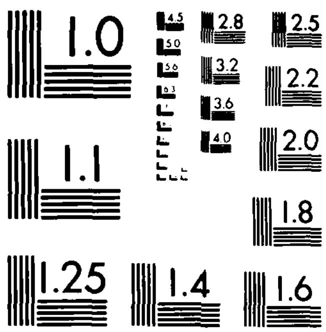
AD-A175 263 NON-COOPERATIVE GROUP DECISION SUPPORT SYSTEMS: PROBLEMS AND SOME SOLUTIONS(U) NAVAL POSTGRADUATE SCHOOL MONTEREY CA A KARDOS ET AL. SEP 86 1/2

UNCLASSIFIED F/G 5/1 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963 A

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

NON-COOPERATIVE GROUP DECISION SUPPORT SYSTEMS:
PROBLEMS AND SOME SOLUTIONS

by

Andre Kardos
Egbert Kutz

September 1986

Thesis Advisor: Tung Bui

ADA 175263

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION: Unclassified

1b. RESTRICTIVE MARKINGS

2a SECURITY CLASSIFICATION AUTHORITY

2b DECLASSIFICATION / DOWNGRADING SCHEDULE

3 DISTRIBUTION / AVAILABILITY OF REPORT: Approved for public release; distribution is unlimited.

4 PERFORMING ORGANIZATION REPORT NUMBER(S)

5 MONITORING ORGANIZATION REPORT NUMBER(S)

6a NAME OF PERFORMING ORGANIZATION: Naval Postgraduate School

6b OFFICE SYMBOL (If applicable): Code 54

7a NAME OF MONITORING ORGANIZATION: Naval Postgraduate School

6c ADDRESS (City, State, and ZIP Code): Monterey, California 93943-5000

7b ADDRESS (City, State, and ZIP Code): Monterey, California 93943-5000

8a NAME OF FUNDING/SPONSORING ORGANIZATION

8b. OFFICE SYMBOL (If applicable)

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER

8c ADDRESS (City, State, and ZIP Code)

10 SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO	PROJECT NO	TASK NO	WORK UNIT ACCESSION NO

11 TITLE (Include Security Classification)

NON-COOPERATIVE GROUP DECISION SUPPORT SYSTEMS: PROBLEMS AND SOME SOLUTIONS

12 PERSONAL AUTHOR(S): Kardos, Andre and Kutz, Egbert

13a TYPE OF REPORT: Master's Thesis

13b TIME COVERED FROM ______ TO ______

14 DATE OF REPORT (Year, Month, Day): 1986 September

15 PAGE COUNT: 120

16 SUPPLEMENTARY NOTATION

17 COSATI CODES

FIELD	GROUP	SUB-GROUP

18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)

Group Decision Support System, Non-cooperation, Negotiation, Negotiable Alternatives Identifier

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

The purpose of this thesis is twofold; (1) Explore some design issues for building group decision support systems for non-cooperation environments, and (2) Expand CO-OP, a cooperative multiple criteria group decision support system, to support particular classes of group decisions. From the conceptual standpoint, this work argues for that cooperation is a special case of non-cooperation. The following design requirements are proposed: (1) Negotiation as a capability within model management, (2) Greater capabilities in database management, and (3) Increased flexibility for the user interface.

The present version of CO-OP has, with this work, implemented the following features: (1) Scrolling windows to handle group problems with large size, (2) Code optimization to provide fast feedback to members, (3) Improved heuristics for the Negotiable Alternatives

20 DISTRIBUTION / AVAILABILITY OF ABSTRACT

☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS

21 ABSTRACT SECURITY CLASSIFICATION: Unclassified

22a NAME OF RESPONSIBLE INDIVIDUAL: Tung Bui

22b TELEPHONE (Include Area Code): 408-646-2630

22c OFFICE SYMBOL: 54Bd

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE

Identifier (NAI), (4) Implementation of the Mediator module, and (5) Allow more advanced data manipulation to promote data exchange in competitive environments (e.g., data security and sharing). The above implementation has encompassed approximately 6,000 lines of original pascal code, and 3,000 lines of modified code.

Approved for public release; distribution is unlimited

Non-cooperative Group Decision Support Systems: Problems and Some Solutions

by

Andre Kardos
Lieutenant Commander, United States Navy
B.S., Florida State University, 1973

and

Egbert Kutz
Lieutenant, Federal Republic of Germany, Navy
Ing. (grad.), TFH Berlin, 1976

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN INFORMATION SYSTEMS

from the

NAVAL POSTGRADUATE SCHOOL
September 1986

Authors: Andre Kardos

Egbert Kutz

Approved by: Tung X. Bui, Thesis Advisor

Carson K. Eoyang, Second Reader

Willis R. Greer Jr., Chairman,
Department of Administrative Sciences

Kneale T. Marshall,
Dean of Information and Policy Sciences

ABSTRACT

The purpose of this thesis is twofold; (i) Explore some design issues for building group decision support systems for non-cooperation environments, and (ii) Expand CO-OP, a cooperative multiple criteria group decision support system, to support particular classes of group decisions. From the conceptual standpoint, this work argues for that cooperation is a special case of non-cooperation. The following design requirements are proposed: (i) Negotiation as a capability within model management, (ii) Greater capabilities in database management, and (iii) Increased flexibility for the user interface.

The present version of Co-oP has, with this work, implemented the following features: (i) Scrolling windows to handle group problems with large size, (ii) Code optimization to provide fast feedback to members, (iii) Improved heuristics for the Negotiable Alternatives Identifier (NAI), (iv) Implementation of the Mediator module, and (v) Allow more advanced data manipulation to promote data exchange in competitive environments (e.g., data security and sharing). The above implementation has encompassed approximately 6,000 lines of original pascal code, and 3,000 lines of modified code.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

The authors of this work would like to express their appreciation to Professor Tung X. Bui for the professional guidance and educational insights that he provided.

Additionally, the authors wish to thank their wives, Jutta and Ivy, for the confidence they had in us and the support they provided during this educational experience.

I. INTRODUCTION

The importance of decision making is readily apparent throughout the daily activities of individuals or groups, often reflecting the goals, tasks, and choices selected during problem solution. Numerous concepts and theories about individual decision making endeavor to isolate the decision making process into a normative set of rules or a descriptive set of procedures that may be utilized by the decision maker (Bass, 1983). However, the majority of really crucial events of the world are a direct result of the group decision-making process rather than isolated, individual decision making.

Research on decision support systems (DSS) has recently focused its attention on supporting collective decision making (Huber, 1984). However, all of the contributions so far have dealt with decision making situations characterized by trusting and cooperative settings. Bass (1983) argues that the lack of sufficient cooperation and coordination is often a determining factor that prevents the group decision making process from providing valid results. Attempting to design a computer based decision support system for non-cooperative decision making situations requires careful redesign of the requirements and functions that are currently utilized in many cooperative individual and group decision support systems (GDSS).

The purpose of this thesis is to explore the issues of non-cooperation and implements some of the principles in Co-oP, a group decision support system for cooperative decision making. The thesis is organized as follows. Chapter 2 discusses and reviews various concepts and motivating forces behind decision-making in general, and ultimately applies these perceptions to group decision-making. This chapter also discusses the nature and causes of conflict in the collective decision environment. Chapter 3 explains various processes or methodologies utilized by associations of individuals to resolve conflict within a non-cooperative setting. Chapter 4 introduces many of the basic concepts that deal with individual computerized decision support systems (IDSS); the latter being designed as a integral part of the GDSS. DSS design concepts, including data base management, model management, and dialogue management, are briefly reviewed to provide the reader with a cursory understanding of decision support systems, prior to the discussion of group decision support systems (GDSS) that follows in the remaining chapters. Chapter 5 covers earlier examples of implemented computer-based group decision support system (GDSS). Design issues in implementing GDSS for non-cooperation are discussed in Chapter 6. The requirement analysis in this chapter concentrates on digital communications and multi-user database management. Chapter 7 applies some of the suggestions outlined in the previous chapter to expand a specific GDSS, namely Co-oP. The rationale for using Co-oP as a basic system architecture is founded on the assumption that designing a GDSS for cooperation is a special case of GDSS for non-cooperation. Therefore, it would make sense, at least from a system design point of view, to implement and validate the ideas advocated in this work by expanding the features of a currently operational cooperative GDSS. In this regard, Chapter 8 proposes a new set of heuristics to support negotiation, providing that the decision makers accept the precepts of the negotiable alternative identifier (N.A.I) algorithm in Co-oP (Bui, 1985). Finally, reflections and cautions regarding the use of a GDSS to support competitive decision problems are detailed in Chapter 9.

In summary, the material presented in this thesis should provide the reader with the major conceptual building blocks necessary to conduct initial and, possibly, continued research into the arena of group decision support systems. Observed as a logical progression of knowledge and implementation, a GDSS is based upon previous qualitative decision theory and quantitative computer algorithms. Figure 1 illustrates these building blocks.

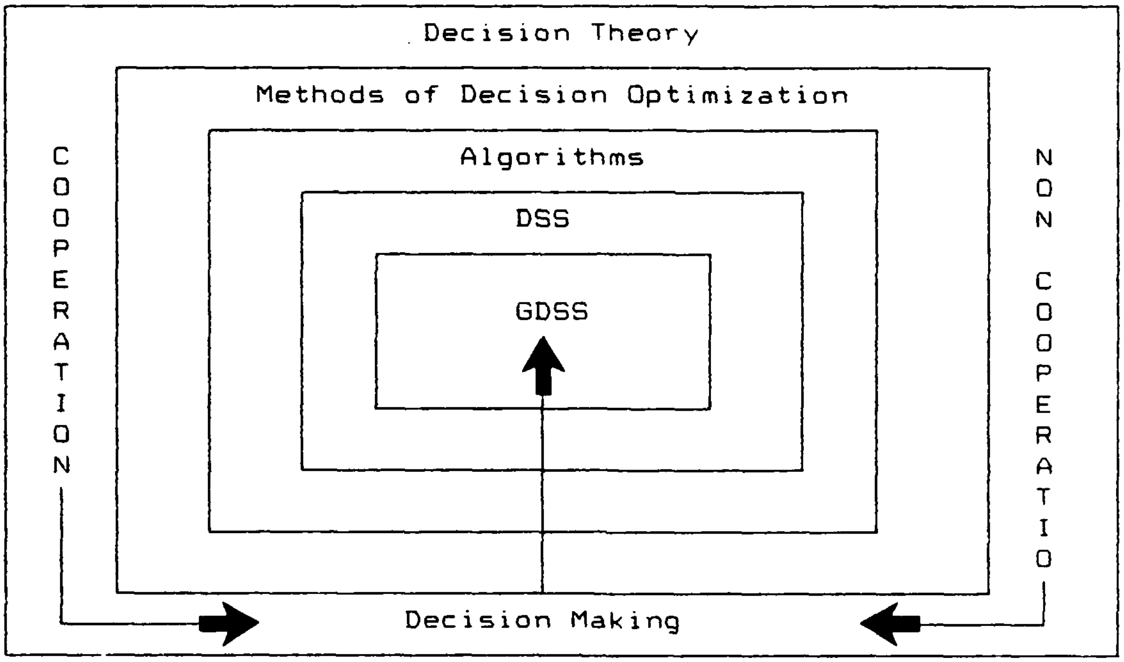


Figure 1. **Conceptual Building Blocks for a GDSS**

II. NON-COOPERATIVE ISSUES IN COLLECTIVE DECISION MAKING

The body of knowledge concerning decision making is great, portraying a large spectrum of issues. This chapter considers primarily that portion of decision theory that deals with collective decision making and non-cooperation. In order to define "non-cooperation", a review of cooperative decision making is conducted. The primary purpose for this discussion is to identify some common areas of concern for decision makers. Once these limitations are known, possible solutions may be attempted through conversion into computer-based algorithms. These algorithms might be thought of as based in the many-faceted decision-theory environment.

A. COLLECTIVE DECISION MAKING

A **collective decision-making** process is characterized by the following traits; (i) There are two or more individuals involved, each with their own explicit personalities, (ii) each player recognizes the existence of a unique and common problem, and (iii) the group will attempt to reach a collective decision (Bui, 1984). A group may attempt to reach consensus during simultaneous discussions, or they might separately reach conclusions, and then regroup to collectively challenge and discuss the results.

A unique type of collective decision making often encountered is one in which there is effectively only one decision maker. In **group decision-making with one person**, the decision is made by a specific individual who assumes full responsibility for the outcome (Bui, 1985). Since this accountable individual maintains a strong supporting infrastructure, the decision is regarded as collective. Swap (1984) defines this entity as the "responsible individual." This decision maker is the person in the organization assigned decision responsibility and authority in a certain functional area. This singular entity may not always be the apparent decision maker, however, the actions of the group will often reveal whether the designated responsible individual has authority and power to influence others. Authority may be defined as "the power to make decisions which guide the behavior of subordinates" (Simon, 1961).

There appears to be numerous factors that determine the strength and responsiveness of the supporting infrastructure or "dense network of influences" provided this responsible individual within the group decision making arena; (i) his or her attitude on the administrative hierarchy, (ii) the line or staff character of the responsible individual's position, (iii) the manner in which the responsible individual obtained the position, (iv) the role the individual has been asked to assume (or has assumed for him or herself) in the group or organization, (v) the personality and management style of the individual which is important to the relationship with the decision group, and, (vi) the degree of support and loyalty that the individual can command among the members of the group. These considerations strongly affect the quality and strength of the leadership possessed by the leader in a "leader-member" group with cohesiveness. However, this particular type of "collective decision making" is unilateral in nature. For this thesis, we focus more on problems that involve a multilateral form of interpersonal relationship or problem solving.

B. COOPERATIVE VERSUS NON-COOPERATIVE DECISION MAKING

In a cooperative decision-making situation, the decision makers attempt to reach a common decision in a friendly and trusting manner, and share the responsibility for that decision. Consensus, compromise, negotiation, and voting stratagem are instances of this type of group decision making.

According to Fischer (1980), the term **consensus** can bear different meanings. Many individuals view consensus as the desire of the majority which results from democratic processes. Others presume that unanimity is required for consensus. Still others believe in a pursuit of common goal, which often does not call for a formal vote but results in an implicit agreement by the group members. The reaching of consensus within groups means that members may agree with a decision, even unanimously, but not reach the final goal. The distinguishing point is highlighted by Zaleznik and Moment (1964, p. 129):

> Our meaning of consensus lies in the degree of personal commitment the members feel toward the group decision after it is reached. This means, for example, that even though some members might disagree with the decision on principle, they will accept it and personally carry out their part. Their emotional commitment to the group is measured by willingness to put the plan decided on into effect, in their own personal behavior.

A **compromise** is a solution, or at least a settlement of differences, whereby each side makes concessions (Zeleny, 1982). A consensus is a collective opinion or agreement. There can be many compromise solutions, but only one consensus. A group can define different compromise solutions; one of them will emerge as a consensus.

The fact that a decision is made and carried out to its ultimate conclusion would seem to lead one to believe that the group members reached consensus and were highly committed to achievement of the final goal. At least from the viewpoint of the public, and possibly the decision-makers themselves, the decision was made in a cooperative setting.

In many cases, the consequence of a decision belies its very nature, in that what was initially thought of as a cooperative environment, actually upon closer examination turned out to be a non-cooperative decision situation with uncertain results. It is the purpose of this chapter to look more deeply into the various aspects of decision making in a non-cooperative atmosphere.

In the **non-cooperative decision situation**, the decision makers play the role of adversaries or disputants. Common forms of non-cooperative decision making often originate as conflict and competition. While the former represents a situation in which disputants seek to impair their opponents to pursue their own interest, the latter is characterized by the fact that each competitor actively attempts to outperform each other.

As decision makers struggle for a mutually acceptable option, differences among them in perceptions, cognition, values, interests, needs, and preferred alternatives give rise to conflicts (Bui, 1986). Conflict is usually most evident in elaborate organizations with highly differentiated structures and operating in an unpredictable environment. Tasks are often highly complex within this framework. The differences in the needs and interests of each individual member and the organization as a whole may generate conflict. It is also generated by differences between organizational entities such as departments. Higher authority and great power may be engaged to resolve this conflict, or the conflict may be settled adaptively in a collective manner utilizing negotiation and bargaining. Mediators and arbitrators may be employed. A mutually advantageous solution, one that is agreeable to all parties and nearly optimal, may be gained through the integration of conflicting interests rather than merely attempting to compromise one side over the other (Bass, 1983).

Coalitions are alliances of organization members combining their individual powers, resources, and persuasive efforts to achieve greater influence on decision processes than the members could accomplish alone. Coalitions are commonly observed when conflicting interests are present in a group or organization. To increase one's negotiating power, an individual may join forces with others in the larger group in order to establish informal cliques and coalitions that will exercise influence over decisions made. Kahan and Rapaport (1984) relate that whenever three or more parties get together to jointly resolve an issue of substantive interest to all of them, it is likely that at least two of them will at some point in time combine forces to their mutual advantage. When this combining of forces is intentional, or executed with the full awareness of all joining parties, a coalition is being formed.

C. CONFLICT AS CAUSES OF NON-COOPERATIVE DECISION MAKING

Conflict often occurs in both individual and collective decision-making processes. Zeleny (1982) defines conflict as when multiple distinct strategies, selected as the means of achieving goals or objectives, are mutually exclusive. This occurs when the strategies become mutually exclusive alternatives, each capable of satisfying only a portion or a particular aspect of a given goal complex. The following are neccessary conditions of a conflicting situation:

(1) One or more decision makers.

(2) Two or more available alternatives of choice.

(3) One or more objectives or criteria of choice.

With respect to the above conditions, conflict exists whenever interaction by multiple (two or more) decision makers uniquely affects their respective environments, and the nature of their interaction is such that it is not possible for all these individuals to simultaneously achieve their desired goals (Kahan and Rapaport, 1984).

The organizational decision process is one source of conflict that is inherent in group decision making since it must meet what may be incompatible multiple criteria of acceptance. The personal interests of the decision makers usually need to be satisfied. Finally, the decision needs to be accepted by those responsible for authorizing and implementing it (Bass, 1983).

A second source of conflict is **due to the way information flows** in the organization or group environment. Instead of an orderly progression through the established hierarchy, it follows a grid of communications made up of conflicting channels (Bass, 1983).

Third, **disagreement about means or ends** can lie between the multiple relationships established within groups or organizations. Organizations find it difficult to tolerate the enterprizing manager and expect only orderly advances. Not forgiving of surprises, corporate management often fail to reward this kind of risk taking and overemphasize the obtaining of immediate or short-term goals, at the expense of distant, future ones (Bass, 1983).

A fourth, and important source of conflict arises in the **allocation of resources.** When corporate leaders attempt to allocate resources optimally, they, in actuality, only approach the ideal. The allocation process often create turmoil or conflict within corporate ranks since the attempt at optimality may reduce perceived "slack" throughout the organization. Therefore, this reduction may not be not seen as beneficial by individual groups, with this slack regarded as necessary buffers against complex timetables (Bass, 1983).

A final source of conflict is **change** itself. On many occasions new but extremely different alternatives to entrenched policies are seen by organizations as dangerous and not considered in any methodical and distinct way (Bass, 1983).

D. GENERAL APPROACHES TO DEAL WITH CONFLICTS

Zeleny (1982) delineates neglect, containment, control, and denial, as four ways of dealing with conflict. However, although the last four are often used, they are complex and hard to reduce into formal terms or study in a structured methodology. A decision maker may disregard, neglect, or ignore a conflict or one can also attempt to contain conflict or "freeze" it to gain time and let things cool off. Since a conflict is usually not a well-defined, unambiguous state of affairs, it is often more convenient to contain it. An individual may attempt to control conflict by adding constraints that limit the results or outcomes. One form of controlled conflict is competition, since it is in reality conflict constrained by pre-determined rules. Another method is to deny the existence of conflict, therby acknowledging only the existance of a certain situation, but then advancing a different, often very imaginative translation in conflict-free terminology. Therefore, conflict denial is often observed when an organization or individual freely uses persuasion, propaganda, and brainwashing.

Ackoff (1978) describes three more methods for dealing with conflict: solution, resolution, and dissolution. The resolution of conflict is the subject of numerous unilateral and multilateral methodologies that are more flexible and attempt to reach the ideal maximization of outcomes for both parties in a conflict. The benefits to a singular entity or parties is often accomplished through mediation. The task of the **mediator** in a group decision-making process can be either; (i) seek to impartially resolve a dispute or (ii) be a judge who wants to end a conflict (Bui, 1985).

In order to resolve a conflict the individual or group should be aware of possible multiple objectives that take into account the other individuals involved. In the normal sense, conflict resolution seeks to obtain a compromise, a settlement, or a consensus. In most cases, as discussed above, the "fair" and equitable conclusion may be reached by negotiation, bargaining, and arbitration. (Zeleny, 1982)

The dissolution of conflict is very often attempted by decision makers, however, this complete removal of conflict is seldom accomplished, the result of which may result in further conflict.

Solving a conflict is characterized by a single individual or group's single objective, and its maximization or optimization is the sole criterion for action (Zeleny, 1982). One may prescriptively accept the factors that induced the conflict, only to do whatever is necessary to obtain a the best outcome one can. For example, a decision maker might try to solve a strike, by outwardly accepting it, and then closing the plant down.

E. SUMMARY

The major point of this chapter was is not to provide an all-encompassing discource concerning decision-making theory (i.e., alternative selection, conflict and its' resolution, etc.), but to indicate to the prospective developer of an individual or group decision support system the importance of this material within DSS design constraints. To relegate decision theory to the peripheral of DSS or GDSS development is akin to building a home without its foundation. One must first recognize that an organization or group of decision makers' utilize a specific method for problem solving. In other words, a DSS built for a chief Executive Officer (CEO) used to making unilateral decisions will differ greatly then one built for a group used to resolving problems through mediation.

Secondly, one must realize that a decision support system must be developed in stages. This is correctly performed with Co-oP, a group decision support system (GDSS), which is discussed in later chapters of this thesis. First, the premise is made (and implemented) that the GDSS will utilize a group resolution decision-theoretic approach to problem solving. Secondly, the cooperative decision-making approach is implemented to prove that the system works. The next stage in development is to develop a module for non-cooperative decision making. Concurrent to this accomplishment, a mediatior module is developed to define rules of interaction within the group decision process. Finally, other modules reflecting numerous other decision making approaches may be implemented as development continues.

III. RESOLUTION METHODOLOGIES FOR NON-COOPERATIVE SETTINGS

This chapter moves the discussion of group decision making one step further, and represents a limited view of various frameworks within which conflict is resolved. Within these structures a conflict may be "resolved" utilizing either non-computer-based or computer-based procedures. The major point to understand is that, depending on ones' view of the decision making process, and acceptance of possible group problem resolution, the method chosen to accomplish this process may have a great effect on the ultimate outcome.

Whatever technique is used; (i) aggregation of preferences, (ii) process-oriented tools, (iii) bootstrapping methods, (iv) multiple criteria decision making (MCDM) methods, or (v) a combination, the final framework or model (in the case of a DSS or GDSS) has to reflect the inherent capabilities and limitations within the system.

A. AGGREGATION OF PREFERENCES

There are two types of aggregation of opinions or preferences. The first is mathematical and the second is behavioral. Because the mathematical technique of aggregation of preferences is relatively easy to use and apparently simple upon interpretation, it is often used over the behavioral method within the group decision making environment. (Wright, 1985)

Although it would appear to the uninitiated user that there is a one-to-one, or linear relationship between an individuals' preference and the agreggation of many in a group preference output, techniques of aggregation of preferences have complexities and anomolies that tend to cloud the picture more than would be thought during initial use. The most common problem is, that given a group of individuals utilizing decision matrices, each individual will probably prefer a different alternative for the same set of criteria on the basis of expected utility. After all individuals have selected their preferred alternatives, an aggregation of choices would be conducted to seek a decision or choice for the entire group. However, different algorithms or methods of aggregation will have different results and lead to conflict. If a group decision matrix is formed by averaging the probabilities and summing the individual utilities to obtain group utility, one outcome is preferred on the basis of expected utility.

If the members vote on pairs of actions or sum of the ranks of their individual preferences based on their own expected utility, then a different outcome ensues.

The differences in outcomes due to the use of various aggregation techniques become explicit and numerous methods may be used in a single DSS to obtain a more uniform group decision.

However, inconsistencies may occur even when there is group agreement on the final outcome because of the following reasons:

(1) Point estimates of unknown quantities are often thought to be linear or fixed in nature rather than probabilistic distributions.

(2) The quality of a group decision may depend on the group size for groups of different expertise and independence.

(3) Biased judgments may appear to be averaged and of higher quality if aggregated over a large group. However, although the average can be more accurate then the best member some of the time, it cannot be so on the average. Therefore a simple weighted average will fall between the group consensus and the best member.

(4) In correlated judgments, the individual preferences are not independent in the statistical sense.

B. PROCESS-ORIENTED TOOLS

The **process-oriented** approach is based on the view that if one understands the decision process, then one can correctly estimate the outcome. Primarily descriptive in nature, this process rests on the principle that knowledge of how decisions are made can instruct us how they should be made (Zeleny, 1982). Three process-oriented approaches to group decision found in the literature (Van de Ven, 1974) are; (i) the **interacting approach**, (ii) the **nominal group process**, and (ii) the **Delphi process**.

The first and most widely used approach is the **interacting group** method, in which collective decision making occurs within a group setting and all communication acts take place between members with minimal restraints imposed by formal configuration of structure (Delbecq, 1968). The resultant decision is reached after a process of (i) unstructured group discussion for gaining and merging ideas of participants, and (ii) majority voting on priorities by hand count.

The **nominal group technique** is a structured group meeting in which decision makers perform in the proximity of others but do not interact in an explicit or verbal manner with other group members for a specified period of time. Each individual is tasked with the writing of ideas on a physical or electronic note-pad during this ensuing period. At the completion of this interval each individual in round-robin fashion contributes one idea from his or her tabulation to be documented by a recorder. The round-robin is in effect until no further ideas are presented, and then a spontaneous discussion occurs among the group. As a final step, voting by all the members is conducted, with the group decision being the aggregated or pooled outcome of the individual votes. This method of group decision making is recapitulated in the following order (Van de Ven, 1974): (i) Silent generation of ideas in writing; (ii) Recorded round-robin feedback from each member for presentation of ideas to the group; (iii) discussion of recorded ideas to evaluate information; and (iv) silent individual voting on priorities.

Participants in the **Delphi process** are physically separated and do not meet as a group for decision-making. This procedure is one way of seeking and finally aggregating group judgments on a particular issue through a set of carefully designed questionnaires. To conduct the Delphi process, at least two separate groups of individuals and at least four roles or functions for individual groups are required. There is a **user body** in which the individuals are expecting a product from the exercise which is useful to their purposes. A **design and monitor team**, which may be separate groups, designs the initial questionnaire, summarizes the returns, and re-designs the subsequent questionnaires. The **respondent group** is chosen to respond to the questionnaires and may sometimes be the user group or a subset of the respondent group.

The sequence of decision making in the Delphi process occurs in the following order : (i) One group responds to the first questionaire with independent generation of information; (ii) A synopsis and feedback of the replies to the first questionnaire by the design and monitoring team; (iii) Providing a response to the second questionaire through detached voting on ideas by a rank order procedure; and (iv) Final aggregation and feedback to the respondent group of concluding priorities by the design and monitoring team. The qualitative differences between nominal, interacting, and Delphi processes are described by Van de Ven (1974) in Table 1. These differences are based upon analysis of evaluations of leaders and group participants of various organizations.

C. BOOTSTRAPPING AIDS

Bootstrapping aids serve to display and automate policy or rules, which then put into effect normatively delineated principles already generated through advice from experts (Wright, 1985). Bootstrapping allows the appraisal of the structure within which the problem is assessed. From this basis, the process of the decision-making operations are predefined. The general idea of bootstrapping rests on the view that if a computer-designed decision aid can be developed that captures the interpretive powers and judgmental principles of an expert, then its performance will be at least as good as or better than, the expert's unaided evaluation. An interesting outcome of this aid is that the expert's process will be protected, or "frozen" against the bias of such changing variables as stress and boredom. Additionally, variables of this nature may not be included within the confines of the established model.

Linear statistical models are usually the basis for bootstrapping methods, and may useful when the same predictive evaluations have to be performed on a repetitive basis. Camerer (1981), in conducting a survey of bootstrapping methods, concluded that "bootstrapping will improve judgments slightly under almost any realistic task condition."

The ability of a bootstrapping aid to be effectively predictive is determined by a linear relationship between the predictor variables within the model and the external criteria (Dawes and Corrigan, 1974).

Two limitations to the overall use of bootstrapping implementations are; (i) The variables within the system do not alter with a change in expert (Hoffman, 1960), and (ii) The overall model remains valid, or the representative decision process remains the same. As soon as additional variables become applicable, the model, and its use will lead to diminished, or even incorrect results. Since the primary aspect of the system is to represent to the maximum the expert knowledge strategy, it does not have to mirror the exact cognitive processes involved in order to consider its output satisfactory.

D. DECOMPOSITION AIDS

One of the roles of a decision aid is to assist the decision maker or group in symbolizing their problems within the limitations of a formalized decision structure. Once the problem structure has been defined, computer-designed recomposition aids can be utilized to aggregate the common

TABLE 1. QUALITATIVE DIFFERENCES FOR NOMINAL, INTERACTING, and DELPHI PROCESSES

	Group Type		
Dimension	Interacting	Nominal	Delphi
Overall Methodology	Unstruct. group meeting High flexibility High variability	Structured group meeting Low flexibility Low variability	Structured questionaire Low Variability
Role Orient. of Proc.	Socio-emotional group maint. focus	Balanced focus on social maint and task role	Task-instrumental focus
Relative Quantity of Ideas	Low	Higher	High
Quality Specific. of ideas	Low quality Generalizations	Higher quality High specific.	High quality High specificity
Search Behavior	Reactive Short problem	Proactive Extended prob.	Proactive Controlled problem
Normative Behavior	Conformity pressures	Tolerance for non-conformity	Freedom not to conform
Equality of partic.	Member dominance	Member equality	Respondent equality
Method of Problem Solving	Person-centered	Problem-centered	Problem-centered
Decision Closure	High lack of closure	Lower lack of closure	Low lack of closure
Attitude Toward Task Problem	Low task motivation	High task motivation	Withdrawn task motivation

Source: Van de Ven, 1974, pp. 96-97

ingredients within the structure. Due to the nature of these type of aids, users have to be proficient in analytical methods so that a valid structure may be applied to the associated problem.

Recomposition aids are used primarily for the task of performing laborious and often repetitive computational operations. One system is designed to use hierarchical multiattribute utility decomposition in order to analyze problems that are characterized by a large number of attributes or criteria (Saaty, 1980). Other systems can be used to examine alternative courses of action, incorporating a mixture of intermediate decisions and uncertain events.

Bootstrapping aids can be described as those that aim to replace the decision maker by automating the entire sequence of the decision-making operation and assisting the user in introducing content within an established decision-making procedure. Opposite this, recomposition aids are those that serve to aid the decision maker in the integration and further examination of the contents specified within the formalized decision model.

The methods discussed above cover decision aids that have been carried out after a formal problem structure has been defined, and rely on the implementation of good programming techniques defined by; (i) the constraints of decision associated algorithms, (ii) the design of the user-system interface, and (iii) the availability of computer technology.

However, the discussion has not touched on decision aids that are designed to be operational before the decision problem has been clearly formalized.

E. PROBLEM-STRUCTURING AIDS

Problem-structuring decision aids give a decision maker the opportunity to build a representation of the problem by integrating the component parts of the problem and the clarifying relationships between them. Most problem-structuring aids include editing programs and repetitive modules that allow the decision maker to inject new information into the problem structure as it is initiated or as the need for it is conceived. One example is MAUD4 (Humphreys and Wisudha, 1983) that uses a series of preconstructed displays that prompt the decision maker to decompose the decision problem in stages. Information entered by the decision maker is used to prompt for further elements of the decision problem resulting in the construction of a problem structure through an iterative process.

F. MULTIPLE-CRITERIA DECISION MAKING

The term "multiple-criteria decision making" (MCDM), signifies an interest in the universal category of problems that deal with multiple attributes, objectives, and goals. Therefore, with multiple-criteria decision making MCDM is utilized to resolve conflict within and between groups. To

resolve a conflict involves the consideration of multiple objectives. It accepts the conditions which created the conflict, and seeks a compromise, a settlement, or a consensus. Each party usually gives up something it originally desired. Both parties strive for a "fair" distribution of gains and losses. Negotiation, bargaining, and arbitration are common tools for seeking conflict resolution. In collective decision making it is often the case that multiple objectives and also multiple decision makers interact. Some sort of compromise then becomes mandatory.

G. GAME THEORY

Game theory was created to study the structure and resolution of conflict. The **theory of games** is a collection of formal models for studying decision making in conflict situations that are most easily displayed as games of strategy. The distinctive quality of game theory as related to decision making tasks is that the outcome to a particular participant, known as a player, depends not only on his own choices and the variability of chance, but also on the choices of one or more other paticipant. Players are normally the autonomous decision maker. However, centers of interest may be developed when two or more individuals decide to jointly agree upon a coordination of efforts resulting in a decision which might not be guaranteed if acting independently. The specified consequence to each player, are necessarily uncertain, because the choices of the other players are not known with certainty.

The defining quality of a **cooperative game** is that players may enter into mutually binding agreements. The assumptions underlying this approach are: (i) negotiations are mandatory with a view that previous conversation must take place; (ii) all previous signals discussed by each player are communicated without deception to their intended targets; (iii) all agreements are binding and enforceable by the rules, and; (iv) all results are unaffected by the prior negotiation process. The term "cooperative" for this type of game comes from the fact that players may conspire to their mutual benefit. A **noncooperative game** is one in which binding agreements prior to decisions are not permitted.

Superadditivity is a property of cooperative games that says that any two disjoint coalitions can do at least as well by joint effort as they can separately.

Essential to game theory is the dichotomy of **constant-sum** vs. **nonconstant-sum.** The most important reason for the distinction between constant-sum and nonconstant-sum games is that in the former, any difference in payoff between two outcomes for one player must be compensated for by differences in opposite sign in the payoffs of other players. In other words, if the game is constant-sum, whatever one player gains in proposing one outcome over another, the remaining players collectively must lose. When only two players are in the game, this means that their interests will always be diametrically opposed. However, the n-person constant-sum game is not one of pure conflict, as it may be possible for each of a group of players to gain at the expense of the remainder.

In **nonconstant-sum** games, gains realized by one player when moving from one outcome to another need not be compensated for by losses of the other players. In these games, interest centers on whether or not there are solutions that enable players to arrive at those cells which maximize the total payoff to all players. Another concern, less demanding than joint gain maximization, is whether outcomes are Pareto optimal. An outcome is **Pareto optimal** if there is no other outcome such that all of the players do better in the latter then the former. While all jointly maximum outcomes are Pareto optimal, the converse is not true since transfers among coalitions are prohibited.

Joint decision-making versus **negotiating** outlines the conditions by which members either work together to solve a problem or negotiate an acceptable joint decision. Problem solving occurs when the joint gain available to both parties is variable. It is a non-zero sum game from which both parties emerge as winners. The total payoffs to both parties will depend on their abilities to discover the compatibility of their interests and to discover or find ways to work together for mutual profit.

On the opposing side, bargaining to reach a decision occurs when the joint profit available to both parties is fixed, and, for the present, their relative shares have not been determined. Whatever one side gains is at the expense of the other. Therefore, it is a zero-sum game. One party is likely to attempt to modify the other party's perceptions of the benefits of various courses of actions so that the other party will be less resistant to a decision favored by the first party. The first party is to attempt to structure the other party's expectations about what outcomes would be minimally acceptable to the first party. The negotiators will take immovable positions and make threats to prevent the opposition from implementing the same operations. Any earlier commitments which become untenable will be rationalized away (Bass, 1983).

Resolution of conflict among multiple decision makers is often approached from the viewpoint of game theory, characterized by formulations with multiple payoffs. In addition to the discussion above concerning zero and non-zero sum games, the prisoners' dilemma illustrates another type of compromise programming used in collective decision making. An example of the prisoners' dilemma is shown in Table 2.

This a non-zero-sum game in which wins of one do not cancel the losses of the other. Being rational, according to game theory, Suspect 1 determines his strategy by taking into account all possible actions of Suspect 2.

If Suspect 1 confesses, then he can either get 5 years or go free depending on his partner's two options; if he remains silent he can get either 20 years or 1 year in jail. Symmetrically, Suspect 2 finds out that he too is better off if he confesses. The dilemma is: if they heed the very best advise and confess, they will both wind up with five years in

TABLE 2. **PRISONER'S DILEMMA**

		Suspect 2	
		Confess	Remain Silent
Suspect 1	Confess	5 Years, 5 Years	Go Free, 20 Years
	Remain Silent	20 Years, Go Free	1 Year, 1 Year

Source: Zeleny, 1982, p. 357

jail. If they both disallow the advise and remain silent, then they will remain in jail for only 1 year. According to von Neuman and Morgenstern (1982) traditional game theory fails at this point, where contact is made with more realistic conflict situations.

H. SUMMARY

Research on the social psychology of conflict emphasizes the need for striving cooperatively rather than with competitive motivation (Douglas, 1962, Morley and Stephesen, 1977). Such a motion stresses the necessity to shift from an initial competitive (or even hostile) stage to a more co-operative one. The ideal group problem solving in a non-co-operative environment should be characterized by a gradually evolving, cooperative search for mutually acceptable, equitable, and innovative solutions with which the group members feel that their individual objectives are met rather than scoring a "victory" over the others. While each party should actively strive to protect and advance their own interests, this should be done with a view to seeking arrangements that will benefit the other party as well. Furthermore, when there is a mediator, maintaining a cooperative orientation also applies to the helping mediator.

Assuming that the adoption of a cooperative orientation is the key to a constructive conflict resolution leads to the hypothesis that non-cooperation is a general case of co-operative decision-making. Therefore, from a system design point of view, it would make a great deal of sense to build non-cooperative Group Decision Support Systems (GDSS) in such a way that they can transform a competitive problem into a cooperative one.

IV. DESIGN ISSUES IN IMPLEMENTATION OF DECISION SUPPORT SYSTEMS

A generalized decision support system may be viewed as an interactive computer designed and implemented system that assists decision makers to solve unstructured problems utilizing data, models, and a dialogue or interface sub-system (Sprague and Carlson, 1982). The above definition seems to precisely delineate the internal workings of a decision support system, and may leave the reader with a perception that numerous "computer-based systems" have been developed today that easily perform to this ideal. However, further discussion of the special characteristics of both individual and group DSS are in order. It is interesting to note that although this discussion deals solely with an individual decision support system, and not a group decision support system, the basic model should remain intact except for added requirements determined by the collective decision process. As seen in Chapters 1 through 3, ones' understanding of decision theory may be utilized to build a framework or model in which to solve a series of problems in a group environment. At this juncture, a DSS or GDSS (such as Co-oP) may be developed.

The three stages of decision making are **intelligence, design, and choice** (Simon, 1960). **Intelligence** is described as a methodology required to search the environment for conditions that require decisions. This phase requires the collection of general data which is processed and then examined for trends indicating problems.

Design relates to the aspects of taking possible problems and creating, and analyzing possible solutions or courses of action.

Choice involves the selection of a particular solutions with subsequent implementation.

These "stages of decision making" assist an individual in understanding that decisions are made in a highly structured, sequential manner. The above stages of decision-making are critical in that they underly the objectives that a decision support system must satisfy (Sprague and Carlson 1982). A DSS should be required to support all three stages of decision-making and facilitate interaction between the phases.

A second objective that must be met by a DSS is the support of difficult, underspecified or **unstructured decisions** as well as structured decisions. Unstructured decisions may be defined as having a decision-making process that does not allow prior description of the problem before making the decision (Simon, 1960). Ad Hoc decisions are often unstructured because of unique circumstances, time pressures, limited or lacking knowledge, or many other reasons.

A decision support system should enhance decision making at all levels of an organization and integrate tasks between these levels. Based on Anthony's (1965) analysis, these levels or tasks are; (i) **strategic planning**; or decisions associated with setting corporate policy, choosing objectives, and selecting resources, (ii) **management control**; decisions related to assuring effectiveness in acquisition and use of resources, (iii) **operational control**; decisions related to assuring effectiveness in performing operations, and (iv) **operational performance**; decisions made in performing the operations. In order for these organizational levels or tasks to have effective meaning a range of information is required. At the operational level the information should be current, timely, and exact. The strategic level requires information that is historical, or summary-type information. Also, the decision tasks inherent at each level range from immediate, highly structured tasks that might be seen at the operational level to those longer-range, unstructured decision tasks at the strategic level.

The communications between decision makers support interdependent decision making and are illustrated by numerous classifications of decision makers. Hackathorn and Keen (1981) list three main types as **Independent, Sequential Interdependent**, and **Pooled Interdependent.** With independent decision making, a decision maker has full responsibility and authority to make a completely executable decision. With sequential interdependent decision making, the decision maker makes part of a decision which is passed on to someone else. Finally, with pooled interdependent, the decision must result from negotiation and interaction among decision makers. Due to the differences in decision making, different capabilities (i.e., personal support, organizational support, and group support) will be required to support each type of decision.

Reviewed briefly above, a decision support system must support a variety of decision-making processes but not be dependent on any single entity. This individual DSS must provide support that is process independent and under full control by the user (manager). It should be generalizable in nature, and easy to use and modify in response to changes in the user, the task, or the environment.

The representation of decision problems can be defined in terms of three structures, which bear a correspondence to the three types of decision aiding systems; bootstrapping, recomposition, and problem-structuring (Wright, 1982). The first method is problem representation with **fixed structures**. Here, the formal problem structure is predefined for a particular type of application. It is used repetitively to analyze different sets of contents. This method requires a simple interface within the decision making process that monitors information that is entered in the correct format and deals with erroneous entries. Although appraisals using the decision aid may be performed by an assistant, the decision maker will likely become familiar with its procedural use and become the sole user of the system. However, should further development and implementation be deemed necessary by the decision maker, the analyst or developer may again enter the arena to perform the task. Problems falling in the "**assumed structure**" category are those that require the decision maker and a system developer to generate a pre-defined structure or model such that the data required for the analysis of the problem is ready in a form appropriate for entry into a chosen decision model. Using an assumed structure, the role of this decision aid is to operate on the given structure, manipulating the designated data and combining it using algorithms that obey decision recomposition rules. The difficulties are encountered due to conflicts in opinion and proposes requisite modeling as a compromise. Requisite modeling requires that everything needed to solve the problem must be included in the model, or if the material does not fit in the basic model then at least it be incorporated in simulation operations to determine its effect.

The last structure portrayed by Wright (1982) deals with **elicited structures**. It is within the decision-making processes where the set of alternatives are known but where poor structure dominates and the analyst may be unavailable for assistance.

In these cases decision aids play a far more important and complex role, especially since the decision maker is uncertain about the ways in which to construe a subjective preference structure. It appears that in these situations the content of the problem and the structure of the problem is "fuzzy." It requires an active cooperation between the decision maker and the decision aid itself in order to determine the structure that accurately represents the decision problem. Because the structure is elicited from the user, it requires a large amount of direct input. At the same time the user will rely heavily on the structural guidance of the DSS.

Finally, one would conclude that this type of problem formulation require decision support systems that are capable of eliciting information about the problem in the most flexible and optimal manner. The problem structure appears to be one in which the DSS and the user interact in order to define and isolate both the specifics of the "unstructured" problem and the optimal DSS structure that will be utilized to answer similar questions.

Sprague and Carlson (1982) portray the components of an individual DSS as the **dialog** or interface subsystem, **data** subsystem, and the **models** subsystem. The individual subsystems and their relationships are portrayed in Figure 2.

The **DBMS** stands for the data base management software; **MBMS** for models base management software; and **DGMS** for dialog generation and management software. The important point to make is that the entire schematic represents a decision support system that may be observed from different levels or aspects such as; the user, the developer, and, as some refer to them, the toolsmith.

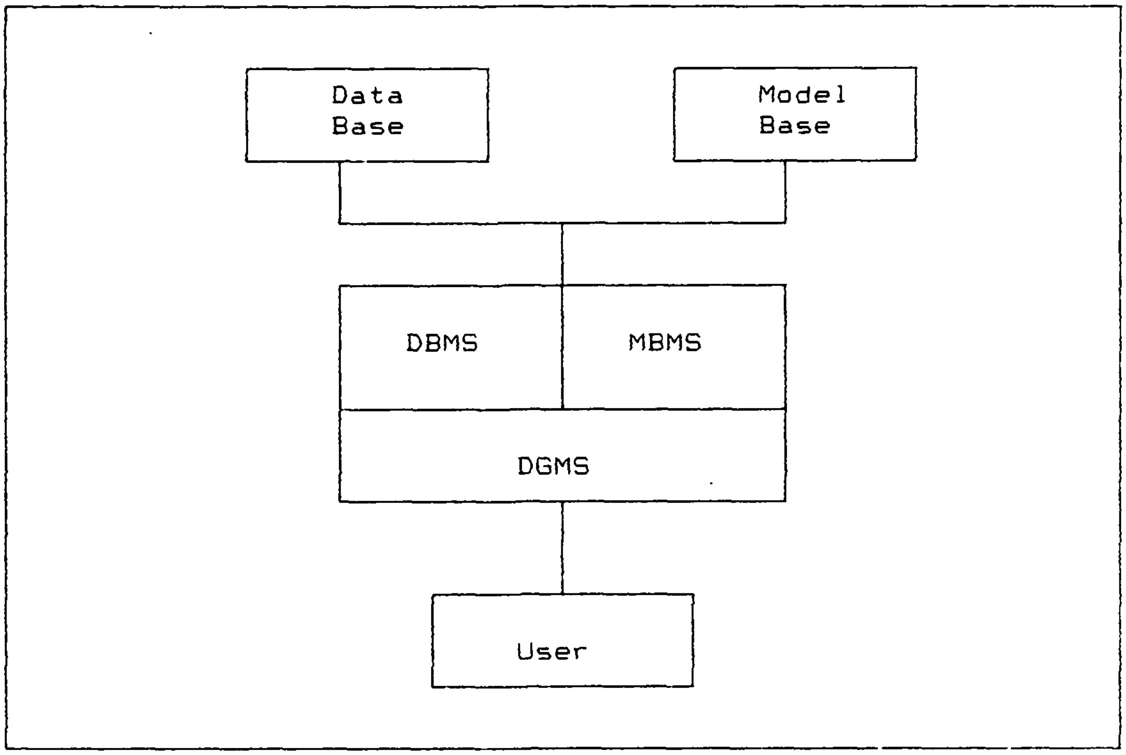


Figure 2. **Generalized Components of a DSS (Sprague and Carlson, 1982, p. 29)**

The dialogue or interface management software would seem to be the most important part of this individual DSS since it manages the interaction between the system and the user. Without a clear and comprehensive link between the user and a system that is supposed to assist in decision making, much of the power, flexibility, and usability characteristics of the system would be lost. More importantly, without a strong interface between user and system, the DSS would lose its value to the individual user, or the group. Bennett (1977) decomposes this user-system interface into the **action language**, the **display** or **presentation language**, and the

knowledge base. The action language is what the user can do to communicate with the system or provide input. The presentation language is what the user sees or visual output. The knowledge base is what the user must know in order to effectively use a specific DSS.

The data subsystem refers to the interaction of internal and external data sources, since decision making relies heavily on data and information not normally found in the strict transaction processing system (TPS). The "data base" approach allows the DSS and the DBMS system to be flexible enough to allow rapid additions and changes in response to adhoc requests by the user. The data base should have the ability to combine a variety of data sources through a data capture and extraction process. It should also have the ability to handle personal data so that the user can experiment with alternatives based on experience or judgement.

A **data base** is normally defined as a collection of data items stored in computer memory, while a **data base management system** is usually defined as a supervisory program used to create, maintain, access, update, and protect data bases. Since data is used by organizations for planning, control, and operation, a DSS planned for a group of decision makers will include external and internal data bases in order to conduct these functions. When the DSS provides the data collection and maintenance functions, data sharing among DSS may be difficult because of the data structures chosen or because of the desire to limit access to data which are particular to each DSS. If data sharing is difficult, maintenance of redundant data is likely. Sprague and Carlson (1982) suggests the design and building of a data base before DSS implementation because:

(1) The data base simplifies collection and maintenance of the data used by the DSS.

(2) The data base limits the set of functions and users that the DSS needs to support.

(3) It simplifies the design of the DSS.

(4) It eliminates potential conflicting performance and security requirements.

(5) The data base increases the chances of data sharing among DSS.

The above discussion points one to the conclusion that if a data base and its attendant DBMS are an initial design issue for a DSS, then there will be reduced costs of building and using the DSS, increased data control and sharing, and reduced data redundancy.

Another primary reason for installing a DBMS as an integral component of a DSS is its ability to integrate a variety of internal and external data that is required in decision making. The five data models used, at least for external data, are the **record model**, **relational model**, **hierarchic model**, **network model**, and the **rule model**.

The **record model** has a data structure combines data fields into records, and the data base is a combination of records. The operations generally conducted within this structure pertain to updating, deleting, and selecting a specific record. The limitations of the record structure are: (i) Each record must contain a key field whose value is unique, (ii) Record structures are "frozen" in that new record types cannot be added, and (iii) Every field must contain a value.

The **relational model** limits the data structure in order to take advantage of "flat files" wherein a field or attribute may not be repeated. This allows the structure to consist of relations based on predefined domains for specified fields. Therefore, this data structure consists of attributes (columns in a table) and tuples (rows in the same table). The characteristics of the relational model are; (i) The operations in this model operate on entire relations rather than on individual records, and (ii) The operations do not depend on the order of the fields or of the records. In other words there is a logical/physical independence between the logical data structure and the physical data storage. The constraints of this model are that each tuple must contain a unique set of values, and that normal forms must be conformed to. These constraints preserve the relationships among fields in a relation.

The **hierarchic model** utilizes multi-level trees to set up one-to-one relationships among records. In other words, the structures in the hierarchic model contains data captured in fields in the relational model. The hierarchy of data values is primary. This means that certain records must exist prior to the existence of others. Therefore, every data structure must have a root record, and no instance of a descendant record can exist without a parent record. Lastly, this model contains multiple copies of data instances which produces data redundancy.

The **network model** is similar to the tree or hierarchical model, except that it utilizes explicit or named links among the records to establish relationships. The resulting structure is many-to-many and thus support navigation in operations through a two-way link. The primary constraints in this structure are the maintenance of the links and the parent-child constraints.

The **rule model** is normally used in artificial intelligence applications and is based upon production rules for structure. This knowledge based structure relies on data that describes rules that allows decision making based on an inference engine.

A generalized DSS data base management system is derived in Figure 3 (Sprague and Carlson, 1982).

The model management subsystem is just as important to the operation of a DSS as is the DGMS or the DBMS. The user should have the ability to create new models quickly, and access and integrate modules flexibly. It should have the ability to catalog and maintain a generalized assortment of models, with direct significance to the decision making process and problem solution. The DSS should have the ability to establish a relationship between the models needed and the data base in use with the correct system commands.

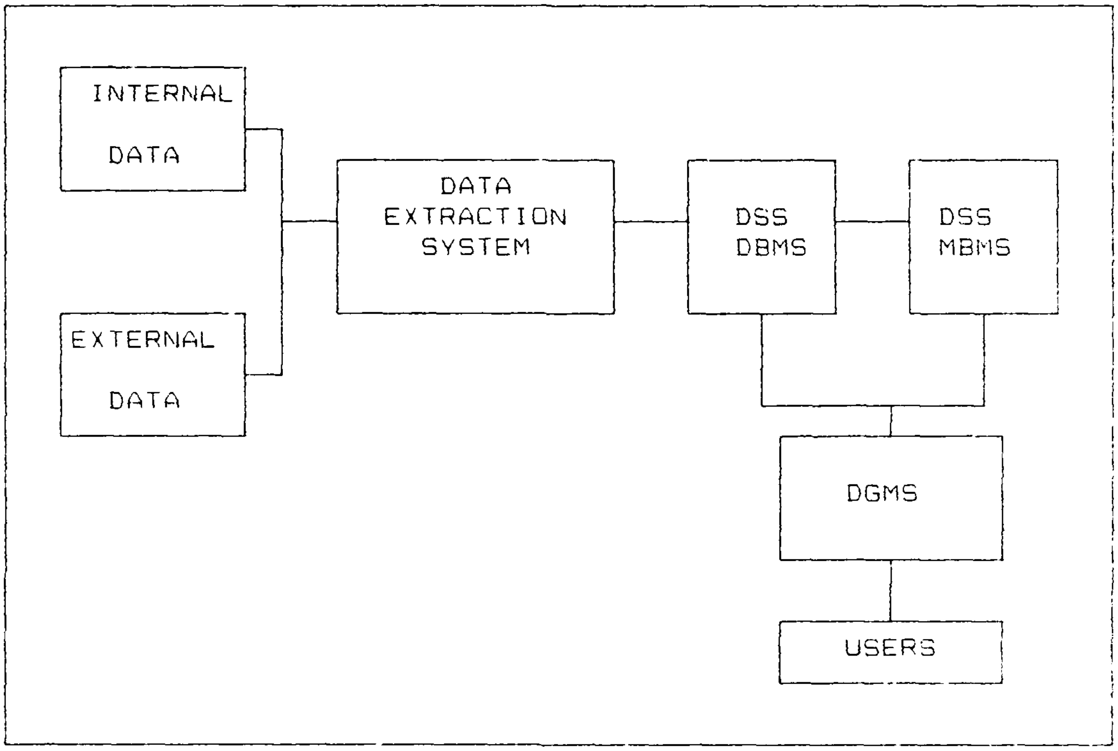


Figure 3. **A Generalized DSS DBMS Architecture (Sprague and Carlson, 1982, p. 244)**

In conclusion, it is critical to the development of a decision support system (DSS) and, certainly to a group decision support system (GDSS), that implementation is conceptualized through decision theory and conflict resolution methods. Ultimately, the formation of the essential computer algorithms or models derived from this conceptualization will be implemented as a computer-based decision support system. In a very real sense, ones' understanding of decision theory and DSS design issues may have dramatic effects on system effectiveness. Further, any enhancement or limitation observed in the final, delivered product may be the result of the above knowledge.

V. EARLY IMPLEMENTATIONS OF COMPUTER-BASED GDSS

This chapter describes previous attempts to move through the boundaries of single-user decision support systems to the arena of multi-user decision support systems. In many cases the early GDSS implementations were actually a series of individual DSS' linked to a centralized location through a simple graphics interface. Usually, the decision making process was site-dependent based on visual interpretation of group results. In other words, the computers were physically co-located with the focus on information aggregation and sharing.

The implementations described below attempt to provide a brief look at the development of group decision support systems and the implementation of various resolution methods.

A. TECHNIQUES OF AGGREGATION OF PREFERENCES

Due to their simplicity in algorithms, the techniques of aggregation of preferences are likely the most popular group decision techniques that have been computerized. Co-oP is an example of such application. It includes the sums-of-the-ranks, the sums-of-outranking-relations, the additive function and the multiplicative function.

B. INTERACTIVE MULTI-CRITERIA DECISION MAKING (MCDM)

An interesting approach to multi-criteria decision making allows for a very fluid approach to problem solving. Zeleny (1982) describes interactive MCDM procedures as methods for incremental articulation of preferences. A basic assumption is that the decision maker's preferences develop and mature only in conjunction with a particular problem. In other words, human preferences are not fixed in time or established in the singular, but are always changing, being situation-dependent, circumstances-shaped thought patterns. The important point to make here is that these evolving preferences act as a learning process and should be taken into account.

In contrast, some MCDM approaches concentrate on a priori articulation of preferences, or for example, they assume that all necessary information about a decision maker's preferences can be extracted prior to the actual problem solving, independently of a given decision situation. In this view, human preferences are relatively fixed and consistent, or there is no significant learning process. These are the primary assumptions underlying multiattribute utility theory and its derivative methodologies.

Other approaches do not attempt any substantial articulation of preferences before or during the problem-solving process. Preferences remain implicit, with the choice being arrived at through other means. After the final decision or solution has been arrived at, the preference structure can be made explicit. Therefore, these are the methods for a posterior articulation of preferences. The approaches included are linear multiobjective programming, multiparametric decomposition, stochastic dominance, and compromise programming.

The articulation of preferences in the interactive approach is conducted through a dialogue management system. An interactive conversational system, probably taking advantage of computer graphics, provides an opportunity for real-time interaction between program developers and the decision maker. Such a system can guide decision makers to what they consider the best compromise, without forcing them into an exhaustive examination of all the trade-offs.

Co-oP is another computerized group DSS based on MCDM. Its current version includes two MCDM techniques, i.e., the Analytic Hierarchy Process (Saaty, 1980) and the ELECTRE method (Bui, 1982).

C. INTERACTIVE DECISION EVOLUTION AID (IDEA)

IDEA is a general framework of requirements and assumptions on which theoretically sound interactive methodologies should be based. The IDEA approach is a graphic interaction tool for aiding the decision maker in the search for a solution. The following assumptions are emphasized.

The decision maker's preference function is unknown and evolving throughout the decision process. It is situation-dependent, subject to learning and "changes of mind."

The set of alternatives can be specified through constraints or through listing. The most preferred values with respect to each objective can be specified. That is, the ideal solution can be specified.

The decision maker prefers a nondominated solution to a dominated one and would most likely accept the ideal if it were available.

No weights of criteria importance are to be specified. They are implicit in the attention levels accorded to individual criteria during the selection process. No goals or satisfactory values are to be specified beforehand.

The decision maker is expected to characterize each solution as acceptable or unacceptable with respect to the ideal. An inability to make such a declaration is interpreted as indicating that the solution is unacceptable.

The decision maker must be allowed to introduce new alternatives, to add or drop some criteria, and to be inconsistent in the expression of criteria importance.

The IDEA approach utilizes the following procedures:

(1) The set of all nondominated solutions or nondominated extreme-point solutions is identified but not displayed to the decision maker.

(2) The ideal and anti-ideal solutions are computed. These two reference points identify the ranges or potentials for change for each criterion. Criteria potentials are displayed as a bar graph. The

direction of improvement proceeds from the bottom to the top of each bar.

(3) The bars could be either presented in their original incommensurate scales or scaled between 0 and 1 in terms of percentage values of the ideal.

(4) The decision maker starts at the anti-ideal. The decision maker generally attempts to exploit available potentials, either fully or by predetermined steps. Feasible increments or decrements are predetermined because of the finite listing of nondominated solutions.

(5) Any change in any potential is translated into all remaining criteria potentials, and a new bar diagram is displayed. Used-up portions are clearly identified, and the remaining permissable changes are displayed.

(6) In a few preliminary steps the decision maker is encouraged to reach for the ideal. Its unavailability is quickly realized, and the notion of necessary trade-offs is quickly learned. The purpose is to make all potentials as small as possible so that the ideal will be approximated as closely as possible. If all potentials could be reduced to zero, the ideal would be perfectly matched. In reality, there will be combinations ofpotential residues which the decision maker must judge in terms of their closeness to the ideal.

(7) The decision maker is allowed to retrace, following different paths, or use trial and error. It is desirable that multiple decision makers first use the technique separately, later joining in a committee for group negotiations. Ultimately the decision maker enters a subset of points which are cyclically entered again and again. A compromise has been reached.

(8) One tests whether none of the compromise solutions is truly acceptable. Their mathematical distances from the ideal are computed, and the results are made available to the decision maker for comparison. Some of the less important criteria can be temporarily removed in order to decrease the dimensionality of the problem.

(9) If none of the compromise solutions has been found acceptable, the problem must be redefined. New constraints and new alternatives must be brought into

the picture, different criteria considered, or the decision recommended for postponement. New alternatives should be generated as closely as possible to the ideal. Then the entire IDEA process should be repeated.

D. EXPERT87tm: A BOOTSTRAPPING TECHNIQUE

This computer-based expert system shell captures the "experts" knowledge base through reduction of the users intuition, and not through specific rule production or programming. Utilizing an individuals' intuitive knowledge, it decomposes the process into objective components that are more easily understood. The system utilizes a "function-based" algorithm or algebraic formulation to replace the decision makers' intuitive processes, which then acts as an expert system. (Magic7, 1986)

E. SUMMARY

This chapter reviewed major implemented group decision support systems and their design characteristics. Each technique described shares a similar characteristic in that they operate under the assumption of cooperation.

The next chapter will attempt to expand a previously implemented GDSS, namely Co-oP, to include non-cooperation among group members.

VI. DESIGN ISSUES IN IMPLEMENTING GROUP DECISION SUPPORT SYSTEMS FOR NON-COOPERATION

A GDSS may be defined as a network of computer systems that attempt to assist a group of decision makers resolve a collective decision making problem, but more than this, it must also assist in the collection or aggregation of multiple inputs from various sites with different systems. This is much more apparent then a single-user DSS since the design of a GDSS depends on the kind of group decision setting to be supported. In other words, many other factors such as distance between decision makers' systems, time effects on the system, centralization of control, and degree of cooperation have a bearing on the implementation and use of a GDSS (Jarke, 1986). Therefore, beyond the localized, site-intensive use of an individual DSS (IDSS), the communications interface is critical to the proper operation of a group of interconnected individual decision support systems, or GDSS. The differences between a individual DSS (IDSS) and a GDSS may be observed in Figure 4. The main thrust is what a developer must be aware of when designing a GDSS for resolving non-cooperative decisions.

The single user in the top part of the figure represents the traditional DSS model. The purpose of such a DSS is to enhance the users' cognitive processing capabilities and/or to facilitate the learning process. The bilateral relationship between user and DSS provides no communications support as required in cooperative decision making (Bui, 1985).

The second, or bottom configuration infers a multilateral relationship between members of a group via a network of individual DSS and group DSS. The functions of such a network of DSS are to support both the decision maker who is a member

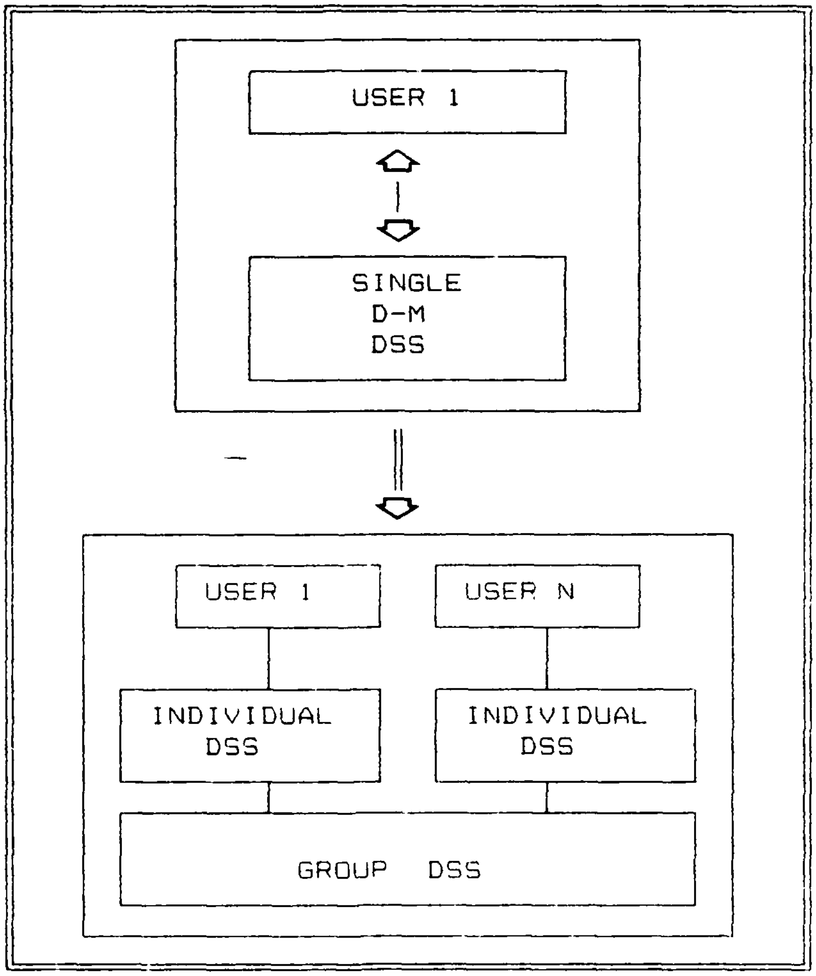


Figure 4. **IDSS to GDSS (Bui, 1985, p. 60)**

of the group and the group itself. However, only individuals interact with the system. The group as a whole is no longer a single user of the system.

The above discussion and factors listed below are concerned with the issues relevent to a GDSS such as Co-oP.

A. COMPUTER COMMUNICATIONS IN A COOPERATIVE GDSS

A possible GDSS architecture is fashioned by four integral components: the interface manager, the data manager, the model manager and the communication manager (Bui and Jarke, 1986). As discussed previously, the first three components are necessary to assure the effective use of Decision Support Systems (DSS). However, the carefully-laid foundation of a computer-based group decision-making process requires an additional function. This communication management function seeks to; (i) reduce miscommunication among (geographically) dispersed decisionmakers, (ii) support formal and informal communication, (iii) simplify data transfer protocols, (iv) offer flexibility in setting levels of information sharing ranging from limited to free exchange, and (5) accomodate protocol changes during the group decision-making process.

Communications control in computer systems includes operations that enable data exchange to take place. In a larger sense, communication protocols act as a set of rules and computer-to-computer formats that allow the proper management of communication between two stations (Puzman and Porizek, 1980). Distributed decision making (DDM) has been cited in the literature as a method to link decision makers or groups together through the use of communication ties (Rathwell and Burns, 1985). As such, DDM results in a mechanism for interaction between multiple systems in an organization to allow groups to cooperate with one another. DDM incorporates the distribution of GDSSs and extends this network to allow communications between DSSs. Built upon the DSS layer, distributed decision making and its' communication component is concerned with organizational communication and conflict in decision making and planning (Huber, 1984).

Bui (1986) states that "establishing reliable and efficient communication can only be viewed as a prerequisite for supporting computer-based distributed group problem solving." In addition, a generalized, communication-based GDSS not only has to indicate to connected, individual systems how to communicate, but how they should interact.

An architecture described to ensure the above reasoning is based on the Open System Architecture OSA-RM (ISO, 1982). This model defines a framework for providing data communication links between systems. Specifically, five communication functions are specified: link establishment, transmission opening, data exchange, transmission termination, and link releasing. The standardized model advances the decomposition of the communication link into seven layers. The services provided by each layer are described in Figure 5.

The reference to such a standard is justified by the fact that the use of an ISO network model has many examples in use and the standardization of protocols have, and are, undergoing constant revision and updating. This model for use with GDSS would; (i) minimize operating system incompatibilities, (ii) simplify protocol interfaces, (iii) assure

ISO LAYER	FUNCTION
1. Physical	Negotiation of access to the transmission media
2. Data Link	Physical management of data transmission
3. Network	Network routing and switching
4. Transport	End-to-end transport of messages traversing any topology
5. Session	Maintenance of the state of the dialogue between nodes
6. Presentation	Management of formats including the format control phase, the data transfer phase, and presentation phase
7. Application	Support of service-oriented functions

Figure 5. **The Layers of the ISO Reference Model (Bui, 1986, p. 151)**

reliability, ease of maintenance and portability, and most importantly, (iv) facilitate the integration of communication protocols in GDSSs.

The ISO architecture offers the modularity and transparency required for growth of distributed group decision support systems within the communications framework and, more importantly, allows the logical standardization of protocols that are necessary for the application and presentation layers to be mapped to an interactive decision-making situation. Therefore the application and presentation layers may be utilized in a GDSS with conversion protocols adapted to a single user decision support system.

Bui (1986) maintains this logical standardization through the use of a **group norm constructor**, **group norm filter**, and an **invocation mechanism** within layer seven (application layer). The group norm constructor will support service-related functions within a GDSS and monitor communication transfers between separate DSSs. This functional aspect will define a framework from which a set of protocols may be agreed upon by individual users of the GDSS. For instance, the consensually agreed upon protocols or set of rules may deal with items such as; (i) communications content, (ii) styles, (iii) channels, (iv) timing, or (v) synchroniztion or priority of messages. These parameters are received by the group norm filter which then enforce the defined protocols whenever communications are initiated by a GDSS user. When a data transfer request is received, the filter will check whether or not the communication activity is within pre-defined limits. If it is within valid guidelines, the communication is continued. Otherwise, the group norm filter would notify the user of the violation, and if required, offer the group member the appropriate protocol values. The last part of the GDSS application layer is the invocation mechanism. The invocation mechanism is another protocol that allows individual group members to request the possible modification of the protocols previously defined within the group norm constructor. Since the entire group has to achieve consensus in order to change the protocols, this mechanism offers flexibility of choice for the GDSS.

The presentation layer of a GDSS contains an Individual DSS-to-Group DSS formatter which maintains a set of presentation protocols for any possible type of data exchange in a group decision situation.

Based upon the above, four specific features necessary in a cooperative GDSS are: (i) support for multiple viewpoints of a problem, (ii) methods for aggregating the preferences of multiple decision makers, (iii) parameters that allow the establishment of several group norms, and (iv) protocols that aid the organizing of the group decision process (Bui and Jarke, 1986).

B. COMMUNICATIONS SETTING FOR NON-COOPERATIVE ENVIRONMENT

The design objective here is to create a computer-based environment favorable for constructive problem solving. Three tasks can be enivsaged. The first and most crucial task is a thorough orientation to clarify the potential GDSS user about the nature and purpose of using the GDSS as a fair mediation. The second is to manage ambivalence. The third is to "socialize" the users into the appropriate norms (Kressel, 1981). The Norm Constructor should be used as a "social" pressure to direct member toward support of the group norm in collective decision making. The rationale and motivation of these tasks are given below.

First, goal-oriented or outcome-oriented group members are more disposed to encourage a "state of harmony" than a "state of disharmony" (Zander, 1983). The Communications component should be used to orient or tune the group members to problem-oriented state. When focusing on problem-solving, members are expected to become more tolerant of group differences since they believe that tolerance is necessary for the good of the group.

Second, equal participation in defining collective intentions and actions have proven important in eliciting a common problem. The GDSS should be built in such a way that it promotes equal participation in constructing the group norm by widening the spectrum of communications support.

Third, the process of building a computer-based group norm should be flexible enough to reflect the collective decisionmaking structure, including power structure, and the possible formation of constructive coalition.

Expanding GDSS to a non-cooperative environment commands three additional design considerations.

First, focus should be on negotiation and settlement support. Supporting a non-cooperative decision making can refer to the manner in which the GDSS facilitates the achievement of constructive group problem solving. Criteria for an effective non-cooperative settlement process can be defined by the following criteria (Blake, 1979; Deutsch, 1973; Pruitt, 1981):

(a) Resolution of all relevant issues.

(b) Technically correct agreements expressed in clear, unambiguous language.

(c) Agreements that are fair and equitable relative to prevailing norms.

(d) Creative agreements searching for new opportunities that are beneficial to both parties.

(e) Satisfaction with the overall results.

(f) Parties comply with their terms of the agreements.

(g) Parties are better able to cooperate.

Second, emphasis should be on the behavioral process. The GDSS should facilitate behavioral processes before helping the decision-makers attacking tasks. The GDSS must be used as a means to build trust and confidence. Such a prerequisite is necessary to increase the degree of acceptance. Among specific tactics by which a sense of trust and confidence may be fostered are explicit statements of reassurance, the judicious use of self-disclosure and the maintenance of confidentiality. The outcome of such an effort is to reach a mutually informed commitment to the recourse to a GDSS as a mediation process. Such an emphasis on the behavioral process can be achieved by providing the following capabilities:

(a) Cooperative orientation to the group members in such a way that the parties define task as a cooperative effort to achieve mutual or compatible goals and avoid pseudo-issues that are merely a bargaining ploy to gain leverage over opponents (Rubin and Brown, 1975). Common interests and similarities are heightened, while downplaying opposite interests and values.

(b) Open Style of Communication: implies active, mutual participation in the give-and-take of negotiating.

(c) Search for reasonable and persuasive goals: that are well focused and achievable enough to resolve.

Third, facilitating the tasks of the mediator. Recourse to a human mediator or some sort of external agent may be necessary can be helpful to increase the degree of acceptance of the GDSS as a novel channel of collective decisionmaking. An effective non-cooperative GDSS should attempt to help the mediator accomplish this difficult task. When the role of the mediator is strongly not appreciated by one or more members, then he/she--and probably the GDSS--is no longer necessary. The issues here rely on how well the mediator can use the GDSS to maintain confidentiality and, when necessary, disclose information. The GDSS must support the mediator and educate him/herself about the nature of the problem confronting him/her. In particular the GDSS should help the mediator accurately diagnose the following aspects:

(a) The prevalence of group member misdiagnosis. The sources of misdiagnosis may include incomplete information available.

(b) The multiple loci at the problem.

(c) The group member's understanding of the role of the mediator. The mediator should have the possibility to access to other sources of information to expand his/her understanding.

C. USER INTERFACE

The development of either a DSS, or a GDSS dictates that a professional, standardized, and context-oriented interface between the user and the decision support system be established (Wright, 1982). For instance, graphics and color can be used to display information in the form of pictures, with standardized digitizing equipment easing the input of data or information into the system. One notion that challenges the above premise concerning standardization is that a group decision support system (GDSS) is a DSS that is specially designed without having the configuration of already existing DSS components (DeSanctis and Gallupe, 1985). In other words, every GDSS would, in reality, have quite different aspects built into the dialogue management or interface components. A generalized or specific GDSS is designed with goal of supporting groups of decision makers in their tasks. The GDSS contains built-in mechanisms which discourage or control development of conflict within these groups. Interestingly enough, the dialog management system covered in Sprague and Carlson (1982) for individual decision support systems contain question-answer designs, command language design, and menu design for input/output between user and system. This dialog management or user-system interface is based in software, as seen above, and hardware.

The GDSS system cannot be a number of copies of a single DSS design with its respective user interface. It must be designed as a group decision support system with specialized and, as required, generalized interfaces. Regardless of the specific decision situation, the group as a whole, or each member, must be able to access a computer processor and display information. According to Desanctis and Gallupe (1985), "Most sophisticated systems will include databases, along with model bases, very high-level languages for program writing, and interfaces with standard managerial-level software (graphics, statistical/OR packages, spreadsheets, etc.)"

A group decision support system would work effectively if decisions were made in a cooperative manner with little or no conflict between members. However, this is very seldom the case. Therefore, the dialogue management software must provide for various conflicting situations. In other words, an objective of GDSS should be to encourage the active participation of all group members. An important feature of the GDSS may be the facility of allowing anonymous input and evaluation of ideas. In addition, GDSS software might have features that actively encourage group members to voice dissident (or conflicting) opinions or play a "devil's advocate" role before a critical decision is made.

Secondly, special accommodations are needed for groups who have no prior experience working together. The GDSS should support the group during the initial phases of group formation. Special software might be used to query members on their expectations of how the group should function, and to feedback points of agreement and disagreement among members.

Finally, the measure of effectiveness for a user-system interface may be based on some detailed, and possibly qualitative measures such as; reduction in group conflict, degree of consensus, and type of group norms to develop.

D. DATABASE MANAGEMENT

The purpose of group decision support systems is to increase the effectiveness of decision groups by facilitating the interactive sharing and use of information among group members and also between group members and the computer (Huber, 1984). Three types of information sharing are described as essential to the decision support system in the group meeting context. Data bases may be utilized for "real time" analysis and discussion during a GDSS session. "What if" and other analytic software will enable the group to use information, even ad hoc information entries brought forth during the meeting, as input to interrogations made and responded to in real time. Numeric information sharing is a constant and critical decision aid during GDSS usage, where what if and goal seeking analyses generate information that leads to more informed choices. Sharing of textual information may occur to increase the effectiveness of decision makers.

The first occasion of textual information sharing is where real time text editing is performed by an interacting group. The second is where the decision group members are to share qualitative variables, such as problems, causes, or solutions that may occur during brainstorming, analysis, or planning. Relational information is often portrayed in the form of relational data basess. A data base, then, is extremely important in the group decision arena. Desanctis and Gallupe (1985) describe the importance of a qualified database when consideration is given to basic activities performed during group decision making; (i) **Information retrieval** includes selection of data values from an existing data base, as well as simple retrieval of information (including attitudes, opinions, and informal observations) from other group members, (ii) **Information sharing** refers to the display of data to the total group on a viewing screen, or sending the data to selected group members' terminals for viewing, (iii) **Information use** involves the application of software technology, procedures, and group problem-solving techniques to data for the purpose of reaching a group decision.

Basic features of a GDSS should include a "state-of-the-art database management system which can handle queries from all participants, create subschemas as necessary for each member, and control access to the database."

Bui, Jarke, and Jelassi (1985) support the view that a large database be made available to a GDSS through a micro-to-mainframe link. The GDSS in this case would be geographically dispersed with the micro-computer being the local node and the mainframe, with its' large economy of scale, containing the centralized database. The reason, it seems, is the rather apparent lack by a single-user DSS of facilities for data sharing and/or exchange between decision makers. The micro-mainframe architecture should support a link that; (i) extracts information from another location for local processing, with or without updating the source, and (ii) initiates requests for remote processing which would be impossible at the local level.

The following considerations for a collective database management system should be taken into account :

Users access data simultaneously. When utilizing a common source for data retrieval and update, or shared database, concurrency control problems may become evident. A concurrency problem occurs when two or more persons access the same data fields while others are attempting to modify them. (Gray,1981; Hewitt, 1976)

Users may have different areas of specialty. Differences in users' skills lead to the need for knowledge sharing by way of user-to-user communication (Jarke, 1986). In order to enforce and allow this type of communications, protocols are necessary to impose partial order on the access of transaction. Finally, there is a need, also within terms of an on-line protocol or pre-arranged management, for a mutual agreement on standardized terms and sub-databases to be used during data transfer (Jarke, Jelassi and Shakun, 1985).

Users may have different viewpoints. A group of decision makers often disagree about the exact alternatives, goals and evaluation methods that may be taken. In even cooperative settings the view of the facts may be different for different decision makers.

Users may change their minds. This will obviously occur during the negotiation process and can be viewed as part of an evolution of systems design (Shakun, 1985). The data base must embed a concept of evolution for individual and/or group record representations.

Users may have secret rules and data. Often, individual differences will not appear during negotiations within group decision-making. These differences may result in hidden agendas and data sets that are not represented openly. There is an organizational context outside the group. An interesting point often forgotten when designing group decision aids is the relationship between the group and their associated GDSS and the organization. Sprague and Carlson (1982) and Bui (1985) indicate that there are overriding data, policies, and strategies of the organization.

Therefore, the database should have two design strategies. In a subschema of the master database, a sub-database used by the group must be provided and protected from outside interference. On a strategic level, the database should contain information and procedures concerning organizational constraints imposed on the group decision process.

Bui, Jarke, and Jelassi (1985) discuss different levels of multiperson activity that should be supported by the database. On the first level or hierarchy a traditional office automation setting is viewed. This is where clerical tasks are supported and where concurrency control and knowledge sharing are at entry level.

A second level of office activity concerns joint analysis and design tasks, such as performed professional staffs. The DBMS must at this level support a higher level of specialized, complex integration of different viewpoints as well as multiple interactions of members within a transaction. It is assumed that users are still willing to cooperate fully on the accomplishment of a task.

At the third, and highest level of DBMS activity lies the semi-structured or unstructured decision making task. It would appear that at this level an inversion in tasks occurs in that full cooperation of decision makers is a special case. Collective decision situations are more often symbolized by apparent conflict in which it appears ill-advised for an individual decision maker to fully disclose his or her structure of preferences. GDSS must therefore deal with restrictions concerning partial preference divulgence and, in extreme cases, must deal exclusively with deliberate misinformation. In order to consolidate the above "information hiding" and "misinformation" into an organization context, the database must incorporate information about organizational policies, goals, and strategies (Bui and Jarke, 1985).

A general database architecture for the GDSS level is displayed in Figure 6. The figure illustrates the interaction of sub-databases in a multiuser environment.

The figure illustrates the use of sub-databases in a multiuser environment, where, for example, player N represents one decision maker and player M represents a second. Also, there is a mediator that may be representative of a group leader, or if without one, a series of software protocol that simulate the mediation process of a negotiator or leader. Player N can access his or her portion of the database which contains that data and problem representation

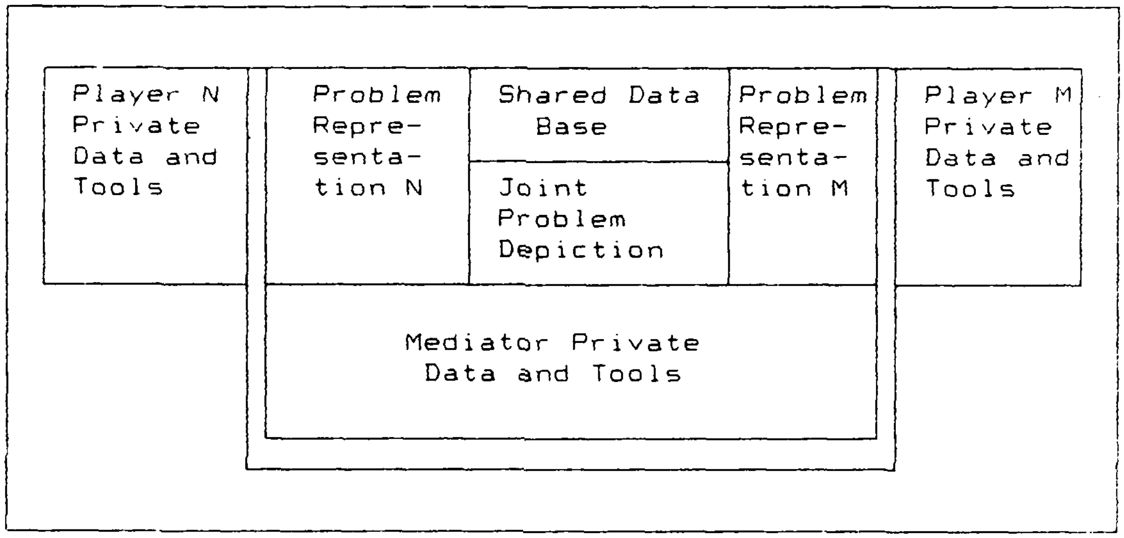


Figure 6. **A Collective Database Design (Bui, 1985, p. 12)**

most meaningful to the individual. Player M may do the same thing with his or her allocation of data memory locations. The mediator, through a set of predetermined protocols may view either sides' problem representation, the shared database, and the joint problem representation. In other words, in addition to the shared database, the mediator's access rights include the semi-private problem representations of each player. It is an interesting point that the group members do not have access to any semi-private problem representations but their own. Therefore, the shared database corresponds to a regulated n-person database which offers concurrency control that may be used as protection against outside interference. Since this is a protection device against interferrence, there should not be any internal concurrency problem.

The difference between this hierarchical database model and traditional databases is that the individual problem representations are only shared between one player (or group) and the leader-mediator. At the same time, the joint problem representation intervenes in the process of integrating and possibly supports the evolution of various viewpoints. Both types of sub-data bases are transparent outside the decision group and the evolution of decision making is guided by organizational information from the shared database.

According to Jarke (1985), there are two disadvantages to the data sharing concepts cited above. The first disadvantage stems from the current concept of a database transaction (Gray, 1981) which is geared more towards concurrency control than towards information exchange. Consequently, DBMS uses a concept of serializability which iterates that the effect of the concurrent execution of a set of transactions must be equal to that of any serial execution (Bernstein and Goodman, 1981). This can be a severe disadvantage if the purpose of a user transaction is communication with another user. Further, since transactions are supposedly independent, no mechanisms exist to provide for them to communicate with them directly.

Secondly, most current DSS (or GDSS) reside on microcomputers. Since the need for data management is explicit, the attempts to integrate microcomputer DSS databases with each other and with centralized mainframe databases are more recent in nature.

The following implementations should be included to facilitate the use of databases within a group decision support system:

(1) Locking mechanism: In a multiple user environment, group members access data simultaneously. This may lead to a situation in which two or more users access that same data item while others are attempting to alter it. This concurrency problem is particularly prevalent in a non-cooperative environment in that the search for orientation and understanding require a lot of data transfer. Also, since data requests are often short but frequent, the data component have quick locking mechanisms that enable locking data items before access, or validating a transaction after its completion.

(2) Partial ordering on the access of transaction: Differences in users's skills lead to the need for knowledge sharing via user-to-user communication. To keep track of information exchange and sharing, the data component should include partial ordering on the access of transaction.

(3) Filtering/sorting/time stamping mechanism: procedure using time stamp flags and alphanumerical sort to highlight differences in opinion.

(4) Distributed and/or sub-databases: Transfer of individual files to group databases or between individual files should be possible.

(5) Procedure to enforce privacy: Privacy enforcement mechanism should be made explicitly to all members.

(6) Communications using extended integrating rules to turn on/off links with external database.

E. MODEL MANAGEMENT FOR NON-COOPERATIVE GDSS

Bui (1985) discusses group decision making in terms of the modularization of group tasks into process-oriented and content-oriented tasks and resulting models. The problems associated with each in developing structured models are listed below:

(1) "First, despite the efforts of the content-oriented technology to help decision makers structure their initially unstructured problem, some unstructured parts will remain. This partial 'unstructurability' is due to uncertainty, fuzziness, ignorance, and an inability to quantitatively measure the complexity of the decision situation and the decision makers' preferences" (Stohr, 1981).

(2) "Second, attempts to resolve a group decision problem are rendered more difficult by human irrationality and emotionality when dealing with group interaction (Pruitt, 1981). It is then necessary to search for some process oriented methods that can support the unstructured part left by the content-oriented DSS, as well as for some communication system that collects, coordinates and disseminates information within the group."

One problem that comes up is the determination of whether a process-oriented or a content-oriented approach is best suited for solving a particular decision problem. Although it is assumed that a GDSS will decompose a group DSS into individual and group decision support (sub-) systems, it has been observed that decision making within organizations is often sequentially performed by different decision makers assuming different levels of expertise and responsibility, and using different decision-making techniques. Therefore, a group decision support system should provide models which support both type of group interaction by (i) maintaining input/output compatibility between the individual model component and the group model component, and (ii) allowing multi-tasking. In a multiple criteria group decision-making environment, the single user items stored in the individual model base should be independent from each other, but logically interrelated with the group decision model base. In other words, the models used by the individual should be connected to the group decision model in order for an aggregation of data into a group decision. Therefore, an individual should be able to utilize multiple models for preference determination at a concurrent rate while possibly using other applications.

Bui (1986) has listed several reasons for utilizing more than one decision modeling technique. None of the techniques of aggregation of preferences currently known in the literature can satisfy all of the five conditions imposed by Arrow's Impossibility Theorem (Arrow, 1963). The five rules are: (i) complete ordering, (ii) responsiveness to individual preferences, (iii) non-imposition, (iv) non-dictatorship and (v) independence of irrelevant alternatives. Arrow proves that, in general, there is no procedure for obtaining a group ordering that satisfies the five axioms or rules. In other words, a collective decision cannot be made without violating one or more of the five above mentioned rules. The combination of various techniques or models will attempt to use the same methods for aggregation of preferences in order to have the best group result possible.

Finally, The combination of techniques will offer an increased chance of group model acceptance and, therefore an increased chance for reaching consensus during negotiation.

There are approximately ten algorithms that may be implemented as models in the Group Decision Support System for derivation of aggregated individual preferences during cooperative decision making. The assumptions utilized under these techniques are; (i) All participants of the group problem solving share the same alternatives, although they may use different evaluation criteria, and (ii) Prior to the group decision making process, each decision maker or group member must have performed his or her own assessment of preferences. The output of such an analysis is a vector of normalized and cardinal ranking, a vector of ordinal ranking, or a vector of outranking relations performed on the alternatives.

The **min-max principle** is applied to concordance/diconcordance concept in ELECTRE, where a_j collectively outranks a_k when its lowest concordance and its highest disconcordance given by the group satisfy the outranking condition sanctioned by the highest concordance threshold and the lowest discordance threshold also given by the group. In a cooperative decision making environment, the minimum of concordance/maximum of disconcordance concept often helps reduce the number of non-dominated alternatives found in individual analyses to a smaller or even unique collective alternatives). However, the min-max principle only works when individual opinions are not extreme, and the number of alternatives are sufficiently large to generate consensus. Each group member can block a decision by setting a low disconcordance threshold or by disagreeing completely in the evaluation of the alternatives.

.The **sums-of-the-outranking-relations principle** is derived from the sum-of-the-ranks technique. According to Bui (1986), its use should be planned carefully since experience has shown that the idea of selecting the alternative that has the highest number of outranking relations works fine only when the number of alternatives are small.

The **pairwise comparison majority rule** suggests that the alternative that receives a majority of votes against every other alternative should be chosen.

The **agenda setting rule** or sequential pairwise comparison (Black, 1958) favors the alternative that enters last in the comparison process.

The **sum-of-the-ranks rule** (Borda, 1781) can be stated as the sum-of-the-ranks given by different decision makers to a specific alternative. In other words, a specific alternative is ranked by a finite number of decision makers and these ranks are added together. The lowest summed rank is chosen and that alternative is selected. Due to its computational simplicity this technique is widely used to determine consensus ranking. However, wher ties occur, results may be different.

The **additive ranking** algorithm applies when individual assessments of alternatives are expressed in cardinal values. A group evaluation of an alternative is the arithmetic mean of the rankings made by all group members.

The **multiplicative ranking** aims to give each group member more impact on the group outcome. A group evaluation of an alternative is the product of the rankings made by all group members..

The **minimum variance method** (Cook and Seiford, 1982) is an extension of the sum-of-the-ranks technique. When there are ties in ranking alternatives, this statistical algorithm searches for an estimated ranking that is close to the true ranking of the alternatives.

The **compromise ranking rule** that has its roots in transportation algorithms (Zeleny, 1982), attempts to minimize individual ranking differences by subtracting the rank mean from each of the decision makers alternative rankings. The lowest value is chosen for the decision process alternative.

The **weighted majority rule** is based on the observation that the participants of a group may not carry the same weight in the decision making process. Therefore, a vector of weights must be included in the aggregation of preference of rankings.

If no non-dominated alternative can be reached in the first round of the group decision making process, negotiations become necessary to resolve individual differences. One method proposed by Bui (1985) and incorporated in this work is the **Negotiable Alternative Identifier (NAI)**. It is based on a three step concept of an expansion/contraction/intersection mechanism. The NAI algorithm attempts to help the decision makers measure their degree of flexibility regarding their individual assessment of preferences by examining their distribution of preferences among alternatives. The NAI algorithm uses differential techniques to group ranked alternatives into two classes of preferences; the preferred and the least preferred sets of alternatives. Within each class, negligible differences in preferences between alternatives would increase the confidence of the decision makers not to discriminate them. Consequently, it would make it easier for the decision maker to trade them.

In other words, grouping alternatives that share close evaluation corresponds to **expanding** the preference space(s) of the decision maker from one best alternative to a set of more or less equally preferred alternatives.

The **contraction** operation constitutes the second phase of the NAI algorithm in which a given subset of satisfactory alternatives obtained from the expansion mapping, is collated into those that might exhibit a stronger preferential distribution than others.

The third and last step is the **intersection** operation. It derives a collective solution that is acceptable to all group members. Consensus is reached when there is at least one alternative that appears in every group member's subset of the most preferred alternatives.

F. SUMMARY

The previous discussion focused on a generalizable description of the design issues and problems inherent in the implementation of a group decision support system (GDSS). Much like a DSS, a GDSS contains primarily the same functional modules (e.g., the database management system, model base management system, etc.). However, the expansion results in a system that is highly communication-dependent, with the key toward group cooperation in a distributed setting. Additionally, it is a primary responsibility of the system to maintain a model base that will enhance group decision aggregation with a strong degree of precision. The user interface should not be machine-dependent, but distributed with context-based graphics. Overall, the challenge is to create a system that will allow the flexibility required within the group decision arena.

VII. EXPANDING CO-OP FOR RESOLVING SOME NON-COOPERATIVE ISSUES

The primary goal of this chapter is to briefly introduce Co-oP, and provide an insight into the major modifications or enhancements made within the program source code. Essentially, following the decription below and the major functions found in Figure 8, one may assess that the major enhacement to Co-oP is the functionality when dealing with non-cooperative group decision-making. The entire Co-oP system was previously implemented with Pascal code, and the present addition of 6,000 lines of code deal with non-cooperative and mediation modules. A further 3,000 lines of Pascal code has been modified to include improvements in the user interface (e.g., scrolling windows), code optimization, and data manipulation functions.

A. NON-COOPERATION AS A GENERAL CASE OF COOPERATION

Assume that a GDSS for cooperation is a special case of the GDSS for non-cooperation. Therefore, from a design point of view, it would make a great deal of sense to use an implemented GDSS for cooperation as a basic architecture that can be expanded, or modified to support non-cooperation.

CO-OP is a GDSS for cooperative multiple criteria group decision support system consisting of the predefined model, communications, interface, and database components described in the previous chapter. In its most developed state Co-oP will be a network of microcomputer-based process-driven DSSs for cooperative multiple criteria group decision making. Every individual member of the decision-making process maintains a DSS whereby the model base is based on multiple criteria decision methods (MCDM) and other personal decision aids. The group DSS, in turn, contains a set of aggregation of preferences modules and consensus seeking algorithms that may be used in addition to the individual MCDM. The primary aspect concerning Co-oP at this juncture is the use of the system for cooperative group decision support in which all group members concur with the basic steps of a multiple criteria problem solving process and norms imposed on the group members of a collective decision problem. These steps may be viewed within context of the CO-OP main menu and Figure 7. These steps consist of (i) problem definition, (ii) group norm definition, (iii) prioritization of evaluation criteria, (iv) individual selection of alternatives, (v) group selection of alternatives, and (vi) consensus seeking and negotiation. These six decision processes dictate the sequencing and timing of a CO-OP session, however, step number six is the non-cooperative portion of the system.

First, the group must collectively reach consensus on a specific decision problem and then define its' limits. In other words, the group will share the same alternatives and

M A I N   M E N U

1. MULTIPLE CRITERIA GROUP PROBLEM DEFINITION

2. GROUP NORM DEFINITION

3. PRIORITIZATION OF EVALUATION CRITERIA

4. INDIVIDUAL EVALUATION OF ALTERNATIVES

5. COMPUTATION OF GROUP DECISION

6. IDENTIFICATION OF NEGOTIABLE ALTERNATIVES

7. Help

8. Exit

Enter a Number: [ ]

Figure 7. The CO-OP Main Menu

evaluation criteria. The main problem with this initial task is the assumption that the group, albeit with a strong leader, will focus enough to agree on the primary problem at hand and identify the alternatives and evaluation criteria.

Second, the group has to identify its members and assign individual passwords and agree upon the way it handles data transfers, interactive conversation, and the type(s) of group decision techniques.

The third step pertains to the prioritization of evaluation criteria. This process may be performed either by having the decision makers assign numerical weights (from 1 to 10) to the criteria directly using the ELECTRE method, or by assigning values according to the Analytic Hierarchy Process (AHP) technique (Bui, 1985). The AHP algorithm consists of the following steps; (i) Perform a pairwise comparison of the evaluation criteria, (ii) Based upon the matrix of evaluation, compute the priority vector by computing the eigenvector of this matrix, (iii) For each evaluation criterion, perform pairwise comparison of alternatives using the same evaluation technique as used for the criteria, and (iv) Calculate the final vector of priorities. The evaluation criteria may also have priorities assigned according to a collective pool, sequentially based on member expertise, or in an aggregate mode.

The fourth step allows the decision makers to utilize his/her preferred algorithm (MCDM) to individually evaluate alternatives.

The fifth stage is the computation of group results using pre-defined aggregation of preferences techniques. Four techniques of aggregation of preferences have been used in CO-OP. They use the individual MCDM outputs to compute group results. CO-OP also allows weighing of users' decisional power.

The most interesting point arises when a unanimous decision is not reached by the group. It is at this sixth and last step that a consensus-seeking algorithm may be evoked to move through the impasse. This algorithm is the **Negotiable Alternatives Identifier (NAI)**.

To review, one can observe that what has been accomplished to this point is the development of a GDSS that has included many of the conflict resolution methods discussed in previous chapters. The current Co-oP system has included the major facilities often thought of as being in a DSS, however, it also facilitates the functions of cooperative, non-cooperative (NAI) decision algorithms, and the communications protocols required of a GDSS. Figure 8 illustrates the functional data flow of basic Co-op system.

B. N.A.I: A COMPROMISE-SEEKING ALGORITHM FOR GROUP DECISION SUPPORT SYSTEMS

According to Bui (1985), the use of techniques of aggregation of preferences are simple and easy to learn and are most commonly used in real world group decision-making situations. However, in many cases, it was noted that consensus was very rarely reached after the first round of decision making. It is related that techniques of aggregation of preferences have proven inappropriate in providing a flexible framework for identifying possible areas of negotiations, and more important, areas of concession making. In essence, a GDSS must be able to provide a facility to reach beyond the simple consensus-driven algorithms utilized in aggregation of preferences because the system would be incomplete and probably not be utilized by a decision-making group. In effect, when a technique of aggregation of preferences (e.g., sum of the ranks) is used to sanction a group decision, it

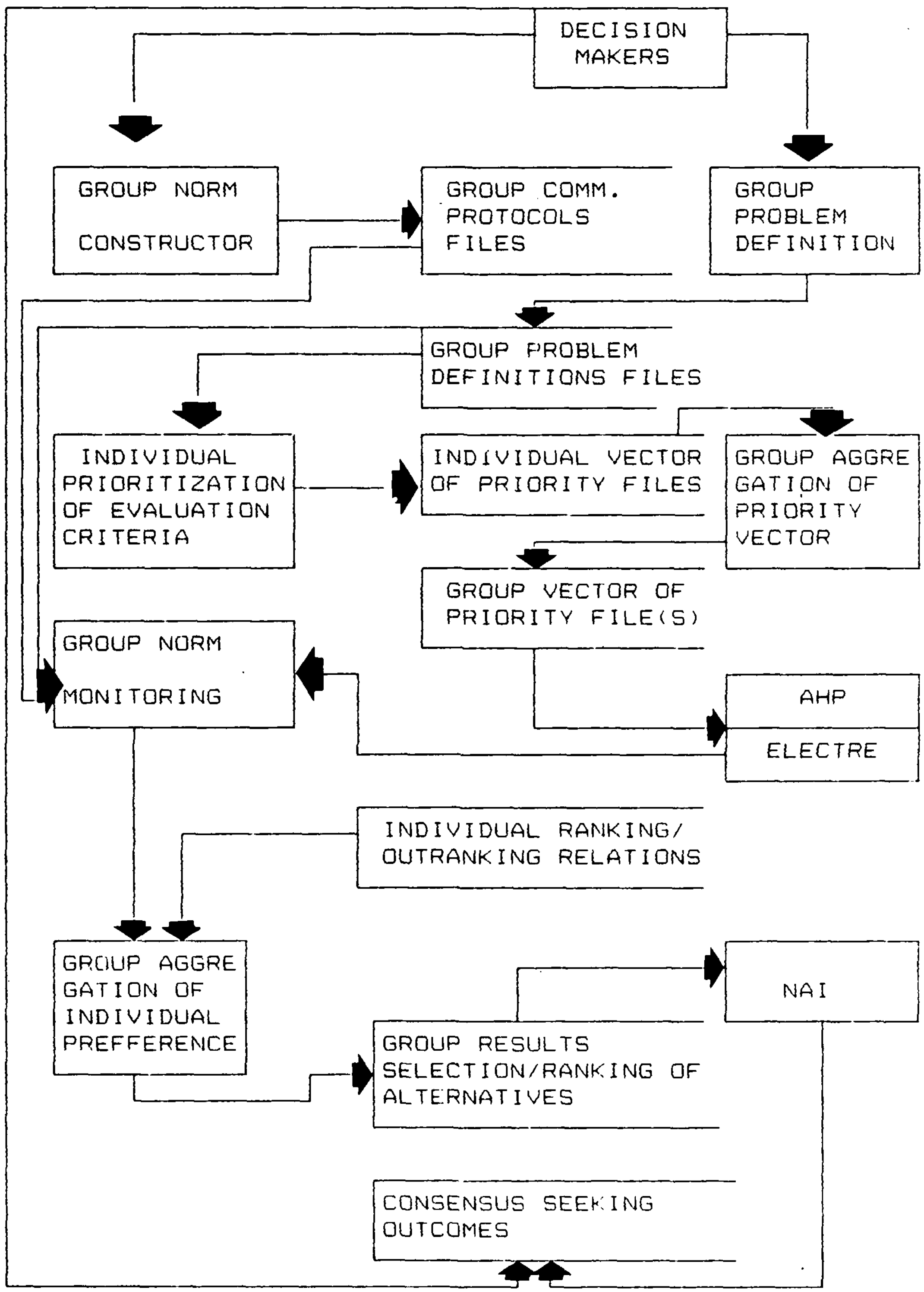


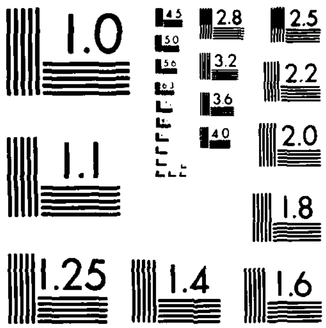
Figure 8. Major Functions of Co-oP (Bui, 1985, p. 197)

may leave the group member(s) who do not agree with the primary collective decision either unhappy or frustrated. It is at this point that the members have to resolve the conflict, and if not accomplished, the decision must be forgone or delayed until conflict dissolution takes place. The group decision support system will probably not meet the needs of this group decision making situation and be discarded.

The importance of quantitative judgments as inputs to the decision making process within groups is discussed by Wright (1985). Judgmental input is needed as input to decision making due to factors such as (i) lack of objective data, (ii) high levels of uncertainty about future conditions and effectiveness of actions, and (iii) the desire to include in the decision things that are difficult or impossible to measure. The most common method to assure the quality of decisions is to obtain multiple opinions or to use a panel of experts. However, if the group itself is the decision-maker, then there are issues of equity among members, of satisfaction with the decision-making process, and of commitment for carrying out the action finally agreed upon. It is further stated that the use of aggregation of preferences or judgments will provide a potentially higher quality decision considering a given set of criteria and evaluation methods.

AD-A175 263 NON-COOPERATIVE GROUP DECISION SUPPORT SYSTEMS: PROBLEMS AND SOME SOLUTIONS(U) NAVAL POSTGRADUATE SCHOOL MONTEREY CA A KARDOS ET AL. SEP 86 2/2

UNCLASSIFIED F/G 5/1 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

C. CO-OP II: AN EXPANDED GDSS ARCHITECTURE

Co-oP II is a expansion of Co-oP a GDSS for cooperative multiple-criteria group decision making to tackle some classes of problems under hostile environment.

The following additional system functions destinguish Co-oP II from Co-oP:

(1) An IDSS for the Mediator is added to the Co-oP system. It gives the mediator the exclusive rights to setup the communications protocols for the involved parties (designed and implemented).

(2) A system component called TOUCHSTONE is attached in parrallel (i.e. multi-tasking) to the Co-oP system to facilitate the search for common goals. The purpose of the TOUCHSTONE module is to provide a gradual pace, involving a shift from a competitive to a cooperative environment.

TOUCHSTONE is multiwindow-based system composed of (i) a modified Delphi technique driven procedure, and (ii) a Chatter Box to allow informal information exchange defined and controlled by the commmunication manager. Through TOUCHSTONE, the mediator guides group members to agree upon a mutually acceptable group norm (being implemented).

(3) Distributed database structure. Individual working data files are stored at the individual level (e.g., local drives). Group results are stored in the group database that can be accessed by individual members according to certain group norms. Selective access rights to databases are granted to the mediator (being implemented).

(4) In each IDSS, a MESSAGE CONSTRUCTOR is attached to the model component. The purpose of the message constructors to help the group member to construct messages in such a way that they are informative and persuasive arguments before they are sent to others (partially designed but not implemented).

(5) A Rule-based system to assist the mediator to dynamically monitor the group norms throughout various phases of the decision making process (not yet designed).

(6) A multi-attribute utility model is added to the group model base to support trade-off between multiple goal space (designed but not implemented yet).

D. PROBLEMS

Often, there are some competitive problems in which a GDSS might not work. Such problems are characterized as follows:

(1) High level of Intra-party conflict: if one of the parties are ambivalent about the problem to be solved or about the desirability of dealing openly and fairly with other parties, it is logical to assume that the recourse to a GDSS can be less useful. Group member's ambivalence about using a GDSS may derive from conscious or unconscious wishes to attack rather than to negotiate; from fears of becoming vulnerable because of the other's greater negotiating skills or resources; and from ignorance of the goals and methods of the mediator.

While the last two possible sources could be remedied by the Group Norm Constructor and the Help facility in GDSS, gaining the crucial cooperation of a group member is beyond the capability of the GDSS. More important, the ambivalence generated by one party can spread and intensify quickly to others.

(2) Well-established, rigid patterns of destructive interaction. Studies of labor mediation reveal that the worse the state of the parties' relationship with one another and the more intense their conflict, the dimmer the prospects for effective mediation. While a GDSS is less apt to be perceived as impartial or biased than the human counterpart, its use in a intense and long-standing conflict can be inconsequential. The recourse to a GDSS is meaningful only to parties experiencing moderate levels of conflict and in whom there is some confidence.

(3) Scarcity of divisible resources. When resources are scare, trade-offs among parties becomes difficult if not impossible (particularly with the Multi-attribute Utility Model.)

(4) Unbalance or disparities in relative bargaining power. When their is unbalance in relative bargaining

power, the stronger party is likely to be less motivated to compromise and more likely to use intransigent tactics. Meanwhile, the less powerful party may react with passive concession or reactive defiance (Deutsch, 1973; Rubin and Brown, 1975). Such an ill-matched confrontation does not constitute a sound basis for settlement (Kressel, 1981). TOUCHSTONE and the mediator may not work.

VIII. CRITIQUE ON N.A.I.

A. INITIAL COMMENTS

In order to follow this chapter, it is imperative to understand the Negotiable Alternatives Identifier presented by Bui (1985). This chapter pertains to a detailed discussion of the NAI algorithm and proposes some new heuristics to improve the contraction scheme. While NAI consists of an important effort to allow a GDSS to support non-cooperative decision making, it is not necessary to fully understand this chapter to capture the overall discussion on non-cooperative issues addressed in this thesis.

Further information attempts to illustrate and resolve weaknesses of the NAI algorithm which arise in some situations. NAI becomes flawed in the **contraction** operation where the most-preferred set of alternatives results. The algorithm to determine the cut-off point for the set of the most preferred alternatives performs worse and worse the higher the degree of indifference of a decision maker. This is shown with an example in section C. In Section D an algorithm is developed to remedy these imperfections. Section E contains a set of examples to demonstrate the results of the proposed algorithm. A brief introduction of the basic concepts is given in the next section.

B. THE NAI ALGORITHM

To demonstrate the working methods of the NAI algorithm the following example is used. Table 3 exhibits the cardinal rankings of a hypothetical couple who are searching for a shopping place. Six alternatives are considered: Macy, Mervyns, Sears, JC Penny, Woolworth, and Navy Exchange.

In the example, the cardinality n of the set of alternatives is 6. After determining the cardinality n the structural index of preferences **SId,i**, where **i** = **2, ...,n**, is computed (expansion operation). The lowest **SId,n** determines the subset of the preferred alternatives:

$$SId,n* = \min (SId,i)$$

where n* (first cut-off point) represents the first n alternatives that form the subset of the preferred alternatives.

In step 2 (contraction operation) the cardinality n of the set of preferred alternatives is n*. Here the ratios **Ld,i**, where **i** = **2, ...,n***, between the sum of all preferences

TABLE 3. **INDIVIDUAL RANKINGS**

	CARDINAL RANKING		ORDINAL RANKING	
	Husband	Wife	Husband	Wife
Macy	.11	.40	5	1
Mervyns	.19	.20	3	2
Sears	.22	.17	2	3
JC Penny	.18	.15	4	4
Woolworth	.05	.06	6	5
Navy Exchange	.25	.02	1	6

assigned to the preferred alternatives and that of the residual alternatives are computed. The maximum ratio (Ld,i*) determines the second cut-off point for the set of the most preferred alternatives.

TABLE 4. THE RESULTS OF THE EXPANSION AND EXTRACTION OPERATION

Husband:

Alternative	SId,n	Ld,i
Navy Exchange	--	--
Sears	.57	.62*
Mervyns	.41	.41
JC Penny	.31*	.48
Macy	.32	--
Woolworth	.39	--

Wife:

Alternative	SId,n	Ld,i
Macy	--	--
Mervyns	1.00	.88
Sears	.65	.57
JC Penny	.49*	.96*
Woolworth	.57	--
Navy Exchange	.84	--

Step 3 intersects first all sets of the preferred alternatives and second all sets of the most preferred alternatives of every decision maker.

In the sample case two persons are involved in the decision making process. The result of the expansion, and contraction operation is illustrated in Table 4. Table 5 shows the intersection operation for both sets of alternatives.

TABLE 5. THE RESULTS OF THE INTERSECTION

Expansion:		
Husband:	Wife:	Intersection:
Navy Exchange		
Sears	Sears	Sears
Mervyns	Mervyns	Mervyns
JC Penny	JC Penny	JC Penny
	Macy	

Contraction:		
Husband:	Wife:	Intersection:
Navy Exchange		
Sears	Sears	Sears
	Mervyns	
	JC Penny	
	Macy	

C. THE INDIFFERENCE CASE

To demonstrate that the algorithm determining the cut-off point for the set of the most preferred variables does not work an example of complete indifference is used. Continuing with the shopping example and assuming that the husband does not care where to go shopping the following must be the outcome of the NAI algorithm:

(1) The set of preferred alternatives is a subset (but not a proper subset) of the set of alternatives.

(2) The set of the most preferred alternatives is a subset (but not a proper subset) of the set of alternatives.

An examination of table 6 shows that the most preferred alternatives of the husband are Macy and Mervyns. In this special case (indifference) the most preferred set of alternatives always contains two elements unconstrained by the cardinal value of the initial set of alternatives. This clearly contradicts with the required outcome (all alternatives must be in the set of the most preferred alternatives) and cannot be accepted as a solution.

It can be observed that in the indifference case $SI_{d,k}$ and $L_{d,k}$ ($k = 2, .., n^* - 1$) have always the same value. This observation is utilized to develop an algorithm to compute the cut-off point for the most preferred alternatives. Instead of considering exclusively $L_{d,i}$ to find out the cut-off point it is proposed to take the difference between $L_{d,k}$ and $SI_{d,k}$ under consideration.

TABLE 6. NAI RESULT OF COMPLETE INDIFFERENCE

Husband:

Alternative	Cardinal Ranking	SId,n	Ld,i
Macy	.17	--	--
Mervyns	.17	.50	.50*
Sears	.17	.33	.33
JC Penny	.17	.25	.25
Woolworth	.17	.20	.20
Navy Exchange	.17	.17*	--

D. CUT-OFF POINT ALGORITHM

Based on the assumption that the set of the most preferred alternatives is always a **proper** subset of the preferred alternatives, except all cardinal rankings are of equal value, the following algorithm to compute the cut-off point for the most preferred alternatives is proposed:

<u>STEP ONE:</u> (Initialization Step)

Set Ld,m = 0, where m = maximum number of alternatives

Set z = 0

Set p = 0

Set D,1 = 0

Remark: To set Ld,m to zero is necessary to assure that the scanning process which is performed in step two works under each possible condition. In the shopping example the zero value is assigned to Ld,6.

<u>STEP TWO:</u> (Scanning Step)

Remark: The idea of the **scanning step** is to obtain the differences **D,k** between Ld,k and SId,k. This scanning process starts at k = 2 and continues until either D,k is less then zero **and** D,k-1 is greater than zero or k is equal to the cut-off point (n*) of the set of the most preferred alternatives.

2.1. Set k = 1, where k = (1, ..,s*), where s* = stopping point of the scanning process

2.2. k = k + 1

2.3. Set D,k = Ld,k - SId,k

2.4.1. If D,k = 0,

then z = z + 1,(Count where D,k = 0)

Go to 2.2

2.4.2. If D,k > 0,

then p = p + 1, (Count where D,k > 0)

Go to 2.2

2.4.3. If (k = n*) OR ((D,k < 0) AND (D,k-1 > 0)),

where n* is the cut-off point for the preferred set,

then set s* = k,

Go to Step THREE (Stopping point is found)

2.4.4. Go to 2.2

STEP THREE: (Solution Step)

3.1. If (s* = 2) and (D,2 < 0),

then L,cut = 1,

where L,cut is the cut-off point for the set of the most preferred alternatives,

Go to END

Remark: Solution for the cut-off point is found.

3.2. If (s* - 1 = 2) AND (n* <> s*) AND (D,2 > 0),

then L,cut = 2,

Go to End

3.3. If z = k - 2,

then L, = s*,

Go to END

Remark: Solution for the indifference case

3.4. If ((D,k-1 >= max (D,l)) OR ((p = k-2) AND (p >= 1)))

AND (p > 1) OR ((k-2-p-z) >= 1)) AND (ca,1 <> ca,2),

where l = 1,...,k-1,

where ca,1 and ca,2 are the cardinal rankings of the alternatives which are ordinal ranked first and second,

then L,cut = 1,

Go to End

3.5. L,cut = max (D,k), where k = 2, ..., n* - 1,

Go to END

Remark: The maximum value of the set of D,k is calculated to determine the cut-off point

End: Solution has been found

E. EXAMPLES

Two examples are shown to demonstrate that the proposed algorithm performs better than the original one. The indifference case and the first example are used to recognize the differences.

The result of the NAI algorithm for the indifference case is shown in Table 7. A satisfactory result is obtained. The sets of the preferred and most preferred alternatives contain all alternatives, which concurs with the expected outcome.

Table 8 illustrates the outcome of the proposed NAI algorithm using the initial example which is discussed in section B. The result for the husband concurs with the initial one and is not shown again. Considerable differences

TABLE 7. THE NAI RESULT OF COMPLETE INDIFFERENCE, CALCULATED WITH THE PROPOSED ALGORITHM

Husband:

Alternative	Cardinal Ranking	SId,n	Ld,i
Macy	.17	--	--
Mervyns	.17	.50	.50
Sears	.17	.33	.33
JC Penny	.17	.25	.25
Woolworth	.17	.20	.20
Navy Exchange	.17	.17*	--*

occur in calculating the most preferred set for the wife. As a result of the proposed algorithm only one alternative (the initial set contains four alternatives) is contained in the most preferred set. Examining the cardinal rankings one can observe the following:

(1) The cardinal value for the favorite alternative of the wife (shopping at Macy) is two times higher than her next favored alternative (shopping at Sears).

(2) She favors (40%) shopping at Macy almost as much as going to the remaining five shopping places.

Both of the above points are reasons why it makes more sense that the set of the most preferred alternatives contains only one element. So, one can conclude that the proposed algorithm shows much better results than the original one where only the maximum value of Ld,i was determined. One cannot claim that the current algorithm shows always desireable results under each condition, since the set to test such

TABLE 8. THE RESULTS OF THE EXPANSION AND EXTRACTION OPERATION USING THE PROPOSED ALGORITM

Wife:

Alternative	Cardinal Ranking	SId,n	Ld,i
Macy	.40	--	--*
Mervyns	.20	1.00	.88
Sears	.17	.65	.57
JC Penny	.15	.49*	.96
Woolworth	.06	.57	--
Navy Exchange	.02	.84	--

algorithm tends to infinity. But one can assume that in most situations, even in very special ones (e.g., indifference case), the proposed algorithm shows a satisfactory solution.

IX. CONCLUSION: REFLECTIONS AND CAUTIONS REGARDING A GDSS APPROACH TO CONFLICT RESOLUTION

The purpose of this thesis was twofold; (i) Explore some design issues for building group decision support systems for non-cooperation environments, and (ii) Expand CO-OP, a cooperative multiple criteria group decision support system, to support some particular classes of non-cooperative group decisions. Specifically, the following features have been added to CO-OP:

(1) Scrolling windows to handle group problems with large size.

(2) Code optimization to provide fast feedback to group members.

(3) Improved heuristics for the Negotiable Alternatives Identifier.

(4) Implementation of the Mediator module.

(5) Allow a more advanced data manipulation algorithm to promote data exchange in competitive environments (e.g., data security and sharing).

Assuming that the sine qua non of effective non-cooperative problem solving is to restore the trust and confidence of the parties, the most important mission of a GDSS is to reduce resistance to use the system as a channel or medium for resolving a collective problem. Even under the best of circumstances, attaining this mission can be very difficult.

A GDSS in a non-cooperative environment as opposed to a cooperative environment should (i) seek and maintain acceptability (ii) while simultaneously intervening to reduce hostilities and effectuate a more promising interpersonal climate. The Group Norm Constructor discussed in this paper can be used to implement an array of tactics to separate the parties, invoke norms of cooperation and fair play, interr-upting dysfunctional or hostile exchanges, educate the parties about their mutual role in negative transactions, invoke their mutual interest in solving the collective problems, and so on.

Since the ability of the parties to cooperative with one another is the primary predictor of a successful outcome (Kochan and Jick, 1978), several observations and caveats of the GDSS approach are in order. First, the use of a GDSS seems appropriate only for parties for whom an ambience of cooperation already exists or where the prospects of developing it quickly are relatively good. The recourse to a GDSS for a joint problem-solving venture should make sense to all involved parties.

Under certain circumstances, GDSS could be used as a appropriate means to handle non-cooperative problems with a minimum of competitive conflict in that a GDSS is a promising adjunct to the exclusive use of human mediation in orchest-rating a constructive group decision making process. Benefits derived from using GDSS may include the improvement of communication, understanding, and problem settlement.

Is the process of designing non-cooperative GDSS germane to a human and formal negotiation? May it be equally, or more helpful in this respect than non-GDSS mediation (probably better than inexperienced mediators)? For what types of parties is what communications norms most likely to be helpful? Much more research is needed before we can even be confident that attempt to build GDSS for negotiation is fruitful. What are the criteria to distinguish a "cooperative" (constructive) stance from a "non-cooperative" (destructive) one? How to identify obstacles that stand in the way of achieving a constructive settlement? What GDSS intervention strategies are the most useful?

LIST OF REFERENCES

Ackoff, R.L., The Art of Problem Solving, Wiley and Sons, 1978.

Anthony, R. N., Planning and Control Systems: A Framework for Analysis, Harvard University Graduate School of Business Administration, Studies in Management Control, Cambridge, Mass., 1965.

Bass, B.M., Organizational Decision Making, Richard D. Irwin, Inc., 1983.

Bennett, J., "User-Oriented Graphics, Systems for Decision Support in Unstructured Tasks," Association for Computing Machinery, pp. 3-11, 1977.

Bernstein, P. A. and Goodman, N., "Concurrency Control in Distributed Database Systems," ACM Computing Surveys 13, v. 2, pp. 185-221, 1981.

Black, D., The Theory of Committees and Elections, Cambridge University Press, Cambridge, Mass., 1958.

Blake, R. R., "Intergroup Problem-Solving in Organizations: From Theory to Practise," in The Social Psychology of Intergroup Relations, Ed., Austin and Worchel, Monterey, California, Brooks, Cole, 1979.

Bui, X. T., Executive Planning with BASIC, Berkeley, Sybex, 1982.

Bui, X.T., "Building Effective Multiple Criteria Decision Making: A Decision Support System Approach," Systems, Objectives, Solutions, v. 4, No. 1, pp. 3-16, 1984.

Bui, X.T., Co-oP, A Decision Support System For Cooperative Multiple Criteria Decision Making, Ph.D Dissertation, New York University Graduate School of Business Administration, October 1985.

Bui, X. T., "NAI -- A Consensus Seeking Algorithm For Group Decision Support Systems", Proceedings of the 1985 IEEE International Conference on Systems, Man, and Cybernetics, Tucson, Arizona, pp. 380-384, November, 1985.

Bui, X. T. and Jarke, M., A Datalogical Model for Multiple Criteria Decision Making, working paper, Naval Postgraduate School, Monterey, California, 1985.

Bui T. X. and Jarke M., "Non-Cooperation in Group Decision Support Systems: Many Problems and Some Solutions," working paper, 1986.

Bui, X.T., and Jarke, M., "Communications Design for Co-oP: A Group Decision Support System," ACM Transactions on Office Information Systems, v. 4, No. 2, April 1986.

Camerer, C., "General Conditions for the Success of Bootstrapping Models", Organizational Behavior and Human Performance, v. 27, pp. 411-421, 1981.

Cook, W. D. and Seiford, L. M., "On the Borda-Kendall Consensus Method For Priority Ranking Problems," Management Science, v. 28, No. 6, pp. 621-637, June 1982.

Dawes, R.M. and Corrigan, B., "Linear Models in Decision Making", Psychological Bulletin, v. 81, pp. 180-188, 1974.

Delbecq, A. L., "The World Within the 'Span of Control': Managerial Behavior in Groups of Varied Size", Business Horizons, August 1968.

DeSanctis, G. and Gallupe, F., "Group Decision Support Systems: A New Frontier," Database, No. 2, pp. 3-11, Winter 1985.

Deutsch, M., The Resolution of Conflict, New Haven, Conn., Yale University Press, 1973.

Douglas, A., Industrial Peace Making, New York, Columbia Press, 1962.

Fisher, B.A., Small Group Decision Making, 2d ed., McGraw-Hill Book Co., 1980.

Gray, J., "The Transaction Concept--Virtues and Limitations," Proceedings of the 7th Very Large Data Base Conference, Cannes, pp. 144-154, 1981.

Gray, P., "The SMU Decision Room Project", Transactions on the First International Conference on Decision Support Systems, pp. 122-129, Atlanta, June 1981.

Hackathorn, R. and Keen, P., "Organizational Strategies for Personal Computing in Decision Support Systems," MIS Quarterly, v. 5, No. 3, pp. 21-27, September 1981.

Hoffman, P.J., "The Paramorphic Representation of Clinical Judgement," Psychological Bulletin, v. 57, pp. 116-131, 1960.

Huber, G., "Issues in the Design of Group Decision Support Systems," MIS Quarterly, v. 8, no. 3, pp. 195-205, September 1984.

ISO, "Data Processing -- Open Systems Interconnection Model," International Organization for Standardization, ISO/TC97/SC16-N890, 1982.

Jarke, M., Knowledge Sharing and Negotiation Support in Multiperson Decision Support Systems, working paper, Graduate School of Business Administration, New York University, New York, May 1985.

Jarke, M., "Knowledge Sharing and Negotiation Support in Multiperson Decision Support Systems," Decision Support Systems, v. 2, No. 1, pp. 93-102, 1986.

Jarke, M., Bui, T. and Jelassi, M., "Micro-mainframe DSS for Remote Multiple User Decision Making," Proceedings NYU Symposium on Integrating Systems for End Users, New York University, New York, May 1985.

Jarke, M., Jelassi, T. and Shakun, M., "Mediator: Towards a Negotiation Support System," to appear in M. F. Shakun, Evolutionary Systems Design--Policy Making Under Uncertainty, Holden Day, 1985.

Kahan, J.P. and Rapoport, A., Theories of Coalition Formation, Lawrence Erlbaum Associates, Inc., 1984.

Kochan, T. A. and Jick, T., "The Public Sector Mediation Process: A Theory and Emperical Examination," Journal of Conflict Resolution, v. 22, pp. 209-240, 1978.

Kressel, K., "Kissinger in the Middle East: An Exploratory Analysis of Role Strain in International Negotiation," in Dynamics of Third Party Intervention: Kissinger in the Middle East, ed. J. Z. Rubin, New York: Praeger Press, 1981.

London School of Economics and Political Science Report 83-5, MAUD--An Interactive Computer Program for the Structuring, Decomposition and Recomposition of Preferences Between Multiattributed Alternatives, by P.C. Humphreys and A. Wisudha, 1983.

Magic7 Software Co., Expert87™, copyright (c) 1986, 101 First Street, Suite 237, Los Altos, California, April 1986

Morley, L. and Stephesen, J., The Social Psychology of Bargaining, London; Allen and Unwin, 1977. 1970.

Pruitt, D. G., Negotiation Behavior, New York, Academic Press, 1981.

Puzman, J. and Porizek, R., Communication Control in Computer Networks, John Wiley and Sons, 1980.

Rathwell, M. A. and Burns, A., "Information Systems Support for Group Planning and Decision Making Activities," MIS Quarterly, v. 9, no. 3, pp. 255-271, Seeptember 1985.

Rubin, J. R. and Brown, B. R., The Social Psychology of Bargaining and Negotiation, New York, Academic Press, 1975.

Saaty, T., The Analytic Hierarchy Process: Planning, Priority, Allocation, McGraw-Hill Book Co., 1980.

Shakun, M., "Decision Support Systems for Negotiations," Proceedings of the 1985 IEEE International Conference on Systems, Man, and Cybernetics, Tucson, Arizona, pp. 380-384, November, 1985.

Simon, H. A., The New Science of Management Decision, Harper and Row, Publishers, 1960.

Simon, H.A., Administrative Behavior, p. 125, New York: MacMillan, 1961.

Sprague, Jr., R. H. and Carlson, E. D., Building Effective Decision Support Systems, Prentice Hall, Inc., 1982.

Stohr, E. A., "DSS for Cooperative Decision Making," Proceedings NATO Conference on Data Base and Decision Support Systems, Amsterdam, North-Holland, 1981.

Swap, W.C., and others, Group Decision Making, pp. 295-298, 305, 309, Sage Publications, Inc., 1984.

Van de Ven, A.H., Group Decision Making and Effectiveness, pp. 93-105, Graduate School of Business, Kent State University, 1974.

von Neumann, J. and Morgenstern, O., Theory of Games and Economic Behavior, 3rd ed., Princeton, New Jersey, Princeton University Press, 1953.

Wright, G. and others, Behavioral Decision Making, pp. 235-256, Plenum Press: New York, 1985.

Zaleznik, A., and Moment, D., The Dynamics of Interpersonal Behavior, Wiley and Sons, 1964.

Zander, A., Making Groups Effective, Jossey-Bass, Inc., Publishers, 1982.

Zeleny, M., Multiple Criteria Decision Making, McGraw-Hill Book Co., 1982.

BIBLIOGRAPHY

Fogg, R. W., "Dealing with Conflict: A Repertoire of Creative, Peaceful Approaches," Journal of Conflict Resolution, v. 29, No. 2, pp. 330-358, June 1985.

Grandori, A., "A Prescriptive Contingency View of Organizational Decision Making," Administrative Science Quarterly, v. 29, pp. 192-209, June 1984.

Keeney, R. L., "A Group Preference Axiomatization with Cardinal Utility," Management Science, v. 23, No. 2, pp. 140-145, October 1976.

Keeney, R. L., and Kirkwood, C. W., "Group Decision Making Using Cardinal Social Welfare Functions," Management Science, v. 22, No. 4, pp. 430-437, December 1975.

Kersten, E. G., "NEGO - Group Decision Support System," Information and Management, v. 3, No. 8, pp. 237-246, May 1985.

Lax, D. A., "Optimal Search in Negotiation Analysis," Journal of Conflict Resolution," v. 29, No. 3, pp. 456-472, September 1985.

Levy, M. A., "Mediation of Prisoners' Dilemma Conflicts and the Importance of the Cooperation Threshold," Journal of Conflict Resolution, v. 29, No. 4, pp. 581-603, December 1985.

Lichbach, M. I., and Gurr, T. R., "The Conflict Process," Journal of Conflict Resolution," v. 25, No. 1, pp. 3-29, March, 1981.

Lockhart, C., "Conflict Actions and Outcomes: Long-term Impacts," Journal of Conflict Resolution, v. 22, No. 4, pp. 565-578, December 1978.

Shenoy, P. P., "On Committee Decision Making: A Game Theoretical Approach," Management Science, v. 26, No. 4, pp. 387-399, April 1980.

Steeb, P., and Johnston, S.C., "A Computer-Based Interactive System for Group Decisionmaking," IEEE Transactions on Systems, Man, and Cybernetics, v. SMC-11, No. 8, pp. 544-552, August 1981.

Wall, J. A., Jr., "Mediation: An Analysis, Review, and Proposed Research," Journal of Conflict Resolution, v. 25, No. 1, pp. 157-180, March 1985.

Wendell, R. E., "Multiple Objective Mathematical Programming with Respect to Multiple Decision-Makers," Operations Research, v. 28, No. 5, pp. 1100-1111, September-October 1980.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22304-6145	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93943-5002	2
3.	Dr. Tung X. Bui, Code 54BD Department of Administrative Sciences Naval Postgraduate School Monterey, California 93943-5000	3
4.	Dr. Carson K. Eoyang, Code 54CE Department of Administrative Sciences Naval Postgraduate School Monterey, California 93943-5000	1
5.	Dr. T. R. Sivasankan, Code 54SV Department of Administrative Sciences Naval Postgraduate School Monterey, California 93943-5000	2
6.	Lcdr A. Kardos Commander Third Fleet Pearl Harbor, Hawaii 96860-7500	7
7.	Lt E. Kutz 3. Schnellbootgeschwader Swinemuender Str. 26 2390 Flensburg West Germany	7
8.	Computer Technology Programs, Code 37 Naval Postgraduate School Monterey, California 93943-5000	1

END

2-81

DTIC