Modeling Common Cause Failures in Diverse Components with Fault Tree Applications

Joseph R. Belland, Isograph Inc.

Key Words: Common Cause Failures, Fault Trees, Beta Factor

SUMMARY & CONCLUSIONS

A common cause failure (CCF) is a single failure event that affects multiple components or functions of a system. Common cause failures are an important part of any reliability or hazard model, since they work to negate the improvements offered by redundancy. They are often the biggest contributors to risk levels, and should thus always be carefully considered. There are many system analysis methods that offer ways of taking common cause failures into account. However, these methods tend to be simple, such as taking a percentage of component failures, and attributing them to common causes. Certain assumptions are often made, such as that components will all share the same common causes, or have the same failure rate and distribution.

For instance, the beta-factor model is a very commonly-used method, found in standards such as IEC 61508. To calculate the failure rate due to common causes, the beta factor is simply multiplied by the component failure rate. In essence, the beta factor simply represents the percentage of component failures that are due to common causes.

Other papers have been written to explore general analysis methods for common causes, where the failure probability of the common cause is known [1]. In this paper, we shall focus more specifically at methods of expanding upon the beta-factor model so that it can be accurately used to calculate the probability of the CCF event, even where component rates, distributions, and common cause group probabilities differ.

In addition to discussing the mathematical models used in these scenarios, we will also consider Fault Tree modeling solutions.

1 INTRODUCTION

1.1 Redundancy

Most reliability-critical systems have some sort of redundancy against failures built into the design. This redundancy is designed to limit the impact of a single component failure to cause a catastrophic event. Sometimes, this redundancy may even be a requirement. The IEC 61508 standard specifies that, to achieve the highest rating, each system function must be able to tolerate at least one hardware failure [2].

Redundancy is an effective way of mitigating failures due to the multiplication law of probability. The multiplication law states that the probability of two independent events occurring is the probability of the first event times the probability of the second [3], or:

$$P(A \cap B) = P(A) \cdot P(B) \quad (1)$$

The effect this has on failure probability is geometric. For instance, if a component has a failure probability with an order of magnitude of 10^{-3} and is redundant to another component with the same probability of failure, then the failure probability of the function they provide is $10^{-3} \times 10^{-3} = 10^{-6}$. Doubling the number of components doesn't halve the failure probability, but rather doubles the *exponent* of the failure probability.

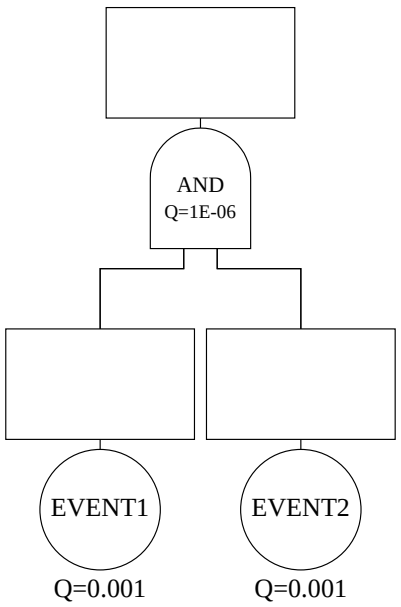

Figure 1 - A fault tree representing a simple redundancy.

A fault tree representing this example can be seen in Figure 1. "Q" is the unavailability, or failure probability.

1.2 Common causes

Common cause failures work against this redundancy by reducing the independence of the events. The probability law,

doubling the exponent, is only valid for independent events. By stipulating a common cause, the events are no longer independent and the approach is no longer valid. The common cause must be treated as a separate, single-point failure. The Rare approximation (only accurate for small probability values) for this would be:

$$P(A \cap B) = P(A) \cdot P(B) + P(CCF) \tag{2}$$

In our example, supposing that 10% of failures are due to common causes, the probability of failure of the function is now:

$$0.9\times10^{-4} \cdot 0.9\times10^{-4} + 0.1\times10^{-4} = 1.0081\times10^{-4} \tag{3}$$

Note that the independent probabilities have been adjusted downwards to account for the common cause percentage. A fault tree representing this is shown in Figure 2.

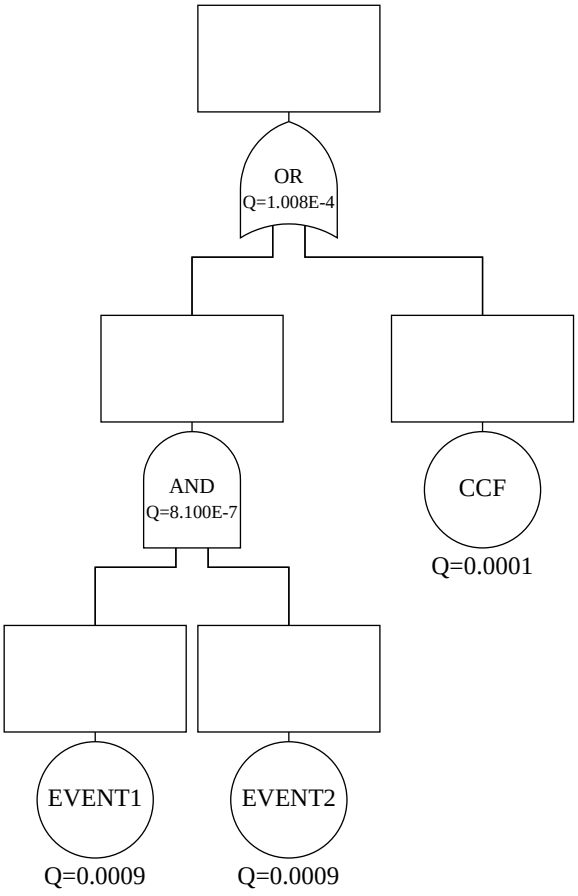

Figure 2 - A fault tree representing a redundancy with a common cause.

2 BETA FACTOR

The most commonly-used method of modeling common cause failures is the beta factor method. The beta factor method stipulates that a certain percentage of component failures are due to common causes. Component failures are then split into *independent failures*, affecting just the component, and *common cause failures*, affecting all components sharing the common failure mode. The beta factor is the ratio of common cause failures to total failures for the component [4]. This is the method illustrated in Figure 2.

2.1 $\beta\lambda$ versus βP

Typical implementation of the beta factor method calculates the common cause failure rate for a component as the beta factor times the total component failure rate [4]. That is,

$$\lambda_{CCF} = \beta \cdot \lambda_T \tag{4}$$

This implementation, however, can prove to be problematic in fault tree analysis. Fault trees are ultimately quantified using event probabilities, not failure rates. The event probabilities are used with mathematical probability laws and Boolean algebra to quantify the fault tree probability [3]. While failure rates are often used in a fault tree and equations exist to calculate an event probability from a failure rate, there is no guarantee that each event will have an associated failure rate. And even if events have a failure rate, it is not necessarily clear how a probability of failure due to common causes would be calculated in cases where events with the same common cause failure mode have different failure characteristics.

For instance, the probability of failure (unavailability) of an event with immediately revealed, exponentially-distributed constant failure and repair rates is given by [4]:

$$P(t) = \frac{\lambda}{\lambda + \mu}(1 - e^{-(\lambda+\mu)t}) \tag{5}$$

Where $P(t)$ is the probability of failure at time t, λ is the failure rate of the component and, μ is the repair rate. This equation is often used to calculate the probability required for fault tree quantification from the component failure and repair rates known for a typical component.

The unavailability of a similar event, but with hidden or dormant failures (only revealed by testing), can be calculated from a simplified equation [5]:

$$P_{mean} = \frac{\lambda\tau}{2} + \lambda \cdot MTTR \tag{6}$$

Where P_{mean} is the average probability, λ is the failure rate, τ is the test interval, and *MTTR* is the mean time to repair.

Suppose a system consists of two redundant components to provide a function. The first component is the operating component, and the second is in standby and only operates when needed. In most cases, the probability of the first component would be calculated in a fault tree using equation (5), while the probability of the second would be calculated from equation (6). If the events share a common cause, even if they have the same failure rate (which is unlikely; the dormant component most likely has a lower failure rate while not operating), then it is unclear how to translate the common cause failure rate to a probability that can be used in fault tree calculations.

Using the fault tree in Figure 2 as an example, supposing that EVENT1 is a continuously-operating component using equation (5) to determine probability, EVENT2 is a standby component whose dormant failure probability is modeled

using equation (6), and given values for failure rate and the beta factor, what probability value should be used for the CCF event?

Because of these concerns, a reasonable practice is to treat the beta factor as a percentage of *unavailability* rather than *failure rate*, when applying the beta factor model to a fault tree. In this manner, the probability of the common cause event is beta factor times the probability (instead of the failure rate) of the component event.

While this method can create a discrepancy with the $\lambda \cdot \beta$ method used in many standards, the difference is usually small, for reasonable failure rate values. Table 1 compares the differences between the two approaches for various failure rates.

λ	$P_{\beta\lambda}$	βP_{λ}	% Difference
0.5	0.048771	0.039347	19.3224%
0.1	0.00995	0.009516	4.3608%
0.01	0.001	0.000995	0.4486%
0.0001	0.00001	0.000	0.0045%
β=0.1, t=1, $P(t)=1-e^{-\lambda t}$			

Table 1 - Comparison of beta factor multiplied by failure rate vs. probability

There are many other scenarios where some conditions exist that require extension of the beta factor to accommodate the system being modeled.

3 DIVERSE FAILURE RATES

The IEC 61508 standard defines the PFD (probability of failure on demand) average for a two-channel redundant system as [6]:

$$PFD_{avg} = 2([1-\beta_D]\lambda_{DD} + [1-\beta]\lambda_{DU})^2 t_{GE} t_{CE} + \beta_D \lambda_{DD} MTTR + \beta \lambda_{DU}\left(\frac{T_1}{2} + MTTR\right) \quad (7)$$

Where

$$t_{CE} = \frac{\lambda_{DU}}{\lambda_D}\left(\frac{T_1}{2} + MTTR\right) + \frac{\lambda_{DD}}{\lambda_D} MTTR \quad (8)$$

$$t_{GE} = \frac{\lambda_{DU}}{\lambda_D}\left(\frac{T_1}{3} + MTTR\right) + \frac{\lambda_{DD}}{\lambda_D} MTTR \quad (9)$$

These equations assume the same failure rate, MTTR, and test interval for both components. But when the components have different failure rates or test intervals, as we saw in the previous example, this equation is inadequate.

This is common in scenarios like the example. The active and standby components will always have different failure probabilities, and will most likely have difference failure rates (due to reduced stresses during non-operational periods).

So if the beta factor is the percentage of failures that are due to common causes, the question becomes, "which component's failures?" Should a beta factor of 10% be applied to the operating component or the standby component?

To account for this, best-case, worst-case, or some form of beta factor averaging may be used. As an example of these different possibilities, we will use the fault tree in Figure 3, a modification of Figure 2, as our baseline.

The scenario is as described in section 2.1. There are two components voted 1 out of 2. Only one component need be operating at a time. The second component is in standby when not needed. Standby failures will go unnoticed until an inspection takes place, and as such the unavailability of the standby component is higher than the active component. Assuming a beta factor of 10%, the goal is to determine what probability value should be used for the CCF event.

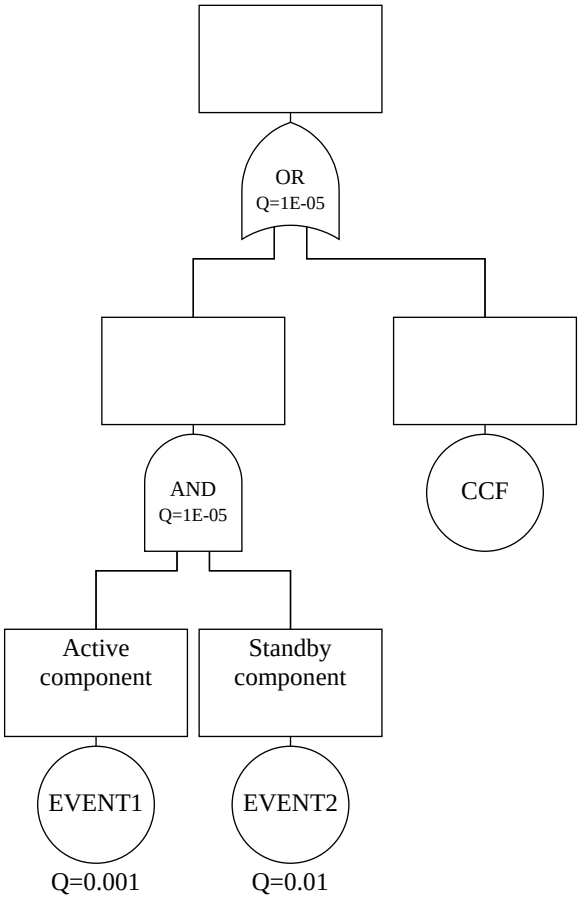

Figure 3 - Baseline example fault tree

3.1 Best Case

The best-case scenario would look at the common cause failure probability as a percentage of the event with the lowest probability, that is, EVENT1. The reasoning is that, since the common failure mode affects both components, it will be immediately revealed and repaired, as are all failures of EVENT1. In this case, the unavailability of the CCF event will be 10% of 0.001 or 1×10^{-4}.

3.2 Worst Case

The worst-case scenario considers the common cause failure probability as a percentage of the event with the highest probability, in this case, EVENT2. This would be used in safety-critical systems, where erring on the side of caution

is more prudent. In the given example, best-case CCF determination assumes that common failures affecting the standby component are immediately revealed and repaired, which may not be the case. This could lead to optimistic predictions.

In the worst-case CCF determination, the unavailability of the CCF event would be 10% of 0.01 or 0.001. This is clearly impractical in this example, as that is the same probability as EVENT1. The implication is that unavailability due to common causes is the same as the unavailability of EVENT1, which is unlikely.

A more practical example of worst-case determination would be if both components have similar operating profiles (e.g., both are active components), yet still have slightly differing failure rates, perhaps because they are of different styles or qualities.

3.3 Beta Factor Averaging

A compromise of best- and worst-case CCF determination is CCF averaging. For this method, the probabilities of all events sharing a common cause are averaged, and the CCF is calculated as a percentage of the mean. For our example scenario, the CCF probability would be:

$$P_{CCF} = 0.1 \cdot \left(\frac{0.001 + 0.01}{2} \right) = 5.5 \times 10^{-4} \quad (10)$$

Note that the geometric mean can also be used. Geometric averaging may make more sense when there are order-of-magnitude differences in probability between the events sharing the common cause. Geometric averaging for our example would give a CCF probability of:

$$P_{CCF} = 0.1 \cdot \sqrt{0.001 \cdot 0.01} = 3.16 \times 10^{-4} \quad (11)$$

Table 2 gives a comparison of the four methods. It provides the CCF probability, the adjusted event probabilities, and the final system result for each method.

Method	P_{CCF}	$P1_{adjusted}$	$P2_{adjusted}$	P_{system}
Best-case	0.0001	0.0009	0.0099	1.089E-4
Worst-case	0.001	0	0.009	1.0E-3
Mean	0.00055	0.00045	0.00945	5.554E-4
Geometric	0.000316	0.000684	0.009684	3.23E-4

Table 2 - Comparison of the four methods of beta factor calculation

4 EXTENDING THE BETA FACTOR

Another scenario that can occur is when multiple components share a common cause, but the percentage of failures due to common causes differs for each component.

Consider a scenario with three redundant components providing the same function. Two of the components are of the same type, while the third is of a different type. For example, a three-valve system may consist of two butterfly valves and a ball valve. The provision of one component of a different type is for diversity; many common cause failures, such as manufacturing CCFs, may be eliminated between the different component types. However some CCFs, such as environmental causes, will remain. As a result, the third component will share common causes with the first two, but have a lower beta factor in common with them than the two of the same type will have together.

To model this in a fault tree, two separate CCF events are needed: one for the common causes that affect all three components, and one for the common causes that affect only the two identical components. Figure 4 illustrates a fault tree model of this scenario.

Application of beta factors requires clear explanation of what the beta factors refer to. If we stipulate that component type A has a 5% CCF rate, and component type B has a 2% CCF rate, we must be clear whether that means 5% of type A failures are shared *only* with type A components (meaning the 2% type B rate is in addition to the 5% rate) or whether that 5% includes *all* CCFs (including those shared with the type B component). Either meaning can be used, so long as clarity and consistency are maintained; the model must mean what we intend.

For our example, we'll postulate that the CCF rate for type A components is 5% *total*, and the CCF rate for the type B components is 2%. This means that, of the common cause failures affecting the two type A components, 3% will affect type A components *only* and the remaining 2% will affect *all* components. Assuming the same probability of failure for all three component types, the probability values for the events and system in Figure 4 are given in Table 3.

Component	Value
Total probability (all components)	0.001
Type A CCF	5%
Type B CCF	2%
A&B CCF probability	2E-5
A only CCF probability	3E-5
Component 1 & 2 independent probability	9.5E-4
Component 3 independent probability	9.8E-4
System failure probability	2.003E-5

Table 3 - Event probabilities for Figure 4

If the different component types have different failure rates or probabilities, which is likely, then an averaging method such as those discussed above, can be used.

This method may be extended if, for instance, there are two type B components as well. Then we could further split the CCF beta factors into: common causes affecting components of type A; common causes affecting components of type B; and common causes affecting components of both types A and B. The fault tree model for this scenario is left as an exercise for the reader.

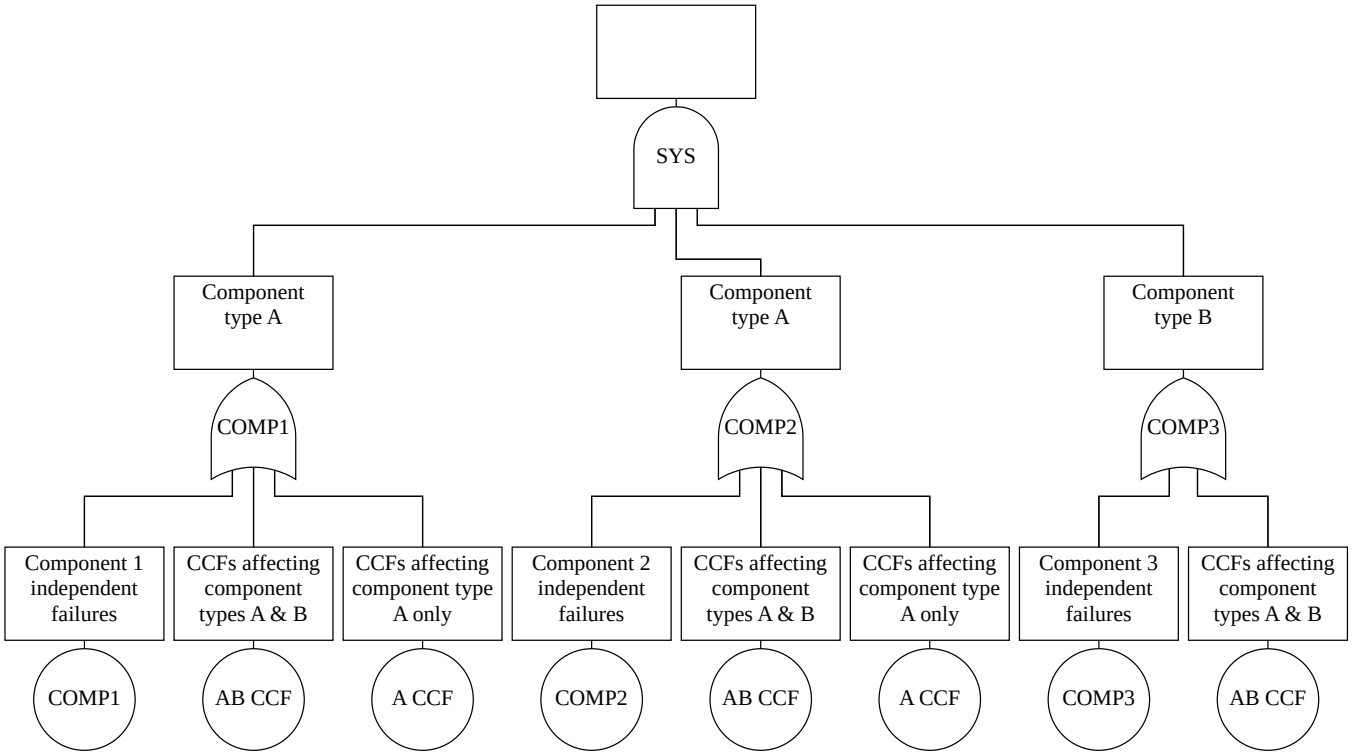

Figure 4 - A fault tree illustrating different common causes shared amongst different component types.

4.1 One Component, Two CCF Groups

This method may be extended, once again, to model a scenario where a component shares a common cause with two others, but the others do not have a failure in common with each other. This can occur if a component provides functions to two different systems. Each system may have its own common cause, requiring the shared component to have two common cause failure events. Again, the fault tree must be modified to show the common component can be affected by either common cause failure.

4.2 Generally

Continuing this method of dividing up component probability amongst various common cause failures, we can extend the beta-factor method to arrive at a general form for the event model, for an arbitrary number of common cause groups to which the event can belong.

The general rule for non-uniform beta factors is that if a component shares common causes with n different CCF groups, then the component must be broken into $n+1$ events in the fault tree. The general form is shown in Figure 5. The general form of the independent event probability is given in equation (12).

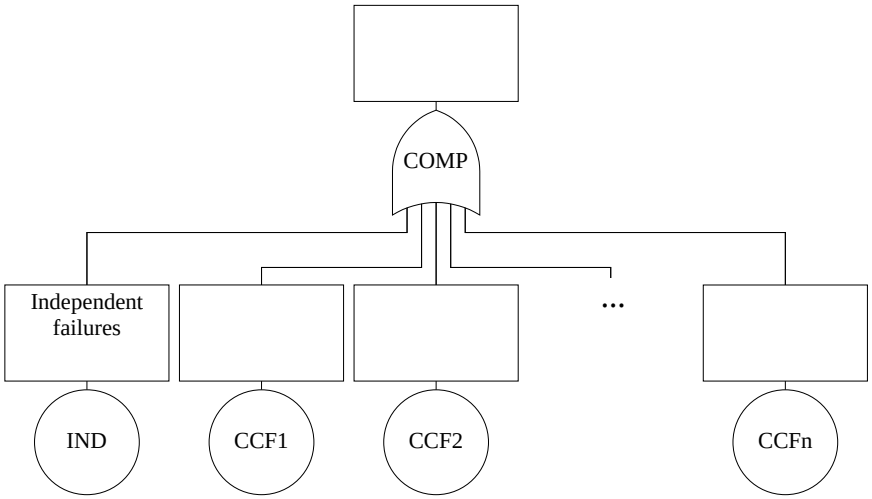

Figure 5 - General form of an event with n common causes

$$P_{IND} = P_{TOTAL} - P_{CCF1} - P_{CCF2} \ldots - P_{CCFn} \quad (12)$$

The probability of each common cause event can be calculated using one of the methods discussed in section 3.

PARTIAL CCF

One last consideration is when there is a single CCF group of three or more components. The components share a potential common cause failure, but the common cause does not necessarily affect every component each time it occurs. It is possible that only two of the three components may fail due to common causes. The beta factor model, in this case, can lead to pessimistic predictions, because it assumes every common cause failure affects every component.

One solution is offered by the IEC 61508-06:2010 standard. This method posits a "beta factor adjustment" value, by which the beta factor is multiplied. The adjustment factor is determined from the voting arrangement of the group of components [6]

Another solution is the "MGL" or Multiple Greek Letter method. This method is an extension of the beta factor model, using additional Greek letters to represent the probabilities that a common cause would affect more than two, three, or more events simultaneously. If β is the percent of component failures due to common cause, then γ (gamma) is the percent of common cause failures that affect more than two components, δ (delta) is the percent of those failures that affect more than three components, and so forth.

With this method, each component event in the fault tree is replaced with an event for independent failures, a common cause event for each possible combination of two event failures, a common cause event for each possible combination of three event failures, and so on for the size of the common cause group. E.g, for a group of four events—A, B, C, and D— sharing a common cause, in a fault tree, event A would be replaced by:

$$A + [AB] + [AC] + [AD] + [ABC] + [ABD] + [ACD] + [ABCD] \quad (13)$$

The terms in brackets represent single CCF events affecting 2, 3, or 4 components in the CCF group. The first event, A, represents the independent failures of event A.

Unavailability values for each event are calculated from the expression [5]:

$$P_k = \frac{1}{\left\lceil \frac{m-1}{k-1} \right\rceil} \left(\prod_{i=1}^{k} \rho_i \right) (1 - \rho_{k+1}) P_T \quad (14)$$

Where P_k is the probability of the k^{th} order CCF failure,
$\rho_1=1$, $\rho_2=\beta$, $\rho_3=\gamma$, $\rho_4=\delta, \ldots, \rho_{m+1}=0$
P_T = total component unavailability, and
m = CCF group size

REFERENCES

[1] C. Wang, L. Xing and G. Levitin, "Explicit and Implicit Methods for Probabilistic Common-Cause Failure Analysis," *Reliability Engineering and System Safety,* vol. 131, pp. 175-184, 2014.

[2] International Electrotechnical Commission, *IEC 61508-2: Functional Safety of Electrical/Electronic/Programmable Electronic Safety-Related Systems,* Brussels: IEC, 2010.

[3] U.S. Nuclear Regulatory Commission, *NUREG-0492,* Washington, D.C.: U.S. Government Printing Office, 1981.

[4] J. D. Andrews and T. R. Moss, Reliability and Risk Assessment, New York: The American Sociery of Mechanical Engineers, 2002.

[5] Isograph, Ltd., *Reliability Workbench User Guide,* Manchester: Isograph, Ltd., 2016.

[6] International Electrotechnical Commission, *IEC 61508-6: Functional Safety of Electrical/Electronic/Programmable Electronic Safety-Related Systems,* Geneva: IEC, 2010.

BIOGRAPHY

Joseph R. Belland
375 South Main
Suite 4
Alpine, UT 84004

jbelland@isograph.com

Joseph Belland is Isograph's technical lead for support and training in North America. He has been with the company for 13 years, conducting training sessions for clients such as Boeing, General Electric Aviation, BP, Dow Chemical, and Continental Automotive. When not training clients or supporting Isograph's software, he is sometimes tasked with development, and has written some of the modules within Isograph's Reliability Workbench software suite, as well as miscellaneous bespoke software tools. Most of Joseph's experience with implementation comes from working closely with clients during training sessions.