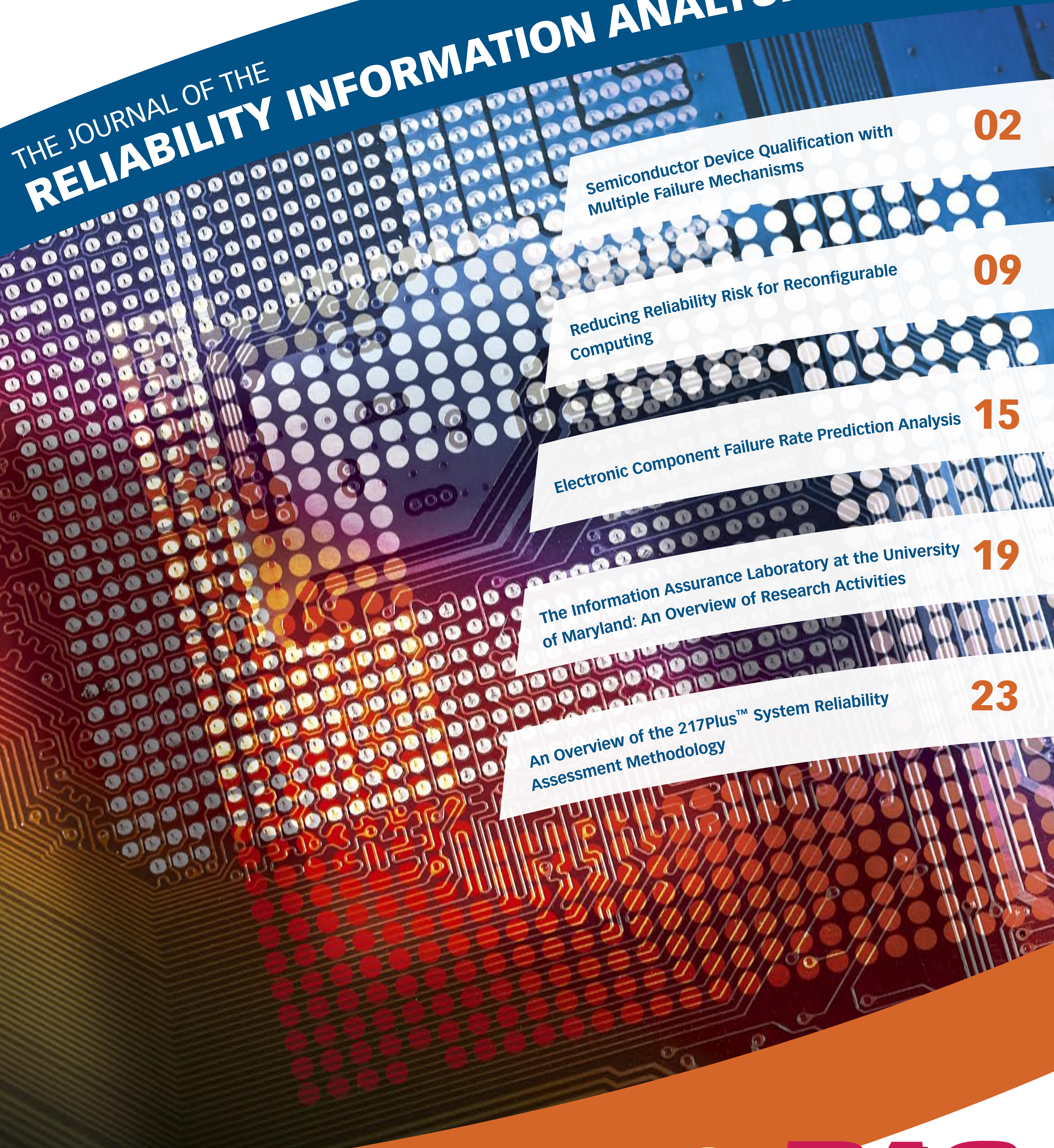
FOURTH QUARTER - 2006

THE JOURNAL OF THE RELIABILITY INFORMATION ANALYSIS CENTER



RIAC

Semiconductor Device Qualif

*This paper was presented at the 16th International Conference of the Israeli Society for Quality, November 2006

Joseph B. Bernstein and Jin Qin, University of Maryland

Microelectronics integration density is limited by the reliability of the manufactured product at a desired circuit density. Design rules, operating voltage and maximum switching speeds are chosen to insure functional operation over the intended lifetime of the product. In order to determine the ultimate performance for a given set of design constraints, the reliability must be modeled for its specific operating condition. Thus, Reliability modeling for the purpose of lifetime prediction is the ultimate task of a failure physics evaluation. Unfortunately, all the industrial approaches to reliability evaluation fall short of predicting failure rates or wear-out lifetime of semiconductor products. This is attributed mainly to two reasons; the lack of a unified approach for predicting device failure rates and the fact that all commercial reliability evaluation methods rely on the acceleration of one, dominant, failure mechanism.

Over the past several decades, our knowledge about the root cause and physical behavior of the critical failure mechanisms in microelectronic devices has grown significantly. Confidence in the reliability models have lead to more aggressive design rules that have been successfully applied to the latest VLSI technology. One result of improved reliability modeling has been accelerated performance, beyond the expectation of Moore's Law. A consequence of more aggressive design rules has been a reduction in the weight of a single failure mechanism. Hence in modern devices, there is no single failure mode that is more likely to occur than any other as guaranteed by the integration of modern failure physics and modern simulation tools in the design process. The consequence of more advanced reliability modeling tools is a new phenomenon of device failures resulting from a combination of several competing failure mechanism.

Today, reliability device simulators have become an integral part of the design process. These simulators successfully model the most significant physical failure mechanisms in modern electronic devices, such as Time Dependent Dielectric Breakdown (TDDB), Negative Bias Temperature Instability (NBTI), Electromigration (EM) and Hot Carrier Injection (HCI). These mechanisms are modeled throughout the circuit design process so that the system will operate for a minimum expected useful life. Modern chips are composed of tens or hundreds of millions of transistors. Hence, chip level reliability prediction methods are mostly statistical. Reliability prediction tools, now model the failure probability of chips at the end of life by analyzing only the single dominant wearout mechanism. Modern prediction tools do not predict the random, post burn-in, failure rate that would be seen in the field.

Chip and packaged system reliability is still measured by failure rate in FIT. The FIT is a unit, defined as one failure per billion part hours. The semiconductor industry provides an expected FIT for every product that is sold based on operation within the specified conditions of voltage, frequency, heat dissipation and etc. Hence, a system reliability model is a prediction of the expected mean time between failures (MTBF) for an entire system as the reciprocal of the sum of the FIT rates for every component. The failure rate of a component can be defined in terms of an acceleration factor, AF, as (Equation 1):

$$\lambda = \frac{Number\,of\,failures}{Number\,of\,tested \times hours \times AF} \times 10^9 FIT$$

where "Number of failures" and "Number of tested" are the number of actual failures that occurred as a fraction of the total number of units subjected to an accelerated test. The acceleration factor, AF, must be supplied by the manufacturer since only they know the failure mechanisms that are being accelerated in the High Temperature Operating Life (HTOL) and it is generally based on a company proprietary variant of the MIL-HDBK-217 approach for accelerated life testing. The true task of reliability modeling, therefore, is to choose an appropriate value for AF based on the physics of the dominant failure mechanisms that would occur in the field for the device.

The HTOL qualification test is usually performed as the final qualification step of a semiconductor manufacturing process. The test consists of stressing some number of parts, usually about 100, for an extended time, usually 1000 hours, at an accelerated voltage and temperature. Two features shed doubt on the accuracy of this procedure. One feature is lack of sufficient statistical data and the second is that companies generally present zero-failure results for their qualification tests. Parts are stressed at relatively low levels to guarantee zero failures during qualification testing in accordance with their guidelines. Zero failures results in zero real data and the true statistical confidence is mathematically imaginary. The assumption, then, is that no more than one failure occurred during the accelerated test and substitute 1/2 failure to circumvent this imaginary confidence dillema. This results, based on our example parameters, in a reported FIT = 5000/AF, which can be almost any value from less than 1 FIT to more than 500 FIT, depending on the conditions and model used for the voltage and temperature acceleration.

The accepted approach for measuring FIT would, in theory, be reasonably appropriate if there is only a single dominant failure mechanism that is excited equally by either voltage or temperature. For example, EM is known to follow Black's equation (described later) and is accelerated by increased stress current in a wire or by increased temperature of the device. If, however, multiple failure mechanisms are responsible for device failures, each failure mechanism should be modeled as an individual "element" in the system and the component survival is modelled as the survival probability of all the "elements" as a function of time.

If multiple failure mechanisms, instead of a single mechanism, are assumed to be time-independent and independent of each other, FIT (constant failure rate approximation) should be a reasonable approximation for realistic field failures. Under the assumption of multiple failure mecha-

ication with Multiple Failure Mechanisms

nisms, each will be accelerated differently depending on the physics that is responsible for each mechanism. If, however, an HTOL test is performed at an arbitrary voltage and temperature for acceleration based only on a single failure mechanism, then only that mechanism will be accelerated. In that instance, which is generally true for most devices, the reported FIT (especially one based on zero failures) will be meaningless with respect to other failure mechanisms.

Individual Failure Mechanism Lifetime Model

Relentless scaling for better performance keeps generating new reliability challenges to every aspects of the process technology. EM, the main reliability concern of interconnects, needs to be handled carefully because feature size decreasing and temperature increasing pose dual threats towards new interconnect technology. To meet the performance and reliability requirement, copper interconnects have gradually take the place of Al(Cu) metallization in the past few years, due to its low resistivity and high resistance towards electromigration. Copper interconnects have different EM characteristics compared with aluminum. It is interface dominated [1] and has larger activation energies [2].

TDDB has always received much attention because device scaling keeps driving the oxide thickness down but the supply voltage scaling doesn't keep pace. The direct impact of this non-ideal voltage scaling is the increase of gate leakage and tunneling current which decreases the oxide lifetime. An empirical observation is that if gate oxide thickness reduces by ΔT_{ox} (in nm) by scaling, the leakage current will increase [3] by:

$$10^{\frac{\Delta T_{ox}}{0.22}}$$

and TDDB lifetime will reduce by the same factor. Oxide breakdown related failures are often reported in device burn-in test of deep submicron technologies [4], [5]. Device scaling also increases susceptibility to another failure mechanism: NBTI, which occurs primarily in p-channel MOSFETs with negative gate voltage bias. The interface-trap density generated by NBTI has an inverse proportionality to oxide thickness (T_{ox}) which means NBTI becomes more severe for ultrathin oxides [6], while the NBTI generated fixed charge has no thickness dependence. Like NBTI for PMOS, HCI induces interface states and causes degradation of NMOS. Although well contained by channel engineering, it still shows up in real applications [7].

To model system reliability, all these intrinsic failure mechanisms should be considered since any one of them may cause system failure. Various lifetime models have been proposed for each failure mechanism. As our goal is to show the unique characteristics of system lifetime and voltage and temperature acceleration, we will adapt the generally accepted models.

Failure rate model and acceleration factors for EM, HCI, TDDB and NBTI are listed below.

(1) EM

From the well known Black's equation [8] and Arrhenius model, the failure rate for EM can be expressed as (Equation 2):

$$\lambda_{EM} \propto (J)^n \cdot \exp[\frac{-E_{aEM}}{kT}]$$

where J is the current density in the interconnect, k is Boltzmann's constant, T is absolute temperature in Kelvin, E_{aEM} is the activation energy, and n is a constant. Both E_{aEM} and n depend on the interconnect metal. Nowadays copper/low-K dielectric material has been rapidly replacing aluminum alloy/SiO_2-based interconnect. For copper, n has been reported with values between 1 and 2 [1] and E_{aEM} varies between 0.7eV and 1.1eV [9]. In Eq. (2), current density J can be replaced with a voltage function (Equation 3)[10]:

$$J = \frac{C \cdot V_D}{W \cdot H} \cdot f \cdot p$$

where C, W and H are the capacitance, width and thickness of the interconnect, respectively. f is the frequency, p is the toggling probability. So λ_{EM} is also a function of voltage (Equation 4):

$$\lambda_{EM} \propto (V_D)^n \cdot \exp[\frac{-E_{aEM}}{kT}]$$

(2) HCI

Based on the empirical HCI voltage lifetime model proposed by Takeda [11] and the Arrhenius relationship, HCI failure rate HCI can be modeled as (Equation 5):

$$\lambda_{HCI} \propto \exp[\frac{-\gamma_{HCI}}{V_D}] \cdot \exp[\frac{-E_{aHCI}}{kT}]$$

where γ_{HCI} is a technology related constant, E_{aHCI} is the activation energy, varies between −0.1eV ~ −0.2eV [12]. The negative activation energy means HCI becomes worse at low temperature.

(3) TDDB

The exponential law for TDDB failure rate voltage dependence has been widely used in gate oxide reliability characterization and extrapolation. Combining with the Arrhenius relationship for temperature dependence, TDDB failure rate (Equation 6) is:

$$\lambda_{TDDB} \propto \exp[\gamma_{TDDB} \cdot V_G] \cdot \exp[\frac{-E_{aTDDB}}{kT}]$$

where γ_{TDDB} is a device related constant and E_{aTDDB} is the activation energy. E_{aTDDB} normally falls in the range of 0.6eV ~ 0.9eV [12].

continued on page 4

Semiconductor Device Qualification with Multiple Failure Mechanisms

continued from page 3

(4) NBTI
Like TDDB, NBTI voltage dependence can also be modeled by the exponential law [13], considering the temperature dependence together, NBTI failure rate is (Equation 7):

$$\lambda_{NBTI} \propto \exp[\gamma_{NBTI} \cdot V_G] \cdot \exp[\frac{-E_{aNBTI}}{kT}]$$

where γ_{NBTI} is a constant and E_{aNBTI} is the activation energy which has been reported to vary from 0.1eV to 0.84eV [14], [15].

System Voltage and Temperature Acceleration

Assuming there is no interaction among failure mechanisms, a system's failure rate can be obtained by sum-of-failure-rate since all failure mechanisms contribute to system failures (Equation 8):

$$\lambda_S = \lambda_{EM} + \lambda_{HCI} + \lambda_{TDDB} + \lambda_{NBTI}$$

The system acceleration factor can be expressed as (Equation 9):

$$AF_S = \frac{\lambda_S^{V_A,T_A}}{\lambda_S^{V_O,T_O}} = \frac{\lambda_{EM}^{V_A,T_A} + \lambda_{HCI}^{V_A,T_A} + \lambda_{TDDB}^{V_A,T_A} + \lambda_{NBTI}^{V_A,T_A}}{\lambda_{EM}^{V_O,T_O} + \lambda_{HCI}^{V_O,T_O} + \lambda_{TDDB}^{V_O,T_O} + \lambda_{NBTI}^{V_O,T_O}}$$

Given the models of individual failure mechanisms, the system acceleration factor (Equation 9) can be further expressed as (Equation 10):

$$AF_S = P_E^{V_O,T_O} \cdot AF_{EM} + P_H^{V_O,T_O} \cdot AF_{HCI} + P_T^{V_O,T_O} \cdot AF_{TDDB} + P_N^{V_O,T_O} \cdot AF_{NBTI}$$

where,

$$P_E^{V_O,T_O},\ P_H^{V_O,T_O},\ P_T^{V_O,T_O} \text{ and } P_N^{V_O,T_O}$$

are failure percentages of EM, HCI, TDDB and NBTI at stress conditions (V_O, T_O), respectively. The advantage of using these failure percentages here is to simplify the derivation process without the need to find out the absolute failure rate for each failure mechanism.

For property issues, original microelectronic device lifetime data is rarely reported in literature. In order to reveal the characteristics of temperature and voltage acceleration at the system level, we do lifetime simulation by using the models given above. The system is assumed to be made with 0.13µm technology and the oxide thickness is 3.2nm. Nominal operating conditions are V_O = 1.3V , T_O = 75°C. HCI, TDDB and NBTI are assumed to contribute equally to system failures at nominal conditions. All the acceleration parameters are extracted from published result related to 0.13µm technology (HCI [16], TDDB [17] and NBTI [18]) and listed in Table 1. We assume V_O = 1.3V, T_O = 75°C.

Table 1. Simulation Parameters for EM, HCI, TDDB and NBTI

	Voltage Acceleration Parameter	Activation Energy (eV)	Failure Percentage
EM	2	1.2	25%
HCI	16	-0.2	25%
TDDB	12	0.7	25%
NBTI	6	0.4	25%

A. Non-Arrhenius Temperature Acceleration

Designate $E_{aSYS}^{V_i,T_i}$ as the activation energy estimated from accelerated tests at (V_i,T_i) and (V_i,T_A). If the Arrhenius relationship still holds at system level, $E_{aSYS}^{V_i,T_i}$ should be the same for all T_i and V_i. The system temperature acceleration factor AF_S^T can be calculated as (Equation 11):

$$AF_S^T = P_E^{V_i,T_i} \cdot AF_{EM}^T + P_H^{V_i,T_i} \cdot AF_{HCI}^T + P_T^{V_i,T_i} \cdot AF_{TDDB}^T + P_N^{V_i,T_i} \cdot AF_{NBTI}^T$$

where,

$$P_E^{V_i,T_i},\ P_H^{V_i,T_i},\ P_T^{V_i,T_i} \text{ and } P_N^{V_i,T_i}$$

are the percentages of EM, HCI,TDDB and NBTI failure at (V_i,T_i), respectively. Using the parameters given in Table I and set T_A=125°C, we did E_{aSYS} estimation at various T_i under three voltages: 1.17V, 1.30V and 1.43V and show the results in Fig. 1. The simulation result clearly shows that E_{aSYS} is not a constant. It depends on the stress voltage V_i and the stress temperature T_i. At given V_i, $E_{aSYS}^{V_i,T_i}$ is an increasing function of T_i. The reason is that the failure mechanism with the larger activation energy will increase its failure percentage at high temperature at a given stress voltage. For illustration, if $|T_A - T_i|$ is considerably small, system activation energy can be approximated by (Equation 12):

$$E_{aSYS}^{V_i,T_i} = P_{EM}^{V_i,T_i} \cdot E_{aEM} + P_{HCI}^{V_i,T_i} \cdot E_{aHCI} + P_{TDDB}^{V_i,T_i} \cdot E_{aTDDB} + P_{NBTI}^{V_i,T_i} \cdot E_{aNBTI}$$

From Equation 12, we can find that at any given E_{aEM}, E_{aHCI} , E_{aTDDB} and E_{aNBTI} , $E_{aSYS}^{V_i,T_i}$ depends on:

$$P_E^{V_i,T_i},\ P_H^{V_i,T_i},\ P_T^{V_i,T_i} \text{ and } P_N^{V_i,T_i}$$

The failure mechanism with the largest activation energy will be accelerated the most as temperature increases and its failure percentage will increase accordingly.

As E_{aSYS} is generally estimated from high temperature acceleration testing, using that activation energy tends to give an optimistic projection at low temperature. For an example, if the acceleration tests were done at (1.43V, 125°C) and (1.43V, 115°C), the estimated E_{aSYS} is 1.0eV . Using this activation energy to extrapolate system failure rate at (1.43V, 50°C) will result in an optimistic estimation which is 1/14 of the real rate because the 'true' E_{aSYS} is 0.60eV .

B. Stress-Dependent Voltage Acceleration Factor
To show the characteristic of voltage acceleration, we assume that AF_S^V follows the exponential law (Equation 13):

$$AF_S^V = \exp[\gamma_{SYS}^{V_i,T_i} \cdot (V_A - V_i)]$$

where $\gamma_{SYS}^{V_i,T_i}$ is the voltage acceleration parameter. AF_S^V is shown below (Equation 14):

$$AF_S^V = P_E^{V_i,T_i} \cdot AF_{EM}^V + P_H^{V_i,T_i} \cdot AF_{HCI}^V + P_T^{V_i,T_i} \cdot AF_{TDDB}^V + P_N^{V_i,T_i} \cdot AF_{NBTI}^V$$

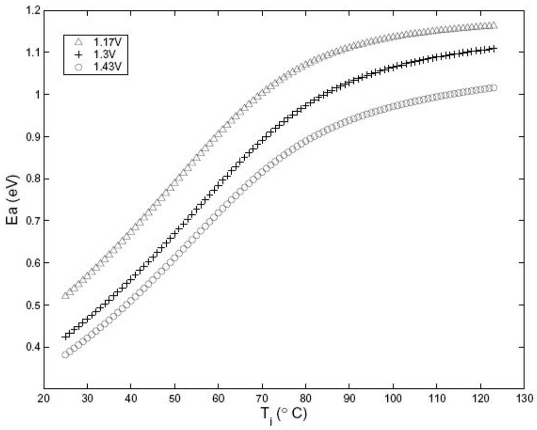


Figure 1. System activation energies estimated from simulated failure rate at (V_i, T_i) and (V_i, T_A). V_i=1.17V, 1.30V and 1.43V. At given V_i, T_A=125 °C and T_i varies from 25°C to 124°C.

where,

$$P_E^{V_i,T_i},\ P_H^{V_i,T_i},\ P_T^{V_i,T_i}\ \text{and}\ P_N^{V_i,T_i}$$

have the same meaning as in Eq. (11). Simulation was done with parameters given in Table 1 and the estimated γ_{SYS} is shown in Fig. 2. The result shows that γ_{SYS} varies according to V_i and T_i. For approximation, if the difference between V_A and V_i is reasonably small, $\gamma_{SYS}^{V_i,T_i}$ can be approximated by Equation 15:

$$b_{SYS}^{V_i,T_i} = P_E^{V_i,T_i} \cdot \frac{n}{V_i} + P_H^{V_i,T_i} \cdot \frac{\gamma_{HCI}}{V_i^2} + P_T^{V_i,T_i} \cdot \gamma_{TDDB} + P_N^{V_i,T_i} \cdot \gamma_{NBTI}$$

Like $E_{aSYS}^{V_i,T_i}$, $\gamma_{SYS}^{V_i,T_i}$ also depends on the failure percentages and the voltage acceleration parameters. As shown in Fig.2, at 125°C, γ_{aSYS} is larger at higher stress voltage because TDDB together with NBTI dominate here and the higher voltage accelerates them more than EM and HCI.

Using γ_{SYS} estimated at (125°C, 1.55V) to extrapolate system failure rate at low voltage will give an optimistic estimation. At 125°C and V_i = 1.55V, γ_{SYS} is estimated to be 10.0, while we will get 7.0 if V_i = 1.30V . There is about 5X difference in failure rate extrapolation.

C. Combined Voltage and Temperature Acceleration Factor

The effect of voltage and temperature acceleration together on system acceleration is further complicated by the interplay between the factors, as shown above. Since there is no universal E_{aSYS} and γ_{SYS} if multiple failure mechanisms are involved, using AF_T with one activation energy and AF_V with one voltage acceleration parameter for reliability extrapolation is not appropriate. Taking the simulation above as an example, we find out that failure rate estimation using the multiplication model gives an optimistic result. The real system failure rate at (50°C, 1.30V) is 20X that of the estimated failure rate using the multiplication model with E_{aSYS} and γ_{SYS} from high temperature, high voltage acceleration test at (125°C, 1.55V).

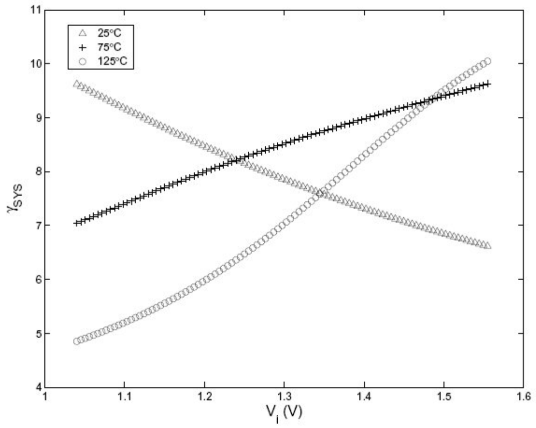


Fig. 2. Estimated γ_{SYS} from failure rates at accelerated conditions (V_i, T_i) and (V_A, T_i). T_i=25°C, 75°C and 125 °C. For each T_i, V_A = 1.56V , V_i varies from 1.04V to 1.55V.

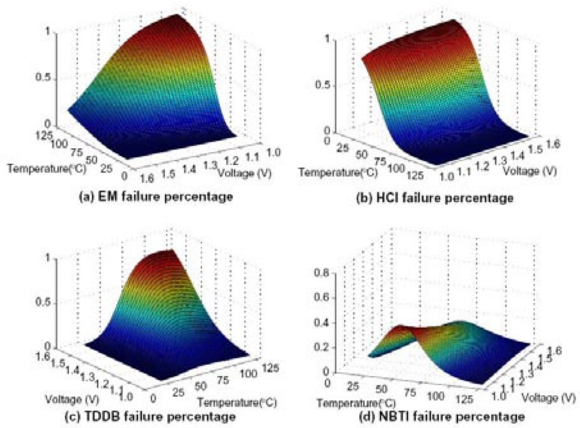

(a) EM failure percentage
(b) HCI failure percentage
(c) TDDB failure percentage
(d) NBTI failure percentage

Fig. 3. Failure percentages of EM, HCI, TDDB and NBTI at different accelerated conditions.

Qualification Based on Failure Mechanism

It is a matter of great complexity to build a system lifetime model to fit all temperatures and voltages if there are multiple failure mechanisms involved. The conventional extrapolation method using one E_{aSYS} and γ_{SYS} tends to give an optimistic estimation. For reliability qualification considering multiple failure mechanisms, acceleration tests should be designed to accelerate the target failure mechanism with specific stress conditions. This is workable because each failure mechanism has its unique activation energy and voltage acceleration parameter. Among these failure mechanisms, only HCI has negative activation energy while others' are positive. This means lowering stress temperature will accelerate HCI while decelerating the other three failure mechanisms. HCI also has a comparable large γ, so at low temperature and reasonable high voltage, HCI failure will dominate. For EM, since the copper interconnect has a larger activation energy and small γ (≤ 2), acceleration tests should be designed with high temperature and low voltage. Traditional acceleration tests with high temperature and voltage can be applied to accelerate TDDB and NBTI since both have large voltage acceleration parameter and activation energy. The failure percentage of each failure mechanism at various accelerated conditions is shown in Fig. 3.

continued on page 6

Semiconductor Device Qualification with Multiple Failure Mechanisms

continued from page 5

Conclusions

For semiconductor devices, reliability modeling at the system level is complicated by the involvement of multiple failure mechanisms which have the same stress factors-voltage and temperature. The Arrhenius relationship with one activation energy for all temperature is shown to be not valid at the system level if these failure mechanisms don't have the same activation energy. The same happens to the modeling of voltage dependence. Using the exponential law with only one constant coefficient is a good option for an individual failure mechanism, but not for a system. We propose a failure mechanism-based qualification method which quantifies each failure mechanism through acceleration testing with specifically designed stress conditions.

References

[1] C. K. Hu and R. Rosenberg, "Scaling effect on electromigration in on-chip cu wiring," in IITC, pp. 267–269, 1999.

[2] C.-K. Hu, L. Gignac, and R. Rosenberg, "Electromigration of cu/low dielectric constant interconnects," Microelectronics and Reliability, vol. 46, no. 2-4, pp. 213–231, 2006.

[3] J. H. Stathis, "Reliability limits for the gate insulator in cmos technology," IBM Journal of Research and Development, vol. 46, no. 2/3, pp. 265–286, 2002.

[4] T. J. Anderson and J. M. Carulli Jr., "Modeling and monitoring of product DPPM with multiple fail modes," in IRPS, pp. 545–551, 2006.

[5] Y. H. Lee, N. Mielke, M. Agostinelli, S. Gupta, R. Lu, and W. McMahon, "Prediction of logic product failure due to thin-gate oxide breakdown," in IRPS, pp. 18–28, 2006.

[6] S. Ogawa and N. Shiono, "Generalized diffusion-reaction model for the low-field charge-buildup instability at the si-SiO2 interface," Physical Review B, vol. 51, no. 7, pp. 4218–4230, 1995.

[7] W. Bornstein, R. Dunn, and T. Spielberg, "Field degradation of memory components due to hot carriers," in IRPS, pp. 294–298, 2006.

[8] J. R. Black, "Mass transport of aluminum by momentum exchange with conducting electron," in Proc. Sixth Ann. Reliability Physics Symp, pp. 148–159, 1967.

[9] J. Srinivasan, S. V. Adve, P. Bose, and J. A. Rivers, "The case for lifetime reliability-aware microprocessor," in IEEE Proceedings of 31st Annual International Symposium on Computer Architecture, 2004.

[10] A. Dasgupta and R. Karri, "Electromigration reliability enhancement via bus activity distribution," in 33rd Design Automation Conference, (Las Vegas, NV, USA), 1996.

[11] E. Takeda and N. Suzuki, "An empirical model for device degradation due to hot-carrier injection," IEEE Electron Device Letters, vol. EDL-4, pp. 111–113, 1983.

[12] JEDEC, Failure Mechanisms and Models for Semiconductor Devices. JEDEC Solid State Technology Association, 2003.

[13] S. Chakravarthi, A. T. Krishnan, V. Reddy, C. F. Machala, and S. Krishnan, "A comprehensive framework for predictive modeling of negative bias temperature instability," in 42nd IRPS, pp. 273–282, IEEE, 2004.

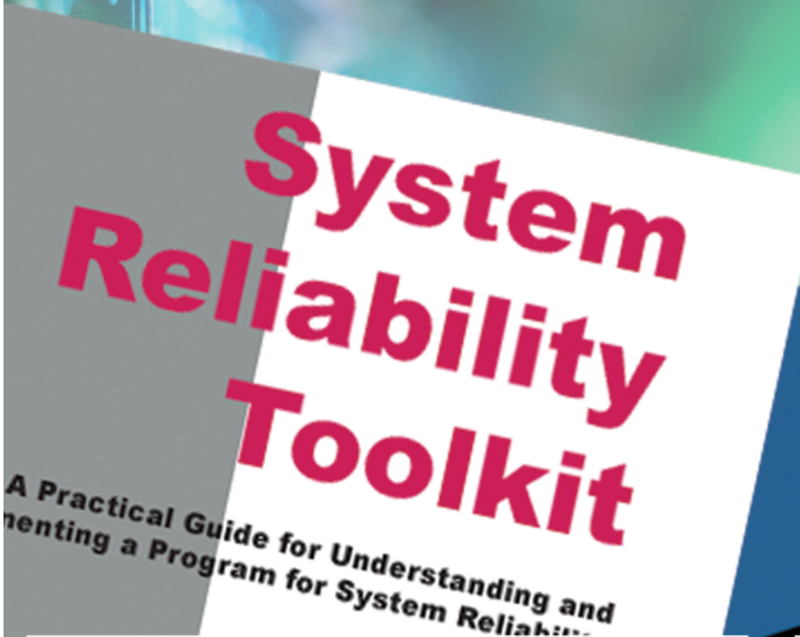
[14] P. Chaparala, J. Shibley, and P. Lim, "Threshold voltage drift in PMOSFETS due to NBTI and HCI," in IRW, pp. 95–97, IEEE, 2000.

[15] S. Mahapatra, P. B. Kumar, and M. A. Alam, "Investigation and modeling of interface and bulk trap generation during negative bias temperature instability of p-MOSFETs," IEEE Transactions on Electron Devices, vol. 51, no. 9, pp. 1371–1379, 2004.

[16] J.-C. Lin, S.-Y. Chen, H.-W. Chen, Z.-W. Jhou, H.-C. Lin, S. Chou, J. Ko, T.-F. Lei, and H.-S. Haung, "Investigation of dc hot-carrier degradation at elevated temperatures for n-channel metal-oxide-semiconductor field-effect-transistor of 0.13μm technology," Japanese Journal of Applied Physics,Part 1: Regular Papers and Short Notes and Review Papers, vol. 45, pp. 3144–3146, Apr 2006.

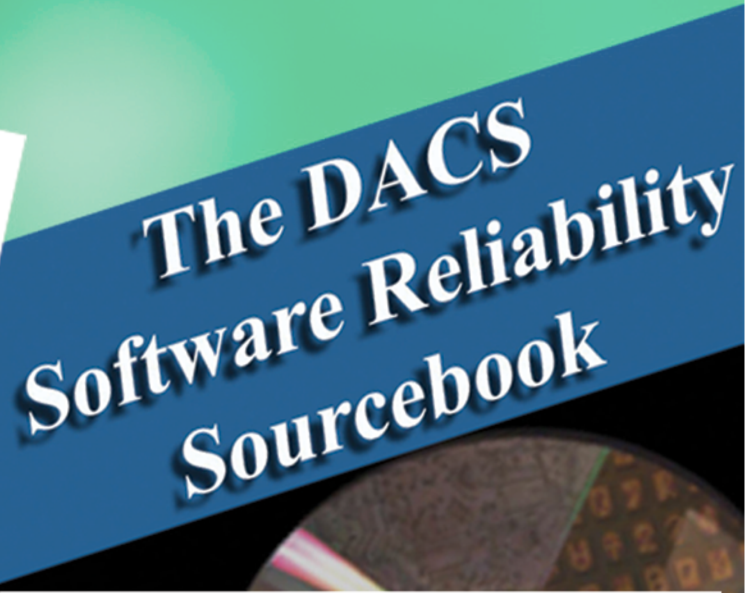
[17] E. Wu and J. Sune, "Power-law voltage acceleration: A key element for ultrathin gate oxide reliability," Microelectronics Reliability, 2005.

[18] D. K. Schroder and J. A. Babcock, "Negative bias temperature instability: Road to cross in deep submicron semiconductor manufacturing," Journal of Applied Physics, vol. 94, pp. 1–18, 2003.[4] T. J. Anderson and J. M. Carulli Jr., "Modeling and monitoring of product DPPM with multiple fail modes," in IRPS, pp. 545–551, 2006.



System Reliability Toolkit

A Practical Guide for Understanding and
...nenting a Program for System Reliab...



The DACS Software Reliability Sourcebook

Reducing Reliability Risks for

Kevin A. Kwiat, Ph.D., Air Force Research Laboratory (AFRL/IFGA)

For over a quarter of a century now – an eon in the course of development of modern computers – the tenet that hardware and software are logically equivalent has remained unchanged [1], [2]. This tenet has not only withstood change, it has become the basis for a vibrant approach to computer architecture: reconfigurable computing. Choosing whether or not to implement logic in hardware or software has always been a design decision based on where to migrate the complexity: to hardware or software. High speed usually means a hardware-intensive implementation whereas a way to reduce the size, weight and power (SWAP) of hardware is to rely more on a software-intensive implementation. The 1990's saw the distinction between hardware and software blurred by static-RAM (SRAM) based Field Programmable Gate Arrays (FPGAs): with every new data loaded into their on-chip configuration memories, these FPGAs realized a new hardware function. The contents of configuration memory became the software that changes into hardware. Now algorithms that previously, due to their complexity, could only be realistically considered for software implementations have a feasible path to a direct hardware implementation. Accepting of this migration from software to hardware are SRAM-based FPGAs. Those SRAM-based FPGAs that permit writing and rewriting portions of their configuration memory concurrent with device operation are called dynamically reconfigurable FPGAs. With the ability to change software into hardware on the granularity of individual logic gates, dynamically reconfigurable FPGAs go the furthest towards making hardware appear like software. Designers have manipulated the flexibility of dynamically reconfigurable FPGAs to construct reconfigurable computers [3] that offer enhanced performance while still remaining within reasonable SWAP measures. However, might the reliability of dynamically reconfigurable FPGAs, as opposed to their traditional, fixed-hardware counterparts, be their downfall?

In digital logic, the unit of measuring complexity is gates. Gates are physical entities that occupy die space, consume power and take up a chip's routing resources, so, in general, if the gate count is higher, then the reliability is lower. However, consider the notion of virtual gates: we see them but they are not there. These ***gates*** (note bold italics) are virtual in the same sense as computer virtual memory - where main memory is made to look larger than it physically is. When pages of main memory are not needed they are swapped out to disk and stored there until they are needed again. A dynamically reconfigurable FPGA's cells implement gates that are connected to form a logical function; yet when this logical function is no longer needed and the cells are being reused by another logical function, where do the gates of the previous function go? This is the key question in migrating a reconfigurable computer's complexity in a reliability-conscious way.

To answer this question, one can extrapolate from a widely accepted reliability-prediction method [4] that calculates the failure rate based on gate-count. You can then calculate the failure rate for the two cases: a conventional approach with fixed hardware logic where traditional gates are used, and the dynamically reconfigurable FPGA approach that uses gates.

For these two cases the failure rate, λ, is calculated by the formula:

$$\lambda = (C_1\pi_T + C_2\pi_E)\pi_Q\pi_L$$

where,

C_1 = die complexity failure rate
π_T = temperature factor
C_2 = package complexity failure rate
π_E = environmental factor
π_Q = quality factor
π_L = learning factor

In the preceding calculations, the following values are constant: $\pi_T = 0.16$, $\pi_E = 0.5$, $\pi_Q = 3$, and $\pi_L = 1$. Regarding the constant learning factor for both cases, one can use the dynamically reconfigurable FPGAs of Atmel's AT6000 series that the company advertises as mature technology (see http://www.atmel.com/products/FPGA/). Also, some early work was carried out with developing CAE tools that simulated the FPGA's dynamic reconfiguration [5]. This too contributed to a climbing of the learning curve in using ***gates*** instead of gates.

For the fixed hardware design, C_1 is measured by counting the gates. Since the on-chip SRAM of dynamically reconfigurable FPGAs is volatile, external, non-volatile memory is needed to store the FPGAs configuration data. The FPGA is considered programmed when the configuration data is read from the external memory and loaded into the FPGA's on-chip, SRAM configuration memory. Therefore, only when external storage (see Figure 1) is added can the FPGA implement usable gates. The C_1 for the unprogrammed FPGA is based solely on the FPGA's gate count. It is comprised of the FPGA's cells, on-chip configuration memory and internal programming logic.

Adding external memory to store the various FPGA configurations increases the C_1 factor of the reconfigurable design. However, the external memory permits us to migrate complexity from gates to gates. For external memory one can

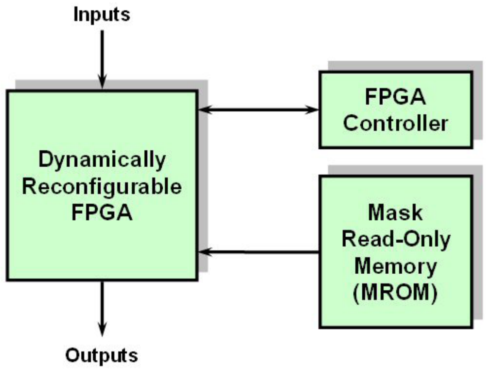


Figure 1. Basic Reconfigurable Computing System

Reconfigurable Computing

turn to a masked-programmed ROM (MROM). Typical densities of MROMs are 16 Megabits per chip. In the modeling of the AT6000, three bytes program a cell, and it is assumed that a single gate is implemented per cell. Increases in the gate count of the target design imply a 3-byte increase in the MROM on a per gate basis. For example, 1 Mbits of MROM can be used to implement 41,600 gates. The C_1 value for a MROM of this size is only 0.0052, while the C_1 value for the equivalent number of hardware gates is 0.29 – a factor of 55 increase in complexity. With 16 Megabits of MROM, the FPGA can implement 666,666 gates.

For the fixed hardware solution, consider two sub-cases. The first is a single package solution and the second is a two-package solution. The failure rate calculation for the reconfigurable approach includes three packages: the FPGA, the controller, and the MROM. The number of pins for the FPGA package is 224. The controller is assigned 1,000 gates and an initial package pin count of 36. Initially, the MROM of byte size 3 (for 1 gate) is assigned a 16-pin package. The number of address pins for the counter and the MROM are then increased with the size of the MROM needed to accommodate the gate count of the target design. Figure 2 shows the failure rates between a fixed hardware design and a reconfigurable design.

The horizontal axis is the number of logic gates required. For the fixed hardware case, this number is the same as counting the number of gates directly. For the FPGA, this is the number of gates implemented through dynamic reconfiguration where the unused gated are stored externally in the MROM. Initially, the failure rate for the unprogrammed FPGA is high due to its non-virtual gate count of 55,296. However, the complexity for each FPGA-implemented gate is placed into a significantly less complex 24-bit increment of MROM. As a result, the failure rate for the FPGA, counter and MROM grows only slightly as compared to the curves for the fixed hardware solutions. Comparing the failure rates of the reconfigurable and fixed designs, the sub-case of the single package fixed-gate solution has greater failure rates when gate counts go above 130,000. At higher gate counts, the two-package sub-case would probably occur, and as shown in Figure 2, its corresponding failure rate is dramatically greater than that of the reconfigurable design.

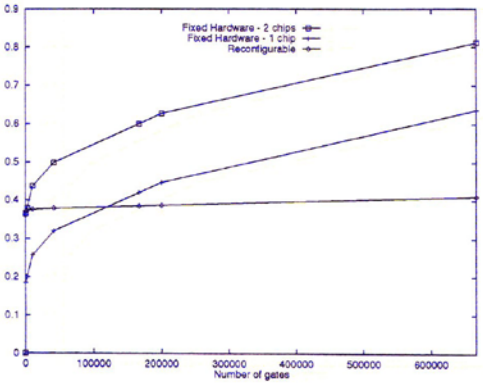


Figure 2. Number of Gates vs. Failure Rate

Fixed gates of a custom application-specific integrated circuit (ASIC) could implement computer algorithms directly in hardware; yet a large number of fixed gates would be necessary. Fixed gates may be used to implement a microprocessor that runs software versions of these algorithms; however, modern microprocessors easily exceed the gate counts shown in Figure 2. Furthermore, the failure rates shown in Figure 2 do not include the memory that is required by software-based implementations of computer algorithms. Typically, microprocessor execution of the software is possible only when cache chips and memory management units augment the processor's basic CPU functions – thus a multiple package solution would be expected. Driving up gate counts even more would be the need to match a reconfigured computer's ability to provide an increase in the variety of the algorithms.

Migrating computer algorithms from software to hardware and meeting SWAP requirements is not as daunting a complexity problem because of the maturing of dynamically reconfigurable FPGAs. The storing of virtual gates in mature MROM places a reconfigurable computer on a solid foundation. Solid, here, has dual meaning: in the context of reliability attributed to low device failure rate and in the context of permanence – once installed in the system, changing MROM entails the prohibitive expense and toil of actual chip replacement. Although this article has demonstrated that logic gate virtualization within MROM shows promise for managing a reconfigurable computer's hardware complexity, a question arises: will the software that designers migrate to hardware be reliable enough to have them commit it to unchangeable MROM? A reliability engineer will likely be sought out for the answer.

References

[1] Andrew Tanenbaum, Structured Computer Organization," 1st edition, Prentice Hall, 1976.

[2] Andrew Tanenbaum, Structured Computer Organization," 5th edition, Prentice Hall, 2006.

[3] Michael Barr, "A Reconfigurable Computing Primer," Multimedia Systems Design, September 1998.

[4] MIL-HDBK-217, "Reliability Prediction of Electronic Equipment," Revision F, Notice 2, February 1995.

[5] Kevin Kwiat and Warren Debany, "Reconfigurable Logic Modeling," Integrated Systems Design, December 1996.

Electronic Component Failure

Vito Faraci, Jr.

Many electronic component manufacturers predict failure rates (λ) of their devices by use of an in-house testing process. For example the test may involve placing a certain number of components in an oven, and allowing them to "bake" at a predetermined temperature for a predetermined amount of time. At the end of the test, the number of failed devices are counted to determine the values of r and "DeviceHours," where r = number of failures, and DeviceHours = number of surviving components times the number of hours they remained in the oven. Since the group of devices under test is typically just a small sample of a much larger population, quotient:

$$\frac{\text{r}}{\text{Device Hours}}$$

is considered an estimate of λ as opposed to the true λ. Since the manufacturer may have difficulty accumulating enough data to calculate true λ, he does the next best thing, by employs a Chi-Square statistical tool that relates the controlled test sample to the entire population. This tool allows him to calculate a "confidence interval" which provides a "statistical" upper bound for the true λ. The equation manufacturers are using [Equation 1] is:

$$\lambda = \frac{\chi^2_{(1-CL,\ 2r+2)}}{2 \text{ x Device Hours}}$$

where $\chi^2_{(1\text{-CL, } 2r+2)}$ is a statistical factor taken from the Chi-Square Table, with 2r+2 = degrees of freedom, and CL = confidence level.

Objective

This paper explains why and how Equation 1 works, and its derivation.

Background Theory

Calculations for confidence level are based on the binomial distribution function described in many statistics textbooks. The binomial distribution function is written as [Equation 2]:

$$P_n(k) = \binom{n}{k} p^{n-k} q^k \quad \text{where} \binom{n}{k} \text{is defined as } \frac{n!}{k!(n-k)!}$$

When we are interested in the probability that r or fewer events occur in n trials (or, conversely, that greater than r events occur), then the cumulative binomial distribution function (CDF) of Equation 2 is used as shown in Equations 3 or 4:

$$P(\varepsilon \le r) = \sum_{k=0}^{r} P_n(k) \quad = \sum_{k=0}^{r} \left(\frac{n!}{k!(n-k)!} \right) p^{n-k} q^k$$

$$P(\varepsilon > r) \ = 1 \ - \ P(\varepsilon \le r) \ = \sum_{k=r+1}^{n} \left(\frac{n!}{k!(n-k)!} \right) p^{n-k} q^k$$

In terms of the cumulative binomial distribution function, the confidence level is defined in either Equations 5 or 6 as:

$$CL = \ P(\varepsilon > r) = 1 \ - \sum_{k=0}^{r} \left(\frac{n!}{k!(n-k)!} \right) (1-q)^{n-k} q^k$$

$$1 \ - \ CL = \ \sum_{k=0}^{r} \left(\frac{n!}{k!(n-k)!} \right) (1-q)^{n-k} q^k$$

where CL is the confidence level in percent.

Now if n is very large, and q is very small, then the Poisson expression:

$$\frac{(nq)^k}{k!} e^{-nq}$$

provides a conservative estimate of the binomial distribution term for term as shown by Equation 7.

$$\left(\frac{n!}{k!(n-k)!} \right) (1-q)^{n-k} q^k \ \approx \ \frac{(nq)^k}{k!} e^{-nq}$$

Joining Equations 6 and 7 together we get Equation 8:

$$1 \ - \ CL \ = \sum_{k=0}^{r} \frac{(nq)^k}{k!} e^{-nq} \ = \ e^{-nq} \left[1 + nq + \cdots + \frac{(nq)^{r-1}}{(r-1)!} + \frac{(nq)^r}{r!} \right]$$

where nq is the expected or mean value.

In terms of Reliability, if λ = the constant failure rate of a component, and t = time in operation, then n can be replaced by t, and q replaced by λ, and λt will be the expected (mean) number of failures. This implies that:

$$P(r) \ = \ \frac{e^{-\lambda t} \cdot (\lambda t)^r}{r!}$$

equals the probability of exactly r failures in time interval 0 to t. Note that when writing each term out:

$$P(0) = e^{-\lambda t}, \ P(1) = e^{-\lambda t} \cdot \lambda t, \ P(2) = e^{-\lambda t} \frac{(\lambda t)^2}{2!}, \ P(3) = e^{-\lambda t} \frac{(\lambda t)^3}{3!}, \text{etc.}$$

it becomes apparent that [Equation 9]:

$$P(r) \ = \ e^{-\lambda t} \left[1 + \lambda t + \cdots + \frac{(\lambda t)^{r-1}}{(r-1)!} + \frac{(\lambda t)^r}{r!} \right]$$

In words, Equation 9 reads P(r) = probability (0 failures) or probability (1 failure) or ... or probability (r failures). Another way of saying this same thing is the probability of r or less failures in a time interval t. Replacing λt for nq in Equation 8 yields Equation 10:

$$1 \ - \ CL \ = \sum_{k=0}^{r} \frac{(\lambda t)^k}{k!} e^{-\lambda t} \ = \ e^{-\lambda t} \left[1 + \lambda t + \cdots + \frac{(\lambda t)^{r-1}}{(r-1)!} + \frac{(\lambda t)^r}{r!} \right]$$

It turns out that for a given CL, the term:

$$\chi^2_{(1\text{-}\chi^2,\ 2r+2)/2}$$

is the exact λt solution to Equation 10. So, for example, choosing CL = 0.6 or 1 - CL = 0.4, Table 1 shows exactly what the entries of the Chi-Square Table represent.

Table 1. Chi-Square Table

Entries from Chi-Square Table (Probability, Degrees of Freedom)		
Failures (r)	(1-CL, 2r+2)/2 = λt	Exact Solution to Poisson Distribution
0	(0.4,2)/2 = 0.916290731	$0.4 = e^{-\lambda t}$
1	(0.4,4)/2 = 2.022313245	$0.4 = e^{-\lambda t} * [1+ \lambda t]$
2	(0.4,6)/2 = 3.105378597	$0.4 = e^{-\lambda t} * [1+ \lambda t+(\lambda t)^2/2!]$
3	(0.4,8)/2 = 4.175262733	$0.4 = e^{-\lambda t} * [1+ \lambda t+(\lambda t^2/2!)+ (\lambda t^3/3!)]$

In other words, the Chi-Square confidence limits in Column 2 of Table 1 yield the exact solutions to

Rate Prediction Analysis

the Poisson distributions in Column 3. In other words, the Chi Square Table solves for λt. (Note that the Chi Square Table does not solve for λ or t alone, but λt.) Now simply divide both sides of the above equation by t (device hours) to get:

$$\lambda = \frac{\chi^2(1-\mathrm{CL}, 2r+2)}{2t}$$

which is exactly Equation 1.

Note that the above method is reserved for devices that exhibit constant failure rates, i.e., electronic devices. For devices that exhibit non-constant failure rates, i.e., mechanical devices, other statistical methods should be employed, which is a subject for another paper.

Notes on the Chi-Square Table

A user friendly Chi-Square Interactive Table can be accessed online at:

http://math.uc.edu/~brycw/classes/148/tables.htm

Both Microsoft Excel and MathCad have built-in Chi-Square Table generators. Any entry of the Chi-Square Table $\chi^2_{(1\text{-CL},\ 2r+2)}$ can be determined using either one of those two programs. For Excel, enter "Chiinv(1-CL, 2r+2)". For MathCad, enter "qchisq(CL,2r+2)".

More Interesting Facts on the Poisson Distribution

The collection of terms:

$$e^{-x},\ e^{-x}x,\ e^{-x}\frac{x^2}{2!},\ e^{-x}\frac{x^3}{3!},\ \cdots$$

is called the Poisson Distribution. Note that the sum of all the terms is equal to one:

$$\sum_{k=0}^{\infty} e^{-x}\frac{x^k}{k!} = e^{-x}\left[1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots\right] = e^{-x}\cdot e^{x} = 1$$

Therefore this Poisson Distribution is a discrete probability density function or PDF. The cumulative distribution function CDF is:

$$\sum_{k=0}^{r} e^{-x}\frac{x^k}{k!} = e^{-x}\left[1 + x + \cdots + \frac{x^{r-1}}{(r-1)!} + \frac{x^r}{r!}\right]$$

For r = 0, 1, 2, 3, etc. The Poisson CDF is also a discrete distribution.

Poisson Approximation to the Binomial

Theorem [For Equation 7]:

If "n" is large and "q" is small. then:

Proof:

$$\frac{n!}{k!(n-k)!}p^{n-k}q^k \approx \frac{(nq)^k}{k!}e^{-nq}$$

Proof:

$$\frac{n!}{k!(n-k)!}p^{n-k}q^k = \frac{n(n-1)(n-2)\cdots(n-k+1)}{k!}(1-q)^{n-k}q^k \quad \text{since } p = 1-q$$

$$\approx \frac{n^k}{k!}(1-q)^{n-k}q^k \quad \text{since n is large}$$

$$= \frac{n^k}{k!}\cdot\frac{(1-q)^n}{(1-q)^k}\cdot q^k \approx \frac{n^k}{k!}\cdot\frac{(1-q)^n}{1}\cdot q^k = \frac{(nq)^k(1-q)^n}{k!} \quad \text{since q is small} \quad (1)$$

now compare $(1-q)^n$ with e^{-nq} by expanding each of them.

$$(1-q)^n = 1 - nq + \frac{n(n-1)}{2!}q^2 - \frac{n(n-1)(n-2)}{3!}q^3 + \cdots$$

$$\approx 1 - nq + \frac{n^2}{2!}q^2 - \frac{n^3}{3!}q^3 + \cdots = 1 - nq + \frac{(nq)^2}{2!} - \frac{(nq)^3}{3!} + \cdots \quad \text{since n is large}$$

$$\therefore\ (1-q)^n \approx 1 - nq + \frac{(nq)^2}{2!} - \frac{(nq)^3}{3!} + \cdots \quad (2)$$

$$e^{-nq} = 1 - nq + \frac{(nq)^2}{2!} - \frac{(nq)^3}{3!} + \cdots \quad (3)$$

$$\text{comparing (2) and (3)} \Rightarrow (1-q)^n \approx e^{-nq} \quad (4)$$

Replacing e^{-nq} for $(1-q)^n$ in (1) we get $\frac{n!}{k!(n-k)!}p^{n-k}q^k \approx \frac{(nq)^k}{k!}e^{-nq}$ //

Proving Equation 7 using L'Hospital's Rule
Theorem 1:

$$\lim_{x\to 0} 1-e^{-x} = x \quad \text{or} \quad \lim_{x\to 0} 1-x = e^{-x}$$

By using L'Hospital's Rule we see the relationship between x and 1-e^{-x} as x gets very small.

$$\lim_{x\to 0}\frac{f(x)}{g(x)} = \frac{1-e^{-x}}{x} = \frac{0}{0} \quad \text{and} \quad \lim_{x\to 0}\frac{f'(x)}{g'(x)} = \frac{e^{-x}}{1} = 1 \Rightarrow \lim_{x\to 0}\frac{1-e^{-x}}{x} = 1 \Rightarrow$$

$$\lim_{x\to 0} 1-e^{-x} = x \quad \text{or} \quad \lim_{x\to 0} 1-x = e^{-x}$$

continued on page 12

Electronic Component Failure Rate Prediction Analysis

continued from page 11

Theorem 2:

If n is very large, and q is very small, then:

$$P_n(k) = \left(\frac{n!}{k!(n-k)!}\right)(1-q)^{n-k}q^k \xrightarrow[n\to\infty]{} \frac{(nq)^k}{k!}e^{-nq}$$

Proof:

$$P_n(0) = (1-q)^n$$
$$P_n(1) = n(1-q)^{n-1}q$$
$$P_n(2) = \frac{n(n-1)(1-q)^{n-2}q^2}{2!}$$
$$P_n(3) = \frac{n(n-1)(n-2)(1-q)^{n-3}q^3}{3!}\text{, etc. by definition.}$$

Since n is very large we can replace n–k with n and get:

$$P_n(0) = (1-q)^n$$
$$P_n(1) = n(1-q)^n q$$
$$P_n(2) \cong \frac{n^2(1-q)^n q^2}{2!}$$
$$P_n(3) \cong \frac{n^3(1-q)^n q^3}{3!}\text{, etc.}$$

Since q is very small we can replace (1–q) with by L'Hospital's Rule. Therefore:

$$P_n(0) \cong (e^{-q})^n = e^{-nq}$$
$$P_n(1) \cong n(e^{-q})^n q = nqe^{-nq}$$
$$P_n(2) \cong \frac{n^2(e^{-q})^n q^2}{2!} = \frac{(nq)^2 e^{-nq}}{2!}$$
$$P_n(3) \cong \frac{n^3(e^{-q})^n q^3}{3!} = \frac{(nq)^3 e^{-nq}}{3!}\text{, etc. //}$$

About the Author

Mr. Faraci is a mathematician by education, an electrical engineer by trade, and a computer programmer by hobby. He has taught math at the New York Institute of Technology (NYIT) as an adjunct professor and has presented seminars to the FAA on the subjects of Probability, Reliability, Markov Analysis, Fault Tree Analysis, and Failure Modes and Effects Analysis. In addition, Mr. Faraci has given several lectures around the US and Canada on calculating probability of failure of electrical and mechanical equipment. He has published several articles on these topics for the Reliability Information Analysis Center (formerly the Reliability Analysis Center) and the Journal of System Safety.

The Journal of the Reliability Information Analysis Center: Fourth Quarter – 2006

The Journal of the Reliability Information Analysis Center is published quarterly by the Reliability Information Analysis Center (RIAC). The RIAC is a DoD Information Analysis Center (IAC) sponsored by the Defense Technical Information Center (DTIC) and operated by a team led by Wyle Laboratories, and including Quanterion Solutions, the Center for Risk and Reliability at the University of Maryland, the Penn State University Applied Research Lab (ARL) and the State University of New York Institute of Technology (SUNYIT).

The Reliability Information Analysis Center
6000 Flanagan Road
Suite 3
Utica, NY 13502-1348
Toll Free: 877.363.RIAC (7422)
Phone: 315.351.4200
Fax: 315.351.4209
inquiry@theRIAC.org
http://theRIAC.org

The appearance of paid advertising in the RIAC Journal does not constitute endorsement by the Department of Defense or the Reliability Information Analysis Center of the products or services advertised.

RIAC Journal Editor
David Nicholls
Toll Free: 877.363.RIAC (7422)
Phone: 315.351.4202
Fax: 315.351.4209
dnicholls@theRIAC.org

Key RIAC Contacts

Richard Hyle
RIAC COR, Air Force Research Laboratory
Phone: 315.330.4857
Fax: 315.330.7647
richard.hyle@rl.af.mil

Joseph Hazeltine
RIAC Director, Technical Area Task (TAT) Manager
Phone: 256.716.4390
Fax: 256.721.0144
joseph.hazeltine@wylelabs.com

Preston MacDiarmid
RIAC Deputy Director
Toll Free: 877.808.0097
Phone: 315.732.0097
Fax: 315.732.3261
pmacdiarmid@quanterion.com

David Nicholls
RIAC Product Manager
Toll Free: 877.363.RIAC (7422)
Phone: 315.351.4202
Fax: 315.351.4209
dnicholls@theRIAC.org

Patricia Smalley
RIAC Training Coordinator
Toll Free: 877.363.RIAC (7422)
Phone: 315.351.4200
Fax: 315.351.4209
pat.smalley@wylelabs.com

David Mahar
RIAC Webmaster, RIAC Software/Database Manager
Toll Free: 877.808.0097
Phone: 315.732.0097
Fax: 315.732.3261
dmahar@theRIAC.org

Register to receive your free subscription to the RIAC Journal:

Patricia Smalley
Toll Free: 877.363.RIAC (7422)
Phone: 315.351.4200
Fax: 315.351.4209
Email: pat.smalley@wylelabs.com

or go to the RIAC website:
http://quanterion.com/RIAC/HelpDesk/ContactRIAC.asp

Reducing Program Risk Through Independent Testing for 57 Years

Wyle Laboratories, Inc. has provided trusted agent test and evaluation services for more than 57 years. Throughout that period, Wyle has provided quality data that has been key to reducing program risk, resulting in increased system effectiveness for the warfighter. From component testing in the early development phases to independent test engineering services supporting the operational test phase, through ongoing life cycle evaluation and support, Wyle's exceptional services have been unparalleled across the test continuum.

➔ TEST CONTINUUM ➔

Advanced Concept Technology Demonstration	Advanced Technology Demonstration	Operational Assessment	Contractor Test and Evaluation	Developmental Test and Evaluation	Initial Operational Test and Evaluation	Live Fire Test and Evaluation	Operational Assessment	Initial Operational Test and Evaluation	Follow-on Operation Test and Evaluation	Joint Test and Evaluation

Through its dedication to provide high level engineering expertise at all stages of the testing process, Wyle today significantly improves the operational performance, effectiveness, and suitability of sea, air, land and space systems and platforms. With capability, capacity and commitment, Wyle reduces program risk, getting the very best systems fielded for the warfighter.

wyle laboratories

www.wylelabs.com e-mail: service@wylelabs.com

PAID ADVERTISEMENT

20 years in the making.
Relex Reliability Studio 2006

Relex Software is proud to announce the highly anticipated release of Relex Reliability Studio 2006. From the company with a history of innovation and a list of impressive firsts, Relex Reliability Studio does it once again: redefines reliability engineering tools with a package of unprecedented power and flexibility (and a little pizzazz!).

See it. Experience it. View the trailer at www.relexthemovie.com

tudio 2006

Want to see the best in action? With inventive features you hadn't even imagined before, our all new Relex Reliability Studio 2006 includes floating, dockable, and tabbed control windows, the quick configure Relex Bar, the indispensable Project Navigator, and a completely customizable desktop.

Want global collaboration? The Relex Enterprise Edition supports both Oracle and SQL Server in a scalable, robust solution with enterprise capabilities such as permission-based roles, customizable workflow, alert notifications, audit tracking, and the new Relex *i*Architect module supporting a web browser interface.

Relex Reliability Studio 2006

Fault Tree/Event Tree
FMEA/FMECA
FRACAS Corrective Action
Human Factors Risk Analysis
Life Cycle Cost
Maintainability Prediction
Markov Analysis
Optimization and Simulation
Reliability Block Diagram
Reliability Prediction
Weibull Analysis

Ready to learn more? Check out the Relex trailer at www.relexthemovie.com and sign up at www.relex.com/news/seminars.asp to attend a preview showing at a location near you. Or let us personally introduce you to Relex Reliability Studio 2006 – call 724.836.8800 today!

Relex

excellence in reliability

The Information Assurance Maryland: An Overview of

Michel Cukier and Carol S. Smidts, Center for Risk and Reliability Engineering,

University of Maryland's Information Assurance Laboratory

The Information Assurance Laboratory at the University of Maryland is co-led by Dr. Michel Cukier and Dr. Carol S. Smidts. The laboratory advises, on average, 15 graduate students, 10 undergraduate students and one postdoctoral student. The laboratory focuses on four research threads: (1) Probabilistic Risk Assessment, (2) Modeling, (3) Bug and Vulnerability Identification Tools, and (4) Empirical Studies. National Science Foundation (NSF), National Aeronautics and Space Administration (NASA), Nuclear Regulatory Commission (NRC), National Security Agency (NSA), Defense Advanced Research Projects Agency (DARPA), Teradyne, Texas Instruments Inc., Tedco, and Maryland Industrial Partnerships (MIPS), among others, have funded these research activities. Figure 1 provides an overview of the laboratory. In the next sections we describe, in more detail, some of the research activities included in the four research threads.

Probabilistic Risk Assessment Research Activities

Probabilistic Risk Assessment (PRA) is a technique used to assess the probability of failure or success of a large technological system such as a chemical plant, nuclear power plant or an assembly such as the Space Station or the Space Shuttle. Results provided by the risk assessment methodology are used to make decisions concerning choice of upgrades, scheduling of maintenance, decision to launch, start-up, shut-down, decision to abort in flight, and other key parameters.

The PRA methodology accounts for hardware and to some extent for human interventions but does not account for software contributions to risk. The consequence of this statement is that the estimated level of risk is inaccurate and probably significantly lower than it should. This might not have been an issue 30 years ago but given the increased dependence of current technological systems on software (and even the possibility of total reliance on autonomous systems which will learn new conditions and the particular responses to these conditions on the fly), the problem has become significant.

Our work has focused on the development of an approach to "integrate" software considerations into the PRA framework (both classical and dynamic). The approach proceeds along the classical PRA process of identification of initiators, definition of accident scenarios, decomposition of events in the accident scenario into basic events using fault trees and quantification using given reliability models and modifies it by adding possible software initiators, software events, and software basic events, by defining four major categories of software-related failures that need quantification (input failures, output failures, failures of the software itself and failures of the software due to the computer platform) and corresponding methods of identification and quantification.

For the dynamic PRA, a modeling scheme has been defined comprised of extensions of hierarchical finite state machines.

Modeling Research Activities

In this section, we describe our modeling activities.

The Functional Architecture Model

From a structural perspective, software reliability models can be classified into "monolithic" or "architectural" depending whether they treat software as a black box or as a grey/white box (i.e. considering the inner workings of the software). The "functional architecture" is an architectural model centered around the concept of functions. The architecture is derived directly from the software requirements specification by identification of high level functions and progressive refinement of functions into lower level functions. In addition, the architecture reflects the non-functional requirements found in the software requirements specification through the concept of attributes. Through a well defined taxonomy of failures modes, potential modes of failure are systematically assigned to lower level functions, higher level functions, and attributes. Quantification is based on Bayesian statistics and allows early prediction of reliability provided that reasonable priors based on similar software development processes can be used. As a consequence, early identification of weak points in the design is possible as well as reorientation of resources towards these areas or consideration of potential design alternatives. The approach is such that it challenges the requirements directly, as requirements are examined carefully in the process of building the architecture and assigning failure modes.

From Measures to Reliability

The objectives of this research have been to bridge

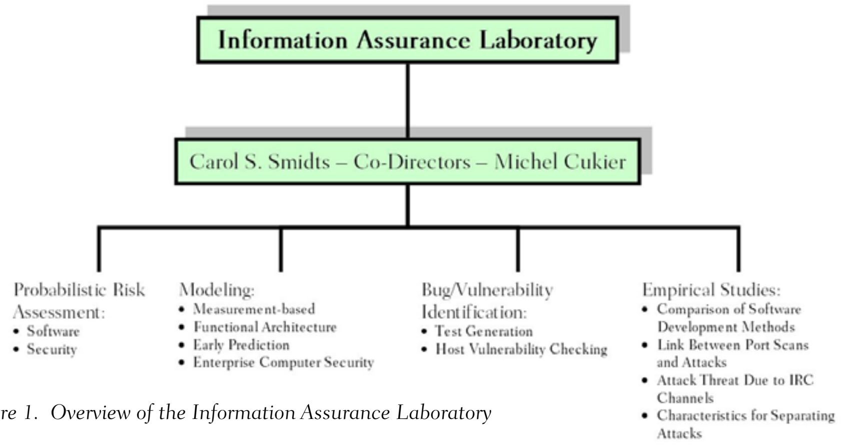

Information Assurance Laboratory
Carol S. Smidts – Co-Directors – Michel Cukier
Probabilistic Risk Assessment:
• Software
• Security
Modeling:
• Measurement-based
• Functional Architecture
• Early Prediction
• Enterprise Computer Security
Bug/Vulnerability Identification:
• Test Generation
• Host Vulnerability Checking
Empirical Studies:
• Comparison of Software Development Methods
• Link Between Port Scans and Attacks
• Attack Threat Due to IRC Channels
• Characteristics for Separating Attacks

Figure 1. Overview of the Information Assurance Laboratory

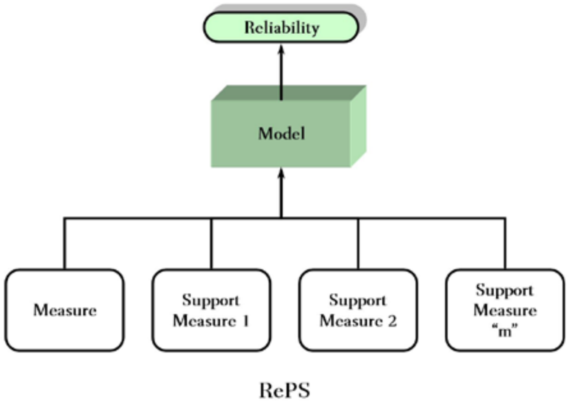


Figure 2. RePS[1]

[1] Figure 2 extracted from C. Smidts, "Research in Software Reliability Engineering", RAMS, Newport Beach, CA, 23-26 January 2006

Laboratory at the University of Research Activities

University of Maryland

the gap between software reliability and software engineering measures. Software engineering measures are dedicated to measurement of diverse software characteristics such as software logic complexity, number and type of defects present in the requirements, design or code, traceability of requirements to code, functional size of the code, etc. The number of characteristics one can measure and consequently the number of potential candidate software engineering measures is infinite or so it seems. Existence of a stable, robust and proven relationship between software reliability and particular software engineering measures allows substitution of the process of observing software failures for that of measuring one or multiple potentially more accessible software characteristics. The primary concept on which this research is based is the concept of Reliability Prediction System (RePS), i.e. a complete set of measures from which software reliability can be predicted. A RePS (See Figure 2) is typically built around a main measure called a root measure. Support measures are then identified which connect the root measure to reliability. A model then connects the measures to reliability. To date, twelve RePSs have been established. The quality of predictions yielded has been assessed against reliability testing and actual filed data on three digital control systems including one safety critical application. The methodology is now entering a review stage. Given positive reviews, the methodology may be used to supplement or replace the current reviewing process used at the NRC for licensing digital safety critical nuclear power plant applications.

A System Dynamics Modeling and Simulation of Enterprise Computer Security

The goal of this research is to explore a novel approach for analyzing and supporting increase of organizational security through modeling and simulation. This will lead to understanding security risk reduction in computer systems, diagnosing such systems and identifying their weaknesses, as well as prospectively examining the effectiveness of different solutions before their implementation. The resulting simulation models will support decisions of various types related to security management and others (e.g., financial/economics, risk, technology, human resources), and analyze the impact of decisions before they are implemented. We use a holistic and systemic approach to organizational security, including the human element in the model, thus facilitating understanding of (and explaining) the observed behavior of security aspects of an organizational ecosystem consisting of machines and people. We will test the extent of feasibility and the benefits of using system dynamics and generic (archetypal) structures for modeling individual and organizational behavior in support of security strategy decision-making.

Bug and Vulnerability Identification Tools Research Activity

In this section, we describe research on automated test generation and host vulnerability checking activities.

Automated Test Generation

Test automation is a discipline of software engineering that aims at reducing test time and number of errors made during testing. Test automation is comprised of two major activities, automating the test generation and test execution processes. While methods and tools for automating the test execution process are now mature (see for instance the widespread use of test execution tools such as Winrunner and LoadRunner from Mercury Interactive), the automation of the test generation process is still an open and vibrant field of research. Model-based testing is an automated black box[2] test generation process that starts from documents such as requirements specifications and design documents to establish a model of the application to be tested from which test cases are then derived. As such the quality of the tests derived from the model depends heavily from the correctness and completeness of these documents. Unfortunately, it is a known fact that requirements documents are typically incomplete, i.e. underspecified. This is due to the fact that users tend to overlook requirements that they may consider as trivial or presume to be known while software analysts aren't aware of the existence of such requirements since they themselves may not be intimately familiar with the application domain. Our research has targeted the development of a model-based testing technique called HOTTest (see Figure 3), which limits the occurrence of these errors and makes testing more effective. HOTTest is an acronym for Higher Ordered Typed Specification based testing. It uses a higher-ordered domain specific language to specify the software. Domain Specific Languages are languages dedicated to a particular application domain. The language Haskell-DB for instance is dedicated to specification of database applications. Since the domain is constrained, the language constructs are limited. These languages are thus easier to learn than typical formal languages. HOTTest develops test cases automatically from this specification. In addition, domain-specific axioms have been defined which are automatically called upon during generation of the test cases. The new test cases generated in this fashion correspond to an extension of the specification which allows coverage of the implicit (unspecified) domain specific requirements.

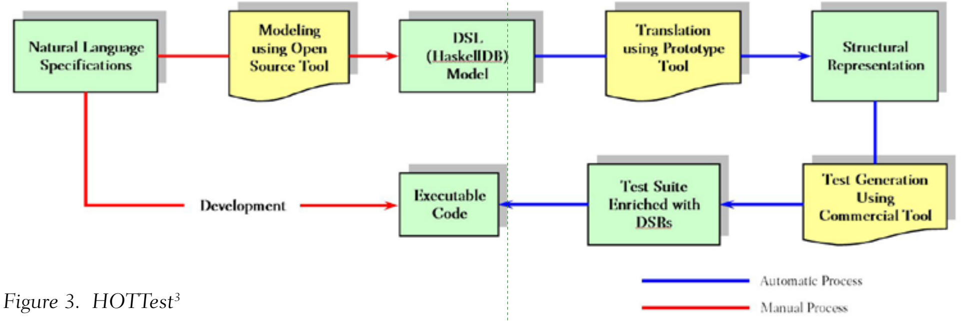

Natural Language Specifications
Modeling using Open Source Tool
DSL (HaskellDB) Model
Translation using Prototype Tool
Structural Representation
Test Generation Using Commercial Tool
Test Suite Enriched with DSRs
Executable Code
Development
Automatic Process
Manual Process

Figure 3. *HOTTest*[3]

HOTTest has been used in the context of database applications and has been extended to other application domains such as Web-based applications and Graphical User Interface (GUI) applications. Usability and feasibility studies have been performed in classroom settings as well as on an industrial application and demonstrate that for the domain of database applications, HOTTest is more effective in capturing domain properties than most other commonly used model-based test design techniques. HOTTest is also more efficient and can provide higher requirements coverage.

continued on page 18

[2] *Black box testing validates the fact that a software component meets its requirements while being impervious to the internal workings of the component being tested.*

[3] *Figure 3 is extracted from A. Sinha, C. Smidts, A. Moran, "Enhanced Testing of Domain Specific Applictions by Automatic Extraction of Axioms from Functional Specifications", Proceedings of the 14th IEEE International Symposium of Software Reliability Engineering, Denver, Colorado, Nov 2003*

The Information Assurance Laboratory at the University of Maryland: An Overview of Research Activities

continued from page 17

Host Vulnerability Checking Tool: Ferret

Evaluation of computing system security requires knowledge of the vulnerabilities present in the system and of potential attacks against the system. Vulnerabilities can be classified based on their location as application vulnerabilities, network vulnerabilities, or host vulnerabilities. We developed Ferret-Unix and Ferret-Windows, two new software tools for checking host vulnerabilities on the Unix and Windows platforms. These tools help system administrators by quickly finding vulnerabilities that are present on a host. They are designed and implemented in a modular way: a plug-in module is used for each vulnerability checked, and each possible output format is specified by a plug-in module. As a result, both tools are extensible, and can easily be kept up-to-date through the addition of checks for new vulnerabilities as they are identified. Finally, both tools are freely available open-source software.

Empirical Studies

In this section, we describe three empirical studies: (1) an analysis of scans and their impact on attacks, (2) an assessment of the attack threat due to Internet Relay Chat (IRC) channels, and (3) an evaluation of attack characteristics for separating attacks.

An Analysis of Scans and Their Impact on Attacks

The primary goal of this research thread is to more precisely characterize port scans, Internet Control Message Protocol (ICMP) scans, vulnerability scans and to determine their impact on attacks. Such an analysis provides a first step in answering questions like: Can scans be used as a good indicator of an attack? What is the probability distribution of the scans? What is the time between a scan and an attack? We analyzed the link between scans and attacks based on empirical data. We then characterized the three scan distributions and the time distribution separating a scan from an attack. Such information is important for the security community to assess the validity that scans are often precursors to an attack and to more accurately predict attacks based the scans observed. A test-bed using target computers for monitoring attackers and collecting attack data was used to analyze the data collected. Two experiments were conducted to determine the relevance of classifying scans and attacks based on the number of packets per connection. The correlation between scans and attacks were studied by focusing on the scans and identifying if attacks were associated with them and then analyzing the attacks and identifying the ones that were linked to a scan. For the scans preceding attacks, the distribution of the three types of scans was characterized as well as the time distribution between the last scan preceding the attack and the attack.

Assessing the Attack Threat Due to IRC Channels

This research thread focuses on assessing the attack threat of different environments. We have investigated the threat of attacks associated with the chat medium Internet Relay Chat (IRC). A combination of simulated users (i.e., bots), some configured with scripts that simulated conversations, and regular users were used. The average number of attacks per day a user on IRC can expect, the effect of channel activity, gender based on the name, and network type on the number of attacks were determined. The social structure of IRC channels and the types of users that use it were analyzed. The only type of attack that occurs consistently daily is malicious private messages, and in and of themselves they pose no threat to computer security. This threat does not seem to depend on whether or not a user is active in a channel. Users with female names are, however, far more likely to receive malicious private messages, slightly more likely to receive files and links, and equally likely to be attacked in other ways. This implies that the attacks are carried out by humans selecting targets rather than automated scripts sending attacks to everyone in the channel. Users with ambiguous names are far less likely to receive malicious private messages than female users, but more likely to receive them than male users. Users in channels that do not allow bots at all are more likely to receive attacks than users in channels that allow a minimal number of bots.

An Evaluation of Attack Characteristics for Separating Attacks

This research thread focuses on finding attack characteristics that efficiently classify attacks. The first step of the research consists of evaluating the efficiency of Transmission Control Protocol (TCP) connection characteristics for separating network attacks into families. The goal of this study is to know what are the best statistical characteristics in a TCP connection to classify different families of attack that are targeting a single TCP port. The dataset analyzed was collected during 117 days using a test-bed of two high interaction honeypots, which are real computers used for the sole purpose of being attacked. Nine characteristics of the network connection were analyzed. Some of them, like the number of packets or the number of bytes, come directly from the TCP connections and others are statistical values computed from the previous ones. The methodology to measure the efficiency of the characteristics consisted first of defining families of attacks based on the payloads collected, and then to run a clustering algorithm on these characteristics. The output of the clustering algorithm was compared to the families of attacks in order to score the efficiency of all the characteristics and combinations of characteristics to separate attacks. The results show that 1) the number of bytes is a remarkable feature to separate unsuccessful from successful attacks within malicious traffic; 2) time-based characteristics are poor feature to separate attacks into families; 3) building combination of characteristics does not improve the efficiency of the automated attack separation.

Contact Information

Dr. Michel Cukier
Assistant Professor
Address: 0151E Martin Hall
University of Maryland
College Park, Maryland 20742
Email: mcukier@umd.edu
Phone: 301-314-2804
Fax: 301-314-9601

Dr. Carol Smidts
Associate Professor
Address: 2129 Martin Hall
University of Maryland
College Park, Maryland 20742
Email: csmidts@umd.edu
Phone: 301-405-7314
Fax: 301-314-9601

An Overview of the 217Plus™

David Nicholls, Reliability Information Analysis Center

In the Third Quarter 2006 edition of the RIAC Journal, we introduced the "Handbook of 217Plus™ Reliability Prediction Models" that the RIAC has published to provide insight into the methodology and models that make up the 217Plus™ approach to system reliability assessment [Reference 1]. We briefly introduced the primary factors that form the basis of the methodology:

1. Whether information exists on a predecessor system
2. The amount of empirical reliability data that is available for that system
3. Whether the reliability analyst chooses to assess the processes used in system development

Figure 1 provides an overview of the 217Plus™ approach to failure rate estimation that is based on the above three factors. Note that, for the purposes of our discussion, "system" applies to the highest level definition of the item defined within 217Plus™. A "system", therefore, can be a true system, a product, an equipment, an assembly, a subassembly, i.e., any level of complexity that the user wishes to define.

If a system to be analyzed using 217Plus™ is an evolution of a predecessor system (i.e., an earlier, but similar, configuration to the new design), then a prediction can be performed on both the predecessor system and the new system. The results of these two predicted system failure rates form the basis of a ratio that can be used to modify the observed failure rate of the predecessor system. The result of this predecessor analysis is λ_1 in Figure 1.

If enough empirical data (field, test or both) is available on the new system to be analyzed, it can be combined with the 217Plus™ predicted failure rate of the new system using a Bayesian approach to form the "best" failure rate estimate possible. As the quantity of empirical data increases, the failure rate using the Bayesian combination will be increasingly dominated by the empirical data. The result of this Bayesian combination is presented as λ_2 in Figure 1.

The minimum amount of analysis required for a 217Plus™ reliability prediction is the summation of component estimated failure rates, plus other data that may be available to the analyst. The current twelve component models used by 217Plus™ are included in the Handbook, and will be introduced in more detail in future editions of the RIAC Journal. The result of the component-based prediction is represented by $\lambda_{IA,\ new}$ in Figure 1. This predicted value can be further modified within 217Plus™ through the application of the optional Process Grade Analysis, or other modifications to default environmental stress or operational profiles. These modifications are reflected in the failure rate represented by $\lambda_{predicted,\ new}$ in Figure 1.

The rest of this article will discuss each element of the 217Plus™ methodology presented in Figure 1 in more detail.

Note that the 217Plus™ methodology calculates failure rates in terms of failures per million calendar hours, not operating hours. Therefore, user inputs for field data or user-defined failure rates need to be converted to a calendar hour basis prior to incorporating them into a 217Plus™ reliability prediction. The conversion factors are:

Calendar hours = Operating hours / Duty cycle

Operating hours = Calendar hours x Duty cycle

$\lambda_{IA,\ predecessor}$

$\lambda_{IA,predecessor}$ represents the initial failure rate assessment of the predecessor system. This is the sum of the predicted component failure rates,

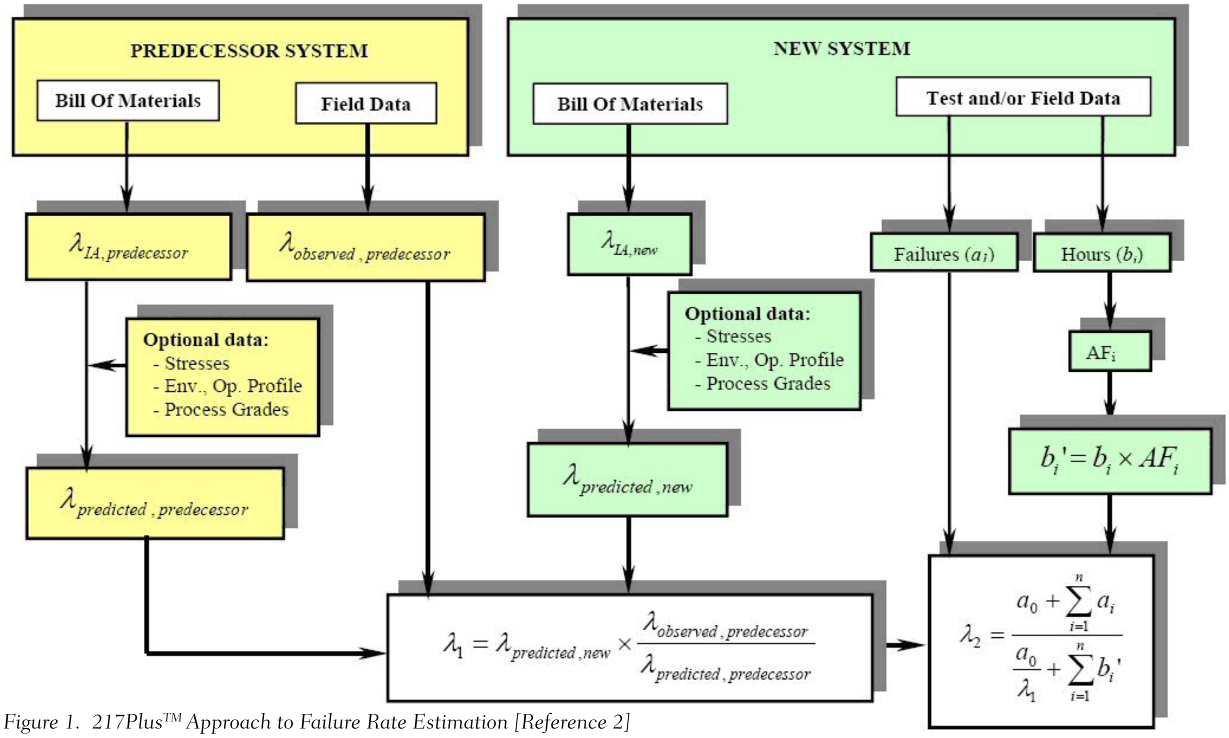

PREDECESSOR SYSTEM
Bill Of Materials
Field Data
$\lambda_{IA,predecessor}$
$\lambda_{observed,predecessor}$
Optional data:
- Stresses
- Env., Op. Profile
- Process Grades
$\lambda_{predicted,predecessor}$
NEW SYSTEM
Bill Of Materials
Test and/or Field Data
$\lambda_{IA,new}$
Failures (a_i)
Hours (b_i)
Optional data:
- Stresses
- Env., Op. Profile
- Process Grades
AF$_i$
$b_i' = b_i \times AF_i$
$\lambda_{predicted,new}$
$\lambda_1 = \lambda_{predicted,new} \times \frac{\lambda_{observed,predecessor}}{\lambda_{predicted,predecessor}}$
$\lambda_2 = \frac{a_0 + \sum_{i=1}^{n} a_i}{\frac{a_0}{\lambda_1} + \sum_{i=1}^{n} b_i'}$

Figure 1. 217Plus™ Approach to Failure Rate Estimation [Reference 2]

System Reliability Assessment Methodology

and uses the twelve 217Plus™ component failure rate models, data from the RIAC Nonelectronic (NPRD) and Electronic (EPRD) Part Databases, or user-defined data on components from other sources.

$\lambda_{observed,\ predecessor}$

$\lambda_{observed,\ predecessor}$ is the observed failure rate of the predecessor system, and represents the point estimate of the failure rate, which is equal to the number of observed failures divided by the cumulative number of operating hours.

Optional Data

Optional data is used to enhance the predicted failure rate by factoring in more detailed information pertaining to environmental stresses, operating profile factors, and Process Grades. 217Plus™ contains default values for the environmental stresses and operational profile, but in the event that actual values of these parameters are known, either through analysis or measurement, they should be used instead of the defaults. The application of Process Grades within 217Plus™ is also optional, allowing the user the option of evaluating the specific processes used in the development and sustainment of a system. If the process grades are not used, default values are provided for each process (failure cause), so that the user can evaluate any or all of the processes. The use of the Process Grade option of 217Plus™ is included in the Handbook, and will be addressed in more detail in a future edition of the RIAC Journal.

$\lambda_{predicted,\ predecessor}$

$\lambda_{predicted,\ predecessor}$ is the predicted failure rate of the predecessor system after combining the initial assessment ($\lambda_{IA,predecessor}$) with the Optional Data, if used.

$\lambda_{IA,new}$

$\lambda_{IA,new}$ represents the initial assessment of the new system. This is calculated as the sum of the predicted component failure rates, and uses the 217Plus™ component failure rate models, data from the RIAC NPRD and EPRD databases, and other data that may be available to the analyst. A reliability prediction performed in accordance with this method is the minimum level of analysis that will result in a predicted reliability value. Applying the Optional Data can further refine this value.

$\lambda_{predicted,\ new}$

$\lambda_{predicted,\ new}$ is the predicted failure rate of the new system after combining the initial assessment with the Optional Data, if used. If the Optional Data is not used, then $\lambda_{predicted,\ new}$ is equal to $\lambda_{IA,new}$.

λ_1

λ_1 is the failure rate estimate of the new system after the predicted failure rate of the new system ($\lambda_{predicted,\ new}$) is combined with the predicted and observed information from the predecessor system ($\lambda_{predicted,\ predecessor}$ and $\lambda_{observed,\ predecessor}$, respectively). The equation that translates the failure rate of the predecessor system to the new system is:

$$\lambda_1 = \lambda_{predicted,new} \times \frac{\lambda_{observed,predecessor}}{\lambda_{predicted,predecessor}}$$

The values of $\lambda_{predicted,new}$ and $\lambda_{predicted,predecessor}$ are obtained using the component reliability prediction procedures, equations and data previously described. The ratio "$\lambda_{observed,predecessor}$ /$\lambda_{predicted,predecessor}$" accounts for the differences in the predicted and observed failure rates of the predecessor system. This ratio inherently accounts for the differences in the systems that are accounted for in the component reliability prediction methodology.

This methodology can be used when the new system is an evolutionary extension of predecessor designs. If similar processes are used to design and manufacture a new system, and the same reliability prediction processes and data are used, then there is every reason to believe that the predicted/observed ratio of the new system will be similar to that observed on the predecessor system.

This methodology implicitly assumes that there is enough operating time and failures on which to base a value of $\lambda_{observed,predecessor}$. For this purpose, the observance of failures is critical to derive a point estimate of the failure rate (i.e., failures divided by hours). A single-sided confidence level estimate of the failure rate should not be used.

a_i

a_i represents the number of failures for the i^th set of data on the new system.

b_i

b_i is the cumulative number of operating hours for the ith set of data on the new system.

AF_i

AF_i is the acceleration factor between the conditions of test or field data on a new system and the conditions under which the predicted failure rate is desired. If the data is from field applications in the same environment for which the prediction is desired, the AF value will be one. If the data is from accelerated test data or from field data in a different environment, then the AF value needs to be determined. If the applied stresses are higher than the anticipated field use environment of the new system, AF will be a value greater than one. The acceleration factor can be determined by performing a reliability prediction at both the test and use conditions, but AF can only be determined in this manner if the reliability prediction model is capable of discerning the effects of the accelerating stress(es) of the test. As an example, consider a life test in which a system was exposed to a temperature higher than what it would be exposed to in field-deployed conditions. In this case, the AF can be calculated as follows:

$$AF = \frac{\lambda_{T1}}{\lambda_{T2}}$$

where,

λ_{T1} = the predicted failure rate at the test conditions obtained by performing a prediction of the system at the test conditions

λ_{T2} = the predicted failure rate at the use conditions obtained by performing a prediction of the system at the use conditions

continued on page 22

An Overview of the 217Plus™ System Reliability Assessment Methodology

continued from page 21

b_i'

b_i' is the effective cumulative number of hours of the test or field data used. If the tests were performed at accelerated conditions, the equivalent number of hours needs to be converted to the conditions of interest, as follows:

$$b_i' = b_i \times AF_i$$

a_o

a_o is the effective number of failures associated with the predicted failure rate. If unknown, use 0.5. In the event that predicted and observed data is available on enough predecessor systems, this value can be tailored. This tailoring method will be discussed shortly.

λ_2

λ_2 is the best estimate of the new system failure rate after using all available data and information. As much empirical data as possible should be used in the assessment. This is done by mathematically combining λ_1 with empirical data. Bayesian techniques are used for this purpose. This technique accounts for the quantity of data by weighting large amounts of data more heavily than small amounts. λ_1 forms the "prior" distribution, comprised of a_0 and a_o/λ_1. If empirical data (i.e., test or field data) is available for the system under analysis, it is combined with λ_1 based on the following equation:

$$\lambda_2 = \frac{a_0 + \sum_{i=1}^{n} a_i}{\frac{a_0}{\lambda_1} + \sum_{i=1}^{n} b_i'}$$

λ_2 is the best estimate of the failure rate, and a_o is the "equivalent" number of failures of the "prior" distribution corresponding to the reliability prediction. For these calculations, 0.5 should be used unless a tailored value can be derived. An example of this tailoring is provided in the next section. a_o/λ_1 is the equivalent number of hours associated with λ_1, and a_1 through a_n are the number of failures experienced in each source of empirical data. There may be "n" different sources of data available (for example, each of the "n" sources corresponds to individual tests or field data from the population of systems). b_1' through b_n' is the equivalent number of cumulative operating hours experienced for each individual data source. These values must be converted to equivalent hours by accounting for any accelerating effects between the use conditions.

Tailoring the Bayesian Constant, a_o, in λ_2

This section discusses tailoring of the a_o value used in the Bayesian equations. The value of a_o is proportional to the degree of weighting given to the predicted value (λ_1). The constant a_0 is chosen such that the uncertainty in the failure rate estimate, as calculated with the Chi-square distribution, equates to the observed uncertainty. The default value of 0.5 to be used in the equation is based on the observed/predicted ratio from a wide variety of systems, applications, industries, etc. As such, there are many "noise factors' contributing to the variability in this ratio. However, if the user of the 217Plus™ methodology has enough data on which to derive a tailored value of a_0, it should be derived and used. While the default value of 0.5 represents the large degree of uncertainty inherent when a diverse data set is used, a typical 217Plus™ user will generally be analyzing systems with a much more narrow focus, in terms of system type, environment, operating profile, etc. As such, with enough data, the value of a_0 can be increased.

To estimate the value of a_o that should be used, a distribution of the following metric is calculated for all systems for which both predicted and observed data is available:

$$\frac{\lambda_{observed,predecessor}}{\lambda_{predicted,predecessor}}$$

The lognormal distribution will generally fit this metric well, but others (i.e., Weibull) can also be used. The cumulative value of this distribution is then plotted. Next, the failure rate multipliers, as determined from the Chi-square distribution, are calculated and plotted. This Chi-square distribution should be determined and plotted for various numbers of failures to ensure that the distribution of observed/predicted failure rate ratios falls between the Chi-square values. In most cases, one, two and three failures should be sufficient. Next, the plots are compared to determine which Chi-square distribution most closely matches the observed uncertainty values. The number of failures associated with that distribution then becomes the value of a_0. Figure 2 illustrates an example for which this analysis was performed.

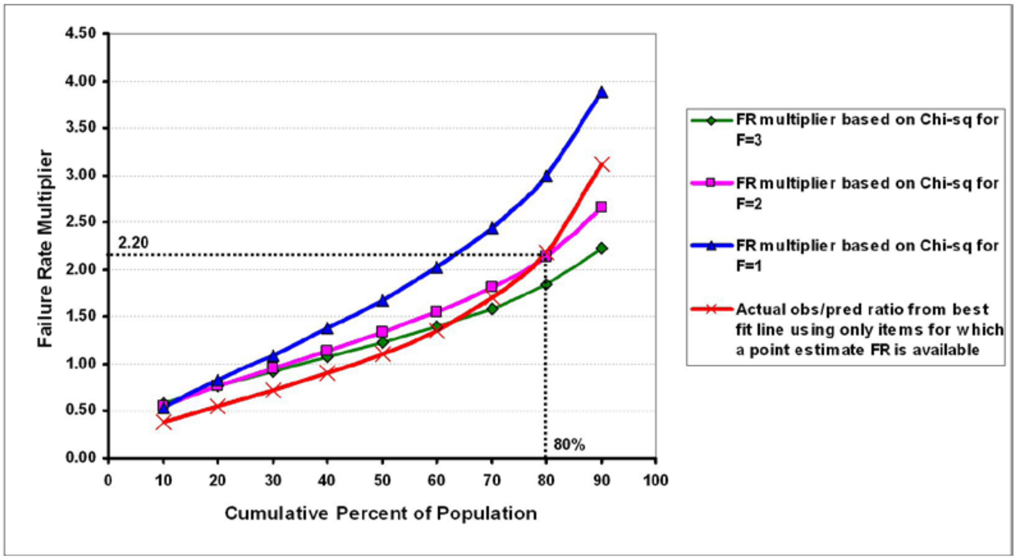


Figure 2. Comparison of Observed Uncertainty with the Uncertainty Calculated Using the Chi-Square Distribution

As can be seen from Figure 2, the observed uncertainty does not precisely match the Chi-square calculated uncertainty for any of the one, two or three failures used in this analysis. This is likely due to the fact that the population of systems on which this analysis is based is not homogeneous, as assumed by the Chi-square calculation. However, the confidence levels of interest are generally in the range of 60 to 90 percent. In this range, the Chi-square calculated uncertainty with 2 failures most closely approximates the observed uncertainty. Therefore, in this example, an a_0 value of 2 was used.

The uncertainties represented by the distribution of observed/predicted failure rates are typical of what can be expected when historical data on predecessor systems are collected and analyzed to improve the reliability prediction process. For example, using this example, one can be 80% certain that the actual failure rate for a system or product will be less than 2.2 times the predicted value.

Next Issue

The next edition of the RIAC Journal (1st Quarter 2007) will present an introduction to the 217Plus™ component failure rate models.

References

1. "The RIAC 'Handbook of 217Plus™ Reliability Prediction Models'", Journal of the Reliability Information Analysis Center, Third Quarter 2006, available for PDF download from the RIAC at http://theRIAC.org

2. Denson, W.K., "Handbook of 217Plus™ Reliability Prediction Models", Reliability Information Analysis Center (RIAC), 26 May 2006, ISBN 1-933904-02-X

RIAC

RIAC JOURNAL SURVEY

FOURTH QUARTER – 2006

Please fax completed survey to 315.351.4209 – ATTN: Journal Editor

Journal Format □ Hard Copy □ Web Download

How satisfied as you with the **content** (article technicaly quality) in *this issue* of the Journal?

□ Very Satisfied □ Satisfied □ Neutral □ Dissatisfied □ Very Dissatisfied

How satisfied as you with the **apperance** (layout, readability) of *this issue* of the Journal?

□ Very Satisfied □ Satisfied □ Neutral □ Dissatisfied □ Very Dissatisfied

How satisfied as you with the **overall quality** of *this issue* of the Journal (compared to similar magazines, newsletters, etc.)?

□ Very Satisfied □ Satisfied □ Neutral □ Dissatisfied □ Very Dissatisfied

How did you become **aware** of *this issue* of the Journal?

□ Subscribe □ Colleague □ Library □ RIAC Website □ RIAC Email □ Conference/Trade Show

Would you **recommend** the RIAC Journal to a colleague?

□ Definitely □ Probably □ Not Sure □ Probably Not □ Definitely Not

Please suggest general **changes / improvements** to the RIAC Journal that would improve your level of satisfaction:

Please suggest further **topics** for the RIAC Journal that might help it to better meet your needs:

Overall satisfaction

□ Very Satisfied □ Satisfied □ Neutral □ Dissatisfied □ Very Dissatisfied

CONTACT INFORMATION (OPTIONAL)

Name Position/Title

Organization Office Symbol

Address City State Zip

Country Email

Phone Fax

My Organization Is □ Army □ Navy □ Air Force □ Other DoD/Government □ Industry □ Academic □ Other

I need help with a reliability, maintainability, quality, supportability, or interoperability problem. □ Please contact me

Future Events In RMQSI

Annual Reliability and Maintainability Symposium (RAMS)

Orlando, FL Jan 22-25, 2007

Contact Information:
David Barber
Scien-Tech Associates, Inc
Ph: 828.898.6375
Email: DBARBSTA@aol.com

ASQ 2007 Six Sigma Conference

Phoenix, AZ February 12-13, 2007

Contact Information:
American Society for Quality (ASQ)
600 North Plankinton Avenue
Milwaukee, WI 53203
Ph: 800.248.1946
Email: help@asq.org

23rd National Logistics Conference & Exhibition

Miami, FL March 19-22, 2007

Contact Information:
Meredith Geary
Meeting Planner
Ph: 703.247.9476
Email: mgeary@ndia.org

Military Technologies Conference

Boston, MA Mar 27-28, 2007

Contact Information:
Nuala Ferdinand
Ph: 910.221.9410
Email: mtcconference@pennwell.com

RCM-2007 The Reliability Centered Maintenance Managers' Forum

Honolulu, HI Apr 3-6, 2007

Contact Information:
RELIABILITY Magazine
Ph: 888.575.1245 x111
Fax: 309.423.7234
Email: customerservice@reliabilityweb.com
web: ReliabilityWeb.com

2007 IEEE International Reliability Physics Symposium

Phoenix, AZ Apr 15-19, 2007

Contact Information:
Ronald Lacoe
The Aerospace Corporation
Ph: 310.336.0118
Email: Ronald.C.Lacoe@aero.org

Aging Aircraft 2007

Palm Springs, CA Apr 16-19, 2007

Contact Information:
J. Jennewine
Universal Technology Corporation
Ph: 937.426.2808
Fax: 937.426.8755
Email: jjennewine@utcdayton.com

AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference

Waikiki, HI Apr 23-26, 2007

Contact Information:
AIAA Headquarters
Reston, VA 20191-4344
Phone: 703.264.7500 or 800.639.AIAA
Fax: 703.264.7551

2007 Joint Service Power Expo

San Diego, CA Apr 23-26, 2007

Contact Information:
Angie DeKleine
Ph: 703.247.2599
Email: adekleine@ndia.org

ESTECH 2007 - IEST 53rd Annual Technical Meeting

Bloomingdale, IL April 29–May 2, 2007

Contact Information:
Institute of Environmental Sciences and Technology (IEST)
Ph: 847.255.1561
Email: iest@iest.org

ASQ World Conference on Quality and Improvement

Orlando, FL April 30-May 2, 2007

Contact Information:
American Society for Quality (ASQ)
600 North Plankinton Avenue
Milwaukee, WI 53203
Ph: 800.248.1946
Email: help@asq.org

JITC 17th Annual Interoperability Conference

Nashville, TN April 30-May 3, 2007

Contact Information:
Joint Interoperability Test Command
Conference Team
Ph: 520.538.5429
Email: interopconference@disa.mil.

IIE Annual Conference and Exposition 2007

Nashville, TN May 19-23, 2007

Contact Information:
Institute of Industrial Engineers
Ph: 800.494.0460 or 770.449.0460
Email: cs@iienet.org

2007 Systems & Software Technology Conference (SSTC)

Tampa, FL Jun 18-21, 2007

Contact Information:
Systems & Software Technology Conference
Ph: 800.538.2663 or 435.797.0423
Email: stc-info@ext.usu.edu
Web: www.sstc-online.org

International Applied Reliability Symposium

San Diego, CA Jun 20-22, 2007

Contact Information:
ReliaSoft Corporation
Ph: 520.886.0410
Email: Info@ARSymposium.org

RIAC TRAINING

RELIABILITY INFORMATION ANALYSIS CENTER

CHOOSE FROM

RELIABILITY 101

Seymour Morris, Quanterion Solutions

MECHANICAL DESIGN RELIABILITY

Timothy Bair, Col, USAF (ret), Penn State ARL

Dr. Stewart Kurtz, Professor Emeritus at Penn State University

PROBABILISTIC RISK ASSESSMENT & MANAGEMENT

Mohammadreza Azarkhail, University of Maryland

$1,195.⁰⁰ PER ATTENDEE

*Discounts apply to multiple registrations from an organization. Please contact the RIAC for details.

FEBRUARY 6-8, 2007

SAN DIEGO, CA

Hosted at the San Diego Training and Conference Center, San Diego, CA

FOR HOTEL INFORMATION & TO REGISTER VISIT http://theRIAC.org OR CALL 877.363.7422

- **Quantitative Risk Assessment**
- **Reliability**
- **Availability**
- **Maintainability**
- **Safety**

Engineered Risk Solutions
Improve with Confidence and Accuracy

Software Tools

Consulting

Training

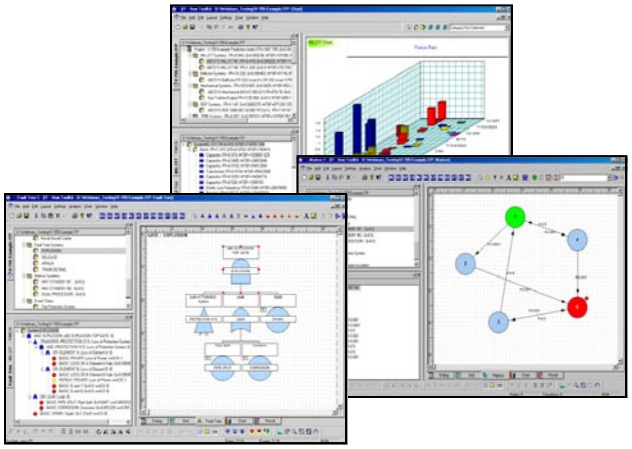
Support

Our pledge to you is Guaranteed Satisfaction.

Focusing on your needs and success is our #1 mission. Providing real and practical approaches to individuals and organizations worldwide, has been our tradition for over two decades.

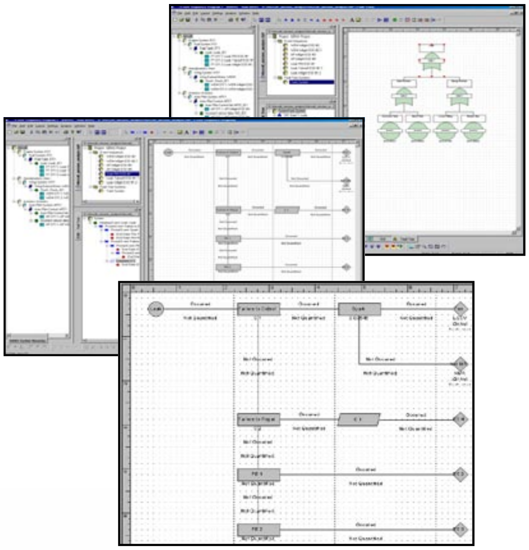
Count on *Item Software* to always support you and your team.

Item ToolKit



- Reliability Prediction
  - Mil-HDBK-217
  - Bellcore/Telcordia
  - NSWC
  - RDF (IEC 62380)
  - China 299B
- Fault Tree Analysis
- Event Tree Analysis
- Markov Analysis
- Reliability Block Diagram
- Failure Mode Effect Analysis (FMEA – ISO 9000)
- Failure Mode Effect & Criticality Analysis (FMECA – Mil-STD-1629)
- Failure Mode Effect & Diagnostics Analysis (FMEDA – IEC 61508)
- Maintainability Analysis
- SpareCost Analysis

Item QRAS



- Quantitative Risk Assessment
- Risk scenario modeling
- Event Sequence Diagram
- Fault Tree Analysis
- System level risk aggregation
- Quantified Risk Levels
- "What if" sensitivity analysis
- Binary Decision Diagram

item Software

Contact us for a product evaluation CD or personalized web demonstration

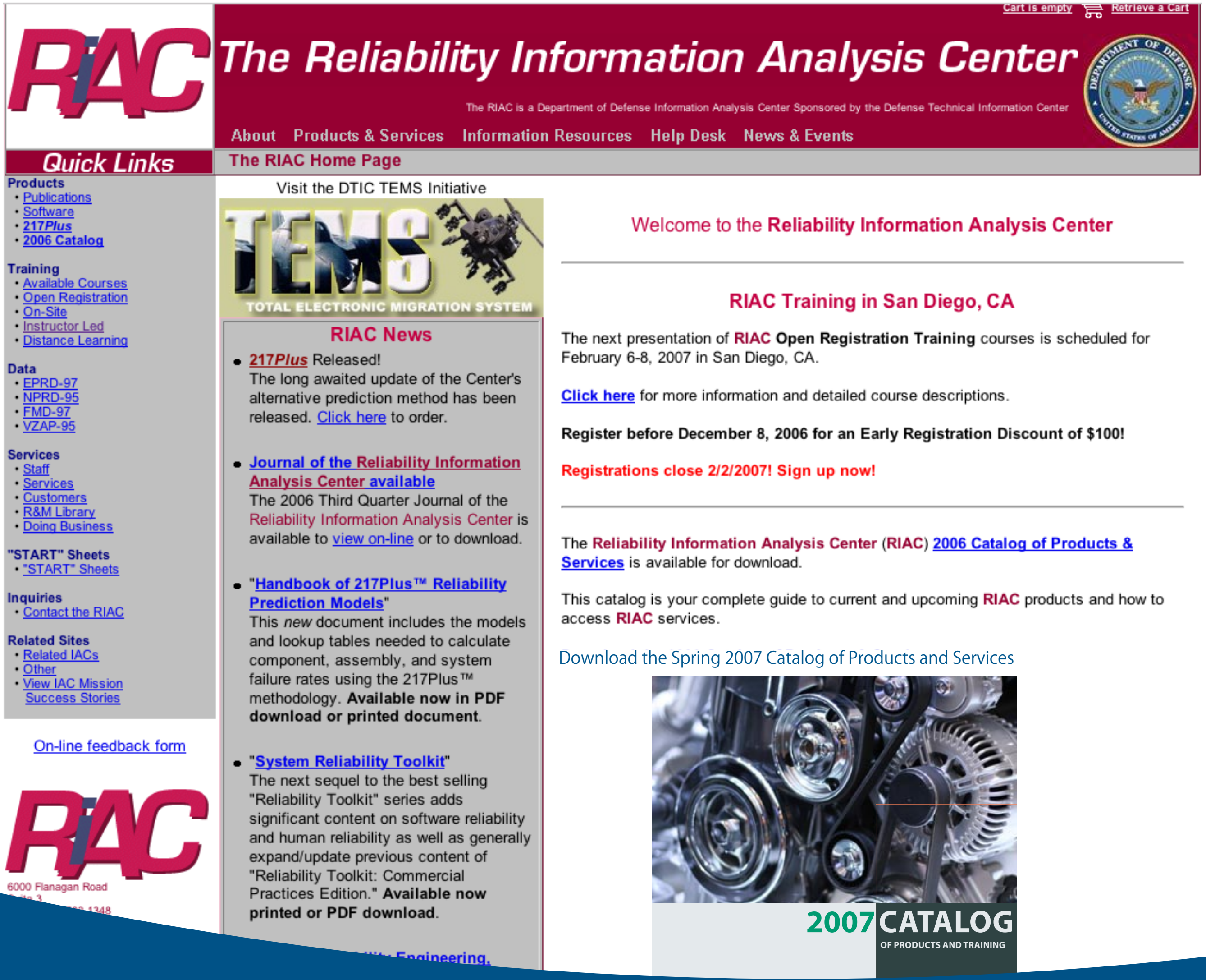
USA East, USA West and UK Regional Locations
USA: Tel: 714-935-2900 ● Fax: 714-935-2911 ● itemusa@itemsoft.com
UK: Tel: +44 (0) 1489 885058 ●Fax: +44 (0) 1489 885065 ●sales@itemuk.com

www.itemsoft.com

THE RIAC ONLINE

http://theRIAC.org

Technical Answers • RMQSI Library • The RIAC Journal • What's New at RIAC • Online Product Store • Upcoming Training Courses



Cart is empty Retrieve a Cart

RIAC The Reliability Information Analysis Center

The RIAC is a Department of Defense Information Analysis Center Sponsored by the Defense Technical Information Center

About Products & Services Information Resources Help Desk News & Events

Quick Links

The RIAC Home Page

Products
- Publications
- Software
- *217Plus*
- 2006 Catalog

Training
- Available Courses
- Open Registration
- On-Site
- Instructor Led
- Distance Learning

Data
- EPRD-97
- NPRD-95
- FMD-97
- VZAP-95

Services
- Staff
- Services
- Customers
- R&M Library
- Doing Business

"START" Sheets
- "START" Sheets

Inquiries
- Contact the RIAC

Related Sites
- Related IACs
- Other
- View IAC Mission Success Stories

On-line feedback form

RIAC
6000 Flanagan Road

Visit the DTIC TEMS Initiative

TEMS TOTAL ELECTRONIC MIGRATION SYSTEM

RIAC News

- *217Plus* Released!
The long awaited update of the Center's alternative prediction method has been released. Click here to order.

- Journal of the Reliability Information Analysis Center available
The 2006 Third Quarter Journal of the Reliability Information Analysis Center is available to view on-line or to download.

- "Handbook of 217Plus™ Reliability Prediction Models"
This *new* document includes the models and lookup tables needed to calculate component, assembly, and system failure rates using the 217Plus™ methodology. **Available now in PDF download or printed document**.

- "System Reliability Toolkit"
The next sequel to the best selling "Reliability Toolkit" series adds significant content on software reliability and human reliability as well as generally expand/update previous content of "Reliability Toolkit: Commercial Practices Edition." **Available now printed or PDF download**.

Engineering,

Welcome to the **Reliability Information Analysis Center**

RIAC Training in San Diego, CA

The next presentation of **RIAC Open Registration Training** courses is scheduled for February 6-8, 2007 in San Diego, CA.

Click here for more information and detailed course descriptions.

Register before December 8, 2006 for an Early Registration Discount of $100!

Registrations close 2/2/2007! Sign up now!

The **Reliability Information Analysis Center** (RIAC) **2006 Catalog of Products & Services** is available for download.

This catalog is your complete guide to current and upcoming **RIAC** products and how to access **RIAC** services.

Download the Spring 2007 Catalog of Products and Services

2007 CATALOG
OF PRODUCTS AND TRAINING

Reliability Information Analysis Center
6000 Flanagan Rd.
Suite 3
Utica, NY 13502-1348

PRSRT STD
U.S. Postage Paid
Atlanta, GA
Permit #7926