LET'S SIMPLIFY THE ADMINISTRATIVE REQUIREMENTS OF COMPUTERIZED EDUCATIONAL SIMULATIONS

David J. Fritzsche, Illinois State University

While the evidence seems to indicate that simulations are effective pedagogical tools, utilizing a simulation in the classroom can be a time consuming and sometimes a frustrating experience for the simulation administrator. Operational problems range from teams failing to turn in decisions on time to errors in the decision input or even computer breakdowns. A significant amount of administrative time is required to process each decision from collection to return following the simulation run. When problems arise, time requirements can escalate exponentially. Of course, Murphy's Law is always at work when it is least needed: If something can go wrong it will. When a simulation is used as an extension of the traditional course requirements, the additional time required usually comes out of the administrator~ hide. Thus, given the effort required to successfully run a simulation, it should come as no surprise that many of our colleagues have not flocked to the Promised Land of computer simulation. It is not unusual for first time simulation users to terminate their simulation activities at the end of one term. Even some of our seasoned colleagues tire of the effort required and revert back to traditional techniques.

How can we reduce the time required to run a simulation while at the same time retaining educational benefits? An examination of the duties of the game administrator offers some interesting opportunities for reducing the administrative effort required while at the same time increasing the quality of the administration activity. Simulation administration duties can be broken down into creative tasks and mechanical tasks. The creative tasks involve environment enrichment and the introduction of techniques designed to help the competing team members better understand the simulated business environment. Environment enrichment activities include strikes, legislative activity, foreign competition, and other "real world" situations which are not part of the routine simulation run. These activities may be built into the simulation as an option or they may be incorporated external to the computer program by the administrator. For example, teams may be permitted to purchase the services of a professional lobbyist to influence a congressional decision concerning a tax on some aspect of business operations. The tax option is built into the simulation; however, the congressional activities and the lobbyist are external to the simulation. The administrator may often find it interesting and educationally advantageous to introduce certain analytical techniques which the teams can use to better understand the competitive environment in which they operate. Creative tasks are generally interesting to both the administrator and the team members.

The mechanical tasks are simply routine work. They involve collecting the team decisions, submitting decisions to the computer, collecting the output from the computer, and returning the decision results to the respective teams. These input- output tasks add nothing to the educational value of the simulation; however, they account for much of the effort which the administrator must put into running a simulation. Fortunately, most of this effort can be eliminated by taking advantage of the capabilities of our modern computer systems with their time share capabilities. First we will examine the decision input task and then we will look at the decision output task.

DECISION INPUT

The prime objective here is to eliminate to the extent possible the administrator's role in submitting team decisions. A secondary objective is to reduce the incidence of error associated with the submission of team decisions. These objectives can be met by requiring teams to enter their decisions directly onto disc via a computer terminal. Computer cards are completely eliminated.

First a decision file must be established for each competing team. An input program must be provided which will enable the team members to correctly enter their decisions into the decision file. The input program should have the following characteristics: First it should be easy to call up by the team member. Second it should lead the member through the process of entering each decision by clearly showing him when to enter the individual decision variables. This should be accomplished without an excessive amount of computer generated verbage. Third the program should enable the team member to change any specific variable in the data file upon command. This is important both for correcting errors and for entering subsequent decisions which require that only a portion of the preceding decision be altered. Finally the program should enable the team member to list in extended form the total decision as currently stored on disc. For example, if scheduled production is entered in units of 1000 and the team decided to produce 550 of these units, the listing in extended form would show scheduled production as 550,000 items. The extended listing would provide the basis for a careful visual edit of the stored decision by the team member. No unit conversions would be required with their opportunities for error. Errors due to confusion of magnitudes should be all but eliminated. Typo errors which occurred on entry could be readily spotted and corrected by simply changing the entry in error.

After the team member is satisfied with the file contents, the decision should be run through a checker routine to determine whether it contains any data that would abort the simulation run. The checker routine should be the last step in the input program. Any errors found should be listed for the member to correct. The checked decision would then be stored on disc until time to run the simulation. We now have a two stage error checking procedure monitored by the team members. The first, the visual check, should eliminate most of the careless decision errors and the second, the checker program, should eliminate any data which would abort the run.

While requiring team members to input their decisions via computer terminal will remove most of the input burden from the administrator, the actual simulation run would still have to be initiated by him. However, this could be accomplished by the administrator in a short session at a terminal. The individual team decision files would have to be merged into industry decision files. The run could be batched from the terminal by industry and entered into the computer job stream. For those fortunate few who have a terminal in their office, the simulation could be run without leaving the office.

DECISION OUTPUT

The simulation output could be distributed without administrator involvement. Simulation program runs can be directed to a disc instead of to a lineprinter. The output should be entered into separate team files which team members can access at their convenience. This will require some modification of output coding for existing simulations. The decision results thus become available to each team upon demand following the current run of the simulation. To obtain their results, the team simply copies their file to the lineprinter or computer terminal depending upon length. It is an easy matter for the team to make multiple copies of the decision if so desired by copying the file as many times as they desire copies. This is certainly easier than loading multiple copy paper into the line- printer prior to printing the simulation results.

In addition to providing files for team output, each team may be provided with a file for data storage. Selected operating data could be stored in the data file with additions being made to the file following each decision. Data could be entered directly from the simulation run at the time the output file is built. The data base file could become the basis for a rudimentary information system. This file could be used in one of several ways. First if sufficient data are available to provide valid results, team members could be encouraged to utilize a statistical package such as SPSS to analyze their operating results. Even if the data are scant, it might be worthwhile to encourage use of such a package in order to provide experience using statistics for purposes other than learning statistics.

The data base might also be used for graphical analysis of the team’s operating results. This could readily be done by using one of the available graphics terminals such as the TEKTRONIX terminal which allows the user to graph several variables simultaneously upon its screen. Some of the terminals have provisions for obtaining a hardcopy of the graph appearing upon the screen. The major advantage of these terminals is that the team members can examine a series of graphical relationships in a relatively short period of time and, if the capability is available, carry the more interesting graphs home with them.

Another method of graphically analyzing the data is with the use of a plotting package available at most computer centers. This can be accomplished by using a plotter or in some cases the lineprinter. The lineprinter routines are generally easier to use than the plotter routines. However, either one can provide useful graphs for further study by the team participants. The computer plots have the advantage of being in hardcopy form. However, the user gets hardcopy of all graphs examined, which can result in a lot of wasted paper for the graphs which yield marginal results.

Whatever method of analysis is used, numerical or graphical, providing the team with a data base and encouraging them to use it should result in a more thorough understanding of the simulated environment plus a better understanding of the use of analytical techniques.

Output for the administrator can be provided in the same manner as that provided for the team members. The standard summary printouts can be placed into a file and copied to the line- printer or hardcopy terminal upon demand. If the administrator has ready access to a CRT terminal, he may forego copying the output and simply access it via the CRT when information is needed.

It would be desirable to create a data file for the administrator on an industry basis. This data could be used to keep abreast of team operations and to quickly spot weak teams which require additional help. The data would also provide industry graphs which could be used at the end of the course in the final debriefing session. A series of graphs could be easily prepared which would summarize the events of the simulated period accurately and concisely.

Another benefit of the data file is that it could provide a source of data upon which to base performance evaluations. Several performance schemes have been used which involve tracking team performance over time using a series of variables. These variables could become part of the administrator’s data base.

FILE REQUIREMENTS

When going from cards to a terminal based data entry system, as discussed above, a number of disc files are required. Three files are needed for each competing team, assuming the data file is provided. The first, a decision file, will require only enough space to hold all of the variables required for one period’s decision. Because this file will be revised each period, its size will not change. The same condition holds for each team's output file. The output from each new run would simply replace the output from the previous period. However, the data base file would need to be expandable. The output for each subsequent period would be added to the existing file so that the file would contain data for all periods run to date.

File security could be maintained by requiring each team to develop a unique password and then passwording the team's three files with their password. It would be the responsibility of the team to protect its password just as it is the responsibility of "real world" firms to protect their operating secrets from competitors.

COMMENT

There are several benefits derived from simplifying the administrative input and output tasks required to execute a simulation run. First the work of the administrator would be substantially reduced. By removing most of the mechanical tasks from the administrator, he would be able to lend more creative activity to the simulation while at the same time devoting less total time to the simulation activity. If we assume that creative tasks enhance the educational value of the simulation, this action would provide positive benefits for students. Second the procedure should result in fewer decision errors on the part of the competing teams. Thus the decisions made by the teams are more likely to be the actual decisions entered into the computer. Third team members would become familiar with the use of computer terminals. They should tend to view the terminal as simply another communication device and hopefully come to look at the computer as one more tool in their kit available for use in the "real world." Fourth the actual entry and pickup of decisions should be more convenient for the team members as these tasks can be accomplished at their convenience within the operating parameters of the simulation environment. These parameters include the time set by the administrator to run the simulation and the system time required to run the simulation.

We finally come to the question of who should develop these capabilities. One answer would be for each simulation author to provide input and output capabilities of the variety discussed above. Another alternative, if feasible, would be to develop an input and an output routine general enough so that they could be used with any simulation. If such routines could be developed in a portable fashion, the wheel would not have to be reinvented for each operating simulation.