A Comparative Look at MBU Hazard Analysis Techniques

Brandon D. Owens and Nancy G. Leveson

Complex System Research Laboratory
Massachusetts Institute of Technology
Cambridge, MA 02139-4307
owensbd@mit.edu

Abstract – **The flux of radiation particles encountered by a spacecraft is a phenomenon that can largely be understood statistically. However, the same cannot be said for the interactions of these particles with the spacecraft as they are far more challenging to grasp and guard against. The ultimate impact of a radiation particle's interaction with a spacecraft depends on factors that often extend beyond the purview of any subject matter expert and typically cannot be represented quantitatively in system-level trade studies without the acceptance of numerous assumptions. In this paper, many of the assumptions associated with the probabilistic assessment of the system-level effects of a specific type of radiation-induced hazard, a Multiple Bit Upset (MBU), are explored in light of MBU events during the Gravity Probe B, Cassini, and X-ray Timing Explorer missions. These events highlight key problems in using probabilistic, quantitative analysis techniques for hazards in highly complex and unique systems such as spacecraft. As a result, a case is made for the use of system-level qualitative techniques for both the identification of potential system-level hazards and the justification of responses to them in the system design.**

I. INTRODUCTION

The complexity of a system is a major component of the overall risk that must be accepted by the system's stakeholders. Systems that are highly complex are said to be subject to accidents that are "normal" not so much for their frequency, but for their inevitability (e.g. it is normal for a living thing to die, but it only does so once) [1]. Therefore, one might argue that the emphasis of analyses for complex system risk reduction should not be placed on determining if/when a system will fail, but how it might fail so that actions can be taken to mitigate the overall impacts of these failures. However, the former method is widely favored partly because it is, when taken at face value, more prone to produce quantitative results that can be used in engineering trade studies.

The objective of this paper is to link some of the difficulties in the statistical comparison of complex systems to fundamental principles of systems theory in order to inform the future analysis of risk in complex systems. In the pages that follow, some of the problems inherent in the quantitative analysis of expected incident/accident frequency will be explored in case studies of Single Bit Upset (SBU) and Multiple Bit Upset (MBU) events on three separate spacecraft: X-ray Timing Explorer (XTE), Cassini, and Gravity Probe B (GP-B). These events demonstrate how the implications of a phenomenon best characterized by its frequency when studied in isolation, change substantially when applied in the context of a complex system. Accordingly, the lessons learned from these events form the basis of an argument presented here for a stronger role of qualitative analysis methods—aimed at understanding, preventing, and/or mitigating the effects of failure modes—in complex system risk reduction.

II. SYSTEMS THEORY

While the analysis of physical and logical systems is a fundamental element of engineering and science, many of the techniques employed have limited applicability within the full range of systems that must be studied. According to systems theory, the applicability of various techniques is driven by the tendency of the analyzed system(s) to exhibit: 1) *Organized Simplicity*, 2) *Unorganized Complexity*, or 3) O*rganized Complexity* [2].

In systems that exhibit *Organized Simplicity*, the precise nature of system component interactions is known and each can be examined pairwise so that the number of interactions to consider together is limited [2]. Systems of this type are commonly seen in the homework assignments and exams given in subjects such as mechanics because simplifying assumptions can be applied to their study to produce analytic solutions that approximate observable behavior [2]. For example, while the n-body problem in celestial mechanics presents 2^n equations to solve, there is a structure to the interactions that allow analysts to reduce the total number of equations that they must consider [2]. Since the interactions between the bodies are similar in nature but not necessarily importance (e.g. the Sun is the dominant body in our Solar System), it is possible to selectively analyze a few bodies of interest to reduce the number of equations to a manageable figure and produce a reasonably accurate result [2]. The tradeoff in applying these simplifying assumptions is that some of the accuracy of the results is sacrificed and the scope of the analysis is narrowed. In systems exhibiting *Organized Simplicity* these simplifications do not sufficiently compromise the usefulness of the information to be obtained; the same, however, cannot be said for systems exhibiting *Unorganized Complexity* and *Organized Complexity*.

Systems exhibiting *Unorganized Complexity* lack an underlying structure that allows reductionism to be effective.

These systems are complex, but they are regular and random enough in their behavior to be studied statistically [2]. Hypothetically speaking, if one were to analyze a system of air molecules in a bottle as an n-body problem, that individual would be left with 2^n equations to solve with n being on the order of 10^{23} before the application of simplifying assumptions [2]. This represents far too many equations to solve and unfortunately, unlike in the study of celestial bodies, the selective analysis of a few bodies is not useful because the molecules are all of comparable significance and an understanding of the behavior of the system as a whole is of more practical value than an understanding of the behavior of specific molecules [2]. However, while the lack of a substantial system structure makes it difficult to reduce analytic equations, it also makes deviations from observable averages less substantial as the overall set of observations increases and thus statistical analysis of the system is possible [2]. Therefore, systems that display *Unorganized Complexity* can be compared to other systems if the effect of whatever underlying structure that does exist in these systems (e.g. thermal energy in a system of gas molecules) is understood and scalable from one system to the next. For example, the behavior of air molecules in one bottle can be used to predict the behavior of air molecules in another bottle whether they are at the same temperature or not.

Conversely, systems that exhibit *Organized Complexity* are too complex for analytical analysis and contain an underlying structure that deranges the averages that might be obtained in statistical analysis [2]. Ultimately, these systems present a quandary to those who wish to study them quantitatively; they have underlying structures that are substantial enough to make them too unique for useful statistical comparison with other systems, but not substantial enough to reduce their analytic equations to a manageable number. Unfortunately, since World War II, engineered systems, including spacecraft, have increasingly exhibited *Organized Complexity* and have in turn stressed many of the traditional quantitative risk reduction techniques employed in their design and operation [3].

III. BACKGROUND ON SBUs AND MBUs

Whenever memory devices in digital computers are exposed to energetic ions and protons, it is possible for bits in the computer memory to unexpectedly flip from 1 to 0 or vice versa. Whenever this occurs, the flipped or upset bits no longer contain the information they are supposed to contain and therefore have the potential to feed erroneous and possibly hazardous information to the computer. Thus, the study of bit upsets has far reaching implications in the design and operation of systems relying on digital computers.

When a memory board exposed to radiation is considered individually (i.e. as a collection of bits rather than a component in a system), bit upsets provide prime examples of events stemming from its *Unorganized Complexity*. The board contains millions of virtually identical memory cells and experiences a flux of many times more numerous radiation particles. There is some underlying structure to the upset events, but it is neither substantial enough to reduce the staggering number of equations that must be considered in analytic analyses of these events nor skew observable averages. While the particles themselves have different energy levels and approach the board from different directions, there are so many of them that the board will experience interactions from particles with each of the energy levels and inbound trajectories many times over. Additionally, many of physical characteristics that might influence the frequency of upset events on such a board (e.g. total number of bits, energy required to flip a bit, etc.) are scaleable to other boards.

Once a logical structure (i.e. distinct groupings of bits into logical words that are used for data organization and upset correction) is added to the board, its behavior moves in the direction of *Organized Complexity* and ultimately, comparison of its behavior to that of other boards becomes more difficult. The addition of the logical structure alters the implications of upset events by both defining two separate classes of upset events, SBUs and MBUs, and altering their relative frequencies. The key difference between these two classes of events revolves around the status of other bits in the logical word where the upset occurs. If the upset occurs in a word that has already been upset in a separate event, or two or more bits in the same logical word are upset in the same event, an MBU occurs. On the other hand, if only one bit is upset in the word, from the start of the event to its correction, the event is called an SBU. Thus, with the addition of the logical structure, the factors that determine whether an event will result in an SBU or MBU include: the number of bits per word, the time it takes for upsets to be corrected (automatically or through human intervention), and the physical position of bits within the same logical word. Each of these factors can vary from one board to the next and create problems in the statistical comparison of the behavior of various boards.

For MBUs, it is usually the case that a single radiation particle follows a trajectory of sufficient incidence to the plane of the memory board to strike several cells containing bits for the same word [4]. This is illustrated in Figure 1 below.

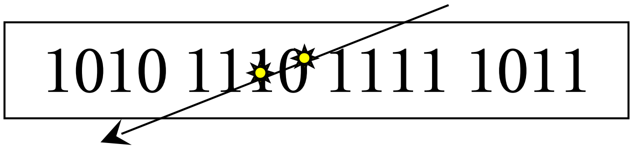
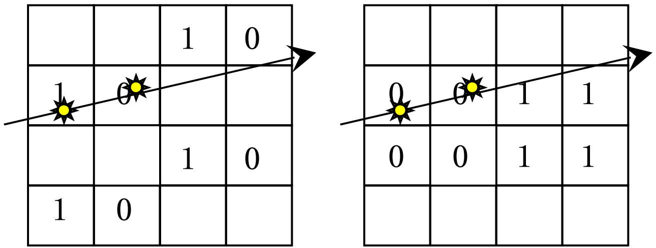


Figure 1. Trajectory of an MBU-inducing radiation particle [5].

In memory devices in which the cells containing the bits of a logical word are physically adjacent, there are a greater number of particle trajectories that lead to MBUs than there are in devices in which such cells are separated. This concept is exemplified in Figure 2, where, for simplicity, two four-bit logical words, 1111 and 0000, are considered in two memory devices with different configurations of the bits. Notice that the total number of upset bits for the given particle trajectory is the same for both configurations. However, the device on the left experiences SBUs in each word while the device on the right experiences an MBU in one word [5].



Two upset particles, two SBUs

Two upset particles, one MBU

Figure 2. Nature of upsets given one particle trajectory and two configurations of bits in the words: 1111 and 0000 [5].

Incidentally, both the physical structure regulating upset events and the logical structures typically employed in spacecraft memory devices make SBUs occur far more frequently than MBUs. The upset of multiple bits requires more energy than the upset of a single bit and requires the particles to approach the boards from specific directions; this makes the overall number of particles that can cause an MBU event significantly lower than the number that can cause an SBU event. Additionally, logical words are typically extremely small relative to board size and SBU events are typically corrected with error detection and correction (EDAC) schemes such as a Hamming Code seconds after they occur. This significantly reduces the likelihood of two or more independent upset events occurring in the same word before the first is corrected[1]. Ultimately, these factors minimize the effect of both the logical and physical structure on statistical averages of SBU events by substantially reducing the frequency of MBUs, which are essentially the only things that prevent all upset events from being classified as SBUs.

Consequently, while the logical structures mentioned above usually have little effect on the statistical averages of SBUs, they simultaneously dominate statistical averages of MBU events and render comparisons between boards virtually useless in a number of cases. For instance, if a prediction of MBU frequency is derived from data collected from boards with different logical structures, the predictions could be off by orders of magnitude [5], [6]. Thus, the addition of even a small degree of underlying structure to a complex system such as a memory board exposed to radiation presents a potential pitfall in the quantitative analysis of MBU frequency.

In the following sections, the contexts in which memory boards will be viewed will expand from those of isolated boards exposed to radiation to those of boards operating as components of larger, more complex systems. As a result, the pitfalls in quantitative MBU prediction will grow more numerous and severe.

IV. SBU AND MBU EVENTS ON SPACECRAFT MISSIONS

Up to this point in the paper, SBUs and MBUs have been examined with little attention paid to the larger issue of why SBU and MBU frequency predictions would matter at all. Essentially, this issue is dominated by the priorities of the system stakeholders. As mentioned earlier, SBUs and MBUs can feed erroneous and hazardous information to the computer if the upset words are executed. Fortunately, if the system stakeholders decide that this threat is unacceptable, they can implement EDAC schemes that automatically correct SBUs and MBUs and effectively relegate their frequencies to nothing more than mere curiosities. However, EDAC schemes are costly to implement in that they occupy a portion of the available memory, use up processor cycles, and add overall complexity to the system. Furthermore, systems such as spacecraft are usually subjected to numerous threats in addition to SBUs and MBUs. Thus the issues surrounding SBUs and MBUs must compete with those of other threats and system performance objectives for system stakeholder priority. As a result, there is always the possibility that the priority given to mitigation of SBU and MBU effects will be lower than it should be and the appropriate steps to prevent them from escalating into major system-level events may not be taken.

A technique that is commonly used to facilitate the competition of threat mitigation and/or performance enhancement priorities is risk-benefit analysis. This technique is regarded as one that can, in addition to quantifying risk[2], show which risk reduction measures will provide the most risk reduction per unit of resource invested. Typically, it relies on the combination of a deterministic evaluation of outcomes related to the threat and a probabilistic assessment of the frequency of events related to the threat to arrive at these estimates. Unfortunately, as hinted at earlier, the prediction of the frequency of events stemming from *Organized Complexity*, is complicated by the fact that the underlying structure of complex systems can induce system-specific responses to events that appear stochastic and generalizable when viewed on the component-level. In the remaining paragraphs of this section, actual events stemming from SBUs and MBUs on three spacecraft will be explored to provide context for the discussion to follow on the quantitative and qualitative analysis of risk in systems that exhibit *Organized Complexity*.

A. SBU and MBU events on X-ray Timing Explorer (XTE)

XTE is an x-ray observatory that has been operating in Low Earth Orbit (LEO) since December 30, 1995 [7]. Its orbital inclination is roughly 23 degrees and its altitude has ranged from 500 km to 580 km over its lifetime [7]. XTE's science data is recorded on a Solid-State Recorder (SSR) comprised of Hitachi HM628128 Static Random Access Memory (SRAM) devices that use a Hamming EDAC Code for the correction of SBUs and detection of MBUs [7]. Spacecraft

[1] On the GP-B mission, each word was only 72 bits long (64 bits for data and 8 bits for EDAC), it took as few as 10 seconds to scan each MB of memory, and every scanned MB experienced roughly 2 SBUs per day [5].

[2] In this paper, the definition of risk involves the combination of the likelihood of an event and the severity of the event [3].

health and status functions are run on a separate memory device that is more hardened to radiation than the Hitachi HM628128 [8].

The frequency of SBUs and MBUs on XTE's SSR throughout its first decade in orbit was affected by the physical structure of the radiation environment that it operated in as well as the logical structure of the Hitachi HM628128 memory devices it utilized. First, the inclination of XTE's orbit allowed it to pass through the South Atlantic Anomaly (SAA), which is the region in LEO that contains the largest density of upset-inducing energetic particles [7]. Second, XTE was launched shortly after Solar Minimum and thus received its highest SBU and MBU rates at the beginning of its mission [7]. However, as its orbit decayed and solar activity increased[3], the number of energetic particles in the spacecraft's operating environment decreased and thus SBU and MBU rates decreased [7]. Finally, the logically adjacent bits in the Hitachi HM628128 are physically adjacent, making it significantly more susceptible to MBUs than a device in which the logically adjacent bits are physically distributed [4], [7].

Ultimately, the SBU and MBU events on XTE had little impact on the overall mission. The Hamming Code corrected all SBUs and the MBUs' corruption of scientific data amounted to a worst case Bit Error Rate (BER) of $2x10^{-9}$ per day in the science data, which is orders of magnitude below the acceptable BER for the mission [7]. Moreover, since the spacecraft health and status functions were run with a memory board separate from the one that used the Hitachi HM628128, the MBUs did not affect spacecraft functionality [8].

B. *SBU and MBU events on Cassini*

The Cassini interplanetary mission to Saturn began in October of 1997 [6]. The spacecraft contains two SSRs utilizing 4Mb OKI DRAMs as the memory device [6]. This device uses a Hamming EDAC Code for SBU correction and MBU detection with each logical word consisting of 32 bits reserved for data and 7 bits reserved for EDAC [6].

As expected, Cassini encountered both SBUs and MBUs during its transit to Saturn. However, while Cassini encountered SBUs at a rate that was in reasonable agreement with preflight predictions, the rate of MBUs exceeded the preflight prediction by orders of magnitude [6]. This was due to the fact that the predictions did not account for the physical adjacency of logically adjacent bits on the OKI DRAM [6]. Fortunately, even though the flight software resides on the SSR, no major system-level reactions to the MBUs have been reported. In fact, mission engineers have even dubbed these events (at least in the transit phase) as merely a nuisance [6].

C. *SBU and MBU events on Gravity Probe B (GP-B)*

*G*P-B was launched on April 20, 2004 into a 642 km circular, polar orbit. After an initialization and on-orbit checkout period, the spacecraft collected primary science data from August 28, 2004 to August 15, 2005 [5]. Its mission was to test Einstein's Theory of General Relativity by directly measuring relativistic changes in the spin axis orientation of four mechanical gyroscopes [9]. The challenge was that these relativistic effects are extremely small and thus the spacecraft and payload subsystems had to perform a number of difficult tasks concurrently, such as:

1. Pointing of the spacecraft's on board telescope to within 200 milli-arc seconds of its guide star, IM Pegasi (HR 8703),
2. Electrostatic suspension of the four mechanical gyroscope rotors above their housings,
3. Maintenance of six degree of freedom drag-free control (i.e. a continuous nullification of disturbances in spacecraft position so that one of the gyro rotors remained in a constant state of free fall),
4. Recording of miniscule (i.e. on the order of 1 milli-arc second) changes in the spin axes orientation of the gyroscope rotors,
5. Payload and vehicle data synchronization, and telemetry handling.

Six computers, mounted in various locations on the spacecraft, were required to properly perform these tasks. Each computer used the memory device that was used on the XTE SSR, the Hitachi HM628128, and a Hamming EDAC scheme was employed for SBU mitigation. This scheme as implemented on GP-B used 8 bits out of every 72 bit logical word for EDAC [5].

As was the case on XTE and Cassini, the frequency of SBUs and MBUs on GP-B was affected by both its environment and the logical structure of its memory devices. GP-B's orbit exposed it to the three geographic areas where upset events most commonly occur in LEO: the SAA and the two Polar Regions. Additionally, like XTE and Cassini, GP-B used memory devices in which the logically adjacent bits were also physically adjacent. This led, as it did on Cassini, to predictions for MBUs that were low by orders of magnitude while predictions for SBUs were fairly accurate.

Unfortunately, unlike XTE and Cassini, the occurrence of MBUs on GP-B caused considerable operational disruptions on GP-B. Though most of the MBUs (i.e. 33 of ~38) occurred in areas of memory that were not read for execution before the operations team could react, some managed to either execute or trigger a safemode response that rebooted the afflicted computer. In all, the various computers on the spacecraft were rebooted five times—three times were due to the safemode response and two were due to execution of the upset word. These events presented operational challenges for the Gravity Probe B mission. The effects of these events on the mission are as follows [5]:

[3] Increased solar activity causes the Earth's atmosphere to expand and absorb more energetic particles at the low altitudes of LEO [7].

1. Thousands of employee hours were spent on investigative and recovery efforts,
2. The risks associated with the recovery of a system from an anomalous event were incurred six times,
3. As many as eleven days worth of science data collection opportunities—roughly 3% of the total data collection opportunities—were lost[4].

Ultimately it was the highly complex requirements of the GP-B mission, along with certain subtleties in the underlying structure of the spacecraft that allowed these component-level anomalies to propagate into system-level disruptions [5]. For instance, GP-B's mission duration was limited by cryogenic consumables and thus time spent recovering from upset events could not be made up as easily as it could be on XTE, which has had an operational lifetime of more than decade. Additionally, unlike Cassini, whose primary data collection phase began after a multi-year transit to Saturn, it was necessary for GP-B to begin data collection as close to the beginning of the mission as possible. This ultimately meant that the GP-B operations team had to progress through the learning curve of responding to MBU events while they were also trying to configure the spacecraft for data collection.

V. EVALUATING AND MITIGATING THE RISK OF MBU IMPACTS ON SPACECRAFT MISSIONS

The events described in the previous section highlight an important point: the differences in the designs and missions of these spacecraft led to highly different responses to MBUs. In other words, each had different underlying structures that would make statistical comparisons between their reactions to MBUs much less meaningful than statistical comparisons of the reactions of individual boards. In fact, one might even rank the importance of the underlying structures of these systems and the difficultly in the statistical assessment of upset phenomena as they are ranked in Figure 3. As stated earlier, SBUs and MBUs are artifacts of the same design characteristic of a memory board: its logical structure. However, the nature of the logical structure tends to make SBUs occur far more frequently than MBUs, thus providing more data for a statistical analysis. Therefore, SBUs are ranked as the easiest phenomenon to analyze statistically. Conversely, GP-B's reactions to MBUs are ranked as the most difficult to analyze statistically because they demonstrated the most coupling between high-level system requirements and system wide vulnerability to MBUs. This potential for coupling between system wide vulnerabilities and system requirements is also present on Cassini, though it appears from the reported impacts of the MBUs on that mission that it is present to lesser extent. Finally, XTE's reactions to MBUs are slightly easier to study statistically than those of Cassini and GP-B because of the utilization of a memory device that is largely resistant to MBUs for execution of spacecraft health functions essentially decouples the effects of MBUs on its SSR on the status and functionality of the spacecraft. However, with that said, the analysis is still far more complex than that which would be required to study the effects of MBUs on the memory board itself.

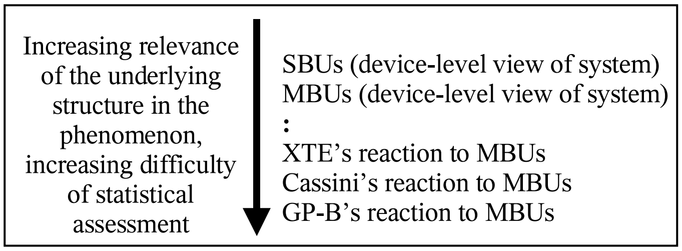

Increasing relevance of the underlying structure in the phenomenon, increasing difficulty of statistical assessment
SBUs (device-level view of system)
MBUs (device-level view of system)
:
XTE's reaction to MBUs
Cassini's reaction to MBUs
GP-B's reaction to MBUs

Figure 3. A ranking of the Organized Complexity in the reactions of the systems described in this paper to SBUs/MBUs.

In the remaining paragraphs of this section, two strategies for mitigating the risks associated with system-level reactions to MBUs will be presented. The first centers on reducing the frequency of MBUs to the point where the threat to the mission is perceived to be low enough to not warrant further efforts to reduce the frequency. In this strategy, the decision to cease efforts to reduce MBU frequency further is essentially a value judgment informed by the probabilistic calculation of the expected losses from MBU events. The second strategy places less emphasis on MBU frequency and instead centers on identifying and implementing methods to control the way that the system responds to MBUs.

A. *An example quantitative, statistical assessment of MBU hazards*

At the heart of most quantitative assessments of the risk posed by a hazard is an attempt to inform a decision (i.e. identify the best option) and/or to put a decision into context (i.e. compare it to another decision). Models of rational choice or behavior typically include the following elements [10]:

1. A set of behavior alternatives,
2. The subset of behavior alternatives that are considered or perceived by the decision maker,
3. The possible outcomes of a choice,
4. A notion of the payoff of each outcome,
5. Information as to which outcomes will actually occur
6. Information as to the probability that a particular outcome will ensue.

Thus, probabilistic assessments of the risk associated with hazards, such as MBUs, often take on the form of a decision tree such as the one in Figure 4. For simplicity, this tree only contains three intermediate events between the decision and the outcome of the decision. The variables in the figure are defined as follows:

A_X = Risk Mitigation Alternative X
N = Total Number of Risk Mitigation Alternatives
E_X = Occurrence of Event X
E_X' = Non-occurrence of Event X
X_{jk} = Outcome of the kth branch of Risk Mitigation Alternative j
U_{jk} = Utility of the Outcome of the kth branch of Risk Mitigation Alternative j
K = Total Number of outcomes stemming from Risk Mitigation Alternative 1.

[4] Despite the problems encountered with MBUs, GP-B was able to collect more data than the requirements specified. The results of the experiment will be released in 2007.

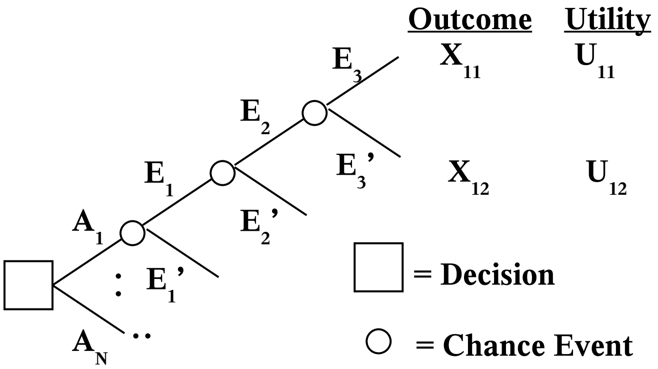


Figure 4. A generic, three-event deep decision tree for risk mitigation alternatives.

In the model of incident/accident causality represented by this tree, each risk mitigation alternative leads to several branches or chains of events that have an outcome in terms of several factors of interest to the analyst. For example, A_1 may cost $50,000 to implement and result in a computer reboot if events E_1 through E_3 occur. Thus, X_{11} would take the following value:

$$X_{11} \;=\; -\$50{,}000 + 1 \text{ computer reboot.}$$

Since X_{11} is not dimensionally consistent, it is necessary to normalize its attributes to dimensionless values or values associated with a common dimension (e.g. dollars) in order to assess the value of the outcome to the decision-maker(s). This is done through a multi-attribute utility function, such as the one below:

$$U_{11} = f(X_{11}).$$

Utility functions—which can be simple or highly complex—are subjective in that they are structured entirely around the value judgments of the decision-maker(s). The objective of using them in the analysis is to combine the utilities of all outcomes stemming from a design alternative with the probabilities of the events that lead to the outcomes in order to determine the expected utility or value of the design alternative. For example, for all outcomes stemming from A_1:

$$P(X_{11}) = P(E_1|A_1) \times P(E_2|E_1,A_1) \times P(E_3|E_2,E_1,A_1)$$
$$P(X_{12}) = P(E_1|A_1) \times P(E_2|E_1,A_1) \times P(E_3'|E_2,E_1,A_1)$$
$$\vdots$$
$$E(U_{A1}) = P(X_{11}) \times U_{11} + P(X_{12}) \times U_{12} + \ldots + P(X_{1K}) \times U_{1K}.$$

Theoretically, if the expected utility of each design alternative can be determined, the decision-maker can optimize the decision by choosing the alternative with the largest expected utility. Unfortunately, attempts to achieve perfect rationality through this kind of analysis will not always work, especially in systems exhibiting *Organized Complexity* because:

1. People will approach this analysis with varying levels of rigor,
2. The information necessary to perform these calculations may not be available or may be too difficult to acquire.

When presented with imperfect information, time limitations, and/or complexities that exceed the limits of human cognitive abilities, people act with a rationality that is said to be "bounded" [10], [11]. This is not to say that people act in an unintelligent manner, rather, it is meant to indicate that the basis on which people make decisions is "intendedly" rational, that is, that their decisions would produce sensible results if their model of the relevant system is correct [11], [12]. When acting upon bounded rationality, people rely on heuristics or rules of thumb and satisficing—a practice where instead of evaluating all options, the analyst stops whenever he or she finds a solution that is "good enough"—for decision-making [10], [11].

With that said, the concept of a decision tree is not inconsistent with bounded rationality—in fact, attempts have been made to model bounded rationality through event trees [12]. Instead, decision trees capture a specific type of intended rationality; one that is implicit in basing the choice of a design option on questions such as, "Will events associated with the hazard occur enough to warrant action?" Even though individuals and organizations may vary in the way that they construct their decision trees, may forgo the formal construction of a tree at all, and/or satisfice in their search for the best design alternative, their intentions will be similar. Decision makers with this intended rationality will be influenced by some notion of a series of events and outcomes that will be triggered by the selection of each option and their level of concern over these events will be dictated by their perception of the frequency and severity of these outcomes. Consequently, even if the outcomes associated with an option are severely negative, the decision maker(s) may not be deterred if they are confident that they can break the chain of events by making at least one of them so infrequent that it will be too unlikely to occur.

Generally, the *Organized Complexity* of many systems that exist today challenges the intended rationality of this decision process by making quantitative models of the system immensely complex and subject to various uncertainties. The fundamental assumptions that would have to be made in this type of analysis are that all of the branches stemming from each chance event node are mutually exclusive (i.e. they cannot occur together) and collectively exhaustive (i.e. they account for all possible outcomes given the chance event). Unfortunately, mutual exclusivity is a slippery concept in complex systems; many complex systems accidents over the last century have involved the coupled occurrence of events whose concurrence was initially viewed as impossible or highly unlikely [1]. Additionally, complex systems have far too many states to analyze exhaustively. One could not analyze every state of a typical laptop computer, for instance, even if he/she were able to program every atom in the universe to analyze

10^{100} states per second starting at the Big Bang and continuing to now [13].

Another limiting assumption in the intended rationality captured in Figure 4 is that the system will behave according to the law of large numbers. This law states that the system's performance will converge over time to the average value. At best, those who manage risk around this assumption must be prepared to accept some losses as the system's performance converges to the average success rate (e.g. a system with a 95% success rate can have five failures in a row followed by ninety-five successes). While there are a number of systems where a calculated number of losses will not thwart the objectives of the system stakeholders (e.g. large constellations of identical spacecraft, insurance policies for motorists in a given geographic area, etc.), there are also a number of systems where just one failure is unacceptable for a variety of reasons (e.g. nuclear weapons, crewed spacecraft, etc.).

Another key challenge in the modeling of the probability and severity of events associated with hazards is capturing the dynamic nature of systems exhibiting *Organized Complexity*. In these systems, the frequency and severity of events associated with specific hazards may increase or decrease over time. For example, as demonstrated on XTE, SBU and MBU rates on LEO spacecraft decrease over their operational lifetime if their lifetime starts during Solar Minimum and continues through Solar Maximum or if the orbit decays appreciably [7]. Additionally, the criticality of the spacecraft's software will evolve as the spacecraft's configuration evolves throughout the mission (e.g. solar array deployment software may become non-critical after solar array deployment). Thus the risk calculated from an analysis with static values for MBU rates and software criticality would almost always be over- or underestimated depending on the values selected. Unfortunately, dynamic modeling of the evolution of system risk is limited as well in the precision that it can add to the estimations as it introduces further complexity and assumptions into the analysis. For example, one technique in probabilistic risk assessment is to use Markov modeling to evaluate system evolution [14], [15]. This technique introduces transition matrices—and in some cases, transition matrices embedded within other transition matrices—into the analysis. These matrices add uncertainty into the calculations because they have to be populated with esoteric parameters (e.g. the probability of transitioning into state Y in a discrete time interval given that the system is in state X) that must be quantified. Data on these parameters simply may not exist and/or the concepts behind these parameters may be too abstract for system experts that deal with them to properly encode them into probabilities. Indeed, empirical studies have shown that cognitive biases often lead laypeople and experienced researchers alike to severe and systemic errors in the assessment of probabilities [16].

In the specific case of a formal or informal probabilistic decision analysis of upset mitigation strategies for a spacecraft mission, additional assumptions and complexity are necessary in the computation of the probabilistic distribution of the outcomes of each event. In the next few paragraphs, these assumptions and complexities are discussed for each event in the decision tree in Figure 4. These events are defined as follows:

E_1 = The system performs an action that will allow harmful propagation of the upset event
E_2 = A critical logical word in the software is upset
E_3 = The upset word executes.

In calculating the probability of the system being in a state that would allow harmful propagation of the upset event, $P(E_1)$, the definition of which outcomes constitute a harmful system response will add additional uncertainty to the calculations. Unfortunately, in systems exhibiting *Organized Complexity*, these definitions will usually be highly context specific. In other words, the transferability of data between systems displaying *Organized Complexity* is limited. For instance, a reboot event leading to several days of missed data collection opportunities could have a minor effect on a spacecraft mission that is meant to collect data over many years while a spacecraft mission with a data collection period on the order of months may be seriously affected. Thus, if one were to look at the successes and failures of previous spacecraft missions to define these criteria, he or she may be misled.

In calculating the probability of a critical word being upset, given that the system is performing an action that will allow harmful propagation of the upset event, $P(E_2|E_1)$, assumptions surrounding the transferability of data between spacecraft missions once again play a critical role in the accuracy of the calculation. The software architectures of spacecraft and the hardware architectures that support them are vastly different from spacecraft to spacecraft. One must first understand how the differences in the architectures will affect the rates of MBUs in the spacecraft. This of course assumes that the analyst would have prior knowledge of the physical mechanisms relevant to MBU frequency. As demonstrated by the data from XTE, GP-B, and Cassini, one cannot use the MBU rates on memory devices with logically adjacent bits that are physically distributed to predict MBU rates on memory devices in which the logically adjacent bits are also physically adjacent. Additionally, one must pay attention to how the uniqueness of the spacecraft's software and hardware architecture will dictate the potential impact of an MBU in a given word. One cannot look at the minimal impact of MBUs on XTE and conclude that in general a small portion of the software on spacecraft will respond poorly to MBUs. While XTE did not use the portion of its onboard memory that was most vulnerable to MBUs to run its flight software, GP-B and Cassini did and therefore they each had a higher percentage of critical software that was vulnerable to MBUs. Moreover, GP-B's software architecture included a safemode response that would reboot one of its computers if three or more logical words—regardless of the function that they served—were detected within 0.3 seconds of each other. This ultimately had the effect of increasing the criticality of non-critical words.

Finally, calculating the probability of the execution of an upset word, given that it's critical and that the system is performing an action that will allow harmful propagation of the upset event, $P(E_3|E_1,E_2)$, will require knowledge of how often words are accessed for execution or overwritten before they are executed and the frequency and efficacy of human and EDAC intervention. While these factors are largely dictated by component-level behavior and are thus simpler to quantify than the factors affecting the other events in the chain, the analysis of them is still somewhat complicated by system-level behavior. For instance, the frequency and efficacy of human intervention over the life of the mission could be reduced by factors such as operator/management complacency, training, and/or staffing. Additionally, all of the factors can be affected by operational workarounds implemented throughout the mission as well as the addition of new mission objectives during operations.

In all, the *Organized Complexity* exhibited by complex systems creates a number of problems for the quantitative, statistical assessment of hazards, such as the MBU hazard to spacecraft. A number of assumptions are often necessary to make this type of analysis tractable and these assumptions introduce significant uncertainty into the estimates. Additionally, the bounded rationality of both the analysts and the actual individuals making the design decisions will play a role in both the rigor in which these numbers are generated and interpreted. While the analyst may strive for perfect rationality to the extent that it is humanly possible, the actual decision-maker may simply wish to select a particular design option provided that the quantified likelihood of a negative outcome associated with it meets a company or industry standard—even if, unbeknownst to him/her, that standard is not appropriate for the specific application. Thus, a question is in order for those who would base design decisions on the frequency or probability of events associated with a hazard: "What happens if the estimates are wrong and/or interpreted in the wrong way?"

B. An example qualitative assessment of MBU hazards

The many subtleties of risk in systems that display *Organized Complexity* cannot be explained fully in quantitative terms. Both the derivation and interpretation of each quantitative figure produced in a risk assessment are informed by qualitative concepts. In the selection of radiation-hardened parts for spacecraft, for example, there are many qualitative considerations that effect data integrity and applicability [17]. However, there is a tradeoff of sorts in the overall use of qualitative and quantitative information in decision-making. In the probabilistic assessment of risk, the paradigm is to base the decision largely on a quantitative assessment of an expected utility of the design alternative derived from a notion of the frequency of events associated with a hazard. While this paradigm is intendedly rational, it is subject to the miscalculation and misinterpretation of the event frequency. In the remaining paragraphs of this section, an approach based on the assessment of a qualitative concept, hazard controllability, will be explored.

STAMP (Systems-Theoretic Accident Modeling and Process) is a qualitative approach to hazard analysis and mitigation based on systems theory that stresses the control of hazards. In STAMP, accidents are conceived as resulting not from component failures, but from inadequate control or enforcement of safety-related constraints on the development, design, and operation of the system. The most basic concept in STAMP is not an event, but a constraint. Safety is viewed as a *control problem*: accidents occur when component failures, external disturbances, and/or dysfunctional interactions among system components are not adequately handled [18].

In a STAMP-based Analysis (STPA), systems are viewed as interrelated components that are kept in a state of dynamic equilibrium by feedback loops of information and control. A system is not treated as a static design, but as a dynamic process that is continually adapting to achieve its ends and to react to changes in itself and its environment. The original design must not only enforce appropriate constraints on behavior to ensure safe operation, but it must continue to operate safely as changes and adaptations occur over time.

Thus, preventing accidents requires designing a control structure encompassing the entire socio-technical system that will enforce the necessary constraints on system development and operations. In a full STPA, this includes dynamic modeling of the intended rationality of the decision processes of the individuals and organizations designing and operating the system through System Dynamics. This modeling discipline is commonly used to test the feedback structures embodying the bounded rationality of decision-makers in the system [11], [19]. However, for the purpose of this paper, the discussion will be limited to the direct analysis of technical hazards such as MBUs.

The steps for a STPA are as follows:

1. Identify the system hazards,
2. Identify design constraints to control the hazards,
3. Assign control of the constraints to various elements of the system,
4. Define the control actions to be employed,
5. Evaluate the control structure for possible changes over time and fix potential problems

The relevant hazard for upset events is, "The software executes in an unintended manner." Thus, the constraint would be something like, "All software and all data written by the software must be read for execution as written by the software logic and/or human programmers." With this constraint defined, the system designers would then implement one or more of the following techniques to enforce the constraint [3]:

Hazard Elimination

- Substitution
- Simplification
- Decoupling
- Elimination of specific human errors
- Reduction of hazardous materials or conditions

Hazard Reduction

- Design for controllability
- Use of Barriers
- Failure Minimization
    - Safety factors and safety margins
    - Redundancy

Hazard Control

- Reducing Exposure
- Isolation and Containment
- Protection Systems and fail-safe design

Damage Reduction.

It is important to note that techniques to minimize the expected losses by reducing the predicted frequencies of hazardous events are technically ways in which designers can attempt to enforce safety constraints. However if the explicit goal in their implementation is a reduction in event frequency rather than hazard controllability, they may represent an *open-loop control* philosophy. In other words, they could be control actions that are meant to work in accordance with a model of the system response that does not take into account feedback from the system. The problem with an *open-loop control* philosophy is that if the model of the system response is wrong, the control action will be inappropriate and potentially uncorrectable. It is sometimes possible for system operators to "close the loop" on control actions designed with an *open-loop control* philosophy through operational workarounds. However, this is not always the case. For instance, Cassini and GP-B both used memory devices that yielded higher MBU rates than what was predicted and these devices, of course, could not be changed after launch. Thus, control actions should be designed with a *closed-loop control* philosophy. Even if the nature of the hazard and/or the budget environment in which the control action is being implemented make it impossible for the control action to be designed for closed-loop control, the system stakeholders should explicitly recognize and prepare for the open-loop nature of the control action.

After the general control structure has been defined (or a candidate structure has been identified), the next step is to determine how the controlled system could get into a hazardous state. This includes identification of potential inadequate control actions that could lead to the hazardous state. In general, a safety constraint controller can provide four general types of inadequate control:

1. A required control action is not provided.
2. An incorrect or unsafe control action is provided.
3. A potentially correct or adequate control action is provided too late (at the wrong time).
4. A correct control action is stopped too soon.

Control actions may be required to handle component failures, environmental disturbances, or dysfunctional interactions among the components. Incorrect or unsafe control actions may cause dysfunctional behavior or interactions among components.

On GP-B, XTE, and Cassini, there were various control actions to enforce the constraint "All software and all data written by the software must be read for execution as written by the software logic and/or human programmers,"—though they were not implemented with this constraint explicitly in mind. Several of the actions were open-loop in nature, such as:

- XTE's flight software operating off of a memory board that was more radiation hardened than its SSR,
- Cassini and GP-B using memory devices with what were expected to be acceptable MBU rates,

Other control actions on these spacecraft were closed-loop such as:

- SBU correction by EDAC on all three spacecraft,
- MBU patching through human intervention on all but one of GP-B's computers,
- The utilization on GP-B of a safemode response on one computer that had an adjustable threshold for rebooting the computer at the detection of multiple MBUs,
- The allocation and utilization of margin in the schedule for data collection.

Improvements that might have been realized with an explicit *closed-loop control* philosophy include:

- The implementation of EDAC schemes that could correct MBUs,
- The ability to manually patch MBUs on all computers,
- The implementation of safemode responses to upsets in software areas that the operators define as critical.

None of these techniques would require prior knowledge of MBU frequency to be effective and each would give the operations teams more flexibility to react to erroneous predictions of MBU frequency. Furthermore, the third technique would represent an upgrade over the safemode response implemented on one of GP-B's computers. Even though the sensitivity of that response could be altered, its focus was on MBU frequency rather than criticality and thus it may have inadequately enforced the constraint at times by rebooting the computer when non-critical words were upset. Thus, the second and third techniques could have allowed the GP-B operations team to prevent up to four of the five reboot events —one occurred because a computer could not be patched and three were due to the safemode response.

VI. CONCLUSIONS

As demonstrated by occurrences on the XTE, Cassini, and GP-B spacecraft missions, even a stochastic event such as a radiation strike on computer memory cells can be difficult to evaluate statistically in the context of systems exhibiting *Organized Complexity*. While system components, such as memory devices can exhibit *Unorganized Complexity* in their response to phenomena when they are considered individually, the same is not true when they are integrated into complex systems. The underlying structure of the systems they operate in make statistical evaluations very difficult without the introduction of assumptions that lead to high levels of uncertainty in the final result. Furthermore, the bounded rationality of the individual(s) that use the evaluations to make risk reduction design decisions may undermine an analyst's best efforts to produce a perfectly rational solution to the problems raised by the hazard.

If these components are to be used in unique, complex systems, strategies for their design and integration into the systems should not be centered on combating *Unorganized Complexit*y. In other words, designers should not seek to address the question, "How can the frequency of events associated with the hazard be reduced to point where they do not warrant further consideration?" Statistical or probabilistic analyses of the system response to these hazards can be wrong/misleading for a number of reasons. Additionally, designing and implementing a controllable response is more important than estimating the response frequency. Unpleasant, yet relatively benign events can occur multiple times while a critical event may need to occur only once to destroy the system.

The design and integration strategies for components—such as memory devices—to be used in unique, complex systems should center on dealing with *Organized Complexity*. It is important for designers to include mechanisms for control of system-level hazards or at the very least to augment the use of quantitative metrics in design decisions with a deep, qualitative understanding of how the risk can be controlled if the metrics are not representative of the actual integrated system's behavior. Doing so may allow both the analyst and decision-maker(s) to avoid the many pitfalls in the quantitative analysis of risk in systems that display *Organized Complexity*.

ACKNOWLEDGEMENTS

The authors would like to extend thanks to Kenneth LaBel for answering questions on the SBU/MBU events on XTE and the following members of the GP-B operations and science team for their involvement in the investigation of GP-B radiation events: Steve Larsen, Gary Lum, and Lou Zampetti. Additionally, the authors are indebted to the following individuals for their comments on draft versions of this paper: William Bencze, Paul Shestople, Nicolas Dulac, and Joseph Laracy.

This research has been accomplished with funding from the Universities Space Research Association (USRA) through Cooperative Agreement No. 05115-C1P2-01.

REFERENCES

[1] Charles Perrow, *Normal Accidents: Living with High-Risk Technologies.* 1999 Edition. Princeton Books, 1999.

[2] Gerald Weinberg, *An Introduction to General Systems Thinking.* New York: John Wiley & Sons, 1975.

[3] Nancy G. Leveson, *Safeware: System Safety and Computers.* Addison-Wesley Publishing Co., 1995.

[4] Koga, R.; Pinkerton, S.D.; Lie, T. J.; and Crawford, K.B., "Single-word multiple bit upsets in static random access devices," *IEEE Transactions on Nuclear Science*, Vol. 40, no. 6, pp. 1941-1946, Dec 1993.

[5] Owens, Brandon D.; Adams, Michael E.; Bencze, William J.; Green, Gaylord; and Shestople, Paul, "The Effects of Radiation Events on Gravity Probe B", *Proceedings of the 9th Annual Military and Aerospace Programmable Logic Devices International Conference (MAPLD),* Sept., 2006. To Be Published.

[6] Swift, Gary M. and Guertin, Steven M., "In-Flight Observations of Multiple Bit Upsets in DRAMs," *IEEE Transactions on Nuclear Science*, Vol. 47, no. 6, pp. 2386-2391, 2000.

[7] Poivey, Christian; Gee, George; LaBel, Kenneth A.; and Barth, Janet L., "In-Flight Observations of Long-Term Single Event Effect (SEE) Performance on X-ray Timing Explorer (XTE) Solid State Recorders (SSRs)," *Radiation Effects Data Workshop,* 2004 IEEE July 19-23, 2004 pp. 54-57.

[8] Kenneth LaBel, private communication.

[9] Shestople, Paul et al. "Gravity Probe B: Experiment and Mission," in *Proceedings of the 14th Japanese Conference for General Relativity and Gravitation,* 2005

[10] Herbert Simon, *Models of Thought*, Yale University Press, 1979, pg. 3-19.

[11] John Sterman, *Business Dynamics: Systems Thinking and Modeling for a Complex World*, McGraw-Hill, 2000.

[12] Murphy, Dean M. and Pate-Cornell, M. Elisabeth. "The SAM Framework: Modeling the Effects of Management Factors on Human Behavior in Risk Analysis," *Risk Analysis*, Vol. 16, No. 4, 1996. pp. 501-515.

[13] Wulf, William A., "Great Achievements and Grand Challenges," *The Bridge*, Vol. 30, no. ¾, pp. 5-10, 2000.

[14] Pate-Cornell, M. Elisabeth; "Fire Risks in Oil Refineries: Economic Analysis of Camera Monitoring," *Risk Analysis*, Vol. 5, No. 4, 1985. pp. 277-288.

[15] Pate-Cornell, Elisabeth; Murphy, Dean; Lakats, Linda; and Gaba, David, "Patient risk in anesthesia: Probabilistic risk analysis and management improvements," *Annals of Operations Research*, Vol. 67, 1996. pp. 211-233.

[16] Tversky, Amos and Kahneman, Daniel; "Judgment under Uncertainty: Heuristics and Biases," *Science*, Vol. 185, pp. 1124-1131, September 1974.

[17] LaBel, K. A.; Johnston, A. H.; Barth, J. L.; Reed, R. A.; and Barnes, C. E., "Emerging Radiation Hardness Assurance (RHA) Issues: A NASA Approach for Space Flight Programs," *IEEE Transactions on Nuclear Science*, Vol. 45, no. 6, pp. 2727-2736, December 1998.

[18] Leveson, Nancy G., "A New Approach to Hazard Analysis in Complex Systems," *International Conference of the System Safety Society*, Ottawa, Canada. August 2003.

[19] Morecroft, John D. W., "System Dynamics: Portraying Bounded Rationality," *Omega-The International Journal of Management Science*, Vol. 11, no. 2, pp. 131-142. 1983.