A TRAINING GAME AND BEHAVIORAL DECISION MAKING RESEARCH TOOL: AN ALTERNATIVE USE OF SYSTEM DYNAMICS SIMULATION

Roger I. Hall
Faculty of Management
University of Manitoba
Winnipeg, Canada R3T 2N2

Tel: (204) 474-9709
Fax: (204) 261-6084
El.Mail: RHall@ccm.UManitoba.CA

ABSTRACT

The overall design is sketched of a training game and behavioral decision making research tool based on a System Dynamics corporate simulation model of a magazine publishing company. The organization of the program to run the game and the type of output generated is described. Some of the possible uses of the game to investigate behavioral decision making and group learning in a complex decision environment is mentioned.

INTRODUCTION

A system simulation model is a representation of the major resource flows in an organization, together with the resource regulators and the policy protocols for regulating the flows. Such models are normally the end result of a System Dynamics type study of a firm (or other self-regulating entity such as a commodity market), and the model is usually programmed in a continuous simulation language such as Dynamo or Stella. Unlike most other types of simulation, a system model contains feedback loops of influence that can generate unexpected and counterintuitive behaviors. The way groups of decision makers cope with such situations that are beyond their collective natural abilities to comprehend is a matter of great interest to researchers in the field of Behavioral Decision Making.

To use such a simulation as a training game or a behavioral decision making research tool requires macros or subroutines to collect and store the results on the simulation model of the individual or team players' decisions, generate the types of reports and comparative statistics typical of the industry, score the achievements of goals typical of the level of management in question, and feed the results back to the participants in an organized and timely fashion for the next decision to be made. Also, time-series charts of key variables and summary statistics are required as aids for debriefing purposes.

The training game and research tool to be described is based on a System Dynamics corporate simulation model of a magazine publishing company (Hall, 1976 and 1984). It was original programmed in the DYNAMO simulation language with external functions programmed in FORTRAN to print the results for each team and store statistics for later comparison. After many revisions, the game is now programmed entirely in Fortran in modules (subroutines) to carry out a number of interrelated functions [1].

AN OVERVIEW OF THE PROGRAM DESIGN

The program is designed around the following four two-dimensional arrays for holding data for 10 teams of participants: (1) a 50 x 10 array named SSTATE that holds the system states for reinitializing the 'level' variables in the set of time difference equations representing the dynamic behavior of each of ten companies (teams of participants), (2) a 60 x 10 array CSTAT that holds the current statistics (decisions made and results obtained) for each of the 10 teams, (3) a similar 60 x 10 array PSTAT that holds the previous values of CSTAT from one simulated year ago, and (4) a 60 x 10 array RSTAT that holds the relative differences between CSTAT and PSTAT, element by element.

The ending system states are held in the array SSTATE. At the end of each simulated year, SSTATE is written to a sequential file where its values are stored between runs. Each simulation run starts from the initial conditions for each company obtained from SSTATE and the decisions of that company's team of executives (participants) obtained from CSTAT. In a similar fashion, the current statistics in array CSTAT are stored in a file between runs. The array RSTAT is also written to a file for later use in debriefing the participants at the end of the game.

Before a run commences, the values of SSTATE are read back from the file to reinitialize the 'level' equations. Similarly, the values of CSTAT are read back to fill the elements of PSTAT that will be used later to compute the relative values for RSTAT. Each run simulates the operations for one year for each of up to ten companies in an industry. Various subroutines are called during the execution of the program to read the decisions made by each team into the first few rows of CSTAT, to run the simulation for one period for each team, and to store the results in subsequent rows of CSTAT.

Yet other subroutines are called to compute operating results from the data stored in CSTAT and RSTAT, to compute performance points for each company executive (team member), and to store these in the remaining rows of the array. When the array is filled, the relative differences between CSTAT and PSTAT are computed cell by cell and the results stored in array RSTAT. Next, subroutines are called to print detailed reports for each team and to compose a Notice Board from comparative statistics and leagues formed from the teams in descending order for various performance criteria. The arrays CSTAT and RSTAT are searched for significant events for printing as 'hot' news items. Finally, the arrays are written to sequential files where they are stored until called upon for the next simulation run. When the game is over, other subroutines and external programs are called upon to print graphs and summary statistics to aid in the debriefing of the participants.

THE SUBROUTINES

To accomplish all this, the magazine game program contains the following subroutines:

C O R P consists of a set of time difference equations, originally programmed in DYNAMO and later translated into FORTRAN, that represents the complex interactive operations of a magazine firm. The equations were taken from a previously published and tested model based on an empirical study of magazine publishing (Hall, 1976).

P R I M E S. From desired approximate values for starting the game in Year=0 (the starting year of the game), PRIMES controls the corporate simulation model CORP and simulates backwards to Year=-2 to provide a two-year history of the company (the decisions and operating results). It then simulates forward to fill in the cells of the arrays containing system states SSTATE, previous operation results PSTAT, and current operating results CSTAT for each team for Year=0. A report is prepared for each team showing the last year's operating results in preparation for their first decision. The ending system states in SSTATE and the current operating results and statistics in CSTAT are written to files to enable the program to be reinitialized when run again.

R U N S. From the initial values of system states held in SSTATE and team decisions held in CSTAT, it controls the corporate simulation CORP and simulates the operations of each company for one year to produce a new set of current operation results (CSTAT) and system states (SSTATE) for each team.

R E S U L T. This routine computes the relative differences, item by item for each team, between the previous year's results stored in PSTAT and the current year's results stored in CSTAT. The results are stored in RSTAT. It computes global industry average statistics and stores them in an array GSTAT. These data are required for the routines REPORT, NOTICE board and EXEVAL (executive evaluation).

N A M E. This routine reads in the names of the magazines identifying each team and the names of their executives (the participants), and the course title, name of the instructor or other titles identifying the industry, if more than one exists. It stores them in a character array NAMES. These stored characters are required for the routines REPORT and NOTICE BOARD.

D E C. This routine reads from a file the decisions made by the teams and stores them in the array CSTAT. It performs various checks to prevent reading in the decisions for the wrong year, team or industry.

D U M P. This routine reads the ending system states and operating data held in the arrays SSTATE, CSTAT and RSTAT into sequential files, where they are stored until the next run.

I N P U T. This routine reads back the previous ending states, etc. from the sequential files into the arrays SSTATE and PSTAT for the next simulation. It performs checks similar to those in DEC.

R E P O R T. This routine produces for each team an end-of-year operating report, a budget projection for the next year and a tear-off form on which to submit the next decision. It can be modified to investigate the effect of the availability of selected operating statistics on decision making behavior.

N O T I C E. This routine computes comparative industry statistics, sorts the teams by descending order into leagues for different measures of performance, selects manager-of-the-year awards and searches CSTAT and RSTAT for 'hot' news items about significant events in the magazine industry. It prints tables and a graph of an economic indicator in a format suitable for displaying on a notice board. It can be modified to investigate the effect of the availability of selected comparative statistics on decision making behavior.

E X E V A L. This routine computes points for the executives based on four levels of evaluation: (1) the performance of the company as a

whole (the owner's criterion), (2) the performance of the company on a chosen item relative to the industry average (the team's chosen 'edge of excellence'), (3) the magnitude of the operations managed by each executive (the Human Resource manager's criterion), and (4) the rating of the executive by the president (to give some power to the team leader). The schedule of rewards and punishments can be altered to investigate the effects on decision making behavior.

D E B R I E F is not strictly a subroutine but a separate program that prints comparative plots of selected variables from the data stored in the sequential files representing successive values of CSTAT and RSTAT. It contains the following subroutines:
(1) H I T S, a routine that produces a summary table of positive hits (policies that were tried and worked) and false positive hits (policies that were tried but did not work) for each team. It is used for debriefing the game participants and for investigating evolutionary group learning behavior, and
(2) P A R A D O X, a subroutine that computes measures of organizational (team) learning, based on a phenomenon known as Bowman's Paradox. It is printed in a graphical format to illustrate the improvement (or otherwise) in performance of each team over time.

OPERATING MODES

The following OPERATING MODES are available to aid in the running of the program. They are the first items read in when the program is run. They set the basic mode for running the program. There are 5 operation modes for the game: 1)PRIME, 2)TEST, 3)RUN, 4)RERUN and 5)DEBRIEFING. Only one mode can be operational at any one time and checks are built in to prevent conflicts.

P R I M E. By setting the mode to PRIME, the program will create two years of previous historical data, prime the files with values for starting the game and provide up to ten sets of company operating reports (one set for each team) for Year=0. The PRIME mode has to be run before the first RUN mode. PRIME controls the preliminary run to provide the first decision forms for the teams.

T E S T is used solely for testing purposes. When the mode is set to TEST, it primes the arrays and associated files and produces a single team report for Year=0. It provides a check of the correct working of the compiled program and storage files.

R U N. When set to RUN, the program runs in simulation mode for one simulated year to produce the operating reports for each team, and etc. A check is made to prevent running the program with the decisions for a previous year or the wrong industry (class), which would degrade the sequential storage files.

R E R U N. The purpose of RERUN is to make corrections to the previous run where a mistake was made in typing in the decisions or in setting the parameters. The mode RERUN can only be used for correcting the last year's decisions or parameters to protect the files from possible corruption. It reruns the simulation to produce a fresh set of reports and values stored on file.

D E B R I E F. The mode DEBRIEF is used at the end of the game to produce debriefing aids. The debriefing aids are: 1) Policy Hits Table, 2) Bowman's Paradox, 3) Plots, and 4) Executive's Brownie Points Final Standings.

GAME PARAMETERS

The following game parameters can be used to modify the game:

N O E X E C. The number of executives in each team can be set to 4 or 5, (Default=4). This provides some flexibility in arranging participants into teams. For example, a class (or group) of 24 can be organized into 6 teams of 4 players, each playing the role of a particular executive. On the other hand, a class of 50 can be organized into 10 teams of 5 participants by adding an extra role.

N O R P T sets the number of reports to be made available to each team (Default=2). The number of copies can be adjusted, depending on the available resources for printing large amounts (in excess of 15,000 lines for, say, 40 participants each receiving their own report).

L E C O N sets the pattern of the business cycle that effects the demand for advertising pages (LECON= 1, 2, 3 or 4, Default=2). LECON=1 will start the game in a depression with rising demand for advertising. LECON=2 starts in a normal condition with rising demand for advertising. LECON=3 starts in a normal condition with falling demand for advertising. LECON=4 starts in a boom with falling demand for advertising. By changing this parameter for different industries (classes or groups of participants), one can investigate the proposition that companies born in an economic depression will learn more and adapt better, than those born in boom conditions.

A M P B S sets the amplitude of the business cycle (AMPBS= 1,2 or 3, Default= 2). AMPBS effects the amplitude of the swings in demand for advertising. AMPBS=1 provides a 5% swing, AMPBS=2 a 10% swing, and AMPBS=3 a 15% swing in advertising sales. It can be used to investigate the effects on varying amounts of external destablizing input on organizational learning.

H I T A B L calls for the printing of the Policy Hit Tables for each team. It is used for debriefing purposes only (HITABL=1 print table, Default=0 no tables printed). It can be used to illustrate the behavioral decision making phenomenon of Superstitious Learning.

T H sets the Threshold of Perception of performance gaps (TH=any number in range 1 to 5%, Default = 2%). It is used in performance evaluations to provide the cut-off limits between Satisfactory and Fail. It is related to the behavioral phenomenon of Just Noticeable Differences.

C I V O M F stands for Change in Value of Money Factor (any number in range 1 to 20, Default =10). It is used at time of priming to adjust prices and costs to current $ since the game is based on the 1960 $. The factor can be adjusted with the passage of time to reflect the current value of prices used in the game.

P R I O R stands for the number of years prior to the last year to be included in Debriefing, (Default = 1 year). This will automatically exclude from the debriefing the selected number of years of data that might be biased by extraordinary decisions made by the participants in the final periods of the game.

A M N S T Y. Amnesty is granted to teams which accumulate a large debt. It writes off all the debts of the teams. It can be used when a team (or teams) become technically bankrupt, such that the interest

payments on debt exceeds their revenue generating ability. It allows them to make a fresh start.

OUTPUT FROM THE PROGRAM

The teams of participants usually make two decisions per week for 6 to 10 weeks (12 to 20 simulated years) to provide enough data for a 'good' debriefing. The participants are provided with a manual that explains the intricacies of the magazine publishing industry and with reports of the last two year's of operations of their companies (produced by running the game in the PRIME mode for Year=-1 and Year=0). They are then 'walked through' the reports to help them grasp the details of the operations of a magazine publishing company. Finally they are organized into teams and required to complete a decision form for Year=1. A hired assistant can usually type in the decisions, run the program and post the results in under an hour. A brief description of the documents produced by the program follows [2].

A report produced by the subroutine REPORT in the RUN mode provides: (1) a heading with the name of the company, the names of the participants by executive role occupied, and industry (class) identifier, (2) the decisions made at the beginning of the simulated year, (3) the condensed operating summaries for the subsequent year concerning readers and magazine pages, (3) the condensed financial operating summaries (the 'bottom line'), (4) the evaluation of the performance of an executive team and the computation of points as produced by the subroutine EXEVAL, (5) a budget prepared to aid each team in making its decisions for the next simulated year, and (6) a tear-off decision form supplied to each team that asks the participants to enter only those items that they wish to change for the next decision; thus reducing the amount of data entry and subsequent errors.

A list of comparative data, typical of the magazine publishing industry, is produced by the subroutine NOTICE on a continuous sheet of paper for displaying on a notice board. The list contains the following information: (1) the industry title and year, (2) a list of the names of the participants by team and the executive roles they have assumed, (3) comparative statistics of typical published industry statistics on such items as prices, readers, advertising sales and financial results, (4) leagues of the magazines by descending order for eight separate performance criteria (the teams are required to select one of these criteria as their 'edge of excellence' performance measure and they are evaluated on their ability to raise this measure above the industry average), (5) a list of Manager-of-the-Year Awards for participants in the five executive roles (computed by choosing the company or companies with the highest achievement in criteria appropriate to the role), (6) a league of all participants by descending order of points accumulated for each executive role, (7) a list of significant events ('hot news' items) recorded by the teams in the last year, and (8) an economic forecast for the next simulated year in a graphical form.

The output produced by the DEBRIEF subprogram comprises: (1) Hit Rate Tables for each team, (2) comparative time series plots of key variables, and (3) Risk-Return Paradox plots. (1) The Hit Rate Tables summarize the number of times each policy variable was changed and the effects on performance variables that the participants might have observed. It can be used to illustrate the effects known as

Superstitious Learning, Justification, and Action Framing. (2) The comparative plots assist teams to trace retrospectively the decisions made and the subsequent effects. The plots also create a vivid visual impact of an industry of diverse companies successfully pursuing different goals. (3) The Risk-Return Paradox plots can be used to illustrate an effect known as Bowman's Paradox. The Average Profit Margin is plotted against Variance of Profit Margin for each team over (a) the first half of the game, and (b) the second half of the game. The movement of each team's plots provides a rough measure of Organizational Learning (the ability of each team to increase the average while at the same time reducing the variance of profit margin). The negative association usually observed between return on earnings (profit margin) and its variance (i.e., more teams are in the high return-low variance and the low return-high variance categories than are in the opposing categories) runs counter to that expected from financial risk-return theory (i.e., teams that pursue a high risk strategy can expect to have a higher average return but at the expense of a higher variance). The teams are asked to explain the movement of their companies in terms of the learning that took place throughout the game, and the adopting and adapting of policies and strategic recipes borrowed from observing the successes of other teams (i.e.,how much learning resulted from internally generated ideas and how much merely copied and adapted from others).

CONCLUDING REMARKS

The magazine publishing game involves the participants in a 'living case study' where the decisions made earlier may have unexpected and counterintuitive results that may return to haunt the decision makers.

Furthermore, it can provide a vehicle for students to experience the behavioral dimensions of decision making in which a group learns to manage a complex and dynamic system in spite of limited understanding of the complexities of the situation and limited time and resources with which to unravel the interwoven skein of causality. From a System Dynamics point of view, the Corporate Model contains some seven feedback loops of causality, of which several have a 'positive' path correlation. Hence, small changes in certain decision variables can have extraordinary results, causing teams to suddenly spiral out of control. There are even more feedback loops effecting behavior, if one takes into consideration the interactions among the members of a team. For the example, the gain or loss of commitment to a course of action or by members to the group itself can further amplify these effects. A major consideration for the participants, therefore, is not maximising profits but how to maintain control and group cohesion when cresting on the wave of a vicious or virtuous cycle of events. How teams manage to achieve this control is a matter of some interest to researchers in Behavioral Decision Making. Even with the sparse knowledge we have, certain concepts and pathologies of the decision making process can be illustrated in a way that the participants are not likely to forget! Participants have remarked that the game was one of the most realistic and challenging exercises that they had experienced.

Finally, the program of subroutines provides an organizing shell around a corporate system model of a magazine firm. It should not be too difficult to embed in the shell other models from different industries to produce yet other training games which exhibit the same degree of perceived realism.

REFERENCES

Hall, Roger I., "A System Pathology of an Organization: the rise and fall of the old Saturday Evening Post," Admin. Sci. Quart., 21 (1976), 185-211.

_____, "The Natural Logic of Management Policy Making: its implications for the survival of an organization," Management Sci., 30 (1984), 905-927.

_____, "Magazine Publishing Game Player's Manual," (1987), available from the author.

_____, and Daniel Lai, "Operating Manual for the Magazine Publishing Game," (1987), available from the author.

NOTES

[1] A version of the game is currently being programmed in the 'C' programming language for running on a PC.

[2] An expanded version of this paper, that includes illustrations of the documents produced by the program, is available from the author.

This study was supported by a Strategic Grant from the Social Sciences and Humanities Research Council of Canada.