(12)

TECHNICAL REPORT TR 80-4-97

RSCREEN AND OPGEN: TWO PROBLEM STRUCTURING DECISION AIDS WHICH EMPLOY DECISION TEMPLATES

DECISIONS AND DESIGNS INCORPORATED

Jonathan J. Weiss
Clinton W. Kelly, III

October 1980

ARPA

Command & Control
Decision & Forecasting
Systems Program

DEFENSE SCIENCES OFFICE • CYBERNETICS TECHNOLOGY DIVISION
DEFENSE ADVANCED RESEARCH PROJECTS AGENCY
Office of Naval Research • Engineering Psychology Programs

12

TECHNICAL REPORT TR 80-4-97

RSCREEN AND OPGEN: TWO PROBLEM STRUCTURING DECISION AIDS WHICH EMPLOY DECISION TEMPLATES

by

Jonathan J. Weiss and Clinton W. Kelly, III

Prepared for

Defense Advanced Research Projects Agency
Arlington, Virginia 22217
DARPA Order No. 3668

Monitored by

Office of Naval Research
Arlington, Virginia 22217
Contract N00014-79-C-0068

October 1980

DECISIONS and DESIGNS, INC.
Suite 600, 8400 Westpark Drive
P.O.Box 907
McLean, Virginia 22101
(703) 821-2828

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

14

1. REPORT NUMBER: TR-80-4-97
2. GOVT ACCESSION NO.: AD-A094784
3. RECIPIENT'S CATALOG NUMBER:
4. TITLE (and Subtitle): RSCREEN AND OPGEN: TWO PROBLEM STRUCTURING DECISION AIDS WHICH EMPLOY DECISION TEMPLATES.
5. TYPE OF REPORT & PERIOD COVERED: Technical Report
6. PERFORMING ORG. REPORT NUMBER:
7. AUTHOR(s): Jonathan J. Weiss, Clinton W. Kelly, III
8. CONTRACT OR GRANT NUMBER(s): N00014-79-C-0068, DARPA Order-3668
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Decisions and Designs, Inc., Suite 600, 8400 Westpark Drive, P.O. Box 907, McLean, VA 22101
10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: DARPA Order No. 3668
11. CONTROLLING OFFICE NAME AND ADDRESS: Director, Defense Advanced Research Projects Agency, 1400 Wilson Boulevard, Arlington, VA 22217
12. REPORT DATE: Oct 1980
13. NUMBER OF PAGES: 113
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): Director, Engineering Psychology Programs, Office of Naval Research, 800 North Quincy Street, Arlington, VA 22217

12 | 120

15. SECURITY CLASS. (of this report): UNCLASSIFIED
15a. DECLASSIFICATION/DOWNGRADING SCHEDULE:
16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Decision theory
Decision analysis
Decision aids
Decision making
Mathematical modelling
Crisis management

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Decision templates are prestructured procedural aids for evaluating various alternative courses of action available to a decision maker. This report describes two computer-based decision aids which are structured using decision analytic methodology.

The first aid, RSCREEN (Rapid SCREENing of decision options), is an interactive computer software program that assists decision makers in a crisis situation characterized by severe constraints on time, detailed information →

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

390664

processing procedures, and training in evaluating several different courses of action. Each decision template in the RSCREEN aid encompasses the critical elements of the political-military situation in much the same manner as a well-developed contingency plan. For example, if a decision maker is faced with the immediate evacuation of diplomatic personnel from a hostile foreign country, he/she can draw on the RSCREEN aid for relevant data on evacuation procedures in general, and to assess his/her particular problem in light of these previous evacuation procedures.

The objective of the second aid, OPGEN (OPtion GENeration), is to provide decision makers with the capability to construct, store, retrieve, exercise, and modify OPGEN decision models. Given a specific decision problem based on user inputs, the system identifies the course of action having the least expected regret. OPGEN is designed to be of use when a given generic problem occurs repeatedly, in a variety of contexts and conditions, but with sufficient similarity to justify the use of a single set of attributes and events, and a single group of generic options.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

FIGURES

TABLES

RSCREEN AND OPGEN: TWO PROBLEM STRUCTURING DECISION AIDS WHICH EMPLOY DECISION TEMPLATES

1.0 INTRODUCTION

The most problematic phase of decision analysis is the early stage in which a formal model is structured to represent a real-world decision problem. Because there are no set procedures which will guarantee successful model structuring, the difficulty of this phase is greatly magnified when an attempt is made to replace the trained professional decision analyst with a computer-based decision aid. The aid may easily duplicate or surpass the analyst in performing the decision-analytic calculations and in displaying results, sensitivity analyses, and so forth; but only with great difficulty could a pre-canned, general routine be designed to cope with the structuring of models to fit any arbitrary problem.

As an alternative to the construction of "ad hoc" problem structuring decision aids, this report describes two approaches which use generic "decision templates" to guide the structuring of certain classes of problems at a level more general than one or a small number of specific problems, but less general than the set of all possible decision problems. The first such approach, called RSCREEN (Rapid SCREENing of decision options), uses pre-canned templates to represent a multi-attribute utility structure for a generic problem area (e.g., "evacuation of U.S. citizens from foreign territory"); it incorporates into the templates that portion of the information which characterizes nearly all problems within the area, and then assesses the situation-specific data from the decision maker who uses the aid. The second approach, OPGEN (OPtion GENeration), represents a template approach to decisions under uncertainty involving multiple criteria, where the templates incorporate information about the available options, the uncertain events, and the criteria for evaluating outcomes, at a generic level (i.e., without reference to a specific problem context or geographic location).

The remainder of this report consists of four chapters. Chapter 2.0 contains an introduction to the problem of model structuring in decision aids; a decision-analytic framework for the design of a decision aid system; a discussion of the major technical issues involved in decision aid design; and a brief survey of the current range of efforts at computer-assisted decision problem structuring. Chapter 3.0 contains a technical discussion of the principles behind the RSCREEN approach, and a procedural description of an experimental implementation of the RSCREEN concept on an IBM 5110 portable computer. Chapter 4.0 contains a similar analysis of the OPGEN concept and its experimental implementation on the IBM 5110. Finally, Chapter 5.0 presents a discussion of the conclusions reached during the research effort, together with a set of guidelines and recommendations for future research and development in the area of decision aiding.

2.0 COMPUTERIZED AIDS TO DECISION PROBLEM STRUCTURING

2.1 Decision Problem Structuring

"Good order is the foundation of all good things."
-Edmund Burke

Decision analysis aids human judgment by representing a complex problem in terms of an explicit, consistent, quantitative framework, or model. The elements of the problem are structured--mapped into options, events, subsequent acts, and outcome evaluation criteria--so that a well-defined numerical procedure can integrate the various factors, risks, and constraints into an overall index of preference or utility. Although the numerical operations involved in eliciting values, probabilities, and importance weights, and in determining from these a set of overall utility scores, are quite straightforward, the first step--constructing a meaningful, logical, and realistic model--remains more an art than a science.

Until very recently, applied decision analysis has more or less demanded the participation of a professional decision analyst, particularly during the problem-structuring phase. While a number of texts are available to impart the formal techniques of decision analysis (Raiffa, 1970; or Brown, Kahr, and Peterson, 1974, for example), the professional decision analyst must supplement this basic knowledge with three important ingredients: intuition, experience, and substantive knowledge. Those three intangibles assume such great importance in the problem-structuring phase precisely because there is no sure-fire structuring routine, only a few guiding principles and a limited amount of "clinical" literature.

What practitioners of decision analysis say about problem structuring supports this view: Humphreys says, "Decision analysts have been uncharacteristically vague in specifying elicitation instructions to be used" (1979a), "but practitioners forced to think about the problem have tended to draw on analogies with problems facing psychoanalysts and clinical psychologists" (1979b). Brown, Kahr, and Peterson (1974) note that the role of a decision analyst "is not too dissimilar to that of a psychoanalyst," while commenting that "the difference between a 'good' and a 'poor' decision very often results from the quality of this preformal part of the analysis." They further state, in describing the development of skills in a decision analyst, that knowledge of the mechanics of decision analysis, i.e., carrying out the formal operations of modeling, elicitation, and calculation, is not sufficient: "It is almost as if we explained to a musical novice the meaning of musical notation and how the keys on the piano work and expected him to play Beethoven's 'Moonlight Sonata'!"

Leal (1976) echoes this theme: "The translation process consists of discussions and interviews as well as attempts to educate the decision maker as to the types of information he is to supply. It often requires a special insight and ingenuity on the part of the analyst to direct the conversation and phrase the queries in a way that would yield both informative and reliable responses." Fischhoff (1977) states that a decision analyst's set of techniques "still constitutes something of a bag of tricks whose use requires the judicious application of clinical judgment," and later that "decision analysis may help a decision maker simply because the analyst's bedside manner helps the decision maker focus attention and resources on the problem and not because of specific techniques and axiomatic justification in their armamentarium."

All of the preceding citations strongly suggest that there is no unique way to structure a decision-analytic model--as Keeney and Raiffa (1976) said, "The spirit is one of Socratic discovery"--nor any way to guarantee the optimality of the resulting structure. Howard (1979) says, "in perhaps no other form of analysis is it so easy for the analyst to produce any result he likes." And Humphreys (1979a), referring to the heuristics used by decision analysts, says, "None of these techniques of course are grounded in any axiomatic theory of preferences. They are all able to elicit structural material that would not have been volunteered without their use, and it is possible to check coherence of the structures so generated. However, there is no guarantee that the resulting coherent structures are in any way optimal." Finally, Bowen (1978) states, "We have no clear guidelines, certainly no explicit ones that tell us what to do and what not to do."

The preceding observations by a number of eminent decision analysts suggest that in carrying out problem structuring, decision analysts act like psychoanalysts and, like psychoanalysts, employ heuristics that have no normative base yet facilitate the elicitation of structural material that otherwise would not spontaneously arise. Buede (1979) suggests that the kinds of heuristics chosen depend at least in part on the decision analyst's view of his work as an engineering science or as a clinical art. In the former case, the structuring heuristics are primarily intended to provide grist for an analytic mill, whereas in the latter instance the structuring process may very well be an end in its own right. That is, those who practice decision analysis as a clinical art spend a much greater percentage of their time on problem structuring and find that oftentimes this is sufficient to develop a solution to even the most complex decision problems, with only minimal dependence on the precise quantification and calculation of values, which may serve primarily as a consistency check or "reality test."

Despite the variety of "philosophies" of decision analysis in general, and of problem structuring in particular, a number of heuristic approaches seem to be generally useful to practitioners. A partial list of such heuristic strategies might include the following:

(1) Apply substantive knowledge derived from previous experience with similar problems, using generalization and analogy to develop tentative structures.

(2) Begin with a greatly simplified model of the situation, derive a tentative set of results, and use exploratory methods such as sensitivity analysis, "reality testing" (comparison with the substantive expert/decision maker's intuitive view of the problem), and memory-searching heuristics to identify likely candidates for further model development; iterate until a satisfactory level of detail and completeness has been achieved.

(3) Use secondary decomposition to subdivide existing elements of a model into subfactors which may serve to explicate the more general terms, while replacing one difficult assessment by a few simple ones.

(4) Use the results of a preliminary analysis to focus on a few options which seem to be superior to the rest, but hardly distinguishable from one another; develop more variations on these promising options, and perform a secondary analysis, designed to provide better resolution among this "richer" selection of choices.

In Chapters 3.0 and 4.0 of this report, we describe two prototypical decision problem structuring aids which incorporate heuristics like these with the ultimate goal of providing the untrained user with a computerized surrogate decision analyst. As might be expected, the goal of replacing the professional decision analyst with something "better" is unrealistic within the limits of present-day technology; however, in a number of situations where availability or cost render the analyst inaccessible, or where organizational or security considerations make his participation in a decision impractical, the best alternative may be to construct and employ a computerized aid with the ability to help in problem structuring. Before describing the two aids which constitute the main focus of this report, we shall present a brief history of the evolution of computer-based decision aids as a background.

2.2 An Operational Context for Computer-Based Decision Aids

Recall that our primary goal in developing decision aids is to provide decision-analytic support in the absence of any professional decision analyst (a much weaker constraint than providing better support than a professional decision analyst could if available). Thus, a baseline for comparison would be the status quo approach to decision making, which can in general be characterized as intuitive, nonquantitative, and nonautomated. If a decision aid system is to be designed to improve upon this mode of operation, it may select a particular subset of a client's decision problems for computer-based modeling and then, as more resources are made available for the effort, either extend the range or improve the quality of decision aid applications.

For any given set of problems, a wide range of levels of performance may be defined, of which the following six constitute a representative sample:

- Level 1 - No decision aid; status quo methods.

- Level 2 - A small number of pencil-and-paper forms which offer procedural guidance on a few specific problem types; numerical calculations done manually or on calculators.

- Level 3 - A computerized aid with a library of "pre-canned" models for specific problems; similar in structure to the pencil-and-paper forms, except that more situations can be conveniently accommodated, and calculations will be done automatically, thus providing the capacity to handle larger, more detailed models.

- Level 4 - A computerized aid with a library of generic models, each of which can be "fine-tuned" to fit the specific problem at hand much more closely than the "pre-canned" models; although greater program sophistication and greater user effort are involved, the added flexibility could improve the quality of modeling as well as extend the range of applications beyond that of a few rigid models.

- Level 5 - A higher-order generic decision aid, which allows the user to interact with the computer in a structured way to select from a wide variety of generic models, each of which can then be "fine-tuned."

- o Level 6 - A completely general decision aid, capable of structuring a model purely by interacting with the user (and possibly with a general-purpose data base), without mediation by any prestored model prototypes, whether specific or generic; this kind of system would synthesize a structure, whereas the first five levels represent more analytic methods.

It may be fairly assumed that as the level of performance along this scale rises, the resource commitment required for research, development, implementation, training, maintenance, and operation will rise also. In particular, the magnitude of the jump from Level 4 to Level 5, and even more so that from Level 5 to Level 6, represents a major increment in resources, particularly on the front end. Thus, in general, an intermediate level of performance will be best from a cost-benefit viewpoint and in many cases, even from a benefit-only viewpoint.

Now, at a more global level, we can consider the design of a decision aid system for a particular user population. If we can somewhat arbitrarily partition the entire set of problems encountered into a number of "problem areas," then any particular decision aid system can be scored as a composite of the degree of improvement it induces in each of the problem areas, weighted according to their importance and their frequency of use. The index of improvement within each problem area is itself a function of two factors: the percentage of cases in that area for which the decision aid is appropriate; and the degree of improvement in speed, quality, user satisfaction, or other criteria, when the aid is appropriately used. That index can in general be expressed on a 0-to-100 scale, where a score of zero indicates no improvement at all in the given problem area, and a score of 100 indicates that 100% of all the problems in that area would be improved to a more or less "ideal" or "gold-plated" level of performance.

Within this framework, we can now consider the problem of allocating resources among the various problem areas. As each additional increment in resources is added to the overall system design, it can be used to develop an aid for a new problem area, or to extend the range of application of an existing aid, or to improve quality for an existing aid. It can be demonstrated that the most efficient design for any given resource level can be constructed by starting at the baseline, and adding or upgrading system performance in order of the highest ratio of incremental improvement to incremental resources required. Once the entire set of "cost-optimal" designs has been determined by this procedure, all that remains is to determine which level of resource commitment represents the point of "diminishing returns," in comparison with the other potential uses of the same resources.

Here is a hypothetical example of the design procedure described above. Suppose a military command has three major problem areas of interest: (1) deciding what to do when it seems likely that U.S. citizens will need to be evacuated from a foreign location; (2) deciding how to react when American citizens are being held hostage by terrorists in a foreign country; and (3) deciding what to do in the event of a theft of nuclear materials. Within each of these problem areas, a variety of approaches could be taken, corresponding to the six levels of sophistication defined above.

Table 2-1 illustrates a hypothetical set of resource costs and improvement benefits for each level within each area (costs in thousands of dollars); the "within criterion weights" represent the experts' assessment of the relative importance/frequency for the three areas. Figure 2-1 selects a proposed package with generic models (Level 4) for the

VARIABLE 1: P.A. 1: EVACUATION

		COST	IMPROV	TOTAL
1	NO DECISION AID	0	0	0
2	PENCIL-AND-PAPER	10	5	5
3	PRE-CANNED MODELS	40	25	25
4	GENERIC MODELS	60	75	75
5	HIGHER-ORDER GENERIC	200	95	95
6	GENERAL-PURPOSE AID	500	100	100
	WITHIN CRITERION WEIGHTS		45	

VARIABLE 2: P.A. 2: HOSTAGES

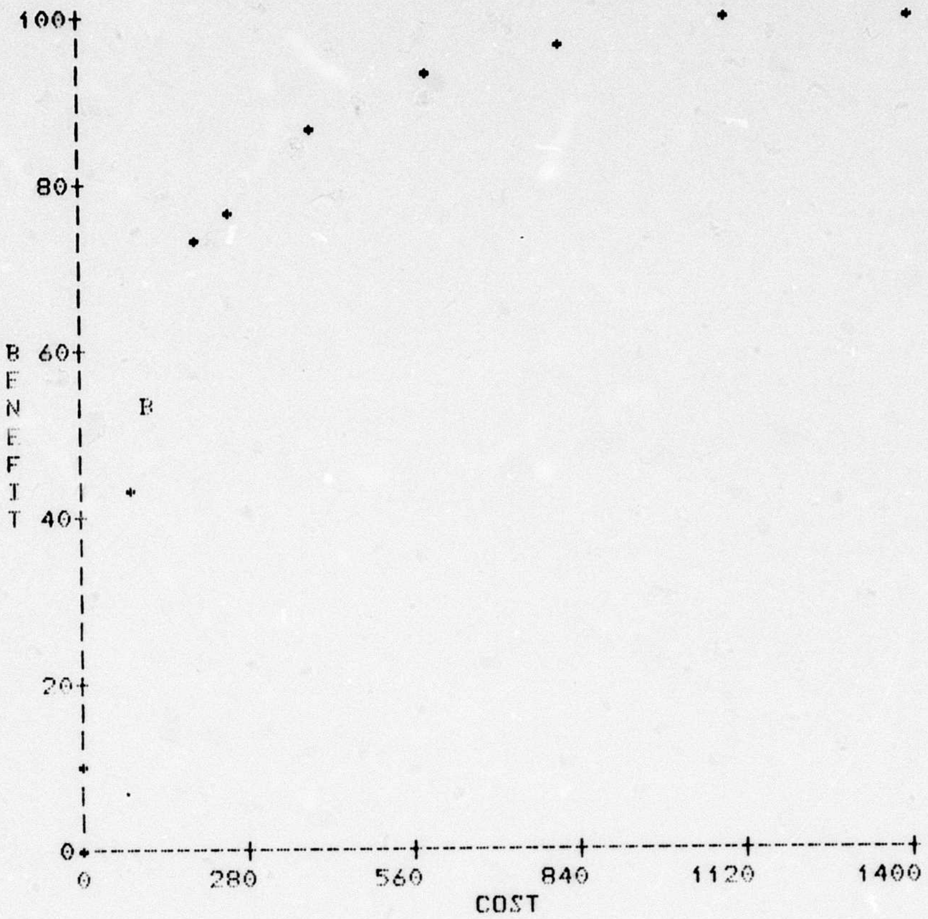
		COST	IMPROV	TOTAL
1	NO DECISION AID	0	0	0
2	PENCIL-AND-PAPER	12	25	25
3	PRE-CANNED MODELS	50	50	50
4	GENERIC MODELS	90	80	80
5	HIGHER-ORDER GENERIC	300	98	98
6	GENERAL-PURPOSE AID	600	100	100
	WITHIN CRITERION WEIGHTS		35	

VARIABLE 3: P.A. 3: NUC. THEFT

		COST	IMPROV	TOTAL
1	NO DECISION AID	0	0	0
2	PENCIL-AND-PAPER	10	2	2
3	PRE-CANNED MODELS	25	5	5
4	GENERIC MODELS	35	50	50
5	HIGHER-ORDER GENERIC	90	80	80
6	GENERAL-PURPOSE	300	100	100
	WITHIN CRITERION WEIGHTS		20	

Table 2-1

ASSESSED VALUES



	VARIABLE	BENEFIT	WTS	COST	LEVEL	
1	P.A. 1: EVACUATION	337	450	60	GENERIC MODELS	(4 OF 6)
2	P.A. 2: HOSTAGES	87	350	12	PENCIL-AND-PAPER	(2 OF 6)
3	P.A. 3: NUC. THEFT	100	200	35	GENERIC MODELS	(4 OF 6)
		525		107		

Figure 2-1
PROPOSED PACKAGE

Evacuation and Nuclear Theft problem areas, and paper-and-pencil aids (Level 2) for the Hostages problem area, thus achieving 52.5 percent of the maximum possible benefit, at a cost of only $107,000 (whereas the full benefit would cost $1,400,000); this package is in fact among the set of "cost-efficient" packages, and is represented on the cost-benefit curve by the point "B." The small circles represent other points in the cost-efficient set; Table 2-2 indicates the series of cost-efficient increments which generate the cost-efficient set of packages (beginning at the top and reading right to left).

Naturally, the design of a complex decision aid system would be substantially more complex than this example. However, the essential methodological framework would still apply, no matter how complex the actual problem. Thus, in discussing the two decision aid approaches described in Chapters 3.0 and 4.0, it will be worthwhile to think in terms of Levels 3 and 4, respectively, and in evaluating those aids, it may be helpful to consider the costs and benefits in comparison to the other possible levels, on a variety of problems.

The "economic" framework described above will in actuality depend on noneconomic data, such as engineering analysis of feasibility, human-factors analysis of acceptance and operational characteristics, and technical analysis of system performance and impact. Without losing track of the overall system-design framework described in this section, the remainder of this report focuses more closely on the technical and operational aspects of the decision aids being discussed. In particular, the next section presents for discussion four issues where technical and operational factors interact in ways which help to determine the design and evaluation of particular decision aids.

```
ALL VARIABLES SET AT LEVEL 1
    BENEFIT       COST
       0            0

CHANGE  1: P.A. 1: EVACUATION
   FROM 1: NO DECISION AID
     TO 4: GENERIC MODELS

    BENEFIT       COST
     425            72

CHANGE  2: P.A. 2: HOSTAGES
   FROM 2: PENCIL-AND-PAPER
     TO 4: GENERIC MODELS

    BENEFIT       COST
     717           185

CHANGE  1: P.A. 1: EVACUATION
   FROM 4: GENERIC MODELS
     TO 5: HIGHER-ORDER GENERIC

    BENEFIT       COST
     867           380

CHANGE  3: P.A. 3: NUC. THEFT
   FROM 5: HIGHER-ORDER GENERIC
     TO 6: GENERAL-PURPOSE

    BENEFIT       COST
     970           800

CHANGE  2: P.A. 2: HOSTAGES
   FROM 5: HIGHER-ORDER GENERIC
     TO 6: GENERAL-PURPOSE AID

    BENEFIT       COST
    1000          1400

CHANGE  2: P.A. 2: HOSTAGES
   FROM 1: NO DECISION AID
     TO 2: PENCIL-AND-PAPER

    BENEFIT       COST
      87            12

CHANGE  3: P.A. 3: NUC. THEFT
   FROM 1: NO DECISION AID
     TO 4: GENERIC MODELS

    BENEFIT       COST
     525           107

CHANGE  3: P.A. 3: NUC. THEFT
   FROM 4: GENERIC MODELS
     TO 5: HIGHER-ORDER GENERIC

    BENEFIT       COST
     777           240

CHANGE  2: P.A. 2: HOSTAGES
   FROM 4: GENERIC MODELS
     TO 5: HIGHER-ORDER GENERIC

    BENEFIT       COST
     930           590

CHANGE  1: P.A. 1: EVACUATION
   FROM 5: HIGHER-ORDER GENERIC
     TO 6: GENERAL-PURPOSE AID

    BENEFIT       COST
     993          1100
```

Table 2-2

LIST OF EFFICIENT PACKAGES

2.3 Technical Issues in the Design of Problem Structuring Aids

The design or selection of a decision aid for any problem or set of problems is a complex issue, involving complex interactions among technical, operational, and economic issues. This section contains discussions of four major technical issues, each of which applies to the evaluation of any decision aid. A recurring theme is the dependence of "ideal" levels on the specific needs of the user, on the problem set being modeled, on the situational constraints that will operate, and on the user's own training, motivation, and technical sophistication. Thus, "good" and "bad" positions on these issues cannot be identified a priori, but depend heavily on context; similarly, a given type of decision aid should not be evaluated as "good" or "bad" but merely as more or less appropriate to a given problem context.

2.3.1 Generality versus context-specificity - As Von Winterfeldt (1979) points out, existing decision aids tend to represent two polar extremes: some are highly specific aids, prestructured and precanned, which apply to highly limited repetitive situations; and others are fundamentally empty structural aids which start with no predisposition towards (or special information about) any substantive problem. He goes on to observe that "Neither extreme is totally satisfactory. The middle ground of problem driven but still generalizable structures and models needs to be filled." This would be accomplished by "searching for generalizable features of problems that identify generic classes of decisions. These generic classes can then be modeled and structured by 'prototypical decision analytic structures'."

The location of a useful "middle ground" represents an important trade-off. As more generality and flexibility are added to an aid, its range of application and quality of representation rise, but so do the demands placed on the user, in terms of time, training, and analytic sophistication. While a completely specific aid can explicitly represent 100% of the substantive information it uses in a precanned form, a more generic aid will need to elicit some or all of its substantive knowledge from the user/expert. For this elicitation to take place without undue risk of modeling errors, the user must know more about how the aid operates, more about the substantive area itself, and more about how to respond correctly to the interactive routine's questions; and of course, the entire process will take more time.

A further risk of overly general structuring aids is the dilemma encountered when the designer must choose between an explosion of detailed information to be stored and retrieved, and an attenuation of detail, which might result in vagueness and confusion. To be fair, it must be observed that one very general model may still be preferable to a host of specific models, where the problem of selecting the right one could be even more confusing and time-consuming.

The absolute limit of generality in a decision aid would be reached at that level where any increase in applicability would entail a degradation in overall performance (cost, training, operating time, complexity, etc.) great enough to outweigh the advantage gained on the new problems covered. Naturally, in a well-designed system, a lower level of generality might be dictated by resource considerations of the type described in Section 2.2.

2.3.2 Extensive completeness (breadth of analysis) - Apart from out-and-out mistakes in elicitation or calculation, the most likely source of a major error in a decision analysis is to overlook some aspect of the problem while constructing the model. To ignore an option which, in retrospect, would have received the highest rating; to leave out a critically important event or scenario; to omit a major component of outcome utility--these errors could nullify the advantages gained from a systematic, logical approach to the problem.

One advantage of decision analysis is that, by presenting the factors in a logically organized manner, a decision-analytic model exposes potential gaps and therefore helps the decision maker to identify the need for more extensive modeling. However (particularly in the case of a computerized decision aid), merely presenting a model for the user to inspect, and then asking for any additional factors he may have overlooked is a rather passive approach to achieving extensive completeness.

The use of prestored substantive information (whether specific or generic) represents a more active approach, on the part of a decision aid, to the task of ensuring extensive completeness. To the extent that the prestored model does span the entire range of options, events, and utility attributes, that model will guarantee that the user has at least been reminded of the kinds of factors to consider. Ideally, a decision aid should be able to start with an exhaustive categorization of all options, events, and attributes (even if they are highly general, without much detail); the user could then identify which of the possible factors to keep in the specific model, and provide specificity as needed. In such a case, the user would be prevented from making errors of omission, and although it might in fact constitute an error to eliminate a factor deliberately, the likelihood of a serious error of this sort from a knowledgeable decision maker should be very low.

Of course, except for the simpler models, it will in general be possible or at least highly uneconomical to store an exhaustive list of factors relating to a given problem. A possible solution might be to provide as complete a structure as the situation will permit, resulting in a partial list of options, events, or attributes; after being presented with this partial list, the user can be asked to complete it with similar items; alternatively, a tentative list may be supplied and, after the preliminary analysis, augmented through a "secondary analysis" procedure.

This partial-list procedure has both advantages and disadvantages when compared with the alternative of providing no pre-stored structure at all. The advantages stem from the guarantee that at least the most important factors will tend to appear in the partial list, and from the facilitating effect of the examples provided for the user. The disadvantages stem from the possible tendency to accept the partial list as if it were complete, from any unintended bias in the makeup of the list, and from the possibility that in trying to complete a list which does not coincide with his natural way of thinking, a user might unintentionally introduce double-counting, ambiguity, and inconsistency into an otherwise sound model. (Of course this latter effect could be a problem even if no partial list were provided, but it is possible that the wrong partial list could exacerbate the problem rather than ease it.)

The issue of extensive completeness, then, represents a trade-off whereby additional generality, if it forces us to abandon the idea of an exhaustive pre-stored model, entails a great cost in terms of potential error. Sometimes, this risk is worthwhile, either due to the advantages which stem from the added generalization or because no riskless alternative is feasible. For any particular problem, it is therefore worthwhile to consider the impact of those potential risks, and to evaluate their associated benefits.

2.3.3 Intensive completeness (depth of analysis) - A somewhat different issue from the previous one is the determination of how much detail a model should strive for, and how much of that detail should be pre-canned with the model. As more detail is pre-canned, the models become more concrete, easier to understand, and more convincing; at the same time, they become less flexible, less manageable (because of their size), and less general in their range of applications.

While a failure of external completeness may lead to serious errors of oversight in modeling, insufficient depth of analysis is likely to lead to less critical errors which result from vagueness, from thinking in abstractions, or from trying to incorporate too many aspects of a problem into a single factor. Apart from these, a more serious risk involves the tendency to focus on prototypical instances of a factor, underrepresenting those subfactors which might be atypical but happen to be important in a particular situation. For example, to paraphrase an example of Amos Tversky's, in assessing the probability of an event category "George is a teacher," it is natural to overlook atypical but perfectly acceptable instances such as "George is a gym teacher" unless they are explicitly mentioned; thus, in the event where an atypical situation does exist, insufficient detail will obscure its impact on the model and therefore result in possible error.

There is no practical bound to the level of detail a model could reach; given a very generic model, a user could continue specifying further and further sublevels of decomposition ad infinitum. However, beyond a certain point the additional detail will cost more in terms of operating time and inconvenience (or, in the extreme, in terms of the computer's memory limitation) than it would provide in terms of simpler, more complete, or better-defined assessments.

We can think of each component of a model as a hierarchy: options can be represented as initial acts at the highest level of aggregation, and then subdivided according to possible subsequent acts; events may be described as general scenarios, each of which may in turn be divided into more specific instances, or subsequent events; and attributes may be divided into subfactors as desired. A pre-canned model will specify some or all of these elements down to a certain level of detail. Some aids will then permit further "secondary decomposition" of those elements for which more detail is desired. The extent of this "secondary decomposition" might be limited to a single level, restricted to some small number of levels, or left entirely to the user's discretion.

Here, as with the extensive completeness issue, the "proper" solution depends very much on the characteristics of the problem set to be modeled, the operational context in which the aid must be used, and the training, motivation, and sophistication of the decision makers who will use it. A practical approach would therefore begin with a study of these contextual variables, which could then be used as parameters to "fine-tune" an aid to fit the user's needs.

2.3.4 <u>"One-shot" versus iterative modeling strategies</u> - Practicing decision analysts have often observed a number of important benefits to accrue from the practice of building a "quick-and-dirty" model as early as possible. In addition to familiarizing the user/client with the methods and the overall approach in a direct and compelling way, this practice may serve to identify some factors as irrelevant and others as worthy of additional attention, to suggest new options which might have been overlooked, and to point out possible inconsistencies or ambiguities in the definitions being used. Naturally, much of the value of such a process depends on the decision analyst's technical knowledge, intuition, and experience; however, in the case of a computer-based decision aid, computational facility and extended memory may provide different, but very useful, forms of guidance based on a preliminary analysis.

Of course, if decision time is a critical constraint, there may be no choice available but to create the most appropriate model possible on a "one-shot" basis. But if time permits the luxury of two or more iterations, and completeness is of paramount importance, it may be of great value to select certain aspects of the problem as particularly worthy of further attention, and to focus more closely on those aspects in subsequent iterations. While this iterative approach necessarily involves a certain amount of duplication, the time saved by avoiding unnecessary detail may well compensate for any such "inefficiency;" and in any event, as new information is elicited, the user's interpretation and evaluation of existing factors may well change accordingly, yielding better values.

After the second iteration, it would seem that further iterations should become less and less efficient from a benefit-cost point of view. Nonetheless, until a point is reached where no further changes are desired, some benefit may result from further iterations, and it might be the case that the user's discretion is the best criterion for when to stop.

Of particular interest with respect to this issue is the problem of timing. Sometimes, reaction time is so constrained that flexibility and iteration are prohibitively slow; sometimes, the user can proceed at a leisurely pace until he is satisfied with his solution. In many cases, however, the time constraint may be highly variable, and the problem is to achieve the best model possible within a time constraint that cannot be known when the decision aid is designed.

When decision time can be reliably predicted, a "one-shot" approach can be designed to fit the known constraint, tailoring the level of detail and flexibility to the time limit. On the other hand, when time constraints are important but unpredictable, only an iterative approach can allow for a rapid decision when needed, and still allow as much further refinement as time and the user's needs permit. As we have observed on other issues, the "ideal" approach must be determined with the user's problem set, operational context, and technical knowledge in mind.

2.4 A Brief Survey of Problem Structuring Decision Aids

In this section, we shall provide general descriptions of nine different approaches to decision aiding and attempt to relate them to the technical and "economic" frameworks developed in the preceding sections. Of the aids described, the first two (SURVAV and OPINT) represent existing software developed by DDI and already in use on applied problems. The next three represent efforts by other researchers, each available at least as experimental software. The remaining four approaches are newly developed by DDI during the present contract effort; three of these aids (QVAL, RSCREEN, and OPGEN/OPSEL) now exist in experimental software versions, while the fourth (GenTree) has been developed only at the conceptual level. The main focus of this report will be RSCREEN and OPGEN/OPSEL; QVAL and GenTree are the subjects of another detailed report (Weiss, 1980).

2.4.1 SURVAV - SURVAV (SURVeillance AVoidance) is a DDI-developed aid dedicated to solving a single problem: determining the optimal speed of advance for a ship following a specified course, in order to avoid detection by

Soviet satellites (Barclay and Randall, 1977). The underlying model is completely pre-canned, and the only inputs from the user involve the specific course of advance and the relevant data parameters (satellite trajectories, weather conditions, etc.). In terms of the six-level context of Section 2.2, SURVAV typifies a single Level 3 aid which, by itself or together with a small number of similarly detailed aids, would constitute a useful but highly stereotyped system.

In terms of the four issues of Section 2.3, SURVAV is extremely context-specific; it entails a high degree of pre-canned extensive completeness, with no opportunity for further expansion of the model; it has a highly detailed model with little or no opportunity for further intensive analysis; and because structuring has been performed beforehand, there is no opportunity for iterative model structuring. In short, SURVAV is designed to do only *one thing, and to do it well.*

2.4.2 OPINT - In an attempt to achieve greater flexibility for a wide variety of problems, DDI constructed an aid called OPINT (OPtion screening and INTelligence assessment), implemented on an IBM 5100 portable computer (Allen et al., 1976). The OPINT aid is an example of a general-purpose structuring program which starts with no substantive information whatsoever, eliciting the structure of the model during a "build" phase and then eliciting values and performing calculations during a "run" phase. Thus, at the actual time of decision making, the user may either recall model structures that he has built previously (from a small library of such structural models), or else build a new structure on the spot. The only degree to which the OPINT software constrains the user in model construction is in limiting the format of the model to a choice among immediate acts, with outcomes which depend on one or more events, and evaluation based on a simple single- or multi-attributed utility function.

Because of the "build-run" distinction, OPINT can function at either Level 3 (specific models) or Level 6 (general-purpose aid) on the scale developed in Section 2.2. The model building itself, however, is an instance of Level 6, as OPINT embodies no substantive information or predispositions. In that sense, it is highly general in application; detail is extremely limited in the specification of options and utility attributes, but considerable detail is allowed in specifying sets of events which might directly or indirectly influence the evaluation of outcomes. Extensive completeness is totally under the control of the user, since OPINT supplies no pre-stored model elements to guarantee completeness; and although there is nothing to prevent a user from building and re-building structures, OPINT provides no particular help in this effort, and may thus be classified as a "one-shot" structuring aid. In summary, OPINT provides a maximum of flexibility and range of applications, in exchange for which the user must sacrifice the guidance and convenience of a pre-stored structure. To the degree that the ability to handle a wide variety of problems is more important than the time, effort, and risk (of modeling error) involved in ad hoc modeling, an OPINT-like aid may suit a given user's needs far better than one or more aids of the SURVAV variety.

2.4.3 MAUD - MAUD (Multi-Attribute Utility Decomposition) is described by Humphreys and Wisudha (1979) as "an interactive computer program for the structuring, decomposition and recomposition of preferences between multiattributed alternatives." Within the formal limitation of a multi-attribute utility approach, MAUD will evaluate a specific set of options by eliciting the names of attributes which distinguish among them, developing a "grid" of ratings on each of the attributes, for each of the options. As opposed to OPINT, MAUD is an example of a general-purpose decision aid which does assist the user in iteratively refining the model. The structure used assumes a fairly small number of options and attributes, in nonhierarchical lists, and does not permit the explicit modeling of events and their probabilities; however, to the degree that any problem can be adequately modeled within these constraints, MAUD is general in application.

Although it is possible for the MAUD user to save and recall models built previously, the major use involves a Level 6 type of operation, in which the program can apply to any arbitrary substantive problem. While less general than OPINT (because it does not allow the explicit modeling of uncertain events), MAUD nonetheless applies to a substantial range of problems. As with most "general-purpose" decision aids, MAUD leaves the problem of extensive completeness in the hands of the user and provides no guarantee of completeness. Intensive "depth of analysis" is not very high, as only a small number of attributes can be manageably and conveniently incorporated (although it is possible to build subsidiary models to provide further detail, if the effort is justified). A valuable feature of MAUD is the ability to iterate, adding new options and attributes based upon the preliminary analyses, while possibly identifying and eliminating options or attributes which seem to be irrelevant, or which duplicate one another. In short, MAUD seems best suited for a user who is likely to be faced with a broad range of substantive problems in which unce inty is not critical, and for which reasonably small, simple evaluation models are likely to suffice. One further application would be as a device for descriptive studies of the user's problem-structuring behavior.

2.4.4 Leal-Pearl decision aid - Leal and Pearl (1976) have described a "computer system for conversational elicitation of problem structures" which, like MAUD, operates on a methodologically restricted category of problems, but also like MAUD, imposes no substantive constraints on the problems it will tackle. Whereas MAUD restricts its models to those in which events are not modeled explicitly, the Leal-Pearl program permits the explicit modeling of events, but assumes a scalar (single-attribute) utility function instead of a multi-attributed one. In this sense, the Leal-Pearl aid and MAUD may be considered complementary, one supplying the probabilistic modeling of events and the other supplying a technique for multi-attribute utility decomposition. The approach of the Leal-Pearl aid is very different from that of MAUD, however, as it involves a heuristic strategy whereby iteration is guided by the computation of a "sensitivity differential" which selects the portion of a model in which the smallest change could affect the option selected.

Because its procedure is independent of the substantive problem domain, the Leal-Pearl decision aid falls into the Level 6 category, according to the scale developed in Section 2.2. It generally is less restricted than MAUD's, because theoretically the multiple attributes could be combined into a unitary utility measure on the spot much more readily than a MAUD user could summarize complex uncertainty patterns. Furthermore, the Leal-Pearl aid permits the user to specify subsequent acts directly, without converting them into conditional aspects of the initial acts, and therefore allows users to capture a wider variety of practical problem situations than the simple specification of a few initial acts would allow. As with all other completely general-purpose aids, this one leaves the problem of extensive completeness at the mercy of the user, although it does provide general prompts to remind the user to be as comprehensive as is practical. Intensive completeness is potentially quite high in terms of the ability to specify complex and detailed sequences of actions and events, while in the area of utility assessment, there is essentially no opportunity for intensive expansion. Finally, the Leal-Pearl aid depends critically on its ability to develop an act-event structure iteratively, allowing the user to specify at various points during the analysis that he is satisfied with the current level of detail and wishes to terminate the session; at that point a summary of the most highly rated action and all indicated conditional subsequent acts will appear. In summary, the Leal-Pearl aid can apply to decisions in any substantive area, although in format it is best suited to problems whose complexity stems from uncertainty in the form of act-event dependencies, and where outcomes can be easily evaluated in terms of monetary expectation or other scalar utility values. It could also be used only when the time is available to build a model from scratch, because no pre-stored structures are available.

2.4.5 SRI decision aid - Merkhofer et al. (1979) have described an interactive aid which extends the overall methodology of the Leal-Pearl decision aid to a somewhat more powerful, but essentially similar, approach. The SRI aid also uses the decision tree formulation, but adds a number of features such as influence diagramming to facilitate probability assessments, a simple multi-attribute utility to accommodate multiple objectives, and a value-of-modeling analysis to guide iterative expansion more effectively than a simple "sensitivity differential" technique would. More important, the SRI aid provides a capability to construct partial decision tree models which incorporate the value structure and the topmost levels of analysis, so that when an instance of a recurring problem type occurs, analysis can begin with much of the model pre-stored, and then proceed to iterate upon the preliminary results as before; this gives the SRI aid a dual-level approach, as a new problem may be analyzed from the beginning, but some generic modeling of the Level 4 variety is available.

This dual-level approach to decision aiding provides the user with pre-canned specific or generic structures when needed in a crisis, or as a preliminary step in more detailed analyses, while retaining the generality which stems from a Level 6 system. The issue of extensive completeness is, as we have seen before, under the control of the user, except when pre-canned structures have been used to guarantee completeness. Because either newly created or pre-canned structures can be iteratively detailed as far as desired, intensive completeness is unlimited, except in the utility structure itself, which is still (in the current version) restricted to a nonhierarchical, additive combination of a few attributes or objectives; of course, off-line auxiliary aids or further development could add the hierarchical capability without much additional technical disruption, but the current version seems sufficient for use as a demonstration of the techniques involved. Finally, due to its inherently iterative nature, the SRI decision aid is well suited to those situations in which the user is likely to run out of time before all the relevant details can be incorporated; its approach allows the user to obtain as much detail as possible without committing him to an extended elicitation session.

In summary, the SRI decision aid can capture a wide variety of decision problems, either from scratch or from pre-stored models which can be further developed. The iterative techniques guide the user through the tree expansion process efficiently under a moderate but variable time constraint, while it is possible to pre-can certain frequently used models for especially quick reaction. Because of its decision-tree-oriented framework, the SRI decision aid is of value primarily in action-event problems for which objectives are well defined and few in number, and the major source of complexity is the interactive influence of the acts and events on one another.

2.4.6 QVAL - QVAL (Quick eVALuation) is a new decision aid developed at DDI under the present contract and described in detail in another report (Weiss, 1980). Briefly, QVAL is a completely general-purpose aid (although it does have the capability to store complete or partial models if desired), which helps the user construct and iteratively expand a multi-attribute utility model used to evaluate alternatives in a non-risky situation. The central principle behind QVAL's operation is the amalgamation of structuring (i.e., adding new attributes to the model) and elicitation (i.e., assessing the scores on, and the weight for, the new attribute elicited). At each iteration, a new attribute is added to the model, scores and weights elicited, and a modified overall rating calculated; then, identifying the option with the highest tentative overall value, QVAL invites the user to play the "devil's advocate" by suggesting a new attribute which might lead to a preference for one of the other attributes. Thus, the model is iteratively expanded until either time or the user's ability to generate possible objections to the indicated choice has run out. If more time is available, additional capabilities are provided for continued modification or expansion of the model, for creation of a hierarchical structure based on the attributes elicited, and for exploratory sensitivity analysis, which may in turn lead to ideas for new attributes.

QVAL, as presently implemented in prototype form on an IBM 5110 portable computer, is primarily designed to function at a completely general level (Level 6), where all evaluation decisions are equally treated. Current capabilities do permit storing a specific model, with the option to modify or expand at a later time, but those specific models are not themselves very flexible (they would correspond to Level 3 of the six levels defined in Section 2.2); thus, the primary use envisioned for QVAL is to provide help on ad hoc problems for which no pre-stored model exists. Because of its general approach, QVAL leaves the issue of extensive completeness in the hands of the user, although the "devil's advocate" technique probably helps to focus attention on the "right" kinds of new attributes to add. Intensive completeness is not a major goal in QVAL, where the primary emphasis is on making quick, time-effective decisions on problems which must be modeled from scratch; some facility for hierarchical structuring may be used as a basis for further decomposition, but in general QVAL's operation is non-hierarchical. In terms of iteration, QVAL may be used on a one-shot basis, but is primarily designed to provide an iterative approach to model expansion; iteration permits the user to stop the modeling process whenever he is satisfied with his decision, or when time runs out, with a current set of scores and weights at any point.

Although QVAL is still at an experimental prototype stage, it is well suited for a number of ad hoc problems which demand a quick choice among alternatives in which risk is not a predominant factor. Future developments might include the ability to store pre-canned generic models of a Level 4 variety, to perform structuring and elicitation hierarchically, and to incorporate uncertain events and act-event dependencies at some level of modeling. It is also conceivable that QVAL or a similar routine might be integrated with a decision-tree-oriented aid such as the Leal-Pearl or the SRI decision aid, to take advantage of their complementary strengths.

2.4.7 GenTree - Weiss (1980) has described at a conceptual level a radically different concept of general-purpose decision aiding, in the form of a plan for an aid called GenTree (GENeric TREE structuring). GenTree differs from the other aids described in this section in that it incorporates substantive information in a model-independent way, by maintaining a complex data base, and then uses the information in that data base to provide a much more active role in generating suggested options, events, and attributes. Thus, rather than storing a library of models, GenTree stores atomic components from which models may be constructed according to a set of rules analogous to a grammar. In particular, GenTree's data base includes two components: a data representation system which contains pre-stored substantive information including updates based upon its previous modeling experience; and a semantic context framework which provides a "map" of the relationships that might be used to link the needs of the current modeling effort with the information stored in the data representation system. One possible conceptual framework for GenTree's data base system might be an organization based upon the case grammar of Fillmore (1968), or upon the more elaborate but similar notion of scripts (Schank, 1972, 1973).

GenTree could operate on a problem about which it had no information or, more typically, as an interactive routine to construct from the "modular" components of the data base (as well as the user's substantive expertise) a model tailored to the specific problem at hand; thus it would be classified as Level 6 on the scale developed in Section 2.2. To the degree that a useful match can be found in the data base, the structuring problem will be greatly simplified and accelerated, and the demands on the user will be reduced. For example, if the problem involves the evaluation of vehicles to be used as a military ambulance, any prior information about similar motor vehicles or about emergency medical care systems could permit a "straw man" model to be built quickly and painlessly and then modified as necessary to incorporate the situation-specific data and judgments provided by the user.

Since GenTree will be designed to apply to any problem, its level of generality will be high, although there will be a distinct preference in its operation for "familiar" problems. To some extent, this system represents a far higher investment of resources (for building, maintaining, and manipulating the data base) than most of the other aids described in this report, and therefore achieves its high level of generality without many of the associated difficulties (vagueness, excessive time and effort required, etc.). Because it allows a considerable amount of pre-storage of information, GenTree provides a high degree of extensive completeness, at least for those problems which match its data base fairly well; as the familiarity of a problem decreases, GenTree's performance will approach that of any other general-purpose decision aid without pre-canned modeling information.

GenTree possesses a greater capacity for intensive completeness than most of the other aids currently conceived, because its structural framework (case grammars or scripts) will prompt the user for specific ways of subdividing items and will provide suggestions to help the user (for example, GenTree will "know" that a system's value may be partitioned in a number of useful ways--by subsystems, by user constituencies, by task, by functional phase, etc.); thus, the user will find it easier to generate an appropriate way of adding detail to a model. Finally, although GenTree will probably operate in a more or less "one-shot" mode, the fact that all prior modeling is incorporated into the data base (indexed by context) will permit interactive iteration to expand or modify the initially constructed model. As presently conceived, GenTree builds a structure first and then does its elicitations based on the structure; if an iterative procedure like that used by QVAL, the SRI aid, or the Leal-Pearl aid were instituted, that might be a valuable addition.

In summary, although GenTree has no physical embodiment at present, it represents a possibility for dramatically improved general-purpose decision aiding. With a full-scale effort including not only decision-analytic techniques, but also data base manipulation systems and artificial intelligence routines for searching and applying the information in the data base productively, a GenTree system might begin to provide the user with a level of sophistication far beyond the abilities of the more limited programs described here.

2.4.8 RSCREEN - The first of two decision aids designed by DDI under the present contract and described fully in this report, RSCREEN (Rapid SCREENing of options) constitutes a Level 4 approach to problem structuring. Briefly, RSCREEN contains a small library of partially detailed generic models designed to establish frameworks for a few frequently encountered problem types, at a moderate level of generality (e.g., the entire set of problems involving evacuation planning, rather than the specific problem of evacuating a certain number of citizens from a particular country in a particular political/military situation). RSCREEN begins by eliciting scores and weights for the pre-canned model (which is a multi-attribute evaluation) and then provides the user (if time is available) with assistance in expanding the model through secondary decomposition, secondary analysis, selection of critical elements for focusing, and so forth. An experimental version of RSCREEN has been implemented on an IBM 5110 portable computer, although further effort will be needed in user-engineering before it can meaningfully be field-tested on real-world problems.

Overall, RSCREEN sacrifices some generality in order to achieve sufficient task-specificity for it to operate in an entirely "pre-canned" (i.e., Level 2) mode, should the user's needs so dictate. Flexibility is added by the model expansion and sensitivity analysis capabilities. RSCREEN's models can be constructed at a specific enough level to ensure extensive completeness with respect to the recurring components of a decision problem set; secondary analysis permits the user to add any factors not present in the pre-stored model. In terms of intensive completeness (level of detail), the pre-stored models achieve a moderate level, while secondary decomposition would allow at least one level of further decomposition (and presumably capture most of the relevant detail on most problems). RSCREEN falls at an intermediate position on the "one-shot versus iterative" scale, being essentially a "two-shot" approach, in which a pre-canned model is evaluated, and then a single further iteration is performed (rather than an unlimited number of such iterations, as is the case with some of the other aids). RSCREEN is discussed in detail in Chapter 3.0.

2.4.9 OPGEN - Another decision aid produced by DDI under the current contract (and discussed in detail in Chapter 4.0 of this report) is OPGEN (OPtion GENeration), which has been implemented as an experimental prototype on an IBM 5110 portable computer. OPGEN represents an extension of the specific modeling approach used by OPINT (see Section 2.4.2), to make use of pre-stored generic models which include actions, event sequences, and evaluation criteria. An interactive routine assists the user in screening, specifying, and evaluating the options, eventually selecting a subset of the most promising ones for more detailed attention. The pre-canned "generic" OPGEN models are actually quite specific about the initially used set of events and attributes, but allow considerable leeway for subsequent expansion, specification, and modification;

the option categories are, on the other hand, highly generic, and demand explication in terms of the specific context of the immediate problem. Overall, then, OPGEN can be said to operate on Level 4 of the scale developed in Section 2.2.

OPGEN's level of generality is about the same as RSCREEN's, which is to say somewhat greater than the completely pre-canned approach, but certainly not as broad as a general-purpose aid such as QVAL. OPGEN's pre-stored generic structures can be fairly extensive, and although it is nearly impossible to guarantee extensive completeness when dealing with unanticipated decision problems, OPGEN should guarantee a substantial degree of coverage for those problems it was designed for. In terms of intensive completeness, OPGEN does not provide much detail about utility structures (it allows only a small number of nonhierarchically organized attributes), but it allows a considerable amount of detail in specifying event sequences (scenarios) and options. Finally, at least in the current version, OPGEN is primarily a "one-shot" approach to structuring; subsequent modification and expansion are possible, but that is not the anticipated mode of operation in general.

In general, OPGEN seems to apply best to problem areas in which the criteria for evaluation of an outcome, the kinds of actions to be taken, and the kinds of events which might influence outcome can be specified in advance, without reference to a specific context or geographical area, and where the specific interpretation of these generic actions, events, and attributes will be fairly evident to the user whenever any particular problem is being analyzed. Because OPGEN generic models can be constructed for a wide variety of contingency situations a given user is likely to encounter, the aid can be more or less tailored to that user's anticipated needs.

3.0 RSCREEN: A DECISION AID WHICH USES PRE-CANNED DECISION PROBLEM TEMPLATES

In this chapter, the theory and procedures of a decision aid approach called RSCREEN (Rapid SCREENing of decision options) are discussed. This discussion concentrates on the RSCREEN approach, rather than on the specifics of the experimental software implementation DDI has programmed on the IBM 5110 portable computer. Thus, it serves more as a functional and conceptual description than as a user's guide (an earlier version of RSCREEN, developed under joint sponsorship of the current contract and the Defense Communications Agency, is documented with a user's guide--see Gulick and Allardyce, 1979).

3.1 Background

During crisis situations, military decision makers and their staffs strive to react swiftly, decide wisely, and communicate accurately. However, by its very definition, a crisis situation imposes significant obstacles to the successful attainment of those three worthwhile objectives.

Some of the obstacles occur because in a crisis situation decision makers and their staffs must necessarily abandon their routine day-to-day working relationships, information channels, and standard, familiar procedures. Other obstacles arise from the increased tension and anxiety introduced by the enormity of the stakes at hand and the attendant uncertainties, risks, and intricate value trade-offs. Still other obstacles stem from the inherent pressures of time constraints and the ambiguity of goals and value structures.

In addition, crisis decision making is usually attended by extraordinary demands for and the production of information. The tasks of information collection, processing, and distribution may well dominate the workflow and unduly monopolize the time and attention of the decision maker. Indeed, crisis decision makers are often inundated with a vast and diverse collection of information, both objective and subjective. Both kinds of information may be of highly varying quality and relevance.

The high premium usually placed on information collection and processing, coupled with the significant obstacles imposed by the crisis situation, greatly enhance the always-present opportunities for misperception, misunderstanding, and miscommunication among decision makers and their staffs. To prevent those opportunities from arising, decision makers need effective decision strategies that impose rigor and provide a logical, structural framework to assist them in the process of choosing an optimal decision alternative in the face of voluminous and often inconclusive evidence, staff reports, expert opinion, and personal judgment.

RSCREEN is a decision strategy that provides just such a framework for deliberation, reasoning, and analysis. RSCREEN aids decision makers by prescribing a straightforward normative procedure for organizing and analyzing difficult evaluation problems requiring the complex value trade-offs pertaining to the ultimate choice of a course of action.

RSCREEN is an interactive computer software program that permits the rapid evaluation of several different courses of action. The program implements standard decision-analytic procedures except that the more commonly used decision tree structures are replaced by a simplified prestructured format called a Decision Template. The templating procedure is a quick, accurate, and useful way of evaluating courses of action. It is used to expedite the decision analysis when the time available for it is very short and to identify the critical features of the problem for more detailed analysis and hypothesis testing.

Each decision template encompasses the critical elements of the political-military situation in much the same manner as a well-developed contingency plan. In fact, the RSCREEN template should be thought of as a sophisticated contingency plan that had been developed ahead of time and that captures the lessons learned from previously accumulated crisis management experience.

Three fundamental steps are involved in the RSCREEN evaluation of decision options. First, the courses of action to be considered are selected; second, the relative value of each course of action is determined through a hierarchical decomposition assessment procedure; finally, the sensitivity of the results of the analysis to changes in the input values is examined.

Removing some of the complexity and detail of an actual problem description to fit the requirements of the basic RSCREEN template entails some attendant loss of information. However, this streamlining process has the advantage of making the most important elements of the problem and their relationships to one another more easily understood and communicated to the decision maker. In each application of RSCREEN, the user must judge whether the simplification which allows for more rapid analysis is so restrictive as to detract seriously from the final results. The primary advantage is that the process of using the RSCREEN approach normally leads to a better understanding and more concise statement of the problem and a more rigorous and enlightening evaluation. Accordingly, the process of using the RSCREEN template approach promotes better understanding of the problem and leads to a more concise problem definition. That, in turn, leads to a more coherent and insightful evaluation of the courses of action.

The fundamental product of RSCREEN is a computer-stored evaluation model of the decision problem at hand. Whereas decision analysis provides the theoretical background and procedural guidance, the RSCREEN evaluation model provides the specific methodological tool for processing relevant information and evaluating the various courses of action open to the decision maker.

3.2 Technical Foundations

RSCREEN represents an adaptation of multi-attribute utility evaluation to assist decision makers in a crisis situation with severe constraints on time, detailed information processing, and training. The following conditions represent a prototypical characterization of those problems for which RSCREEN was designed:

- a particular class of decision problems can be expected to recur often enough to justify the effort of prior analysis and modeling;

- although the overall need for a model can be anticipated, the actual onset of any problem situation requiring its use may be a surprise;

- once such a problem arises, rapid response will be an important requirement;

- major emphasis will be placed upon an immediate action, rather than on the collection and analysis of data over an extended time.

Most practicing decision analysts agree (as Section 2.1 indicates) that the most time-consuming and error-prone phase of decision analysis is model construction, even when time is plentiful and a professional decision analyst is available to assist the decision maker. Under severe time constraints, experiencing some psychological stress due to the immediacy and importance of a crisis decision task, and deprived of the technical knowledge and experience of the professional decision analyst, the decision maker would probably balk at the idea of structuring an entire model from scratch, even if that model might in fact be useful. In order to develop an aid which is most useful to the decision maker, and one which will be convenient and stress-reducing, DDI has developed the idea of a decision template. This template represents a pre-canned model structure intended to approximate as closely as possible those essential decision factors which characterize a particular class of decision problems. While it lacks specific reference to the details of any individual instance, a decision template does provide a framework in which the decision maker and his staff can consistently and conveniently discuss the impact of the various options, and evaluate the overall desirability of each.

By pre-canning a structure, RSCREEN guarantees that the decision makers will at least be reminded of those considerations which might impact the decision. To be sure, the decision makers may find some of the criteria irrelevant in a particular situation and may wish to add new criteria or to add detail to existing ones, but these modifications will in general be far easier and faster than building a new model from the start. Naturally, if additional time is available, or if a subsequent decision is to be made based on additional information, the ability to iterate and refine the model will prove useful; RSCREEN's secondary analysis and secondary decomposition features permit such continuations.

Many decision-analytic approaches include as fundamental components the modeling of uncertain events (enemy actions, weather conditions, political situations, etc.) by simple or hierarchical probability models. RSCREEN deliberately avoids such modeling for a number of reasons: first, the elicitation of accurate probability judgments is a difficult and time-consuming task even with a professional analyst present, and would be extremely difficult, slow, and risky under the suboptimal conditions anticipated. In addition, sophisticated modeling approaches such as Bayesian Hierarchical Inference usually require advance knowledge of the events and indicators, knowledge which is in general too situation-specific to be included in a pre-canned model. Finally, the complexity of a probabilistic approach may confuse the untrained decision makers and obscure the "big picture" which relates the output of the decision aid to the immediate problem situation. Instead of detailed modeling of the uncertainties involved in a situation, RSCREEN relies on the knowledge and experience of the decision maker and his staff to relate the impact of such possible events to the evaluation criteria in the multi-attribute utility model.

In summary, RSCREEN uses pre-canned multi-attribute utility models to provide decision makers with a conceptual framework for rapid problem solving. The stored model need not capture every detail of the situation, and as long as the essential attributes of the problem are covered, a simple model is preferable to a more complicated one. Once the preliminary analysis is completed, additional time may be productively used to perform sensitivity analysis, to add detail to the model, or to select and focus on a few of the most attractive-looking options.

3.3 A Procedural Analysis of RSCREEN

In this section, the operation of RSCREEN is described in detail. First, we shall describe the standard sequence of operations involved in applying one of the prestored model templates to a specific situation, this creating a special-purpose model with the relevant numerical assessments. In subsequent sections, various special features are described which involve options available once the original model has been constructed. Finally, a structural overview summarizes the current RSCREEN system.

3.3.1 Initial model construction - The initial phase of RSCREEN operation involves the following steps: (1) selection of the appropriate decision template; (2) entry of the specific options under consideration; (3) elicitation of scores for the data-level (i.e., most specific) attributes; (4) elicitation of weights for data-level attributes; (5) "bottom-up" assessment of weights for higher-level attribute categories; and (6) calculation and display of results.

3.3.1.1 Template selection - A small library of template models is presented, from which the user specifies which type of model is appropriate to the immediate problem (alternately, each model may reside on a separate magnetic diskette or tape cassette.) In most situations, a user will either be familiar with the possible model templates, or have access to some documentation (printed or, preferably, resident in the software) which describes the model templates and the problems they apply to. Ideally, though, RSCREEN (and any other decision aiding system) could be designed to operate at several levels: a highly expansive, tutorial mode which explains each step; a "normal" mode, which proceeds more rapidly and stops to provide information or instructions only when requested; and a "streamlined" mode, which allows expert or frequent users to take shortcuts that might have been confusing to the ordinary user. In particular, as the number of model templates increases beyond five or six, it might become advisable to provide assistance in selecting the proper template; however, such template selection issues are peripheral to the RSCREEN approach, so we shall assume the number of models to be small enough for unaided choice.

Currently, there are model templates which apply to three problem areas: (1) projection of forces for political purposes; (2) evacuation of U.S. citizens from foreign territory; and (3) military risk evaluation. Figures 3-1, 3-2, and 3-3 present diagrams of the utility structures of the three model templates. While it is not the purpose of this report to discuss the substantive content of those structures, the figures should provide an idea of the level of generality associated with RSCREEN models.

In practice, an RSCREEN aid as delivered to a user would include the RSCREEN program itself, plus a set of model templates designed by a team of decision analysts and substantive experts to represent the most frequent application areas. Since these templates can be constructed and stored in advance of any pressing need, they can be tested and evaluated, to provide the best combination of substantive correctness, user orientation, and analytic simplicity. Thus, these templates should help to minimize the stress and time pressure that arise in an actual crisis and should constantly improve the speed and quality of decision making.

3.3.1.2 Option identification - Initially, RSCREEN simply asks the user to list his possible courses of action, with a brief definition for each one. It would be highly desirable to provide the user with more guidance in selecting

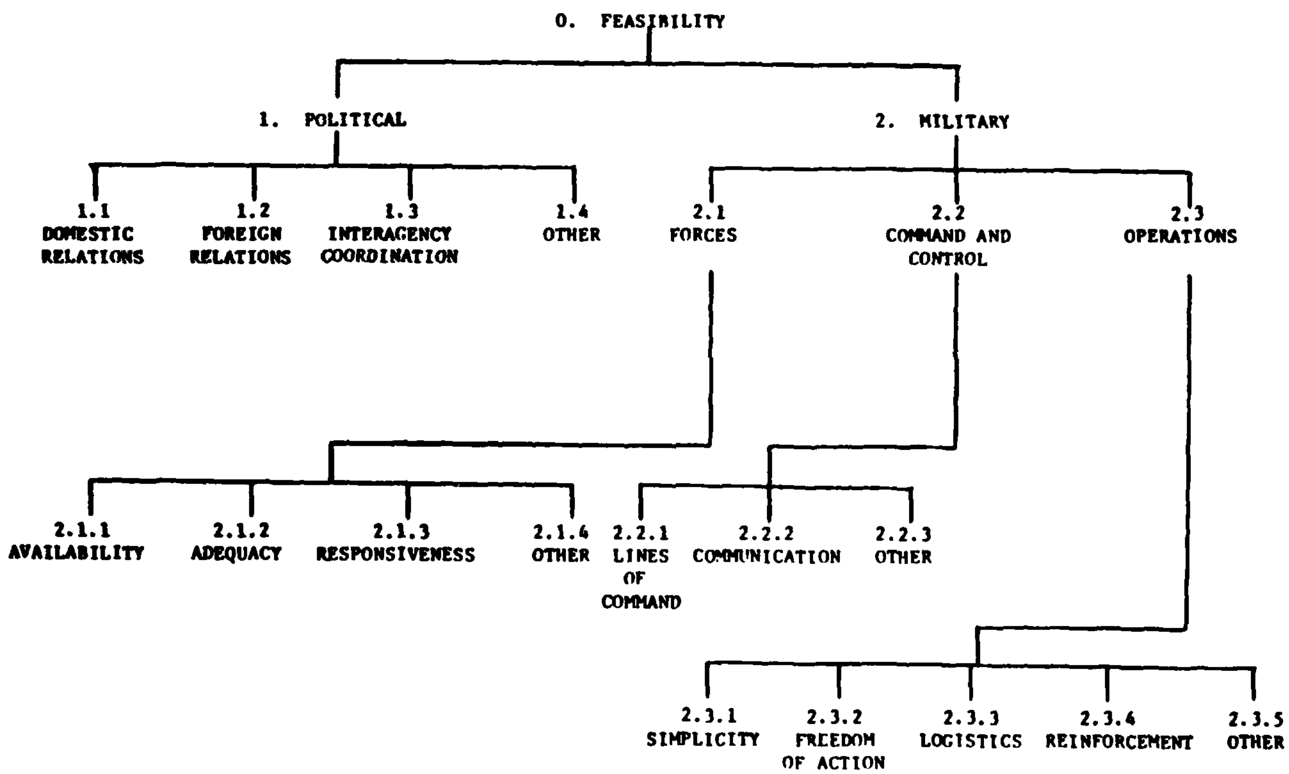


Figure 3-1

PROJECTION OF FORCES FOR POLITICAL PURPOSES

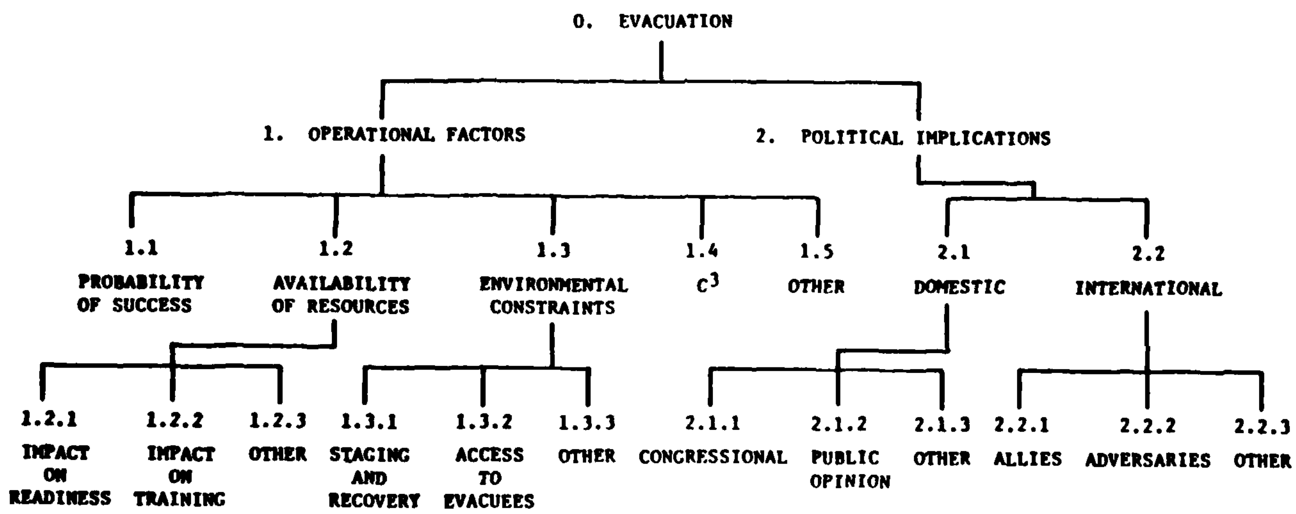


Figure 3-2
EVACUATION TEMPLATE

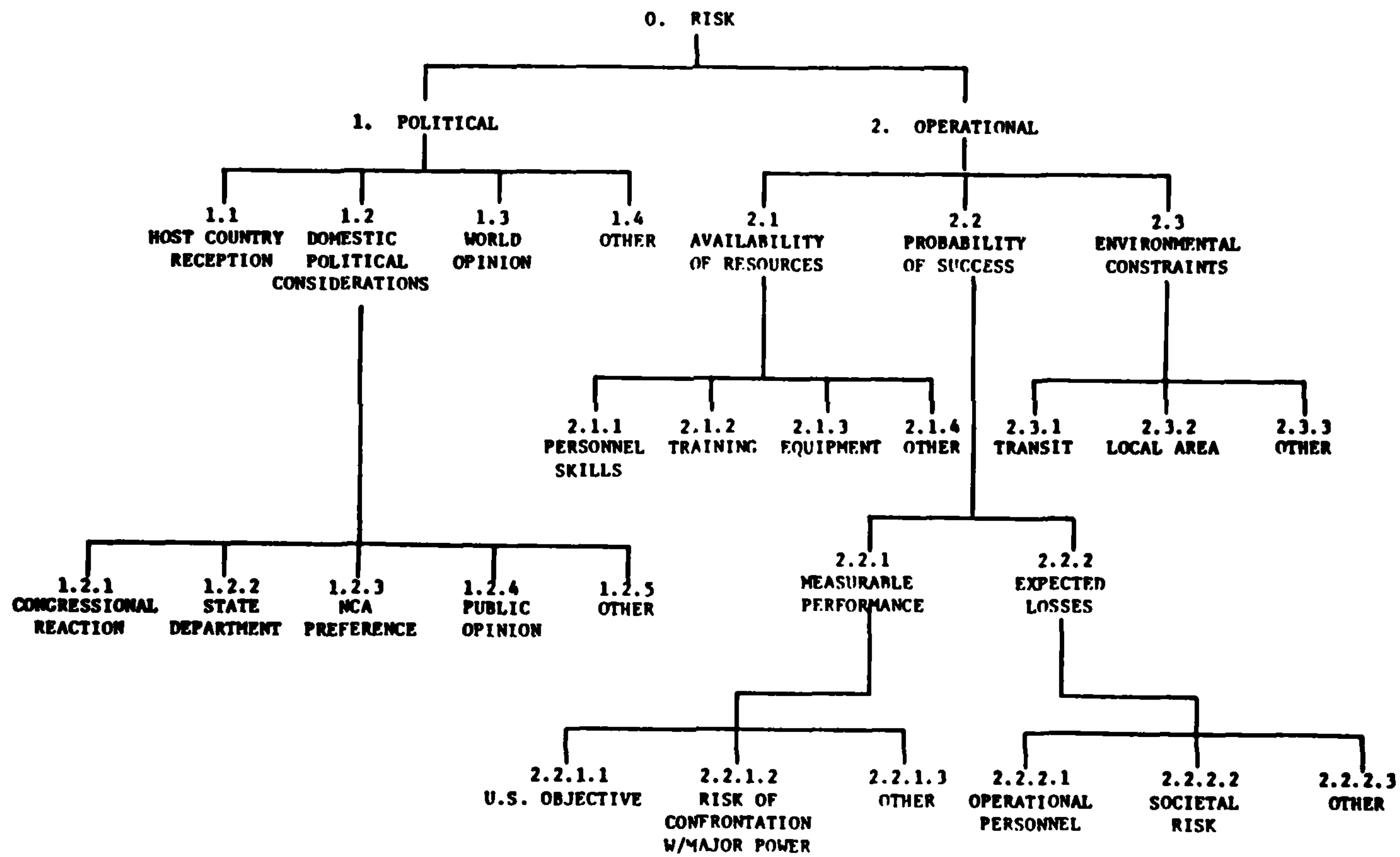


Figure 3-3
RISK TEMPLATE

an appropriate set of options to evaluate; but experience has shown that in most problems, the options are so situation-dependent that efforts at pre-storing options may create more trouble than any possible benefits would justify. (However, for certain problems where it is possible to specify a set of generic option types, an approach such as OPGEN, as detailed in Chapter 4.0, might be adapted to the RSCREEN format.)

Apart from providing pre-canned options or generic option types, about the only guidance which RSCREEN might be able to provide (but does not currently provide) would be general-purpose questions based upon the attributes in the template model. For example, RSCREEN might, in the context of the template in Figure 3-1, ask the user a question like, "Suppose the only criterion of any importance to you was Item 1.3, 'Interagency Coordination'; in that event, what option would appear the most attractive to you?" A series of such questions might serve to probe the user's thoughts to elicit options which might not occur to him spontaneously. (A similar procedure is incorporated in the OPGEN aid and may be adapted for RSCREEN if future investigation so indicates.)

3.3.1.3 Entry of values - Once the user has specified the appropriate decision template and the names of the options being considered, RSCREEN has the framework into which the numerical assessments will fit. The next step in initializing the model is the systematic elicitation of scores which rate each option on each of the bottom-level utility components.

The method for eliciting scores entails two phases: First, the user is asked to specify the best and the worst options for a given attribute: having specified these, the user then indicates the ratings for the intermediate options, relative to those endpoints. In order to simplify the elicitation and to take advantage of the user's superior response to graphically organized material, RSCREEN performs this phase of the elicitation graphically, as shown in the example below.

Suppose there are four options being considered and that Option 2 has been designated the best and Option 4 the worst, for a given attribute. RSCREEN presents a scale on which the number 2 appears at the right-hand end of the scale (indicating a score of 100 on a relative scoring system), while the number 4 appears at the left (indicating a relative score of 0), as shown:

```
SCORE     0        20        40        60        80        100
          +----+----+----+----+----+----+----+----+----+----+
OPTION    4                                                 2
```

The user is then instructed to fill in the numbers '1' and '3' in the appropriate positions beneath the scale; on the IBM 5110, this is done by moving a cursor across the line and typing the character in the space desired. Supposing, for example, that Option 1 had a value of 54 and Option 3 had a value of 80 on the scale, the display would look like this:

```
SCORE     0        20        40        60        80        100
          +----+----+----+----+----+----+----+----+----+----+
OPTION    4                          1            3         2
```

Because the spatial presentation of relative distances between the options provides an intuitive feel for the scores, this elicitation procedure should reduce some of the response biases that originate when numbers are assigned directly. Of course, in order to allow the user to check for accuracy, to specify scores with greater precision than the two-percent intervals of the scale, or to indicate that two or more options should receive identical scores, RSCREEN then presents the indicated scores numerically and allows the user to make any necessary adjustment.

3.3.1.4 Elicitation of data-level weights - The next step in filling out the model asks the user to specify the relative weights of data-level attributes which appear on the same branch of the tree structure. For example, if the attribute corresponding to node 2.2 is subdivided into three factors--2.2.1, 2.2.2, and 2.2.3--and each of the subfactors is a data-level node for which scores have been elicited directly, then this step will obtain relative weights for the three subfactors.

The technical definition of the weights being elicited requires the user to make a moderately complex set of judgments: the ratio between the weight on attribute A and the weight on attribute B is defined as the relative magnitude or importance of the differences (or "swing") between the best and the worst results on attribute B. In other words, the user must perform the following tasks:

- visualize the best and the worst possible outcomes with respect to attribute A;

- determine the value of the best-to-worst swing between these two extremes;

- visualize the best and worst possible outcomes with respect to attribute B (which will not, in general, be associated with the same pair of options considered with respect to attribute A);

- determine the best-to-worst swing between the extremes of attribute B; and then

- compare the swing on attribute A with that on attribute B, expressing the result as a numerical ratio. When more than two attributes are being compared, the complexity of the task increases correspondingly.

The RSCREEN procedure for obtaining these weights simplifies the user's task somewhat by presenting the names of the subfactors, together with the best and worst options associated with each, in an array for the user's reference. The actual determination of weights is approached in two phases: first, the user rank-orders the swings on the various factors; then, having specified the order of the factors, the user assigns weights with reference to a nominal weight of 100 for the most important swing, using a graphic elicitation similar to the score elicitation.

In the following example, the attribute "environmental constraints" has been subdivided into three subfactors: "transmit," "local area," and "other." The options A1, A2, A3, and A4, have been scored with respect to the three subfactors, as follows:

OPTION	TRANSIT	LOCAL AREA	OTHER
A1	100	32	50
A2	0	100	100
A3	24	0	20
A4	86	44	0

RSCREEN then presents the user with the following array:

FACTOR	1 TRANSIT	2 LOCAL AREA	3 OTHER
BEST	A1	A2	A2
WORST	A2	A3	A4
RANK	—	—	—

In the three blanks, the user fills in the numbers 1, 2, and 3, in the appropriate order so that the most important swing corresponds to rank 1, the next most important to rank 2, and the least important to rank 3. Suppose the order indicated is 1-3-2 (i.e., "transit" is most important.) Now, RSCREEN assigns the most important subfactor ("transmit" in this instance) a nominal weight of 100, and displays a scale upon which the user will indicated the relative positions of the other two factors, as shown:

```
WEIGHT    0        20        40        60        80        100
          +----+----+----+----+----+----+----+----+----+----+
FACTOR                                                      1
```

The user must now supply (again, by typing characters in the appropriate positions beneath the scale) the relative swing weights for factors 2 and 3. For example, if "local area" resulted in a swing between A2 and A3 that was half as significant as that between A1 and A2 on "transit," the assigned weight would be 50; if "other" had a swing somewhat greater than that, but not nearly so great as "transit," its weight might be something like 62. On the scale, then, the user would enter the following values:

```
WEIGHT    0        20        40        60        80        100
          +----+----+----+----+----+----+----+----+----+----+
FACTOR                             2     3                  1
```

The elicited weights (100, 50, and 62 for "transit," "local area," and "other," respectively) are then normalized--reduced proportionately so that the re-scaled weights would be 47, 24, and 29 (after rounding off); these weights would be presented to the user for verification and possible adjustment, as indicated.

3.3.1.5 Elicitation of higher-order weights - When a higher-order factor is composed of subfactors, some of which are themselves subdivided, the weight elicitation procedure becomes still more complicated, particularly because the number of factors to be considered simultaneously is so great. It is well known in practice that depending on exactly what questions are asked, and in what order, it is possible to influence the weights elicited in a biased way, because of the difficulty associated with the comparisons of factors on different branches of the tree structure. Particularly, if the items presented for comparison differ in their level of detail (as when comparing one data-level factor with one aggregated factor), the more detailed item tends to receive more weight than it should.

One remedy for this bias, used by RSCREEN as well as by practicing decision analysts, is known as "bottom-up" weighting. Each aggregate factor is represented by one of its data-level subfactors as a sort of "proxy." Initially, RSCREEN uses for its proxy the subfactor with the highest weight (although that arbitrary choice is not at all critical to the success of the procedure). The elicited comparisons are thus performed not on the aggregated factors, but upon their data-level proxies; since the data-level weights have already been assessed, the proper weights for the higher-level factors can be determined from the ratio of the proxy factor weights, as shown in the following example.

Suppose, as shown above, the "environmental constraints" factor has three data-level components: "transit," "local area," and "other," with relative weights of 47, 24, and 29, respectively. Furthermore, suppose another factor, "availability of resources," has four subfactors: "personnel skills," "training," "equipment," and "other," with relative weights of 36, 20, 18, and 26, respectively. Now, rather than compare "environmental constraints" directly with "availability of resources" (a difficult and abstract task), the bottom-up procedure will select two proxies (e.g., "transit" and "personnel skills") and determine the ratio between the proxies in much the same manner as if they had appeared on the same branch of the tree structure.

The procedure might involve more than two factors, each represented by a data-level proxy. For each factor, the array displays the factor name, the name of the proxy subfactor, and the best and worst options with respect to the proxy subfactor. Then, as before, RSCREEN asks the user to determine the rank order of the swings on the proxy factors and to express each proxy weight on a ratio scale, again assigning the factor with the most important swing a nominal value of 100. Continuing the two-factor example, suppose "transit" was judged to have a more important swing than "personnel skills," such that the relative weights were 100:25 (or, normalized, 80:20.) Now, sufficient information exists to determine the appropriate weights for the higher-order factors: "transit" accounts for 47% of the weight attributable to "environmental constraints," and "personnel skills" accounts for 36% of the weight for "availability of resources." By dividing the proxy weights of 80 and 20 by 47 and 36, respectively, we can obtain the true weights, which are in the proportion $(80 \div 47):(20 \div 36) = 1.70:0.56 = 75:25$. Thus, in this example, "environmental constraints" would receive a weight of 75, while "availability of resources" would receive a weight of 25.

In this manner, working from the bottom of the tree up to the top, weights at each level can be determined, until all of the weights on the tree structure have been assigned values.

3.3.1.6 Calculation and display of results - Once all of the inputs have been elicited, RSCREEN calculates the overall scores on the aggregate attributes by combining the component scores in proportion to the weights they have been assigned. These scores (at both the aggregate and the component levels) may then be represented in a compact matrix format (such as the one in Table 3-1), or in an easier-to-read diagrammatic version which depicts the tree structure underlying the model (see Figure 3-4). This display of results marks the end of the initial modeling effort.

3.3.2 Model explication, exploratory analysis, and modifications - Once the initial model has been constructed, the user may wish to review some of its implications, to examine the possible impact of certain modifications, and perhaps to edit the values in the light of further consideration or new information. By storing the results of a model on the computer's magnetic disk, RSCREEN permits such analysis to be executed in a risk-free manner. Using the model's results as a basis, RSCREEN then allows three further capabilities: model explication, sensitivity analysis, and model editing.

3.3.2.1 Model explication - It is often helpful for the user to specify along with the numerical and structural inputs a set of definitions for the options being considered, and a paragraph of rationale for each attribute, explaining why the options were rated as they were. This information may be entered into the computer and stored on the magnetic disk along with the model, either at the time of initial model construction, or subsequently as an after-the-fact explanation. Apart from the obvious benefits of documentation for future reference, the process of verbalizing explicit definitions and rationale provides the user with the opportunity to re-examine the assumptions and the

CA1: TAKE NO ACTION. AWAIT FURTHER DEVELOPMENTS.

CA2: COMPLAIN IN U.N. WARN USSR AND RAMBO.

CA3: CONDUCT NIGHT RAID. DESTROY OFFENSIVE WEAPONS.

1 POLITICAL

	--FACTOR--	WT	CA1	CA2	CA3	CUMWT
1)	DOMESTIC RELATIONS	*(53)	100	70	0	14.5
2)	FOREIGN RELATIONS	*(42)	0	100	30	9.0
3)	INTERAGENCY COORD	*(5)	100	50	0	1.1
	--TOTAL--					

Table 3-1

MATRICES FOR RSCREEN RESULTS

factual knowledge underlying his model. Such re-examination may result in the identification of new options to consider; it may identify areas of conceptual or factual weakness where more information or further analysis would be useful; or it may simply provoke new insights into the problem and the model. The rationale and definitions can be stored, recalled, displayed, and edited along with the numerical model, providing simultaneous access to both types of information.

3.3.2.2 Sensitivity analysis - RSCREEN enables the user to perform two different kinds of sensitivity analysis. The first of these determines, for any aggregate attribute, the minimal change in the weights of its components which would result in a shift of preference from the option which currently has the highest aggregate score (on that attribute) to some other option. The second allows the user to examine the impact on overall score of a specified change in the percentage of overall weight attributed to any subfactor, by varying the factor's weight between two user-specified limits. Both varieties of sensitivity analysis allow the user to place his assessed model values in perspective, in order to see just how critical a change in the particular values selected would be.

"Next-best" sensitivity analysis. The following procedure is used to determine the minimum change necessary for unseating the first-place option.

If there are n attributes and m courses of action, the evaluation procedure calculates

$$\bar{b}_i = \sum_{j=1}^{n} w_j \, b_{ij}$$

for each course of action i. Here $(w_1, \ldots, w_n)$ are the weights and $(b_{i1}, \ldots, b_{in})$ are the scores for course of action i. The best course of action is the one with the highest weighted score $\bar{b}_i$.

The problem to be solved in performing the sensitivity test described above can be stated as follows. Let $\{\overset{o}{w}_j\}$ be the weights initially elicited; suppose with these weights, course of action p is best. Then the problem is to determine $\underline{w}$ such that

$$\sum_{j=1}^{n} (w_j - \overset{o}{w}_j)^2$$

is minimized, subject to the constraints

$$\sum_{j=1}^{n} w_j\, b_{ij} < \sum_{j=1}^{n} w_j\, b_{pj}, \text{ all } i \neq p,$$

$$\sum_{j=1}^{n} w_j = 100, \text{ and } 0 < w_j < 100, \text{ all } j.$$

This will find the nearest point in $\underline{w}$-space to $\overset{o}{\underline{w}}$, such that course of action p is equivalent to some other course of action.

Although the problem as formulated is a quadratic program and can, therefore, be solved by using a quadratic programming algorithm, there is a more efficient algorithm which employs the special structure of the problem, and this has been used here.

It is convenient to talk of regions of the feasible part of $\underline{w}$-space in which each option has best weighted score. "Region i" is the part of $\underline{w}$-space in which $\bar{b}_i > \bar{b}_j$, for all $j \neq i$. The problem consists of finding the point on the boundary of region p which is nearest to $\overset{o}{\underline{w}}$. Note first that if the perpendicular distance from $\overset{o}{\underline{w}}$ to the hyperplane on which $\underline{\bar{b}}_q = \underline{\bar{b}}_p$, for some other option q, is in fact on the boundary of region p, then that perpendicular distance is a contender for the minimum distance before change occurs.

The following algorithm emerges:

1. For each $i \neq p$, calculate the perpendicular projection of $\overset{o}{\underline{w}}$ onto the hyperplane

$$\sum_{i=1}^{n} w_i (b_{ij} - b_{pj}) = 0$$

subject to the constraint

$$\sum_{i=1}^{n} w_i = 100.$$

This is given by the expression:

$$w_i = \overset{o}{w}_i - \frac{nC}{nB - A^2} b_{pi} + \frac{AC}{nB - A^2}$$

where $A = \sum_{j=1}^{n} b_{pj}$, $B = \sum_{j=1}^{n} b_{pj}^2$, $C = \sum_{j=1}^{n} \overset{o}{w}_j b_{pj}$.

2. Check to see if the point so calculated satisfies $w_i > 0$, for all i; if not, construct a new problem by omitting all components of each vector corresponding to indices for which $w_i < 0$, and recalculate a similar expression to the one above (with new definitions of n, A, B, C, and subject this time to the constraint

$$\sum_{i=1}^{n} w_i = \sum_{i=1}^{n} \overset{o}{w}_i).$$

Iterating this procedure (several times if necessary) will produce a reduced vector all of whose elements are non-negative. A vector in $\underline{w}$-space is now created by inserting zeroes for all components which have been excluded by this procedure.

The vector created by this method is the nearest point from $\overset{o}{\underline{w}}$ to the hyperplane $\bar{\underline{b}}_i = \bar{\underline{b}}_p$. Let us call it $\overset{o}{\underline{w}}_i$.

3. Now check to determine if $\overset{o}{\underline{w}}_i$ is "covered" in the sense that there is some $j \neq i$ or p such that

$$\sum_{k=1}^{n} \overset{o}{w}_{ik} \, b_{jk} > \sum_{k=1}^{n} \overset{o}{w}_{ik} \, b_{ik}.$$

If this is the case, the nearest point to $\overset{o}{\underline{w}}$ is not in region i, and this possibility can be excluded.

4. For all nonexcluded $i(\neq p)$, calculate the distance from $\overset{o}{\underline{w}}$ to $\overset{o}{\underline{w}}_{ij}$. The smallest of these indicates where the minimum change occurs, and it is this vector which represents the smallest change in weights necessary to change the preferred option.

Using the above procedure, RSCREEN calculates and displays the closest set of weights which results in a change of preference, alongside the original set of weights, as well as the originally preferred action and one which would replace it if the weights were changed.

"Single-factor" sensitivity analysis. Using this option, the user selects any attribute (data-level or aggregated), and specifies a range over which the weight for that attribute is to be varied. RSCREEN then generates ten equally spaced intervals between those two extremes, and for each of the generated values, calculates the overall scores for all of the options, assuming the weights of the remaining options are altered proportionally to their original values (so that the total weights add to unity). Also, for each of the generated weights, RSCREEN indicates by an asterisk the option which would receive the highest score; thus, it is easy to visualize the magnitude of change needed to "flip" the decision from one option to another. An example of the display from this sort of sensitivity analysis appears in Table 3-2.

3.3.2.3 Sorting of attributes - In addition to performing sensitivity analysis, RSCREEN permits the user to inspect the data sorted for display in a variety of useful ways.

Sort by weights. In this variety of sort, the data-level attributes are arranged in order of their proportions of the total weight (i.e., in accordance with the weights determined by the procedures described in Sections 3.3.1.3, 3.3.1.4, and 3.3.1.5. Here, the attributes at the top of the list are those whose best-to-worst swings are most significant in magnitude (irrespective of the direction of those swings). A display of this sorting procedure may help the user to focus further attention on those attributes which are the most critical in the sense of having the potential for large impacts on the overall score.

Sort by "pro/con". This sorting procedure compares two options: the one with the highest overall score, and the one with the second-highest score. For each data-level attribute, the difference between those two options' component scores is multiplied by the weight for that attribute, resulting in an index of discrimination. For example, if the topmost option has received a score of 80 on a particular attribute, while the overall second-best option has received a score of 62, and if the weight assigned

2.2 - COMMAND AND CONTROL CURRENT CUMWT: 11.5

THE MINIMUM CUMWT IS?: 5

THE MAXIMUM CUMWT IS?: 30

2.2 - COMMAND AND CONTROL CURRENT CUMWT: 11.5

WT	CA1	CA2	CA3
5.0	56	74*	70
7.5	57	72*	70
10.0	58	70*	69
12.5	60	68	69*
15.0	61	67	68*
17.5	62	65	67*
20.0	64	63	66*
22.5	65	60	66*
25.0	66*	58	65
27.5	68*	56	64
30.0	69*	55	64

Table 3-2
RSCREEN SENSITIVITY ANALYSIS

to that attribute is .15 of the total weight, the discrimination index will be .15 x (80 - 62), or 2.7. If the most highly rated option scores lower than the second-best option on a particular attribute, the discrimination index for that attribute will be negative. By examining the discrimination indices (ordered so that those attributes which most favor the top-rated option are first, and those which favor the second-best option are at the bottom of the list), the user can easily identify the most important factors which caused the top two options to receive the overall values they did, and to detect any possible inconsistencies or inaccuracies which might indicate the need for editing or sensitivity analysis.

Sort by correlations. In this type of sorting procedure, the component scores on a given attribute are correlated with the overall scores which would result if that attribute were omitted from consideration. For example, if a set of options receives scores of 30, 0, 80, 100, and 60, respectively, on one attribute, and overall scores of 25, 10, 60, 65, and 35, and if the weight of the attribute under consideration has been assessed as .40, the correlations will be computed in the following manner:

- compute the component of overall scores accounted for by the attribute under consideration (in this case, those components would be .40 times each of the corresponding component scores, or 12, 0, 32, 40, and 24, respectively);
- subtract these weighted component scores from the total overall scores (in the example given, the differences would be 13, 10, 28, 25, and 11, respectively);

- compute the correlation between the scores on the component under scrutiny and the adjusted totals obtained by the subtraction (in this case, the correlation between the x and y values in the following table):

X	12	0	32	40	24
Y	13	10	28	25	11

Here, the correlation is .793, indicating a fairly high degree of correspondence.

For those attributes which are highly positively correlated with the adjusted total scores, it will be difficult to influence the overall result simply by altering the weights; in a sense, those attributes constitute a redundant justification for the outcome of the remainder of the model. A high negative correlation, on the other hand, indicates an attribute which seems to be in conflict with the overall thrust of the remainder of the model; thus, if such an attribute were increased sufficiently in weight, the options would tend to reverse in order of preference. Those attributes with highly negative correlations are therefore critical ones in terms of sensitivity to changes in weight. If an attribute's scores are almost uncorrelated with the adjusted overall scores, the results of a change in weight for that attribute will tend to be a "random" rearrangement (rather than a strict reversal) in the order of preference among the options; since such changes may still affect the position of the most highly rated option, these low-correlation attributes still deserve considerable attention and possibly more detailed sensitivity analysis.

3.3.2.4 Model editing - For a number of reasons, the user may wish to replace the scores, weights, definitions, or rationale with new values or descriptions. New information may have become available, or further thinking may have motivated a reconsideration, or errors in the mechanics of elicitation may have been detected. RSCREEN allows the user to make such changes by reassessing the necessary data and performing the implied calculations to obtain updated values. These new values may be stored permanently on the disk, used as a basis for further sensitivity analysis, or modified further, according to the user's wishes.

3.3.3 Refinements and continuations - After the user has constructed the initial model, examined its implications, and modified it if necessary, additional time might be available to refine the model. The resulting refinements might produce more reliable, bias-free numerical assessments, or greater detail, or extension of the original model to include additional factors of secondary importance. RSCREEN provides three specific capabilities to aid the user in this continued analysis: "least important factor" comparison weighting, secondary decomposition, and secondary analysis.

3.3.3.1 "Least important factor" comparison weighting - In its initial model elicitations, RSCREEN determines the relative weights of the aggregate factors by comparing the most important (i.e., most highly weighted) subfactors as proxies (see Section 3.3.1.5). In theory, the selection of a subfactor to use as a proxy is arbitrary, and any rule should produce the same ultimate results. However, the noise inherent in any subjective measurement procedure, together with possible biases resulting from the selection of the most important subfactor for each attribute, could produce inaccuracies in the calculated values.

One remedy for this problem would be to repeat the cross-comparison procedure that was used to weight the higher-order attributes, this time using different subfactors as proxies. Once again, the choice of a selection rule is arbitrary; RSCREEN uses the least important subfactor in each attribute for a proxy this time. Apart from this change, the procedure used in the elicitation of higher-order factor weights is identical to the one described in Section 3.3.1.5. Typically, the method will produce a somewhat different set of overall weights for the aggregate factors, and therefore a different set of overall scores for the options.

Often, the newly assessed values and weights will represent a change, but not one of sufficient impact to alter the user's indicated decision; in this event, the "least important factor" weighting will have served to demonstrate robustness, thus increasing the user's confidence in the model's results. On the other hand, should the second assessment procedure lead to substantially different calculated values and weights, the inconsistency may be regarded as evidence of a need for more careful thinking about the values; an effort to reconcile the inconsistency might pay off with not only a better set of numbers but also a better understanding of the problem itself.

3.3.3.2 Secondary decomposition - Sometimes, a user is uncertain about the proper scores for a given data-level factor on the grounds that that factor is itself too aggregated to think of as a whole. In situations like that, it would be of some use for the decision maker to separate that factor into even smaller subfactors, each of which should be easier to assess than the original one. Because this additional level of detail is likely to be necessary only in a few circumstances, and because the specific subfactors chosen may depend on the current problem being decided, it would be impractical for RSCREEN to attempt to pre-can any such secondary decompositions. On the other hand, because there is no guarantee that the user will include all of the important factors in the secondary decomposition, it is perhaps best viewed as an exploratory procedure.

When a user specifies (as a part of the editing stage of the analysis) that he would like to perform a secondary decomposition, RSCREEN has the user identify the attribute to be decomposed and specify the subfactors into which it will be subdivided. Then, using the same procedures as the initial score and weight elicitations (see Sections 3.3.1.3 and 3.3.1.4), RSCREEN guides the user through the process of assigning scores and weights. Then, since the secondary decomposition provides only one additional level of analysis, RSCREEN is able to calculate the implied scores for the various options on the attribute which has been subdivided. These scores are not (at least in the current version) entered into the existing numerical model, nor is the model altered to include the newly defined subfactors. Instead, RSCREEN treats the procedure as an off-line service to the user, who can then determine whether to enter the calculated values into the model, to adjust them in some way, or simply to ignore them. Of course, if the values are used, the user will have a printed copy of the subfactors and their related scores and weights, for briefing and documentation purposes.

3.3.3.3 Secondary analysis - The attributes in the pre-stored utility structure used by RSCREEN must be designed for generality of application and therefore may not be a completely exhaustive list with respect to any specific problem. Although the pre-canned attributes should cover most of the factors (otherwise, the model would be inappropriate!), the user may wish to investigate a few additional ones, with RSCREEN's help.

RSCREEN selects the three most highly rated options resulting from the standard analysis (there must be at least three options for this procedure to work), and proceeds to elicit from the user names for attributes other than those already incorporated in the model, attributes which serve to discriminate further among the three selected options. Thus, in a sense, RSCREEN uses its initial routine to select a few promising options and then focuses in more detail on those options by collecting new information.

The analytic procedure used in RSCREEN's secondary analysis routine is adapted from MAUD (Humphreys and Wisudha, 1979), which is described in general terms in Section 2.5.3 of this report. The specific steps used in the RSCREEN routine can be summarized as follows:

- RSCREEN presents the three option names and asks the user to specify a (previously unmentioned) way in which one of the three options differs from the other two;

- the user names the two extremes with respect to the indicated dimension, in response to a sentence completion questions such as "Option X is/has ________; on the other hand, Option Y and Option Z are/have ________.";

- RSCREEN then generates a seven-point scale, using the two indicated extremes as endpoints, and asks the user to locate each of the options on that scale;

- next, RSCREEN asks the user to identify the "ideal" location on that scale and calculates the score for each option as the distance from its location on the scale to that ideal point;

- o RSCREEN repeats the above procedure for at least two different attributes and continues until the user cannot specify any further factors;

- o finally, RSCREEN guides the user through a standard weight elicitation procedure, similar to that described in Section 3.3.1.4, in order to calculate overall scores for the options.

Currently, RSCREEN simply uses the above routine to develop a model of factors independent of the original model, in order to discriminate among the most promising original options. In future developments, the information contained in this newly constructed model may be integrated into the original model to reconcile any apparent inconsistencies, but recent experience has indicated that the secondary analysis is best used primarily when the three top options are very close on the original analysis, and therefore the secondary factors should determine the true preference. Of course, the final arbiter of such preferences should be the decision maker himself, so that when the final recommendation is made, all of the relevant factors can be reconsidered in their proper perspective.

3.4 RSCREEN Summary

RSCREEN provides a decision maker with a pre-canned utility structure which characterizes a particular type of problem. For any specific decision, the user will be guided through an initial model construction phase during which options are listed and the relevant scores and weights are elicited, resulting in a calculation of the overall utility for each option. Further operations permit the user to document the definitions of the terms and the rationale behind the scores involved, as well as allowing him to perform sensitivity analyses and to edit the model as needed.

Finally, three *post hoc* procedures ("least important factors" weighting, secondary decomposition, and secondary analysis) are available to continue the analysis by providing more robust, better refined, more detailed, or more complete models.

4.0 OPGEN: A MULTI-PHASE AID FOR RISKY DECISION MAKING WITH PRE-STORED GENERIC OPTIONS

In this chapter, the theory and procedures of a decision aid approach called OPGEN (OPtion GENeration) will be discussed. The discussion aid will focus on the OPGEN approach to decision aiding, rather than on the specifics of the experimental prototype software implementation which DDI has programmed on the IBM 5110 portable computer. Because OPGEN marks the first appearance of a generic approach to model structuring, many of its specific procedures should be regarded as illustrative techniques, rather than the best or only possible methods.

4.1 Background

The following characteristics apply to a number of decision problems, representing a rich enough field of applications to justify a specially designed methodological approach:

- o the problem involves selecting a course of action whose outcome may depend on uncertain events beyond the decision maker's control;

- o the evaluation of any outcome may involve a tradeoff among effects on a variety of criteria·

- o although the specific courses of action to be evaluated may depend strongly on the context (geographic location, time, resources available, etc.), almost all reasonable options can be captured within a small number of generic categories (simply by abstracting details).

Because fast reaction in these situations is often imperative, it will be useful to provide the user with the ability to generat a rough analysis as quickly as possible, and then to use additional available time to refine the model by incorporating more and more of the detail available, while generating more and more accurate and specific assessments. Thus, OPGEN (1) provides an initial analysis at the highly generic level; (2) selects, in subsequent iterations, the options which appear most promising; and (3) describes those options in more detail and subdivides them into specific instances for closer evaluation.

4.2 Technical Foundations

The decision-analytic framework underlying OPGEN is a simple one-stage decision with uncertain outcomes and multiple evaluation criteria. In other words, the space of options is a finite set; an option is selected, and then when events unfold, the outcomes can be measured on a variety of attributes, which can be combined to form an overall measure of utility for the particular action-event sequence.

A simplifying assumption which characterizes OPGEN (and which is at least approximately valid for many problems) is that the choice of any particular action does not affect the relative probabilities of the various event sequences which might occur; that is, the events are independent of the action chosen, rather than direct or indirect responses to that action. For example, whether or not you carry an umbrella on a given day does not alter the probability that it will rain where you are; but it certainly affects your probability of losing the umbrella! Thus, if the former consideration is the event to be considered, OPGEN will be appropriate; on the other hand, if the latter consideration enters into the model, a more complex type of model will be needed (such as the influence diagram or the full-scale decision tree). Typically, by using the decision analyst's expertise in constructing the original pre-canned model, it is possible to minimize the impact of act-event dependencies, and therefore to adapt the OPGEN method to a wide variety of problems without difficulty.

A second simplifying assumption is that the criteria which will be used to evaluate outcomes are organized simply, rather than hierarchically. While this restriction imposes no limitation on the range of problems to which OPGEN is applicable, it does suggest a preference for problems whose main difficulty stems from the uncertainties involved, rather than from complex trade-offs among large numbers of attributes.

One somewhat unusual feature of OPGEN is a departure from traditional convention: instead of assessing the utility scores on a 0-to-100 point scale, OPGEN elicits scores in the form of regrets, on a scale from -100 to 0. Because the utility scores in question form an interval scale, there is no technical or theoretical significance to the change in scoring systems. (One could, for example, simply add 100 to every value to achieve a positive score in the usual range.) However, because most of the problems for which OPGEN was designed involve risks, and because the risks are perceived primarily in terms of the potential regrets which might ensue, the "negative" approach has received favorable response from users.

The following procedural outline represents a skeleton version of the analysis which characterizes OPGEN's approach. As shown in Section 4.3, this skeleton framework actually occurs more than once, as iterations provide selection, focus, and refinement to the model.

- Select and define a number of options.

- Identify the event-sequence scenarios which might affect the outcome of any option (i.e., which might lead to preferring one option over another).

- Assess the probability of each event-sequence scenario.

- For each attribute in the pre-canned list of attributes, develop a matrix of regrets, on a scale for which a score of -100 represents the maximum possible regret given any specified scenario, while 0 represents the best (i.e., no regret) option for any scenario.

- Assess the relative importance weights for the various attributes.

- By combining the regret scores in the matrices, weighted according to the scenario probabilities and the attribute weights, determine the overall expected regret score for each option.

Having performed the above operations, the user can now simply proceed to implement the most attractive option, or he can continue to refine the selection by narrowing his attention to a few promising ones, while expanding those options into more detailed variants (e.g., an option which originally involved simply "deploying aircraft" might, upon further analysis, be subdivided into "deploy 3 or 4 squadrons of aircraft," and "deploy 5 or 6 squadrons of aircraft").

The value of an OPGEN approach stems from two important features: first, the fact that pre-canned generic options, events, and attributes ensure a model which can encompass most of the vital information in a minimum of time; and second, the ability to tailor the level of detail and the specificity of the model to the time available, without losing track of the need for a speedy but complete decision.

4.3 A Procedural Guide to OPGEN

This section summarizes the sequence of operations in the currently designed OPGEN routine. While a prototype has been programmed on the IBM 5110 portable computer, this prototype is still experimental in nature, and several immediate improvements are in order before field testing can be profitably undertaken. In particular, although the programmed routines are analytically correct, they are often slow and difficult to operate and may require more extensive tutorial and explanatory features for successful use. Therefore, the present report will deal with a more or less idealized version of OPGEN, technically similar to the experimental prototype, but different in a number of details.

OPGEN operations can be subdivided into three phases: (1) initial generic analysis, (2) detailed analysis; and (3) option expansion. In theory, after the third phase, the newly expanded options should be used to iterate phases (2) and (3), until the user is satisfied that the best options have been considered (or until time runs short). Currently, OPGEN halts after phase (3); in practice, this is likely to prove a reasonable stopping point, but it would be advisable to provide the user with the ability to loop through phases (2) and (3) iteratively if he so wishes. The remainder of this section describes the three phases of OPGEN's operations in detail.

4.3.1 Initial generic analysis - During this phase, OPGEN helps the user to select an appropriate generic framework for his problem, and then uses a combination of pre-canned options, events, attributes, and values, together with simple direct elicitations to provide a preliminary screening of the generic options, indicating the most promising areas for further, more detailed investigation.

4.3.1.1 Selection of a generic model - The current version of OPGEN includes five pre-stored generic models: NEO (Noncombatant Evacuation Operations); Counter-Terrorism--Hostage or Takeover; Counter-Terrorism--Recover Nuclear Weapons; Show-of-Force; and Force Augmentation. The user may select one of these models, or he may select "none of the above" in which case a new set of generic options, events, and attributes will be elicited. Since the "none of the above option" is a default which fails to capture the advantages of a pre-canned model, the current discussion will assume that the user has selected one of the pre-canned generic models. In particular, for purposes of illustration we shall assume that the model selected was NEO.

4.3.1.2 Initial screening of generic options - OPGEN first displays a list of several generic options, each with a definition. The options available should cover practically all of the possible actions and should be divided in such a way as to capture the essential choices facing the decision maker. In the case of NEO, they may range from doing nothing to committing a large force in a complex operation involving penetration of a hostile area to remove a large number of personnel.

After the generic options and their definitions have been displayed, the user may determine that some of those options are either impossible given the situation and available resources, or undesirable in some obviously critical way. Those options may be eliminated from consideration immediately, thus saving the time and effort involved in evaluating noncontenders.

4.3.1.3 Pre-canned elicitation of event probabilities - OPGEN has, as part of its pre-canned model, the various events which might occur to influence the outcome of any action. In the case of the NEO problem, crucial events will be the following:

- o Will an evacuation be necessary?
- o If so, when will it occur?
- o If an evacuation occurs, will it be permissive or nonpermissive?
- o If an evacuation occurs, how many people will need to be evacuated?

OPGEN elicits (by direct questions) the probabilities of the various events. At this initial stage, OPGEN assumes that the three conditional events are independent, so that the probability of each possible event sequence can be determined simply by multiplying its component event probabilities. The event sequences (scenarios) contained in the pre-canned model can be summarized as follows:

- o no evacuation;
- o evacuation/within 72 hours/permissive/evacuate less than 500 people;
- o evacuation/within 72 hours/permissive/evacuate about 1000 people;
- o evacuation/within 72 hours/permissive/evacuate more than 2000 people;
- o evacuation/within 72 hours/nonpermissive/evacuate less than 500 people;
- o evacuation/within 72 hours/nonpermissive/evacuate about 1000 people;

- evacuation/within 72 hours/nonpermissive/evacuate more than 2000 people;

- evacuation/after 72 hours/permissive/evacuate less than 500 people;

- evacuation/after 72 hours/permissive/evacuate about 1000 people;

- evacuation/after 72 hours/permissive/evacuate more than 2000 people;

- evacuation/after 72 hours/nonpermissive/evacuate less than 500 people;

- evacuation/after 72 hours/nonpermissive/evacuate about 1000 people;

- evacuation/after 72 hours/nonpermissive/evacuate more than 2000 people.

4.3.1.4 Attribute importance weight elicitation - Another part of the pre-canned model is a list of attributes for evaluating the outcomes of any action-event sequence. As a quick screening device, OPGEN asks the user which of the attributes is most important ("of most concern") and then asks the user to rate the importance of the other attributes as a percentage of the most important one. These relative weights will be adjusted (normalized) so that they sum to 100 percent and then presented to the user for verification or modifications if desired.

For example, in the NEO model, the four attributes are:

- risk to potential evacuees;

- degradation to primary mission;
- training and readiness; and
- political risk.

The user may, for example, judge "risk to potential evacuees" to be the most important and then express the four weights as 100, 50, 10, and 25, respectively. Normalized, these weights would transform proportionally to (approximately) 54, 27, 5, and 14.

4.3.1.5 Calculation of tentative best options - OPGEN has within its pre-stored model a pre-canned regret matrix, capturing the opinions of the experts who developed the model about the general value of each possible action given each possible sequence of events. Of course, this pre-canned regret matrix is designed to apply to an entire generic set of problems (in this case, to all NEO problems) and therefore may err by neglecting some unique feature of the specific problem at hand. However, for a rapid screening before the specific information about the problem can be evaluated, it will prove worthwhile to identify some of the options as generally promising and others as unlikely candidates. Therefore, OPGEN determines, for each option on each attribute, an overall expected score by combining the regrets associated with the various option-scenario combinations, each weighted according to the probability of the designated scenario. Then, for each option, the attribute scores are combined according to the weights elicited. Thus, for each possible option, a tentative overall expected regret can be calculated.

Rather than present the necessarily inaccurate scores calculated by this procedure, OPGEN simply suggests, based on those scores, the four apparently best generic options. Those options (plus any others that the user has special reason to designate for consideration) will form the basis for the next phase of the analysis.

4.3.2 Detailed analysis of promising options - During the second phase, OPGEN helps the user to specify in greater detail the factors which contribute to the choice among options.

4.3.2.1 Designation of additional options - At the end of the first phase of the analysis, the user has been presented with four suggested generic options which seem to be the most promising. Because it is possible that additional options--either those which had been previously eliminated or some which might be unique to the present problem environment--might represent viable choices, OPGEN asks the user to specify any such additional options. These new options, if indicated, should be comparable in level of detail to the original options and should represent different approaches from those listed, rather than elaborations of methods of accomplishing the ones already under evaluation.

4.3.2.2 Refinement of event/outcome model - The original set of events considered by the pre-canned model may be only a fair approximation of the user's conception of the problem at hand. In order to achieve a more accurate analysis, it will be necessary to reconsider the possible events and outcomes, with an eye toward eliminating those events which fail to impact the evaluation of any options, rephrasing the possible outcomes so that they correspond better to the user's own distinctions and definitions, and adding new outcomes when they are needed.

For each of the events in the pre-canned model, OPGEN first asks if prior knowledge about the outcome of the given event would affect the user's choice of an action. If this would not be the case, it may be that the event truly has no impact on the evaluation (and could therefore be safely eliminated from the model); on the other hand, it may simply be the case that the outcomes need to be redefined in some more appropriate way. For example, it might be the case that the time available for evacuation is in fact an important factor, but that the proper distinction is not one of "less than 72 hours" versus "greater than 72 hours," but rather "within 24 hours" as opposed to "after 24 hours." Even if the specified event does impact the choice of actions, it may still be desirable to sharpen the distinction by redefining the outcomes, adding new ones which might have different effects, or by combining some which seem to behave in more or less the same way as one another.

OPGEN reminds the user that if the outcomes of any event are to be redefined, the newly specified outcomes should be mutually exclusive: it must be possible for only one of the outcomes of a given event to occur. Furthermore, the list of outcomes must be exhaustive: exactly one of the listed outcomes must occur. As a general rule of thumb, OPGEN suggests that the outcomes be defined in a manner no more detailed than is necessary to help distinguish among the decision options.

Once the user has eliminated or redefined the outcomes for each event, OPGEN provides a review of the new set of events and their outcomes and asks the user to verify the list, making any further changes that are indicated. Once the user indicates satisfaction with the list, the indicated changes are finalized.

4.3.2.3 Reassessment of outcome probabilities - For any event whose outcomes have been redefined or otherwise altered, it will be necessary to reassess the outcome probabilities. Furthermore, the user may, upon reflection, wish to refine or alter his assessment of some of the other probability judgments. OPGEN presents each event for which outcomes have been changed, and directly elicit the new probabilities. Once all new probabilities have been entered, OPGEN displays a complete listing of the events, outcomes, and probabilities for the user to inspect. At this point, the user is free to alter the probabilities of any other event outcomes he wishes to change.

4.3.2.4 Factorial combination of event outcomes into scenarios - By combining all feasible sequences of events, using the redefined outcomes and their new probabilities, OPGEN constructs a complete (mutually exclusive and exhaustive) list of the possible event scenarios. For example, after editing, the event scenarios listed in Section 4.3.1.3 might have been transformed into the following:

- must evacuate/within 24 hours/voluntary;
- must evacuate/within 24 hours/all nonessential personnel;
- must evacuate/within 24 hours/all civilians;
- must evacuate/after 24 hours/voluntary;
- must evacuate/after 24 hours/all nonessential personnel;
- must evacuate/after 24 hours/all civilians;
- no evacuation required.

4.3.2.5 Ranking of options for each scenario - The next step involves asking the user to rank the options in order of preference, assuming a particular scenario will occur for certain. This ranking will be undertaken for each of the possible event scenarios. By comparing the rankings on the various scenarios, OPGEN may detect that the same ranking of options occurs in two or more different scenarios. In this event, the user is asked if in fact those scenarios may be considered instances of a single sequence of events. If so, the similar scenarios will be combined and a new name elicited to apply to the aggregate category; if the user does not wish to combine the scenarios, they will remain separate. Once a final set of scenarios has been arrived at, the user is invited to provide shorter, more meaningful titles to characterize each scenario.

4.3.2.6 Screening of attributes for relevance - The original list of attributes in the pre-canned model may be designed to apply to a variety of situations, and therefore may contain a number of attributes which are not important in distinguishing among the options for the particular problem at hand. To shorten the remainder of the analysis, OPGEN asks the user to identify any attributes on which no preference can be detected for any option-scenario outcome with respect to any other. If a difference can be detected, the attribute is relevant and therefore included in the model; if no difference can be found, this is evidence that either the performance of all possible options in all possible scenarios is identical or that the attribute itself is of no importance in evaluating outcomes. In any event, OPGEN can eliminate from consideration those attributes for which the user indicates no preference for any option-scenario combination.

4.3.2.7 Elicitation of additional attributes - Although the pre-canned generic model strives to incorporate as many of the generally important attributes as possible, additional attributes may arise, relevant to the problem at hand, which are not covered by the original list. The user is now given the opportunity to suggest other attributes which might prove important.

To uncover such additional attributes, the user might try to think of any pair of options (or option-scenario outcomes) which are approximately equal with respect to the current list of attributes, but between which the user still feels a distinct preference. If he can think of any such pair and then describe how they differ, it will be possible to identify new attributes which should be added to the list.

4.3.2.8 Relative regret elicitation for each attribute - OPGEN uses a two-step procedure to elicit the regret matrix for each attribute: in the first step, relative regrets are determined for each of the scenarios (with respect only to the attribute under consideration); in the second step, the magnitudes of the maximum possible regrets for each of the scenarios are compared to provide an overall scale of regret for the attribute.

Within-scenario regret elicitation. For each of the possible scenarios, the user is asked to identify the best possible outcome (with respect to the attribute being considered); that option is assigned a regret value of zero. Next, the user identifies the least desirable option (again, for the given attribute and the given scenario); that option is assigned a regret value of -100. Finally, the user is asked to provide appropriate regret scores for the remaining options, on a scale from -100 to 0. These scores, once elicited, are presented for the user's verification and for any indicated modifications.

Across-scenario regret scaling. In this next stage, OPGEN displays each of the option-scenario combinations which received a score of -100 for the scenario.

The user is then instructed to rank-order these worst-case combinations according to how bad they would be (identifying the worst combination first, then the next-worst, and so on). Having determined the order among the option-scenario combinations, OPGEN then elicits their relative regret scores, using the following procedure.

The option-scenario combination judged the worst overall (with respect to the attribute being considered) is assigned a value of -100. Now, OPGEN displays a scale, on which the user may indicate the relative regrets for the remaining combinations, in the appropriate order. For example, suppose at a later stage of analysis in the example we have been considering, we are evaluating regrets with respect to the attribute "risk to potential evacuees." Suppose further that the four options under consideration are the following:

- o alert air-mobile security force;
- o forward base in-theatre aircraft;
- o deploy CONUS aircraft; and
- o deploy CONUS ground evacuation force.

The following list indicates the five scenarios and the worst option associated with each:

1. late evacuation/nonessential or all civilian personnel: alert air-mobile security force;

2. late evacuation/voluntary: alert air-mobile security force;

3. no evacuation: deploy CONUS/aircraft;

4. early evacuation/voluntary: deploy CONUS ground evacuation force;

5. early evacuation/nonessential or all civilian personnel: deploy CONUS ground evacuation force.

If the user has indicated that the regret due to option-scenario combination (5) is the worst, the following display appears:

```
PRESERVING THE FOLLOWING RANK ORDER ( 3 2 1 4 5 ),
INDICATE THE RELATIVE REGRETS OF THE REMAINING OPTION-
SCENARIO PAIRS ON THE SCALE:

     0      -20      -40      -60      -80      -100
     +------+-------+-------+-------+--------+
                                               5
```

The user then enters the digits 3, 2, 1, and 4 in the appropriate places on the scale (preserving the left-to-right order).

OPGEN then transforms the indicated scale locations to the corresponding numerical values and presents them to the user for inspection, verification, and editing, if necessary. Having rescaled the worst option-scenario pair for each scenario, OPGEN then rescales the remaining regret values in proportion. For example, if the four options received regrets of -100, -40, 0, and -20, with respect to the scenario "late evacuation/nonessential or all civilian personnel," and the worst of those combinations (i.e., the one involving the option "alert air-mobile security force") was rated -62 on the scale when compared with the overall worst option-scenario pair, then the rescaled regrets will be -62, -25, 0, and -12, respectively (after rounding off).

Thus, for each attribute, a regret matrix can be constructed whose rows correspond to the options, whose columns correspond to the scenarios, and whose entries correspond to the rescaled regret values with respect to that attribute.

4.3.2.9 Assessment of relative attribute weights - The next step in constructing the detailed model is to determine the true weights which will be used to combine the various attributes regrets into a single overall regret measure. OPGEN performs this step in two stages: first, the user is asked to rank the attributes in terms of the seriousness of the worst regret for each attribute; then, using a graphic scale technique, the user determines the exact numerical weights for the attributes.

OPGEN first searches each attribute's regret matrix and identifies the one option-scenario pair which represents the most serious regret for that attribute (i.e., the one whose adjusted regret value is -100). The user is then presented with an array whose columns correspond to the attributes, and whose rows correspond to the options. Beneath the name of each attribute is the name of the scenario which contains the worst regret for that attribute. The entries in the matrix are simply the regret scores for the named scenario on each of the attributes. From this matrix, the user can identify the attribute whose maximum regret has the greatest impact, the second greatest, and so forth, until the rank order among the attributes has been established.

Once the order has been determined, OPGEN presents a scale on which the significance of the attribute with the greatest potential regret is assigned a value of 100; the user is asked to indicate by position on the scale the relative impact of the other attributes' maximum regrets. The numerical translation of these scale positions is then presented to the user for verification and possible modification.

4.3.2.10 Calculation of overall expected regrets - Having completed all the necessary numerical elicitations, OPGEN can now calculate, for each option, the overall expected regret, aggregating the attributes according to the weights determined in the previous step and combining the scenarios according to their assessed probabilities. OPGEN lists the options, together with their calculated expected overall regrets.

4.3.3 Option expansion - At the end of the detailed modeling phase described above, the user may feel confident enough to select one of the generic options, but still not know exactly how to implement it in more specific terms. Furthermore, if the outcome suggests that two or more generic options are quite close to being tied for the most promising, the user may be reluctant to eliminate the close ones prematurely. In any case, once the general aspects of the problem have been analyzed (and assuming there is sufficient time to continue), it will be of value for the user to think in more concrete terms about details of definition, implementation, and context.

The currently implemented OPGEN program limits its attention toward this phase to a simple list of questions, designed to prompt the user for specific redefinition and specification of options. For example, if the generic option "alert air-mobile security force" has been chosen for closer scrutiny, OPGEN prompts the user to answer the following questions:

- How many people are to be involved?
- What types of equipment and weapons are to be utilized?
- From where will the security force be taken?

- How many transport aircraft?

- Will riot control agents by used, and if so, which ones?

Based on the possible answers to these questions in the actual situation, the user may be able to generate several concrete action plans which could then be evaluated according to the same procedures outlined in Section 4.3.2. Although the current program has not implemented the capability, it would be desirable in general to allow the user to continue to iterate the process of option evaluation, selection, and expansion until a sufficient level of detail has been achieved for the user to feel confident about having chosen the best course of action (or until time has run out).

An extension of the currently implemented methods might incorporate additional pre-canned aids to option expansion. For example, instead of simply asking the user the option expansion questions as a mental prompt, OPGEN might incorporate a routine to generate possible options from combinations of pre-canned answers to those questions, and help the user to screen some of those combinations for further consideration. Of course, even more sophisticated methods, using artificial intelligence, semantically organized data bases, and so forth, would be possible also, but those additional capabilities--even if they were valuable improvements--would not necessarily represent further advances in the OPGEN concept itself.

4.4 OPGEN Summary

OPGEN is designed to be of use when a given generic problem occurs repeatedly, in a variety of contexts and conditions, but with sufficient similarity to justify the use of a single set of attributes and events, and a single group of generic options. In a preliminary analysis, OPGEN allows the user to screen these generic options, focusing on the few which appear to be most relevant to the given problem. Next, OPGEN performs a more detailed analysis, refining and expanding the treatment of the selected options, in order to narrow the choice further. Finally, having selected one or two of the most promising generic options, the user may (with OPGEN's assistance) expand those options into more specific, concrete alternatives for continued evaluation, selection, and expansion.

5.0 CONCLUSION: GUIDELINES FOR FUTURE DECISION AID DEVELOPMENT

A brief summary of the lessons learned during DDI's recent efforts at decision aid development will contain the foundation for a new outlook on future development. Perhaps the most important observation is that decision-theoretic correctness is far less of a constraint than the more difficult issues of user acceptance and procedural design; unless the user population can actually be motivated to utilize a decision aid, and unless the aid's procedures are capable of producing meaningful, unbiased results in the hands of an inexperienced user, analytic correctness is of no practical value. On the other hand, a well-motivating model which produces good assessments will be of value even if its analytic approach is only approximately accurate. Hence, the primary goals in future decision aid design must be simplicity, robustness, and ease of operation. Within these overall constraints, we shall discuss a number of more specific issues.

5.1 Model Determination

Except for the simplest cases, substantive information about the decision maker's problem can be divided into two types: some items are not known until the problem has actually arisen, and could not be profitably anticipated at the time a model is constructed. On the other hand, when a general class of problems of some importance can be expected to arise in the future, certain generic aspects of the situation may be incorporated into a decision aid and used to guide the analytic process. Between the two extremes--an aid which applies a single fixed model to problems of only one type, and one which provides no constraints at all on content but merely gives procedural guidance and assistance--a variety of schemes are available, each providing the user with a different amount of substantive guidance (or alternatively, imposing a different amount of substantive constraint on the user).

There can be no global solution to the determination issue: no one aid can ever be judged "optimal" without reference to the range of problems treated, the population of users, the time constraints, the substantive context of the decision problems, and the users' degree of substantive and methodological sophistication. For example, an aid like QVAL would be most appropriate in a situation where relatively sophisticated users had to deal with a variety of ad hoc problems with little substantive content in common. On the other hand, if a more or less fixed body of substantive knowledge could be used frequently enough to justify the effort of building and maintaining a data base, a GenTree approach might be more appropriate. When the problems tend to fall into more repetitive categories, so that pre-canned models might be used, either the highly determined approach of RSCREEN or the more generic one of OPGEN might prove more applicable. Naturally, those four aids represent only a sampling of the possible range of decision aid methodologies; and as further experience sorts matters out, undoubtedly, new aids will be developed which combine the best aspects of the currently existing ones, with new features yet to be conceived.

In summary, the issue of model determination can best be resolved not by deciding once and for all on a single modeling approach, but rather by tailoring a variety of approaches to different types of user populations with different sorts of problems. (The tailoring metaphor is particularly apt here, in the sense that a "one size fits all" suit of clothes is likely to prove just as unsatisfactory as a "one approach fits all" model; an individually made-to-order decision aid would be ideal, but often too expensive. The best compromise will be a selection from a variety of "off-the-rack" models which can achieve satisfactory fits with only minor alterations.)

5.2 Specificity versus Generality of Pre-Stored Models

Assuming that a user's problems have enough in common that a pre-structuring approach would be indicated, the next issue involves the degree of generality built into the models. As models become more general in nature, the possible risks involve possible errors and confusion due to the vagueness and overaggregation of the model components, and insufficient sensitivity to the specific details of individual problems. On the other hand, more generic models can apply to a wider range of problems (alternatively, a smaller library of models would be needed to cover any given range of problems), provide more flexibility in adapting to unanticipated deviations from the "prototype" problem, and achieve a certain level of simplicity and commonality which might foster more effective communication and justification for the actions selected.

It is possible, as is the case in OPGEN's option expansion routine and RSCREEN's secondary decomposition feature, for an aid to process a more general model, allowing the user to add further detail in one or more subsequent iterations. This approach seems to be an advantageous compromise in that it allows for a quick screening of options when time is extremely short, but permits the user with additional time to continue refining the model.

A further advance which bears some future investigation is a more directed way of controlling the level of detail through a combination of user inputs, a substantive data base (such as that used by GenTree), and a built-in set of heuristics for guiding the course of the analysis. Ideally, this sort of approach not only would determine the overall level of detail desired, but also would isolate those specific portions of the model where the additional detail would be useful (i.e., where it would be most likely to impact the outcomes of the model or the user's satisfaction with and understanding of the results).

5.3 Helping the User Locate and Screen Appropriate Models

When the user is presented with a number of possible models, it may not be entirely clear whether any of the available models is well suited to a new problem, or which of the models is best suited. Naturally, if the number of models is small and the user is more or less familiar with all of the models, this problem should not be serious, but as the size of the model library increases (especially if the models are highly specific ones with little flexibility), the secondary problem of locating the appropriate approach may outweigh the primary decision problem in difficulty.

One approach to this problem is to provide the user with a model-selection routine designed to solve this secondary problem efficiently by screening models according to the features, keywords, substantive areas involved, and so forth. For example, a hierarchical taxonomy could be provided to the user (something like a Dewey Decimal System for models), either as a complete directory or as an interactive routine (such as a branching questionnaire). Naturally, even this interactive method, effective though it might be for moderate numbers of models, is limited in scope, and for extremely large numbers of models, another approach would be indicated.

The way to deal with extremely large numbers of specific models is to store not the models themselves, but their components, in the form of a semantic data base such as the one used by GenTree (see Weiss, 1980). Just as humans build sentences out of a vocabulary of words or morphemes, according to grammatical rules, GenTree could help the user build models from a "vocabulary" of components, according to a set of model construction rules.

5.4 Assessment of Quantitative Inputs

Traditional (unaided) decision analysis has always depended upon the decision analyst to translate the user's subjective, and often qualitative, verbal assessments into a consistent set of numerical ratings. Even when a subject is willing to assign a set of numbers to the probabilities of various events, or to the utility scores for a set of options, the analyst must process those numbers to determine whether they are consistent with one another, as well as with the subject's previous statements about the problem. The final set of numbers must, therefore, reflect substantial intervention on the part of the professionally trained decision analyst.

In the case of a decision aid operating without the continued presence of the decision analyst, a dilemma arises: the naive "trust-the-user" approach, which simply asks the user to supply numerical probabilities, utilities, and attribute weights, is notoriously risky; on the other hand, any interactive routine designed to guarantee consistent, meaningful assessments is likely to be boring, confusing, or time-consuming. Furthermore, without the ability to perceive confusion or discouragement on the part of the user, a computerized routine is likely to behave in an insensitive manner, further discouraging the user instead of doing what the professional analyst would do, which would be to stop, offer explanations and examples, generally provide encouragement, and possibly alter the order of elicitation or the strategy and methods used.

A first approach to the elicitation problem is to use assessments which correspond in a much more direct sense to the user's understanding of the quantitative relations being assessed. For example, while numbers are merely symbols with no clearly visible relation to the quantities they represent, physical analogues (bar graphs, pie charts, maps, etc.) provide the user with a much more immediate sense of the quantities involved. The scale-location methods used by RSCREEN, QVAL, and OPGEN represent a first step in this direction. Of course, as more sophisticated graphics capabilities become available, the range for experimentation and advanced design will increase dramatically.

A different approach to obtaining better numbers would be to reduce the elicitation to smaller components (e.g., to binary comparisons rather than direct numerical estimates), requiring a much larger number of much simpler judgments, and using the computer's rapid calculation ability to identify potential inconsistencies and to translate the subject's judgments into the desired numerical formats. This approach has been partially implemented: in RSCREEN or OPGEN, the user is asked to rank order a set of options or attributes, before locating their values on a displayed scale. This method could be extended beyond the simple ordinal ranking, by asking the user to rate the gaps from one item to the next (perhaps on a simplified 0-to-5 scale). As such assessments are integrated, eventually enough information will be available to determine the implied numerical ratings as closely as desired. Furthermore, if it is correct that these more basic judgments are easier for the user and therefore more bias-free, the implied numbers may in fact be better representations of the user's true values than a direct estimation procedure would provide.

Further advances in assessment techniques are conceivable, suggesting useful areas for future research. For example, one difficulty with many assessments is the subjects' reluctance to place numerical values on what they perceive to be "soft," imprecise, or uncertain data. Verbal responses of a more qualitative nature might permit the computerized aid to infer sufficient quantitative information in the form of approximate values (or fuzzy sets) to proceed without more formal elicitations, resorting to the more precise numerical procedures only when finer-grained assessments are required. Similarly, a variety of more sophisticated graphical and sensorimotor modes of elicitation could be designed and tested, to try to maximize the amount of useful information collected for a minimum of time and effort on the part of the user.

5.5 Aids to Exploratory, Interpretive, and Extended Analysis

One of the major functions of the professional decision analyst, largely neglected thusfar in decision aid development efforts, is to help interpret the results of a decision-analytic model, exploring the impact of possible modifications or additions, and guiding the user towards appropriate continuations and extensions of the analysts. While the aids discussed in this report (especially RSCREEN and OPGEN) include preliminary steps in this direction, there is a great deal of room for advancement of concepts in the area.

Without the decision analyst's guidance, the user is likely to adopt an overly simple reaction to the results of the decision model, either rejecting it out of hand, accepting it blindly, or filing it away for future reference, without actively trying to understand the results of the model and their implications. The desired state of affairs would be quite different, with the user examining the assumptions underlying the model and the numerical assessments, examining the effects of possible modifications, looking for previously overlooked options or evaluation factors, and in general internalizing the model and its implications by integrating it into his own perspective.

Interactive routines (verbal, graphic, or hybrid) should be designed to help the user find out the answers to a variety of questions that might contribute to overall understanding and to effective extensions of the analysis. Using heuristic techniques to guide the user through the continued analysis, a decision aid could ensure more effective decision making and, at the same time, increase the user's confidence in the results of the process and his motivation to use it on future problems. Among the questions which should be considered are the following:

- What assumptions underlying the current model might be qualitatively wrong in a major sense; if those assumptions were changed or weakened, would the results of the analysis change significantly; and if so, what is the likelihood that the assumptions are in error, what additional information might be available to clarify the situation, and would that additional information be worth the cost of collecting and analyzing it?

- What are the possible error ranges around the various numerical assessments; would a plausible degree of variation in any of the assessed quantities result in a major impact on the order of preference among the options; and if this sort of sensitivity exists, is there a way to improve measurements, elicit further information, or redefine terms in such a way as to minimize the risk of error?

- Given the results of the current model, might there be additional factors, not incorporated into the existing model, which could contradict the indicated preference; if so, could they be incorporated into the model in a meaningful way?

- Are there ways of constructing additional options beyond those already considered, or of subdividing existing options into more specific categories, that could produce a new option with most of the advantages of the currently favored one, but fewer of the drawbacks; are there ways of ensuring against risks which have been identified as critical, or redesigning options to achieve a more efficient trade-off among the various goals; could some of the constraints be relaxed to permit a completely different kind of action; or might there be some useful compromises which combine the better aspects of two or more other choices?

- Would someone else with different experience, goals, or perspectives on the problem be likely to agree with the model and the assessed numerical inputs; if not, what additional considerations might they suggest for modeling, and what other options might they wish to evaluate?

By prompting the user to consider questions such as these, a decision aid could identify the most promising areas for further modeling, while increasing the user's sense of active participation in the analysis and his overall impression of having completed a thorough and useful study of the problem at hand.

5.6 Self-Explanatory Tutorial Guidance and User-Oriented Modes of Interaction

Because most decision aids will be used by a variety of decision makers, representing a wide range of levels of training, experience, and methodological sophistication, it is unrealistic to expect a single interactive routine to satisfy all users. Any routine which is so thorough in its procedures that a novice can use it successfully is likely to prove too slow and boring for anyone with moderate or high experience; similarly, an extremely frequent user may require an especially streamlined version to keep up with the pace of the analysis.

One simple approach which could satisfy most of the problem would be to define three or four "tracks" of analysis, ranging from a highly tutorial, step-by-step method with frequent checks for consistency and accuracy, to a very rapid version using abbreviations, and eliminating most of the error checks to ensure maximum speed. An initial question could establish the user's level of training and experience, and subsequent interaction could permit the user to "escape" to the next simpler level or to speed up by switching to a higher level. Requests for specific assistance on certain subroutines might take the form of a question mark or HELP input, which might generate a brief tutorial with an example, or a summary of the current state of modeling. If such a method were implemented, it should be possible to do away entirely with the printed user's manual which usually accompanies decision aids and which often discourages the user during the training process or disappears at a critical phase of the analysis when it is needed for reference.

Once this multiple-level interactive capability has been developed and tested, a number of extensions might add to its usefulness. For example, if an aid contains a certain amount of pre-stored information (e.g., substantive data, definitions, maps, etc.), it might be useful to allow an expert subject to override (or even to change) the pre-canned model, while still providing it as an aid to the less knowledgeable user. Similarly, when a variety of assessment techniques would be analytically interchangeable, the user's individual preference for one mode of assessment over another might be accommodated (for example, some users might prefer to assess probabilities by estimating numerical odds, while others would prefer to use a graphical aid such as a probability wheel). Without unduly complicating the aid, it should be possible to let a particular user set a number of parameters (using a question-and-answer routine) which, once established, will be remembered and re-established whenever that user identifies himself (until the user wishes to modify them).

While it seems a bit optimistic at present, it is not inconceivable that future decision aids will be able to infer the user's level of sophistication and degree of understanding from the nature and timing of his responses (or from direct questioning), in much the same way that a professional decision analyst does. Ultimately, a sophisticated decision aid should be able to construct and store a fairly detailed "picture" of each individual user, including the user's level of training, his amount of experience using the aid, his preferences for modes of interaction, his areas of substantive expertise, and perhaps a special, personalized vocabulary of user-defined terms to facilitate user-machine communication.

5.7 User Motivation

Probably the most serious objection to those decision aids which exist today (including those documented in this report and in Weiss, 1980) is that their use is perceived by field personnel as intimidating, difficult, confusing, time-consuming, or boring; even when those field users acknowledge the need for decision aids, they are reluctant to use the existing aids unless someone is urging them to do so. The problem does not lie with the decision-aiding approach itself; it is the result of a design strategy which has emphasized technical and analytic correctness over user motivation. An extreme opposite which illustrates the potential for future development is the meteoric rise in popularity of computerized video games, in homes and arcades, as well as the even more complex wargames which abound on university computer systems. Often, the decisions made by players of these games rival or exceed common real-world decisions in complexity, yet the participants will voluntarily spend many hours (and often many dollars) playing.

Although there is a widespread bias within the "serious" research community against projects which appear to be fun, and therefore frivolous, it is well known (and well documented in the literature of cognitive and developmental psychology) that play behavior is often more useful than serious, goal-directed behavior at encouraging creative thought and learning. Particularly, in view of the well-demonstrated analytic soundness of the procedures used in the current generation of decision aids, it seems that the most productive direction for continued decision aid research may be the construction of more game-like devices which would apply decision-analytic techniques in simulated wargame situations, games of strategy--for example, chess, Go, poker, etc.--or other challenging but enjoyable tasks such as puzzle solving. The goal of such an effort should be to develop decision aiding methods which can compete in user-orientation to the commercially available devices, by providing immediate feedback and frequent reinforcement to the user and minimizing the difficult, confusing, and tedious aspects of decision making.

By constructing and experimenting on an attractive, user-oriented decision aid for "fun" problems, researchers could provide valuable insights into the motivational aspects of decision and design, while providing a useful training device for novice users of decision aids, and a vehicle for testing new approaches to decision aiding. Of particular merit would be studies designed to improve our understanding of the timing factors and the role of immediate visual and auditory feedback in motivating users to allocate more time and attention to the analytic tasks.

Once the technology of user motivation has caught up with the currently available technology of decision analysis, it should be possible to conceive of and to design decision aids with far greater capabilities than any available today. Combined with the anticipated trend toward smaller, faster, cheaper, and more plentiful electronic devices, and toward ever more sophisticated graphical and acoustical input and output facilities, these advanced techniques will render the technology of decision analysis accessible and available to an ever-increasing range of problems, and an expanding population of decision makers.

5.8 Final Recommendations for Future Development

Based upon the work DDI has done in researching and developing analytic techniques, implementing decision aids, and testing some routines with real-world decision makers, the following items are recommended for research during the next few years:

- **research, design, and develop sophisticated user-oriented systems which not only provide decision-analytic aid, but also stimulate user motivation through the use of sensory feedback, reinforcement, and naturalistic interaction;**

- investigate methods for incorporating pre-stored substantive information into computer-aided analysis to help in model construction and assessment;

- experiment with various schemes to vary the levels of generality, detail, and flexibility in pre-stored models to accommodate various problem categories, user populations, time constraints, and substantive contexts; and

- design and develop new techniques to aid the user in interpreting, displaying, and explaining the results of an analysis, and to facilitate exploratory and extended analyses.

REFERENCES

Allen, J.J.; Kelly, C.W., III; Phillips, L.H.; and Stewart, R.R. Computer-Assisted Option Screening and Intelligence Assessment: Software Users' Guide. McLean, VA: Decisions and Designs, Inc., October 1976.

Barclay, S., and Randall, L.S. Satellite Surveillance Avoidance Optimization Aid. Final Report PR 77-14-36. McLean, VA: Decisions and Designs, Inc., December 1977.

Bowen, K.C. "Personal and Organizational Value Systems: How should we treat these in OR studies?" Paper presented at the Third International Research Conference on Operational Research and Management Science, organized by the University of Sussex (UK) and held at Bowness-on-Windmere, 3-6 April 1979.

Brown, R.V.; Kahr, A.S.; and Peterson, C.R. Decision Analysis for the Manager. New York, NY: Holt, Rinehart and Winston, 1974.

Buede, D.M. Decision Analysis: Engineering Science or Clinical Art? Technical Report TR 79-2-97. McLean, VA: Decisions and Designs, Inc., November 1979.

Fillmore, C. "The Case for Case." In Bach and Harms (eds.), Universals in Linguistic Theory. New York, NY: Holt, Rinehart and Winston, 1968.

Fischhoff, B. "Clinical Decision Analysis." Operations Research. Vol. 28, No. 1, January-February 1980.

Gulick, R.M., and Allardyce, L.B. Documentation of Decision-Aiding Software: R-SCREEN Users Manual. Users Manual 79-3-99. McLean, VA: Decisions and Designs, Inc., October 1979.

Howard, R.A. "An Assessment of Decision Analysis." Operations Research. Vol. 28, No. 1, January-February 1980.

Humphreys, P.C. "Decision Aids: Aiding Decisions." To appear in L. Sjöberg, T. Tyszka, and J.A. Wise (eds.), Decision Analyses and Decision Processes: Vol. 1 Lund: Doxa, 1979.

Humphreys, P.C., and McFadden, W. Experiences with MAUD: Aiding Decision Structuring Versus Bootstrapping the Decision Maker. Case study using an interactive Multiattribute Utility computer program for aiding decision making in the arts and mass media. Uxbridge, Middlesex, UK: Brunel University, 1980.

Humphreys, P.C., and Wisudha, A. MAUD - An Interactive Computer for the Structuring, Decomposition, and Recomposition of Preference between Multiattributed Atlernatives. Decision Analysis Unit Technical Report 79-2, Uxbridge, Middlesex, UK: Brunel University, 1979.

Keeney, R.L., and Raiffa, H. Decisions with Multiple Objectives: Preferences and Value Tradeoffs. New York, NY: Wiley, 1976.

Kibler, A.W.; Watson, S.R.; Kelly, C.W., III; and Phelps, R.H. A Prototype Aid for Evaluating Alternative Courses of Action for Tactical Engagement. Final Technical Report 78-38A. Alexandria, VA: U.S. Army Research Institute for the Behavioral and Social Sciences, November 1978.

Leal, A. "An Interactive Program for Conversational Elicitation of Decision Structures." Ph.D. dissertation, University of California at Los Angelos, 1976.

Leal, A., and Pearl, J. A Computer System for Conversational Elicitation of Problem Structures. Report UCLA-ENG-REP-7665. Los Angelos, CA: University of California at Los Angelos, June 1976.

Merkhoffer, M.W.; Robinson, B.; and Korsan, R.J. A Computer-Aided Decision Structuring Process. Menlo Park, CA: SRI International, June 1979.

Raiffa, H. Decision Analysis: Introductory Lectures on Choices Under Uncertainty. Reading, MA: Addison-Wesley, 1970.

Schank, R. "Conceptual Dependency: A Theory of Natural Language Understanding." Cognitive Psychology, Vol. 3, No. 4, 1972.

Schank, R. "Identification of Conceptualizations Underlying Natural Language." Chapter in Schank, R., and Colby, K.M. (eds.), Computer Models of Thought and Language. San Francisco, CA: W.H. Freeman and Co., 1973.

Von Winterfeldt, D. "Structuring Decision Problems." Invited paper presented at the Seventh Research Conference on Subjective Probability, Utility, and Human Decision Making, Göteborg, Sweden, August 1979.

Weiss, J.J. QVAL and GenTree: Two Approaches to Problem Structuring in Decision Aids. Technical Report TR 80-3-97. McLean, VA: Decisions and Designs, Inc., April 1980.

CONTRACT DISTRIBUTION LIST
(Unclassified Technical Reports)

Director Advanced Research Projects Agency Attention: Program Management Office 1400 Wilson Boulevard Arlington, Virginia 22209	2 copies
Office of Naval Research Attention: Code 455 800 North Quincy Street Arlington, Virginia 22217	3 copies
Defense Technical Information Center Attention: DDC-TC Cameron Station Alexandria, Virginia 22314	12 copies
DCASMA Baltimore Office Attention: Mrs. Betty L. Driskill 300 East Joppa Road Towson, Maryland 21204	1 copy
Director Naval Research Laboratory Attention: Code 2627 Washington, D.C. 20375	6 copies
Office of Naval Research Boston Branch Office 666 Summer Street Boston, Massachusetts 02210	1 copy

APRIL 1980

SUPPLEMENTAL DISTRIBUTION LIST
(Unclassified Technical Reports)

Department of Defense

Assistant Director (Net Technical Assessment)
Office of the Deputy Director of Defense
Research and Engineering (Test and
Evaluation)
The Pentagon, Room 3C125
Washington, DC 20301

Director, Defense Advanced Research
Projects Agency
1400 Wilson Boulevard
Arlington, VA 22209

Director, Cybernetics Technology Office
Defense Advanced Research Projects Agency
1400 Wilson Boulevard
Arlington, VA 22209

Director, ARPA Regional Office (Europe)
Headquarters, U.S. European Command
APO New York 09128

Director, ARPA Regional Office (Pacific)
Staff CINCPAC, Box 13
Camp H. M. Smith, Hawaii 96861

Dr. Don Hirta
Defense Systems Management School
Building 202
Ft. Belvoir, VA 22060

Chairman, Department of Curriculum
Development
National War College
Ft. McNair, 4th and P Streets, SW
Washington, DC 20319

Defense Intelligence School
Attention: Professor Douglas E. Hunter
Washington, DC 20374

Vice Director for Production
Management Office (Special Actions)
Defense Intelligence Agency
Room 1E863, The Pentagon
Washington, DC 20301

Command and Control Technical Center
Defense Communications Agency
Attention: Mr. John D. Hwang
Washington, DC 20301

Department of the Navy

Office of the Chief of Naval Operations
(OP-951)
Washington, DC 20450

Office of Naval Research
Assistant Chief for Technology (Code 200)
800 N. Quincy Street
Arlington, VA 22217

Office of Naval Research (Code 230)
800 North Quincy Street
Arlington, VA 22217

Office of Naval Research
Naval Analysis Programs (Code 431)
800 North Quincy Street
Arlington, VA 22217

Office of Naval Research
Operations Research Programs (Code 434)
800 North Quincy Street
Arlington, VA 22217

Office of Naval Research
Information Systems Program (Code 437)
800 North Quincy Street
Arlington, VA 22217

Director, ONR Branch Office
Attention: Dr. Charles Davis
536 South Clark Street
Chicago, IL 60605

Director, ONR Branch Office
Attention: Dr. J. Lester
495 Summer Street
Boston, MA 02210

Director, ONR Branch Office
Attention: Dr. E. Gloye
1030 East Green Street
Pasadena, CA 91106

Director, ONR Branch Office
Attention: Mr. R. Lawson
1030 East Green Street
Pasadena, CA 91106

Office of Naval Research
Scientific Liaison Group
Attention: Dr. M. Bertin
American Embassy - Room A-407
APO San Francisco 96503

Dr. A. L. Slafkosky
Scientific Advisor
Commandant of the Marine Corps (Code RD-1)
Washington, DC 20380

Headquarters, Naval Material Command
(Code 0331)
Attention: Dr. Heber G. Moore
Washington, DC 20360

Dean of Research Administration
Naval Postgraduate School
Attention: Patrick C. Parker
Monterey, CA 93940

Superintendent
Naval Postgraduate School
Attention: R. J. Roland, (Code 52R1)
C^3 Curriculum
Monterey, CA 93940

Naval Personnel Research and Development
Center (Code 305)
Attention: LCDR O'Bar
San Diego, CA 92152

Navy Personnel Research and Development
Center
Manned Systems Design (Code 311)
Attention: Dr. Fred Muckler
San Diego, CA 92152

Naval Training Equipment Center
Human Factors Department (Code N215)
Orlando, FL 32813

Naval Training Equipment Center
Training Analysis and Evaluation Group
(Code N-00T)
Attention: Dr. Alfred F. Smode
Orlando, FL 32813

Director, Center for Advanced Research
Naval War College
Attention: Professor C. Lewis
Newport, RI 02840

Naval Research Laboratory
Communications Sciences Division (Code 5403)
Attention: Dr. John Shore
Washington, DC 20375

Dean of the Academic Departments
U.S. Naval Academy
Annapolis, MD 21402

Dr. Glen R. Algaier
Head
Command & Control
Information Processing Branch
Naval Ocean Systems Center
San Diego, CA 92152

Chief, Intelligence Division
Marine Corps Development Center
Quantico, VA 22134

Department of the Army

Deputy Under Secretary of the Army
(Operations Research)
The Pentagon, Room 2E621
Washington, DC 20310

Director, Army Library
Army Studies (ASDIRS)
The Pentagon, Room 1A534
Washington, DC 20310

U.S. Army Research Institute
Organizations and Systems Research Laborator
Attention: Dr. Edgar M. Johnson
5001 Eisenhower Avenue
Alexandria, VA 22333

Director, Organizations and Systems
Research Laboratory
U.S. Army Institute for the Behavioral
and Social Sciences
5001 Eisenhower Avenue
Alexandria, VA 22333

Technical Director, U.S. Army Concepts
Analysis Agency
8120 Woodmont Avenue
Bethesda, MD 20014

Director, Strategic Studies Institute
U.S. Army Combat Developments Command
Carlisle Barracks, PA 17013

Commandant, Army Logistics Management Center
Attention: DRXMC-LS-SCAD (ORSA)
Ft. Lee, VA 23801

Department of Engineering
United States Military Academy
Attention: COL A. F. Grum
West Point, NY 10996

Marine Corps Representative
U.S. Army War College
Carlisle Barracks, PA 17013

Chief, Studies and Analysis Office
Headquarters, Army Training and Doctrine Command
Ft. Monroe, VA 23351

Commander, U.S. Army Research Office (Durham)
Box CM, Duke Station
Durham, NC 27706

Department of the Air Force

Assistant for Requirements Development and Acquisition Programs
Office of the Deputy Chief of Staff for Research and Development
The Pentagon, Room 4C331
Washington, DC 20330

Air Force Office of Scientific Research
Life Sciences Directorate
Building 410, Bolling AFB
Washington, DC 20332

Commandant, Air University
Maxwell AFB, AL 36112

Chief, Systems Effectiveness Branch
Human Engineering Division
Attention: Dr. Donald A. Topmiller
Wright-Patterson AFB, OH 45433

Deputy Chief of Staff, Plans, and Operations
Directorate of Concepts (AR/XOCCC)
Attention: Major R. Linhard
The Pentagon, Room 4D 1047
Washington, DC 20330

Director, Advanced Systems Division (AFHRL/AS)
Attention: Dr. Gordon Eckstrand
Wright-Patterson AFB, OH 45433

Commander, Rome Air Development Center
Attention: Mr. John Atkinson
Griffis AFB
Rome, NY 13440

IRD, Rome Air Development Center
Attention: Mr. Frederic A. Dion
Griffis AFB
Rome, NY 13440

HQS Tactical Air Command
Attention: LTCOL David Dianich
Langley AFB, VA 23665

Colonel Margenthaler
AFIT/EN
Wright-Patterson AFB, OH 45433

LTCOL Robert G. Gough
AFTEC/DAY
Curtland AFB, NM 87117

Other Government Agencies

Chief, Strategic Evaluation Center
Central Intelligence Agency
Headquarters, Room 2G24
Washington, DC 20505

Director, Center for the Study of Intelligence
Central Intelligence Agency
Attention: Mr. Dean Moor
Washington, DC 20505

Mr. Richard Heuer
27585 VIA Sereno
Carmel, CA 93923

Office of Life Sciences
Headquarters, National Aeronautics and Space Administration
Attention: Dr. Stanley Deutsch
600 Independence Avenue
Washington, DC 20546

Other Institutions

Department of Psychology
The Johns Hopkins University
Attention: Dr. Alphonse Chapanis
Charles and 34th Streets
Baltimore, MD 21218

Institute for Defense Analyses
Attention: Dr. Jesse Orlansky
400 Army Navy Drive
Arlington, VA 22202

Director, Social Science Research Institute
University of Southern California
Attention: Dr. Ward Edwards
Los Angeles, CA 90007

Perceptronics, Incorporated
Attention: Dr. Amos Freedy
6271 Variel Avenue
Woodland Hills, CA 91364

Stanford University
Attention: Dr. R. A. Howard
Stanford, CA 94305

Director, Applied Psychology Unit
Medical Research Council
Attention: Dr. A. D. Baddeley
15 Chaucer Road
Cambridge, CB 2EF
England

Department of Psychology
Brunel University
Attention: Dr. Lawrence D. Phillips
Uxbridge, Middlesex UB8 3PH
England

Decision Analysis Group
Stanford Research Institute
Attention: Dr. Miley W. Merkhofer
Menlo Park, CA 94025

Decision Research
1201 Oak Street
Eugene, OR 97401

Department of Psychology
University of Washington
Attention: Dr. Lee Roy Beach
Seattle, WA 98195

Department of Electrical and Computer Engineering
University of Michigan
Attention: Professor Kan Chen
Ann Arbor, MI 94135

Department of Government and Politics
University of Maryland
Attention: Dr. Davis B. Bobrow
College Park, MD 20747

Department of Psychology
Hebrew University
Attention: Dr. Amos Tversky
Jerusalem, Israel

Dr. Andrew P. Sage
School of Engineering and Applied Science
University of Virginia
Charlottesville, VA 22901

Professor Raymond Tanter
Political Science Department
The University of Michigan
Ann Arbor, MI 48109

Professor Howard Raiffa
Morgan 302
Harvard Business School
Harvard University
Cambridge, MA 02163

Department of Psychology
University of Oklahoma
Attention: Dr. Charles Gettys
455 West Lindsey
Dale Hall Tower
Norman, OK 73069

Institute of Behavioral Science #3
University of Colorado
Attention: Dr. Kenneth Hammond
Room 201
Boulder, Colorado 80309