UCRL-13937
APPLIED DECISION ANALYSIS, inc.
3000 Sand Hill Road, Menlo Park, CA 94025 (415) 854-7101

December 15, 1978

OVERVIEW OF THE STRUCTURED ASSESSMENT
APPROACH AND DOCUMENTATION OF
ALGORITHMS TO COMPUTE THE
PROBABILITY OF ADVERSARY DETECTION

Final Report for Tasks 5 and 6 of
LLL P.O. 9897503

Submitted by:

A handwritten signature in cursive script, appearing to read 'Thomas R. Rice'.

Thomas R. Rice and Stephen L. Derby

APPLIED DECISION ANALYSIS, INC.

CONTENTS

	Page
LIST OF FIGURES	iii
I INTRODUCTION	1
A. Project Team Approach and ADA's Role	2
B. Problem Statement	4
General Description of the Staged Approach	6
C. Advantages of the Staged Approach Over a General Purpose Computational Methodology	13
II DEFINITION OF STRUCTURED ASSESSMENT APPROACH	17
A. General Approach	19
Preliminaries	19
B. Discussion of the Staged Assessment Procedure	25
Problem Statement for Level 1	25
Problem Statement for Level 2	28
Problem Statement for Level 3	32
Problem Statement for Level 4	38
Logical Progression of Analysis Within Level 4	41
C. Comparison of Structured Assessment Approach to the Aggregated System Model	44
III CALCULATION OF THE PROBABILITY OF DETECTION	47
A. Overview of the Probability Algorithm Development	49
Logical Representation of Detection by an MC&A System	49
General Guidelines	51
Development Status	52
Basic Algorithm Design Approaches	53
Selection of Approach to Probability Algorithm	55
B. Description of the "EVENT" Probability Computer Code	57
Relation of EVENT to SAA	57
Generation of the Boolean Expressions for the Detection Event	57
EVENT, A Computer Algorithm to Calculate the Probability of Detection	59
C. Technical Definition for the Current Version of the Probability Calculation	61
Solution Algorithm for Boolean Expression	61
Solution Approach for Probability Algorithm	64
Program Limits	68
Example Application	70
D. Comparison of SAA Probability Calculation with the "IMPORTANCE" Code	75

Min Cut Set Approximations	75
Can IMPORTANCE Be Used for Level 3 Analysis?	78
IV ADA TECHNICAL MEMORANDA	85
REFERENCES	87
APPENDIX A	

FIGURES

	<u>Page</u>
1.1 Adjacency Matrix Illustration	5
1.2 General Description of Staged Approach to Determine MC&A System Effectiveness	7
2.1 Probabilistic Expansion	21
2.2 Comparison of Adversary's Decision Problem and System Effectiveness Measure	23
2.3 Level One Problem Description	26
2.4 Determination of Monitor Target Sets	27
2.5 Level Two Problem Description	29
2.6 Determination of the Level Two Probability of Detection . . .	31
2.7 Possible Interpretations of Level Three Analysis	33
2.8 Level 3B Sensitivity Analysis Procedure	36
2.9 Precedence Ordering for Sublevels within Level 4	43
2.10 Relationship to Aggregated Systems Model	45
3.1 Probability Algorithm Relationship to Structured Assessment Approach	58
3.2 Structure of Event	62
3.3 Illustration of a Unit Model	63
3.4 Conceptual Structure of Probability Algorithm	66
3.5 Example Network Description of MC&A Detection	71
3.6 Relative Error in "Min Cut Set" Bound Approximation for k of N System Unreliability	79
3.7 Event Definition for Fussell-Vesely Cut Set Importance Measure	81
3.8 Event Definition for Fussell-Vesely Component Importance Measure	82

I. INTRODUCTION

This report summarizes the work done by Applied Decision Analysis (ADA) on the development and refinement of Lawrence Livermore Laboratory's (LLL) Structured Assessment Approach for the evaluation of material control and accounting systems at facilities that process Special Nuclear Material (SNM). The Structured Assessment Approach is defined in this chapter, and ADA's contributions to it are summarized in the remaining two chapters.

A. PROJECT TEAM APPROACH AND ADA'S ROLE

The Structured Assessment Approach (SAA) to the analysis of material control and accounting (MC&A) systems was originally proposed by Dr. Ivan Sacks of LLL, [Sacks, Parziale, Schrot, Long, "A Structural Approach to the Assessment of Material Control and Accounting Systems," MC 78-203, LLL, March 1978]. The specific analytical techniques and algorithms to be used in the approach have been developed by a project team consisting of both ADA and LLL personnel. ADA's has been primarily concerned with probabilistic modeling while LLL has been primarily concerned with system dynamics.

The Structured Assessment Approach defines the MC&A system response to adversaries who have increasing levels of sophistication. Four groups of analytical techniques are developed to analyze four general adversary types.

- Level 1 deals with the adversary who does not tamper, has no knowledge of system availability, and will not attack if there is a risk of detection.
- Level 2 treats the adversary who does not tamper, knows the system operating mode but does not know about random component failures. He attacks at a random time hoping that the system is down.
- Level 3 addresses the adversary who does not tamper, but who has complete or partial knowledge of the status of system components, which he exploits in choosing when and where to attack.
- Level 4 considers the adversary who tampers with the system whenever and wherever he can. Various knowledge states are postulated.

ADA has contributed to the four-level Structured Assessment Approach in two ways. First, we interacted continuously with LLL engineers to expand, refine and improve the analytical framework. Second, we have developed specific algorithms for probabilistic analysis.

Chapter II is an overview of the four-level Structured Assessment Approach. In this overview we introduced a notation and a list of assumptions that states explicitly what is being developed. This effort has already helped communications among the people working on the Structured Assessment Approach, isolating logical inconsistencies in proposed procedures and suggesting refinements and extensions to them. We hope that Chapter II will also provide a basis for improved communications between the people working on the Structured Assessment Approach and the other people involved in the MC&A project.

Chapter III focuses on our work regarding probabilistic algorithms, related primarily to Levels 2 and 3. We explain the algorithms that we have developed, compare them to similar existing algorithms, and discuss proposed enhancements to improve computational efficiency.

B. PROBLEM STATEMENT

The staged approach determines the vulnerability of an SNM facility to various types of adversaries. Key terms in this definition are:

- Vulnerability is the probability that the MC&A system will detect an attempted diversion of SNM.
- SNM facilities are described by their physical areas (rooms, passageways, fenced yards, etc.), the process elements (pipes, valves, tanks, etc.), MC&A elements (monitors, transmission lines, computers, guards, etc.), and utility elements (generators, distribution lines, etc.).
- Adversary types are characterized by the levels of adversary knowledge and capability. These levels are described by the resources available to them (tools, number of people, etc.), knowledge state (insider vs. outsider, knowledge of system vulnerability, through malfunctioning safeguards components, etc.), objectives (quantity sought, time constraints), and decision rules (wait until the system is uncovered vs. try when the system reaches a given level of vulnerability, single attempt vs. repeated attempt, etc.).
Ivan Sacks, "Sequence Categories," MC 78-14, LLJ, January 13, 1978 .

The term "adversary type" is used in both a specific and a general sense in the report. All analysis in the report is conditioned on a specific adversary type such as a single insider with small tools who is capable of tampering, knows the status of all system components, wants ten kilograms of SNM, and will make a single attempt this year. Specific adversary types are grouped to form the general types considered in Levels 1 through 4.

The analysis focuses on the vulnerability of the SNM facility to a specific adversary type. No attempt is made to define the relative likelihood of attacks by various adversary types. No attempt is made to assess the consequences of adversary success. The analysis is limited to determining if a given type of adversary can "beat" the MC&A system by entering the plant, obtaining material and leaving the plant without being detected.

The physical data required for the analysis are normally represented by adjacency matrices. As illustrated in Figure 1.1, adjacency matrices, like directed graphs, represent the connection between elements. The convention

a. Physical System

b. Directed Graph

		TO		
		Tank 1	Check Valve	Tank 2
FROM	Tank 1	0	1	0
	Check Valve	0	0	1
	Tank 2	0	0	0

c. Adjacency Matrix

Figure 1.1 Adjacency Matrix Illustration

is that a one in element (i, j) means that movement from i to j is allowable. For example, in Figure 1.1 material can flow from the Check Valve to Tank 2 but not from Tank 2 to the Check Valve.

General Description Of The Staged Approach

Figure 1.2 gives a general description of the four levels of the staged approach. Each level is summarized by a general description, the characteristics that are assumed for all adversary types, and the major inputs and outputs for the level of analysis. Each level corresponds to a different stage in the Structured Assessment Approach.

Level 1

The intent of Level 1 is to determine if a non-tampering adversary can divert SNM with no risk of detection. In other words, assuming that no component has failed, is the system "covered" by the MC&A system?

The key concept in Level 1 is the generation of target sets (TS), which are lists of monitored elements that will be encountered by an adversary seeking SNM. A target set includes the areas and portals used by the adversary in entering and leaving a facility, the valves that he must open to get the SNM, and the process volumes such as tanks whose state will be altered as the SNM leaves the system.

Target sets are defined by exhaustive enumeration of the ways to divert SNM; therefore, for each adversary type, there may be many target sets. The first step is to identify all of the material access points (MAP) where a given adversary type can obtain sufficient SNM to meet his goals. Then, both material flows from potential storage locations to material access points, and physical paths from entry areas to material access points and from material access points to exit areas are exhaustively enumerated. The resulting combined physical access paths (PAP) and material flow paths (MFP) are called target sets.

By introducing information about each monitor's field of view, the monitors that protect each target set can be determined. The list of monitors that protect a target set is called a monitor target set (MTS). The data required to define the monitor target sets include a physical description of the plant, monitor field-of-view data, and adversary information.

Figure 1.2 General Description of Staged Approach to Determine MC&A System Effectiveness

Level	General Description	Characteristics Common to All Adversary Types	Major Inputs	Major Outputs
1	<p><u>MC&A System Coverage:</u></p> <p>Assuming that all MC&A components are available, does the system cover all target sets for all adversary types?</p>	<p><u>Non-tampering, Risk-Averse Enemy:</u></p> <p>Adversary does not tamper and has no knowledge of system availability. He will attack any uncovered target set.</p>	<p><u>Plant Physical Description:</u></p> <ul style="list-style-type: none"> - Area adjacency matrix - Process element adjacency matrix - MC&A system adjacency matrix - Monitor field-of-view matrix <p><u>Adversary Information:</u></p> <ul style="list-style-type: none"> - List of adversary types - List of material access points for each adversary type. 	<p><u>Outputs for each Adversary Type:</u></p> <ul style="list-style-type: none"> - Uncovered Target Sets - Monitor Target Sets.
2	<p><u>Adequate System Availability:</u></p> <p>Is the system reliable enough against each adversary type and for each monitor target set?</p>	<p><u>Non-tampering, Risk-Taking Enemy:</u></p> <p>Adversary does not tamper. He knows the system operating mode but is ignorant of component availabilities. He attacks once at a random time, hoping that MC&A system is down.</p>	<p><u>Plant Physical Description:</u></p> <ul style="list-style-type: none"> - Utility adjacency matrix - Utility/MC&A connection matrix (or unit models) - Mode list. <p><u>Availability Data:</u></p> <ul style="list-style-type: none"> - Component availability by mode. 	<p><u>Output for each Adversary Type and for each Monitor Target Set:</u></p> <ul style="list-style-type: none"> - Probability of detection. - Sensitivity to failed components.

Figure 1.2 (Continued)

Level	General Description	Characteristics Common to All Adversary Types	Major Inputs	Major Outputs
3	<p><u>System Vulnerability to Adversary with Special Knowledge:</u></p> <p>How vulnerable is the system to an adversary who knows the availability of some or all MCSA system elements?</p>	<p><u>Smart, Non-tampering Enemy:</u></p> <p>Adversary, does not tamper. He knows modes and some or all component availabilities. Types of adversary knowledge states are:</p> <p>A. Complete knowledge of all component availabilities; attacks only uncovered target sets</p> <p>B. Knowledge of status of some components, ignorance of others.</p>	<p><u>Observation Sets:</u></p> <ul style="list-style-type: none"> - For each adversary type list of components whose status is known. <p><u>Availability Data</u></p> <ul style="list-style-type: none"> - Mean time to failure - Mean time to repair 	<p><u>Output for each Adversary Type:</u></p> <p>A. Frequency with which each target set becomes uncovered. Frequency with which system becomes uncovered.</p> <p>B. Ranking of system components by impact of single component failure and by impact of adversary observing single component.</p>
4	<p><u>System Vulnerability to Tampering:</u></p> <p>What is system performance against adversaries who have special knowledge and who tamper? What is the unconditional probability of successful diversion for each adversary type?</p>	<p><u>Smart, Tampering Enemy:</u></p> <p>Adversary will tamper if he can reach the area where he has access to a component. Levels of analysis are classified by dynamics and knowledge:</p> <p>Dynamics</p> <p>4.1 No sequencing constraints No timing constraints</p> <p>4.2 Sequencing constraints No timing constraints</p> <p>4.3 Sequencing constraints Timing constraints</p> <p>Knowledge</p> <p>A Level 1 plus tampering</p> <p>B Level 2 plus tampering</p> <p>C Level 3 plus tampering</p>	<p><u>Location Data:</u></p> <ul style="list-style-type: none"> - Areas from which components can be compromised. <p><u>Timing Data</u></p> <p><u>Sequencing Data</u></p> <p><u>Availability Data</u></p>	<p><u>For each Adversary Type, Assuming Tampering:</u></p> <ul style="list-style-type: none"> - Uncovered target sets - Probability of detection for covered target sets. <p><u>For SNM Facility</u></p> <ul style="list-style-type: none"> - Acceptable or Unacceptable

The physical plant is described by an area adjacency matrix, showing which areas are directly connected to each other; a process element adjacency matrix, showing which elements SNM can flow between; and an MC&A system adjacency matrix, showing the components that the information flows between in passing from a monitor to the response point. The monitor field-of-view matrix shows which plant components are observed by which monitors. The adversary data includes a list of adversary types and a list of material access points for each type of adversary.

The output from the Level 1 analysis identifies to the NRC analyst all uncovered target sets, the ones for which the monitor target set contains no elements. In addition, for each covered target set, the monitor target set is listed.

Level 2

The Level 2 analysis extends the Level 1 analysis to consider system reliability. For each adversary type and for each monitor target set, the system reliability is calculated.

We assume that the adversary does not tamper with the system, that he does not have knowledge of the system status except the operating mode, and that he makes only one attempt to divert SNM. Consequently, the appropriate system availability measure is the probability that a given monitor target set will be uncovered if attacked once by a given adversary type at a random entry time during any given operating mode of the facility.

The calculation of the probability of detection conditioned on adversary type, mode, and monitor target set is complicated by the common mode failure problem. Utilities such as electricity or compressed air can fail causing several MC&A components to fail at once. The utility structure is part of the input to the Level 2 analysis, allowing dependence among components to be explicitly modeled.

In calculating the probability of detection, an important feature is the ability to do sensitivity studies on the unavailability of sets of components. This is a feature that will be useful in Levels 3 and 4.

Level 3

Level 3 introduces more sophisticated adversary types with special knowledge of the status of the MC&A system. These adversaries do not tam-

per with the system, but they do have knowledge of the status of some or all of the MCSA system components, i.e., the ones that are not working, leaving the system vulnerable. [Adversaries could have complete knowledge (Level 3A) or partial knowledge (Level 3B) of the system status.] The basic question is: How vulnerable is the system to somebody who knows its status?

Complete knowledge is equivalent to observing a status board with a light that goes on for every component that is operational, and that goes out for every component that has failed. Under complete knowledge without tampering (Level 3A), we assume that an adversary will attack only uncovered target sets. The output for this type of adversary is the frequency with which various target sets become uncovered and the frequency with which the facility becomes uncovered.

Adversaries with partial knowledge of the system (Level 3B) know the status of some components and are uncertain about the status of the remaining components. Level 3B analysis is currently used for sensitivity analyses to single component failures. The output of this analysis ranks the individual components according to the net change in probability of detection caused by their random failures. This probability change is separable into two factors: the known failure rate of the component and the fact that the adversary can observe the component's status. The first ranking identifies components upon whose reliability the system is heavily dependent. The second ranking answers the question: How bad is it if an adversary can observe a component's status?

As explained in Chapter II, additional analysis at Level 3 may be useful, but additional information about the adversary is required before such algorithms can be developed.

Level 4

Level 4 asks the question: What is the unconditional probability of successful diversion for each adversary type? This is the most sophisticated level of analysis, and if completed successfully and modeled in the same level of detail as the previous levels, it will include all of the results of Levels 1 through 3 as special cases.

Level 4 has more special cases than any of the previous levels. We have identified nine potential sublevels: three different specifications of adversary dynamics for each of the three different types of adversary knowledge.

Dynamics are characterized by the adversary's constraints on sequencing and timing. An example of a sequence constraint is that an adversary might be able to enter either room A or room B from the outside, but once inside A or B he cannot enter the other. An example of a timing constraint is that the adversary has to fill a container with SNM within a certain time span if a roving guard comes by at regular intervals. The three levels of dynamics are: Level 4.1 - no constraints; Level 4.2 - sequencing constraints; Level 4.3 - both timing and sequencing constraints.

Various knowledge states are introduced in Levels 4A, 4B and 4C because knowledge is an important adversary resource. Knowledge of the status of the MC&A system status is a resource just like a handtool because it helps the adversary to "beat the system," the more the adversary knows, the less likely that the system will detect him. Just as Levels 1 through 3 differentiate knowledge states for the non-tampering adversary type, Levels 4A through 4C differentiate knowledge states for the tampering adversary. Level 4A corresponds to Level 1, where the adversary does not know the availability of the components in the MC&A system and attacks only uncovered target sets. Of course, with tampering allowed, at least as many target sets are uncovered for Level 4A than for Level 1. Level 4B corresponds to Level 2, where the adversary still does not know component availabilities but he will attack covered target sets, hoping for random failures to uncover them. Level 4C corresponds to Level 3 where the adversary knows the status of system components.

The nine special cases for Level 4 corresponding to the various dynamic states and knowledge are conveniently denoted in Level 4.1A, 4.1B, 4.1C, 4.2A, 4.2B, 4.2C, 4.3A, 4.3B, and 4.3C. Although all nine sublevels appear to represent credible adversary types, each sublevel does not require a separate computational technique, as explained in Chapter II.

Additional input data will be required for Level A analysis. Clearly, the areas from which the components can be accessed by a tampering adversary is an important input. Specific needs for timing, sequencing, and availability data have not been established.

The principal output from Level 4 is the acceptability or unacceptability of an SNM facility. Other outputs include lists of uncovered target sets and the probability of detection for covered target sets.

C. ADVANTAGES OF THE STAGED APPROACH OVER A GENERAL PURPOSE
COMPUTATIONAL METHODOLOGY

It is obvious that any computational tool that can solve the most complex Level 4 problem -- Level 4.3C, which allows tampering, exploitation of knowledge of the availability of system components, and probabilistic treatment of the system dynamics -- can also solve the less complex problems of Levels 1 through 3 and the special cases of Level 4. If this is true, why bother with the preliminary steps? Won't Levels 1, 2, 3 and the special cases of Level 4 be useless when the "ultimate analysis" is developed?

The skeptic's answer is that the "ultimate analysis" will never be developed within a finite budget, but that does not really answer the question. Suppose that we postulate the existence of an approach for Level 4.1A -- the tampering case with deterministic component availabilities, and without sequence or timing detail -- and are confident of the approach feasibility. Level 4.1A dominates Level 1 in the sense that any system that passes Level 4.1A will pass Level 1. In this case, why bother with Level 1 if Level 4.1A is feasible?

The answer is that significant insight may be gained through the preliminary stages of analysis, include the computational and logistical advantages of using simple tools to solve simple problems, the advantages of solving special cases before solving the general case, the degree of automation that can be introduced through the staged approach, the ability to postulate and evaluate performance criterion at less than the total system level, and the ability to change the level of detail of the analysis while focusing on various aspects of the problem.

The Computational and Logistical Advantages of Using Simple Tools
to Solve Simple Problems

Many adversary types will be considered before a plant can be licensed. Presumably, almost all adversary types will be intelligent enough to attack an uncovered target set, but only a few will simultaneously have ability to tamper, have complete knowledge of the system availability, and be sensitive to timing and sequencing constraints. Consequently, for some adversary types the argument can be turned around: Why bother with a Level 4 analysis when a Level 1, 2, or 3 analysis is sufficient?

Another advantage of using simple tools to solve simple problems is the identification of inadequate facility designs. If a facility design fails a simple analysis in Levels 1, 2, or 3 then why should NRC bother with a complex Level 4 analysis to reach the same conclusion?

The Advantages of Solving Special Cases Before Solving the General Case

From a practical point of view, good analysts often solve special cases of a problem before attacking the general case. Intuition is built up through the analysis of special cases which allows the analyst to check the general solution for validity. For example, if a Level 1 analysis is available for the case without tampering, the results should be reproduced by a Level 4.1A analysis assuming that the adversary can reach no components. If the two analyses produce the same results, confidence in the Level 4.1A analysis increases.

A Staged Approach Maximizes Automation and Minimizes Judgmental Inputs

Another advantage of the staged approach is the ability to automate. The staged procedure is designed so that an SNM system evaluator without special training in probabilistic system dynamics can use it. Level 1 requires only highly standardized inputs such as information about what is connected to what. Very little judgment is required on the part of the analyst. Level 2 requires some additional analytical input to define system operating modes, and to estimate system availabilities based on those modes. Level 3 analysis requires approximately the same level of judgment as Level 2. Level 4 analysis requires a great deal more judgment from the analyst, both in specifying the circumstances under which tampering can occur, and, for problems with finite computing resources, determining the level of detail of the model.

As the level of judgment required from the analyst rises, the repeatability of the results and the resultant credibility of the analysis decreases. In other words, hardly anybody would argue with the rejection of a facility because it failed to pass a Level 1 analysis. However, if a system failed to pass a Level 4 analysis, there is the potential for the proponents of the facility to claim that the problem was not with the facility design itself

but with some simplifying assumption introduced by the analyst in modeling the system. A staged approach rejects a system for the strongest possible basis if it is to be rejected and requires the minimum amount of judgmental input if it is to be accepted.

The Staged Approach Allows for a Variety of Generic Performance Criteria

The staged approach allows performance criterion at less than the total system level. If a Level 4 analysis were the only analysis performed, the result would be that a system was acceptable or unacceptable. Regulatory history indicates that such an aggregate level approach is likely to be insufficient by itself. If performance criterion such as "the system shall have at least ____ probability of detecting any adversary attacking any target set, assuming no tampering and no knowledge of the system status" is proposed then a staged approach is necessary in order to calculate the probabilities of diversion conditioned on the events specified in the performance criterion.

The Staged Approach Focuses Analytical Resources Where They Are Needed

The final argument for a staged approach is that given a computational constraint, the maximum level of detail varies inversely with the generality of the analysis. If an unnecessarily general approach is taken, unnecessary simplifying assumptions may have to be introduced. For example, in Level 2 analysis common mode failures may be examined in great detail by modeling the utility networks at a very fine-grained level. On the other hand, in the Level 4 analysis, the complexity introduced by the analysis of tampering, might indicate that the utility system has to be modeled in a relatively aggregate way. Put another way, the staged approach may determine that certain issues are important or unimportant so that the analytical resources of Level 4 can be directed at the areas that have the greatest chance of changing the conclusions of the analysis.

A Final Observation

A final observation on the staged approach is that by forcing the analyst to think carefully through the various stages to develop the staged tools, the analytical structure that is developed for the Level 4

analysis is likely to be richer than the one that would be developed if the problem was attacked in its most general form from the outset.

II. DEFINITION OF STRUCTURED ASSESSMENT APPROACH

Chapter I gave an overview of the Structured Assessment Approach. This chapter provides a more detailed description of that procedure in order to define clearly the objectives for system modeling.

The algorithms and procedures to implement the Structured Assessment Approach are more fully developed for Levels 1 and 2 than for Levels 3 and 4. For Levels 1 and 2 well-defined algorithms and prototype computer codes are being developed at LLL. Algorithms for Level 3 exist, but we expect to change and improve them as the project goes on. Algorithms for the dynamics in Level 4 are under development at LLL. Implementing the probabilistic aspects of Level 4 remains as future work.

Level 1 and the nonprobabilistic parts of Level 2 are currently being implemented at LLL. The probabilistic aspects of Level 2 have been implemented in the developmental computer code described in Chapter III of this report. By developmental, we mean that the codes have not been optimized with respect to computational efficiency, or with respect to the maximum number of MC&A system components that can be included. The intent is to demonstrate the methodology and to provide a check that the methodology is complete and logically consistent. The efforts to date in Levels 1 and 2 are encouraging. The tools that have been developed have been used to analyze several simplified examples, and the end results are credible. Future work on Level 1 and 2 algorithms will likely result in refinements and improvements for computational efficiency; the scope and definitions for these levels appear to have stabilized.

Levels 3 and 4 are less developed. The problem specification for these approaches has not yet stabilized. As computational approaches are tried and examples are worked out for Level 3 and Level 4 analyses, we are still finding logical inconsistencies in our problem definitions, and feasible conditions that are not well-handled within the methodology. The basic analytical approaches that are proposed for Levels 3 and 4 are likely to endure, but their specific implementations will change.

Because Levels 1 and 2 are more completely developed than Levels 3 and 4, the material in this chapter serves a different purpose for the various levels. For Levels 1 and 2, we summarize the existing methodology. For Levels 3 and 4, we define a framework to compare various efforts using a consistent notation and terminology. For Level 3, we postulate an algorithm. This algorithm has not been reviewed by the project team, and consequently, it should be taken as a "straw man." For Level 4 we do not review any specific algorithms; the LLL members of the project team will document the algorithms that they are developing in separate project memos.

The chapter proceeds in three parts. In Section A, terms are defined and the general approach is specified. In Section B, each of the four stages is related to the general approach. In Section C, the general approach is used to compare the staged assessment procedure to other analytical efforts related to the MC&A project.

A. GENERAL APPROACH

The general approach is to define an explicit adversary model, an explicit characterization of the MC&A system availability, and a measure of system performance. In order to do this, we need to introduce some concepts from decision theory as well as some specific definitions and notational conventions.

Preliminaries

Decomposition is a key concept in problem structuring. It allows the analyst to focus attention on individual pieces of the problem, rather than the complex whole. In probabilistic modeling, one procedure for decomposing a problem is called expansion. The technique allows the analyst to condition the probability of an event on some related event. This may make computation or assessment of the event easier. For example, most experts would rather provide the probability that the bubbler is available assuming that the plant is in normal operating mode and that no tampering has occurred, rather than providing the unconditional probability of bubbler availability.

Defining events A and B_i ($i=1, 2, 3, \dots$), the expansion rule is as follows:

$$p(A) = \sum_i p(A, B_i) .$$

Using the definition of conditional probability, this can be rewritten:

$$p(A) = \sum_i p(A|B_i) p(B_i) .$$

If, for example, we think of A as the availability of the bubbler and B_i as all possible combinations of the operating modes of the plant and the adversary's ability to tamper with the bubbler, then it is clear that one can assess conditional probabilities of detection $p(A|B_i)$ rather than assessing the unconditional event $p(A)$. The tradeoff is that the unconditional probabilities of system mode and tampering ability must now be assessed.

Expansion allows one to do computations on paper or in the computer rather than in one's head. The name "expansion" is used because one first expands the event space (from A to A, B_i) and then makes probability

assignments over the expanded space. The assessments are made to joint events: $P(A, B_i)$ or to conditional events $P(A|B_i)$.

For SNM facility evaluation, we are always concerned with the binary event that the system detects (D) or does not detect an adversary ($\bar{D}$). Expansion as it applies to the SNM problem is shown in Figure 2.1. The unconditional probability of detection is equal to the summation over all possible system statuses of the product of the conditional probability of detection given a specific system status multiplied by the probability of that specific system status when the adversary attacks. The expansion rule can also be represented as a probability tree. In Figure 2.1 we show on the left a two branch node with the labels D and $\bar{D}$, which represents the events detection and no detection respectively. The two branch tree represents the event of detection and shows that it has a binary set of outcomes. The larger tree to the right of the binary tree represents the right hand side of the expansion equation. There are a number of possible system statuses represented by the branches labeled number s_1 through s_n . Each possible system status is followed by the binary branch for detection. This is shorthand for the fact that the probability of detection is conditioned on the system status, and the system status is not conditioned on any other explicit events or variables. An alternate form of the shorthand is shown at the right of the figure. All of the possible system states are shown by a node with the generic variable s in the middle. The binary detection node is shown to the right of the system status node but is not attached to it. This indicates the same information as the more expanded tree to the left of it. In addition, it is important to notice that each node, the point from which the branches emanate, is represented by a circle. In our notation, circles represent chance or uncertain points, while squares will be used later to denote decision points.

Using Decision Trees and the Expansion Rule to Represent the Adversary's Decision Problem and Measures of System Effectiveness

In this section we demonstrate that the adversary's decision problem and the system effectiveness measure contain essentially the same information. Without this insight we tend to postulate system performance measures directly; e.g., we might postulate a Level 2 system performance measure as: the system will have a criterion on the probability of detection for any non-

Expansion

$$p(D) = \sum_{s_i} p(D|s_i) p(s_i)$$

Tree Form

Definitions:

- D Event of successful detection of adversary
- $\bar{D}$ Event of unsuccessful detection of adversary
- s A variable specifying the system status when the adversary attacks
- $\sum_{s_i}$ The generalized summation over s ; e.g., the sum over all values if s is a discrete random variable or the integral with respect to s if s is a continuous random variable
- $p(*|**)$ The probability of variable or event $*$ given variable or event $**$
- $p(*)$ The unconditional probability of event $*$

Figure 2.1 Probabilistic Expansion

tampering adversary and any target set. Directly postulating system performance measures leaves open the questions of why we chose the measure and what it implies. Introducing the link to the adversary's decision problem allows us to use the notation of decision trees and expansion to develop system performance measures and to record the assumptions associated with them.

In Figure 2.2 we define a generic decision problem for the adversary and a generic system effectiveness measure. We have used a terminology that is general enough to apply to each of Levels 1 through 4 as well as to many other analytical efforts such as the Aggregate System Model by Stein Weissenberger of LLL, Bruce Judd of ADA, and others [B. Judd, "Methodology and Preliminary Models for Analyzing Nuclear Safeguards Decisions," ADA, July 1978, Draft].

The adversary's decision problem can be represented by the following sequence of events:

- The adversary chooses a diversion strategy
- The MC&A system responds to the challenge by the adversary
- The adversary is detected or not detected
- The adversary achieves his goals or he fails.

The sequence of events is represented graphically by the schematic decision tree at the top of Figure 2.2.

Throughout the Structured Assessment Approach, we assume that the value of an adversary's attempt depends only on whether or not he is detected. While at first this may seem to be a restrictive assumption, it is important to remember that this analysis is conditioned on adversary type. Consequently, variables such as the quantity of SNM sought have already been considered in defining the adversary type. We assume that the adversary types attack only targets that will achieve their goals, and that they are indifferent to how they achieve those goals as long as they successfully divert SNM without being detected.

Given that the adversary's value depends on the probability of detection, without loss of generality, we can assign a value of 0 to the case where the adversary is detected and a value of 1 to the case where the adversary is not detected. Assuming that the adversary maximizes his expected

The Adversary's Decision Problem

Adversary's Expected Value =

$$\sum_{\text{Response}} \left(p(\bar{D} | \text{Response}, \text{Strategy}^*) p(\text{Response} | \text{Strategy}^*) \right)$$

The System Effectiveness Measure

The System Effectiveness Measure is defined as the probability of detection given that the adversary chooses his best option:

$$p(D | \text{Strategy}^*),$$

but by expansion

$$= \sum_{\text{Response}} p(D | \text{Response}, \text{Strategy}^*) p(\text{Response} | \text{Strategy}^*)$$

Figure 2.2 Comparison of Adversary's Decision Problem and System Effectiveness Measure

value, the expression for the expected value is shown below the tree in Figure 2.2. The star on the strategy implies that the specific strategy is chosen that maximizes the expected value. The expression states that the expected value is the summation over all possible system responses of the product of the probability of no detection (successful diversion) given a particular system response and the optimal strategy, multiplied by the probability of that system response given the optimal strategy.

Moving to the lower portion of Figure 2.2, the definition of the measure of system effectiveness is the probability that the system detects an attempted diversion of SNM given that the adversary will choose the option that gives him the highest probability of success. Writing down the symbols for this expression and using the expansion rule to introduce the system responses, we find that we get an expression for system effectiveness measures that is similar to the one for the adversary's expected value. The only difference is that for the adversary we focus on the probability of successful diversion ($\bar{D}$) while for the system we focus on the probability of successful detection (D). However, since the former is one minus the latter, the two quantities represent the same information.

The conclusion from Figure 2.2 is that the definition of system performance measure is equivalent to the definition of the adversary's expected value. Consequently, in the following sections we look carefully at the adversary's decision problem for the various levels of the staged analysis in order to define system performance measures.

B. DISCUSSION OF THE STAGED ASSESSMENT PROCEDURE

This section gives an overview of the logical basis for each of the stages. Detailed algorithmic developments of the stages are given in project memos (see Chapter IV for abstract of ADA memos and references for list of LLL memos) and in Chapter III of this report.

Problem Statement for Level 1

In the Level 1 analysis we assume that the adversary does not tamper, has no knowledge of the system availability, and will not attack unless the system is uncovered. Consequently, the adversary model is the simple one represented in Figure 2.3. The adversary's decision is to choose a monitor target set (MTS). The monitor target set is defined as the set of monitors that covers a given target set, and the target set is a list of all areas and process elements that the adversary must come in contact with during an attempted diversion of SNM. As shown in Figure 2.3, the probability of detection is defined as either 0 or 1, making the analysis deterministic. This is equivalent to assuming that all monitors are perfectly reliable; the analysis simply checks to see whether all possible target sets are covered by at least one monitor. The Level 1 system performance measure is that the system fails if there is an uncovered target set and that the system passes if all monitor target sets are covered.

Knowing that a system has passed a Level 1 analysis does not provide as much information as knowing that it failed Level 1. Failure indicates a poor system design and the facility should be rejected. Success can be regarded as a consistency check for Level 2 analysis. It does not make sense to do a probabilistic analysis of an MC&A system if it fails deterministically.

The major analytical effort of Level 1 is the determination of an exhaustive list of monitor target sets. Figure 2.4 summarizes the logical steps in the generation of the monitor target sets (MTS). Initially, for each adversary type, all of the possible material access points and the elements of the material flow system from which SNM can be obtained are specified. In addition, adjacency and location data are provided in the form of an area adjacency matrix, a process element by area location matrix, a monitor by area field-of-view matrix, a monitor by process element field-of-

Adversary's Decision Problem

$$p(D) = \begin{cases} 0 & \text{if MTS contains no elements} \\ 1 & \text{otherwise} \end{cases}$$

System Performance Measure

$$\text{SPM} = \begin{cases} 0 & \text{if an empty MTS exists} \\ 1 & \text{otherwise} \end{cases}$$

Figure 2.3 Level One Problem Description

Definitions

A	Area
M	MC&A System element
m	Monitor
P	Process element
AT	Model of Adversary Type - attributes include quantity sought, resources, knowledge state.
MAP	Material Access Point - an element of material flow system
PAP	Physical Access Paths - exit and entry - lists of areas
MFP	Material Flow Paths - lists of material flow system elements
MTS	Monitors that cover Target Set - list of monitors
AV	Set of MC&A system component availabilities

Figure 2.4 Determination of Monitor Target Sets

view matrix, and a process element by process element adjacency matrix. Additional inputs are the areas from which a given adversary enters the SNM facility and the locations within the process elements where material can be stored in sufficient quantity for the given adversary type.

The first step is to determine the physical access paths. A path-finding computer code is used to find all physical access paths from the entry area to the material access points. The area by area adjacency matrix is the major input to the computer code. The starting points are defined by the areas where the adversary can enter the SNM facility. The ending points are determined by the areas in which the material access points are located, which are determined by using the process element by area location matrix.

The physical access paths can be translated into a set of monitors that cover those paths by a multiplication of the physical access paths times the monitor by area field-of-view matrix. It should be noted that the actual algorithm being developed to determine the monitors that cover physical access paths is more complicated than this simplified description. First, the algorithm considers entry and exits paths. Second, the adversary may have to visit areas in addition to the one containing the material access point in order to open and close valves (the Acquisition Target Set).

After the physical access paths have been developed, the next step is to compute the material flow paths. This calculation is analogous to the one for the physical access paths except that the basic input matrix is the process element matrix, the material storage areas are the entry point, and the material access points are the exit points. The process monitors affected can be determined by multiplying the material flow path vector by the monitor by process element field-of-view matrix.

Problem Statement for Level 2

The adversary's decision problem and system performance measures for Level 2 are shown in Figure 2.5. The adversary's decision problem is similar to that for Level 1. The change is that instead of assuming that the MCSA system is perfectly reliable, we formally introduce uncertainty through the detection probability and availability of MCSA system components. The expansion rule is used to condition the probability of detection on the availa-

Adversary's Decision Problem

Expansion

$$p(D|AT, MTS) = \sum_{AV} p(D|\underline{AV}, AT, MTS) p(\underline{AV}|AT, MTS)$$

System Performance Measure

$$SPM = \sum_{AV} p(D|\underline{AV}, AT, MTS^*) p(\underline{AV}|AT, MTS^*)$$

Figure 2.5 Level Two Problem Description

bility of the system components, as well as the previous conditioning events of adversary type and monitor target set. The first term in the expansion--the probability of detection conditional on the availability of the system components, the adversary type, and the optimum monitor target set -- is the description of how the overall system performance is related to component availabilities. This term is modeled as Boolean; that is, if the availabilities of all the components are known

$$AV_i = \begin{cases} 0 & \text{if } i^{\text{th}} \text{ component unavailable} \\ 1 & \text{if } i^{\text{th}} \text{ component available} \end{cases}$$

then a Boolean expression $\Phi(AV)$ determines if system detects

$$\Phi(AV) = \begin{cases} 0 & \text{if no detection} \\ 1 & \text{if detection.} \end{cases}$$

The approach that is being developed at LLL to develop this Boolean expression is described below.

The resulting system performance measure is shown at the bottom of Figure 2.5. The first term is the value, either 1 or 0, of the Boolean expression. The second term is the probability that the set of components are available given the adversary type and the monitor target sets. This probability is calculated from the component reliability data. For Level 2 analysis, the component reliability data represent the average component availability.

The star on the monitor target set (MTS*) indicates that the probability of detection is calculated for each monitor target set and then the minimum probability of detection (the one that maximizes the adversary's chances) is selected as the systems performance measure. Although we don't show it explicitly in our notation, we assume that the adversary is also allowed to maximize over plant operating modes. In other words, a given set of monitors can constitute several different monitor target sets depending upon the mode of the plant when the attack occurs. It is important to differentiate between modes because the component availabilities may change from mode to mode.

The calculation of the probability of detection for Level 2 analysis is shown in Figure 2.6. Notice that the probability that is calculated, shown at the bottom of the figure, is conditioned on both the adversary type and the monitor target set. The calculation must be repeated for every adversary type and every target set. Starting at the upper left of the figure, signal

Figure 2.6 Determination of the Level Two Probability of Detection

flow paths are developed by applying the PATH code to the MC&A system component adjacency matrix. The starting point for the paths are the monitors contained within the monitor target set, and the ending points are the response nodes.

At the upper right of Figure 2.6 the calculation of the utility paths is illustrated. The PATH code operates on the utility component adjacency matrix to find the paths from the utility generation sources to the elements in the signal flow paths.

Unit models of the elements in the signal flow paths are introduced to define the logical relationship between the signal flow in the unit model and the utilities attached to it. (See Project Memo by Steve Derby "Procedures for Combining Utility Networks with Signal Flow Networks," September 22, 1978, and Memo MC-78-884 by I.J. Sacks and A.A. Parziale, "Unit Models." The output from the unit model calculation are a set of events that must be true in order for the probability of detection to occur.

The reliability code calculates the probability of detection based on the Boolean expressions and the system availabilities that are contained in a computer code entitled PROB. The code that ADA developed during this project to perform the reliability calculation is the subject of Chapter III.

Problem Statement for Level 3

The algorithms for Levels 1 and 2 have been specified and are being tested. For Levels 3 and 4 the analytical procedures are still under development. Consequently, the discussion for Levels 3 and 4 is more speculative than the previous discussion.

Two possible adversary models for Level 3 analysis are shown in Figure 2.7. We call them Level 3A and Level 3B.

Level 3A

In the first model, we assume that the adversary knows the status of every component in the MC&A system, and that he will not attack until he observes an uncovered target set. This is illustrated by the sequence shown in the graph:

- The component availabilities are specified.

Adversary's Decision Problem

System Performance Measure = $p(\bar{D} | MTS^*)$ = probability that system is uncovered

(a) Adversary Who Knows Entire System Status and Waits for Uncovered Target Set.

(b) Adversary Who Knows Status of a Subset of Components.

Figure 2.7 Possible Interpretations of Level Three Analysis

- Based on the component availabilities the system will or will not detect an adversary attacking a given target set.
- Then the adversary decides whether or not to attack a given target set.
- Finally, the adversary wins or loses.

Since the adversary will attack an uncovered target set if possible, the system performance measure becomes the probability that the system will not detect for some target set. In other words, the system performance measure is the probability that the system is uncovered.

In theory this calculation is straightforward, assuming an exponential failure model and given the mean time between failures and the mean time to repair. In fact, the question is complicated by the fact that the plant operator may change the plant mode based on the system availability; eg. if the back-up diesel generator will not start during a routine test, plant operating procedures may dictate that the plant be shut down. More work is needed to relate the availability model in PROB to plant operating procedures before Level 3A can be fully developed.

Level 3B

A modified view of the adversary is shown in Figure 2.7b, assuming that the adversary can observe only a subset of the MC&A system components. Based on his knowledge of those components, he chooses the target set that gives him the minimum chance of being detected. Then, based on the availability of remaining components, he is either detected or not detected, and he wins or loses. The simplest type of Level 3B analysis considers the system effectiveness for single component failures.

In the general case, the systems effectiveness is more complicated than Level 3A. The adversary's decision rule is not uniquely defined by the information in Figure 2.7b. Will the adversary attack as soon as a component fails? Will he wait for some threshold level of system degradation? Will he maximize his chances of success in a given time period such as one year? Is he allowed to try repeatedly? Questions such as these are important in determining the systems effectiveness measure for Level 3B. Many of these same questions are being addressed currently by other members of the MC&A project team. As detailed adversary models are developed,

Level 3B can be refined.

At present, Level 3B analysis is a single failure analysis to determine which components are most vulnerable to an adversary who observes their status. The procedure is based on the tree in Figure 2.7b, assuming that the observable set of components is a single component A_k . The sensitivity analysis addresses two distinct questions. First, what is the system vulnerability to the known loss of a single component? Second, how is the system performance affected by the fact that the adversary can observe the availability of a component? The first question finds components upon whose reliability the system is heavily dependent. The analyst will want to double-check the reliability assigned to these components and carefully assess the potential for tampering with them. The second question is a direct answer to the Level 3 question: How bad is it if an adversary exploits knowledge of the system status?

Although we discuss the Level 3B analysis in terms of single component failures, the extension to multiple component failures--all possible combinations of components taken two at a time, three at a time, etc.--is straightforward.

The single failure analysis is summarized in Figure 2.8. In Step 1, the probability of detection conditioned on adversary type, monitor target set, and single component failure is computed. This calculation is straightforward using the Level 2 probabilistic model. One component at a time, the component availabilities are set to zero and the probability of detection is recalculated.

Step 2 finds the target set that affords the adversary the highest chance of success for each single component failure. The key assumption in Level 3B is that the adversary is allowed to determine his strategy after he observes a component failure. Pictorially, this is represented by the fact that the adversary decision node follows the AV_1 node in Figure 2.7b. This means that at the time the adversary chooses the monitor target set he knows which components have failed and he uses this knowledge. Consequently, the monitor target set that maximized the adversary's value for Level 2 will not necessarily maximize his value

Step 1

For each adversary type AT_i , and each monitor target set MTS_j , and each single component failure $\overline{AV}_k$, calculate the probability of detection:

$$p(D|AT_i, MTS_j, \overline{AV}_k)$$

Step 2

Minimize over the monitor target sets to find the lowest probability of detection for each adversary type and single component failure

$$p(D|AT_i, \overline{AV}_k) = p(D|AT_i, MTS^*, \overline{AV}_k)$$

Step 3

Subtract the result of Step 2 from $p(D|AT_i, AV_k)$, the output from Level 2 to derive

$$\text{MEASURE 1} = p(D|AT_i, AV_k) - p(D|AT_i, \overline{AV}_k)$$

Sort to determine the system vulnerability to the known failure of a single component.

Step 4

Compute the impact of the adversary being able to observe component k :

$$\begin{aligned} p(D|AT_i, AV_k \text{ observable}) &= p(D|AV_k, AT_i) p(AV_k) \\ &+ p(D|\overline{AV}_k, AT_i) p(\overline{AV}_k). \end{aligned}$$

Step 5

Subtract the result of Step 4 from the output from Level 2 and sort to derive

$$\text{MEASURE 2} = p(D|AT_i, AV_k) - p(D|AT_i, AV_k \text{ observable}).$$

Figure 2.8 Level 3B Sensitivity Analysis Procedure

for Level 3B. Given the additional information of a known failed component, we must minimize the probability of detection over monitor target sets.

In Step 3 of the single failure analysis two measures are defined. For MEASURE 1, the result from Step 2, the probability of detection given adversary type and that component k has failed, is subtracted from the result from the Level 2 analysis, the probability of detection given only adversary type. The differences are sorted to rank the vulnerability of the MC&A system to the known failure of components.

In Step 4, we expand the probability of detection given that component k is observable $p(D|AT_i, AV_k \text{ observable})$ over the availability of the k th component to get an expression with probabilities that are calculated by the Structured Assessment Approach.

$$p(D|AT_i, AV_k \text{ observable}) = p(D, AV_k | AT_i, AV_k \text{ observable}) \\ + p(D, \overline{AV}_k | AT_i, AV_k \text{ observable}).$$

Applying the definition of conditional probability and dropping unnecessary conditioning events, the expression reduces to

$$p(D|AT_i, AV_k \text{ observable}) = p(D|AV_k, AT_i) p(AV_k) \\ + p(D|\overline{AV}_k, AT_i) p(\overline{AV}_k)$$

The terms in this last expression have all been computed before. The probability of detection given that AV_k is available $p(D|AV_k, AT_i)$ is the Level 2 result. The probabilities of availability and unavailability for component k , $p(AV_k)$ and $p(\overline{AV}_k)$, are obtained from PROB. The probability of detection given that AV_k is unavailable $p(D|\overline{AV}_k, AT_i)$ is the quantity calculated in the sensitivity study above.

In Step 5 the result of Step 4 is subtracted from the result of Level 2, to derive MEASURE 2 for each component. Sorting the results ranks the components by the increase in system vulnerability given that an adversary can observe that component.

Summarizing, MEASURE 1 ranks the components by system impact given that they are out while MEASURE 2 ranks the components by system impact given only that they are observable.

Problem Statement for Level 4

In Chapter I we identified nine sublevels for Level 4, corresponding to all possible combinations of three separate system models (4.1, 4.2, 4.3) and three separate adversary models (4A, 4B, 4C). System models, which reflect dynamics, are under development at LLL. The ADA approach to the development of adversary models, which reflect probabilistic assumptions regarding the adversary, are discussed in this section.

System Dynamics

Sublevel 4.1 analysis is under active development at LLL. The approach is to build a logic diagram similar to a Petri Net showing the necessary conditions for an adversary who is capable of tampering to succeed (the event of no detection, D). The logic diagram is transformed into an adjacency matrix. Reachability calculations are performed on the adjacency matrix to determine the conditions under which the adversary can succeed. See memo MC 78-928-D, "Techniques for the Determination of Potential Adversary Success With Tampering," October 17, 1978 by I. J. Sacks.

Sublevel 4.2 starts from a diagram very similar to the one for Level 4.1. However, in this case rather than developing a reachability algorithm, Petri Net simulation is performed directly. See memo number WM 78-514, "Modeling Adversary Tampering of a Safeguard System with a Petri Net," June 30, 1978 by A.A. Parziale.

Sublevel 4.3 is addressed by a general purpose simulation model, (GPSS). This simulation approach was developed at LLL prior to ADA's involvement on the project and will not be discussed in this report.

Adversary Models

Levels 1 through 3 are differentiated by adversary knowledge states given no tampering. In the tampering case, these three adversary know-

ledge states are still useful. We define knowledge states under tampering as sublevels 4A, 4B and 4C. Sublevel 4A corresponds to Level 1, Sublevel 4B to Level 2, and Sublevel 4C to Level 3. We discuss each below.

Sublevel 4A asks whether a target set is covered given the assumption that if an adversary can tamper with zero risk of detection, he will. The output from this sublevel is sets of components, where the unavailability of all of the components in a set is sufficient to allow a tampering adversary to succeed without risk of detection. The methodology for this case is documented in the memo by Ivan Sacks referenced under Sublevel 4.1.

The correspondence between Sublevel 4B and Level 2 is more subtle than the correspondence between Sublevel 4A and Level 1. In both Sublevels 4B and in Level 2, we wish to compute system reliability based on average component availability, without allowing the adversary to exploit knowledge of the instantaneous availability of components. The subtlety arises in recognizing second-order effects. To illustrate, consider a monitor target set that a Level 1 analysis has determined to be covered by six monitors. The adversary can reach the first monitor (M1) directly, and he can reach monitors M2 through M6 if he can get by a seventh monitor (M7) on the control room door. The control room door monitor is not reachable by the adversary.

What is the system availability for this case? Clearly, M1 should be considered compromised, and the availability of M1 should be set to zero.

If the control room monitor is not in the monitor target set, as we have specified in the example above, then the probability of detection is equal to the probability that any one of monitors M2 through M5 is operative. This is basically the calculation that a Level 2 analysis would perform, given that monitor M1 has been eliminated.

On the other hand, if monitor M7 on the control room door were included in the monitor target set, then the system reliability would be multiplied by the probability that the door switch is available. This

can be demonstrated by expanding the system availability over the availability of M7:

$$\begin{aligned} \text{System availability} = & p(\text{M7 available}) \times (\text{system availability} \\ & \text{given M7 available}) + p(\text{M7 unavailable}) \times \\ & (\text{system availability given M7 unavailable}). \end{aligned}$$

The last term is zero where tampering is assumed, so the expression becomes

$$\text{System availability} = p(\text{M7 available}) (\text{system availability given M7 available}).$$

Clearly, the system availability is dependent in the choice of the monitor target set.

The choice of the appropriate monitor target set for Sublevel 4B depends on how smart the adversary is. On the one hand, it does not make sense to confine the Sublevel 4B adversary to monitor target sets from Level 1, because then he would never enter an area solely for the purpose of tampering. In the example above, the adversary would never enter the control room. On the other hand, it does not make sense to allow the adversary to exploit all possible random failures by entering new areas to tamper given that a random failure has occurred involves an adversary knowledge level that is appropriate for Sublevel 4C, not for Sublevel 4B.

Although analytical methods have not yet been specified for Sublevel 4B, we speculate that the Sublevel 4B system reliability, which assumes tampering but no knowledge of instant availability of the MC&A component, is feasible. We think that there are two key elements to this analysis:

1. The monitor target set must be based on a Sublevel 4.1 type of analysis, so that they are appropriate for a tampering adversary.
2. The probabilistic code to calculate system reliability will need to handle the dependent probabilities in order to account for the fact that

certain components are unreliable given that the adversary has access to the area from which they can be tampered.

We need to wait until Sublevel 4.1 analytical procedures are defined further, before we can determine the feasibility of generating the monitor target set under tampering. The current ADA system reliability code assumes independent component probabilities, but the second generation code that is currently under development, will handle dependent probabilities.

In Sublevel 4C, the adversary has a state of knowledge comparable to Level 3. He first observes whether or not the system components are available, then he decides how to attack the system. A conceptually straightforward approach is available although we do not know its computational limitations. It is simply to add random failure conditions into the logic diagram developed under Level 4.1A. A reachability calculation on a logic diagram containing random failure conditions will generate a Boolean expression that gives the conditions under which the adversary can succeed.

Logical Progression of Analysis Within Level 4

Just as it is logical to perform Levels 1, 2 and 3 sequentially, there is a precedence ordering for the analysis of the sublevels within Level 4. Not all levels will be active for a given adversary type -- for example, for an outsider who has no special knowledge of the system, a Level C analysis is never appropriate. The question addressed in this section is given that more than one sublevel of analysis is appropriate for a given adversary type, in what order should the analyses be performed, and what does passing or failing a given level mean?

In characterizing the sublevels, it is important to realize that as we increase the computational difficulty, we increase the probability that a given MC&A system will pass. On the other hand, as we increase the computational difficulty by going from Sublevel A to Sublevel B to Sublevel C, we decrease the probability that a given system will pass. In other words, more realistic system models (dynamic assumptions) result

in a less stringent test for the MC&A system while more realistic adversary models (probabilistic assumptions) result in a more stringent test for the MC&A system. Therefore, to minimize computational problems, while subjecting a facility to the most stringent tests, it is logical to pass through each of Sublevels A, B and C that are appropriate to a given problem, while attempting to avoid sublevels 4.2 and 4.3 where possible.

This philosophy is illustrated in Figure 2.9. Analysis of an SNM facility starts at Sublevel 4.1A. If the facility passes this test, there is no need for a Sublevel 4.2A or a Sublevel 4.3A analysis. If the system fails Sublevel 4.1A, Sublevel 4.2A analysis is used to determine whether the failure was due to an MC&A system weakness, or whether it was because the Sublevel 4.1A analysis ignored an important sequencing constraint.

The implications of passing and failing Sublevel 4.2A are similar to those for Sublevel 4.1A. If the system passes 4.2A it means that Sublevel 4.3A is unnecessary; the analysis can move to Sublevel B. If the system fails Sublevel 4.2A, then Sublevel 4.3A is necessary to determine if the system still fails when timing constraints are treated carefully. If the system fails Sublevel 4.3A, then it is uncovered, the facility should be rejected, and the analysis terminates.

Under sublevels B and C, the precedence ordering is similar to that under Sublevel A. Starting with the dynamics described by Sublevel 4.1 and working towards the dynamics of Sublevel 4.3, if the system passes for any dynamic characterization, control passes immediately to the next knowledge level. If the system fails a given dynamic sublevel, then the next most complicated level must be tried to determine if the failure was because of a computational assumption or because of a system weakness. The procedure will eventually terminate by either accepting or rejecting the facility.

Figure 2.9 Precedence Ordering for Sublevels within Level 4.

C. COMPARISON OF STRUCTURED ASSESSMENT APPROACH TO THE AGGREGATED SYSTEM MODEL

The relation between the Structured Assessment Approach (SAA) and the Aggregated Systems Model (ASM) effort by Bruce Judd of ADA, Stein Weissenberger of LLL and others is shown in Figure 2.10. The schematic tree at the top of the page shows schematically the tree that Judd and Weissenberger use to illustrate their model. A description is found in ADA's draft report "Methodology and Preliminary Models for Analyzing Nuclear Safeguard Decisions," July 1978. Current work by Judd and Weissenberger includes expanding the model. Changes found in the expanded schematic tree are shown in the center of Figure 2.10. In our terminology, the nine stages in the ASM schematic tree reduce to the six shown at the center of the figure:

- First, the frequency of attempts is assessed.
- Second, the probability of various adversary types given an attempt are assessed.
- Third, various adversary diversion strategies are postulated.
- Fourth, the probability of detection conditional on attempt, adversary type and adversary diversion strategy is assessed.
- Fifth, the probability of interruption given detection is assessed.
- Sixth, the societal consequences of successful diversion of SNM are evaluated.

The SAA is concerned only with the central portion of the reduced schematic tree. The SAA is conditioned on the fact that an attack by a given adversary has occurred, so that frequency of attempt and probability of the adversary type are not needed. At the other end of the sequence, the SAA stops with the event detection or no detection without considering the probability of interruption or the consequences of diverted SNM.

The adversary diversion strategies of the ASM correspond to the target sets of the SAA. Because the target sets are generated through exhaustive enumeration, the SAA will consider thousands of target sets. In the SAA, diversion strategies are found by explicitly maximizing the

Aggregated Systems Model

Schematic
Tree

July 1978

Schematic
Tree

October 1978

same

same

Structured Assessment Procedure (SAA)

Pre SAA

SAA = ASM

Post SAA

ASM
Relation
to SAA

Figure 2.10 Relationship to Aggregated Systems Model

adversary's expected utility. Diversion strategies are represented as optimal monitor target sets (MTS*) in our terminology.

The probability of detection is also treated in more detail in the SAA than in the ASM. In the ASM, the probability of detection is a judgmental input. In the SAA, on the other hand, the probability of detection is expanded over the set of component availabilities (AV). The probability is then calculated based on component availability data.

Comparing the reduced schematic tree of Figure 2.10 to the schematic trees in Figures 2.3, 2.5, and 2.7 should help to make these comparisons clear.

III. CALCULATION OF THE PROBABILITY OF DETECTION

An algorithm for the exact calculation of the probability of detection was developed for the Structural Assessment Approach (SAA). The calculation is exact on the sense that no approximations are used to calculate the probability for a Boolean expression defining the detection event. The algorithm was applied to several examples, both to determine its capabilities and limitations and to identify specific modeling issues in the SAA. Useful applications include a set of sensitivity studies that demonstrate several MC&A performance measures are possible within the definition of a Level 3 analysis.

The present form of the algorithm is the first step of a development task that will provide an efficient, practical calculation of the probability of detection for large, complex safeguards systems. In this first step, we chose a simple algorithm with limited capabilities that was easily implemented rather than choosing a more complex algorithm with full capabilities that would have required considerable development effort. The simple algorithm has the advantage of avoiding a technical "bottleneck" for computer programming of the full SAA. Refinement of the probability algorithm should greatly improve its efficiency and expand its size limitations.

Section 3.1 reviews the general guidelines that the LLL-ADA project team are using to develop the probability algorithm. The section also summarizes the different approaches to the calculation of the probability of detection. The summary identifies the type of size limitations that we accepted in selecting the current version of the algorithm. Section 3.2 relates the features of the current probability algorithm to the outline of the Structured Assessment Approach presented in Section II of this report. Section 3.3 defines the algorithm and its computer implementation. An example application in this section illustrates the capabilities of the current version. Section 3.4 compares the exact calculation of probability

with an approximate procedure. This comparison illustrates the conditions where the "min cut set" lower bound for system reliability does not approximate the exact calculation. Further, importance measures for components and cut sets are interpreted as probability measures to illustrate their significance to a Level 3 analysis.

A. OVERVIEW OF THE PROBABILITY ALGORITHM DEVELOPMENT

This section first explains the form of the logical expression representing adversary detection by an MC&A system and reviews the general guidelines that we are following to develop the probability algorithm for the Structured Assessment Approach. The current algorithm is then characterized by the degree that it meets these guidelines. The section concludes with a summary of alternate design approaches to a probability algorithm, explaining why we chose the approach we did.

Logical Representation of Detection by an MC&A System

Detection by an MC&A system is described in mathematical terms as a logic expression using Boolean event algebra. The Boolean expression of the detection event describes, in mathematical logic notation, all the possible ways the monitors and other components in an MC&A system will detect an adversary. The logical equation contains symbols (like A, B, ..., 1, 2, ...) for the events that each component works with "+" and "." signs indicating logical OR and logical AND, respectively.

The following simple example is used to illustrate the logical representation of detection by an MC&A system:

This figure is a connection network for a simple detection system. Monitors 1, 7, and 10 can each detect the presence of an adversary. Monitors 1 and 7 both send their signals to component 2, which then sends a signal to component 11. Monitor 10 sends its detection signal directly to component 11. The system will detect if component 11 sends a signal after receiving a signal from any of the monitors. A network of the MC&A system, like this one, can be used to describe detection in various representations of a Boolean expression.

Three different representations of the Boolean expression for the detection event are logically equivalent: the path representation, the cut set representation, the highly-factored representation. Both the path and cut set representation are event equations derived from an MC&A system diagram. The factored representation is derived from either a path or cut set representation. Although these representations are logically the same, each lead to different algorithms for calculating the probability of detection. Each representation is briefly described in the following paragraphs.

A path representation of the Boolean expression for the detection event is constructed from all the ways that various sets of components detect an adversary, (a set of components is called a "path"):

$$\text{Path 1: } P1 = 1 \cdot 2 \cdot 11$$

$$\text{Path 2: } P2 = 7 \cdot 2 \cdot 11$$

$$\text{Path 3: } P3 = 10 \cdot 11$$

$$\text{Detection: } D = P1 + P2 + P3$$

The notation 1, 7, ..., i represents the event that "the component i works." Recall that "+" sign represents a logical OR; a "." sign represents a logical AND. The expression for detection is read as follows: an adversary will be detected if all the components in Path 1 or Path 2 or Path 3 are working. The expression for Path 1 is read similarly: The MC&A system will detect if components 1 and 2 and 11 are working. The other paths are interpreted the same as Path 1. The path representation of the Boolean expression is also known as a "sum of products" or a canonical form. A minimal path representation contains the minimum number of paths that are essential to the MC&A system detection function. The components in the paths that are eliminated can either work or fail without affecting the ability of the system to detect.

A cut set representation of the Boolean expression is constructed from all of the ways that various sets of components prevent detection of an adversary, (a set of these components is called a "cut set"):

$$\text{Cut Set 1: } K1 = \overline{11}$$

$$\text{Cut Set 2: } K2 = \overline{10 \cdot 2}$$

$$\begin{aligned}
\text{Cut Set 3: } K3 &= \overline{1} \cdot \overline{7} \cdot \overline{10} \\
\text{No Detection: } \overline{D} &= K1 + K2 + K3 \\
\text{Detection: } D &= \overline{K1} \cdot \overline{K2} \cdot \overline{K3} \\
&= (\overline{11}) \cdot (\overline{10+2}) \cdot (\overline{1+7+10})
\end{aligned}$$

The notation " $\overline{11}, \overline{10}, \dots, \overline{i}$ " represents the event that the component i does not work. The other notation is the same as in the path representation. The expression for no detection is read as follows: An adversary will not be detected if all components in Cut Set 1 or Cut Set 2 or Cut Set 3 have failed. Conversely, the expression for detection is read as: The adversary will be detected if at least one component is working in Cut Set 1 and Cut Set 2 and Cut Set 3. As an example, the expression for Cut Set 2 is read as: The MC&A system will not detect if component 10 and 2 are not working. The cut set representation of the detection event, also known as a "product of sums", is the counterpart to the path representation. Likewise, a minimal ("min") cut set representation contains only those cut sets that contain the minimum number of components required to fail the MC&A system. The cut sets that are eliminated can cause system failure before all the components in the set have failed.

The highly-factored representation of the Boolean expression is derived from either a path or cut set representation. A highly-factored form uses parenthesis to eliminate or reduce repeated component events in the expression. Using the path representation of the simple network as an example, a highly-factored form of the detection event expression is illustrated:

$$\begin{aligned}
D &= 1 \cdot 2 \cdot 11 + 7 \cdot 2 \cdot 11 + 10 \cdot 11 \\
D &= 11 \cdot (1 \cdot 2 + 7 \cdot 2 + 10) && \text{Factor 11} \\
D &= 11 \cdot [2 \cdot (1+7) + 10] && \text{Factor 2}
\end{aligned}$$

After factoring, the example expression contains only a single reference to each component event. Factoring is most efficient when applied to minimal path or minimal cut set representations of the Boolean expression.

General Guidelines

The algorithm must calculate the probability of adversary detection

during an attempt to divert special nuclear material (SNM), conditional on the type of adversary, facility operating mode, and monitor target set attacked. To be useful, the final version of the algorithm must maximize the number of MC&A system components that can be analyzed without exceeding the practical limits of the LLL computer. The analysis of the probability of detection should include the following three aspects:

- Include probabilistic dependence among events,
- Include sensitivity analysis on the numerical values and algebraic expressions, and
- Provide reduced equations for the Boolean expression representing the detection event.

Most algorithms assume that the component events that determine the detection event are probabilistically independent. This assumption can strongly distort the calculated system reliability if it is not realistic. Normally, the effect of this assumption is reduced by explicitly modeling the important dependencies among events in the analysis.

Relevant sensitivity studies, both on numerical values and on the algebraic expressions, are important for evaluating the robustness of an analysis. Robustness in an analysis means that the solution remains essentially unchanged with variations in the model parameters and expressions.

Finally, a probability algorithm uses a Boolean expression of the detection event to evaluate the probability of detection. Often the algorithm manipulates this expression to provide a reduced form for evaluation. This reduced form is often very useful to an analyst for judging completeness in the analysis and communicating the calculated results.

Development Status

The current version of the SAA algorithm to calculate the probability of detection satisfies each of these guidelines:

- Probabilistic dependence is treated by modeling the deterministic dependence among components, such as shared utilities. However, probabilities of component events are assumed to be independent

- A sensitivity analysis is implemented
- The algorithm reduces equations for the Boolean expression by deriving the minimal form of any path set. (A minimal form has had all logically equivalent and logically dominated paths removed.) However, this reduced expression is not a highly-factored form. (A highly-factored form uses parenthesis to factor a logical expression so individual elements are repeated as little as possible.)

In this initial phase of the development, the dominant consideration was rapid implementation; size limitations were a secondary consideration.

Expanding the size capability and increasing the computation efficiency of a probability calculation are key elements of the next phase of the development.

Basic Algorithm Design Approaches

The challenge of size confronts the development of a feasible and applicable probability algorithm. The goal is to be able to evaluate the reliability of a safeguards system with several thousand components.

Three approaches to developing the algorithm, characterized by the degree of sophistication, are:

- Explicit representation in mutually exclusive form -- a brute force technique
- Path or Cut Set representation and the comparable probability formula -- more efficient but limited to moderate size systems (50-100 components)
- Highly-factored form representation with sophisticated procedures for probability evaluation.

Of three approaches to developing an algorithm, the first two become obviously infeasible for complex systems with many components.

Direct Approach of Explicit Representation

The most direct approach is the explicit representation of the Boolean expression in terms of each event and its negation. This explicit representation provides a mutually exclusive form of the expression that allows a simple evaluation of the detection event. The practical limitation to this approach is the size of computer memory required for storing the representation. The memory requirements rise exponentially with N , the number of components:

$$\text{Number of words of memory} \approx 2^N$$

For a moderate size safeguards system with 60 components, the memory requirements for the Boolean expression exceeds 10^{18} words, exceeding any practical limit on LLL computers. This direct approach is clearly not even feasible for a preliminary algorithm for the Structured Assessment Approach.

Probability Algebra Using Path Representation

An example of the second approach uses the path representation of the Boolean expression and the following probability formula:

$$P(A+B) = P(A) + P(B) - P(A \cdot B)$$

where, for example

$$A = a \cdot b \cdot c$$

$$B = b \cdot c \cdot d \text{ and}$$

$$A \cdot B = a \cdot b \cdot c \cdot d.$$

The labels A and B are paths representing a set of component events that give a detection event. The formula is valid for the two paths even if some components are common to both paths. For a detection event with many paths, repeated use of the two-path formula gives the formula for many paths. Practical limitations identified with this approach are: 1) the amount of computing time required to evaluate the formula, and 2) the memory requirements for defining the components in each path. For moderate size MC&A systems, especially for less than 60 component events, the second limitation is not a problem. The primary limitation is computing

time requirements. Computation time increases approximately exponentially with P, the number of paths:

$$\text{Computation time} \propto 2^P$$

Some numerical experiments with this algorithm have shown that 10 paths require 0.3 seconds of computing time while 25 paths are expected to require about 3 hours.

Highly-factored Form

The third approach is the most sophisticated. The Boolean expression is reduced to a highly-factored form by using parenthesis to repeat component events as little as possible. The algorithm then produces an algebraic formula for the probability of detection. The keys to achieving the development goals with this approach are:

- Factor the Boolean expression to take account of any probabilistic independence among component events
- Eliminate all unnecessary repetitions of component events.

The first step allows the use of simple algebra formulas for calculating large portions of the probability expression. The second factor allows for reduced computer memory requirements for large (~ 1000 components) safeguards systems.

Selection of Approach to Probability Algorithm

The second approach was implemented as the preliminary probability algorithm. This type of algorithm is suitable for moderate size (50-100 components) safeguards systems with less than 15 paths in the canonical form of the Boolean expression. Implementing this algorithm has provided a development tool for identifying modeling issues and a first-cut calculation for the Structured Assessment Approach.

Requirements for an algorithm based on the third approach will be prepared in the next phase of development. This phase will address directly the need for a probability calculation that can efficiently translate a highly-factored Boolean expression and component event probabilities into the probability of detection for a complex, large-scale safeguards system.

B. DESCRIPTION OF THE "EVENT" PROBABILITY COMPUTER ALGORITHM

This section describes the features of the current computer algorithm, EVENT, developed by ADA to calculate exactly the probability of detection from a Boolean expression. These features are first related to the outline of the Structured Assessment Approach presented in Section II. Then the input and output from each feature are briefly summarized with special attention given to the information received from the LLL project team.

Relation of EVENT to SAA

Figure 3.1 repeats Figure 2.6 in the previous section to illustrate the relationship between the computer code and the Structured Assessment Approach. Essentially, LLL supplies both the Boolean expression in canonical form for the detection event and then data on component reliability or availability. EVENT uses the event definition and component data to calculate the probability of detection. The algorithm also includes a feature that accepts LLL information on signal paths, utility paths, and unit models to obtain the event definition. This feature was added to simplify the information requirements on LLL and to speed the implementation of the probability calculation.

Generation of the Boolean Expressions for the Detection Event

This feature of EVENT allows the option of providing signal paths, utility paths and unit models instead of the canonical form of the Boolean expression for the detection event. The expression for the detection event depends both on the functioning of signal components and the utility components providing the MC&A system with electricity, air, or other external services. For example, the availability of several electric-powered monitors are interdependent through the interconnections of the electric network. This interdependence can be modeled explicitly by incorporating the utility networks into the signal network to form a Boolean expression for the detection event. Incorporating the utility network allows the assignment of independent probabilities for the component availabilities.

EVENT

Figure 3.1 Probability Algorithm Relationship to Structured Assessment Approach (see also Figure 2.6 discussion)

Unit models describe the utility connections for each component in the signal path. These models are Boolean expressions that describe the availability events of external utilities. These events are simple statements, such as "electricity is supplied" or "compressed air is available." In lieu of providing the algorithm with the Boolean expression for the detection event, the user has the option of providing these unit models along with signal paths and utility paths. The current version of EVENT will then determine the canonical form of the detection event expression.

EVENT, A Computer Algorithm to Calculate the Probability of Detection

Main Features

The main feature of the ADA computer algorithm is the exact calculation of the probability of detection for a path representation of given Boolean expression and a given set of component event probability data. The user provides the data and the expression. Both the data and the expression represent the specific conditions associated with the adversary type, the monitor target set, and facility operating mode.

Size Limitations

The current version of the algorithm has size constraints that limit its usefulness for assessments of large, complex safeguards systems. The maximum number of components is 60; the maximum number of paths is 13, although this limit can be extended to 20 if required. This algorithm is suitable for assessments of small systems or separate parts of a large system. Section 3.3 gives a technical definition of the solution approach and the algorithm limits. A simple example is also presented to illustrate the output calculated by EVENT.

Sensitivity Calculations

Another aspect of the probability calculation is the option for sensitivity studies. The algorithm can calculate conditional and joint probabilities, which are used to evaluate the significance of component failures. The conditional probability is the probability of detection given that a

specified component has failed. The joint probability is the probability of no detection and a specified component has failed. The conditional and joint probabilities can also be evaluated for the simultaneous failure of two components. These sensitivity calculations were included as possible measures of performance for a Level 3 assessment. As other measures are identified, their computation will be included in the algorithm.

C. TECHNICAL DEFINITION FOR THE CURRENT VERSION OF THE PROBABILITY CALCULATION

This section presents a technical discussion of the solution approach for the two main features of the EVENT algorithm. Figure 3.2 outlines these features with their input and output descriptions. The first feature encountered by the code user is the reduction of signal paths, utility paths and unit models to a Boolean expression. Alternatively, EVENT will accept a path representation of the Boolean expression for the detection event. The second feature of the code is the calculation of the probability of detection given the event definition and component event probabilities. The user then has the option to continue the analysis with sensitivity studies.

Detailed logic flow charts are presented in Appendix A1. The following sections first explain the solution algorithm for the detection event expression and probability calculation. Specific size limits in the code for number of components and number of paths are then described. An explanation on how to relax the restriction on the number of paths is included. Finally, an example network is evaluated to illustrate the capabilities of the algorithm.

Solution Algorithm for Boolean Expression

Use of Unit Models

Paths for the detection in the signal network alone and the utility paths that connect to the signal network are provided by the user. Unit models for each component in the signal paths are also provided. A unit model for a signal path component is a single diagram of the external utilities required for component operation. Figure 3.3 gives an example for a unit model. An important distinction separates an adjacency matrix from this definition of a unit model. In an adjacency matrix, several connections to a single component are, by convention, interpreted as several parallel paths. The connections are treated as logical intersections ("OR"). For the unit model, several connections to the component are interpreted just the opposite, as logical unions ("AND"). This change in convention from the adjacency matrices to the unit model provides an efficient approach to combining the utility and signal paths.

Figure 3.2 Structure of "EVENT"

Figure 3.3
Illustration of a Unit Model

Diagram

Interpretation

Both compressed air and electric power must be supplied to component 1 for the device to operate.

Combination of Signal and Utility Path Sets

These unit models determine the logical relationship between the signal and utility paths. Each component in the signal network is replaced by a Boolean expression representing its unit model. This process expands the number of paths in the detection event definition. Identifying repeated components in individual paths and eliminating logically equivalent and logically dominated paths produce a "minimal path" Boolean expression. This expression represents the detection event in terms of the availabilities of signal flow components and necessary utility network components. Interdependence among signal components through the utility networks is modeled explicitly in the definition of the detection event. An example of this procedure is presented in the latter part of this section.

Solution Approach for Probability Algorithm

An Example Solution

The simple network discussed in Section III.A is used to illustrate the solution approach. In this example, the path representation of the detection event D for the following simple network is the logical sum of the three paths:

The algorithm in EVENT receives a set of paths that define the detection event with each path described by a list of components.

The probability of detection for this example is calculated by the following algebraic formula:

$$P(D) = P(P1) + p(P2) + p(P3) - p(P1 \cdot P2) - p(P1 \cdot P3) - p(P2 \cdot P3) + p(P1 \cdot P2 \cdot P3)$$

The path probabilities are calculated from the product of the component event probabilities input to the code, for example:

$$p(P1) = p(1) p(2) p(11)$$

The code extends this algebraic formula to accommodate as many paths as will fit in the computing system's memory capacity or the analyst has time to wait for. Practical limits on the number of paths are discussed later in this section.

Computer Solution Procedure

Mechanically, this formula is implemented by the procedure outlined in Figure 3.4. Each component has an integer number and a probability, stored in arrays ID and PROB, respectively. Each path is assigned a 60 bit word in the array PATH. Path components are represented by a "1" in a specific bit location within the word. An "0" for a component means that it is not contained in the path. (This program is implemented on Lawrence Berkeley Lab and United Computing Systems Computers, both CDC machines having 60 bit words, thus limiting to 60 the number of components in the system.)

Graphically, the example above would be represented as:

PATH (*)	ID (*)		PROB (*)		P(1)		P(2)		P(11)		P(12)		P(21)		P(22)		P(10)		P(ID #6)		P(ID #7)		...		P(ID #60)	
	1	2	11	7	10	ID #6	ID #7	...	ID #60																	
P1	1	1	1	0	0	0	0		0																	
P2	0	1	1	1	0	0	0	...	0																	
P3	0	0	1	0	1	0	0		0																	

INPUTS

Path structure

Component data

PROBABILITY
ALGORITHM

OUTPUT

Probability of Detection

Figure 3.4 Conceptual Structure of Probability Algorithm

Each path probability is then calculated in Function PATHP as the product of the probabilities of the components corresponding to the 1 bits in the path word. The sum of the path probabilities is stored. The code then continues to find the remaining probabilities in the equation for the probability of detection.

Probabilities for each member of the set of logical products (e.g., $P1 \cdot P2$) with two paths are evaluated. The logical products of two paths are also stored as 60 bit words with either "1" or "0" representing whether or not a component is included in result in the array PTEMP. The probability of each logical product is then calculated by the function PATHP and subtracted from the sum of path probabilities. The logical product not containing path P1 is saved ($P2 \cdot P3$) for generating the next set of logical products. In this example, only one member ($P1 \cdot P2 \cdot P3$) is in this set. The probability of this logical product is then added to the others to yield the exact calculation of the probability of detection.

The General Solution Procedure

A general calculation with N paths repeats these procedures for $N-1$ sets of logical products. The K^{th} set of logical products contains all combinations of paths that can occur taken K paths at a time, excluding the first path. The size of a K^{th} set is determined by the number of ways K paths can be combined out of a total of $N-1$ paths:

$$\begin{array}{l} \text{Number of logical} \\ \text{products stored in} \\ K^{th} \text{ set} \end{array} = \frac{(N-1)!}{K! (N-K-1)!} = C_K^{N-1}$$

To generate the $K+1$ set of logical products, path(1) through path($N-K$) are each used with the K^{th} set of logical products. Path(1) forms the logical product with all members of the set. Path(2) will form a logical product with all but the first ($C_K^{N-1} - C_K^{N-2}$) paths. The path(L), where L is not larger than $N-K$, will form logical products with the same elements in the K^{th} set as path ($L-1$), except for the first ($C_K^{N-L+1} - C_K^{N-L}$) elements. This rule forms a recursive relation that is easily programmed to evaluate the probability of detection for any number of paths within limits of the code.

Program Limits

The current version of EVENT is limited to 13 paths in the probability calculation. More than 13 can be input for path minimization or the combination of signal paths with utility paths. However, no more than 13 paths can begin a probability calculation. EVENT is also limited to no more than 60 component events within the Boolean expression for the detection event. This limit is set by the number of bits in a word for the CDC computer. These component events can have any integer number associated to them, but the total number of events cannot exceed 60.

To increase the maximum number of paths, program storage must increase. The following arrays in the INTEGER specification statement at the beginning of the program listing must change to reflect the increased maximum number of paths (MXPATH):

- PATH(MXPATH)
- DUMMY(MXPATH+1)
- FACTRL(MXPATH)
- PTEMP(C), where $C = \frac{2(MXPATH-1)!}{\left(\frac{MXPATH}{2} - 1\right)! \left(\frac{MXPATH}{2}\right)!}$

for even MXPATH

$$\text{and } C = \frac{2 \text{ MXPATH}}{(MXPATH+1)} \left[\frac{(MXPATH-1)!}{\left(\frac{MXPATH-1}{2}\right)!} \right]^2$$

for odd MXPATH .

However, the limit on the number of components cannot be increased by changing array specifications.

Prior to increasing the limits on the number of paths, keep in mind the associated exponential increase in computing time. Table 3.1 gives the temporary storage required for increased limits on paths and the estimated computation time for evaluating the probability of that number of paths. Note that the time requirements can dominate the memory requirements.

TABLE 3.1

CONSTRAINTS ON NUMBER OF PATHS

Number of Paths	Required Temporary	Estimated Running
	<u>Storage</u> (words)	<u>Time</u> (seconds)
10	252	.3
11	462	.6
12	924	1
13	1,716	2
14	3,432	5
15	6,435	10
16	12,870	20
17	24,310	40
18	48,620	80
19	92,378	~3 minutes
20	184,756	~6 minutes

Example Application

Figure 3.5 shows the simple model of an MC&A system expanded to include the utility network interfaces. The component failure probability or availability is written above the component node in each network. All component event probabilities are equal to .9, representing in this example a reliable safeguards system. These networks are used to illustrate the algorithm. The signal network and utility network will be combined through a set of unit models to determine the canonical form of the expression for the detection event. The probability algorithm will then calculate each path probability and the probability of detection. Sensitivity analysis on the effect of single component failures conclude the example.

Derivation of the Boolean Expression

The first application of the EVENT algorithm is to obtain a Boolean expression for the detection event. The minimal paths for the detection event in the signal network are identified from the figure.

<u>Path No.</u>	<u>Path</u>
1	1 → 2 → 11
2	7 → 2 → 11
3	10 → 11

For larger systems, adjacency matrices would be evaluated by PATH, an LLL computer code. Next, the unit model for each signal flow component connected to the utility network is specified.

<u>Signal Flow Component</u>	<u>Unit Model</u>
1	3,6 (Both 3 and 6 must function)
2	6
10	9

Using the unit model information on the connection between signal and utility networks, the minimal paths from each utility source to each unit interface are identified.

<u>Unit Model Element</u>	<u>Utility Path</u>	<u>Utility Sources {3,4,5,8}</u>
3	3	
6	4→6	
	5→6	
9	5→9	
	8→9	

Figure 3.5

Example Network Description of MC&A Detection

Signal Network

Utility Connection Network

The EVENT code is then run with these inputs to provide a set of minimal paths from the expanded representation of the detection event.

<u>Path No.</u>	<u>Minimal Path Representation of Detection Event</u>
1a	1 + 2 + 3 + 4 + 6 + 11
1b	1 + 2 + 3 + 5 + 6 + 11
2a	7 + 2 + 4 + 6 + 11
2b	10 + 5 + 9 + 11
3a	10 + 8 + 9 + 11
3b	

Calculation of Probability of Detection

With the detection event obtained, the component event probabilities (all equal .9) are input. EVENT calculates the probability of detection for this system to be .86, with each separate path having a path probability of detection as follows:

Path Probability of Detection

Path 1	.53
Path 2	.53
Path 3	.59
Path 4	.59
Path 5	.66
Path 6	.66

Sensitivity Studies

A Level 3 analysis for the SAA requires calculations on the probability of detection given the knowledge of what components have failed. Two different examples of this type of calculation illustrate the flexibility of the current version of the code to provide a Level 3 analysis. Neither of these examples represent a full Level 3 analysis as outlined in Section II.

The first sensitivity measure evaluated is the target set probability of detection given the failure of a single component. The probability analysis sets the probability of detection or availability to zero for a component. The resulting calculation gives the probability of detection conditional on the failure of that component. This calculation

is repeated for each component in the path set. The smallest value of the probability of detection is ranked first. The component rankings using this measure are given in the following table:

<u>Failed Component</u>	<u>Probability of Detection Given Component Failure</u>
11	0.0
10	.71
9	.71
2	.72
6	.72
5	.83
7	.84
8	.84
4	.84
1	.85
3	.85

From this example sensitivity study, the example system has only one real significant component, number 11. The probability of detection given any other component failure is reasonably satisfactory.

The second sensitivity measure evaluated is the probability of the joint event for no detection by the target set and the component failure. Ranking the components by the maximum of this joint probability gives the following result:

<u>Failed Component</u>	<u>P{no detection by MC&A System and component failures}</u>
11	.1
10	.029
9	.029
2	.028
6	.028
5	.017
7	.016
8	.016
4	.016
1	.015
3	.015

This measure of system performance assumes that an adversary trades off an increased probability of no detection for an increased probability of a component failure. Whether or not this tradeoff for an adversary is represented by the maximum of the probability of this joint event depends on the adversary model.

In a target set with all components of equal reliability or availability such as this example, distinctions among sensitivity analyses are not important. However, other sensitivity studies have highlighted the differences that can be obtained with changes in component event probabilities. Whatever the measure of performance selected for the Structured Assessment Approach of safeguards systems, the purpose of this example is only to illustrate that the algorithm is flexible enough to calculate any necessary sensitivity measure.

D. COMPARISON OF SAA PROBABILITY CALCULATIONS WITH THE "IMPORTANCE" CODE

In this section, the exact calculation of the probability of detection is compared with the, "min cut set" lower bound calculated by the IMPORTANCE code. This comparison illustrates the conditions under which the "min cut set" approximation becomes unsatisfactory, even for highly reliable systems. Further, the probabilistic importance measures used for sensitivity analyses by IMPORTANCE are reviewed. This review indicates that these measures are useful for system design diagnosis but not useful definitions of system performance measures for the Structured Assessment Approach.

Min Cut Set Approximations

A min cut set approximation gives a lower bound to the probability of detection and, correspondingly, an upper bound for the probability of no detection. Since the exact calculation of the probability of detection for a large MC&A system requires significant development, the value of this approximation is important to clarify.

The min cut set approximation can be illustrated with the simple MC&A system described in Section III.A. The minimal cut set representation for the detection event is used for the probability calculation:

$$D = \overline{K1} \cdot \overline{K2} \cdot \overline{K3}$$

The notation is the same as in the example in III.A where " $\overline{K1}$, $\overline{K2}$, $\overline{K3}$ " represent the events that the cut sets do not occur, i.e., that at least one component in the cut set works. The probability of detection is then the probability of the joint event that at least one component in each of the three cut sets is working:

$$p(D) = p(\overline{K1} \cdot \overline{K2} \cdot \overline{K3})$$

The min cut set approximation assumes that these three events are probabilistically independent:

$$p(D) = p^*(D) = p(\overline{K1}) p(\overline{K2}) p(\overline{K3})$$

This assumption is not generally valid, since the cut sets often have components in common. The simple example illustrates the nature of the error made by this assumption. The probability that Cut Set 1 does not occur is

independent of the probability that both Cut Set 2 and 3 not occur.

$$p(D) = p[1\bar{1}] p[(\bar{10} + \bar{2}) \cdot (\bar{1} + \bar{7} + \bar{10})]$$

However, the probability of the joint event that Cut Set 2 and 3 not occur is not equal the product of the probabilities for each cut set event, since both cut sets contain component 10:

$$p[(\bar{10} + \bar{2}) \cdot (\bar{1} + \bar{7} + \bar{10})] \neq p[(\bar{10} + \bar{2})] p[(\bar{1} + \bar{7} + \bar{10})]$$

After some algebra this expression is expanded to give the following:

$$1 - p[2 \cdot 10] - p[1 \cdot 7 \cdot 10] + p[1 \cdot 2 \cdot 7 \cdot 10] \neq 1 - p[2 \cdot 10] - p[1 \cdot 7 \cdot 10] + p[2 \cdot 10] p[1 \cdot 7 \cdot 10]$$

By eliminating equal terms from both sides and assuming that the component event probabilities are independent, the error in the approximation becomes apparent:

$$p(1)p(2)p(7)p(10) \neq p(2)p(10)p(1)p(7)p(10) \\ p(10) \neq p(10)p(10)$$

Note that the example illustrates that min cut set approximation gives a lower bound to the exact calculation of the probability of detection, since

$$p(D) - p^*(D) = 1 - p(10) > 0$$

The following paragraphs examine the usefulness of this approximation. A model representing a complex MC&A system is used to demonstrate that the error can be significant for certain types of system designs.

Experience with the Approximation

The condition generally given for replacing an exact calculation with the min cut set approximation is when the reliability of a system exceeds 0.9 (H.E. Lambert and F.M. Gilman, The IMPORTANCE Computer Code, UCRL - 79269, March 14, 1977). This condition should be representative for the MC&A systems that will be evaluated by the Structured Assessment Approach. The question remains whether the condition of a highly reliable system is sufficient for choosing the approximation over the exact calculation.

The current engineering experience is with MC&A systems where the failure of only a few components prevents the detection of an adversary.

This characteristic of the system reliability can be modeled according to whether 1 or 2 out of 3 significant components are working. Significant components are identified by whether or not these component reliabilities are dominant contributions to the exact calculation for the probability of system failure. For this type of system with few significant components, the min cut set approximation is close to the exact calculation of the probability of detection. However, no comparable experience is available with highly reliable safeguards systems having more than a few significant components. The following analysis examines the relative error of the approximation as the number of significant components is increased.

Exact and Approximate Probability Formulas for K of N Type Systems

A way to generalize experience with MCSA systems is to draw the analogy with systems where any K of N components must function for the system to detect an adversary. Current experience is with systems where the number of significant components, N, is small. We expect future systems to have many more significant components.

The following comparison of exact and approximate probability calculations assumes that a highly reliable system of N identical significant components fails to detect with about 10^{-3} probability. The equation for the exact calculation of system unreliability follows:

$$Q^E = \sum_{r=0}^{K-1} C_r^N p^r (1-p)^{N-r}$$

where Q^E = system unreliability, the probability of no detection

p = component reliability, the probability of working

$$C_r^N = \frac{N!}{r!(N-r)!}$$

Setting the component reliability to $p=.9$ and using the above equation with $Q^E=10^{-3}$ determines the value of K. Since K must be an integer, Q^E can be only approximately 10^{-3} .

The corresponding equation for the min cut set approximation is given as follows:

$$Q^{UB} = 1 - \{1 - (1-p)^{N-K+1}\}^C {}^N_{N-K+1}$$

where Q^{UB} = min cut set upper bound approximation for system unreliability

The relative error used in the comparison of the exact and approximate calculation is the ratio Q^{UB}/Q^E .

Comparison of Exact and Approximate Calculations for Large N

Figure 3.6 gives the range of the relative error made by the min cut set approximation for large K of N systems. As expected from the current project experience, the unreliability of small N systems have little relative error. As a system grows more complex and the number of significant components, N, increases, the relative error can be over an order of magnitude. For large N, the min cut set approximation overestimates the system unreliability by more than a factor of 10. In other words, the min cut set approximation says the system is 10 times less safe than it actually is.

This result supports the need for continued development of an algorithm for the exact calculation of the probability of detection. Realistic, well-designed MC&A systems are expected to be highly reliable and complex, where the min cut set approximation can significantly underestimate the probability of detection for such a system.

Can IMPORTANCE Be Used For Level 3 Analysis?

Level 3 analysis described in Section II answers the question: What chance of detection will adversaries face if they know the status of the MC&A system availability? The measures of system performance for this analysis are based on the calculation of the probability of detection conditional on knowing the system status. As a result of these calculations, important components and sets of components are singled out.

The LLL computer code IMPORTANCE also calculates importance measures for components and sets of components. To judge whether these calcula-

Definitions:

Q^{UB} : Upper Bound of K of N system unreliability calculation

Q^E : Exact K of N system unreliability calculation

Q^{UB}/Q^E : Relative error

Figure 3.6 Relative Error in "Min Cut Set" Upper Bound
Approximation for K of N System Unreliability

tions should be used for Level 3 analysis, the questions answered by IMPORTANCE are contrasted with the questions answered by the Structured Assessment Approach.

Figure 3.7 and Figure 3.8 define the Fussell-Vesely cut set and component measures calculated in IMPORTANCE. These definitions are developed in more detail in the latter part of this section. In general, these measures answer the question: What is the chance that a cut set (or component) caused the failure given that the system has already failed? This is the type of question that a system designer asks in order to evaluate what components are most important for further design effort. For an adversary, the question of what caused the MC&A system failure is unimportant; only that the system failed is important. The Level 3 analysis evaluates the MC&A system effectiveness by using the adversary's decision, not the engineer's decision. With its different perspective, the calculations in IMPORTANCE cannot be used for Level 3 analysis in the Structured Assessment Approach.

Fussell-Vesely Cut Set Importance

Figure 3.7 derives the cut set importance measure from a set of event definitions. The measure represents the conditional probability that a specific cut occurred given that the system failed to detect an adversary. Since the cut set is a set of components, the measure is the probability that the failure of a set of components (the cut set) caused the system to not detect conditional on the system failure.

The conditional probabilities used in the SAA are not conditioned on system failure. Rather, the measures of system performance identified in Section II use the probability that the system will fail to detect given that a set of components fail.

Fussell-Vesely Component Importance

Figure 3.8 derives the component importance measure from event definitions in the same manner as the cut set importance measure. The component

Definitions:

$\bar{D}$ = failure to detect adversary

K_j = min cut set j occurs

$p(K_j)$ = probability that min cut set j occurs
 $= \prod_{i \in j} q_i$, the product of the probability
of failure for all components
 i in the cut set j

$p(\bar{D})$ = probability of the failure to detect the
adversary

$= 1 - \prod_j \{1 - p(K_j)\}$

$p(K_j | \bar{D})$ = probability that the min cut set j
occurred given that the system fails to
detect the adversary

Derivation of the Fussell-Vesely Cut Set Importance Measure

$$p(K_j | \bar{D}) = \frac{p(\bar{D} | K_j) p(K_j)}{p(\bar{D})}$$

$$= \frac{p(K_j)}{p(\bar{D})}, \quad p(\bar{D} | K_j) = 1, \text{ since the system}$$

is certain not to
detect given that cut
set j occurred

Figure 3.7 Event Definition for Fussell-Vesely
Cut Set Importance Measure

Definitions: (in addition to those defined in Figure 3.7)

$K(i)$ = any min cut set occurs that contains component i

$p\{K(i)\}$ = probability that any min cut set occurs containing component i

$$= 1 - \prod_{i \in K_j} \{1 - p(K_j)\}$$

where $i \in K_j$ means component i is contained min cut set j .

$p(K(i)|\bar{D})$ = probability of $K(i)$ given that the system failed to detect an adversary

Derivation of the Fussell-Vesely Component Importance Measure

$$p(K(i)|\bar{D}) = \frac{p\{\bar{D}|K(i)\} \cdot p\{K(i)\}}{p(\bar{D})}$$

$$= \frac{p\{K(i)\}}{p(\bar{D})} ; \quad p\{\bar{D}|K(i)\} = 1, \text{ since the occurrence of any min cut set containing the component } i \text{ is defined to cause a failure to detect.}$$

Figure 3.8 Event Definition for Fussell-Vesely Component Importance Measure

importance represents the probability that the specific component contributed to the known failure of the system. This calculation is a post-mortem measure of the system performance like the cut set importance measure. The component importance measure is not suitable for an SAA Level 3 performance measure for the same reason the cut set importance is unsuitable: Level 3 analysis uses the probability of detection given the failure of a component.

Summary of the Comparison

The min cut set approximations used by IMPORTANCE may not be suitable for the evaluation of well-designed MC&A systems. Only experience with the exact calculation of the probability of detection for specific designs will be the judge. For now, however, a simple calculation demonstrates that significant error may occur in evaluating the probability of detection for large, complex MC&A systems.

The importance measures calculated by IMPORTANCE were found to be different than the definitions of Level 3 system effectiveness. The Fussell-Vesely measures in IMPORTANCE represent the probability that component failures caused the known failure of a system, an importance measure for a design engineer's evaluation. Since the Level 3 analysis is based on the adversary decision, not the designer, IMPORTANCE calculations cannot be used for the Structured Assessment Approach.

IV. ADA TECHNICAL MEMORANDA

This section summarizes the technical memos that ADA has previously submitted to the LLL project team.

1. "Algorithm for Calculating Event Probability from a Directed Graph" by S.L. Derby, June 14, 1978.

This memo outlined an approach to a calculation for probability of detection. The algorithm in the code EVENT grew out of this initial definition of the problem.

2. "Implications from Two Definitions of the Level 3 Criterion" by S.L. Derby, September 22, 1978.

This memo reported on a modeling issue in the definition of a Level 3 criterion. The issue was revealed when two proposed sensitivity studies, calculated by EVENT, gave separate measures of the system effectiveness. The definitions given in Section II of this report developed from the considerations explained in the memo.

3. "Procedure for Combining Utility Networks with Signal Flow Networks" by S. L. Derby, September 22, 1978.

This memo described the procedure used in the EVENT code to develop a Boolean expression from unit models, utility paths, and detection signal paths.

4. "Characterization of Analytical Procedures to Calculate the Probability of Successful Diversion of SNM" by T.R. Rice and S.L. Derby, September 25, 1978.

This memo briefly reviewed the various approaches and approximations to the calculation of the probability of detection by a safeguards system.

5. "User Experience with the BKY Computer" by S.L. Derby, September 28, 1978.

This memo described the difficulties and delays encountered by the ADA project team in developing and in running the EVENT code on the BKY computer operated by Lawrence Berkeley Laboratory Computer Center.

REFERENCES

1. Sacks, I., Parziale, A., Schrot, M., Long, J., "A Structured Approach to the Assessment of Material Control and Accounting Systems", MC 78-203, LLL, March, 1978.
2. Parziale, A., "Analysis of Network Connectivity/Separability, and the Structural Integrity of a Safeguards Communication System," MC 78-289, LLL, April 28, 1978.
3. Parziale, A., "Determining Dominant Paths In A Network and Their Significance In Analyzing the Safeguarding of Adversary Area Traversal," MC 78-301, LLL, May 2, 1978.
4. Parziale, A., "Analysis of Utility Networks and their Significance In Identifying Vulnerability in a Safeguards Communication System," MC 78-403, LLL, May 31, 1978.
5. Parziale, A., "Modeling Adversary Tampering of a Safeguards System With a Petri Net," WM 78-514, LLL, June 30, 1978.
6. Sacks, I., "Contents of Subroutine PROB", MC 78-689, LLL, August 15, 1978.
7. Sacks, I., Parziale, A., "Unit Models," MC 78-884, LLL, September 20, 1978.
8. Lambert, H.E., "Will Anybody Listen? or Computation of Network Reliability from Directed Graphs," MC 78-854D, LLL, September 12, 1978.
9. Lambert, H.E., Gilman F.M., "The IMPORTANCE Computer Code" UCRL-79269, March 14, 1977.
10. Sacks, I., "Sequence Categories," MC 78-14, LLL, January 13, 1978.
11. Sacks, I., "Techniques for the Determination of Potential Adversary Success With Tampering (Level 4.1)," MC 78-928-D, LLL, October 17, 1978.

Figure A1
EVENT Algorithm Flow Chart

Appendix A1

The EVENT Algorithm

This appendix gives a more detailed description of the probability evaluation algorithm, EVENT. A logical flow chart for the calculation is shown in Figure A1.

"Work performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore Laboratory under contract number W-7405-ENG-48."

Reference to a company or product name does not imply approval or recommendation of the product by the University of California or the U.S. Department of Energy to the exclusion of others that may be suitable.

NOTICE

"This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Department of Energy, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately-owned rights."