. SPAR-H was selected as the basis method because other HRA methods that had been used were found to generate unreasonably high HEPs or have low interrater reliability [I-15]. Because SPAR-H is primarily a quantification approach, additional guidance was developed to aid analysts in completing the qualitative portion of HRA, including translating a task analysis to HFEs when they are not already defined by a PRA. Because HRAs are performed to support the safety evaluation of new technologies in the Norwegian oil industry, guidance is provided to improve the system design or operations process to minimize human errors.

I-2. EXAMPLE HUMAN RELIABILITY ANALYSIS FOR WELL KICK

I-2.1 Example SPAR-H Analysis

Here, we demonstrate SPAR-H as a simplified method to help understand how to quantify an HFE. This example should not imply endorsement of SPAR-H over any other method. An important aspect of the HRA should, in fact, be the selection of a particular HRA method. SPAR-H is demonstrated here because it lends itself to a brief description and because it is the basis of the Petro-HRA method.

A SPAR-H quantification requires several steps:

9. Define the HFE
10. Determine the appropriate SPAR-H worksheet
11. Determine the appropriate SPAR-H nominal HEP
12. Evaluate the PSFs

13. Calculate the product of the nominal HEP and the PSF multipliers
14. Apply correction factor for dependence.

These steps are walked through in separate subsections below.

I-2.1.1 Define the HFE

SPAR-H assumes the HFE has been defined in the PRA. For the present purposes, we have characterized two HFEs related to well kick (also Figure I-1 for a simple graphical depiction):

- *HFE_1:* Detection of well kick
- *HFE_2:* Recovery activities after well kick.

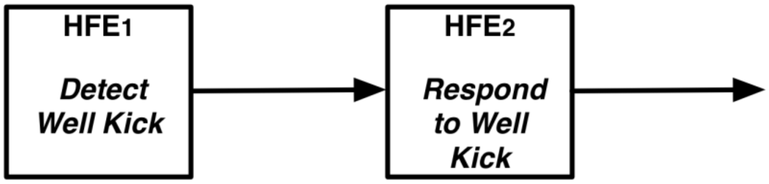


Figure I-1. Example HFEs in sequence.

In reality, the recovery activities after the well kick might consist of many separate HFEs. However, the general context as represented by the PSFs for each of those post well kick activities largely remains the same. Additionally, if there is a failure to detect the well kick, there is obviously little opportunity for recovery actions nor the need to model a second HFE.

I-2.1.2 Determine the Appropriate SPAR-H Worksheet

SPAR-H contains two types of analysis worksheets:

- *At power* (NUREG/CR-6883, Appendix A)
- *Low power and shutdown* (NUREG/CR-6883, Appendix B).

The origin of SPAR-H as an HRA method for nuclear power applications is clear here. The basic difference between these two worksheets involves whether the plant is producing electricity (i.e., at power) or in maintenance or refueling mode (i.e., low power and shutdown). It is assumed that there is more opportunity for high consequence events and tighter time windows to take recovery actions during at-power operations. An offshore analogy for at power would be during drilling activities.

For the well kick scenarios, we assume the SPAR-H at-power worksheets are applicable.

I-2.1.3 Determine the Appropriate SPAR-H Nominal HEP

The SPAR-H worksheets for at-power and low-power-and-shutdown each have two task types that are modeled. The task types determine the nominal or default HEP for the HFE:

- *Diagnosis:* This HFE primarily involves cognitive activities such as monitoring or decision-making. The nominal HEP for diagnosis HFEs is 1E-2 (0.01).
- *Action:* This HFE primarily involves carrying out physical activities such as manipulating equipment. The nominal HEP for action HFEs is 1E-3 (0.001).

Because an HFE may involve a series of activities by the human involved, it is not uncommon for the HFE to be classified as both diagnosis and action. In that case, the *joint HFE* can logically be thought to occur due to diagnosis OR action errors. Mathematically, this means that the nominal HEPs for diagnosis and action are added together.

In our well kick example, both HFEs involve diagnosis and action components, since they require cognitive monitoring and decision-making and interaction with equipment.

I-2.1.4 Evaluate the PSFs

SPAR-H uses nominal HEPs to represent basic diagnosis and action tasks performed within the HFE. These nominal HEPs are then modified using multipliers corresponding to different levels of influence of the PSFs. SPAR-H makes use of eight PSFs, encompassing:

- Available time to complete the task (which is independent of any time pressure the personnel may experience)
- Internal stress and external stressors
- The complexity of the task and scenario
- The experience and training of the personnel completing the tasks under analysis
- The procedures—either written or oral—to guide the personnel in completing the task
- The ergonomics of the system being used and the human-machine interfaces available to the personnel
- The fitness for duty—including degraded fitness due to fatigue of long-duration events—of the personnel completing the task
- Work processes, including organizational factors, command and control, and communications

Generally, the SPAR-H PSFs can have three types of effects:

15. Negative: A negative effect means that the PSF decreases human reliability. For example, to denote the negative effect of available time would mean to suggest that there was inadequate time available to complete the task.
16. Nominal: A nominal effect means that the default applies. Nominal time, for example, suggests that there’s adequate time to complete the task without undue time pressure or extra time.
17. Positive: A positive effect means that the PSF increases human reliability. Positive available time means that there is extra time over what is needed to accomplish the task.

In the absence of information to inform the assignment, the analyst would denote “inadequate information,” which simply assigns a nominal value.

To assign SPAR-H PSFs, it is useful for the human reliability analyst to consult with an operations specialist to answer the following questions:

- Which personnel are involved in this task?
- What indicators are available for the task?
- What are the timing constraints?
- Are personnel trained/ do they have experience on the task?
- What’s needed to perform this task successfully?
- What can go wrong?
- What could influence personnel performance in terms of actions or decision-making?

For our two example HFEs, the following PSF effects could be noted:

- For detection of the well kick (HFE_1), the time available will vary from situation to situation, but once a kick occurs, there is a limited window of time before the formation fluid reaches the blowout preventer. As the available time erodes, and the ability of the drilling crew to respond decreases proportionately to the decreasing time window. It may be assumed that, the available time to detect will adversely affect the HEP. The clock is ticking, so to speak, which can only operate negatively on the outcome of the event. All other PSFs are assumed to be nominal.
- The detection of a well kick triggers a change: response actions are needed in order to prevent a blowout (HFE_2). This operational shift will generally result in multiple elevated negative PSFs relative to nominal or normal operations. The time window is closing, but there may also be elevated negative stress and complexity, potentially diminished levels of experience for this type of situation, and potentially poor to incomplete procedures. Underlying the situation, negative work processes such as breakdowns in communication, coordination, or command and control may also manifest.

While detection of the well kick (HFE_1) can be seen as a mostly nominal influence of the PSFs, the transition to emergency operations to prevent blowout (HFE_2) will likely invoke multiple negative PSFs.

I-2.1.5 Calculate the Product of the Nominal HEP and the PSF Multipliers

When negative, nominal, or positive effects of PSFs have been determined, these are matched to the appropriate level in the SPAR-H PSF multiplier tables. If there is a negative or positive effect of a PSF, this phase involves determining the degree of that effect, which corresponds to a multiplier. A summary of SPAR-H multiplier assignments for the well kick detection and response HFEs is found in Table I-3. For the response HFE, three slightly negative PSFs—available time, stress, and complexity—are assumed.

Table I-3. SPAR-H table showing possible multiplier assignments for a generic well kick detection and response.

PSFs	PSF Levels	HFE_1: Generic Well Kick Detection		HFE_2: Generic Response	
		Multiplier for Diagnosis	**Multiplier for Action**	**Multiplier for Diagnosis**	**Multiplier for Action**
Available time	Inadequate time	P(failure) = 1.0	P(failure) = 1.0	P(failure) = 1.0	P(failure) = 1.0
	Barely adequate time	10	10	10	10
	Nominal time	1	1	1	1
	Extra time	0.1	0.1	0.1	0.1
	Expansive time	0.01	0.01	0.1 to 0.01	0.01
	Insufficient information	1	1	1	1
Stress/stressors	Extreme	5	5	5	5
	High	2	2	2	2
	Nominal	1	1	1	1
	Insufficient information	1	1	1	1
Complexity	Highly complex	5	5	5	5
	Moderately complex	2	2	2	2
	Nominal	1	1	1	1
	Obvious diagnosis	0.1	N/A	0.1	N/A
	Insufficient information	1	1	1	1
Experience/ training	Low	10	3	10	3
	Nominal	1	1	1	1

	High	0.5	0.5	0.5	0.5
	Insufficient information	1	1	1	1
Procedures	Not available	50	50	50	50
	Incomplete	20	20	20	20
	Available, but poor	5	5	5	5
	Nominal	1	1	1	1
	Diagnostic/symptom oriented	0.5	N/A	0.5	N/A
	Insufficient information	1	1	1	1
Ergonomics/ HMI	Missing/ misleading	50	50	50	50
	Poor	10	10	10	10
	Nominal	1	1	1	1
	Good	0.5	0.5	0.5	0.5
	Insufficient information	1	1	1	1
Fitness for duty	Unfit	P(failure) = 1.0	P(failure) = 1.0	P(failure) = 1.0	P(failure) = 1.0
	Degraded fitness	5	5	5	5
	Nominal	1	1	1	1
	Insufficient information	1	1	1	1
Work processes	Poor	2	5	2	5
	Nominal	1	1	1	1
	Good	0.8	0.5	0.8	0.5
	Insufficient information	1	1	1	1

The basic HEP is defined in SPAR-H as the nominal HEP multiplied by the product of all PSF multipliers:

$$\text{Basic HEP} = \text{Nominal HEP} \times \prod \text{PSF Multipliers}$$

For HFE_1 related to well kick detection, the PSF is calculated separately for diagnosis and action:

$$HFE_1 \text{ Diagnosis Basic HEP} = 1E\text{-}2 \times 10 \times 1 \times 1 \times 1 \times 1 \times 1 \times 1 \times 1 = 1E\text{-}1 = 0.1$$

$$HFE_1 \text{ Action Basic HEP} = 1E\text{-}3 \times 10 \times 1 \times 1 \times 1 \times 1 \times 1 \times 1 \times 1 = 1E\text{-}2 = 0.01$$

The joint basic HEP is simply the sum of the diagnosis and action basic HEPs:

$$HFE_1 \text{ Joint Basic HEP} = \text{Diagnosis Basic HEP} + \text{Action Basic HEP}$$

$$= 1E\text{-}1 + 1E\text{-}2 = 1.1E\text{-}1 = 0.11$$

The same equation applies to HFE_2 related to the response to the well kick, but with one exception. Because it is possible to have a resultant HEP greater than 1.0 when there are more than three negative HEPs, SPAR-H prescribes a correction factor:

$$\text{Corrected Basic HEP} = \frac{\text{Nominal HEP} \times \Pi \text{ PSF Multipliers}}{\text{Nominal HEP} \times (\Pi \text{ PSF Multipliers } \; 1) + 1}$$

Thus, we first calculate the product of the PSF multipliers, which in this case is identical for the diagnosis and action tasks:

$\prod$ PSF Multipliers = 10 × 2 × 2 × 1 × 1 × 1 × 1 × 1 = 40

This product is then applied in the corrected basic HEP equation for diagnosis and action:

$$\text{HFE}_2 \text{ Corrected Diagnosis Basic HEP} = \frac{1\text{E-}2 \times 40}{1\text{E-}2 \times (40 - 1) + 1} = 0.288$$

$$\text{HFE}_2 \text{ Corrected Action Basic HEP} = \frac{1\text{E-}3 \times 40}{1\text{E-}3 \times (40 - 1) + 1} = 0.0385$$

The joint basic HEP for HFE_2 is calculated by adding the two basic HEPs:

HFE_2 Joint Basic HEP = 0.288 + 0.0385 = 0.326

There is nearly a threefold increase in the basic HEP between HFE_1 and HFE_2 due to the increased effects of negative PSFs for stress and complexity between well kick detection and response.

I-2.1.6 Apply Correction Factor for Dependence

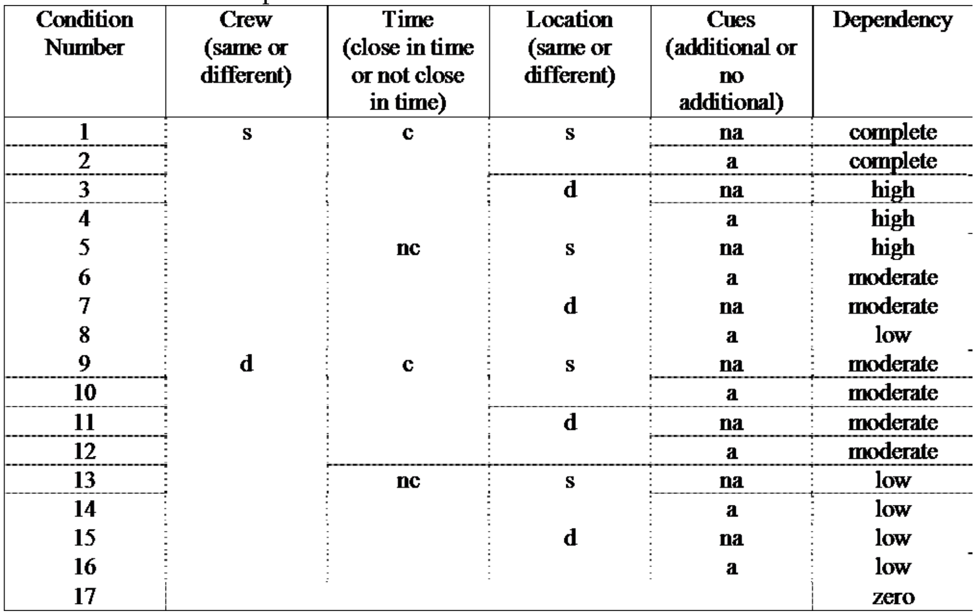
In the final stage of SPAR-H quantification, a correction factor is applied for dependence. Dependence in SPAR-H means that the second or subsequent HFE in sequence may result in greater likelihood of human error. If appropriate, a correction factor is applied to the basic HEP.

For sequences of two or more HFEs, SPAR-H considers four factors that influence dependence:

18. Same (s) or different (d) *crew* between the HFEs
19. Close (c) in *time* or not close (nc) in time between the HFEs
20. Same (s) or different (d) *location* between the HFEs
21. Additional (a) or no additional (na) *cues* (i.e., information) between the HFEs.

The more the HFEs share crew, time, location, and cues, the more likely there is to be dependence between them. SPAR-H uses a dependency condition table (see Table I-4) to classify dependence along a scale from *Zero*, *Low*, *Moderate*, *High*, to *Complete*.

Table I-4. SPAR-H dependence table.



Condition Number	Crew (same or different)	Time (close in time or not close in time)	Location (same or different)	Cues (additional or no additional)	Dependency
1	s	c	s	na	complete
2				a	complete
3			d	na	high
4				a	high
5		nc	s	na	high
6				a	moderate
7			d	na	moderate
8				a	low
9	d	c	s	na	moderate
10				a	moderate
11			d	na	moderate
12				a	moderate
13		nc	s	na	low
14				a	low
15			d	na	low
16				a	low
17					zero

HFE$_1$ is the first HFE in the sequence and by definition does not have dependence. We assume HFE$_2$ to have somewhat different crew responding to the well kick. HFE$_2$ follows closely in time, has the same location, but has additional cues. The resultant dependence level as traced (d-c-s-a) in Table I-4 is *moderate dependence*.

The conditional HEP is the basic HEP corrected for dependence. SPAR-H features the following equations for levels of the conditional HEP:

- *Zero Dependence:* Conditional HEP = Basic HEP
- *Low Dependence:* Conditional HEP = (1 + 19 × Basic HEP) / 20
- *Moderate Dependence:* Conditional HEP = (1 + 6 × Basic HEP) / 7
- *High Dependence:* Conditional HEP = (1 + Basic HEP) / 2
- *Complete Dependence:* Conditional HEP = 1.0.

For HFE$_2$, assuming moderate dependence, we have:

HFE$_2$ Conditional HEP = (1 + 6 × 0.326) / 7 = 0.422

Moderate dependence resulted in the HEP for HFE$_2$ increasing by nearly 0.1 in our example.

Using the SPAR-H method, we quantified the HEPs for the two HFEs, arriving at:

- Detect well kick: HEP$_{HFE1}$ = 0.11
- Respond to well kick: HEP$_{HFE2}$ = 0.422.

More specific information, such as the accident report related to the Macondo accident, would allow greater precision of the PSF assignments beyond the general assignments made here.

As noted, the SPAR-H approach employed here is almost identical to the quantification step in the Petro-HRA method, and a similar result can be expected.

A final note on SPAR-H is that it only provides the HEP, not a measure of uncertainty. The HEP is calculated using the constrained noninformative prior, a method for calculating uncertainty parameters assuming a single input parameter on a gamma distribution. Some PRA software feature the ability to calculate the uncertainty in SPAR-H if required by the analyst.

I-3. REFERENCES

I-1. Bell, B. J. and J. Holroyd, 2009, *Review of human reliability assessment methods*, RR679, Buxton, UK: Health and Safety Executive.

I-2. Boring, R. L., 2015, Adapting Human Reliability Analysis from Nuclear Power to Oil and Gas Applications, in Podifilini et al. (Eds.), *Safety and Reliability of Complex Engineered Systems*, ESREL 2015, pp. 2853-2860, London: Taylor & Francis.

I-3. Swain, A. D. and H. E. Guttmann, 1983, *Handbook of Human Reliability Analysis with Emphasis on Nuclear Power Plant Applications*, Final Report, NUREG/CR-1278, Washington, DC: U.S. Nuclear Regulatory Commission.

I-4. *An Approach to the Analysis of Operator Actions in Probabilistic Risk Assessment,* TR-100259, Palo Alto: Electric Power Research Institute, 1992.

I-5. Gertman, D., H. Blackman, J. Marble, J. Byers, and C. Smith, 2005, *The SPAR-H Human Reliability Analysis Method*, NUREG/CR-6883, Washington, DC: U.S. Nuclear Regulatory Commission.

I-6. *Technical Basis and Implementation Guidelines for A Technique for Human Event Analysis (ATHEANA)*, NUREG-1624, Rev. 1, Washington, DC: U.S. Nuclear Regulatory Commission, 2000.

I-7. Boring, R. L. and D. I. Gertman, 2005, “Atomistic and holistic approaches to human reliability analysis in the U.S. power industry,” *Journal of the Safety and Reliability Society*, Vol. 25, No. 2, pp. 21–37.

I-8. Forester, J., A. Kolaczkowski, E. Lois, and D. Kelly, 2006, *Evaluation of Human Reliability Analysis Methods Against Good Practices*, NUREG-1842, Washington, DC: U.S. Nuclear Regulatory Commission.

I-9. Chandler, F. T., Y. H. J. Chang, A. Mosleh, J. L. Marble, R. L. Boring, and D. I. Gertman, 2006, *Human Reliability Analysis Methods: Selection Guidance for NASA*, Washington, DC: NASA Office of Safety and Mission Assurance Technical Report.

I-10. Hollnagel, E., 1998, *Cognitive Reliability and Error Analysis Method – CREAM*, 1st ed., Oxford, England: Elsevier Science.

I-11. Kirwan, B., H. Gibson, R. Kennedy, J. Edmunds, G. Cooksley, and I. Umbers, 2005, “Nuclear Action Reliability Assessment (NARA): A data-based HRA tool,” *Safety & Reliability Journal*, Vol. 25, No. 2.

I-12. Aven, T., S. Sklet, and J. E. Vinnem, 2006, “Barrier and operational risk analysis of hydrocarbon releases (BORA-Release), Part I. Method description,” *Journal of Hazardous Materials*, Vol. A137, pp. 681–691.

I-13. Bye, A., K. Laumann, C. Taylor, M. Rasmussen, S. Øie, K. van de Merwe, K., Øien, R. Boring, N. Paltrinieri, I. Wæro, S. Massaiu, and K. Gould, K, 2017, *The Petro-HRA Guideline,* Lysaker, Norway: The Research Council of Norway.

I-14. *Draft Guide for Incorporating Human Reliability Analysis into Probabilistic Risk Assessments for Nuclear Power Generating Stations and Other Nuclear Facilities, IEEE-1082/D8,* New York City: Institute for Electrical and Electronics Engineers, 2017.

I-15. Gould, K. S., A. J. Ringstad, and K. van de Merwe, 2012, "Human reliability analysis in major accident risk analyses in the Norwegian petroleum industry," *Proceedings of the 56th Annual Meeting of the Human Factors and Ergonomics Society,* 2016-2020.

Appendix J

Failure-Space-Based Importance Measures

It will be convenient in the following to refer to a formula for the risk metric (e.g., top-event probability) with respect to which the measures are being calculated:

R = f(x_1,x_2 ,..., x_i ,x_j ,..., x_n)

where x_k is the kth basic event, having probability p_k , and R_0 means "R evaluated with all p's set to their nominal values." It is an aid to understanding the following formulas to bear in mind that the reduced Boolean expression for the minimal cut sets maps simply into an arithmetic expression for the rare event approximation to top-event probability, and the same idea applies to any subset of the minimal cut sets. The notation "|" means "given": A|B means "A given B."

J-1. FUSSELL-VESELY AND RISK REDUCTION WORTH IMPORTANCE MEASURES

The Fussell-Vesely (F-V) importance measure is used to determine the importance of individual minimal cut sets containing basic event x_i to the risk. F-V of event x_i is given by:

$$I_{x_i}^{FV} = \frac{Pr\left(\bigcup_j MCS_j^{x_i}\right)}{Pr\left(\bigcup_j MCS_j\right)} = \frac{Pr\left(\bigcup_j MCS_j\right)}{R_0}$$

where

I^{FV} is the F-V importance for event x_i,

$Pr(\bigcup_j MCS_j^{x_i})$ is probability of the union of the minimal cut sets containing event x_i ;

$Pr\left(\bigcup_j MCS_j\right) = R_0$ (the probability of the union of ALL of the minimal cut sets) is the baseline risk.

The simple interpretation of the FV is that it is the fraction of total risk involving x_i. Corollary interpretations are (1) that the FV is the conditional probability that at least one minimal cut set containing event x_i will occur, given that the system has failed, or (2) the fraction by which risk would decrease if *Pr(x_i)* were reduced to zero. The latter interpretation points to another way of calculating FV:

$$I_{x_i}^{FV} = \frac{R_0 - R|Pr(x_i) = 0}{R_0}$$

where $R|Pr(x_i) = 0$ is the value of the risk metric when the probability of event x_i is set to zero. In this calculation, in the numerator, we are subtracting off the contribution from minimal cut sets that do NOT contain x_i, leaving the minimal cut sets that DO contain x_i.

The closely related risk reduction worth (RRW) is a measure of the change in risk when a basic event probability (e.g., unavailability of a hardware device) is set to zero. It measures the amount by which risk would decrease if the event would never occur. The RRW measure is calculated as the ratio[l] of the baseline expected risk to the conditional expected risk when the probability of event x_i is set to zero (assuming that the hardware device is "perfect"):

$$I_{x_i}^{RRW} = \frac{R_0}{R|Pr(x_i) = 0}$$

where $I_{x_i}^{RRW}$ is the risk reduction worth for event x_i .

It should be clear that FV and RRW will produce essentially the same ranking: event lists ordered by decreasing FV and decreasing RRW are the same. In fact, it is straightforward to show that:

$$I_{x_i}^{FV} = 1 - \frac{1}{I_{x_i}^{RRW}}.$$

J-2. BIRNBAUM (B) AND RISK ACHIEVEMENT WORTH (RAW)

The B is the rate of change of the expected risk as a result of the change in the probability of an individual event. Mathematically, the B importance of event x_i is:

$$I_{x_i}^{B} = \frac{\partial R}{\partial x_i}.$$

In many cases, B can be calculated as:

$$I_{x_i}^{B} = (R|Pr(x_i) = 1) - (R|Pr(x_i) = 0)$$

where $R|\Pr(x_i) = 1\ (0)$ is the risk metric calculated with Pr(x_i) set to 1 (0).

In general, the B of a basic event x_i does not depend on the probability of x_i; it depends on the probabilities of the other basic events in the cut sets in which x_i appears.

Risk Achievement Worth (RAW) is a measure of the change in risk when the probability of a basic event (e.g., unavailability of a component) is set to unity. Analogously to RRW, the calculation is typically done as a ratio:

$$I_{x_i}^{RAW} = \frac{R|Pr(x_i) = 1}{R_0}.$$

Again analogously to RRW, some probabilistic risk assessment (PRA) codes calculate an interval measure corresponding to RAW, the "risk increase interval," which is the *difference* between the conditional expected risk when event x_i is set to unity, and the baseline risk.

[l] Instead of ratio, some PRA codes calculate "Risk Decrease Interval," which is the *difference* between baseline risk and the conditional risk when event x_i is set to zero.

Both RAW and RRW correspond to drastic sensitivity studies, displaying how much difference it makes when a basic event probability is maximized (RAW) or minimized (RRW). This kind of information points to properties of the model, and perhaps the system; for example, a high RAW can result from a component for which there is relatively little backup, such as a single item that is required to succeed regardless of whether anything else succeeds or fails. But as discussed in [I-1], measures such as RAW are difficult to use in quantitative reasoning processes.

J-3. COMPUTING B, FV, RAW, RRW

It is straightforward to compute FV and RRW within the rare event approximation, given a Boolean expression for the top event in properly reduced form. If the basic event names are replaced by their probability values, AND by multiplication symbols, and OR by addition symbols, one has an expression for top event probability (again, within the rare event approximation). If a more precise answer is required, better approximations can be applied (such as the min cut upper bound).

Strictly speaking, evaluating RAW calls for actually restructuring the expression. Computing the RAW of a basic event calls for setting that event to "TRUE" (typically, the corresponding component to "failed" or perhaps "unavailable") and re-reducing the top-event expression. Consider computing the numerator of the RAW of event A in an expression including:

A*B*C + X*B*C + … .

If we simply set A to a value of 1, we will still include the contribution of X*B*C, which, strictly speaking, we should not. Setting A to "TRUE" and re-reducing leaves us with:

B*C + … , the "X*B*C" having been absorbed.

However, computing B(A) gives us:

R(A=1)-R(A=0) =[B*C + X*B*C + …] – [X*B*C + …] = B*C (plus perhaps other terms).

J-4. DIFFERENTIAL IMPORTANCE MEASURE FOR BASIC EVENTS AND PARAMETERS

The importance measures discussed previously are defined to deal with basic event probabilities *one event at a time*, and, as formulated, they do not reflect the influence of the underlying parameters in the models of event probability: they do not measure the importance of changes that affect component properties or failure modes. They also lack an additive property that some analysts consider desirable. For these reasons, the "differential importance measure (DIM) was introduced.

J-4.1 Definition of DIM

Let R be the risk metric of interest expressed as a function of basic events or fundamental parameters of the PRA model as shown below:

R= f(x_1,x_2 ,..., x_i ,x_j ,..., x_n) where x $_i$ is the generic parameter such as basic event probability of a component x_i or the failure rate of a component x_i .

The differential importance measure of x_i is defined as:

$$I_{x_i}^{DIM} = \frac{dR_{x_i}}{dR} = \frac{\frac{\partial R}{\partial x_i} dx_i}{\sum_j \frac{\partial R}{\partial x_j} dx_j}$$

DIM reflects the fraction of the total change in R due to a change in parameter x $_{\mathrm{i}}$. It can be shown that DIM is additive, that is,

$$I^{DIM}_{x_i \cup x_{j \dots} \cup x_k} = I^{DIM}_{x_i} + I^{DIM}_{x_j} + \cdots + I^{DIM}_{x_k}$$

J-4.2 Calculations of DIM

With respect to calculation of DIM for a parameter of the PRA model, there are two computational inconveniences:

22. The DIM can be calculated only if the expression for the risk is in parametric form, which is not a standard output form generated by the PRA codes.
23. There is no available computer program for use.

However, one can compute DIM for basic events using the F-V and RAW importance measures. The latter measures are often generated by standard PRA codes by applying formulas developed in the previous subsection.

As noted, calculation of DIM deals with change in R (its differential). Since the change depends on how the values assigned to a parameters are varied, DIM can be calculated in different ways. Two possibilities are:

24. Assume a uniform change for all parameters (i.e., $\delta x_i = \delta x_j = \delta x_k \dots$). Under this operation, parameters are ranked according to the effect they produce on R when they undergo small changes that are the same for all. This has meaning when parameters of the model have the same dimensions (e.g., the risk metric is expressed in terms of basic event probabilities only). DIM for parameter x_{i} is calculated as follows:

$$I^{DIM}_{x_i} = \frac{\frac{\partial R}{\partial x_i} dx_i}{\sum_j \frac{\partial R}{\partial x_j} dx_j} = \frac{\frac{\partial R}{\partial x_i}}{\sum_j \frac{\partial R}{\partial x_j}}$$

25. Assume a uniform *percentage* change for all parameters ($\frac{\delta x_i}{x_i} = \frac{\delta x_j}{x_j} = \frac{\delta x_k}{x_k} \dots$). Under this operation, PRA parameters are ranked according to the effect they produce on R when they are changed by the same fraction from their nominal values. This ranking scheme, which is applicable to all analysis conditions, can be calculated from:

$$I^{DIM}_{x_i} = \frac{\frac{\partial R}{\partial x_i} dx_i}{\sum_j \frac{\partial R}{\partial x_j} dx_j} = \frac{\frac{\partial R}{\partial x_i} \frac{dx_i}{x_i} x_i}{\sum_j \frac{\partial R}{\partial x_j} \frac{dx_j}{x_j} x_j} = \frac{\frac{\partial R}{\partial x_i} x_i}{\sum_j \frac{\partial R}{\partial x_j} x_j}$$

The relation between DIM and F-V, RAW, and BM are shown in Table J-1. These relationships hold only when the risk metric is (1) linear, and (2) expressed in terms of basic events only.

Table J-1. Relation between DIM and F-V, RAW, and Birnbaum Importance Measures.

	Relation between DIM and …		
	F-V	**RAW**	**B**

Constant Increment: $I_{x_i}^{DIM} =$	$\frac{\frac{I_{x_i}^{F-V}}{Pr(x_i)}}{\sum_k \frac{I_{x_k}^{F-V}}{Pr(x_k)}}$	$\frac{\frac{I_{x_i}^{RAW}-1}{1-Pr(x_i)}}{\sum_k \frac{I_{x_k}^{RAW}-1}{1-Pr(x_k)}}$	$\frac{I_{x_i}^{B}}{\sum_k I_{x_k}^{B}}$
Constant Percentage Increment: $I_{x_i}^{DIM} =$	$\frac{I_{x_i}^{F-V}}{\sum_k I_{x_k}^{F-V}}$	$\frac{\frac{I_{x_i}^{RAW}-1}{\frac{1}{Pr(x_i)}-1}}{\sum_k \frac{I_{x_k}^{RAW}-1}{\frac{1}{Pr(x_k)}-1}}$	$\frac{I_{x_i}^{B} Pr(x_i)}{\sum_k I_{x_k}^{B} Pr(x_k)}$

J-5. REFERENCES

J-1. Cheok, M. C., G. W. Parry, and R. R. Sherry, “Use of Importance Measures in Risk Informed Regulatory Applications,” Reliability Engineering and System Safety, 1998.

Appendix K

Prevention Worth / Prevention Analysis

K-1. TOP-EVENT PREVENTION WORTH

The measures of "basic event" importance are not really about just the basic events; each basic event measure reflects properties of the union of cut sets that contain the designated basic event. So the "importance" of a basic event is really an attribute of the cut sets (the failure scenarios) in which that event appears. All of the measures previously introduced are couched in failure space: they reflect contributions to risk, or the sensitivity of risk to changes in basic event characteristics. Analogous measures can be defined in success space: we can examine the properties of the union of *path* sets containing a designated component. One such measure is "Prevention Worth (PW)," defined as:

$$PW^i = P\left(\bigcup_j MPS_j^i\right),$$

where i indexes basic events, and MPS_j^i are the minimal path sets containing basic event i. This is a bit like the numerator of the Fussell-Vesely (F-V) measure, substituting path sets for cut sets; but since success path probabilities are generally of order unity, the rare event approximation cannot be used to calculate the probability associated with a union of success paths. However, in many cases, we can approximate the right-hand side by computing the probability of *failure* of that union of path sets, and subtracting it from unity:

$$PW^i \cong 1 - P\overline{\left(\bigcup_J MPS_j^{\imath}\right)}$$

Finally, for ease of interpretation, it is useful to introduce the "NINES" index, which, for a given PW, is calculated as:

$$NINES(PW) = -\log(1 - PW).$$

This says how many "nines" of reliability are afforded by the union of path sets considered: for example, a reliability of 0.999 is said to provide three nines of reliability.

Table K-1 shows the results for the case of the simple problem shown later to introduce prevention analysis.

Table K-1. Comparison of PW with risk achievement worth (RAW) and F-V.

Importance Measure	**Element (From Figure K-1)**				
	N2	A	B	C	D
F-V	1.0	~1	0.001	0.001	0.001
RAW	10^4	10^3	1.1	1.1	1.1
NINES	4	2.9996	1.9586	1.9586	1.9586

For a given element, this measure reflects the safety significance of the success paths containing that element. Put another way: each element potentiates the success paths that contain it, and its PW is measured by the worth of the totality of those success paths. N2 has the highest PW: for the numbers assumed, the path that contains N2 is "worth" more than all of the other success paths put together. This is true because all of those other success paths contain A, whose failure probability (success probability) is greater (less) than that of N2.

Other insights are available from the table. N2 has a F-V of 1, independently of its failure probability, because it appears in every minimal cut set. But all of the other measures tabulated depend, to some extent, on the nominal failure probabilities assigned to the associated basic events.

PW was formulated originally for the purpose of illustrating the benefits of thinking both in success space and in failure space, rather than focusing exclusively on failure space. However, although the measure arguably provides an interesting perspective on the role played by events in the model, as far as the authors are aware, no commercial probabilistic risk assessment software computes PW. Unfortunately, for realistic problems, the calculations are rather difficult; one needs first to parse out the success paths containing the element(s) of interest, and then (to use the above approximation to compute PW) evaluate the complement of that expression in order to approximate PW.

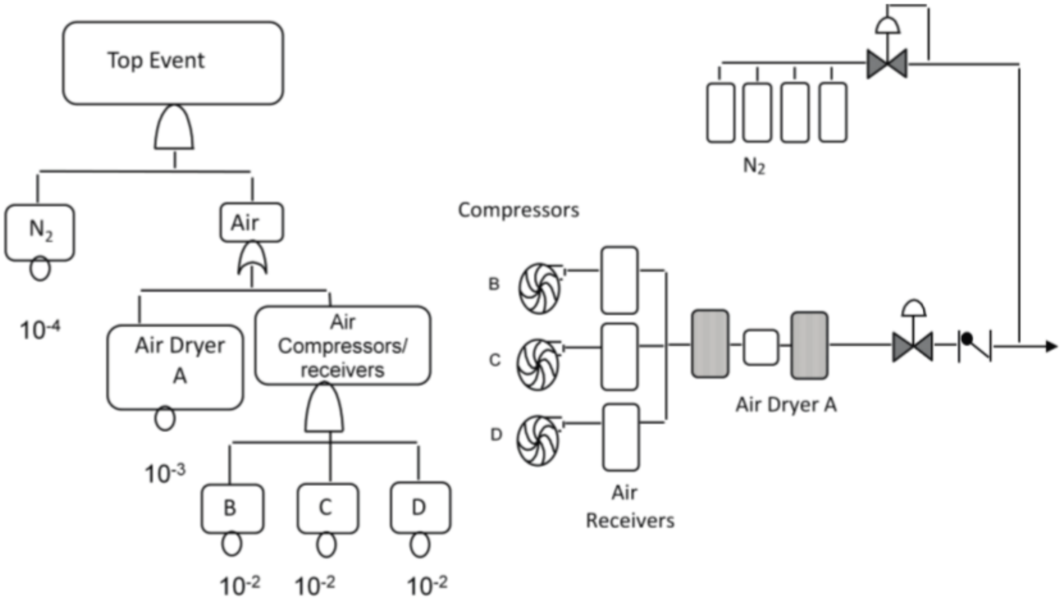
K-2. TOP-EVENT PREVENTION ANALYSIS

Consider the problem of determining the allocation of resources to activities aimed at maintaining and verifying the performance and reliability of safety equipment ("special treatment," as it is called in the nuclear industry). This is important both to facility operators and to their regulators. To see why basic event importance measures are not necessarily a reliable guide to solving this problem, consider the example presented in Figure K-1. The system shown is supposed to supply compressed nitrogen (or air) to another system downstream. In order to succeed, we need either to supply air from one of the compressors via the receivers and the air dryers shown in Figure K-1, or to supply compressed nitrogen from the tanks shown in the upper portion of the figure. A simplified fault tree is shown on the left, showing that the top event is an AND of the failure of these two options ("Air" and "N2"). For simplicity, the compressed-nitrogen option is modeled as a single event "N2." All components in that leg are logically in series, so no information is lost by this, unless there is some linkage between components in that segment and components in functionally redundant segments. In a real system, this is a real possibility, but the present illustration does not require us to address it. Similarly, the Air Dryer segment is modeled as a unit, and each compressor-receiver pair is modeled as a unit. Again, shared dependency of the compressors (e.g., of power supply) is a real possibility, but the present illustration does not require us to address it.

There are two minimal cut sets of the fault tree shown: N2 * A and N2 * B * C * D. Notional basic event probabilities are assigned on the fault-tree figure itself, and based on these, the F-V and RAW are tabulated below the system diagram. One sees that N2 and the Air Dryer have large values for both RAW and F-V, while the compressors do not. This is a result of the compressors being mutually redundant: if B fails, you still probably have C and D; if C fails, you still probably have B and D; and if D fails, you still probably have B and C. This is an example of the "portfolio" effect mentioned above. It would be inappropriate (but not unprecedented) to conclude from these F-V and RAW values that the compressors are not "important." This example is simple enough to see through without much machinery, but not all applications have that property.

Instead of trying to determine "special treatment" from importance measures, consider a different approach, called "Top Event Prevention." Within extant versions of this approach, one first formulates a prevention criterion to be satisfied by the *complement* of equipment to be considered "special." A simple example is to require single-failure tolerance in the complement of credited equipment: require the function(s) to succeed despite any single failure. Next, one applies an algorithm to identify subsets of the equipment potentially available, each subset having the property of satisfying the prevention criterion. In the lower left portion of the figure, we see the mechanics and the results of applying the single-failure criterion to the problem given. Start with the minimal cut sets given in the lower left of the figure (under "Top Event"). Evidently, any subset satisfying the single-failure criterion must contain both N2 and A; if a subset contains only A, and not N2, then the single failure of A fails the function, and vice versa. The second cut set requires us to work out some combinations: any single-failure-tolerant subset of the elements in a cut set must contain at least two of the elements in each cut set, and the logic expression for the six possibilities for the second cut set is shown. Since we need to "prevent" all of the cut sets, in order to obtain the prevention sets for the system, we "AND" together the prevention sets for each minimal cut set, and reduce the resulting expression. The resulting "minimal prevention sets" (the sets of events that collectively satisfy the prevention criterion) are shown in the lower right. It is straightforward to verify by inspection that each prevention set satisfies the prevention criterion.

A noteworthy feature of these prevention sets is that they all contain at least one compressor, a result that the importance-measure-based heuristic does not achieve. In general, prevention analysis always yields solutions that comprise unions of complete success paths, a result that is not to be expected from importance-measure-based reasoning.



$10^{-7} = N_2 * A$
$10^{-10} = N_2 * B * C * D$

	N_2	Air Dryer A	Fail to Start B	FTS C	FTS D
Fussell-Vesely	1.0	~1	10^{-3}	10^{-3}	10^{-3}
RAW	10^4	10^3	1.1	1.1	1.1

Top Event	**Prevention Sets, Level 2**	**Minimal Prevention Sets**
$N_2 * A +$	$(N_2*A) *$	$N_2*A*B +$
$N_2 * B * C * D.$	$(N_2*B + N_2*C + N_2*D +$	$N_2*A*C +$
	$B*C + B*D + C*D).$	$N_2*A*D,$

Figure K-1. Top-event prevention (simple example) (after [K-1, K-2]).

Each prevention set satisfies the prevention criterion, without credit for any other elements. As shown in the figure, the method has given us three options. In a real application, one could simply choose any one of the options, and assure that sufficient resources are allocated to every component in that set to achieve a good quantitative outcome (for example, test the active components at some regular interval).

The importance-measure-based heuristic does not, in general, point to unions of complete path sets. This is not to say that importance measures are "wrong"; they provide information about what the model is saying. But they do not always answer questions that need to be addressed at the portfolio level.

It is straightforward to extend the calculations illustrated above to address prevention criteria that call for quantitative reliability estimates, rather than essentially barrier-counting, although that form of the algorithm is not a true global reliability optimizer. However, it illustrates the more general process of choosing not only what items of equipment (operator actions, instrumentation, …) need to be credited, but also what assumptions, initial conditions, and so on need to be assured (and perhaps monitored during the operational phase) to provide reasonable assurance of the claims presented in the claims tree of Figure 4-1. This iterative process of self-consistently determining this portfolio of items is illustrated in Figure K-2.

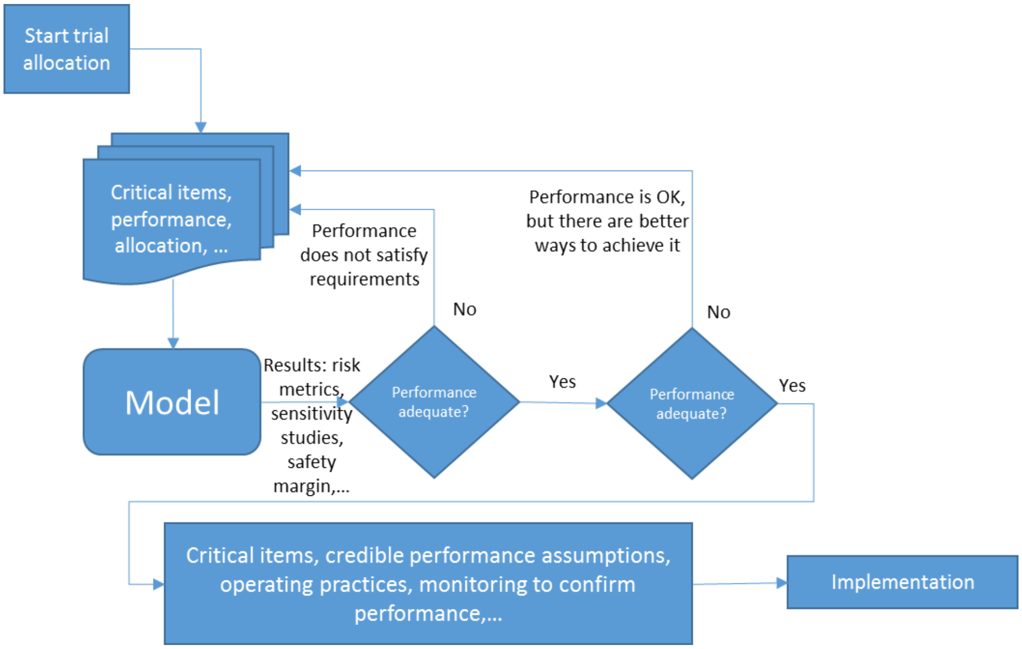

Start trial allocation
Critical items, performance, allocation, ...
Performance does not satisfy requirements
Performance is OK, but there are better ways to achieve it
No
No
Model
Results: risk metrics, sensitivity studies, safety margin,...
Performance adequate?
Yes
Performance adequate?
Yes
Critical items, credible performance assumptions, operating practices, monitoring to confirm performance,...
Implementation

Figure K-2. Process for confirming overall performance based on items credited in the assurance case.

K-3. REFERENCES

1. Brinsfield, W. and J. Voskuli, "Focusing the Scope of Fire PRA Human Reliability Analysis Using Top Event Prevention (TEP)," PSA 2015, 2015.

2. Blanchard, D. P. and R. W. Youngblood, "Risk-Informed Safety Margin Characterization Case Study: Use of Prevention Analysis in the Selection of Electrical Equipment to Be Subjected to Environmental Qualification," PSAM 12, 2014.