SIMULATION OF AN INDIVIDUAL MAKING DECISIONS UNDER UNCERTAINTY

Francis D. Tuggle, School of Business and Department of Computer Science

F. Hutton Barron, School of Business

Richard O. Day, Department of Computer Science

University of Kansas
Lawrence, Kansas 66044

Abstract

A computer simulation model, SIDIP (Simulation of Individual Decisions through Information Processing), of a person making nine decisions under uncertainty is sketched. Eight of a subject's (S's) choices are consistent with the Laplace or maximize expected value criteria and S's other is consistent with the Savage (minimax regret) criterion (see Luce and Raiffa, 1957). SIDIP implies that the subject does not use the conventional computational processes dictated by those criteria. SIDIP reproduces S's articulated choice behavior: inconsistent use of choice criteria, rejection of some alternatives, and eventual choice from the preferred alternatives. Analysis of information processing models of suboptimal decision behavior suggests operational techniques by which decision making can be improved.

I. Introduction

A. Setting

The problems of individual decision making have been classified by Luce and Raiffa (1957) as decision making under (1) certainty, (2) risk, (3) uncertainty, and (4) partial ignorance (a combination of risk and uncertainty). These four classes of decision problems are defined in terms of knowledge of the probability distribution over the states of nature, given the usual decision theoretic formulation (decision maker; choices, acts, or alternatives; states of nature; payoffs) of a decision situation. Thus, decision making under certainty is trivial from a decision theoretic point of view.

Normative decision theory prescribes choice for a given structure and classification (risk, uncertainty, or partial ignorance) by specifying a criterion of choice. Normative theory prescribes in the sense that, if the criterion and formulation are accepted, choice is unambiguous. There are several decision theories -- meaning that for certain conditions there are several "reasonable" choice criteria.

It is well known that people do not always behave in a manner consistent with various normative decision theories. The descriptive failures of normative criteria are documented elsewhere (e.g., under risk, MacCrimmon, 1969; under uncertainty, Tuggle, 1972; under partial ignorance, Barron, 1970). Since these experimental results were derived from laboratory studies using reasonably artificial problems, it is likely that actual decisions made daily by decision makers facing complex real-world problems would also exhibit inconsistencies with normative theory.

In this paper we choose to study decision making under uncertainty. We believe partial ignorance is a reasonable representation of real world decision problems; however, several proposed approaches for dealing with partial ignorance first reformulate the problem as decision making under uncertainty. (Those who subscribe to a subjective or personalistic theory of probability would convert partial ignorance to risk. We temporarily reject risk since in many problems the probabilities are, at best, vaguely known and the decision maker is unwilling to accept the probability estimates for decision making purposes.) Other possible approaches to partial ignorance include deciding "as if" it were an uncertainty situation or deciding "as if" it were risk (i.e., maximize expected value or expected utility), but first rejecting (or considering) alternatives based on uncertainty criteria. These approaches place heavy emphasis on decision-making under uncertainty.

B. Proposal

Our aim is to study individual human decision-making under uncertainty so as to learn how decision processes are used and how to introduce realistic modifications into a person's cognitive behavioral repertoire so that he makes an optimal decision. This paper puts heaviest weight on unravelling and simulating nonoptimal decision processes; later papers will address the second subgoal of internalizing different cognitive processes.

In order to ensure that our understanding of current (suboptimal) human decision-making processes is explicit, operational, and falsifiable, we have encoded our model as a computer simulation program. Our program, entitled SIDIP for Simulation of Individual Decision-making through Information Processing, is described in detail in Section IV and is tested and analyzed in Section V.

In order to have a framework in which to express our model, we chose the Information Processing System (IPS) approach of Newell and Simon (1972). Accordingly, our mode of operation is as follows: first, we take verbal protocols from a subject while he is making decisions (see Sections II and III). Second, we construct an IPS model of the subject's cognitive processes (see Section IV). Third, we examine, both qualitatively and empirically, the adequacy of the IPS model (see Section V). Fourth, we use an IPS model assumed to be validated and from it infer reasons why the subject did not comply with the set of normative decision processes (see Section VI, Part B). Fifth, knowing the subject's IPS, we suggest changes in his information processing to get conformity to normative theory (Section VI, Part B); and last, we suggest how actually to implement the modified processes (Section VI, Part B).

II. Method

A single subject (S) was enjoined to make decisions under uncertainty and to verbalize as much of his thought process as possible. His utterances were recorded on audio tape and later transcribed to paper. S's protocol is analyzed in the next section. A computer program (SIDIP) was written (see Section IV) to simulate the essential parts of S's decision-making behavior. Goodness-of-fit tests of SIDIP's behavior to S's behavior are performed in Section V. Action recommendations are made in Section VI.

The subject was faced by nine decision situations, which were sequentially presented to him. The nine decision situations are independent; S was not permitted to see the next situation until he had made a final choice on the previous one. After all nine choices had been made, a single situation was selected by the experimenter (E) to be played for real money. The entire session with S lasted about 50 minutes.

A. Decision Problems

The nine situations faced by S were tabulated as nine different payoff matrices for uncertain decisions. Table 1 exhibits the first matrix that S was given. (Copies of all instruments used in this work are available from the authors.) Each matrix contains eight rows, corresponding to the strategies or actions available to S, who had to select one of them. The four columns correspond to the possible states of nature that could occur. S did not know what process was to be used for generating the states of nature: uniform random, friendly (maximax), antagonistic (minimax), or some other process.

		States			
		ZEJ	XEQ	WUH	QUG
Decision Strategies	S_1	12	0	4	4
	S_2	2	7	6	5
	S_3	0	11	3	4
	S_4	4	6	6	4
	S_5	10	4	4	2
	S_6	4	5	4	5
	S_7	4	9	2	3
	S_8	8	2	6	6

Table 1: First Decision Situation

To prevent learning S was never told what state of nature occurred after his row choice.

Each of the nine decision matrices was constructed as follows: four of the eight rows are consistent with four major decision theoretic criteria -- maximax, maximin, expected value (in the Laplace sense), and minimax regret (see Luce and Raiffa, 1957). For example, in Table 1, row S_1, since it has a payoff of 12 (larger than all other payoffs in that table), corresponds to the maximax strategy. Row S_4 corresponds to the maximin strategy, row S_8 to the expected value strategy (again, assuming a uniform probability distribution), and row S_5 to the minimax regret strategy. The other four rows correspond to suboptimal choices: each of these rows is dominated by at least one of the four optimal rows. In Table 1, row S_1 dominates row S_3 ($12 > 11$, $4 = 4$, $4 > 3$, and $0 = 0$), S_4 dominates S_6 ($6 > 5$, $6 > 5$, $4 = 4$, and $4 = 4$), S_8 dominates S_2, and S_5 dominates S_7.

The order of appearance of the eight types of rows in each of the nine matrices was randomized as was the order of payoffs within each row. (An exception is the minimax regret row and its associated dominated row, since the regret calculation depends upon other payoffs that appear in the same column.) The order of presentation to S of the nine decision matrices was sequential: 1, 2, ..., 9. Matrices 1, 4, and 7 had all positive payoffs; matrices 2, 5, and 8 had both positive and negative payoffs; and matrices 3, 6, and 9 had all negative payoffs. Additionally, matrices 7, 8, and 9 had significantly larger numeric entries than matrices 1 through 6 (an average of 9.23 versus 5.29, respectively).

The subject could exhibit inconsistent choice behavior using these matrices (he could select a row corresponding to one decision criterion on one matrix and corresponding to a different decision criterion on another matrix) and/or suboptimal choice behavior (he could select a row dominated by another). Previous experimentation (Tuggle, 1972) shows that subjects (undergraduate and graduate students and practicing managers) exhibit both of these behaviors on these very problems. However, this S, in fact, makes no suboptimal choices and evidences only one inconsistency (see Section III). Yet, as Sections III, IV, and V will detail, his choice processes differ substantially from those dictated by decision theory. (See Luce and Raiffa, 1957, for a statement of normative decision theory.)

B. The Subject

The only subject studied in this paper was a male, first-year M.B.A. candidate at the University of Kansas who had not had courses in operations research or in decision theory. He was invited directly by one of the authors (FHB) to participate in an experiment on decision-making, and S was promised that he could either receive a flat payment of $2 for participation or gamble based on the decisions he would make. We informed S, if he chose to gamble, that after he had made his nine decisions, we would apply some unspecified generating processes to select one of the matrices and to select a state of nature. He did not have to announce whether he wanted to gamble or not until he had seen all nine tables and made his choices. (The subject did decide to gamble and won an additional $16.)

III. Protocol Analysis

Limits to the space available here prohibit us from presenting and analyzing S's entire protocol and Problem Behavior Graph (PBG), a time-ordered graph of S's verbal behaviors that are then to be simulated by the IPS (Newell and Simon, 1972). The complete protocol and PBG are available from the authors. Excerpts from S's protocol and his PBG are presented in Figure 1, primarily on payoff matrix 1 as illustrated in Table 1.

A. Crude Generalizations

Roughly, the behavior of S over all nine matrices seemed to be as follows: He first labeled the table as to whether it is all positive, mixed, or all negative (presumably doing a quick scan of the payoffs), although his later actions are not differentiable based on the label given.

Second, he sequentially searched the action alternatives open to him, fixating on those that have a distinguishing characteristic (e.g., one very large or very small payoff, or a number of "strong" or "weak" payoffs).

Third, he partitioned, explicitly and implicitly, his eight action alternatives into three sets (<u>while</u> sequentially examining them): those that he dislikes (a "reject" list), those that appeal to him (a "consider" list), and those that received no verbal indications (an "ignore" list).

Fourth, a choice was made from those alternatives that are present on the "consider" list. In the case of numerous alternatives, a pairwise comparison and rejecting process is used. From S's protocol, the data led us to infer the following comparison process:

(1) S <u>never</u> talked about computing a sum (row sum or column sum) or an expected value.

(2) Of all the numbers S verbalized, he <u>never</u> verbalized one that was close to a sum or

an expected value.

(3) S did verbalize on several occasions about comparing "... these possibilities across the board" (emphasis added).

From these data, we infer that S was doing some sort of column-by-column comparison of the two rows in question. Simply to have some definite procedure to follow, we devised a process that compares corresponding payoffs in each of two rows and that rejects the row that has fewer dominating payoffs. (This process is explicated later in Section IV, Part C, where SIDIP's CHOICE subroutine is discussed.)

Finally, it is instructive to mention the other types of verbal behavior present in S's protocol. Besides the anticipated vague statements, expressed confusions, and overt catharses, S did engage in a singular behavioral pattern, from time to time. Particularly on matrix 7 and to a lesser extent on matrices 1, 2, and 8, S spent some time (2, 5, 10, and 6 protocol lines on matrices 1, 2, 7, and 8, respectively) attempting to isolate identifying characteristics of the four columns so as to be able to assign subjective probability estimates to them. For example, on matrix 7, S noted that in one column there is a larger proportion of the larger payoffs in the matrix, and he tentatively concluded that that column had a lesser chance of occurring. However, in each case, S apparently eventually discarded that "column-processing" line of analysis and continued his original "row-processing" type of analysis.

B. Examination of PBG

Problem Behavior Graphs (PBGs) are concise ways of encoding the dynamics in the problem-solving and decision-making behaviors present in a protocol. (See Newell, 1966, for a complete discussion of the construction and utility of PBGs.) The nodes in such a graph are the actions or statements verbalized by the subject, either presented verbatim or paraphrased. The nodes are interconnected by lines (arcs or edges of the graph), which put a time ordering on the graph: time runs (first) to the right and (then) downwards. (The reason for allowing time to run to the right is to allow succinct presentation of episodic exploration on the part of the subject.)

Figure 1 presents the PBG over decision situation 1 that we developed and used in our study of S. In this PBG, there are behaviors consistent with our generalizations of Part A, behaviors inconsistent with those generalizations (but not necessarily inconsistent with our detailed model in the next Section), and behaviors not included in those generalizations.

The consistent behaviors are (1) the matrix is correctly labeled as being an all-positive (or, more accurately, all-nonnegative); (2) only those rows with double-digit entries (S_1, S_3, and S_5) are "considered;" the rest are ignored;

All positive here on the first table

$12, $0, $10, $4, and $8

Note 12 in S_1 -- Consider

Note 10 in S_5 -- Consider

Note ? in S_3 -- Consider

Note 0 in S_1 -- Note 0 in S_3

XEQ column looks larger

S_1 -- more money -- but has a 0 -- Discard

S_3 -- also has a large outcome -- but a 0 -- Discard

S_5 -- has a large outcome

Accept S_5

Figure 1: Problem Behavior Graph Over Decision Situation 1

(3) S_1 and S_3 are "rejected" for having the table minimum, namely, a payoff of zero.

The inconsistent behaviors are (1) S_5 is "considered" before S_3, violating our sequential row-processing hypothesis; (2) column processing may be going on when S verbalizes "$12, $0, $10, $4, and $8", as all of these payoffs are to be found in column ZEJ (see Table 1). Alternatively, S could be reading parts of S_1, S_5, and S_8, doing a row-processing analysis, or doing something altogether different.

A behavior by S not present in our generalization is S's observation that the "XEQ column looks larger." In fact, it is larger than the WUH and QUG columns, but only equal to the ZEJ column. Our subject apparently did not notice or utilize this bit of information.

Examination of all nine PBGs yields some evidence -- pro, con, and irrelevant -- to the preceding crude generalizations. But this is not the point. The question is how well our detailed computer simulation model of Section IV matches the choices and significant processes of the nine PBGs. This question is answered rhetorically in Section V.

IV. Simulation Program -- SIDIP

Our computer simulation program (SIDIP, an acronym for Simulation of Individual Decisions through Information Processing) is written in L6 (see Knowlton, 1966, for an explanation of the Bell Telephone Laboratory's Low Level Linked-List Language) for the Honeywell 635 computer. The program and job cards occupy approximately 500 card images, and the data take 40 card images. (Complete listings of both are available from the authors.)

A. Data Structures

The primary reason behind the selection of L6 as our language was its ability to construct and manipulate complex data structures. According to Newell and Simon (1972, pp. 19ff.), there are three important aspects of a simulation program of human thought: a set of symbol structures, encoding the information present in S's short-term, long-term, and external memories (covered in the remainder of this part of the paper); a set of Elementary Information Processes (EIPs) with which to operate on the symbolic information (presented in Part B); and a program, an ordered collection of EIPs, organized to accomplish a whole task (presented in Part C).

The major symbol structure posited for our S is an encoding of the payoff matrix, replete with row, column, and table description lists. The matrix is represented as a 32 node network (8 rows by 4 columns). Nodes in the same row are doubly-linked (from a payoff in the matrix, one can access its right neighbor or its left neighbor); likewise, nodes in the same column are doubly-linked. Besides the node linkages, the following node information is loaded and is static throughout a run: its row number, its column number, its value, and its sign.

The following descriptive information is dynamically created during the execution of SIDIP: the matrix is labeled as all positive, all negative, or mixed; maximum and minimum payoff values for the matrix are computed. The number of negative values in the table is computed. Nodes are labeled as double-digit if their value is larger than 9. Both rows and columns have these attributes created: maximum value, minimum value, count of the number of negative numbers, and count of the number of double digit numbers. Finally, each row is identified by its status or evaluation during the decision process: "Reject," "Consider," "Good," or "Accept."

B. EIPs

In one sense the Elementary Information Processes we presume are the legal L6 commands, and in another sense they are the seven types of EIPs specified by Newell and Simon (1972, pp. 29-30): Discrimination, Tests and Comparisons, Symbol Creation, Writing Symbol Structures, Reading and Writing Externally, Designating Symbol Structures, and Storing Symbol Structures. More specifically, SIDIP is based upon (and implicitly, we presume S has the capabilities of) these EIPs: the ability to create attribute information such as described in Part A, the ability to retrieve, compare, and distinguish such information, the ability to input (hear or read, for S and SIDIP, respectively) and to output (speak or print) symbolic information, the ability to do simple numeric processing (e.g., to add two numbers, to recognize that 7 > 5 and that -4 < -3, to save intermediate results), and the ability to interpret a program of EIPs. These are the only EIPs we require of SIDIP, and the only ones we posit about S. The sufficiency of these EIPs for SIDIP will be empirically demonstrated indirectly and implicitly in Section V. The necessity of these EIPs for S does not violate anything we now know about the cognitive capabilities of humans.

C. Macroprograms

In this part we shall indicate how SIDIP operates, primarily by reference to its flowcharts in Figures 2, 3, 4, and 5 and by reference to the output it produces, such as in Figure 6. SIDIP can be conceptualized as a main program (Figure 2) and three subroutines: EVALUATE (Figure 3), CHOICE (Figure 4), and COMPARE (Figure 5). The main program initializes data areas, reads and echo prints the decision matrices, does some background information processing, then calls on EVALUATE to sequentially examine and label each row (as "Considered," "Rejected," or ignored), and finally calls on CHOICE to determine the row to be "Accepted."

The EVALUATE subroutine examines each row in sequence for the properties listed in Figure 3 and makes an evaluation based on the presence or absence of those properties. Like our human subject, EVALUATE "states" (prints) what its evaluation of a row is (as soon as one is made) and also "states" (prints) what the reasons for the evaluation are. (If both a "consider" and a "reject" evaluation is made about a row, only the one made earlier is retained.)

Two of the terms in Figure 3 are vague: "strong entry" and "weak entry." Our operational definition of these terms, based upon S's verbalizations, follow. A "strong entry" is a double-digit payoff, when the matrix is not all negative and when the proportion of double-digit payoffs in the table is 25% or less. In all other cases, a payoff is "strong" if it is one of the top three payoffs just below the maximum of the matrix.

A "weak entry" is a negative payoff, in the case of a mixed matrix. Otherwise, a payoff is "weak" if it is one of the two payoffs just immediately above the minimum of the matrix. The CHOICE subroutine simply reorders the payoffs in each row and continues to call COMPARE until only one row remains "considered." The remaining row is "accepted."

The COMPARE subroutine performs a pairwise comparison of two (internally ordered) rows. A count is kept of the number of times one row's entries dominate the other row's. The row whose total count is smaller (if either is) is labeled "reject" and control returns back to the CHOICE subroutine.

(One may inquire what action SIDIP takes when two rows are labeled "consider" and neither one dominates the other. In this case, SIDIP would apparently be in an infinite loop; noting such cases, SIDIP "accepts" all such rows. Our

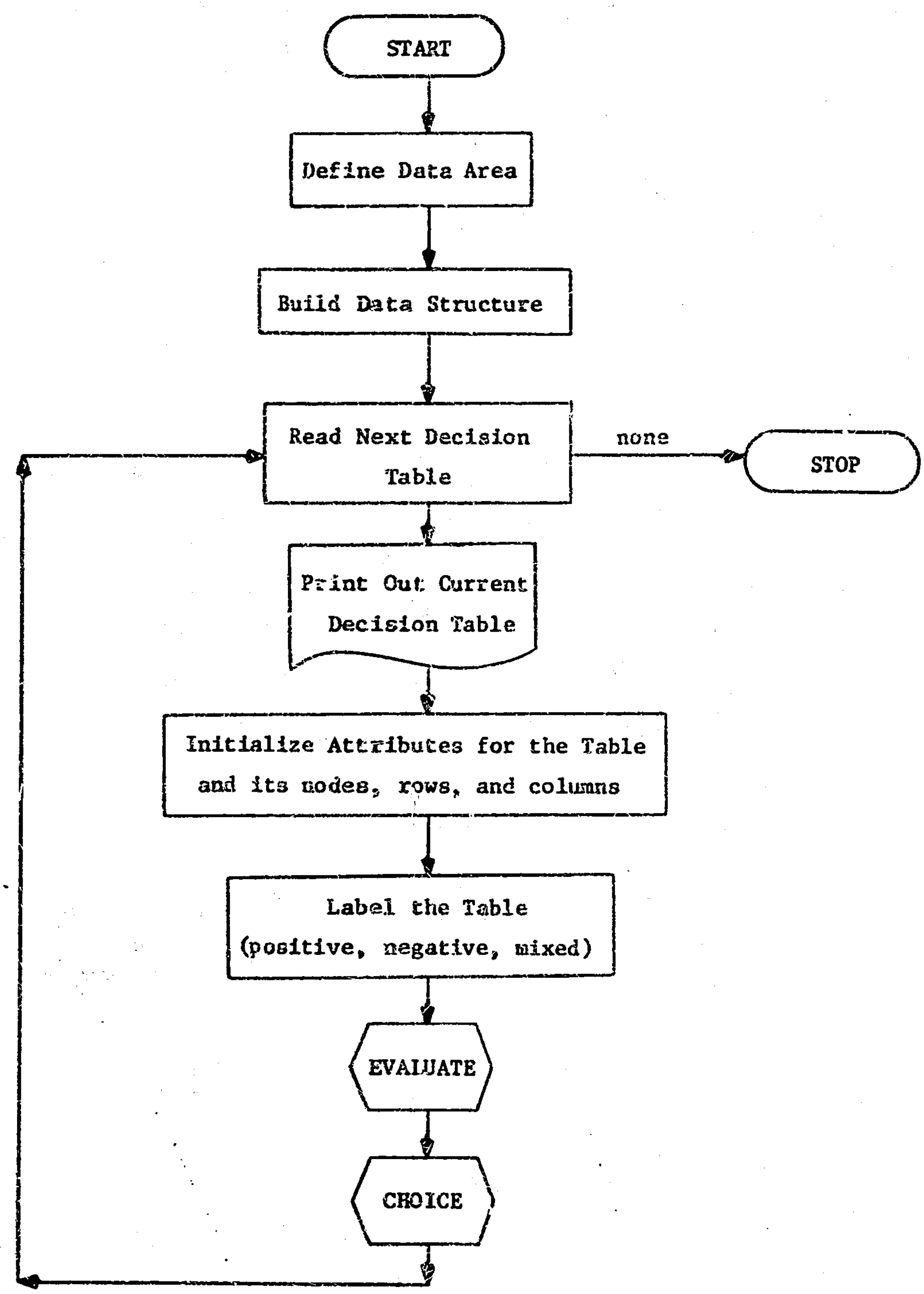

Figure 2: Main Flowchart of SIDIP

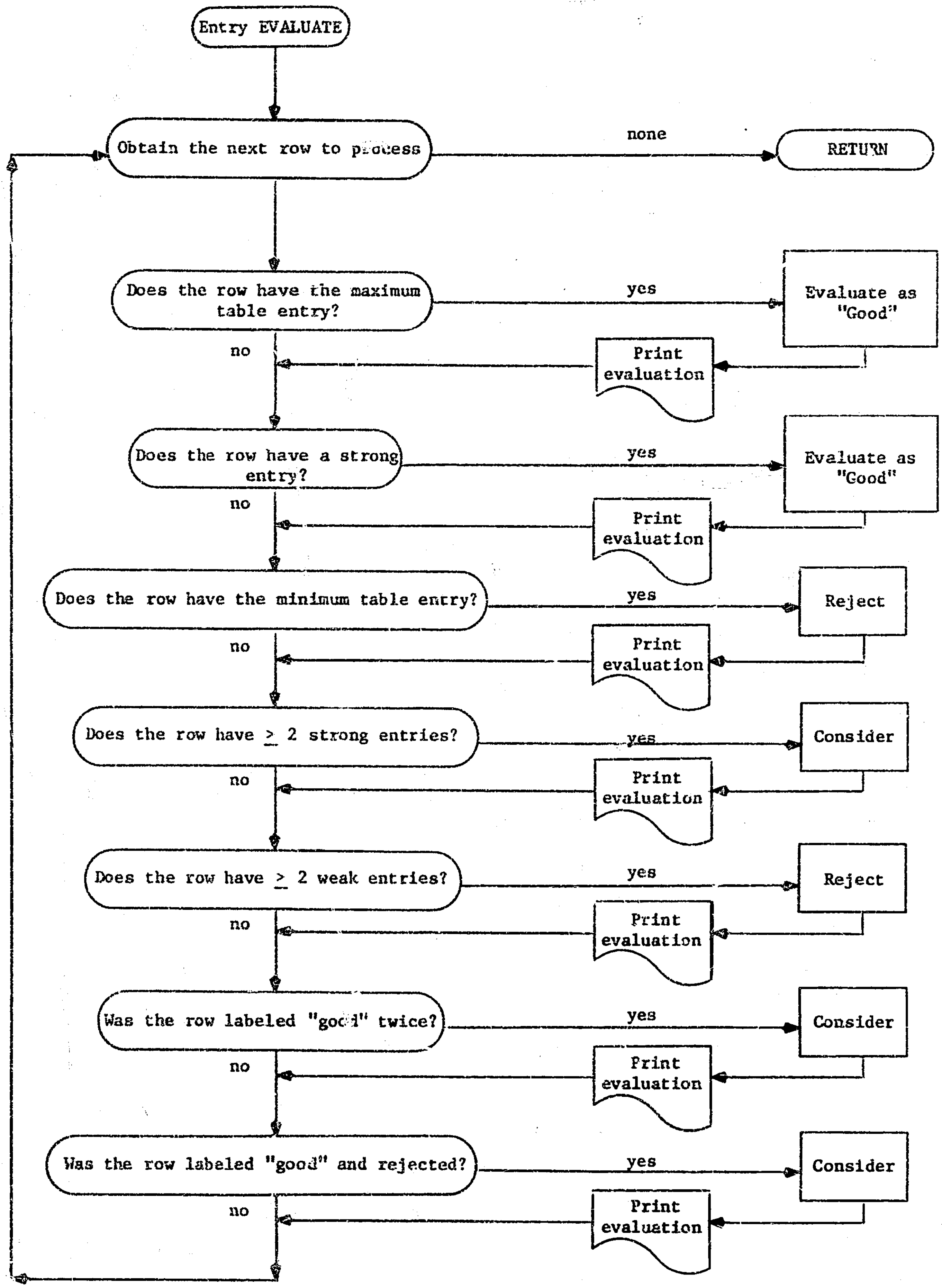

Figure 3: SIDIP Subroutine EVALUATE

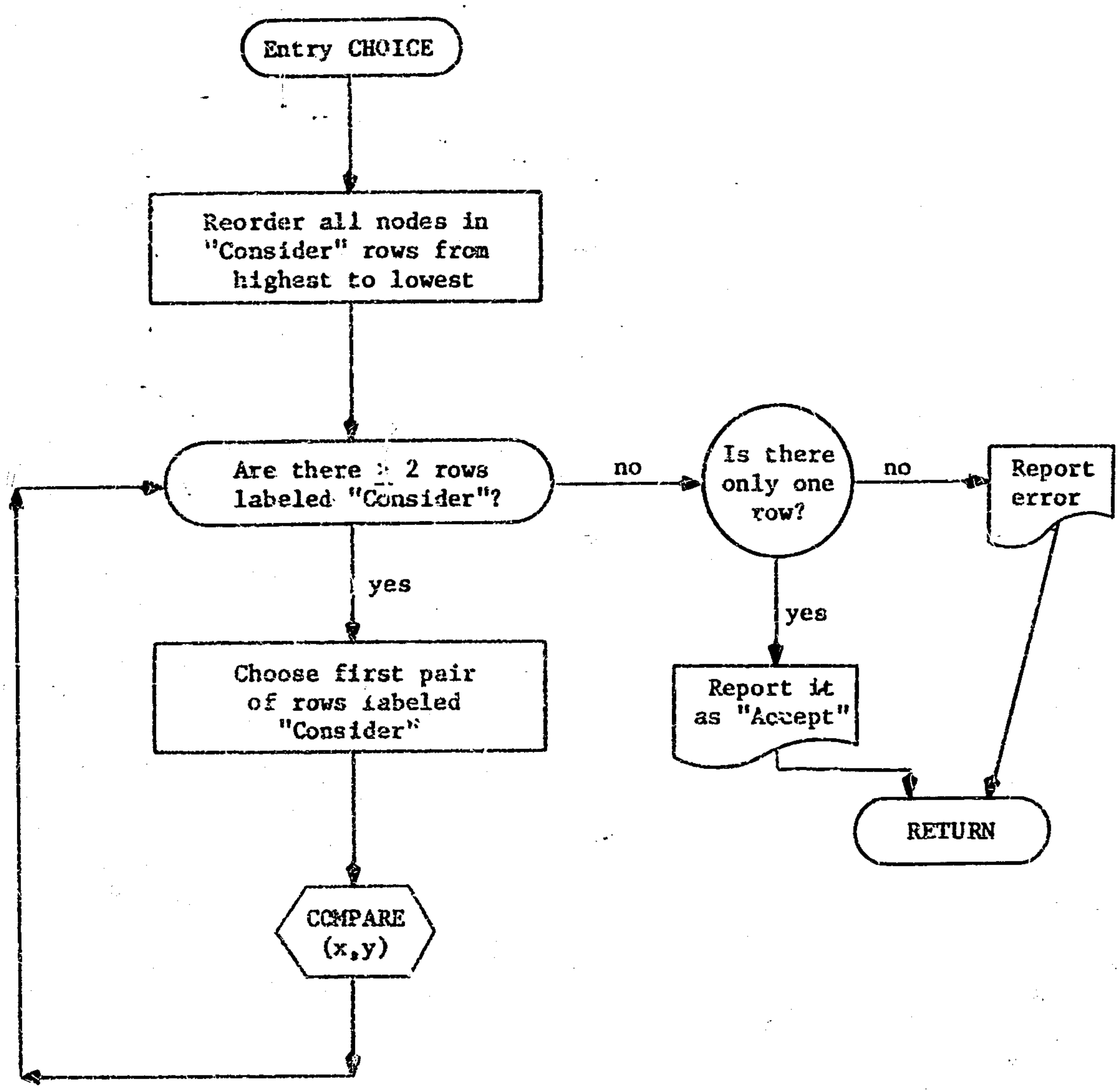

Figure 4: SIDIP Subroutine CHOICE

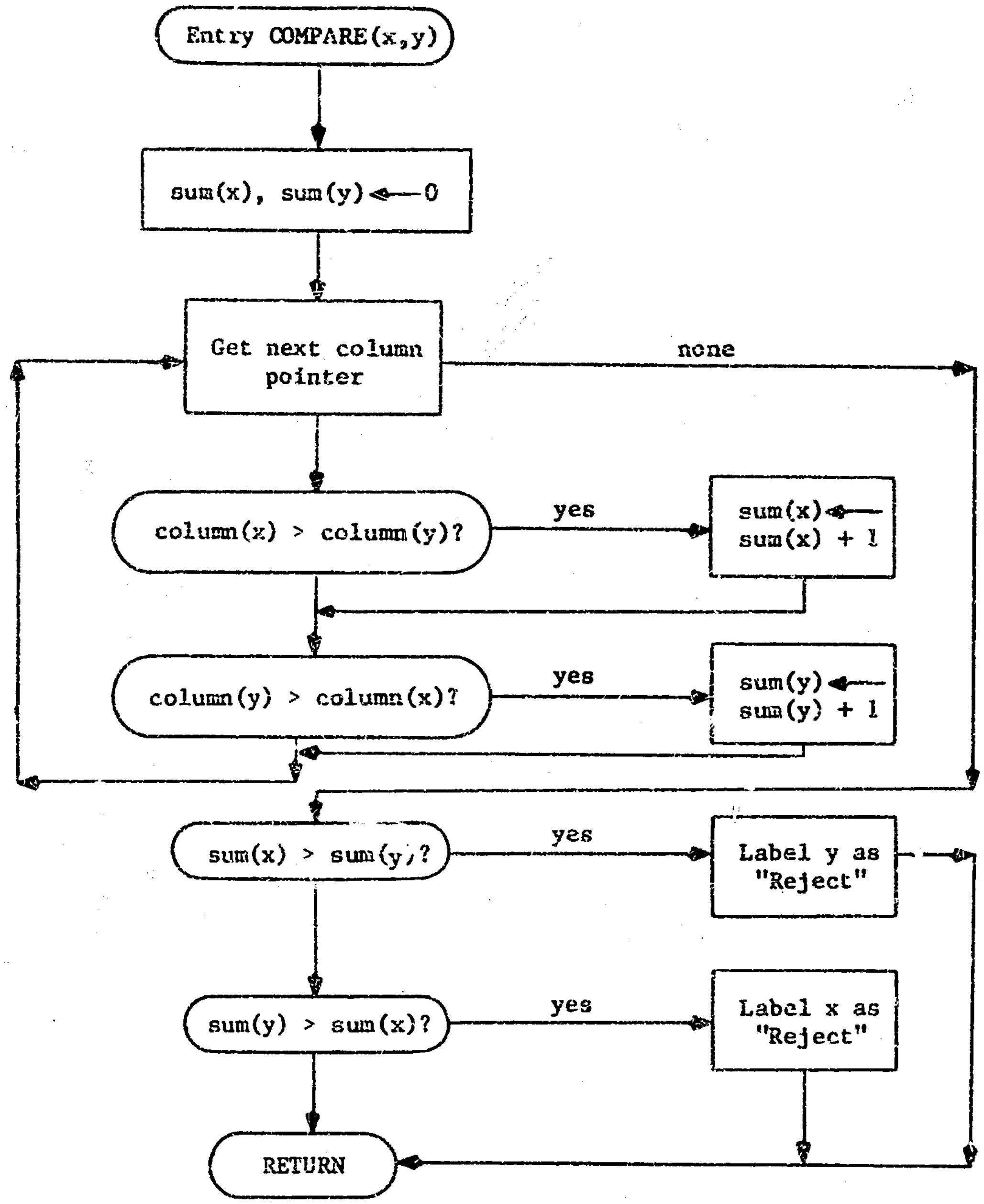

Figure 5: SIDIP Subroutine COMPARE

human subject never mentioned facing such a situation, but SIDIP encountered it in four cases.)

Finally, in Figure 6, one can see what is printed out by the computer as it processed decision situation 1: an echo print of the matrix, a labeling of the matrix, an evaluation of the rows, and a final choice. These latter three elements effectively constitute a trace or a protocol of SIDIP's behavior, which will be compared with S's protocol (PBG, really) in the next section.

V. Results and Analysis

A. Performance of SIDIP

When confronted with the same nine decision matrices as the subject, SIDIP made exactly the same choices on four matrices, made an incorrect choice on one matrix, and could not choose between two rows on each of four matrices -- but in these four cases, one of the two undecided rows was always the one selected by the subject. In a sense, then, SIDIP was "right" four times, "wrong" once, and "half-right" four times. The significance of these findings is examined in Part B. A report of the test of the decision process appears in Part C.

It may interest the reader to know that SIDIP's wrong choice occurred on decision situation 5 (mixed payoffs) where S chose the expected value row* and SIDIP chose the maximin row. In this case S explicitly did not reject the row having the minimum table value. The "half-right" choices occurred on matrices 3 (all negative), 7 (all positive), 8 (mixed), and 9 (all negative). S chose the expected value row in these four cases; SIDIP also chose the expected value row and respectively chose the regret row, the maximin row, the maximax row, and the maximin row in addition.

B. Performance Tests

To gain insight into the ability of SIDIP to simulate S's decisions, we shall compare its

*We shall continue to confuse the maximum expected value criterion and the Laplacian criterion -- which presumes a uniform probability distribution.

This table is all positive
|
Row S_1 looks good because of the maximum table value
|
Row S_1 looks pretty good, large numbers
|
Row S_3 looks pretty good, large numbers
|
Row S_5 looks pretty good, large numbers
|
Row S_1 looks bad since minimum table value
|
Row S_3 looks bad since minimum table value
|
My choice is S_5

Figure 6: PBG of SIDIP Over Matrix 1

choices to those of two models with random selection mechanisms.

The first model presumes choices are made by a random selection from the eight rows on a matrix with a uniform probability of $p = 1/8$. Since there are nine independent trials (decision situations), the Bernoulli assumptions apply, and we can derive two essential characteristics about our model:

$n = 9$, $p = 1/8$, and $q = 1 - p = 7/8$

therefore, the mean number of correct choices would be

$\mu = np = 9 \cdot 1/8 = 1\ 1/8$

with a standard deviation of

$\sigma = \sqrt{npq} = \sqrt{9 \cdot 1/8 \cdot 7/8} \cong 1.$

But SIDIP makes either 6 correct choices (4 fully correct plus 4 half-correct) or 4 correct choices, depending on how one chooses to treat the four "half-right" choices. This yields a Z score ($Z = \frac{x_1 - x_2}{\sigma}$) of either

$$Z_6 = \frac{6 - 1\ 1/8}{1} = 4.875 \quad \text{or}$$

$$Z_4 = \frac{4 - 1\ 1/8}{1} = 2.875$$

The interpretations are that the Z_6 score is 4.875 standard deviations better than the random model (statistically significant at $p < .001$), and the Z_4 score is 2.875 standard deviations better (significant at $p < .003$). A similar analysis assuming random choice only occurs among the four nondominated alternatives yields

$n = 9$, $p' = 1/4$, $q' = 1 - p' = 3/4$

with resulting changes in $\mu = np' = 9 \cdot 1/4 = 2\ 1/4$ and $\sigma' = \sqrt{np'q'} = \sqrt{9 \cdot 1/4 \cdot 3/4} \cong 1.3$ with corresponding Z scores of

$$Z'_6 = \frac{6 - 2\ 1/4}{1.3} \cong 2.88 \quad \text{and}$$

$$Z'_4 = \frac{4 - 2\ 1/4}{1.3} \cong 1.35.$$

Z'_6 is significant at $p < .003$, and Z'_4 is significant at $p < .1$.

Since we feel that it is too harsh to assume that a "half-right" choice is entirely wrong, we are forced to judge SIDIP's choices based on the Z_6 and Z'_6 scores. Consequently, we conclude that SIDIP's performance is significantly superior to the choices generated by these two random models of decision-making.

Next we could test "as if" choice behavior. The null hypotheses become S chooses "as if" he is using a maximax (or maximin, or minimax regret, or Laplace -- expected value) criterion. We cannot reject such hypotheses on a purely statistical basis (unless something like a strong inference point of view is accepted as in Barron, 1970) since we have no appropriate error theory and thus, no acceptable statistical methodology.

Rather than argue on a statistical basis we would suggest that S's protocols clearly show that since none of these decision models (random, maximax, etc.) is determining S's choices a more complex model such as SIDIP is required.

C. Test of Process Similarity

Turing's test (Turing, 1963) is a classical technique in the field of artificial intelligence by which one tests the processes in a program that allegedly simulates a part of human cognition. For reasons best illustrated by reference to Table 2, Turing's test is not completely applicable in this situation: S's protocol omits inarticulated processes. Thus, S's PBG is necessarily incomplete. In addition, S's PBG includes unnecessary or irrelevant behaviors as well as inconsistencies (errors). SIDIP is necessarily consistent, thus introducing perhaps insignificant processes (from a decision theoretic point of view). To test for congruence in the structures of the two PBG's, we quite arbitrarily aggregated all row evaluations and comparisons (calling them the "total behaviors" in Table 2), ignored all other behaviors, and then compared <u>all</u> of the "total behaviors" of SIDIP to the "total behaviors" of S. Since presumably S neglected to mention some of his thoughts, his "total behavior" is necessarily less than that of SIDIP, which we can force to be completely verbal. This does mean, though, that for many of SIDIP's behaviors, there will be neither collaborative nor disconfirming evidence present in S's (skimpy) PBG. Of the 50 row evaluations and comparisons by SIDIP which can be tested by behaviors of S, 39 of them (78%) agree with behaviors by S and 11 disagree. There are more behaviors by SIDIP that agree with behaviors of S than disagree in each of those nine cases, so SIDIP seems to be uniformly good. To get some feeling for the significance of the 78% correct figure, examine the following naive random model: suppose that this model either emits a correct behavior or an incorrect behavior with a uniform probability of 1/2. This is very conservative, because there are so many ways a row evaluation can be wrong (SIDIP's evaluation of a row can agree or disagree with S's evaluation; even if

Matrix	Total SIDIP Behaviors	Agreement	Disagreement	Absent	Total S Behaviors
1	9	8	0	1	8
2	19	7	3	9	10
3	9	2	1	6	5
4	14	3	1	10	4
5	9	4	2	3	9
6	10	1	0	9	4
7	24	8	3	13	12
8	11	3	1	7	4
9	11	3	0	8	5
Total	116	39	11	66	61
% of column 1		33.6%	9.5%	56.9%	52.6%

Table 2: SIDIP's PBG Compared to S's PBG

there is agreement, SIDIP is wrong if its reasoning differs from S's) and because there are three -- not two -- possible results from a comparison (a preferred to b, b preferred to a, indifference between a and b). Nevertheless, with that assumption and by assuming that the 50 behaviors are independent, we can again apply the Bernoulli model to derive the mean number correct (from the random model). This mean is $\mu = np = 50 \cdot 1/2 = 25$ with a standard deviation of

$$\sigma = \sqrt{npq} = \sqrt{50 \cdot 1/2 \cdot 1/2} = \frac{\sqrt{50}}{2} \cong 7/2 = 3\ 1/2.$$

The score for the behaviors of SIDIP is $Z = \frac{39 - 25}{3\ 1/2} = \frac{14}{3\ 1/2} = 4$, which is statistically significant at $p < .001$. Thus, SIDIP's intermediary behaviors were significantly closer to S's behaviors than this simple random model.

VI. Discussion

A. Conclusions

According to Van Horn (1971), there are several ways by which one can validate a computer simulation experiment: use models with high face validity, run "Turing" type tests, etc. (Van Horn lists several other techniques). The statistical tests that have been run so far on SIDIP suggest our simulation is significantly better than random decision models. The numbers and kinds of articulated behaviors summarized in Table 2 suggest that simple decision models such as maximax, Laplace, etc., are inadequate. As indicated SIDIP's intermediary behaviors are significantly closer to S's behaviors.

Mihran (1972) presents several procedures for verifying and validating both deterministic and stochastic computer simulation programs. However, none of these tests are really appropriate for protocol simulations. For this reason we have not further tested the structural congruence of the PEGs of SIDIP and our S.

There are some deficiencies in SIDIP. SIDIP makes one wholly incorrect choice, and four others are only partially correct. Eleven of SIDIP's 50 applicable behaviors are wrong, and 22 of S's 61 decision-making behaviors (36%) remain unexplained by this version of SIDIP. Thus we conclude that SIDIP explains reasonably well the nucleus of our subject's decision-making processes, but that there are still peripheral processes of importance by S that the current SIDIP does not capture.

B. Incremental Improvement in Decision-Making

Given the caveats in the previous part, it is obviously premature to press forward strongly in the area of improving decision-making processes by studying simulations of individuals. In order to conclude the research thrust begun in this paper, though, we shall pretend that SIDIP is near 100% successful to sketch the remaining work to be accomplished.

Assuming (heroically) that SIDIP adequately simulates the decision processes of S, we can now perform experiments upon the computer simulation program. Suppose, by way of illustration, that S articulates a desire to behave in a manner consistent with the maximize expected value criterion, but S does not consistently do so. Then an easy way to change SIDIP so that it behaves in that manner is to alter the COMPARE subroutine: after reordering the columns within the two rows (from highest payoff to lowest), do not compare the columns by simply noting "above," "below," or "equal." Instead, determine and record how much above or below one row's column is over the other. Total these differences, and the row with the higher sum is then the row with larger expected value.

Once it is learned that the suggested changes in COMPARE cause SIDIP to behave in the desired manner, then this information can be presented to S. By allowing S to learn of his shortcomings or through some similar procedure, improvements in S's decision-making behavior may be incrementally introduced. (The design of an acceptable training procedure is still an unsettled issue.) He retains the familiar and comfortable essentials of his decision-making process, but his decision-making ability is now improved.

C. Future Work

There are basically four avenues along which this research should be continued: First, improvements in SIDIP need to be made so that it simulates S's behaviors even more closely. For example, S clearly expends much effort in attempting to differentiate c .umns. SIDIP should also look for regularities or peculiarities in columns and then introduce corresponding changes into the subjective probabilities associated with those columns.

Second, protocols from more subjects should be collected so that more can be learned about the actual decision-making processes of humans.

Third, experiments should be conducted to learn a training procedure that is successful at modifying decision-making processes.

Fourth, this entire paradigm should eventually be moved out of the laboratory and into the real world. Ultimately, it is not the decision-making processes of college students that one is interested in studying, simulating, and improving, but rather the decision processes of military leaders, government officials, and business executives.

Bibliography

Barron, F. H., "The Potential Adopter's Decision Rule: A Constrained Optimization Model of Decision Making Under Risk and Uncertainty," Unpublished doctoral dissertation, University of Pennsylvania, 1970, and Working Paper No. 35, Department of Management Sciences, University of Waterloo, June, 1970.

Knowlton, K. C., "A Programmer's View of L6," Communications of the ACM, Vol. 9, No. 8, Aug., 1966, pp. 616-625.

Luce, R. D. and Raiffa, H., Games and Decisions, Wiley, New York, 1957.

MacCrimmon, K. R., "Descriptive and Normative Implications of the Decision Theory Postulates," in Borch, K. and Mossin, J. (eds.), Risk and Uncertainty, St. Martin's Press, New York, 1969.

Mihram, G. A., "Some Practical Aspects of the Verification and Validation of Simulation Models," Operational Research Quarterly, Vol. 23, No. 1, Feb.-March, 1972, pp. 17-29.

Newell, A., "On the Analysis of Human Problem-Solving Protocols," paper delivered at the International Symposium on Mathematical and Computational Methods in the Social Sciences, Rome, Italy, July 4-9, 1966.

Newell, A. and Simon, H. A., Human Problem Solving, Prentice-Hall, Englewood Cliffs, New Jersey, 1972.

Tuggle, F. D., "Decision Making Under Uncertainty: Empirical Tests of Normative Theory," Journal of Business Administration, in press, 1972.

Turing, A. M., "Computing Machinery and Intelligence," in Feigenbaum, E. A. and Feldman, J. (eds.), Computers and Thought, McGraw-Hill, New York, 1963.

Van Horn, R. L., "Validation of Simulation Results," Management Science, Vol. 17, No. 5, Jan., 1971, pp. 247-258.