




UNITED STATES
NUCLEAR REGULATORY COMMISSION
ADVISORY COMMITTEE ON REACTOR SAFEGUARDS
WASHINGTON, DC 20555 - 0001

September 7, 2006

MEMORANDUM TO: ACRS Members

FROM: Eric A. Thornsbury, ACRS Senior Staff Engineer 

SUBJECT: CERTIFICATION OF THE MINUTES OF THE MEETING OF THE
ACRS SUBCOMMITTEE ON DIGITAL INSTRUMENTATION &
CONTROL SYSTEMS, JUNE 27, 2006 - ROCKVILLE,
MARYLAND

The minutes of the subject meeting, issued July 6, 2006, have been certified as the official record of the proceedings of that meeting. A copy of the certified minutes is attached.

Attachment: As stated

electronic cc: J. Larkins
M. Snodderly
S. Duraiswamy



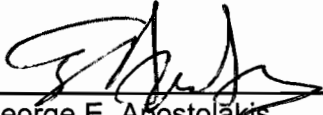
UNITED STATES
NUCLEAR REGULATORY COMMISSION
ADVISORY COMMITTEE ON REACTOR SAFEGUARDS
WASHINGTON, DC 20555 - 0001

MEMORANDUM TO: Eric A. Thornsby, ACRS Senior Staff Engineer

FROM: George E. Apostolakis, Chairman
Digital Instrumentation & Control Systems Subcommittee

SUBJECT: CERTIFICATION OF THE MINUTES OF THE MEETING OF THE
ACRS SUBCOMMITTEE ON DIGITAL INSTRUMENTATION &
CONTROL SYSTEMS, JUNE 27, 2006 - ROCKVILLE,
MARYLAND

I do hereby certify that, to the best of my knowledge and belief, the minutes of the subject meeting on June 27, 2006, are an accurate record of the proceedings for that meeting.


George E. Apostolakis
Subcommittee Chairman

9/7/06
Date

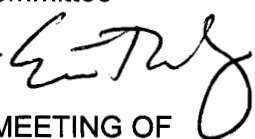


UNITED STATES
NUCLEAR REGULATORY COMMISSION
ADVISORY COMMITTEE ON REACTOR SAFEGUARDS
WASHINGTON, DC 20555 - 0001

PRE-DECISIONAL

July 6, 2006

MEMORANDUM TO: George E. Apostolakis, Chairman
Digital Instrumentation & Control Systems Subcommittee

FROM: Eric A. Thornsby, ACRS Senior Staff Engineer 

SUBJECT: WORKING COPY OF THE MINUTES OF THE MEETING OF
THE ACRS SUBCOMMITTEE ON DIGITAL INSTRUMENTATION
& CONTROL SYSTEMS, JUNE 27, 2006 - ROCKVILLE,
MARYLAND

A working copy of the minutes for the subject meeting is attached for your review. Please review and comment on them. If you are satisfied with these minutes, please sign, date, and return the attached certification letter.

Attachment: Minutes (DRAFT)

cc: Digital Instrumentation & Control Systems Subcommittee Members
J. Larkins
M. Snodderly
S. Duraiswamy
C. Santos

ADVISORY COMMITTEE ON REACTOR SAFEGUARDS
MEETING OF THE ACRS SUBCOMMITTEE ON
DIGITAL INSTRUMENTATION & CONTROL SYSTEMS
MEETING MINUTES - JUNE 27, 2006
ROCKVILLE, MARYLAND

INTRODUCTION

The ACRS Subcommittee on Digital Instrumentation & Control Systems held a meeting on June 27, 2006, in Room T-2B3, 11545 Rockville Pike, Rockville, MD. The purpose of this meeting was to review the ongoing digital system risk program and the development of a regulatory guide on risk-informed digital system reviews. The meeting was open to public attendance. Eric Thornsby was the Designated Federal Official for this meeting. There were no written comments from the public. Representatives from industry requested time to make an oral statement, which they presented at the end of the meeting. The Subcommittee Chairman convened the meeting at 8:30 a.m. on June 27, 2006 and adjourned it at 5:15 p.m..

ATTENDEES

ACRS

G. Apostolakis, Subcommittee Chairman
M. Bonaca, Member
T. Kress, Member

J. Bickel, Consultant
E. Thornsby, Designated Federal Official

Principal NRC Speakers

W. Kemper, RES
T. Aldemir, OSU
T. Chu, BNL

S. Arndt, RES
T. Hilsmeier, RES
G. Martinez-Guridi, BNL

Other Principal Speakers

A. Marion, NEI

Other members of the public attended this meeting. A complete list of attendees is in the ACRS Office File and is available upon request. The presentation slides and handouts used during the meeting are attached to the office copy of these minutes.

OPENING REMARKS BY CHAIRMAN APOSTOLAKIS

George Apostolakis, Chairman of the ACRS Subcommittee on Digital Instrumentation & Control Systems, convened the meeting at 8:30 a.m. Dr. Apostolakis stated that the purpose of this meeting was to review the ongoing digital system risk program and the development of a regulatory guide on risk-informed digital system reviews. He said the subcommittee would

gather information, analyze relevant issues and facts, and formulate proposed positions and actions, as appropriate, for deliberation by the full Committee. The rules for participation in the meeting were announced as part of the notice of the meeting published in the Federal Register on May 25, 2006. Dr. Apostolakis acknowledged that the subcommittee did not receive any written comments from the public, and that representatives from industry requested time to make an oral statement, which occurred at the end of the meeting.

DISCUSSION OF AGENDA ITEMS

Overview of Digital System Risk Research Program

Mr. William Kemper, Chief of the Instrumentation and Electrical Engineering Branch in the Office of Nuclear Regulatory Research, introduced the presentations and the speakers for the day. He briefly reminded the subcommittee of the previous interactions the staff has had with the subcommittee regarding several areas of the digital system research program. He then passed the presentation to Mr. Steven Arndt.

Mr. Arndt provided an overview of the digital system risk portion of the overall research program, which includes assessing which modeling methods that might be usable, determining which systems need to be modeled and at what level of detail, developing and testing modeling methods, and developing regulatory acceptance criteria. He also linked the work to issues facing the NRC as licensees replace older analog systems with digital systems. The industry has expressed interest in using risk-informed techniques as an alternative method for licensing the upgrades, but the agency's current state of knowledge does not currently support such an approach. This meeting served as a progress-reporting meeting to followup on the subcommittee's request for interactions during the course of the research.

Mr. Arndt described the details of the digital system risk program, which is investigating new methods for integrating current digital system models into probabilistic risk assessment (PRA). The staff is pursuing both traditional and dynamic methods and plans to perform a benchmark exercise to identify the strengths and limitations of the different approaches. Ultimately, the research will culminate in guidance for regulatory applications involving digital system risk arguments. The staff is seeking subcommittee input on the general direction of the research and regulatory guidance at this time. Mr. Arndt illustrated the overall risk program using a figure that the presenters would refer to throughout the meeting.

Mr. Arndt noted that the research is focused on three major outcomes: the determination of what systems need to be modeled at what level of detail and accuracy, the development of an independent NRC analysis capability, and the development of acceptance criteria for risk-informed approaches. Therefore, the research plans to provide data and analysis methods to support risk-informed regulatory methods and plans to interact frequently with the Committee to obtain their input during the process and ultimately, their endorsement of the proposed methods and regulatory guidance.

General Comments and Observations From the Subcommittee Members and Consultants

- Dr. Apostolakis suggested that the review of failure data for digital systems was the most important activity on the figure describing the digital system risk program. He asked if it

feeds into both the traditional and dynamic modeling activities. Mr. Arndt confirmed that it does.

- Dr. Apostolakis stated that the critical outcome is the determination of what systems need to be modeled, to what level of detail, to what level of accuracy, and by which method. He felt that the reports give the impression that they treat everything the same. Mr. Arndt replied that the staff is developing a set of characteristics of digital systems that will point to the necessary modeling requirements.
- Dr. Bickel asked how the staff is dividing the resources among the different types of systems. Mr. Arndt replied that from the research view, they are applying more resources in areas where the level of knowledge is less, while from the regulatory view, they are using more effort where there is a greater effect on health and safety. Dr. Kress stated that this concern points to the need for risk-importance measures.

Development of a Probabilistic Approach for Modeling Failures of Digital Systems Using Dynamic Methods

For this portion of the presentation, Mr. Arndt was joined by Professor Tunc Aldemir, the principal investigator from The Ohio State University. Mr. Arndt began by reviewing some of the background information on this project, including the concepts of evaluating the system from a system standpoint and compatibility with current PRA methods. He stated that there may be a need to account for the dynamic interactions of a digital system to capture both interactions between the digital system and the physical plant (Type 1) and interactions within digital systems (Type 2). He then reviewed the objectives of this portion of the research, namely, to perform a pilot study of the proposed methods, identify any potential pitfalls of the methods, and review the supporting analysis and data to develop appropriate acceptance criteria.

Mr. Arndt briefly discussed the overall approach to the research project and its progress to date, which identified two leading candidate methods to evaluate, the Markov method and the dynamic flowgraph methodology (DFM). They have published the initial steps of the research in NUREG/CR-6901. This report also concluded that they should define a benchmark system to allow assessment of the methods. Mr. Arndt then described the benchmark system based on a digital feedwater control system from an operating pressurized water reactor. He provided the operating characteristics for the system and the control laws that govern its operation. The key point of discussing the control laws was to notice that the system uses operating history in its state space, which complicates the analysis of the system. Mr. Arndt also described the system's fault-tolerant features designed to increase its reliability.

Dr. Aldemir then discussed some sample operations of the benchmark system, including an example where the exact timing of the failure could lead to either an overflow or underfill condition. He then discussed the modeling philosophy for the Markov and DFM models. The models will use failure information from both plant historical data and a generic failure database, along with new system testing data. Dr. Aldemir then described how the project is using fault injection to produce failure data, specifically the coverage factor. He finished the description of the benchmark system by discussing the controller failure model and some example failure parameters that they can produce.

Following the break, Mr. Carl Elks, a researcher from the University of Virginia who is participating in the project, offered some additional insights regarding the use of fault injection. He described its use in a specific testing regime to collect specific information, primarily the coverage factor. He noted that it is important when using fault injections to be rigorous and support any assumptions made. The project is using both random fault injection and a guided fault injection to produce failure data.

Dr. Aldemir then resumed his presentation by describing the PRA model being used for the dynamic modeling exercise. The modeled plant is a three-loop PWR with a PRA modeled in SAPHIRE. As an example, he described the feedwater control system's operation during a turbine trip event illustrated via the PRA event trees.

Dr. Aldemir continued the discussion by describing the DFM model for the exercise. He provided a brief background on the development and features of DFM. At Dr. Apostolakis's request, Dr. Aldemir skipped through the basic steps of DFM and its uses to discuss how DFM supports a risk assessment. To do so, he used an example to show how to identify the pivotal events which is addressed by finding the prime implicants. He showed how to use DFM to model the causality flow of events.

Dr. Michael Yau, ASCA Inc., another participant in the project, presented a short description of an example DFM analysis. They analyzed the benchmark system for two possible failures: steam generator high level and steam generator low level. He described how they defined these top events in terms of system variables, and showed how they solved the model to identify 11 prime implicants for the high level failure.

Dr. Aldemir next discussed the Markov model of the same benchmark system. He briefly described the modeling process and noted that the primary difference between DFM and Markov is that DFM uses binary values (i.e., 0/1) in its decision tables, while the Markov model uses continuous values. He then showed how they implement the control laws in the Markov decision tables

Dr. Aldemir continued by briefly describing how the results from the DFM and Markov models are incorporated back into the PRA models of the overall plant, specifically noting how to implement the results of the models into SAPHIRE fault trees. He then concluded his portion of the presentation by reviewing how the characteristics of the benchmark system comply with the benchmark requirements and how the models comply with the modeling requirements set forth in previous portions of the research.

Mr. Arndt concluded this session of the meeting by describing the future plans for this project. He specifically mentioned plans for a public workshop in August and application of the modeling methods on a second benchmark system, a simpler actuation system.

General Comments and Observations From the Subcommittee Members and Consultants

- Dr. Apostolakis asked if these types of digital systems are currently in use. Mr. Arndt replied that various systems are in use, but we have not licensed any under the current regulations. Dr. Bickel noted that Combustion Engineering plants have been running software-based systems since 1978. Mr. Kemper added that some older digital systems have even been upgraded under 50.59.

- Dr. Bonaca asked if any plants model a system like the benchmark system in their PRA. Mr. Arndt answered that some do, though usually as a black-box component. He noted that the UK's Sizewell plant uses a detailed model in their PRA, though it is not a dynamic model.
- Dr. Kress asked if the timing and mode of the failure matters. Dr. Aldemir replied that it does matter, since the type of failure leads to different responses.
- Dr. Apostolakis asked if the details of the failure matter, as long as the analyst captures both failure modes. Dr. Bickel pointed out that such a phenomenon is not unique to digital systems. Mr. Arndt and Dr. Aldemir answered that the dynamic modeling has the greatest effect on quantification.
- Dr. Apostolakis commented that the method used to estimate the probability of no faults when no failures are found is inconsistent with current PRA approaches. He also pointed out that the relevant paper cited in the report does not support the proposed method. Mr. Elks replied that the approach has been used and documented as a way to estimate a bound on the reliability. Dr. Apostolakis disputed this assertion.
- Dr. Apostolakis also asked about the meaning of the λ transition rate, and whether it matches the BNL failures causes. Mr. Arndt clarified that the BNL research is looking at the types of failures, while the DFM and Markov models are system models which use transition rates for particular failure causes. Dr. Apostolakis cautioned the staff to be careful what kind of information a model provides and how it can be used. He also suggested that they need to coordinate the data tables between the traditional and dynamic methods. He later added that the important question is the meaning and derivation of meaningful values for the λ 's. Mr. Arndt offered to address that concern at the next meeting.
- Dr. Apostolakis asked how the data is used when the discovered errors are fixed. Mr. Arndt replied that they examine two things: the likelihood of faults still remaining, and the likelihood that some parts of the system were not tested.
- Dr. Apostolakis asked if the controller failure model also captures software failures. Mr. Arndt answered that the type of fault causing the failure in the model does not matter, so it does handle software failures as well. Dr. Apostolakis asked if λ can be used for software. Mr. Arndt answered that it can, in theory. Dr. Apostolakis expressed doubts and requested that its use be explained better.
- Dr. Apostolakis requested a comparison of results from the two methods. Mr. Arndt noted that they have not yet produced such a comparison since the staff is using a staged approach. Mr. Kemper added that they are working to make the information public quickly since industry is asking for guidance, but that the staff will see what they can do to address the request.

Development of a Probabilistic Approach for Modeling Failures of Digital Systems Using Traditional PRA Methods

After a brief introduction by Mr. Arndt to remind the subcommittee how this session fits into the overall risk research program, Mr. Todd Hilsmeier began the first of three presentations related to this topic. He reviewed the background for the project and discussed its objective to develop a probabilistic method for modeling failures using traditional PRA methods such as static fault trees and event trees that they can integrate with a PRA. He provided an overview of the tasks associated with the project and the status of each. Mr. Hilsmeier specifically highlighted task 5, the gathering and analysis of reliability data, and task 8a, the review of system failure events, which would be the topic of the subsequent presentations.

Dr. Tsong-Lun (Louis) Chu provided a detailed discussion of the development of a failure database for digital system hardware. The objective of this project was the development of a generic failure parameter database for digital components based on currently available data. To perform this task, Dr. Chu described their review of various failure rate databases and hardware reliability prediction methods. He specifically discussed the use of PRISM, a software tool developed by the Reliability Analysis Center for making reliability predictions of hardware components. However, he noted that large variations exist in the PRISM data. Dr. Chu then described their work on a hierarchical Bayesian analysis of the PRISM data, which still resulted in wide population variability distributions due to the variations in the failure records. He finished by providing the digital component failure rates produced by the analysis and suggested that more applicable data should be collected. Mr. Jeff Stone, Constellation Energy, asked if the research was examining on-demand failures as well as failure rates in time. Dr. Chu answered that they did not. Mr. Stone stated that this is important due to the "shock" effects of the demand and the time needed to respond to the demand.

Mr. Gerardo Martinez-Guridi led the next presentation on Brookhaven's review of software-induced failure events. This review included events at domestic nuclear power plants, other industries, and foreign nuclear power plants. He stated that the objectives of the study are to discuss software failures, the approach used to collect operational events, and address ACRS comments in light of the insights gained during the review of the events. Mr. Martinez-Guridi continued by discussing their preliminary model of software failure, which examines both internal and external causes and the propagation of the failure through to the controlled device, the associated system, and the overall plant. He also pointed out the potential for dependent failures due to the use of common or similar software in redundant channels of a system.

Mr. Martinez-Guridi then reviewed the results of their search for software failures at domestic nuclear power plants. The search included 22 years of Licensee Event Reports (LERs) for the keyword "software" and added six events from NUREG/CR-6734 and one additional event known by the researcher's personal knowledge. The staff found 130 software failures in 113 LERs. They analyzed the 45 most recent results to classify the software failure modes. Mr. Martinez-Guridi discussed these results as well, noting that 31 of the 45 events were failures where the software ran with incorrect results that were not evident. He described the main causes of failure as software requirements analysis (36%) and operation and maintenance (27%). In 29 of the events, some type of dependent failure occurred.

Mr. Martinez-Guridi continued by describing an internet search used to identify software failure events in other industries and at foreign nuclear power plants. Identified sources included the

NTSB Aviation Accident Database, the NASA web site, news media web sites, and various reports compiled by other organizations. He described the analysis of 48 events in 10 industries and the categorization scheme developed to capture the failure modes and causes. Mr. Martinez-Guridi then discussed the insights gained from the review and provided details on several events.

Mr. Martinez-Guridi concluded by discussing recent ACRS comments, including their viewpoints on system-centric versus software-centric approaches. He stated that software failures occur due to triggering events, which occur randomly. Therefore, the failures can be modeled probabilistically. He briefly discussed their review of methods for identifying software faults, such as formal methods, and methods for quantifying software reliability, such as Bayesian Belief Networks.

General Comments and Observations From the Subcommittee Members and Consultants

- Dr. Apostolakis asked if the staff was still studying the EPRI report 1002835 on defense-in-depth and diversity for digital systems. Mr. Arndt replied that the staff has not performed a formal review, but did examine the report and assimilated its information.
- Dr. Apostolakis asked about the use of LER data. Dr. Chu stated that more information on the usage needed to be collected in order to make the number of failures useful. Dr. Bickel suggested that more information exists within the LER database than is being used, but conceded that it does not include the operating information Dr. Chu referred to.
- Dr. Apostolakis asked if we can learn from events in other industries. Mr. Martinez-Guridi replied that we can, because some of those failures can be catastrophic and the failure modes of the software are applicable.
- Dr. Bonaca asked if software in other industries meets the same quality requirements that we require. Dr. Chu replied that many variations exist in the quality requirements across different industries, and that they did not examine these differences.

Development of Regulatory Guidance for Risk-Informing Digital System Reviews

Mr. Arndt returned to lead the final formal presentation of the day on the development of regulatory guidance for risk-informing digital system reviews. This guidance will rely on the research discussed during the previous sessions of this meeting. Mr. Arndt noted that industry has expressed a strong interest in using risk-informed regulation as an alternate method for licensing digital systems. He discussed the need for the guidance since Regulatory Guide 1.174 does not provide specific criteria for digital systems. The new guidance will address the unique characteristics of digital systems with regard to digital system modeling, maintaining sufficient safety margins, meeting our defense-in-depth philosophy, and identifying performance measurement strategies.

Mr. Arndt then discussed the overall structure for the draft guide, which will incorporate modeling requirements, integration with PRA models, data requirements, uncertainty analysis, acceptance criteria, and other issues unique to digital systems. He discussed details of the preliminary modeling requirements and addressed the issue of the level of modeling detail

required. The staff plans to have additional subcommittee and/or full Committee involvement in the development of the regulatory guide.

Mr. John Gaertner, EPRI, added his concerns regarding the development of the regulatory guide, specifically with regard to the incorporation of digital I&C details into the PRA and the defense-in-depth requirements.

General Comments and Observations From the Subcommittee Members and Consultants

- Dr. Apostolakis asked if many digital upgrades are occurring under 50.59. Mr. Arndt answered affirmatively, but added that the staff has specifically asked industry to bring in some types of upgrades for review.

Comments by Industry

Following the staff's formal presentation, Mr. Alex Marion, NEI, provided oral comments on behalf of the industry. He stated that the development of the regulatory guide is extremely important to the industry, as it could become a barrier to the deployment of digital systems. Their primary concern revolves around their need to understand how to meet the staff's expectations. He stated support for a risk-informed process to prioritize efforts on risk-significant topics, though he did not think that the topics of this meeting were risk-significant. He also raised a concern regarding the timeliness of the work with regard to new plants.

Mr. Marion expressed hopes that a formal review of the EPRI Defense-in-Depth and Diversity document will proceed. He also expressed support for benchmark activities and plans to offer the NRC an integrated action plan to involve industry. He stated a desire to have industry involvement in the peer review process of the research. He pointed out the importance of understanding the differences between simple and complex digital applications. He added that he sees many digital system errors more as configuration management issues.

General Comments and Observations From the Subcommittee Members and Consultants

- Dr. Apostolakis stated that he thinks a cooperative research program in this area would be useful.
- Dr. Bonaca stated that having a real application on the table to refer to would be helpful, such as the resubmission of the Oconee upgrade.

Closing Discussions

General Comments and Observations From the Subcommittee Members and Consultants

- Dr. Kress saw lots of progress and believes the staff is on the right track. He noted the need for early judgement regarding the systems that need to be modeled and suggested the use of risk-importance measures. He suggested more work to clarify the statistical method when no failures are found in the data. Dr. Kress requested to hear more detail regarding the development of the λ 's. He agreed that failures per demand would be more interesting than failures in time. He stated that it seems like digital upgrades should decrease the overall risk. He supports reevaluating defense-in-depth and

diversity for digital systems and industry involvement in the peer review process. He closed by noting that applications in new plants may need different acceptance criteria.

- Dr. Bonaca stated his agreement with most of Dr. Kress's comments, and specifically agreed that the staff is making good progress. He stated that determining which systems need modeled is important. He noted Mr. Gaertner's comments on the incorporation of digital I&C into the PRA and suggested that maybe other ways exist to accomplish the goal. He also stated that he has high expectations for the regulatory guide.
- Dr. Bickel suggested the need for more focused prioritization for the modeling and analysis capability. He sees a need for a projection of the types of systems that will be coming for review, and suggests a need to focus on trip and emergency actuation systems. He stated that the staff could improve the data mining methods to investigate issues like configuration control and gaining more information from the LER database.
- Dr. Apostolakis closed the meeting by adding that he is also pleased with the progress, though he would like to gain a better understanding of the transition rates.

SUBCOMMITTEE DECISIONS AND ACTIONS

The research discussed at this meeting will ultimately feed into the development of a regulatory guide on risk-informing digital system reviews. The staff plans to bring this regulatory guide to the full Committee once a draft is ready.

BACKGROUND MATERIALS PROVIDED TO THE SUBCOMMITTEE PRIOR TO THIS MEETING

1. Aldemir, T., et al., "Reliability Modeling of Digital Instrumentation and Control Systems for Nuclear Reactor Probabilistic Risk Assessments," Draft Report for Comment, May 2006.
2. Chu, T.S., et al., "Collection of Failure Data and Development of Database for Probabilistic Modeling of Digital Systems," August 2006.
3. Chu, T.S., et al., "A Review of Software-Induced Failure Experience," Draft Letter Report, May 2006.

Note: Additional details of this meeting can be obtained from a transcript of this meeting available for downloading or viewing on the Internet at <http://www.nrc.gov/what-we-do/regulatory/advisory/acrs.html> or purchase from Neal R. Gross and Co., Inc., (Court Reporters and Transcribers) 1323 Rhode Island Avenue, NW, Washington, DC 20005 (202) 234-4433.

**Advisory Committee on Reactor Safeguards
Digital Instrumentation and Control Systems Subcommittee Meeting
Rockville, MD
27 June 2006**

- Proposed Agenda -
Rev. 6/26/06

Cognizant Staff Engineer: Eric Thornsby (301-415-8716, eat2@nrc.gov)

Topic	Presenter(s)	Time
June 27 - Open Session		
	Opening Remarks and Objectives	G. Apostolakis, ACRS 8:30 - 8:35 am
	Introduction and Overview of Digital System Risk Research Program	<i>W. Kemper, RES</i> S. Arndt, RES 8:35 - 9:00 am
I	Development of a Probabilistic Approach for Modeling Failures of Digital Systems using Dynamic Methods	S. Arndt, RES T. Aldemir, OSU 9:00 am - 12:00 pm (10:30-10:45 break)
	Lunch	12:00 - 12:45 pm
II	Development of a Probabilistic Approach for Modeling Failures of Digital Systems using Traditional PRA Methods	T. Hilsmeier, RES T. Chu, BNL 12:45 - 3:00 pm
	Break	3:00 - 3:15 pm
III	Development of Regulatory Guidance for Risk-Informing Digital Systems Reviews	S. Arndt, RES 3:15 - 5:00 pm
IV	Comments by Industry	J. Harris, NEI 5:00 - 5:30 pm
	Adjourn	<i>A. Marston</i> 5:30 pm

Notes:

- Presentation time should not exceed 50% of the total time allocated for a specific item.
- Number of copies of presentation materials to be provided to the ACRS - 35.

The joint Subcommittees will review three current human reliability assessment issues: the ATHEANA User's Guide, the application of ATHEANA to pressurized thermal shock, and comments received on the HRA Methods Evaluation NUREG. The Subcommittee will hear presentations by and hold discussions with representatives of the NRC staff and industry regarding this matter. The Subcommittees will gather information, analyze relevant issues and facts, and formulate proposed positions and actions, as appropriate, for deliberation by the full Committee.

Members of the public desiring to provide oral statements and/or written comments should notify the Designated Federal Official, Mr. Eric A. Thornsbury (Telephone: 301-415-8716) five days prior to the meeting, if possible, so that appropriate arrangements can be made. Electronic recordings will be permitted.

Further information regarding this meeting can be obtained by contacting the Designated Federal Official between 7:30 a.m. and 4:15 p.m.(ET). Persons planning to attend this meeting are urged to contact the above named individual at least two working days prior to the meeting to be advised of any potential changes to the agenda.

Dated: May 18, 2006.

Michael R. Snodderly,
Acting Branch Chief, ACRS/ACNW.
[FR Doc. E6-8033 Filed 5-24-06; 8:45 am]
BILLING CODE 7590-01-P

NUCLEAR REGULATORY COMMISSION

Advisory Committee on Reactor Safeguards; Meeting of the ACRS Subcommittee on Digital Instrumentation and Control Systems; Notice of Meeting

The ACRS Subcommittee on Digital Instrumentation and Control Systems will hold a meeting on June 27, 2006, Room T-2B3, 11545 Rockville Pike, Rockville, Maryland.

The entire meeting will be open to public attendance.

The agenda for the subject meeting shall be as follows:

Tuesday, June 27, 2006—8:30 a.m. until the conclusion of business.

The Subcommittee plans to review the ongoing digital system risk program and the development of regulatory guidance on risk informed digital system reviews. The Subcommittee will hear presentations by and hold discussions with representatives of the NRC staff regarding this matter. The

Subcommittee will gather information, analyze relevant issues and facts, and formulate proposed positions and actions, as appropriate, for deliberation by the full Committee.

Members of the public desiring to provide oral statements and/or written comments should notify the Designated Federal Official, Mr. Eric A. Thornsbury, (Telephone: 301-415-8716) five days prior to the meeting, if possible, so that appropriate arrangements can be made. Electronic recordings will be permitted.

Further information regarding this meeting can be obtained by contacting the Designated Federal Official between 7:30 a.m. and 4:15 p.m.(ET). Persons planning to attend this meeting are urged to contact the above named individual at least two working days prior to the meeting to be advised of any potential changes to the agenda.

Dated: May 18, 2006.

Michael R. Snodderly,
Acting Branch Chief, ACRS/ACNW.
[FR Doc. E6-8034 Filed 5-24-06; 8:45 am]
BILLING CODE 7590-01-P

PRESIDIO TRUST

Notice of Public Meeting

AGENCY: The Presidio Trust.

ACTION: Notice of public meeting.

SUMMARY: In accordance with section 103(c)(6) of the Presidio Trust Act, 16 U.S.C. 460bb note, Title I of Public Law 104-333, 110 Stat. 4097, as amended, and in accordance with the Presidio Trust's bylaws, notice is hereby given that a public meeting of the Presidio Trust Board of Directors will be held commencing 6:30 p.m. on Thursday, June 15, 2006, at the Golden Gate Club, 135 Fisher Loop, Presidio of San Francisco, California. The Presidio Trust was created by Congress in 1996 to manage approximately eighty percent of the former U.S. Army base known as the Presidio, in San Francisco, California.

The purposes of this meeting are to approve minutes from the last Board meeting, to adopt a revised budget for Fiscal Year 2006, to provide an Executive Director's Report, to present the final Supplemental Environmental Impact Statement in connection with the rehabilitation of the Public Health Service Hospital, and to receive public comment in accordance with the Trust's Public Outreach Policy.

Accommodation: Individuals requiring special accommodation at this meeting, such as needing a sign language interpreter, should contact

Mollie Matull at (415) 561-5300 prior to May 31, 2006.

FOR FURTHER INFORMATION CONTACT:
Karen Cook, General Counsel, the Presidio Trust, 34 Graham Street, P.O. Box 29052, San Francisco, California 94129-0052, Telephone: (415) 561-5300.

Dated: May 22, 2006.

Karen A. Cook,
General Counsel.
[FR Doc. E6-8114 Filed 5-24-06; 8:45 am]
BILLING CODE 4310-4R-P

RAILROAD RETIREMENT BOARD

Agency Forms Submitted for OMB Review

Summary: In accordance with the Paperwork Reduction Act of 1995 (44 U.S.C. Chapter 35), the Railroad Retirement Board (RRB) has submitted the following proposal(s) for the collection of information to the Office of Management and Budget for review and approval.

Summary of Proposal(s)

- (1) *Collection title:* Employee Representatives' Status and Compensation Reports.
 - (2) *Form(s) submitted:* DC-2a, DC-2.
 - (3) *OMB Number:* 3220-0014.
 - (4) *Expiration date of current OMB clearance:* 7/31/2006.
 - (5) *Type of request:* Extension of a currently approved collection.
 - (6) *Respondents:* Business or other for-profit.
 - (7) *Estimated annual number of respondents:* 65.
 - (8) *Total annual responses:* 65.
 - (9) *Total annual reporting hours:* 33.
 - (10) *Collection description:* Benefits are provided under the Railroad Retirement Act (RRA) for individuals who are employee representatives as defined in section 1 of the RRA. The collection obtains information regarding the status of such individuals and their compensation.
- Additional Information or Comments:** Copies of the forms and supporting documents can be obtained from Charles Mierzwa, the agency clearance officer (312-751-3363) or Charles.Mierzwa@rrb.gov.

Comments regarding the information collection should be addressed to Ronald J. Hodapp, Railroad Retirement Board, 844 North Rush Street, Chicago, Illinois, 60611-2092 or Ronald.Hodapp@rrb.gov and to the OMB Desk Officer for the RRB, at the Office of Management and Budget,

ADVISORY COMMITTEE ON REACTOR SAFEGUARDS

SUBCOMMITTEE MEETING ON DIGITAL INSTRUMENTATION
AND CONTROL SYSTEMS

JUNE 27, 2006

TODAY'S DATE: JUNE 27, 2006

NRC STAFF ATTENDEES PLEASE SIGN BELOW

PLEASE PRINT

NAME	AFFILIATION
1. STEVEN ARNDT	RES/DFERR
2. Todd H. Humes	RES/DRASP
3. JOHN LAI	RES/DRASP
4. CLIFF DONT	NRR/DRA
5. Roman Shaffer	RES/DFERR
6. BILL KEMPER	RES/DFERR/IEES
7. MATT CHIRAMAL	NRR/DE
8. PAUL LOESER	NRR/DE/ELCB
9. Tekia Gowan	RES/DFERR/IEES
10. RONALDO V. JENKINS	RES/DRASP
11. Tolani Owusu	RES/DFERR/IEES
12.	
13.	
14.	
15.	
16.	
17.	
18.	
19.	
20.	

ADVISORY COMMITTEE ON REACTOR SAFEGUARDS

SUBCOMMITTEE MEETING ON DIGITAL INSTRUMENTATION AND CONTROL SYSTEMS

June 27, 2006

June 27, 2006
Date

PLEASE PRINT

ATTENDEES PLEASE SIGN-IN FOR THE MEETING

	<u>NAME</u>	<u>AFFILIATION</u>
1	Jim Riley	NEI
2	Thuy Nguyen	EPRI / EDF
3	Tunc Aldemir	Ohio State Univ.
4	Xiaodong Sun	" " "
5	LARRY ERIN	WESTINGHOUSE
6	TIMOTHY WILSON	" "
7	Alex Marion	NEI
8	TSONG-CUN CHU	BNL
9	G. Martinez-Gundi	BNL
10	DAN MCLAUGHLIN	WESTINGHOUSE
11	Tony HARRIS	NEI
12	JEFF STONE	CONSTELLATION ENERGY
13	Rick WACHOWIAK	GE
14	JOHN GAERTNER	EPRI
15	MICHAEL YAU	ASCA, INC.
16	JOH.	
17	Bob ENZINNA	AREVA
18	Phil Liddle	AREVA
19	Tunc Aldemir	OHIO STATE U
20	ROBERT MANU	AREVA

ADVISORY COMMITTEE ON REACTOR SAFEGUARDS

SUBCOMMITTEE MEETING ON DIGITAL INSTRUMENTATION AND CONTROL SYSTEMS

June 27, 2006

June 27, 2006

Date

PLEASE PRINT

ATTENDEES PLEASE SIGN-IN FOR THE MEETING

	<u>NAME</u>	<u>AFFILIATION</u>
1	Kimberly Keithline	NEI
2	(cur) FLKS	UVA
3	Mary Prestley	MIT
4		
5		
6		
7		
8		
9		
10		
11		
12		
13		
14		
15		
16		
17		
18		
19		
20		



OVERVIEW OF DIGITAL SYSTEM RISK RESEARCH PROGRAM

Advisory Committee on Reactor Safeguards
Digital Instrumentation and Control Subcommittee

June 27, 2006

Steven A. Arndt

Instrumentation and Electrical Engineering Branch
Division of Fuel, Engineering & Radiological Research
Office of Nuclear Regulatory Research
(301-415-6502, saa@nrc.gov)



OVERVIEW(1/2)

- Research will investigate potential procedures and methods for inclusion of reliability models for digital systems into current generation nuclear power plant PRA, develop these methods to the point they can be integrative into current agency tools, and develop needed regulatory guidance
 - Assessing what modeling methods might be usable
 - Determining which systems need to be modeled and at what level of detail
 - Developing and testing methods
 - Developing regulatory acceptance criteria



OVERVIEW (2/2)

- Issues facing NRC

- Licensees are replacing analog systems with digital systems
- Licensing these digital systems presents challenges to NRC
 - Industry has expressed interest in using risk-informed regulation (Regulatory Guide 1.174) as an alternate method for licensing these systems
 - Research into the limitations of digital systems reliability modeling does not currently support expanded use of risk information in licensing digital systems
- As the NRC licensees replace analog systems with digital systems the current PRA's are not keeping up with these changes
- NRC risk analysis tools and data (SAPHIRE and SPAR models) do not provide an independent means of assessing licensee analyses at present



Meeting with ACRS in June 2006

- ACRS Digital Instrumentation and Control Systems Subcommittee was briefed on the program plan
 - Wished to be consulted as the program progressed
 - Encouraged the review of software-induced failures, and recommended that lessons learned be feedback into the research conclusions
 - Encouraged the staff to critically review methods for assessment of reliability of systems
 - Encouraged the staff to view digital systems from a system standpoint, while acknowledging there may be some systems that can be treated as decoupled systems of components.

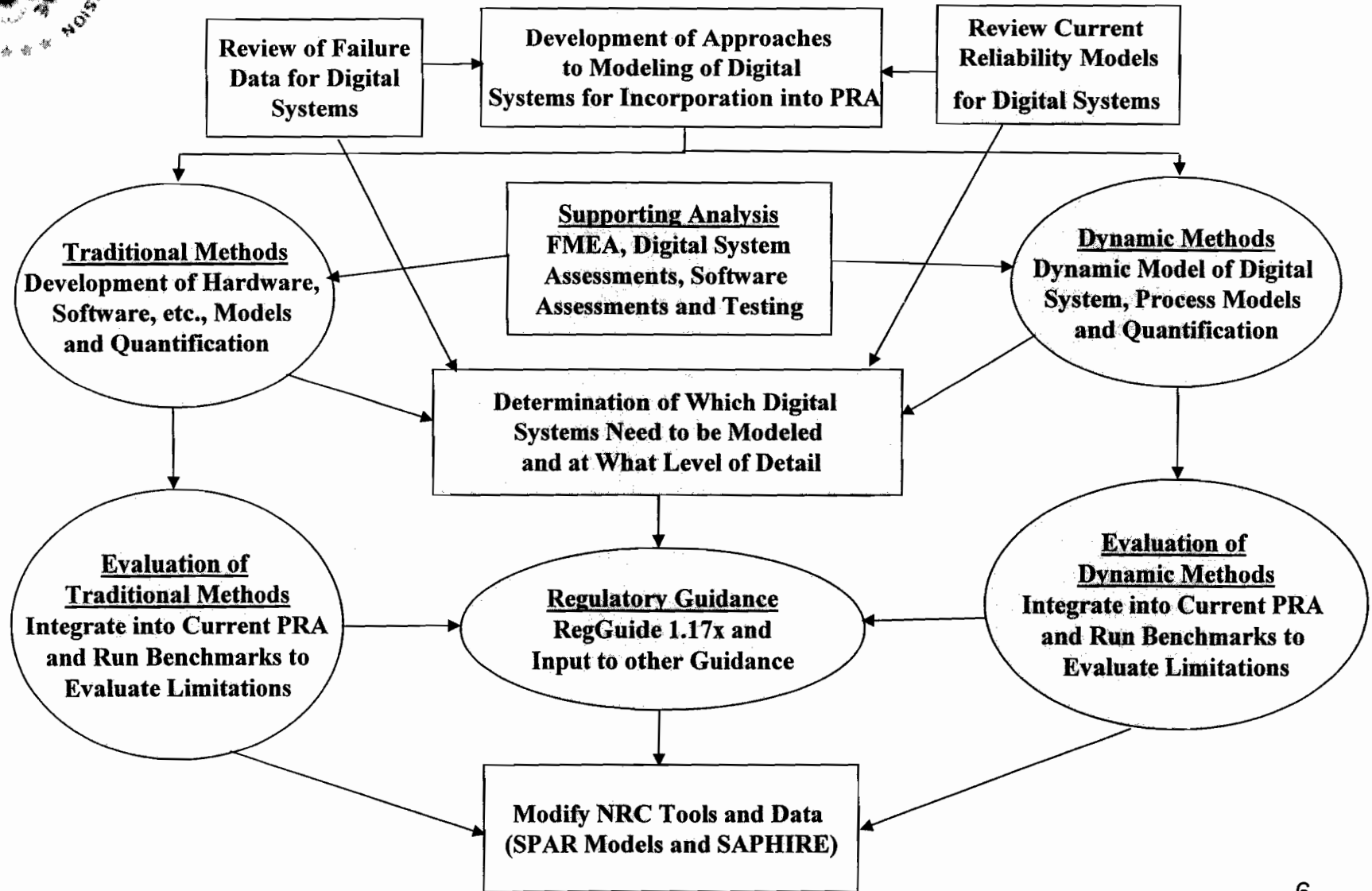


Digital System Risk Program

- **New methods for integrating current digital system models into PRAs are being developed**
 - **Pilot methods using both traditional methods and dynamic methods using models**
 - **Benchmarks of the capabilities of several methods will be completed**
 - **Uses and limitations of methods will be explored**
- **Guidance for regulatory applications involving digital systems reliability**
 - **acceptance criteria**
 - **limitations**
 - **evaluation methods**
 - **reliability data**



NRC Digital System Risk Program





RESEARCH FOCUS

- Structured to support three major outcomes
 - Determining what systems need to be modeled, at what level of detail, and what level of accuracy
 - Developing new capability to support independent analysis of digital systems
 - New or modified versions of current NRC PRA tools and data
 - Developing acceptance criteria for application of risk-informed approaches
- Broad-based research, focusing on review of possible methods, and data to support reliability analysis and acceptance criteria



SUMMARY

- This research will provide data, analysis methods, and acceptance criteria to support the use of risk-informed regulatory methods for the review of digital systems
- RES is looking forward to working closely with the ACRS as this program is implemented
 - Review of progress
 - Advise on best available methods
 - Review and endorsement of proposed methods
 - Review and endorsement of Regulatory Guidance



RELIABILITY MODELING OF DIGITAL INSTRUMENTATION AND CONTROL SYSTEMS FOR NUCLEAR REACTOR PROBABILISTIC RISK ASSESSMENTS

Advisory Committee on Reactor Safeguards
Digital Instrumentation and Control Subcommittee
June 27, 2006

Steven A. Arndt

Division of Fuel, Engineering & Radiological Research
Office of Nuclear Regulatory Research
(301-415-6502, saa@nrc.gov)

Tunc Aldemir

Nuclear Engineering Program
The Ohio State University
(614-292-4627, aldemir.1@osu.edu)

1



Presentation Organization

- Background
- Benchmark System
- Failure Data Generation
- Example PRA Model
- Dynamic Flowgraph Methodology
- Markov Methodology
- Incorporating DFM and Markov Models into the PRA
- Interfacing with SAPHIRE
- Procedures and the Requirements for the Reliability Modeling of Digital I&C
- Conclusion to Date and Next Steps

2



Background (1/2)

- U.S. NRC policy encourages the use of PRA and associated analyses to the extent supported by the state-of-the-art and data
- NRC is in the process of developing methods for estimating failure probabilities for digital systems and modeling methods needed to support risk-informed regulation of these systems
- The preferred method of evaluating a digital system is from a system stand point that requires modeling system interaction as well as hardware and software modeling
- For near term PRA applications, a digital I&C system reliability model needs to be compatible with the structure of current nuclear power plant PRAs, which use the static event-tree/fault-tree (ET/FT) approach

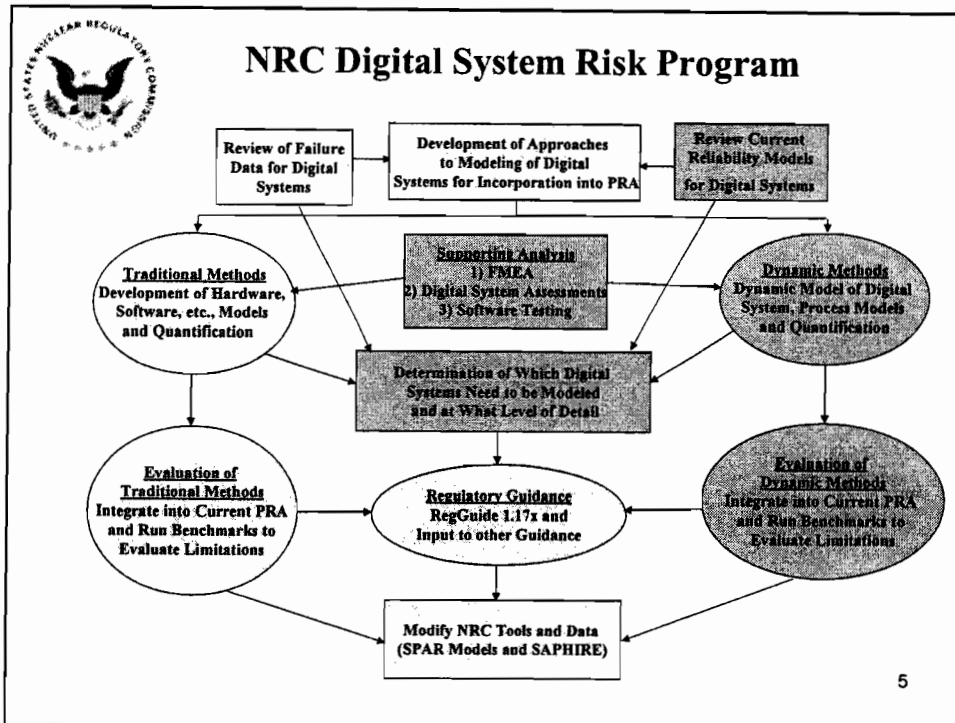
3



Background (2/2)

- From a reliability modeling perspective, this implies that there may be a need to account for the dynamic interactions
 - between digital I&C systems and controlled/monitored plant physical processes (e.g., heatup, pressurization), and
 - within digital I&C systems (e.g., communication between different components, multi-tasking, multiplexing)
- Digital I&C system reliability models accounting for such effects need to be incorporated into the existing PRA to assess whether the Δ CDF and Δ LERF due to proposed change in the I&C system vs. existing system meet an acceptance criteria

4



Objectives

Develop both procedures and methods for inclusion of reliability models for digital systems into current generation nuclear power plant PRAs, including

- a pilot study of the proposed methods,
- detailed reviews of the potential pitfalls of the methods developed, and
- detailed reviews of supporting analysis and data needed to develop Δ CDF and Δ LERF to support risk-informed regulation of nuclear power plant instrumentation and control criteria

6



Overall Approach

1. Investigate the applicability of the current static event tree/fault tree (ET/FT) approach to digital I&C systems
2. Review the advantages and limitations of available dynamic methodologies as they pertain to digital I&C systems relevant to reactor protection and control
3. Review other industries for practices in the reliability modeling of digital I&C systems
4. Review the existing regulatory framework with regard to requirements that a digital I&C control system must meet
5. Identify the minimum requirements a digital system model must meet for successful incorporation into an existing PRA
6. Identify available methodologies that meet these requirements
7. Demonstrate the methodologies identified in Step 6 using relevant benchmark systems

7



Progress to Date

- Steps 1 through 6 have been completed and the findings have been published in NUREG/CR-6901
- NUREG/CR-6901 has identified the Markov methodology and the dynamic flowgraph methodology (DFM) as methodologies that rank as the top two with most positive features and least negative or uncertain features when evaluated against the requirements for the reliability modeling of digital I&C systems.
- NUREG/CR-6901 also concluded that benchmark systems should be defined to allow assessment of the methodologies proposed for the reliability modeling of digital I&C systems using a common set of hardware/software/ firmware states and state transition data.

8



Benchmark System

- The benchmark system specification is based on the digital feedwater control system for an operating PWR.
- It has been generalized to be more representative of this type of digital systems.
- The feedwater system serves two steam generators (SGs).
- The purpose of the feedwater controller is to maintain the water level inside each of the SGs optimally within ± 2 inches (with respect to some reference point) of the setpoint level (defined at 0 inches).

9



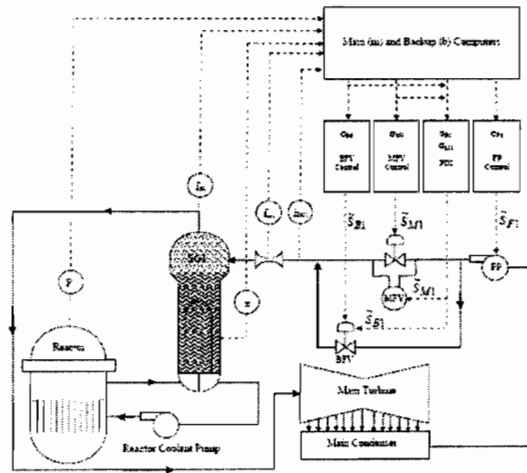
Benchmark System

- The controller is regarded failed if water level in a SG rises above +30 and falls below -24 inches.
- Each digital feedwater controller is connected to a feedwater pump (FP), a main feedwater regulating valve (MFV), and a bypass feedwater regulating valve (BFV).
- The controller:
 - regulates the flow of feedwater to the steam generators to maintain a constant water level in the steam generators,
 - provides a means for raising the temperature of the condensate received by the feed pumps, and,
 - provides a means for injecting chemicals into the steam generators from the chemical addition system.

10



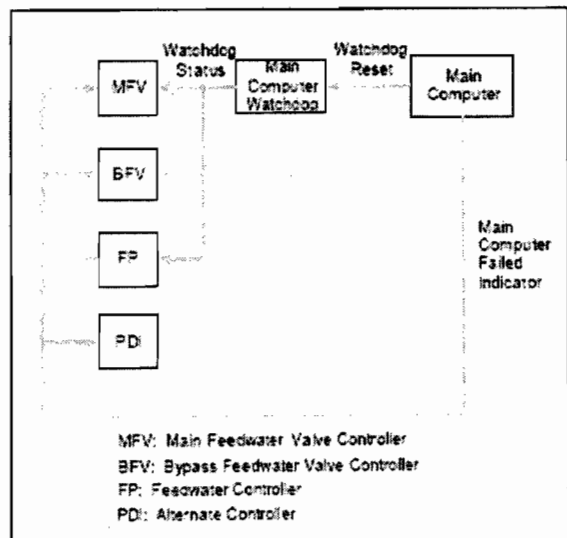
Benchmark System – Detailed View for Each Steam Generator



11



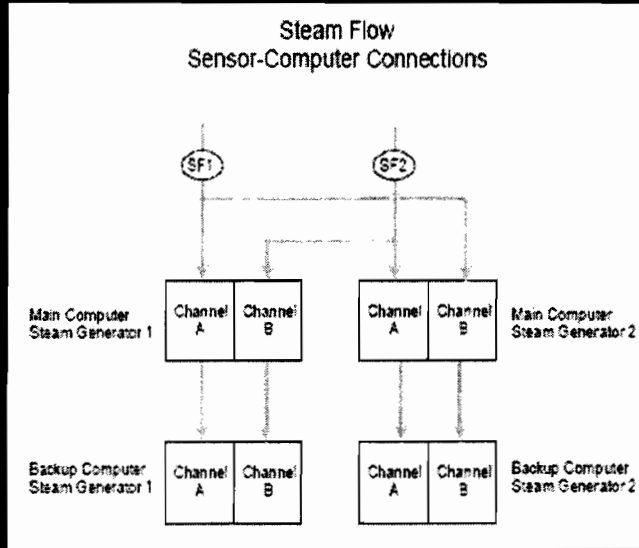
Benchmark System – Example Status Interconnections



12



Benchmark System - Example Sensor Signals



Benchmark System - Control Laws (1/2)

Rate of level change:

$$\frac{dh}{dt} = R(t) - I(t)$$

Flow Demand:

$$C(t) = \beta [C_1(t) + \alpha [C_2(t) - I(t)]] - \gamma(t)$$

Compensated Water Level:

$$v_c = \frac{dh}{dt} - C(t) + v + \tau \frac{dh}{dt}$$

Compensated Flow Error:

$$e_c = \frac{dh}{dt} - I_c(t) + \tau \left[\frac{dh}{dt} - \frac{dh}{dt} \right]$$

BFV Demand:

$$C_1(t) = v_c + \alpha v_c + \beta [C_2(t) - I(t)] - \gamma(t)$$

Compensated Power:

$$\tau \frac{dp}{dt} = -C(t) + p + \tau \frac{dp}{dt}$$

FP Demand:

$$r_c(t) = \begin{cases} \sigma_{hp} & \text{If High Power Operation} \\ \sigma_{lp} (\max\{C_1(t), C_2(t)\}) & \text{If Low Power Operation} \end{cases}$$

MFV Demand:

$$r_c(t) = \begin{cases} \sigma_{hp} C(t) & \text{If High Power Operation} \\ 0 & \text{If Low Power Operation} \end{cases}$$

BFV Demand:

$$r_c(t) = \begin{cases} 0 & \text{If High Power Operation} \\ C(t) & \text{If Low Power Operation} \end{cases}$$

FP Speed:

$$S_{fp} = \begin{cases} \sigma_{hp} & \text{Main CPU Operational} \\ \sigma_{bp} & \text{Main CPU Failed, Backup CPU Operational} \\ \sigma_{fp} & \text{Main CPU Failed, Backup CPU Failed} \end{cases}$$

MFV Position:

$$S_{mv} = \begin{cases} \sigma_{hp} & \text{Main CPU Operational} \\ \sigma_{bp} & \text{Main CPU Failed, Backup CPU Operational} \\ \sigma_{fp} & \text{Main CPU Failed, Backup CPU Failed} \end{cases}$$

BFV Position:

$$S_{bv} = \begin{cases} \sigma_{hp} & \text{Main CPU Operational} \\ \sigma_{bp} & \text{Main CPU Failed, Backup CPU Operational} \\ \sigma_{fp} & \text{Main CPU Failed, Backup CPU Failed} \end{cases}$$

PDI Decision:

$$S_{pd} = \begin{cases} 0 & S_{fp} = 0 \\ 1_{pd} & \text{Otherwise} \end{cases}$$



Benchmark System - Control Laws (2/2)

- The water inflow rate f_{wn} , steam flowrate f_{sn} , heat flux from the primary to the secondary side, level x_n , feedwater temperature for SGn are determined from the 2-volume SGn simulator package modeling the mass and energy transfer in SGn
- The control system provides feedpump speed, main flow valve position and bypass valve position to the simulator package
- The dynamic gain $\beta_{Fn}(f_{sn})$ and $\lambda_{Fn}(\sigma_{Bn})$ are obtained from table lookups
- η_{Fn} , η_{Mn} and η_{Bn} denote history data for the FP, MFV and BFV positions, respectively. If both MC and BC are failed, these data are used to determine the FP, MFV and BFV positions.

15



Benchmark System - Fault Tolerant Features

- Since the MFV, BFV, FP controllers forward the control signals to the corresponding control points, they provide a level of fault tolerance if both computers fail by allowing the operators time to intervene by holding the outputs of each to a previously valid value.
- The computers, MFV and BFV and FP, and PDI controllers are each connected to an independent power source wired to a separate bus. A single power source failure can only affect one computer, all of the MFV/BFV/FP controllers, or the PDI controller at one time.
- The computers are able to process the sensor inputs and perform the control algorithms within one third of the needed response frequency of the physical process. A failure in either computer can be detected and the fail over to a healthy component can occur with enough time to meet the response requirements of the process.

16



Benchmark System - Fault Tolerant Features

- The water level setpoint is taken from a switch connected to the MFV and is propagated to all computers. If the setpoint signal goes out of range, then the computers fall back on a preprogrammed setpoint value.
- Each computer is connected to a watchdog timer.
- Each computer verifies and validates its inputs, checking for out range and excessive rate changes in the inputs that would indicate errors in the sensor readings or problems with the analog to digital conversion of the values. Each computer will ignore input that fails these checks if the other inputs are still valid.
- The values of the inputs are averaged across redundant sensors.
- Deviation between the two sensors is detected and, if the deviation is large enough, the computer can signal a deviation error to the MFV, BFV, and FP controllers so they may switch to another computer.

17



Benchmark System - Fault Tolerant Features

- The PDI controller provides one more level of fault tolerance, in that it holds the MFV to a needed position if the MFV controller does not produce output. The MFV, BFV and FP controllers also check their inputs for range and rate of change checks; providing the ability to detect failures in the main and backup computers as well as the sensor data propagated to them.

18



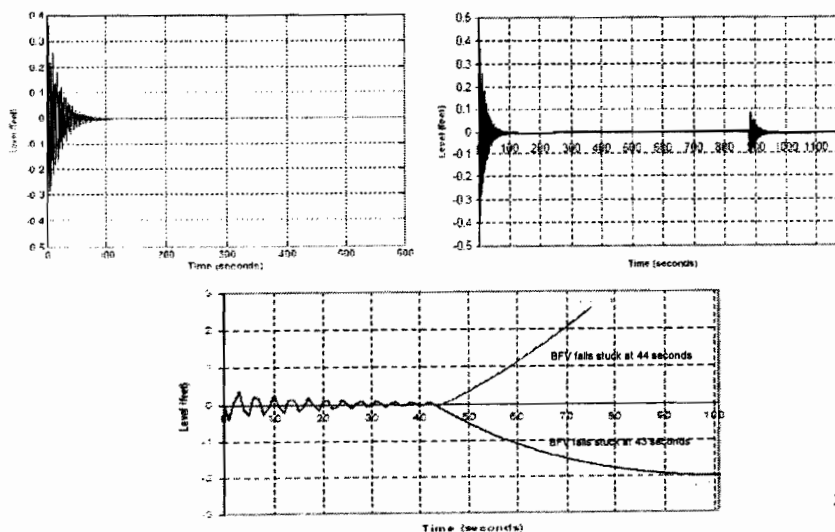
Benchmark System - Other Relevant Features

- Incorporates all of the properties of loosely-control coupled systems and most of the properties of tightly-control coupled systems.
- Properties of tightly-control coupled systems that are not represented are not relevant to instrumentation and control systems currently used in nuclear reactors (e.g. networking, shared external resources)
- Incorporates system history dependent control laws.
- Can lead to artifact generation under certain circumstances.
- System failure mode may depend on the exact timing of failure events.

19



Benchmark System - Operation Following a Turbine Trip with Main Computer Failed



20



Data Generation – Modeling Philosophy

- Define or choose metrics that allow models to be solved accurately.
- Choose models that are supported by observable, credible, measurable data.
- Choose models that are supported by plausible assumptions.
- All parameters of the model that cannot be deduced from the logical system design requirements must be measured.
- All such parameters must be measurable within a feasible amount of time.
- Uncertainties in the models should be accounted.
- Critical Parameters in the model must be statistically estimated with a confidence bound that is commensurate with overall system reliability.

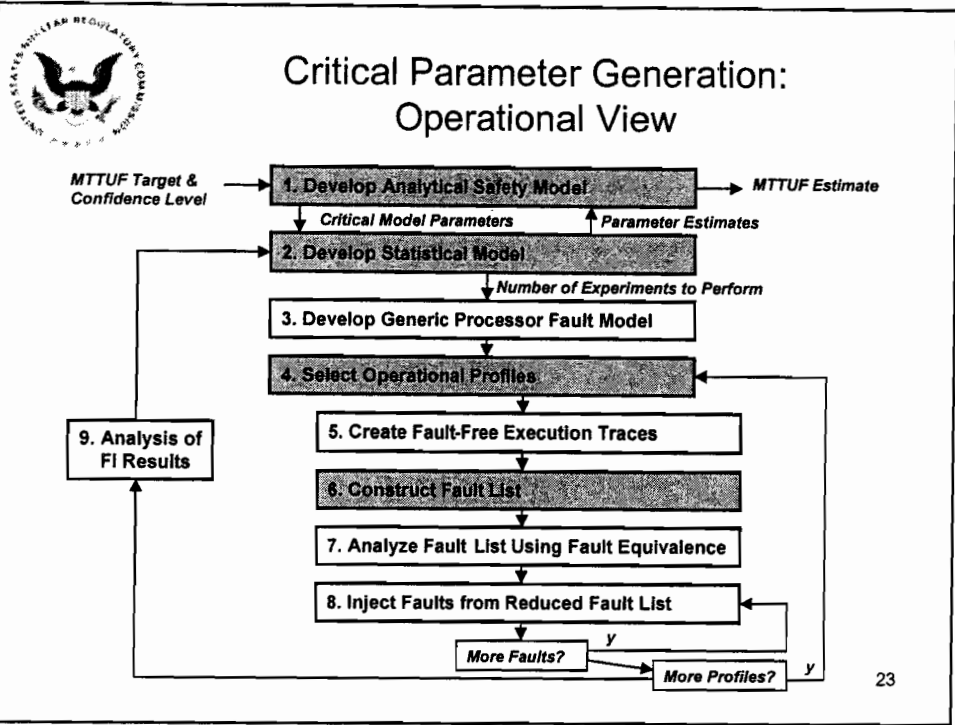
21



Development of Safety/Reliability Models

- Choose models that are supported by observable, credible, measurable data.
- Markov Models and DFM models need:
 - DFWCS component failure rates: Plant Historical data and RAC Prism database.
 - DFWCS Repair times: Plant Historical data.
 - System testing is used to develop additional needed data
 - Failure rates and fault or diagnostic coverage are experimentally determined through Fault Injection campaigns.
- Coverage is used to determine the likelihood for a undetected failure mode

22



23

Fault Injection Data Generation – How it works

- A fault injection experiment begins by selecting a set of faults from the fault library.
- Using the "bit flip injection method" we corrupt registers, memory locations where vital data is stored or processed. These faults induce the system into failure mode (say disrupting the feedback loop).
- For example and without loss of generality, say we inject 100 faults into the register files of the processor that store critical gain feedback parameters. Corruption of these parameters would de-stabilize the loop.
- Most of the time the system detects the injected errors, and correctly reconfigures the system to isolate the faulty processor. However, depending on the timing and duration of the fault we can get erroneous responses that were not detected by the system. These non-detected responses are the non-coverage (1-C) parameter for the models.
- This establishes a likelihood for a undetected unsafe failure mode. Non-Coverage 1-C.
- A detected failure is covered, and represented by the conditional probability C.

24



Operational Profiles

- Any testing or assessment process is sensitive to the input profile.
- Operational (Input/Output) profile data is collected from the *Cliff_time* plant monitoring data archive files.
 - Three years of data collected. Sampled every minute for 24 hours/day, every day.
- Contains plant data from various operational modes: Low power, high power, transitional, outage, testing, automatic, manual, failed components.
 - Log files will be used to synthesize accurate operational profiles for the Fault Injection experiments.
- Operational profiles (system inputs) are under the control of the assessor.

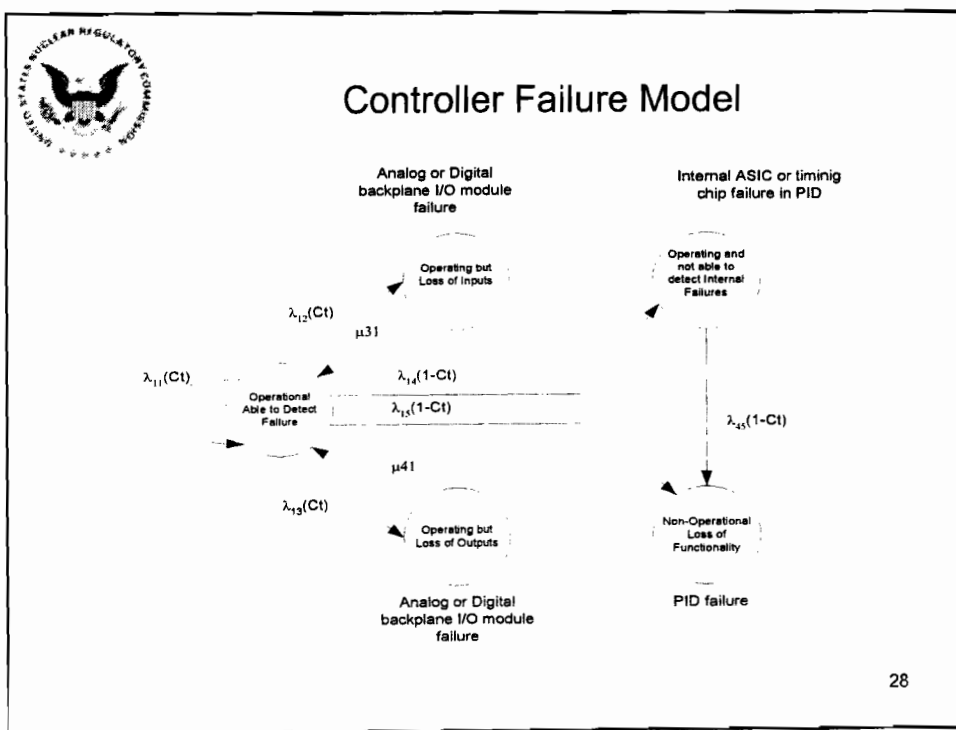
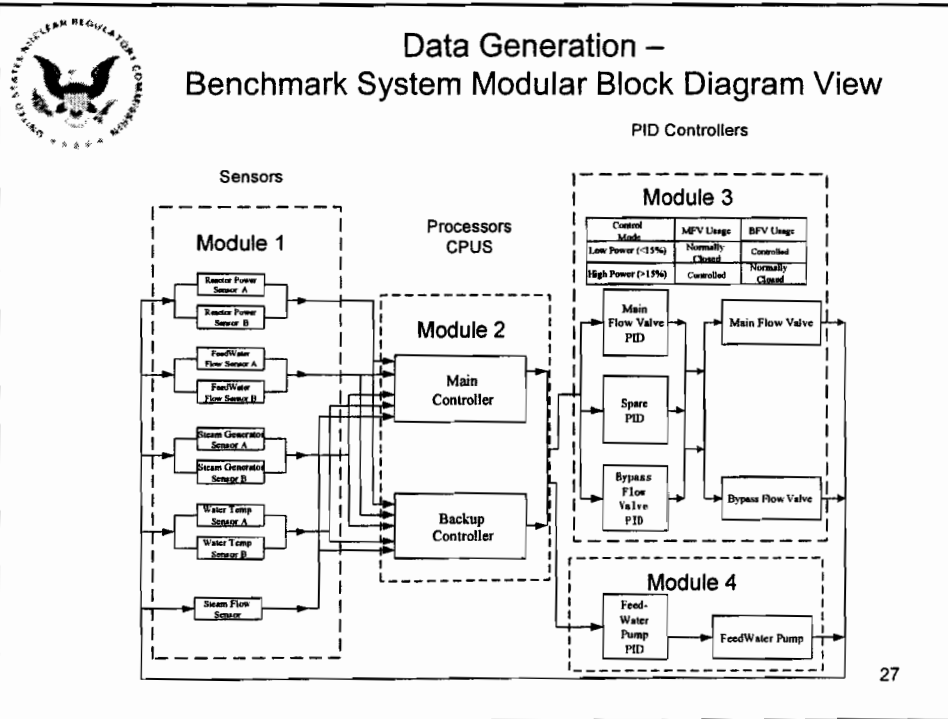
25



Safety and Reliability Models: Modular Markov Chain Modeling (UVA)

- Traditional Markov and Semi-Markov Models: Very general, make few assumptions, capable of modeling many different types of system behaviors and interactions.
- Disadvantages:
 - Computational State explosion
 - Model complexity impedes understanding and model validation (from a visual point of view)
- *Modular Markov Modeling:*
- A formal methodology that allows markov models to be composed in a modular way.
 - Addresses the issue of visual model complexity.
 - More closely tied to the functional architecture of the system.
 - A formal calculus of decomposition and composition
- Safety and reliability computed from the same model.
- Formally composes modules by their potential failure mode state.

26





Data Generation – Example Failure Parameters

Component NO	Component Name	Failure Rate	Example Parameter (per hour)	Coverage	Example Parameter
Component 1.1(A/B)	Power Level Sensor	λ_{11}	1×10^{-6}	C_{11}	0.99
Component 1.2(A/B)	Steam Flow Sensor	λ_{12}	1×10^{-6}	C_{12}	0.99
Component 1.3(A/B)	Water Flow Sensor	λ_{13}	1.5×10^{-6}	C_{13}	0.99
Component 1.4(A/B)	Water Temp Sensor	λ_{14}	1×10^{-6}	C_{14}	0.99
Component 1.5	Water Level Sensor	λ_{15}	1×10^{-6}	C_{15}	0.99
Component 2	Main Controller	λ_2	3.65×10^{-5}	C_2	0.995
Component 3	Backup Controller	λ_3	3.65×10^{-5}	C_3	0.995
Component 4	Main Flow Valve PID	λ_4	1×10^{-6}	C_4	0.995
Component 5	Bypass Flow Valve PID	λ_5	1×10^{-6}	C_5	0.995
Component 6	Spare PID	λ_6	1×10^{-6}	C_6	0.995
Component 7	Main Flow Valve	λ_7	1.2×10^{-6}	C_7	0.99
Component 8	Bypass Flow Valve	λ_8	1×10^{-6}	C_8	0.99
Component 9	Feed-Water Pump PID	λ_9	1×10^{-6}	C_9	0.995
Component 10	Feed-Water Pump	λ_{10}	1×10^{-6}	C_{10}	0.99

29



Example PRA Model

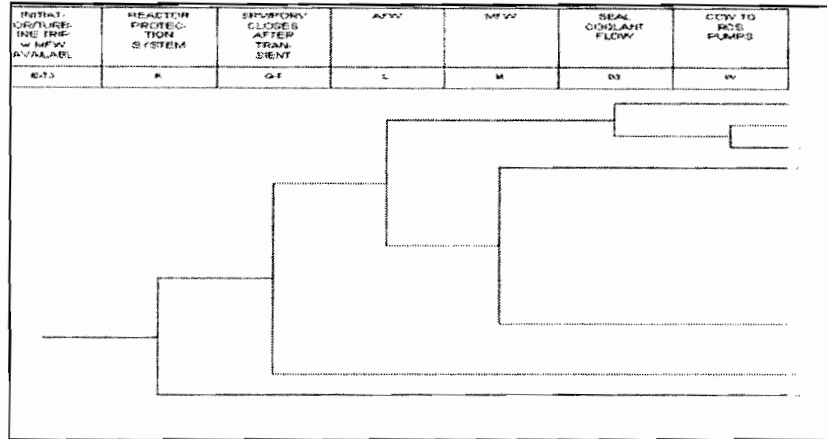
- A 3-loop design with each unit rated at 2441 MW_{th} or 788 MW_e
- The PRA model used is based on NUREG-1150 and constructed using SAPHIRE.
- The benchmark system is assumed to be applicable to each loop.*

*While the benchmark system is based on a 2-loop design, this assumption is necessitated by: a) availability of a documentation on digital feedwater control systems, and, b) accessibility of available PRA models

30



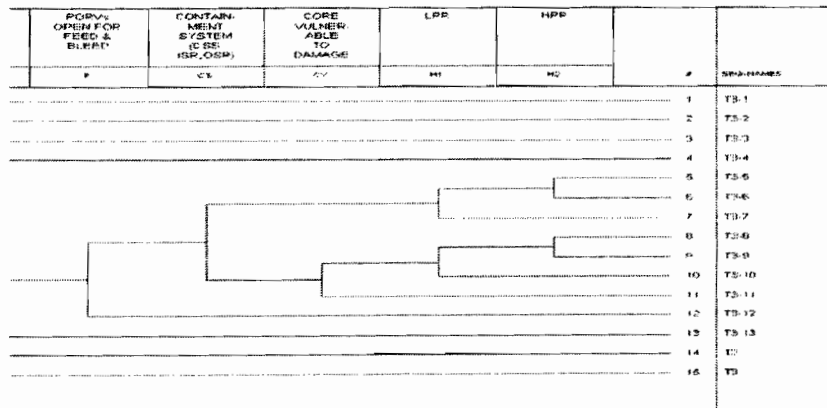
Example PRA Model – Turbine Trip Event Tree (1/2)



31



Example PRA Model – Turbine Trip Event Tree (2/2)



32



DFM - Background

- Developed by ASCA, Inc. in the 1990s as a software tool to support Probabilistic Risk Assessment (PRA)
- Software was used in the safety analysis of several software controlled systems. The results validated DFM's ability to handle software & hardware interactions and to perform dynamic analysis
 - Digital feedwater control system in an advanced Pressurized Water Reactor (NUREG/CR 6465 – April 1996)
 - Control system for the Combustion Module-1 System (NASA Glenn Research Center Shuttle Experiment)

33



DFM – Features (1/2)

- Graphic modeling environment and automated analysis engine that can handle
 - cause-effect relationships
 - time-dependent relationships
 - feedback loops
- Discretized state-vectors represent key process parameters
- Mapping between the discretized state-vectors governed by multi-valued logic rules
 - decision tables
 - transfer-boxes
 - transition-boxes

34



DFM - Features (2/2)

- A DFM model can be analyzed
 - inductively (i.e., in forward-tracking / discrete-event-simulation mode) to verify intended behavior and/or to track the effects of possible combinations of component failures on overall system operation / behavior
 - deductively to determine all possible combinations of basic causes leading to any system event which can be represented in terms of the modeled process variables. This is equivalent to developing dynamic fault trees
- The single system DFM model can be interrogated in many ways:
 - Deductively to analyze a large number of top events
 - Inductively to simulate the sequences from many different initial conditions
- In the deductive mode, current software identifies the prime implicants. Prime implicants are the multi-valued logic equivalent of minimal cut sets in fault tree analysis

35



DFM - Quantification

- In a deductive analysis, the top event can be quantified from the probabilities of the basic events that make up the prime implicants
- The set of prime implicants is first converted to a logically equivalent set of mutually exclusive implicants
 - This process is the multi-valued logic equivalent of the Binary Decision Diagram (BDD) procedure for solving fault-trees
- The top event probability is obtained as the sum of the probabilities of the mutually exclusively implicants
- The quantification results are compatible with standard PRA software formats (e.g., SAPHIRE)
 - The top event probability and/or the set of mutually exclusive implicants (with probabilities) can be exported onto SAPHIRE event-tree and/or fault-tree structures

36



Basic Steps in a Typical DFM Analysis

- Step 1: Model construction
 - Construct DFM model of system of interest
 - Representing the system behavior and flow of causality
 - Model is a network of nodes, transfer-boxes, transition-boxes and associated arc connections
- Step 2: System Analysis
 - Use DFM inductive and deductive engines to:
 - Verify specified system behavior (can be done on system "design model"), and/or,
 - Systematically identify causal links between system failure modes and basic component failure modes (Automated FMEA and/or identification of prime implicants for system failure "Top-Events" of interest), and/or,
 - Define test sequences specifically suited to identify and isolate various classes of possible faults. This feature is especially useful for generating input vectors for testing software based systems 37



Uses of DFM Analyses

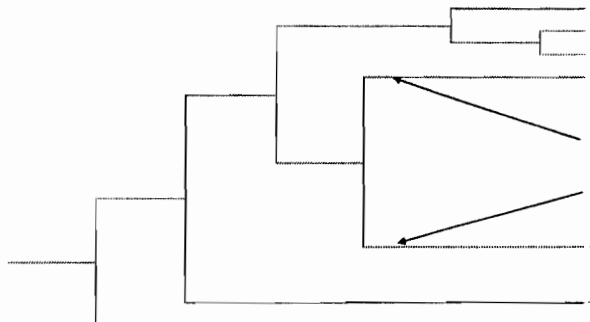
- Deductive and inductive procedures can be combined to carry out 3 types of analyses.
 - System Verification
 - Using mostly the inductive procedure, check that the system will behave as it is supposed to under different initial and boundary conditions
 - Failure and Fault Analysis
 - Automated Failure Modes and Effects Analysis (FMEA)
 - Use inductive analysis to propagate of basic component failure combinations to identify consequences at the system level
 - Prime Implicants
 - Use deductive analysis to identify combinations of component failure modes and software conditions that could cause an undesirable system event to occur
 - Test Sequences
 - Identify test patterns to prove or disprove the presence of specific types of faults in the **actual** software modules
 - An extension of the procedure used in testing of binary circuits



Example of DFM Supported Risk Assessment

From the Event Tree model in the master PRA, identify the pivotal event that needs to be analyzed by DFM

INITIAL OPERABLE TRIP - MFW AVAILABLE	REACTOR PROTECTION SYSTEM	SPRINTOR CUESSES AFTER TRANSIENT	AFW	MFW	SEAL COOLANT FLOW	CCW TO RCS PUMPS
E-17	X	G.T	L	H	M	W



Analyze the digital feedwater control system with DFM to find the prime implicants for these 2 branches

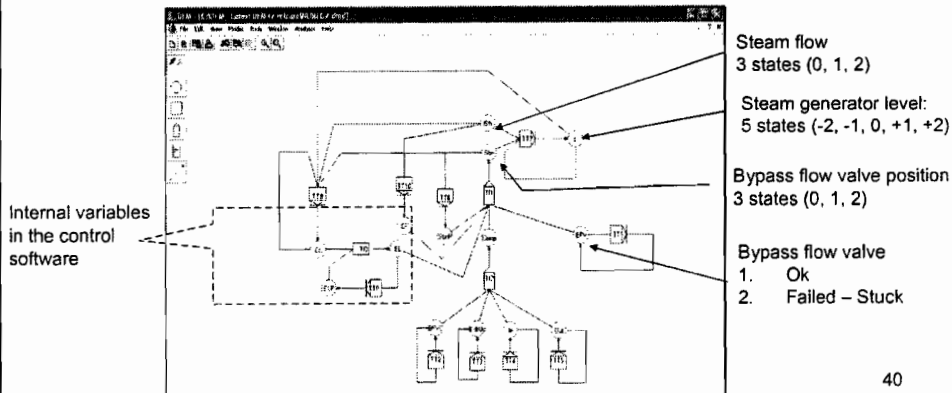
39



Example Initiating Event DFM Model

Construct a DFM model to represent the causality flow of the example initiating event:

- Discretized DFM nodes represent key process parameters.
- Transfer functions between nodes expressed as decision tables.



40



Example Initiating Event DFM Analysis

- Use DFM to determine the prime implicants for the top events:
 - Steam generator high level
 - Steam generator low level
- The Top Events were defined as a conjunction of the node states at different time steps.
 - The SG high level top event was defined as:

$L = 2 @ t = 0 \wedge$
 $L = 1 @ t = -1 \wedge$
 $L = 0 @ t = -2 \wedge$
 $ELP = 0 @ t = -2 \wedge$
 $CL = 0 @ t = -2$

41



DFM - Prime Implicants for SG High Level (1/2)

- The SG high level top event was analyzed deductively for 2 time steps
- 11 prime implicants were identified
- The "BFV fails stuck at 44 s condition that leads to high SG level" is a subset of the initial condition identified in Prime Implicant #5

1	$L = 0 @ t = -2 \wedge$ $ELP = 0 @ t = -2 \wedge$ $CL = 0 @ t = -2 \wedge$ $SbSP = 2 @ t = -2 \wedge$ $SN = 1 @ t = -2 \wedge$ $Out = Loss @ t = -2$	$Level is normal @ t = -2$ $Level is not high normal @ t = -2$ $Compressed level is normal @ t = -2$ $Feed Flow is Steam flow @ t = -2$ $Loss of Outlets @ t = -2$
2	$L = 0 @ t = -2 \wedge$ $ELP = 0 @ t = -2 \wedge$ $CL = 0 @ t = -2 \wedge$ $SbSP = 2 @ t = -2 \wedge$ $SN = 1 @ t = -2 \wedge$ $BFVC = Failed @ t = -2$	$Level is normal @ t = -2$ $Level is not high normal @ t = -2$ $Compressed level is normal @ t = -2$ $Feed Flow is Steam flow @ t = -2$ $Excess flow valve control is ok @ t = -2$
3	$L = 0 @ t = -2 \wedge$ $ELP = 0 @ t = -2 \wedge$ $CL = 0 @ t = -2 \wedge$ $SbSP = 2 @ t = -2 \wedge$ $SN = 1 @ t = -2 \wedge$ $Backup = Down @ t = -2$	$Level is normal @ t = -2$ $Level is not high normal @ t = -2$ $Compressed level is normal @ t = -2$ $Feed Flow is Steam flow @ t = -2$ $Backup complete is ok @ t = -2$
4	$L = 0 @ t = -2 \wedge$ $ELP = 0 @ t = -2 \wedge$ $CL = 0 @ t = -2 \wedge$ $SbSP = 2 @ t = -2 \wedge$ $SN = 1 @ t = -2 \wedge$ $In = Loss @ t = -2$	$Level is normal @ t = -2$ $Level is not high normal @ t = -2$ $Compressed level is normal @ t = -2$ $Feed Flow is Steam flow @ t = -2$ $Loss of Inlets @ t = -2$
5	$L = 0 @ t = -2 \wedge$ $ELP = 0 @ t = -2 \wedge$ $CL = 0 @ t = -2 \wedge$ $SbSP = 2 @ t = -2 \wedge$ $SN = 1 @ t = -2 \wedge$ $BFV = F.S @ t = -2$	$Level is normal @ t = -2$ $Level is not high normal @ t = -2$ $Compressed level is normal @ t = -2$ $Feed Flow is Steam flow @ t = -2$ $Excess flow valve is ok @ t = -2$

42



DFM - Prime Implicants for SG High Level (2/2)

6	L = 0 @ 1 = -2 A ELP = 0 @ 1 = -2 A CL = 0 @ 1 = -2 A SbnP = 1 @ 1 = -2 A ISN = 0 @ 1 = -2 A Out = Loss @ 1 = -2	Level is normal @ 1 = -2 Level error is nominal @ 1 = -2 Compensated level is nominal @ 1 = -2 Feed flow > Steam flow @ 1 = -2 Loss of outputs @ 1 = -2
7	L = 0 @ 1 = -2 A ELP = 0 @ 1 = -2 A CL = 0 @ 1 = -2 A SbnP = 1 @ 1 = -2 A ISN = 0 @ 1 = -2 A BFVC = Failed @ 1 = -2	Level is normal @ 1 = -2 Level error is nominal @ 1 = -2 Compensated level is nominal @ 1 = -2 Feed flow > Steam flow @ 1 = -2 Bypass flow valve controller failed @ 1 = -2
8	L = 0 @ 1 = -2 A ELP = 0 @ 1 = -2 A CL = 0 @ 1 = -2 A SbnP = 1 @ 1 = -2 A ISN = 0 @ 1 = -2 A BkUp = Down @ 1 = -2	Level is normal @ 1 = -2 Level error is nominal @ 1 = -2 Compensated level is nominal @ 1 = -2 Feed flow > Steam flow @ 1 = -2 Backup computer is down @ 1 = -2
9	L = 0 @ 1 = -2 A ELP = 0 @ 1 = -2 A CL = 0 @ 1 = -2 A SbnP = 1 @ 1 = -2 A ISN = 0 @ 1 = -2 A In = Loss @ 1 = -2	Level is normal @ 1 = -2 Level error is nominal @ 1 = -2 Compensated level is nominal @ 1 = -2 Feed flow > Steam flow @ 1 = -2 Loss of inputs @ 1 = -2
10	L = 0 @ 1 = -2 A ELP = 0 @ 1 = -2 A CL = 0 @ 1 = -2 A SbnP = 1 @ 1 = -2 A ISN = 0 @ 1 = -2 A BFV = F-S @ 1 = -2	Level is normal @ 1 = -2 Level error is nominal @ 1 = -2 Compensated level is nominal @ 1 = -2 Feed flow > Steam flow @ 1 = -2 Bypass flow valve failed stuck @ 1 = -2
11	L = 0 @ 1 = -2 A ELP = 0 @ 1 = -2 A CL = 0 @ 1 = -2 A SbnP = 2 @ 1 = -2 A ISN = 0 @ 1 = -2 A BkUp = OK @ 1 = -2 A In = OK @ 1 = -2 A BFVC = OK @ 1 = -2 A Out = OK @ 1 = -2 A BFV = F-S @ 1 = -2	Level is normal @ 1 = -2 Level error is nominal @ 1 = -2 Compensated level is nominal @ 1 = -2 Feed flow > Steam flow @ 1 = -2 Backup computer is OK @ 1 = -2 Input is OK @ 1 = -2 Bypass flow valve controller is OK @ 1 = -2 Output is OK @ 1 = -2 Bypass flow valve is OK @ 1 = -2

43



DFM - Prime Implicants for SG Low Level

- The SG low level top event was analyzed deductively for 2 time steps
- 11 prime implicants were identified
 - 10 prime implicants correspond to steam flow > feed flow and the one of the following failures:
 - Loss of outputs, OR
 - Bypass flow valve controller failure, OR
 - Backup computer failure, OR
 - Loss of inputs, OR
 - Bypass flow valve failed stuck
 - The "BFV fails stuck at 43 s condition that leads to low SG level" is a subset of the initial condition identified in this Prime Implicant
 - 1 prime implicant corresponds to steam flow >> feed flow such that the controller is not able to correct the mismatch fast enough the prevent the SG level from dropping to the very low level

44



Markov Methodology

1. Define Top Events
2. Partition the state space or the controlled variable state space (CVSS) into computational cells
3. Determine the system hardware/software/ firmware configurations
4. Determine the cell-to-cell transition probabilities
5. Determine the component state transition probabilities
6. Determine the pdf and Cdf for the Top Events and s-importance of component state configurations to the Top Events

45



Markov Methodology – Step 1

The controller is regarded as failed if water level in SGn rises above +30 inches and falls below -24 inches. Subsequently, there are two Top Events:

1. $x_n < -24$ inches (Low Level), and,
2. $x_n > +30$ inches (High Level).

46



Markov Methodology – Step 2 (for an example turbine trip with main computer failed)

The relevant system equations are

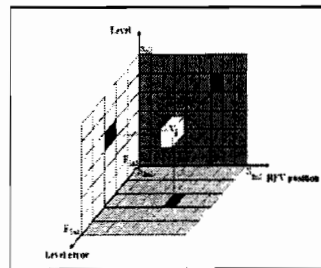
$$\frac{dx_{1n}}{dt} = \frac{0.001 \tilde{S}_{1n} - 1.268 \tilde{S}_{1n} - 1.2019}{109} + \frac{0.066 * 21.028}{10 - t} \tilde{S}_{1n}$$

$$r_2 = \frac{dC_{1n}}{dt} = C_{1n} - x_{1n}(t) - r_1(0.0014 \tilde{S}_{1n} - 1.268 \tilde{S}_{1n} - 1.2019) + \frac{0.066 * 33.05 r_1}{(10 - t)^{0.2}}$$

$$-r_2 \frac{dE_{1n}}{dt} = C_{1n}$$

$$\tilde{S}_{1n}(t) = 1 - \left[0.066 * 1500 e^{-t/10} + 0.066 * 1500 \frac{(1 - r_2)}{r_2} \int_0^t \frac{e^{-(10-u)/10} du}{(10 - u)^{0.2}} \right] + 1200 E_{1n}(t) \quad (54)$$

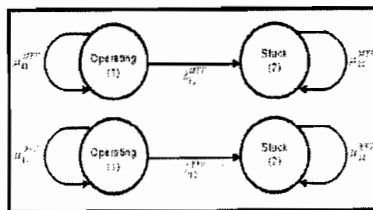
A corresponding CVSS partitioning scheme is



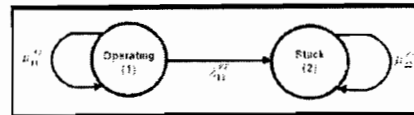
47



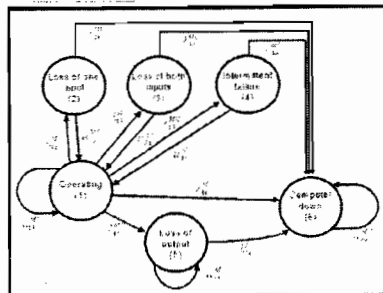
Markov Methodology – Step 3 (1/2) (for some benchmark system example components)



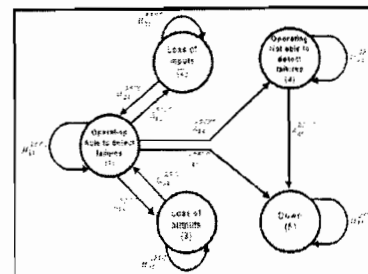
Failure States for the MFV and BFV



Failure States for the FP



Failure state for the Main Computer

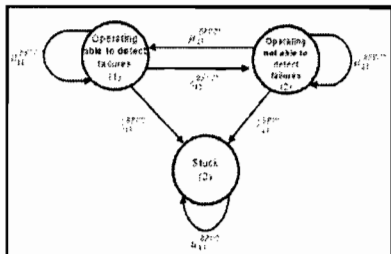


Failure States for the BFV Controller

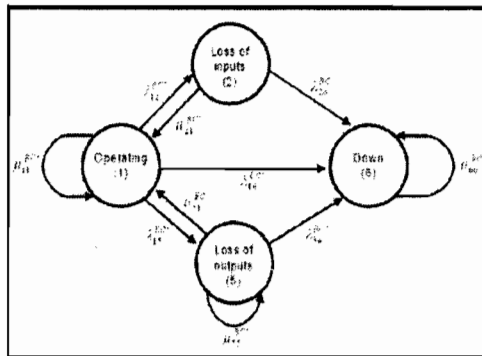
48



Markov Methodology – Step 3 (2/2) (for an example turbine trip with main computer failed)



Combined BFV and Controller



Backup Computer

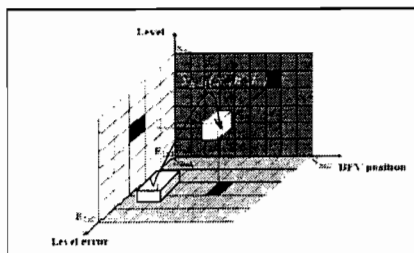


Markov Methodology – Step 4 (1/2)

Transition probability from cell j' to j for system configuration n' can be found from

$$g(j | j', n', k) = \frac{1}{V_{j'} V_j} \int dx e_j[\tilde{x}(x', n', k)]$$

$$e_j(y) = \begin{cases} 1 & \text{if } y \in V_j \\ 0 & \text{otherwise} \end{cases}$$





Markov Methodology – Step 4 (2/2)

Level
 Level Error
 Compensated Level
 BFV Aperture

BFV	CPU	From To	0-0-0-0	1-0-0-0	2-0-0-0	3-0-0-0	4-0-0-0	1-1-0-0	2-1-0-0	3-1-0-0	4-1-0-0	0-0-1-0	1-0-1-0	2-0-1-0	3-0-1-0
OK/ABLE	OK	0-0-0-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	1-0-0-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	2-0-0-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	3-0-0-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	4-0-0-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	1-1-0-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	2-1-0-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	3-1-0-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	4-1-0-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	0-0-1-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	1-0-1-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	2-0-1-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	3-0-1-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	4-0-1-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	1-1-1-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	2-1-1-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	3-1-1-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	4-1-1-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	0-0-2-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	1-0-2-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	2-0-2-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	3-0-2-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	4-0-2-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	1-1-2-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	2-1-2-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	3-1-2-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	4-1-2-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	0-0-3-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	1-0-3-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	2-0-3-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	3-0-3-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	4-0-3-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	1-1-3-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	2-1-3-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	3-1-3-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OK/ABLE	OK	4-1-3-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

A small portion of the overall matrix which contains the elements $g(n',j',k)$ for an example turbine trip with main computer failed. The first two columns define the components state combination n' while the third one defines the cell V_j' . The first row represents the cell V_j' and is represented as an array of four elements corresponding to level, level error, compensated level and BFV position, respectively.

51



Markov Methodology – Step 5

Transition probability $h(n|n',j' \rightarrow j, \Delta t)$ from configuration n' to n given that the controlled variables move from cell j' to j can be found from the control laws and component failure modes. Table below shows the $h(n|n',j' \rightarrow j, \Delta t)$ for an example turbine trip with main computer failed

$n' n$	1	2	3	4	5	6	7	8	9	10	11	12
1	$\mu_{1,1} \Delta t$	$\lambda_{1,2} \Delta t$	$\lambda_{1,3} \Delta t$	$\lambda_{1,4} \Delta t$	$\lambda_{1,5} \Delta t$	$\lambda_{1,6} \Delta t$	$\lambda_{1,7} \Delta t$	$\lambda_{1,8} \Delta t$	$\lambda_{1,9} \Delta t$	$\lambda_{1,10} \Delta t$	$\lambda_{1,11} \Delta t$	$\lambda_{1,12} \Delta t$
2	$\lambda_{2,1} \Delta t$	0	0	$\lambda_{2,4} \Delta t$	$\lambda_{2,5} \Delta t$	C	0	$\lambda_{2,8} \Delta t$	$\lambda_{2,9} \Delta t$	0	0	$\lambda_{2,12} \Delta t$
3	0	0	0	0	0	C	0	0	$\lambda_{3,9} \Delta t$	0	$\lambda_{3,11} \Delta t$	$\lambda_{3,12} \Delta t$
4	0	0	0	0	0	0	0	0	0	0	0	$\lambda_{4,12} \Delta t$
5	$\lambda_{5,1} \Delta t$	$\lambda_{5,2} \Delta t$	$\lambda_{5,3} \Delta t$	$\lambda_{5,4} \Delta t$	$\lambda_{5,5} \Delta t$	$\lambda_{5,6} \Delta t$	$\lambda_{5,7} \Delta t$	$\lambda_{5,8} \Delta t$	$\lambda_{5,9} \Delta t$	$\lambda_{5,10} \Delta t$	$\lambda_{5,11} \Delta t$	$\lambda_{5,12} \Delta t$
6	$\lambda_{6,1} \Delta t$	0	0	$\lambda_{6,4} \Delta t$	$\lambda_{6,5} \Delta t$	0	0	$\lambda_{6,8} \Delta t$	$\lambda_{6,9} \Delta t$	0	0	$\lambda_{6,12} \Delta t$
7	$\lambda_{7,1} \Delta t$	0	$\lambda_{7,3} \Delta t$	$\lambda_{7,4} \Delta t$	$\lambda_{7,5} \Delta t$	C	$\lambda_{7,7} \Delta t$	$\lambda_{7,8} \Delta t$	$\lambda_{7,9} \Delta t$	0	$\lambda_{7,11} \Delta t$	$\lambda_{7,12} \Delta t$
8	0	0	0	$\lambda_{8,4} \Delta t$	C	0	0	$\lambda_{8,8} \Delta t$	0	0	0	$\lambda_{8,12} \Delta t$
9	0	0	0	0	0	C	0	0	$\mu_{9,9} \Delta t$	$\lambda_{9,10} \Delta t$	$\lambda_{9,11} \Delta t$	$\lambda_{9,12} \Delta t$
10	0	0	0	0	0	C	0	0	$\lambda_{10,9} \Delta t$	0	0	$\lambda_{10,12} \Delta t$
11	0	0	0	0	0	0	0	0	$\lambda_{11,9} \Delta t$	0	$\mu_{11,11} \Delta t$	$\lambda_{11,12} \Delta t$
12	0	0	0	0	0	0	0	0	0	0	0	$\mu_{12,12} \Delta t$

BFV'	BC	n
OK able	OK	1
OK able	Loss of input	2
OK able	Loss of output	3
OK able	Down	4
OK not able	OK	5
OK not able	Loss of input	6
OK not able	Loss of output	7
OK not able	Down	8
Stack	OK	9
Stack	Loss of input	10
Stack	Loss of output	11
Stack	Down	12

Component State Combinations

52



Markov Methodology – Step 6

$$p_{n,j}[(k+1)\Delta t] = \sum_{j'} \sum_{n'} g(j|j',n',k)h(n \rightarrow m|j' \rightarrow j)p_{n',j'}(k\Delta t)$$

$$F_{\gamma}(k) = \sum_{n=1}^N p_{n,\gamma}(k) \quad \equiv \text{Cdf for Top Event } \gamma$$

$$w_{n,\gamma}(k) = \frac{1}{\Delta t} [p_{n,\gamma}(k+1) - p_{n,\gamma}(k)] \quad \equiv \text{pdf for Top Event } \gamma$$

$$(Im)_{n,\gamma}(k) = \frac{\sum_{n=1}^N w_{n,\gamma}(k)}{\sum_{n=1}^N w_{n,\gamma}(k)} \quad \equiv s\text{-importance of configuration } n \text{ to Top Event } \gamma$$

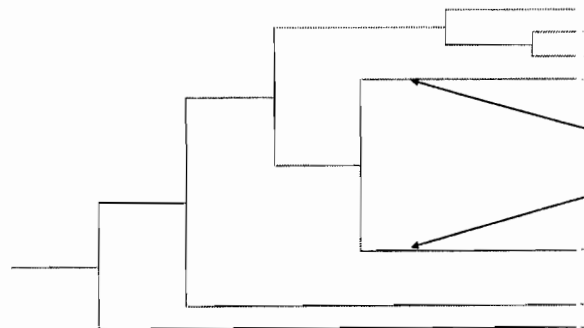
53



Incorporation of the DFM and Markov Models into PRA – DFM

The outputs from the analysis of the MFW DFM model are integrated back to the Event Tree model of the master PRA.

INITIAL OPERATIONS TRIP w/ MFW AVAILABLE	REACTOR PROTECTION SYSTEM	PRIMARY CLOSURE AFTER TRANSIENT	A/W	M/W	SEAL COOLANT FLOW	FLOW TO RES PLUMPS
e, f, g	x	o, t	l	h	oo	iv



DFM prime implicants are integrated back into the Master PRA

54



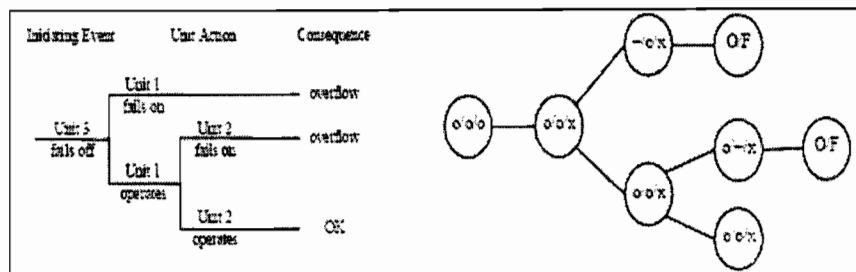
Incorporation of the DFM and Markov Models into PRA – Markov Model (1/5)

- The basic idea of this approach is to use the transition matrix of the Markov model of the system as a graph representation of a finite state machine
- With this representation and standard search algorithms it is possible to explore all possible paths to failure (scenarios) with associated probabilities and to construct dynamic event trees (DETs) of arbitrary depth.
- The DET is represented by a tree data structure. A tree data structure is composed of "nodes" –where information is stored–and "links" that connect the nodes. The nodes in the tree data structure correspond to the branching points in the DET and the links represent the branches.
- The DETs can then be incorporated into an existing PRA model through the regular features of the software that created it (e.g. SAPHIRE)

55



Incorporation of the DFM and Markov Models into PRA – Markov Model (2/5)



56



Incorporation of the DFM and Markov Models into PRA – Markov Model (3/5)

```
initialise DET root node to initial state and probability 1
add DET root node to queue Q of nodes to process
while Q is not empty
  remove next node N = (S,P) from Q
  if S is not a sink state
    for each possible state S'
      if Prob[S,S'] > 0
        compute probability P' for this branch as Prob[S,S'] * P
        if P' > epsilon
          create new node N' = (S',P')
          add N' to the list of children of N in the DET
          add N' to queue Q of nodes to process
        end if
      end if
    end for each
  end if
end while
```

Algorithm 1 to Generate DETs from Markov Model

57



Incorporation of the DFM and Markov Models into PRA – Markov Model (4/5)

```
initialise DET root node to initial state(s) and probability 1
add DET root node to queue Q of nodes to process
while Q is not empty
  remove node N = <(S1,P1),...,(Sk,Pk)> from Q
  initialise A: array [1..number of configurations] of nodes
  for each pair (S,P) in the list of pairs in N
    if S is not a sink state
      for each possible state S'
        if Prob[S,S'] > 0
          compute probability P' for this branch as Prob[S,S'] * P
          if P' > epsilon
            if S' is not in the list of states in node A[Conf(S')]
              add (S',P') to the list of states in node A[Conf(S')]
            else
              add P' to the current probability value associated with S'
              in the list of states in node A[Conf(S')]
            end if
          end if
        end if
      end for each
    end if
  end for each
  add all the nodes in A that contain at least one pair
  to the list of children of N in the DET and to queue Q
end while
```

Algorithm 2 to Generate DETs from Markov Model

58



Incorporation of the DFM and Markov Models into PRA – Markov Model (5/5)

EventTreeDisplay

Time: 5 seconds

State	Configuration	Process	Probability
1680	BFV: STUCK BUC: OK	$-2.00 \leq X_n \leq -1.00$ $-1,000.00 \leq E_{Ln} \leq -1.00$ $-500.00 \leq C_{Ln} \leq -100.00$ $0.00 \leq S_{Dn} \leq 5.00$	2.862E-7
1685	BFV: STUCK BUC: OK	$X_n < -2.00$ (LOW) $-1,000.00 \leq E_{Ln} \leq -1.00$ $-500.00 \leq C_{Ln} \leq -100.00$ $0.00 \leq S_{Dn} \leq 5.00$	1.818E-6
1736	BFV: STUCK BUC: OK	$-2.00 \leq X_n \leq -1.00$ $-1,000.00 \leq E_{Ln} \leq -1.00$ $100.00 \leq C_{Ln} \leq 100.00$ $0.00 \leq S_{Dn} \leq 5.00$	5.244E-7
1743	BFV: STUCK BUC: OK	$X_n < -2.00$ (LOW) $-1,000.00 \leq E_{Ln} \leq -1.00$ $-100.00 \leq C_{Ln} \leq 100.00$ $0.00 \leq S_{Dn} \leq 5.00$	1.563E-6
1715	BFV: STUCK BUC: OK	$2.00 \leq X_n \leq 1.00$ $4.00 \leq E_{Ln} \leq 100.00$ $-500.00 \leq C_{Ln} \leq -100.00$ $0.00 \leq S_{Dn} \leq 5.00$	4.341E-6

Graphical Interface for the Standalone Analysis of DETs

59



Interfacing with SAPHIRE - General

- SAPHIRE (Systems Analysis Programs for Hands-on Integrated Reliability Evaluations) been developed at INL with U.S. N.R.C support.
- The code was first developed by INL in the 1980's in order to create a software PRA code for personal computers.
- The first version was known IRRAS (Integrated Risk and Reliability Analysis System).
- Several modules were written to compliment IRRAS and were all integrated into a single package forming the SAPHIRE code.
- SAPHIRE uses both graphical and logic editors to construct and modify ETs and FTs

60



Interfacing with SAPHIRE - Input

Time	System Configuration	Process State	Explanation
t=0	BFV: OK/ABLE BC: OK	$-0.17 \leq x_i \leq 0.17$ $-1.00 \leq E_{ij} \leq 1.00$ $-100.00 \leq C_{ij} \leq 100.00$ $5.00 \leq S_{ij} \leq 30.00$	Both BFV and BC are in their operational state, and all process variables are in their nominal range
t=1	BFV: OK/ABLE BC: OK	$-1.08 \leq x_i \leq -0.17$ $2.00 \leq E_{ij} \leq 3.00$ $-100.00 \leq C_{ij} \leq 100.00$ $62.00 \leq S_{ij} \leq 82.00$	Level is low, BFV opens more
t=2	BFV: OK/UN-ABLE BC: LOSS-OUT	$-1.08 \leq x_i \leq -0.17$ $-1000.00 \leq E_{ij} \leq -1.00$ $-100.00 \leq C_{ij} \leq 100.00$ $0.00 \leq S_{ij} \leq 5.00$	BFV becomes unable to recognize impedance with BC and BC experiences a loss of output (which goes undetected by BFV but results in BFV closing)
t=3	BFV: OK/ABLE BC: LOSS-OUT	$-2.00 \leq x_i \leq -1.00$ $-1000.00 \leq E_{ij} \leq -1.00$ $-500.00 \leq C_{ij} \leq -100.00$ $0.00 \leq S_{ij} \leq 5.00$	Level is lower, BC experiences loss of output again, but now BFV recognizes the problem and switches to STUCK
t=4	BFV: STUCK BC: OK	$-2.00 \leq x_i \leq -1.00$ $-1000.00 \leq E_{ij} \leq -1.00$ $-500.00 \leq C_{ij} \leq -100.00$ $0.00 \leq S_{ij} \leq 5.00$	BC recovers its output ability, but that's too late
t=5	BFV: STUCK BC: OK	$x_i < -2.00$ (LOW) $-1000.00 \leq E_{ij} \leq -1.00$ $-500.00 \leq C_{ij} \leq -100.00$ $0.00 \leq S_{ij} \leq 5.00$	The level falls below the LOW setpoint and the system fails

XXXX-DEMO, DET-D0 =
 DET-D0 AND /BFV-OK-UNABLE-T0 /BC-LOSS-OUT-T0
 CONT /BFV-OK-UNABLE-T1 /BC-LOSS-OUT-T1
 CONT BFV-OK-UNABLE-T2 BC-LOSS-OUT-T2
 CONT /BFV-OK-UNABLE-T3 BC-LOSS-OUT-T3
 OUT-T4 CONT BFV-STUCK-T4 /BC-LOSS-OUT-T5
 CONT BFV-STUCK-T5 /BC-LOSS-OUT-T5

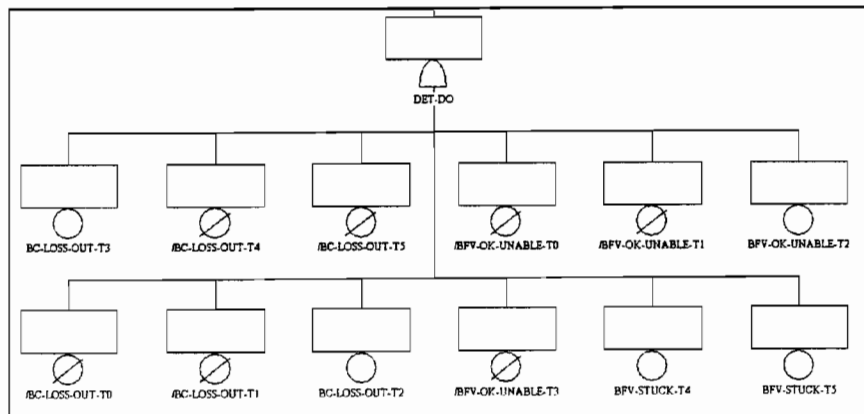
SAPHIRE Input for the Example DET

Event Sequence from an Example DET

61



Interfacing with SAPHIRE - Output



SAPHIRE Output for the Example DET

62



Interfacing with SAPHIRE – Post-Processing

1. Select the MAR-D feature under Utilities
2. Extract the desired fault tree, end state, or sequence cut sets to be exported.
3. This process will create a text file with a .FTC extension (.ESC for event tree end state cut sets, or .SQC for sequence cut sets).
4. Edit the text file to remove time inconsistencies.
5. Re-import cut sets back into SAPHIRE and then quantify using appropriate failure data.

63



Proposed Benchmark, Procedures and the Requirements for the Reliability Modeling of Digital Instrumentation and Control Systems – Benchmark Requirements*

Loosely-Control Coupled Benchmark System Requirements

1. Provides a digital system with a dock
 1. Provides information about a physical process through sampling
 2. Provides a digital system that uses the clock to perform measurements
 3. Provides a system that has roundoff
 4. Provides a system that has truncation
2. Provides explicit representation of the power requirements that are needed for the digital systems
 1. Includes loss of power
 2. Includes low power
 3. Includes power spikes
3. Provides digital systems in which there are real-time constraints
4. Provides a polling-based digital system.
 1. Events can occur in between polls.
 2. Sensors that are being polled can fail to report value
5. Provides an interrupt-driven digital system.
 1. Interrupts can occur simultaneously.
 2. Interrupts can occur at an excessive rate.
 3. There are unused interrupts that may be activated
6. Provides long term storage for a digital system
 1. Includes failures that can occur in the retrieval of information
 2. Include failures that can occur in the saving of information
 3. Include Loosely-Coupled Requirement 3
 4. Include Loosely-Coupled Requirement 2
7. Provides a digital system that computes values based on the process physics
8. Provides a self-diagnostic system
 1. Contradictory data can be delivered to the system
 2. Events can occur while in self-diagnostic mode
9. Provides a watchdog timer
 1. Instances in which there is no safe state
 2. Instances in which the watchdog timer fails

Tightly-Control Coupled Benchmark System Requirements

1. Includes Loosely-Control Coupled Requirements
2. Provides digital systems networked together
 1. Includes failures in the networked systems
 2. Includes failures in connecting components (wires, routers, etc.)
 3. Include failures of any protocol used
 4. Include failures as a result of the network topology
 5. Includes transient failures in the network
3. Provides an analog backups to digital systems that include failures in which either the digital or analog system has failed
4. Provides digital systems that share memory
 1. Includes failures which involve data races
 2. Include failures which involves both deadlocks and starvation
5. Provides digital systems that share external resources
 1. Includes failures which involves both deadlocks and starvation
 2. Includes network failures
6. Provides a digital system with fault tolerance that includes Byzantine failures
7. Provides a database for a digital system
 1. Include Loosely-Control Coupled Requirement 6.
 2. Include failures that can force the database to be inconsistent
8. Provides digital systems that have different configurations/versions of software installed on each of the systems
 1. Includes all permutations of homogeneous and heterogeneous software and/or hardware

*J. Kirichenbaum, M. Stovsky, P. Bucci, T. Aldemir, S.A. Arndt, "Benchmark Development for Comparing Digital Instrumentation and Control System Reliability Modeling Approaches", PSA 05, on CD-ROM, American Nuclear Society, LaGrange Park, IL (September 2005)

64



Proposed Benchmark, Procedures and the Requirements for the Reliability Modeling of Digital Instrumentation and Control Systems – Benchmark Compliance

- The benchmark problem satisfies most of the benchmark requirements
- It is also representative of the digital SG feedwater control systems used in operating PWRs.
- Some of the requirements are less relevant to systems use in the current nuclear reactor protection and control systems and are not represented by the benchmark system (e.g. networking, shared external resources).
- Two particularly challenging feature of the benchmark system from a reliability modeling viewpoint are the following:
 - Reliability modeling of some of its fault tolerance capabilities requires consideration of the system history
 - System failure mode may depend on the exact timing of failure events, and not just the order of failure events

65



Proposed Benchmark, Procedures and the Requirements for the Reliability Modeling of Digital Instrumentation and Control Systems – Modeling Requirements*

1. The model must be able to predict encountered and future failures well.
2. The model must account for the relevant features of the system under consideration.
3. The model must make valid and plausible assumptions.
4. The model must quantitatively be able to represent dependencies between failure events accurately.
5. The model must be designed so it is not hard for an analyst to learn the concepts and it is not be hard to implement.
6. The data used in the quantification process must be credible to a significant portion of the technical community.
7. The model must be able to differentiate between a state that fails one safety check and those that fail multiple ones.
8. The model must be able to differentiate between faults that cause function failures and intermittent failures.
9. The model must have the ability to provide relevant information to users, including cut sets, probabilities of failure and uncertainties associated with the results.
10. The methodology must be able to model the digital I&C system portions of accident scenarios to such a level of detail and completeness that non-digital I&C system portions of the scenario can be properly analyzed and practical decisions can be formulated and analyzed.
11. The model should not require highly time-dependent or continuous plant state information.

*T. Aldemir, D.W. Miller, M. P. Stovsky, J. Krachenbaum, P. Bucco, A. W. Fentiman, L. A. Mangan, Current State of Reliability Modeling Methodologies for Digital Systems and Their Acceptance Criteria for Nuclear Power Plant Assessments, NUREG/CR-6901, U. S. Nuclear Regulatory Commission, Washington, D.C. (February 2006)

66



Proposed Benchmark, Procedures and the Requirements for the Reliability Modeling of Digital Instrumentation and Control Systems – Modeling Compliance

- Neither methodology (Markov or DFM) is based on purely operating experience and both have been tested on both loosely and tightly control-coupled systems. In that respect, both methodologies predict encountered and future failures well (Requirement 1).
- Both methodologies can account for all the features of the benchmark system which is representative of the digital SG feedwater control systems used in operating PWRs as well as containing the features of digital I&C systems used in nuclear power plants, in general (Requirement 2).
- Both methodologies make valid and plausible assumptions* (Requirement 3).
- Both methodologies can quantitatively represent dependencies between failure events accurately (Requirement 4).

*For example, the assumption that process dynamics can be represented through a Markov transition matrix or a decision table (of DFM) have been validated through previous work. Similarly, the normal operation of the benchmark system and its assumed failure modes were based on operating PWRs as well as other digital I&C systems encountered in practice. Both methodologies can account for all the features of the benchmark system.

67



Proposed Benchmark, Procedures and the Requirements for the Reliability Modeling of Digital Instrumentation and Control Systems – Modeling Compliance

- Both methodologies can differentiate between a state that fails one safety check and those that fail multiple ones, as well as between faults that cause function failures and intermittent failures (Requirement 8)
- Both methodologies have the ability to provide relevant information to users, including cut sets, probabilities of failure and uncertainties associated with the results (Requirement 9).
- Both methodologies can model the digital I&C system portions of accident scenarios to such a level of detail and completeness that non-digital I&C system portions of the scenario can be properly analyzed and practical decisions can be formulated and analyzed (Requirement 10).

68



Proposed Benchmark, Procedures and the Requirements for the Reliability Modeling of Digital Instrumentation and Control Systems – Challenges

- Both methodologies have substantially steeper learning curves and are more labor intensive than the conventional ET/FT methodology (Requirement 5).
- The failure data used by either methodology for quantification are not necessarily credible to a significant portion of the technical community (Requirement 6). However, the proposed methodologies can be used to obtain qualitative information on the failure characteristics of digital I&C systems (i.e. prime implicants) as well as quantitative.
- Finally, the proposed methodologies may require highly time-dependent or continuous plant state information (Requirement 11). On the other hand, both methodologies can be also used for simple description of the connectivity between events if the correct system behavior under normal and abnormal operation can be inferred from qualitative arguments only.

69



Summary and Conclusion (1/2)

- A benchmark digital I&C system (feedwater controller of a PWR) has been specified for the assessment of the methodologies proposed for the reliability modeling of digital I&C systems using a common set of hardware/software/firmware states.
- The benchmark system specification includes procedures for system component failure mode identification and failure data acquisition.
- An example initiating event (turbine trip) has been used with the benchmark system to illustrate how the DFM and the Markov methodology can be used for the reliability modeling of digital I&C systems. These methodologies were identified by NUREG/CR-6901 as the methodologies that rank as the top two when evaluated against the requirements for the reliability modeling of digital I&C systems.

70



Summary and Conclusion (2/2)

- Both methodologies can be used to obtain qualitative as well as quantitative reliability information for digital I&C systems
- Possible challenges with the methodologies include:
 - analyst skill levels needed for the implementation of the methodologies,
 - computational demand for the correct description of the coupling between failure event,
 - acceptability of the data used for quantification by a significant portion of the technical community,
 - need for highly time-dependent or continuous plant state information for correct reliability modeling of the system failure modes if the system failure modes depend on the exact timing of the events
- Some of properties of the benchmark system considered in this first study may not apply to all the reactor protection and control systems in nuclear power plants . For digital I&C systems which may have less complex interaction between the failure events, the conventional ET/FT approach may be adequate for the reliability modeling of the system

71



Next Steps

1. A standalone reliability modeling of the full benchmark system using the DFM, Markov methodology and the conventional ET/FT approach.
2. Qualitative comparison of the event combinations that lead to the benchmark system failure as obtained by the DFM, Markov methodology and the conventional ET/FT approach
3. Quantitative evaluation of the models in Item 1 using data obtained through the fault injection procedure as well as other means (e.g. field data, data libraries)
4. Incorporation of models in Item 1 into an existing PRA for selected initiating events (e.g. turbine trip, station blackout, loss of main feedwater)
5. Specification of another benchmark problem reflecting the properties of the reactor protection system
6. Performing Items 1 through 4 for the new benchmark problem.

72

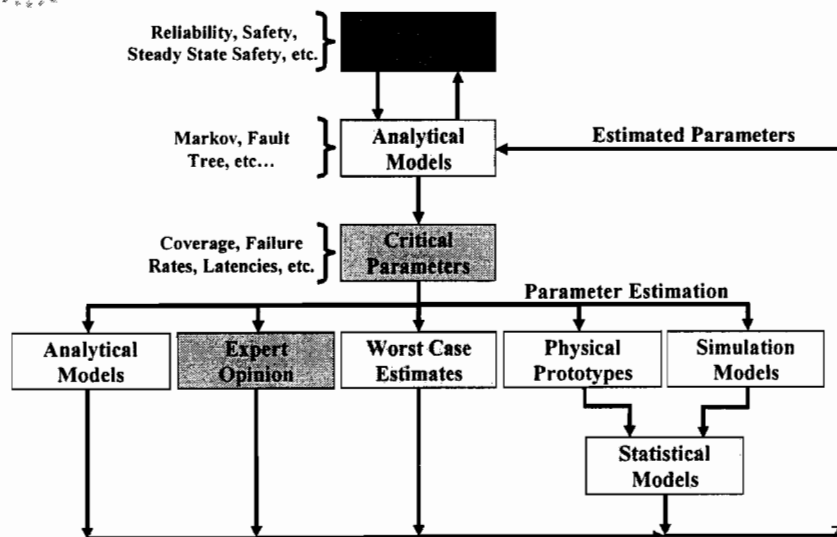


BACKUP SLIDES

73



Quantitative Safety Assessment Process





Fault Injection Methods: Collecting Critical Parameters.

- Principle nature of fault Injection:
 - *Validation technique that is based on the realization of controlled experiments where the observation of the system behavior in presence of faults, is explicitly induced by the deliberate introduction (injection) of faults into the system. Artificial faults are injected into the system and the resulting behavior is observed.*
- Tests the response behavior of the system.
 - *How effective is the system's error detection capability to a class of expected faults.*
- The Purpose of fault Injection:
 - *To uncover deficiencies, oversights, and non-compliant error detection responses of fault tolerant systems.*
- What model parameters are generated by fault injection?
 - *Fault coverage, fault latency times, reconfiguration times, system failure mode response data.*

75



Generic Fault Modeling

- In general, completely proving the sufficiency of the fault model is usually very difficult, if not impossible
- It is more traditional to assume that the fault model is sufficient, justifying this assumption to the greatest extent possible with
 - Experimental data
 - Historical data
 - Results published in literature
- To this end, UVA has developed a behavioral-level generic processor fault model, based on state-of-the-art in fault modeling literature
- Applied this generic processor fault model to the AMD486 processor architecture (benchmark system).
- Tested generic processor fault model for sufficiency via simulations.

76



Generic Fault Modeling

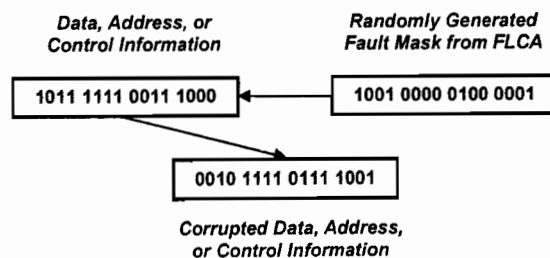
- Generic fault model based on traditional von Neumann architecture performing basic fetch-execute cycle
- Any accessible registers and memory locations can be corrupted
- Detailed fault models have been derived from the literature for
 - Register file/memory faults
 - Register selection faults
 - Program Counter (PC) faults
 - Control Unit/Instruction Decode logic faults
 - Data/address/control bus faults
 - Arithmetic and Logic Unit (ALU) faults

77



Generic Fault Modeling: Fault Injection Implementation

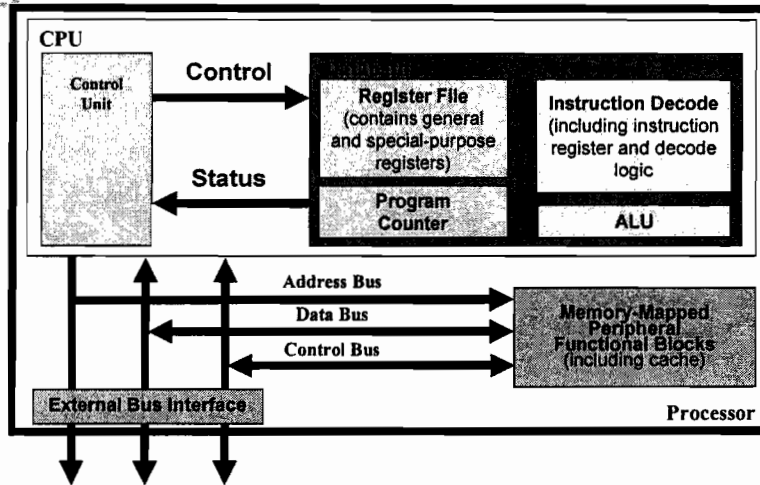
- It is shown that the fault behaviors can be represented by a random fault/error masking process



78



Generic Fault Modeling: Processor Model

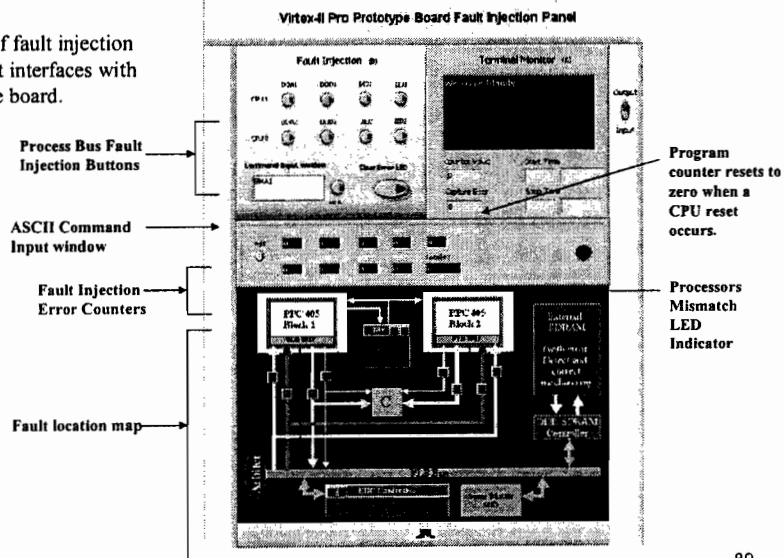


79



Preliminary Labview Fault Injection Panel

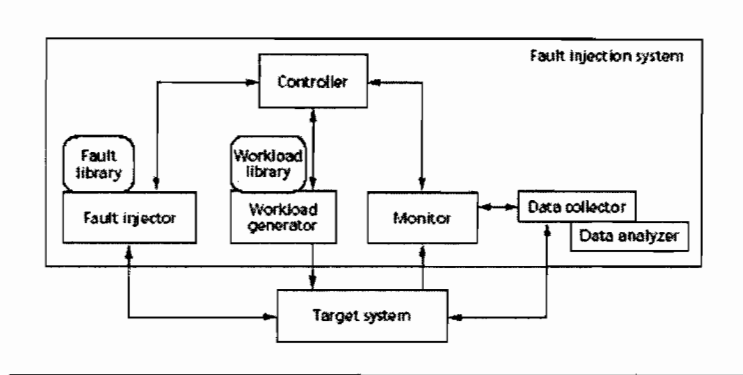
Screenshot of fault injection emulator that interfaces with the prototype board.



80



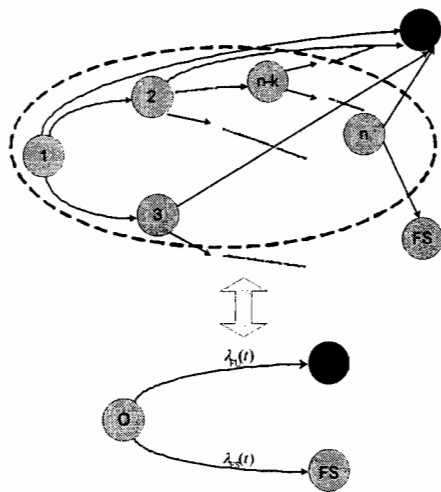
Typical Fault Injection Environment



81



Modular Markov Chain Construction



Lemma 5.2

$$\lambda_{FS}(t) = \frac{\sum_{x \in \left[\begin{array}{l} \text{a set of states which} \\ \text{have outgoing} \\ \text{transition to FS} \end{array} \right]} P_{A_x}(t) \cdot \lambda_{xFS}}{\sum_{j=1}^n P_{A_j}(t)}$$

$$\lambda_{FU}(t) = \frac{\sum_{y \in \left[\begin{array}{l} \text{a set of states which} \\ \text{have outgoing} \\ \text{transition to FU} \end{array} \right]} P_{A_y}(t) \cdot \lambda_{yFU}}{\sum_{j=1}^n P_{A_j}(t)}$$

= 82



Summary of Fault Injection Based Safety Assessment

- Compared to other SW/HW testing techniques:
 - Relatively Inexpensive.
 - Requires minimal information about the design of the HW/SW systems.
 - Makes minimal assumptions about the system operation.
 - Fault injection under complete control of the assessor.
 - Can Inject a fault at any location, for any duration of time at any time.
 - High stress testing of the SW/HW system.
- Operational profiles (system inputs) are under the control of the assessor.



Development of a Probabilistic Approach for Modeling Failures of Digital Systems Using Traditional PRA Methods

Advisory Committee on Reactor Safeguards
Subcommittee on Digital Instrumentation and Control Systems

June 27, 2006

Todd Hilsmeier

Division of Risk Assessment and Special Projects
Office of Nuclear Regulatory Research
(301-415-6788, tah1@nrc.gov)

Tsong-Lun Chu

Brookhaven National Laboratory
(631-344-2389, chu@bnl.gov)

Gerardo Martinez-Guridi

Brookhaven National Laboratory
(631-344-7907, martinez@bnl.gov)



Presentation Outline

- Background
- Project plan
- Provide status of project
- Discuss development of a failure parameter database for quantifying probabilistic failure models of the hardware of digital systems
- Review of system failure events induced by software faults to identify failure modes and mechanisms/causes of software

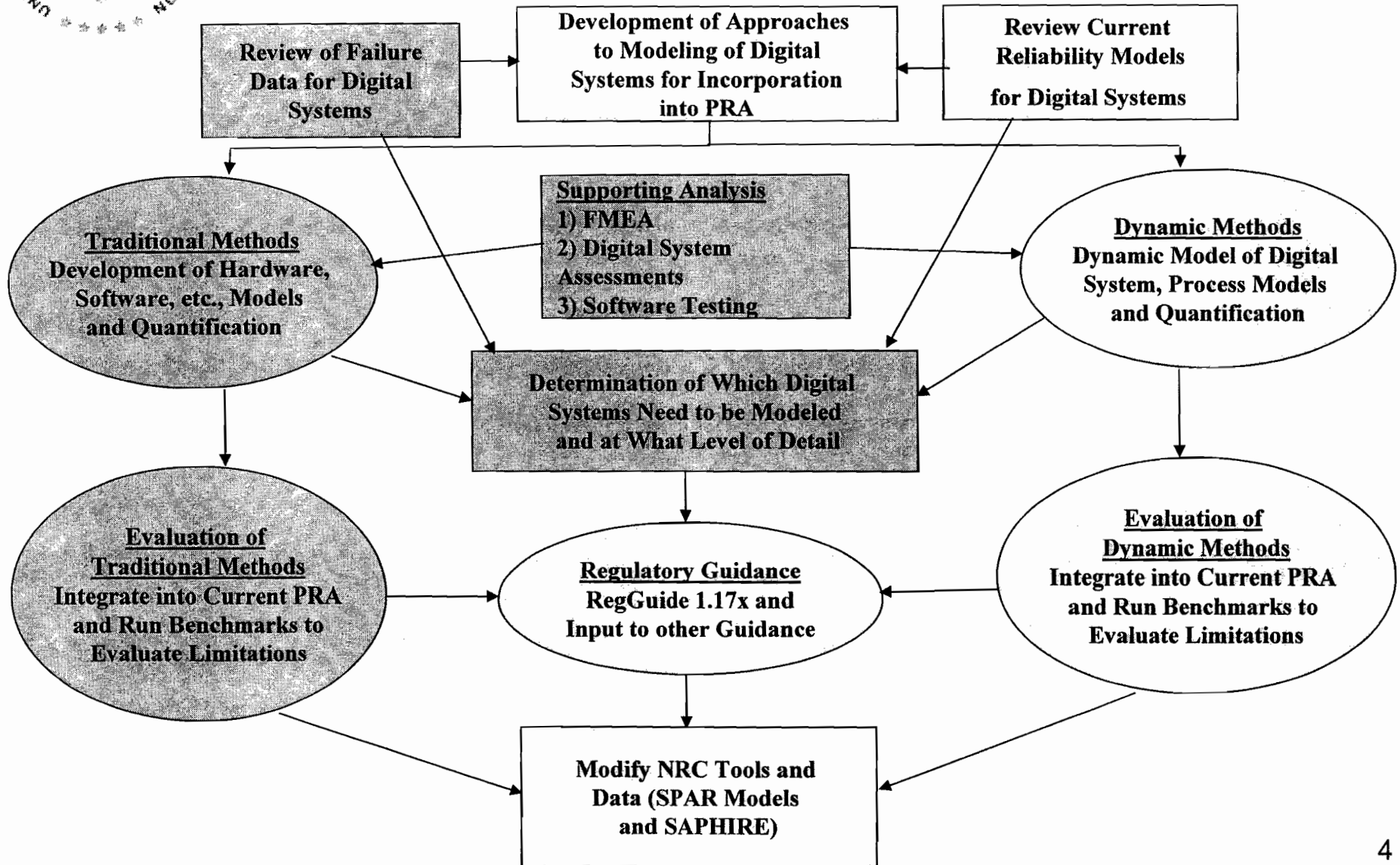


Background

- NRC has a comprehensive Digital System Research Plan that complements existing regulatory activities governing the safe and secure use of digital systems in U.S. nuclear facilities and applications
 - Includes probabilistic modeling of digital system failures using Traditional and Dynamic PRA methods that can be integrated with a PRA
 - The “Digital Systems PRA” project focuses on the use of Traditional PRA methods



NRC Digital System Risk Program





Objective of the “Digital Systems PRA” Project

- Develop a probabilistic method for modeling failures of digital systems using Traditional PRA methods (static fault trees and event trees) that can be integrated with a PRA, for those systems that do not require dynamic methods
- Provide input into Regulatory Guidance including needed modeling detail

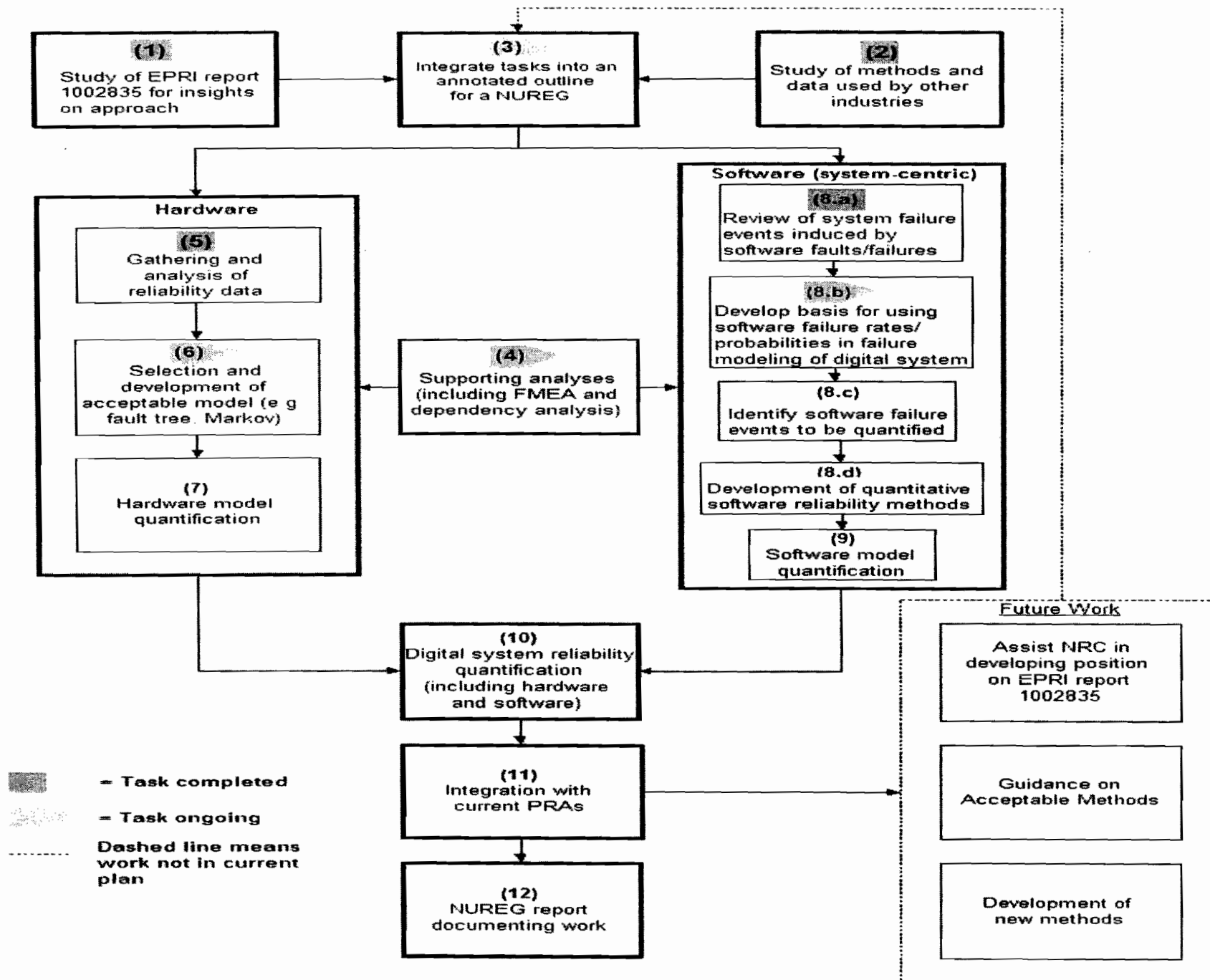


Figure 1
 Technical Tasks/Activities Associated with Digital Systems PRA Project

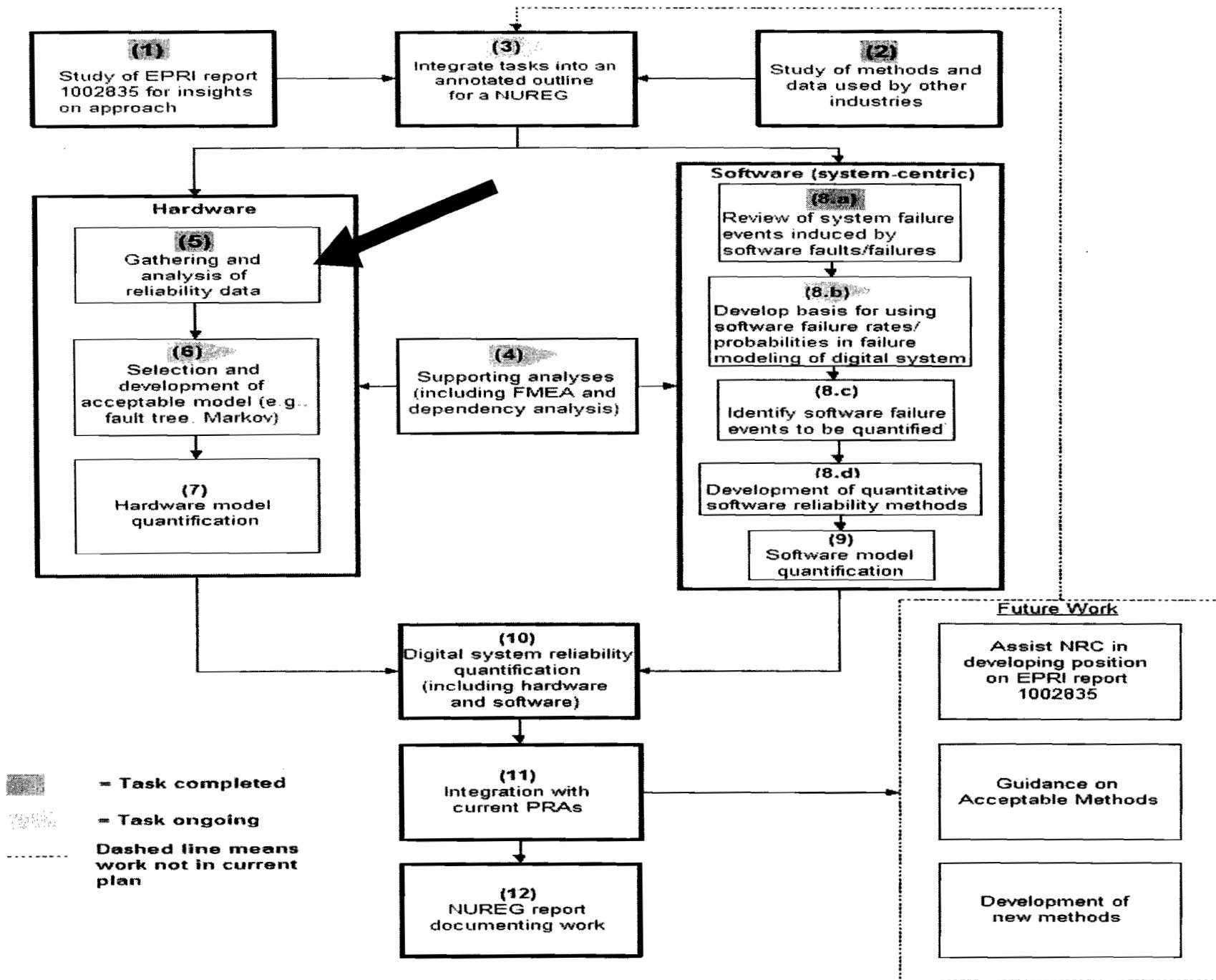


Figure 1
 Technical Tasks/Activities Associated with Digital Systems PRA Project



Development of a Failure Parameter Database for Quantifying Probabilistic Failure Models of the Hardware of Digital Systems (Task 5)

Objective:

Develop failure parameter database for digital hardware, based on currently available data, for quantifying digital system reliability models

Approach and Analysis:

Presented by Brookhaven National Laboratory

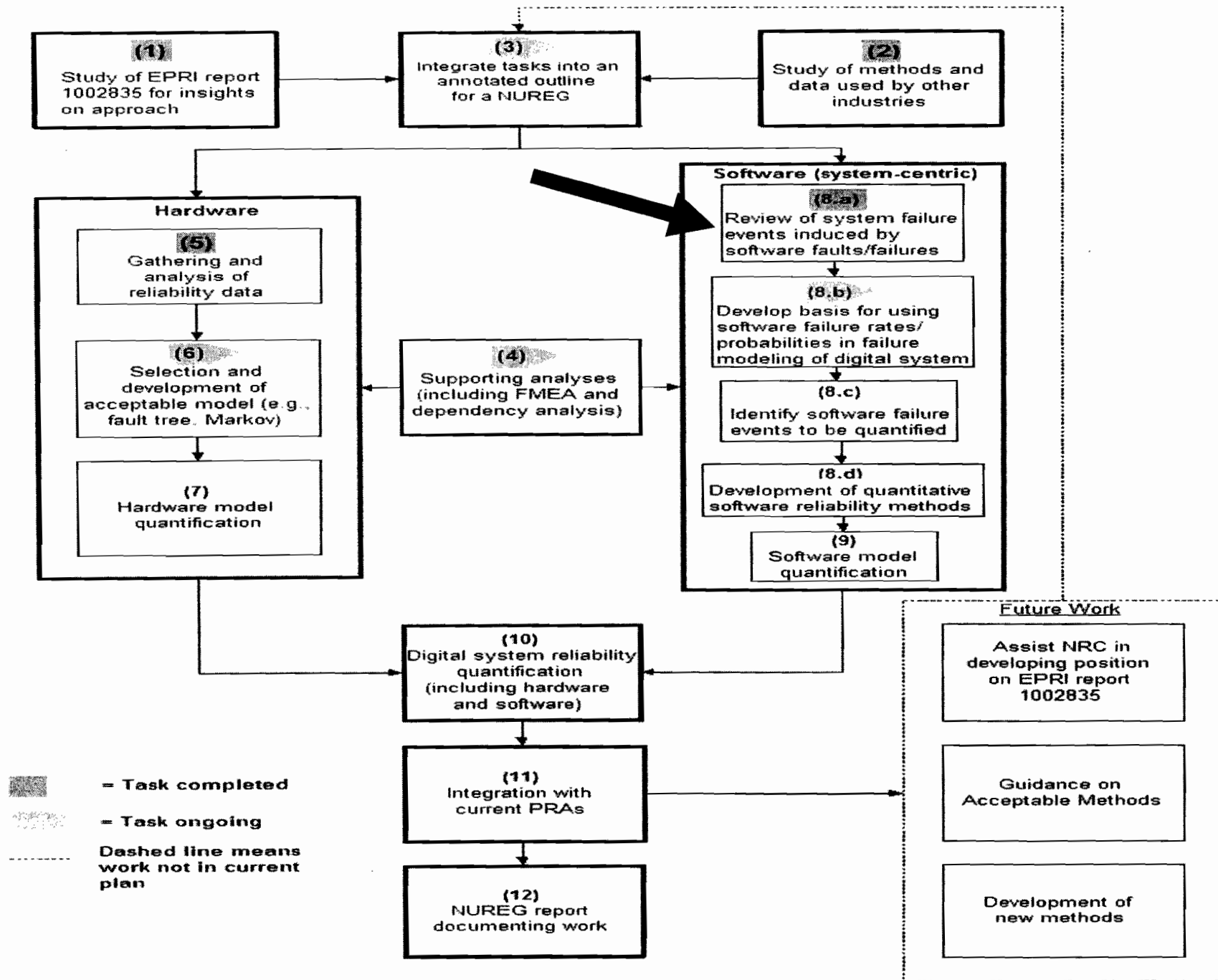


Figure 1
 Technical Tasks/Activities Associated with Digital Systems PRA Project



Review of System Failure Events Induced by Software Faults/Failures (Task 8.a)

Objective:

Review system failure events induced by software faults/failures to identify the failure modes, failure causes, occurrence frequencies, and the insights on modeling software failures in a PRA

Approach and Analysis:

A preliminary (draft) report has been completed by BNL and is currently undergoing NRC peer review

Evaluation of software-induced failure events (presented by BNL)

Development of a Failure Database for Digital System Hardware

Advisory Committee on Reactor Safeguards
Digital Instrumentation and Control Systems Subcommittee
Meeting

Rockville, MD

June 27, 2006

T. L. Chu

(631 344-2389, Chu@BNL.GOV)

Energy Sciences and Technology Department
Brookhaven National Laboratory

Outline

- Objective
- Review of failure rate databases
- Hardware reliability prediction methods
- Hierarchical Bayesian method (HBM)
- Failure rate estimates using HBM
- Conclusions
- Proposed additional data collection

Objective

Development of a generic failure parameter database of digital components, based on currently available data, in support of developing reliability models, i.e., fault tree and Markov methods, of digital systems.

Approach

- Review of reliability methods and databases
- Hierarchical Bayesian analysis of raw data extracted out of PRISM
- Proposal on additional data collection

Review of Failure Rate Databases

- Existing nuclear databases (IEEE Std 500, SPAR, T-book, ZEBD) do not contain digital component failure rates.
- Some studies (AP600, Korean Standard Nuclear Power Plant) contain scattered failure rate estimates based on proprietary data.
- Hardware reliability prediction methods (Military Handbook 217, Telcordia, PRISM) are commonly used by defense, aerospace, and telecommunication industries.
- LER database and EPIX database contain failure events subject to limitation on reporting criteria, and limited information on total demands or time in service.
- SINTEF has a data handbook supporting Markov model of IEC 61508.

Hardware Reliability Prediction Methods

- Military Handbook 217, Telcordia SR-332, and software tool PRISM developed by Reliability Analysis Center (RAC).
- Attempting to capture many causes of variability explicitly is too ambitious.
- Use of empirical formula (not laws of physics) in predicting failure rates has been found to be inaccurate.
- Applicability of empirical formula is limited to cases where good applicable failure data is available. Extrapolation could lead to significant errors.
- Lack of uncertainty consideration.

Population Variability Distributions of Digital Components Using PRISM Failure Records

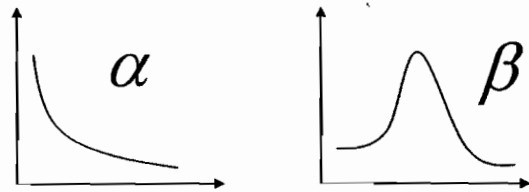
- PRISM is a software developed by the Reliability Analysis Center (RAC) for making reliability predictions of series systems, .e.g. circuit boards.
 - Failure records of components, e.g., microprocessors and RAMs, from different sources, i.e., warranty repair data, are in the form of “n failures in m hours”.
 - Large variations (see table) exist in data from different sources due to different specific designs, operating conditions, manufacturers etc.

Failure Data of A Digital Component

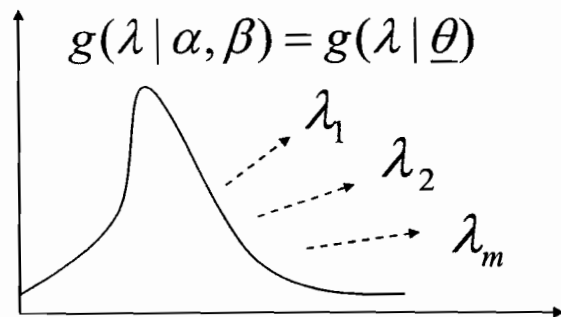
Quality	Environment	Number of Failures	Number of Hours (*1.0E6)	Point Estimate Failure Rate (per million hours)
Commercial	GB	12	633.8929	1.89e-02
Unknown	GB	0	0.2600	
Unknown	GB	0	0.0625	
Commercial	GB	16	2597.365	6.16e-03
Commercial	GM	4	701.1615	5.70e-03
Commercial	N/R	2	509.1335	3.93e-03
Commercial	GB	28	22751.18	1.23e-03
Commercial	GB	0	1105.13	
Unknown	GB	80	444.0000	1.80e-01
Unknown	GB	44	307.8874	1.43e-01
Unknown	GB	0	6.5937	
Commercial	GB	0	19.3613	
Commercial	GB	188	20069.9345	9.37e-03
Commercial	GM	1	692.6390	1.44e-03
Military	N/R	1	149.2384	6.70e-03
Military	AIF	0	0.0253	
Military	AIF	0	1.8755	
Military	AIF	0	11.3706	

Hierarchical Bayesian Method: A Illustration of Two-stage Analysis

Hyper-priors:



PVC:



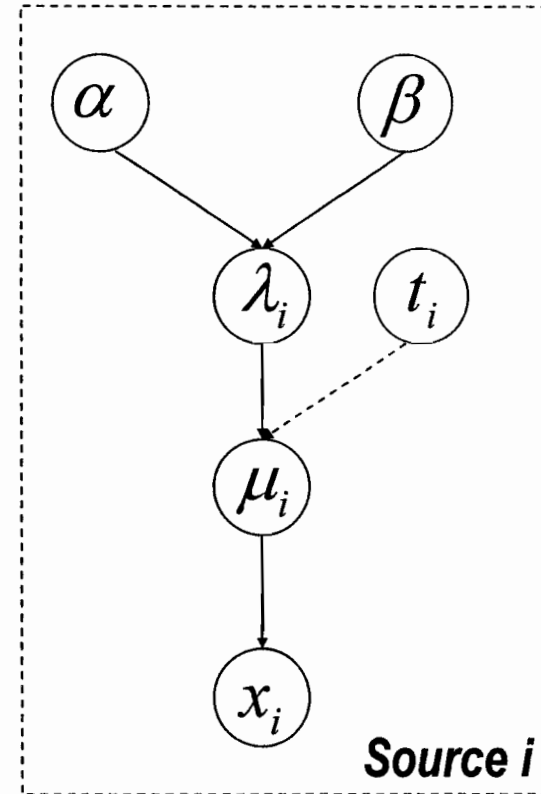
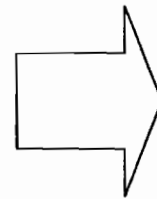
Source Specific Data:

Source 1: $(\lambda_1, t_1) \rightarrow X_1 \sim \text{Poisson}(\lambda_1, t_1)$

Source 2: $(\lambda_2, t_2) \rightarrow X_2 \sim \text{Poisson}(\lambda_2, t_2)$

...

Source m : $(\lambda_m, t_m) \rightarrow X_m \sim \text{Poisson}(\lambda_m, t_m)$



Hierarchical Bayes Analysis of PRISM Data

- 30 digital components were analyzed.
- WinBUGS software for solving hierarchical bayes models was used.
- Failure rates were assumed to be Lognormal, and Gamma distributions.
- The parameters of the distributions (hyperprior distributions) were assumed to be uniform, exponential, and normal distributed.
- Wide population variability distributions were obtained due to large variations in failure records.

Failure Rates of Gamma Distribution

- For Gamma distributed failure rates, the likelihood function
 - ✓ becomes the likelihood of a common incident rate model for large α and β
 - ✓ is improper and difficult to select hyper-priors to make the hyper-posterior proper
 - ✓ has no maximum and is asymptotically maximal along a ridge. Thus, a finite rectangle truncation of α and β can not be defined to contain most of the hyper-posterior mass, and different choices could significantly shift the region in which the population variation is localized
- Problems can be avoided using lognormal distribution

Failure Rates of Digital Components (1)

Component	Mean	5th	Median	95 th	Error Factor
Buffer	0.39	1.0E-4	1.0E-2	0.80	88
Control	0.70	4.8E-5	6.6E-3	0.98	142
Counter/Divider	9.4E-2	7.8E-6	1.7E-3	0.17	147
Decoder	7.0E-2	9.2E-4	1.7E-2	0.24	16
Encoder	3.8	2.0E-4	4.0E-2	5.6	170
EPROM	2.4E-3	1.3E-5	2.9E-4	6.7E-3	23
Error Detection/Correction Gate	13	7.1E-4	0.11	21	173
Latch	1.2E-2	1.6E-3	7.7E-3	3.6E-2	4.7
Line Bus Driver	4.6E-1	3.4E-4	2.0E-2	1.02	55
Line Bus Receiver	6.2E-2	2.2E-3	2.2E-2	2.2E-1	10
Linear Amplifier	2.1E-2	2.6E-3	1.4E-2	6.0E-2	4.8
Linear Comparator	2.0E-1	8.1E-4	2.3E-2	5.8E-1	26.8
Linear Converter	3.9E-2	6.2E-4	9.4E-3	1.4E-1	15
Linear Multiplexer	4.3E-2	9.9E-4	1.4E-2	1.5E-1	12.3
Linear Operational Amplifier	1.1E-1	1.8E-4	3.8E-4	3.4E-1	43.5
Linear Timer	1.4E-1	5.3E-3	3.6E-2	4.4E-1	9.1
Linear Voltage Regulator	4.1E-02	1.8E-3	1.7E-2	1.4E-1	8.8

Failure Rates of Digital Components (2)

Component	Mean	5th	Median	95 th	Error Factor
Micro Controller	5.5E-2	5.1E-5	3.7E-3	1.3E-1	50
Microprocessor	3.3E-2	4.6E-4	8.5E-3	1.2E-1	16
Multiplexer	3.3E-2	1.6E-4	4.0E-3	9.6E-2	25
Optoisolator	1.0E-2	4.2E-3	3.4E-2	3.2E-1	8.7
Processing Unit	3.3	1.3E-4	4.6E-2	15	339
PROM	2.6E-2	2.3E-3	1.3E-2	6.6E-2	5.3
RAM	0.33	8.8E-5	7.2E-3	0.51	76
Receiver-Transmitter	9.2E-2	7.8E-4	1.6E-2	0.34	21
Register	6.1E-2	4.0E-4	8.3E-3	1.9E-1	22
ROM	4.0E-2	6.0E-4	8.2E-3	0.11	14
UVEPROM	0.37	4.7E-3	8.6E-02	1.2	16
Tranceiver	3.5E-2	9.4E-4	1.1E-2	1.2E-1	11

Conclusions

- A process for estimating failure rates using raw data in a Hierarchical Bayesian analysis was developed.
- Population variability curves of many components are too wide due to large variability of limited raw data.
- Estimated failure rates in published studies are scattered and based on unknown proprietary data.
- Modeling using Gamma distribution should be reconsidered.
- Better data should be collected for future work.

Proposed Additional Data Collection

- The objective is to collect better data that are more applicable to I&C components used at nuclear power plants.
- Identify contacts at equipment manufacturers, e.g., Siemens, Westinghouse, GE, Triconex, MicroMac, and Fisher and Porter, and request failure data of digital components.
- Perform LER and EPIX search to identify digital component failures, and establish contacts at the plants to obtain information on the number of the same components in use and their operating hours.
- Evaluation of SINTEF data handbook for its use in Markov analysis.
- Cooperation with NASA on data collection and analysis.

A Review of Software-Induced Failure Events in Different Industries

**Advisory Committee on Reactor Safeguards
Digital Instrumentation and Control Systems Subcommittee
Meeting**

Rockville, MD

June 27, 2006

T. L. Chu and G. Martinez-Guridi

(631 344-2389, Chu@BNL.GOV;

631 344-7907, Martinez@BNL.GOV)

**Energy Sciences and Technology Department
Brookhaven National Laboratory**

Outline

- Objective
- Approach
- A preliminary model of software failures
- Review of events at domestic nuclear power plants
- Review of events of other industries and foreign nuclear plants
- Categorization of software-induced failure events
- Description of selected events
- Discussion of ACRS comments
- Review of software reliability methods
- Conclusion

Objective

The objectives of this study are:

- to discuss software failures,
- present the approach used for collecting operational events related to these failures, and
- address ACRS comments in light of the insights gained during the review of these events.

Approach

- Search LER database for software-induced failure events at domestic nuclear power plants.
- Search for events in other industries.
- Develop a preliminary model of software failure.
- Analyze in detail selected software-induced failure events.
- Review literature of software FMEA and develop a categorization method of software failure events.
- Update earlier reviews of software reliability methods.
- Review ACRS comments.

A Preliminary Model of Software Failure

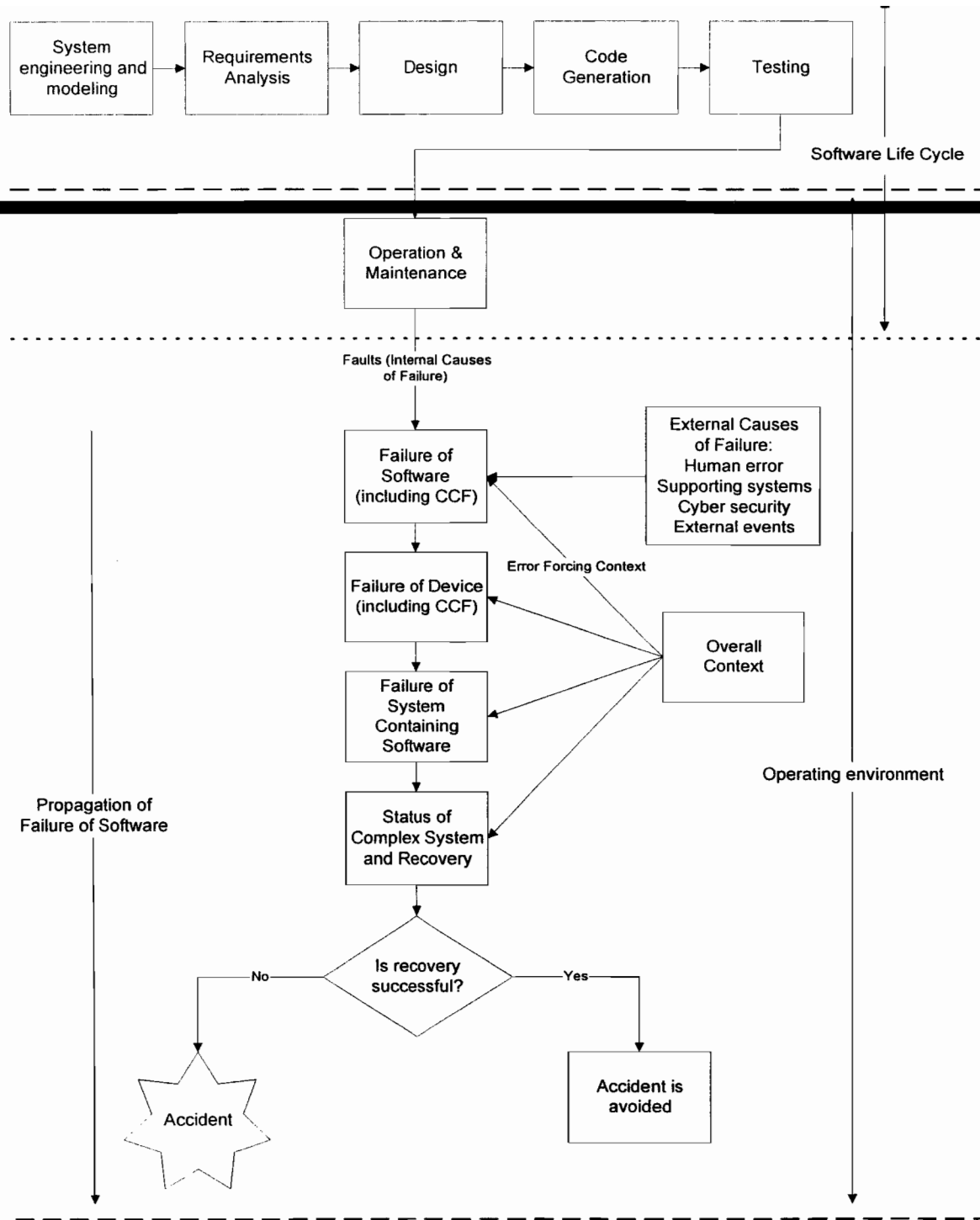
- A conceptual model of the causes of software failures, and the propagation of these failures in a complex engineered system
- The objectives are:
 - to gain a good understanding of the nature of software failures
 - To establish the basis for developing a probabilistic model of software failure (later task)
- Causes of software failures
 - Internal causes
 - External causes

Propagation of Software Failures

- In general, a software failure may be propagated to:
 - The device(s) controlled by the software (e.g., the flow control valves of the MFW),
 - The associated system
 - The overall plant
- Propagation depends on:
 - The overall context of the plant, and
 - The tolerance to failures of the design of the software, device(s), system, and the plant

Potential for Dependent Failures

- The redundant trains (or channels) of a system may use the same or similar software.
- The failure of the software means that the software in all trains fails, thus failing all trains.
- If this dependent or common-cause failure (CCF) occurs, it may cause a failure of:
 - All the device(s) controlled by the software (e.g., the flow control valves of the MFW)
 - The entire associated system



Review of Software Failures at Domestic Nuclear Power Plants

- Software failures in domestic NPPs were identified to gain insights into the nature of these failures in terms of such characteristics as:
 - The specific cause of failure of the software
 - The associated error-forcing context
 - Any dependent failures, such as common cause failures
- Identification of software failures by:
 - Using the Licensee Event Report (LER) Search System
 - 22 years were searched for software failures: from January 1, 1984 through December 31, 2005
 - All plants that operated during this period
 - All modes of operation of the plants
 - Searching for LERs containing the keyword “software” in the LER’s abstract and title
- The search was complemented with:
 - 6 additional events from Volume 2 of NUREG/CR-6734
 - We were aware of an additional event (LER 293-1997-007)

Database of Software Failures at Domestic Nuclear Power Plants

- Each LER obtained using this process was reviewed
- Those LERs documenting a software failure were selected for a database
- The current total number of LERs included in the database is 113
- Each LER is characterized in the database in terms of the following properties:
 - LER Number
 - Event Date
 - Specific nuclear unit(s) involved
 - Title of the event given by the LER
 - Description of the software failure
 - Cause(s) of the software failure
 - Consequences of the software failure
 - Error forcing context
 - Dependent failure

Insights of Review of Software Failures at Domestic NPPs

- 71 different nuclear units have at least one event related to software failure during the period studied.
 - Software failures have occurred in a significant number of units
 - This type of failure may occur in any of the operating units that use software-supported systems.
- 130 software failures in operating nuclear units are described in the 113 LERs that document software failures (i.e., 17 of the 113 LERs involved two nuclear units).

Insights of Review of Software Failures at Domestic NPPs (2)

- The 45 LERs that occurred during the last 10 years of the period stored in the database were analyzed to classify the “software failure mode” and the cause of the failure
- 31 out of the 45 events (i.e., about 69%) had the failure mode “Runs with wrong results that are not evident.”
 - This may be a reason for concern because it is undesirable to have software that is executing, sometimes for long periods of time, and producing incorrect results.
- The two main causes of failure are:
 - “Software requirements analysis” with 16 out of the 45 events (i.e., about 36%). In general, when software fails due to this cause, it fails to perform a function because when its requirements were specified, they did not include this function.
 - “Operation and maintenance” with 12 out of the 45 events (i.e., about 27%). Most of these events involve a failure introduced during modifications of the software after the software operated for some time.

Insights of Review of Software Failures at Domestic NPPs (3)

- In many cases, the EFC was identified for a particular LER.
 - In some cases a failure may occur as soon as the software becomes operational, and may remain hidden for a long time, i.e., several years. In these cases, the EFC is the normal operation of the plant.
 - The failure may be discovered by indirect means, such as discrepancies in the results produced by alternative calculations.
- In 29 of the events, i.e., about 26% of the 113 LERs, some type of dependent failure, including CCF, occurred.
 - An additional 13 LERs, i.e., about 12% of the 113 LERs, potentially involved dependent failures.
 - Hence, the potential of software failures to cause dependent failures, including CCF, is demonstrated.
 - Since a dependent failure can be significant to the risk of a NPP, a software failure has the potential to be a significant contributor to the risk.

Identification of Events of Other Industries and Foreign Nuclear Power Plants

- Internet search is the main method for identifying software-induced events.
 - “Computer Horror Stories” compiled by professor Nachum Dershowitz, School of Computer Science at Tel Aviv University,
 - “Collection of Software Bugs” compiled by professor Thomas Huckle, Institute of Information, Technical University, Munich,, Germany
 - Risks Digest compiled by Peter G. Neumann of SRI International Computer Science Laboratory
- NTSB Aviation Accident Database was reviewed.
- NASA website description of missions was reviewed.
- Other sources include news media, DOE and university websites.
- A Report written by PWR-1 Task Group on Computer-based Systems Important to Safety, NEA/CSNI/R(97)23, September 10, 1998 is the source of events at foreign nuclear plants.
- COMPSIS is developing guidelines and database structure on international operational experience.

Screening of Software-Induced Failure Events in Other Industries

- Most events were selected based on the severity of the consequences.
- Some events were selected because their failure modes (e.g., communication related failures) and causes (e.g., cyber security related events) are interesting.
- Some events were selected to cover specific industries, e.g., railway industry.
- A total of 48 events in 10 different industries were analyzed, i.e., medical service, electric power supply, commercial aviation, space, defense, telecommunication, financial service, water treatment, natural gas distribution, railway.

Categorization of Software-Induced Failure Events Based on Failure Modes and Causes

- In general, generic software failure modes are difficult to define because they depend on the level of detail at which the software is being evaluated and the specific application of the software.
- A literature review of software FMEA was performed to see how others have defined software failure modes.
- Often, failure causes, modes and effects are mixed up, probably they are used at different levels of detail.
- A categorization scheme of failure modes and causes was developed based on both the literature review and the review of software failure events.

Failure Modes of “Software System” and “Software Elements”

Software System Failure Modes (SFM)		Software Elements Failure Modes
M-I-1	SFM-1: Halt/abnormal termination with clear message	<i>Software Elements:</i> E-1: INPUT E-2: OUTPUT E-3: COMMUNICATION E-4: RESOURCE ALLOCATION E-5: PROCESSING <i>Generic Failure Modes of Software Elements:</i> <ol style="list-style-type: none"> 1. Timing/order failure, 2. Interrupt induced failure, 3. Omission of a required function or attribute, 4. Unintended function or attribute in addition to intended functions and attributes, 5. Incorrect implementation of a function or attribute, 6. Data error which cannot be identified and rejected by software logic
	SFM-2: Halt/abnormal termination without clear message	
M-I-2	SFM-3: Runs with evidently wrong results	
	SFM-4: Runs with wrong results that are not evident	
M-II	SFM-5: Problematic, confusing, or less informative interface	

Examples of Software Element Specific Failure Modes

- INPUT - Failure to interact with I/O board, excessive demand on I/O devices.
- OUTPUT- Failure to interact with I/O board, excessive demand on I/O devices, faulty message, checkpoint file failure, e.g., a file that describes status of hardware checked by operating system during the computer reboot.
- COMMUNICATION - Failed interaction (in subroutine calls, data communications) between processes, failed synchronization, dead lock (two processes prevent each other communicating)
- RESOURCE ALLOCATION - Failure to interact with CPU resources, competing for resource, priority error, resource conflict; internal capability exceeded, dead lock (two processes prevent each other obtaining resource), lockout (a process is never able to acquire the resource).

Software Failure Causes

- Software failure causes are defined in terms of errors committed during software lifecycle stages or external causes such as cyber security related, incorrect human input, support system failures, and environmental problems.
- The failure causes of the events may potentially be used to support developing quantitative software reliability methods.

Classification of Software Failure Causes

- C-I System engineering and modeling
- C-II Software requirement analysis
- C-III Software analysis and design
- C-IV Code generation
- C-V Testing
- C-VI Operation and maintenance
- C-VII External causes

Insights of Review of Software-induced Failures in Other Industries

- Software failures occur in every industry.
- Incorrect implementation and omission of functions or attributes are important failure modes.
- Errors during software requirement analysis stage are the most important failure causes.
- The occurrence of error forcing context triggering a software failure is a reasonable way of considering software failures
- Software failures may occur at a very low level which requires low level-of-detail modeling to account for their occurrence.
- Some software failures involve software that are not application software, e.g., hardware diagnostics, operating systems, and communication software.
- Software CCFs do occur.
- Man-machine interface is a contributor to some events.

Turkey Point Diesel Generator Sequencer 1994

- During a test in Unit 4, the 3A HHSI pump failed to start due to a failure in the software of the 3A sequencer. The software logic defect is limited to the test function, but the defect is common to all four sequencers.
- There was another error in the software that would preclude the automatic start of the CS pumps. The condition identified occurs when the High-High Containment Pressure (HHCP) signal is received by the sequencer during an approximate 60 millisecond (ms) time window just prior to the end of sequencer load block 3 for LOCA or LOOP coincident with LOCA events.
- System failure mode: Runs with wrong results that may not be evident.
- Element failure mode: One of the elements of the software (possibly, the processing element) incorrectly implemented some functions of the sequencer.
- Internal causes:
 - The software error causing failure of a sequencer to respond to an SI signal was introduced during the stage "System analysis and design" of the software development.
 - The cause of the error in the sequencer software that would preclude the automatic start of the CS pumps was not found in the LER. Possibly, it is the same cause.
- EFC:
 - Regarding failure of a sequencer to respond to an SI signal, in general, the EFC is the sequencer executing some tests.
 - Regarding failure of a sequencer to automatically start the CS pumps, the EFC is a HHCP signal received by the sequencer during an approximate 60 ms time window just prior to the end of sequencer load block 3 for LOCA or LOOP/LOCA events.
- Consequences:
 - The periodic inoperability of all four sequencers has existed since the sequencers were installed in 1990/1991. Since the sequencers would not have responded properly to an SI signal as designed, Units 3 and 4 were operating outside their design basis.
 - The LER considered the failure of the automatic start of the Containment Spray (CS) pumps to be not significant to safety.

Common Cause Failure of Vital 120 volt AC Buses at Pilgrim - 1997

- Pilgrim was in cold shut down. During a severe storm, the safety-related 120 volt AC buses 'A' and 'B' de-energized on two occasions.
- The cause of the de-energizing of these buses was the automatic shut downs of voltage regulating transformers X55 and X56.
- The 345 Kv system experienced brief but severe voltage transients.
- The voltage on the 480 volt load center was as low as 350 volts.
- Regulating transformers were designed to regulate input voltages of 480 volts 20 percent (384 - 576 volts).
- Each regulating transformer contains a microprocessor (MCU).
- The software contained in an MCU automatically shut down its regulating transformer if input voltage was outside the range of 384 to 576 volts.
- System failure mode: Runs with evidently wrong results.
- Element failure mode: One of the elements of the software (possibly, the processing element) of an MCU has the unintended function of shutting down the regulating transformer when the input voltage is less than 384 volts (greater than zero volts).
- Internal cause: Inadequate requirements of the software, in particular, unspecified exception conditions.
- EFC: An event, such as the severe storm, that could cause the 480 volt load center to be below 384 volts.
- Consequence: The undervoltage shut downs of the regulating transformers was outside the Pilgrim Station design basis.

Core Protection Calculators Inoperable at Palo Verde 2 - 2005

- The Core Protection Calculators (CPCs) consist of four software-supported redundant channels. The CPC system provides two trip signals to the RPS.
- When both analog input modules within a CPC channel indicate an error simultaneously, the CPC uses the last known good value. However, a channel trip should be initiated for this event. Software release 6.1 resulted in the CPCs not being able to generate this trip signal.
- System failure mode: Runs with potentially wrong results that are not evident.
- Element failure mode: There was an omission of the function that should generate the channel trip signal. One of the elements of the software (possibly, the processing element) was missing this function.
- Internal causes: The LER states that investigation into the cause of this event is ongoing, and that preliminary results indicate the direct cause is that a CPC system requirement specification was not properly translated into the CPC software by the vendor. Accordingly, it appears that the error was introduced during the development of the software, possibly during the stage of "System analysis and design."
- EFC: The simultaneous failure of both analog input modules within a CPC channel. Possibly, the EFC also includes failures of the analog sensors providing input to both analog input modules within a CPC channel.
- Consequences: All four channels of the CPCs were inoperable, and the plant operation violated Technical Specifications since the software was installed. In addition, the plant had to be shutdown from approximately 100% power.

Refueling Accident at Unit 4 of Ontario Hydro's Bruce plant 1998

- The CANDU reactors perform fueling operation while the reactor is online. A fueling machine which is moved by a bridge must lock onto each end of the fuel channel and be pressurized. The end plugs of the channel are then removed and new fuel is pushed in from one end and spent fuel is pushed out of the other end. A fueling machine can be positioned at the bridges of any reactor and be controlled by a computer system.
- A computer system which was used to control a fueling machine which is clamped to one end of a fuel channel had a previous error. The error handling routine had a fault (introduced in a software revision) which caused the return address be incorrectly set to the routine which would release the brakes on the bridge.
- When an operator trying to use the computer system to control a different bridge triggered an error which caused the software to remember the previous event and called for release of the brakes. The fueling machine moved down 40 cm and caused damage to the fuel channel fitting and a loss of D2O.
- A protective computer which would have prevented the accident was not in service.
- Software failure categorization
 - System failure mode: Software runs with wrong results that are evident
 - Element failure mode: Incorrect interrupt return
 - Failure causes: Coding error, inadequate testing subsequent to a software revision
 - A small loss of coolant accident

Discussion of ACRS Comments

- We developed a preliminary model of software failure which depicts how software failures occur, and how these failures may propagate into accidents.
- We reviewed software-induced failures in different industries, and developed a way of categorizing the events based on their failure modes and causes.
- Software failures occur because there are faults in the software and triggering events/EFC activate the faults. The occurrence of triggering events is random and can be modeled probabilistically.
- The frequency that a software failure occurs is the same as the frequency that the EFC occurs. Constant failure rate is a reasonable assumption for software failures as long as the operating conditions do not change.
- Identification of EFC is difficult.

On “System-Centric” vs “Software-Centric” Viewpoints

- The “system-centric” view point includes the interactions of the software with the rest of the plant. Conceptually, it is possible to identify the EFCs.
- Viewing software failure as a property of the software itself is incorrect. The issue is that it appears that the “software-centric” view point would only analyze the software in “isolation”. In this sense, we agree that such narrow analysis of software would fail to discover many relevant EFCs.
- Consideration of the operating environments and operational modes is an important part of the development lifecycle of a software.
- The “system-centric” view point considers and models the world around the software, while the “software-centric” view point considers the operating environments as boundary conditions of the software.
- There is no contradiction between the two viewpoints. They have different emphases.

Review of Methods on Software Reliability

- Two types of methods were reviewed, methods for identifying software faults, and methods for quantitative reliability modeling of software.
- Methods for identifying software faults – hazard analysis, FMEA, testing, formal methods, DFM.
- Methods for quantifying software reliability- reliability prediction methods, Markov model and Petri net, fault tree analysis, Bayesian belief network, reliability growth models, IEC 61508.
- A more critical review will be done in our next task.

Methods for Identifying Software Faults

- **Formal methods**

- Formal methods are mathematically based languages, techniques, and tools for specifying and verifying design requirements of hardware and software systems.
- The process of specification using these methods is the act of writing requirements down precisely. It allows a developer to gain a deeper understanding of the system specified and to discover design flaws, inconsistencies, ambiguities, and incompleteness.
- An example is the application to Traffic Collision Avoidance System II [Heimdah and Leveson 1996].
- Formal techniques such as model checking and theorem proving are also used for verification of hardware and protocols, instead of simulation models.
- Application of formal methods recognizes 1) the original requirements are usually specified in a natural language, and may be incorrect or incomplete; 2) the translation into a formal language may introduce errors; and 3) the formal model of software requirements is not the same as the source code which may contain additional faults.

Methods for Quantitative Reliability Modeling of Software

- **Bayesian Belief Networks (BBN)** are complex diagrams that organize the body of knowledge in any given area by mapping out cause-and-effect relationships among key variables and encoding them with distributions that represent the extent to which one variable is likely to affect another. Tables of conditional probabilities are used to represent the influence relationships of the nodes. Bayes' rule is used as the mechanism for updating probabilities given that additional evidence is obtained.
- Recently, BBN has been used in making prediction about software defects, determining the number of tests needed to achieve a given dependability, and assessing probability of system failure. We consider that it is possible to build a software reliability prediction model based on BBN.
- The basic idea is to set the characteristics/metrics of a software as one of the nodes, and the other nodes are factors influencing or determining the metrics. The metrics are dependent on factors that cannot be measured directly, such as the quality of the process used in its development. Expert judgment, based on observations of these factors of software, and other information such as failure data can be used to estimate the probabilities of these nodes.

Conclusions (1)

- Software failures occur in many different ways. Experience of other industries is in general applicable to the nuclear industry.
- There is no contradiction between software-centric and system-centric viewpoints. They have different emphases.
- Some failures took place in such a way that implies very detailed modeling would be required.
- Some failures involve non-application software, e.g., operating system, hardware diagnostics, and communication software. This has implication on the scope of any software analyses.
- It is reasonable to model software failures in terms of their frequencies, because the occurrence of the failure triggering events is random.
- It is possible to estimate the frequency of past software-induced accidents. The frequency represents that of historical events, and may not be useful in predicting future events.

Conclusions (2)

- Different methods can be used to identify software faults. They have different advantages and limitations. It appears that no single method is able to find all faults in a software.
- Formal methods are designed to support requirement specifications. These are promising methods deserving exploration.
- No commonly accepted method for quantitative software reliability exists.
- For safety-critical software systems, e.g., RPS, subjective judgment of experts is probably the only way to model software failures, given the current state of the art. BBN is one of such methods and its use will be further explored.



DEVELOPMENT OF REGULATORY GUIDANCE FOR RISK-INFORMING DIGITAL SYSTEM RISK REVIEW

Advisory Committee on Reactor Safeguards
Digital Instrumentation and Control Subcommittee

June 27, 2006

Steven A. Arndt

Instrumentation and Electrical Engineering Branch
Division of Fuel, Engineering & Radiological Research
Office of Nuclear Regulatory Research
(301-415-6502, saa@nrc.gov)



OVERVIEW

- As part of the overall Digital System Risk Research Program the NRC will develop needed regulatory guidance to support risk-informing digital system reviews
- To develop this guidance the NRC is working to
 - Understand the status of failure data
 - Assess which modeling methods might be usable
 - Determine which systems need to be modeled and at what level of detail
 - Develop acceptable methods
 - Develop regulatory acceptance criteria



CURRENT SITUATION

- Licensees are replacing analog systems with digital systems
- Industry has expressed interest in using risk-informed regulation (Regulatory Guide 1.174) as an alternate method for licensing these systems
- As the NRC licensees replace analog systems with digital systems, the current PRA's are not keeping up with these changes
- An NRC program to develop risk analysis tools and data is providing input into what models and methods are needed



NEED FOR GUIDANCE

- Regulatory Guide 1.174 provides guidance for risk-informed decision-making, but does not provide specific criteria for digital systems
- Because of the unique characteristics of digital systems, additional guidance needs to be provided associated with
 - Digital system modeling
 - Maintaining sufficient safety margin
 - Meeting current regulations and defense-in-depth philosophy
 - Performance measurement strategies



STRATEGY FOR DEVELOPMENT

- Develop an understanding of the characteristics of digital systems that need to be modeled (NUREG/CR-6901 and other work)
- Identify methodologies for modeling digital systems and incorporating these models into existing PRA's
- Develop an understanding of the data issues associated with digital system reliability modeling
- Develop draft regulatory guidance (DG-1151 "An Approach for Plant-Specific, Risk-Informed decision making for digital systems)
- Conduct public meetings to discuss proposed regulatory guidance (August 2006)
- Publish for comment draft regulatory guidance (December 2006)



OVERALL STRUCTURE FOR DG-1151 “AN APPROACH FOR PLANT-SPECIFIC, RISK-INFORMED DECISION MAKING FOR DIGITAL SYSTEMS”

- Modeling requirements
- Integration of digital system models with full PRA models
- Data requirements
- Uncertainty analysis
 - Model uncertainty
 - Operational profile uncertainty
 - Data uncertainty
 - Operational history
 - Testing
- Acceptance criteria
- Meeting current regulations and defense-in-depth philosophy
- Maintaining sufficient safety margin
- Performance measurement strategies



MODELING REQUIREMENTS

- The model must account for the important relevant features of the system under consideration.
- The model must make valid and plausible assumptions about system characteristics and justify these assumptions.
- The model must quantitatively be able to represent dependencies between failure events accurately, including support systems failures, common mode failures, and dynamic interactions associated with the process and digital systems, or demonstrate that they are not important
- The model must be able to differentiate between faults that cause function failures and intermittent failures; and differentiate between a state that fails one safety feature and those that fail multiple features or demonstrate that there is no important significance to the differences.
- The model must have the ability to provide relevant information to users, including cut sets, probabilities of failure and uncertainties associated with the results.
- The methodology must be able to model the digital I&C system portions of accident scenarios to such a level of detail and completeness that non-digital I&C system portions of the scenario can be properly analyzed and practical decisions can be formulated and analyzed.



LEVEL OF MODELING DETAIL

- Needs to be adequate to capture all of the unique aspects of digital systems:
 - Discrete time aspects of digital systems
 - Complex interactions between the components of the digital I&C system and between the digital I&C system and process physics which may lead to potentially significant dependencies
 - Unique failure modes of digital I&C systems
 - Digital systems environmental failure modes
 - Interaction between hardware and software that may lead to failures, including internal and external communication
 - Digital I&C systems shared data transmissions, functions, and process that may lead to common cause failure (CCF).
 - Unique characteristics of software failures and testing
 - Digital system non-continuous behavior



LEVEL OF MODELING DETAIL (CONT.)

- If simplified models are used
 - Validate that unique aspects are not important to the particular system or application
 - Validate that the data used in the simplified model captures the important aspects of the failure modes
 - Validate that common mode failures can be accounted for
 - Validate that events that have happened, can be adequately modeled at that level of modeling abstraction
- Examples will be included in DG-1151



INTEGRATION OF DIGITAL SYSTEMS MODEL WITH PRA'S

- Integration of digital system models with full PRA models
 - Needs to include all important interactions and dependencies
 - Needs to include all systems that will impact/will be impacted by the digital system changes



DATA REQUIREMENTS

- Data requirements
 - Generic Operational Data
 - LER and other nuclear data
 - Generic databases (RAC, etc.)
 - Plant/System Specific
 - Testing-Based Data
 - Needs to demonstrate applicability to delivered product
 - Needs to quantify coverage
- Data issues
 - Data collection needs to be done systematically and in a structured manner
 - Configuration control based on measures and metrics used
 - Detailed Root Cause Analysis



UNCERTAINTY ANALYSIS

- Uncertainty analysis
 - Model uncertainty
 - Operational profile uncertainty
 - Knowledge of possible input states and probability distributions
 - Data uncertainty
 - Operational history
 - Testing



ADDITIONAL REQUIREMENTS

- Acceptance criteria
 - RG-1.174
 - Additional guidance on acceptable uncertainty
- Meeting current regulations and defense-in-depth philosophy
 - 10CFR50.55a(h).
- Maintaining sufficient safety margin
- Performance measurement strategies
 - Validation of data used
 - Monitoring of industry wide events to assure assumptions continue to be valid



SUMMARY

- This research into current state of data, analysis methods, and acceptance criteria will support the development of regulatory guidance for risk-informing digital system reviews
- Broad-based program that will look at a number of potentially viable methods for developing acceptable digital system risk models
- Assess the capabilities and limitations of the state-of-the-art and develop appropriate regulatory requirements
- Regulatory guidance will be performance-based₄