Controlling Bias in User Assertions in Expert Decision Support Systems for Problem Formulation

DAVID B. PARADICE and JAMES F. COURTNEY, JR.

David Paradice is an Assistant Professor of Management Information Systems (MIS) at Texas A & M University. He received a Ph.D. in MIS from Texas Tech University in 1986. He has a B.S. degree in Information and Computer Science and an M.S. degree in Industrial Management from Georgia Tech. Prior to entering Texas Tech, he worked as a programmer/analyst for Energy Management Associates, Inc., in Atlanta. His research interests include decision support systems, model management, and expert systems.

James F. Courtney, Jr., is Professor of Management Information Systems in the College of Business Administration at Texas A & M University. He received his Ph.D. in Business Administration (Management Science) from the University of Texas at Austin in 1974. His experience includes positions as Information Systems Analyst with MRI Systems Corporation, Visiting Research Scientist at the NASA Johnson Space Center in Houston, Assistant Professor of Management Science at Georgia Tech, Project Director at Georgia Tech's Engineering Experiment Station, Visiting Professor at the State University of New York at Buffalo, and Professor of Management Information Systems at Texas Tech University. His papers have appeared in several journals, including *Management Science, Communications of the ACM, MIS Quarterly, IEEE Transactions on Systems, Man, and Cybernetics, Database, Socio-Economic Planning Sciences, Applied Systems Analysis, Journal of Bank Research, Decision Sciences, Interfaces,* and the *Journal of Experiential Learning and Simulation*. He is the co-developer of the Systems Laboratory for Information Management (Business Publications, Inc., 1981), a software system to support research and education in management information systems. His present research interests are in management information systems, especially decision support systems, and database management.

ABSTRACT: Information on cause–effect relationships among variables in the problem domain is one type of knowledge required in expert decision support systems for problem formulation. This knowledge must be acquired from "expert" managers and stored in the system's knowledge base. Unfortunately even experienced managers may be biased in their beliefs about cause–effect relationships. We present a system which uses causal modeling, path analysis, and an historical database to statistically verify asserted relationships as they are entered into the system. Since it

An earlier version of this paper was presented at the Nineteenth Hawaii International Conference on System Sciences, Honolulu, Hawaii, January 1986.

is possible that a valid assertion is not statistically supported, the user has the option to insert a relationship into the knowledge base even though the analysis may not indicate statistical validity. Information on rejected relationships is maintained in a "rejection base" which is used later to retest assertions whose validity may have changed due to updates to the database. The intent is to provide a system which helps the user learn, in an unbiased manner, about the true nature of causal relationships in the problem domain.

KEY WORDS AND PHRASES: Decision support systems, expert systems, problem formulation.

DECISION MAKING is the essence of management. Accurate formulation of the decision problem is perhaps the most critical aspect of decision making. By problem formulation, we refer to the process of determining what the elements or variables of the problem are, how these variables are interrelated, what the objectives and constraints are, and what measurement techniques should be used.

The objective of the paper is to report on the development of a decision support system which is intended to help users overcome biases in the formulation of business problems. We discuss, in order, human biases which may have an influence on problem formulation, a Kantian basis for inquiring systems models, and our system to support unbiased problem formulation, which integrates concepts from decision support systems, causal modeling, path analysis, database systems, and expert systems.

Human Biases in Problem Formulation

WHILE MUCH RESEARCH has been done in the decision support systems (DSS) area, very little of it has been directed at supporting the process of formulating the proper structure of the problem at hand. The process of problem formulation is characterized by hypothesizing relationships in domains of variables in which the relationships are uncertain, temporal, and dynamic. Problem formulation is itself clearly an unstructured process. As such, it is a fertile area for research which sheds new light on the process. The need for research such as this was recognized a decade ago by Leavitt [22] and more recently by Lyles and Mitroff [24]. Further, it seems reasonable to propose that a DSS should support the formulation aspect of the decision-making process [21].

Mintzberg, Raisinghani, and Theoret [28] note that little is known about the problem formulation phase of the decision-making process, although much of the prior research in the area would appear to indicate that most humans are not particularly adept at it. Problem formulation has been found to be a function of which critical stimulus affects the problem solver first [19]—the system that undertakes the process of formulating the problem [1, 35], the perceived deviation from expectations [32], the environment of the problem space [31], the ability to decompose complex, unfamiliar situations into simpler, more familiar situations [31, 35], the creativity of the person formulating the problem [40], or the experience of the person formulating the problem [36]. These coping strategies occur during the

Intelligence and Design stages of Simon's [38] model of decision making.

Mintzberg, Raisinghani, and Theoret [28, p. 274] refer to "problem diagnosis" as "probably the single most important routine, since it determines in large part, however implicitly, the subsequent course of action." Mitroff, Emshoff, and Kilmann [30] state that, in typical organizational environments, problem forming and defining are as important, if not more so, than problem solving. However, due to the cognitive limitations of human beings, coping strategies such as those mentioned above are not fail-safe, or in many cases even rational [37, 41, 42]. DSS seek to support decision makers by obviating problems which result from the limitations of human cognitive processes.

In order to compensate for their cognitive limitations and biases, decision makers have turned increasingly to the use of models. Research by Kasper [20] has demonstrated that model development and use significantly improve the decision-making process. Kasper found that subjects who developed models consistently outperformed nonmodelers in general and that those who used their models further outperformed those who developed a model but then chose not to use it. This may well be due to the models' insensitivity to the biases mentioned above and to the fact that factors such as fatigue or distractions in the environment are not an issue.

More recently, research has addressed the importance of formulating the correct problem [43]. Pracht [33] and Loy [23], for example, have researched the use of graphical structural modeling tools to support the problem formulation process. McLean and Shepherd [26, p. 41] note that ". . . the derivation of model structure is analogous to the framing of hypotheses" and ". . . these models should be scientifically tested."

Indeed, since prior research has not incorporated any means for analyzing the models developed, it is possible that the subjects in prior experiments have been solving the wrong problem. Solving the wrong problem is known as committing an "error of the third kind" [34]. Mitroff and Betz [29] have suggested that an error of the third kind is of fundamental importance and that it outweighs the usual statistical errors of the first and second kinds. Solving the wrong problem can be an expensive error [44].

Computer-based systems have no inherent cognitive limitations. Nor do they exhibit the biases which characterize human decision making. Ata [4] has recently taken a step toward the development of a DSS that would assume more of the burden of the problem formulation process. Ata extended research by Bouwman [7] in which relationships between variables were maintained in the form of "causation trees." The emphasis of Ata's work was on explicit specification of relationships and subjective estimates of the strength of these relationships. After a user (i.e, manager) had supplied this type of information, the prototype system developed by Ata would then seek to determine if changes in the status of one variable in the problem domain could be adequately "explained" by changes in other variables related to the first variable via the causation tree structure.

Where Ata's prototype was able to resolve the reason for changes in variables, a "diagnosis" was found. In cases where the causation tree did not lead to an adequate diagnosis of the change in a variable, the prototype would attempt to create new links between nodes in the causation tree that would provide the information needed to explain the change in the variable of interest.

Unfortunately, Ata's prototype depended on the manager to input correct models. If a model were poorly specified, the prototype would not produce an adequate diagnosis. While the prototype was able to give an indication of the parts of the model that were deficient, it seems desirable to include some type of analysis of the model as it is being constructed.

Theoretical Basis and Conceptual Model

WE VIEW PROBLEM FORMULATION as a specialized form of inquiry. Hence, theoretical models of inquiry are examined to determine which best suits the situation of problem formulation. Churchman [11] distinguishes five archetypal ways of modeling and generating evidence for any problem: Lockean, Leibnitzian, Kantian, Hegelian, and Singerian. In particular, Kantian inquiring systems make explicit the strong interaction between scientific theory and data. Such systems function by assuming that before collecting data on a problem a posterori one always had to presuppose the existence of some scientific theory a priori, no matter how implicit and informal that theory might be [25, p. 481]. Since a Kantian inquiring system is neither purely theoretical nor experimental, the final information content is a function of both. Mason and Mitroff [25, p. 482] suggest that Kantian inquiring systems may be best suited for handling problems of "moderate" ill-structure.

A Kantian inquirer requires three components: (1) a space–time coordinate system, (2) a way of recognizing inputs, and (3) a set of "categories" in which to classify the inputs [11]. The coordinate system may be conceptualized as the position of a firm at a point in time relative to its goals, while the inputs to the system are periodic evaluations of the status of the firm. For example, quarterly reports used by management give an indication of the current position of the firm and are acceptable as inputs. For categories, we will substitute potential models. That is, we will provide linear and higher order models which will be used to categorize our inputs.

The conceptual model for the architecture of the system is derived from the work of Blum [5, 6] and is presented in Figure 1. While conceptually similar to Blum's, our approach differs vastly in an operational sense. Experiential knowledge has moved from an ancillary process to assume a more central, and controlling, role in order to navigate the much larger search space of a business problem domain. Additionally, Blum's "experiential" knowledge comes from the medical literature, which is almost invariant over time and patient. The experiential knowledge required for the business domain varies over time and "patient" (i.e., firm).

The components in Figure 1 represent major aspects of the current version of the system. Hypothesized relationships are formed by a manager based on his or her experience. These hypotheses are then tested statistically by drawing data from a corporate database. The results of the tests are presented to the manager, who then makes a decision on whether these results should be

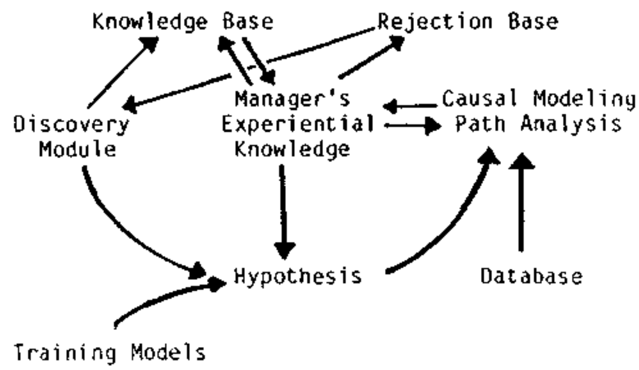

Figure 1. Conceptual Schema of the Problem Formulation System

incorporated into the knowledge base.

Later these results are manipulated via causal modeling and path analysis to analyze more complicated relationships. Eventually, a discovery module [5, 6] will be incorporated which will assume more of the burden for generating viable hypotheses.

The experiential knowledge base initially contains the biases of the user who trains it. Over time, the system statistically tests these assertions. Statistical evidence is presented to the user, who may call for alterations. In this way the system and the user "learn" together and work toward unbiased formulation of the problem. The manager's experiential knowledge must, however, guide all phases of the system's action. Hypotheses generated by the discovery module must be authorized by the experiential knowledge component. Only the most promising hypotheses are selected for examination. Finally, modifications to the knowledge base must be authorized by the experiential knowledge component.

System Implementation

WE DISCUSS ON-GOING research to refine the theoretical methodology embodied in Figure 1. To implement this Kantian approach, an existing DSS has been enhanced to include commands that allow a user to propose and test relationships between the data items in a database. To begin, linear modeling capabilities have been implemented. First, existing software used in the research is reviewed in order to provide a perspective in which to discuss enhancements. Then two of the currently available commands are described.

Business Management Laboratory (BML)

The Business Management Laboratory (BML) [18] is a widely used [14, 17, 20, 23, 33], readily available software package which simulates a variety of decision-making functions of the management of a manufacturing firm. Decisions to be made include setting product price levels, production levels, number of sales representatives, advertising expenses, and raw materials purchased. BML may simulate a firm marketing one or two products in one or two market areas, with up to eight firms competing in any one industry.

BML generates a database which has been used to test the software developed in this research. The database provides historical data for developing relationships. The simulation provides a means for analyzing suggested courses of action under a degree of uncertainty. Use of this software allows for experimental control, since it is possible to determine the degree of complexity of the decisions to be made by varying the parameters of the system. Consequently, BML provides an excellent basis for research which is cumulative and replicable [12].

Systems Laboratory for Information Management (SLIM)

The Systems Laboratory for Information Management (SLIM), developed by Courtney and Jensen [13], provides support for the decision maker in a BML research experiment. SLIM's basic capability is to provide a query language and modeling interface to a database which is generated by the BML. The basic version of SLIM is a data-oriented system [2]. It provides a database, a database administration system, and an interactive query facility. Kasper [20] has added statistical routines to develop a more model-oriented DSS.

Software Design

Many new commands and procedures have been added to the SLIM software which implement the procedures required to perform hypothesis generation, causal modeling, path analysis, results interpretation, knowledge acquisition, and knowledge usage. For brevity, only two of the commands are discussed. These are sufficient to give a feel for the system.

The CAUSE command is used to specify the variables which will compose a causal model. An example of the command is

CAUSE SALES, PRICE, ADVERTISING;

In this example, SALES is the database item "sales volume," PRICE is "price," and ADVERTISING is "advertising expense." The first variable listed represents the endogenous variable in the analysis. All other variables are taken to be exogenous variables. Hence, in this example it is postulated that price and advertising expense affect sales volume. This corresponds to the causal model diagram (or "structural model") shown in Figure 2.

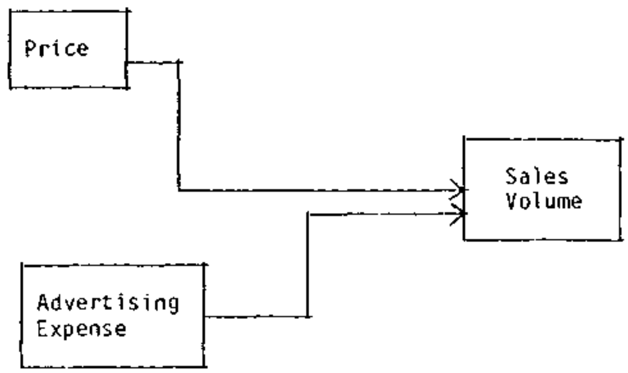

Figure 2. A Simple Causal Model

Execution of this command invokes a stepwise regression routine which fits the best model possible (based on Mallows's Cp criterion, which may be interpreted in terms of parameter bias and precision.). Path analysis requires a no-intercept model, since there is no interpretation for the intercept term. The path coefficient for the residual term (not shown in Figure 2) is $\sqrt{(1 - R^2)}$, where R^2 is the square of the multiple correlation coefficient and is commonly referred to as the fraction or proportion of unexplained variance. Since standardized variables have a variance of one, the general expression $\sqrt{(1 - R^2)}$ is simply the square root of the unexplained variation in the dependent variable in question. Asher [3] demonstrates that, for a "direct influence" case such as this (i.e., there are no intervening variables between the exogenous and endogenous variables), this analysis yields the proper path estimates. These path coefficients are calculated and presented to the user, along with a brief explanation of their meaning.

As Tversky and Kahneman [42] have noted, the way this information is presented may bias the subsequent choice of action taken by the user. Hence, care must be exercised in the development of this aspect of the user interface. This interface currently presents a qualitative analysis of the coefficient estimates. Since the path coefficients are standardized regression coefficients, statements can be made about a unit change in one standardized variable producing a given amount (equal to the regression coefficient) of change in the other standardized variable. Since typical users do not think in terms of standardized values, the software converts back to unstandardized values automatically.

The results are presented to the user in terms of pairwise relationships (Figure 3), where the user is given a chance to input a level of agreement with the statistical findings (on a scale of one to ten); the amount of time lag (if any) between a change in the independent variable impacting the dependent variable; whether the independent variable acts to constrain the dependent variable; and whether a certain level of the independent variable "guarantees" a specific level of the dependent variable. We

```
...
CAUSE SALES,PRICE,ADVERTISING;

THE VARIABLES YOU HAVE SPECIFIED ACCOUNT FOR 89.9506 PER CENT OF THE
VARIANCE IN SALES.

THE FOLLOWING RELATIONSHIPS HAVE BEEN FOUND REGARDING THE VARIABLE SALES:

PRICE IS STATISTICALLY SIGNIFICANT.
A ONE UNIT INCREASE IN PRICE RESULTS IN A -0.5285 DECREASE IN SALES.

SHOULD THIS RELATIONSHIP BE STORED FOR FUTURE REFERENCE?  YES

ON A SCALE FROM 1 TO 10 WHERE 10 INDICATES STRONG AGREEMENT, HOW DO
YOU AGREE WITH THIS?  9

HOW MANY QUARTERS PASS BEFORE PRICE AFFECTS SV
(ENTER 0 FOR AFFECTS IN SAME QUARTER)?  0
...
```

Figure 3. Example of the Current State of the Causal Modeling Software

also capture information at this time about whether the relationship between the two variables holds only under certain conditions, such as "only when operating at full capacity." Finally we request a statement from the user describing the relationship, such as the idea that "increases in advertising expense generally increase sales."

In all cases, the user determines whether the relationship and the user-supplied data will be written to a file. This file could be considered a "knowledge base" of the relationships embodied in the BML game.

Relationships which fail to be included in the knowledge base are recorded in a "rejection base." When a relationship is included in the rejection base, the system then knows that a statistically significant relationship was not found between these variables, although it may remain a fruitful area for investigation.

The rejection base can be used for many purposes. It serves as a source of potential relationships; it can be used to avoid repetitive examination of pairwise relationships which have been rejected; and it can be analyzed to determine the biases of a manager using the system. Since the number of possible relationships grows exponentially with the number of variables, the problem of finding all paths in the knowledge base becomes computationally intractable without the use of heuristics to direct the search. The rejection base provides one set of heuristics. To our knowledge, this is the first attempt to do something constructive with facts that do not warrant inclusion in a knowledge base.

Path analysis is initiated by executing a PATH command (Figure 4). This command searches the knowledge base created by executions of the CAUSE command for existence of indirect relationships between variables. It is used to determine whether "missing links" exist between variables by determining whether a direct link between the beginning and ending variables in the path explains more of the variance than the indirect paths. If so, it may be desirable to update the knowledge base to include the new link.

The PATH command allows the user to specify the relationship he wishes to

```
...
PATH ADVERTISING, REVENUE;

ADVERTISING → SALES → REVENUE
CONTRIBUTION:    0.357876

ADVERTISING → MARKETING COSTS → REVENUE
CONTRIBUTION:    0.226548

TOTAL CONTRIBUTION OF INDIRECT PATHS IS      0.584424
...
```

Figure 4. Example of the Current State of the Path Analysis Software

analyze in detail. All paths between the specified variables are created using a depth-first, backtracking algorithm. In the example, the system responds by presenting all paths which currently exist between ADVERTISING and REVENUE, along with the indirect path "contribution" as defined in Asher [3]. Detailed analysis of a path seeks to resolve the question of whether the indirect paths from the first variable to the last variable in the selected path indicate the need to include a direct path between these variables.

Again following Asher [3], it can be shown that, if all indirect paths which exist have been included in the knowledge base, then the sum of products of the path coefficients for the indirect paths should equal the standardized regression coefficient estimate of the first variable regressed on the last (the correlation). Accordingly, the software examines this hypothesis and reports the results to the user.

If all relevant indirect paths have been specified, then there is no need to incorporate the direct relationship between the first and last variable into the knowledge base. If the indirect relationships do not account for as much information as the direct relationship, the user is asked whether to include this new relationship into the knowledge base.

To summarize briefly, the system currently has the capability of helping to overcome user biases in two ways: by (1) statistically testing asserted relationships between variables and allowing the user to modify assertions; and (2) searching for missing links between beginning and ending variables in paths of arbitrary length and allowing the user to include direct links in the knowledge base, where appropriate.

Future Research

FUTURE RESEARCH will focus on continuing the development of the hypothesis generation and testing procedure outlined above and on investigating means for refining the manager's experiential knowledge component.

Several aspects of the path anlysis need to be resolved. The software does not currently recognize cyclical paths in the knowledge base. The methodology also does not allow reciprocal relationships (A affects B affects A), forcing the user to choose which relationship to include.

One issue requiring substantial further investigation regards determining "how close is close" when evaluating whether the sum of the indirect path contributions "equals" the correlation between two variables. In the case where there are three variables A, B, and C, in a model such that A affects B and B affects C, it can be shown [3] that the test for determining whether a path from A to C should be included is equivalent to a test of whether the partial correlation of A and C controlling B is zero.

In more complicated cases, this approach quickly becomes intractable due to problems caused by multicollinearity. An indication of multicollinearity occurs when the sum of the indirect paths between two variables sums to a value greater than one. In this case, one is in the position of trying to explain how the exogenous variables account for more than 100% of the variance in the endogenous variable. When this occurs, it is necessary to rely on the judgment of the user. However, Burns and Winstead [8, 9] demonstrate a geometric approach to determining the redundancy of paths in a model, based on the "excitatory" or "inhibitory" nature of the paths. This approach is being examined in light of the needs of this research.

Once the system is fully capable of executing path analysis, it can begin to acquire knowledge. The system is "boot-strapped" through a training phase which examines a set of structural models created in prior research by Pracht [33]. These models contain the biases of their developers. The ability of the system to remove these biases will be tested by comparing the subjects' models and the system-modified models to the actual parameters in the BML code. Thus the BML program, a highly unbiased guarantor, is used to test the system.

Completion of the system training phase results in a system with knowledge about the problem domain, but little knowledge regarding how to use these relationships. This difference has been referred to as "what" versus "how to" knowledge [10, 15, 27]. The next step after system training will be to add the "surface" knowledge that is required to use the knowledge base effectively. This knowledge will be embodied in a combination of mathematical models and heuristics developed by the researchers. The deep knowledge currently envisioned will take the form of heuristics regarding the use of statistical and linear programming models (following [39]). For example, a user may believe strongly that a relationship exists between two variables which is not verified by prior analysis. The system will take the variables specified by the user and attempt to arrive at a reasonable relationship by guiding the user through transformations of the data and/or the fitting of higher order relationships. In cases where the system has established relationships between variables, the system will be capable of supporting the following scenario:

> A manager seeks guidance regarding a course of action involving a particular variable, say, sales volume. He wishes to know how sales volume may be increased. He asks the system to suggest a course of action which will result in increased sales volume. The system examines its knowledge base for relevant relationships, creates and executes a mathematical model, and suggests a course of action based on the results generated by the model.

Summary

Our research specifically addresses the issue of problem formulation in a business context with an emphasis on formulating the correct problem. A Kantian system is designed that uses causal modeling and path analysis to mitigate the cognitive biases exhibited by human decision makers.

Dss theory indicates that a DSS should support all aspects of the decision-making process. With the exception of very recent research by Pracht [33] and Loy [23], research examining problem formulation and DSS design has been limited. This study extends research in this area.

The development of the DSS enhancements clearly demonstrates the feasibility of incorporating interactive statistical tools aimed at taking an active role in supporting problem formulation. Insight will be gained into what forms of knowledge representation and organization are feasible in the domain of business problem formulation. The training stage of the study will yield new knowledge regarding the design of knowledge acquisition systems. Each of these aspects is in need of research that extends current abilities to deliver a system of practical value.

Finally, the enhanced DSS provides a foundation upon which to pursue further research aimed at incorporating more "intelligence" into business software. For example, once the "learning" capabilities are in place, interesting questions regarding the machine's error rate during the learning process can be examined. This could then lead to research examining a machine's capability to employ sampling theory and optimal stopping rules in order to pace its learning process, or even a machine's possible ability to "choose" to ignore a user's input if the system's optimal rule indicates that such action is in the system's best interest [16].

References

1. Ackoff, Russell L. The future of operational research is past. *Journal of the Operations Research Society*, 20, 2 (February 1979), 94–104.

2. Alter, S. A. A taxonomy of decision support systems. *Sloan Management Review*, 19, 1, (1977), 39–56.

3. Asher, Herbert B. *Causal Modeling*. Beverly Hills, CA: Sage Publications, 1983.

4. Ata, Nassar. Design and implementation of an expert system for problem recognition and diagnosis. Unpublished DBA dissertation, Texas Tech University, 1985.

5. Blum, Robert L. Discovery, confirmation, and incorporation of causal relationships from a large time-oriented clinical data base: the RX Project. *Computers and Biomedical Research*, 15 (1982), 164–187.

6. Blum, Robert L. Modeling and encoding clinical causal relationships from a medical knowledge base. *Seventh Annual Symposium on Computers in Medical Care* (1983), 837–841.

7. Bouwman, Marinus J. Human diagnostic reasoning by computer: an illustration from financial analysis. *Management Science*, 29, 6 (1983), 653–672.

8. Burns, J. R., and Winstead, Wayland H. An input/output approach to the structural analysis of digraphs. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-12, 1 (January/February, 1982a), 15–24.

9. Burns, J. R., and Winstead, Wayland H. Input and output redundancy. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-12, 6 (November/December, 1982b), 785–793.

10. Chandrasekaran, B., and Mittal, S. Deep versus compiled knowledge approaches to diagnostic problem-solving. *International Journal of Man–Machine Studies*, 19 (1983), 425–436.

11. Churchman, C. West. *The Design of Inquiring Systems*. New York: Basic Books, Inc., 1971.

12. Courtney, James F.; De Sanctis, G.; and Kasper, G. M. Continuity in MIS/DSS laboratory research: the case of a common gaming simulator. *Decision Sciences*, 14, 3 (1983), 419–439.

13. Courtney, James F., and Jensen, R. L. *SLIM User's Manual*. Dallas: Business Publications, Inc. 1981.

14. DeSanctis, G. An examination of an expectancy theory model of decision support system use. Unpublished DBA dissertation, Texas Tech University, 1982.

15. Hart, P. E. Direction for AI in the eighties. *SIGART Newsletter*, 79 (1982).

16. Hora, Stephen C. Learning rates in supervised and unsupervised intelligent systems. Paper presented at the Workshop on Artificial Intelligence and Statistics, AT&T Bell Laboratories, Murray Hill, N. J., April 1985.

17. Hurts, C. Decision making in a dynamic environment: the effects of problem representation and outcome feedback. Unpublished DBA dissertation, Texas Tech University, 1985.

18. Jensen, R. L., and Cherrington, M. L. *BML Participant's Manual*. Dallas: Business Publications, Inc., 1977.

19. Judson, Abe J., and Cofer, Charles N. Reasoning as an associative process: I. Direction in a simple verbal problem. *Psychological Reports*, 2 (1956), 469–473.

20. Kasper, George M. The effect of user-developed DSS applications on forecasting decision-making performance in an experimental setting. *Journal of Management Information Systems*, 2, 2 (Fall 1985), 26–39.

21. Konsynski, Benn, and Dolk, Dan. Knowledge abstractions in model management. In Dickson, Gary W., ed. *DSS-82 Transactions*, 1982.

22. Leavitt, Harold J. Beyond the analytic manager: parts I and II. *California Management Review*, 17, 3 and 4 (Spring 1975), 5–12; (Summer 1975), 11–21.

23. Loy, Steve. An experimental investigation of the effects of a graphical interactive problem-structuring aid on small group decision-making performance. Unpublished DBA dissertation, Texas Tech University, 1986.

24. Lyles, Marjorie A., and Mitroff, Ian I. Organizational problem formulation: an empirical study. *Administrative Science Quarterly*, 25 (March 1980), 102–119.

25. Mason, Richard O., and Mitroff, Ian I. A program for research on management information systems. *Management Science*, 19, 5 (January 1973), 475–487.

26. McLean, Mick, and Shepherd, Paul. The importance of model structure. *Futures* (February, 1976), 40–51.

27. Michie, Donald. High-road and low-road programs. *AI Magazine*, 3 (1982).

28. Mintzberg, Henry; Raisinghani, Duru; and Theoret, Andre. The structure of "unstructured" decision processes. *Administrative Science Quarterly*, 21 (June 1976), 246–275.

29. Mitroff, Ian I., and Betz, Frederick. Dialectical decision theory: a meta-theory of decision making. *Management Science*, 19, 1 (September 1972), 11–24.

30. Mitroff, Ian I.; Emshoff, J. R.; and Kilmann, R. H. Assumptional analysis: a methodology for strategic problem solving. *Management Science*, 25, 6 (1979), 538–593.

31. Newell, Allen, and Simon, Herbert A. *Human Problem Solving*. Englewood Cliffs, N. J.: Prentice-Hall, 1972.

32. Pounds, W. F. The process of problem finding. *Industrial Management Review*, 11, 1 (Fall 1969), 1–19.

33. Pracht, William E. An experimental investigation of a graphical interactive structural modeling aid for decision support systems. Unpublished DBA dissertation, Texas Tech University, 1984.

34. Raiffa, Howard. *Decision Analysis*. Reading, MA: Addison-Wesley, 1968.

35. Reitman, Walter R. Heuristic decision procedures, open constraints, and the structure of ill-defined problems. In Shelly, Maynard W., II, and Bryan, Glenn L., eds. *Human Judgments and Optimality*: New York: John Wiley and Sons, 1964, 282–315.

36. Rowe, Alan J. How do senior managers make decisions? *Business and Economics* (Winter 1977), 17–20.

37. Sage, Andrew P. Behavioral and organizational considerations in the design of information systems and processes for planning and decision support. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-11, 9 (September 1982), 640–678.

38. Simon, H. A. *The New Science of Management Decisions*. New York: Harper and Row, 1960.

39. Stohr, Edward. Artificial intelligence support for LP formulation. Paper presented at the Model Management Workshop, University of Texas–Austin, May 16, 1985.

40. Taylor, Irving A. A retrospective view of creativity investigation. In Taylor, Irving A., and Getzels, J. W., eds. *Perspectives in Creativity*. Chicago: Aldine Publishing Company, 1975.

41. Tversky, Amos, and Kahneman, Daniel. Judgement under uncertainty: heuristics and biases. *Science*, 184 (September 1974), 1124–1131.

42. Tversky, Amos, and Kahneman, Daniel. The framing of decisions and the psychology of choice. *Science*, 211 (January 1981), 453–458.

43. Volkema, Roger J. Problem formulation in planning and design. *Management Science*, 29, 6 (1983), 639-652.

44. Yadav, Surya B., and Korukonda, Apparao. Management of type III error in problem identification. *Interfaces*, 15, 4 (July–August 1985), 55–61.