(2) Payoffs

(A) Main effects:
    (1) Statistically significant
        (A) "Original"—80 pts. each
        (B) Replication—20 pts. each
        (C) Second replication—0 pts.
    (2) Statistically nonsignificant
        (A) "Original"—20 pts. each
        (B) Replication—0 pts.

(B) Interactions:
    (1) Statistically significant
        (A) "Original"—30 pts. each
        (B) Replication—10 pts. each
        (C) Second replication—0 pts.
    (2) Statistically nonsignificant

Behavior Research Methods & Instrumentation
1974, Vol. 6, No. 2, 121-123

        (A) "Original"—10 pts. each
        (B) Replication—0 pts.

(C) Confounded experiments—0 pts.

REFERENCES

Berztiss, A. T. *Data structures: Theory and practice.* New York: Academic Press, 1971.
Johnson, R. DATACALL: What it is and how to do it. Unpublished paper, Exxon Education Foundation, New York, New York, 1973.
Main, D. A computer simulation approach for teaching experimental design. Paper presented to the American Psychological Association, Washington, D.C., 1971.
Main, D. Toward a future-oriented curriculum. American Psychologist, 1972, 27, 245-248.
Main, D., & Head, S. Computer simulations in the elementary psychology laboratory. Paper presented at the Conference on Computers in the Undergraduate Curricula, Dartmouth College, June 1971.

Modeling and the Michigan Experimental Simulation Supervisor: An overview and some prospects

ROBERT L. STOUT

Mathematical Psychology Program, University of Michigan, Ann Arbor, Michigan 48104

A data-generating system (MESS) is described which facilitates the construction and study of behavioral science models of various types. The system provides a spectrum of options that make it possible to employ MESS in a variety of classroom situations with relatively unsophisticated students.

The Michigan Experimental Simulation Supervisor (MESS) is a program written in IBM FORTRAN IV (G); its basic purpose is to facilitate the construction and study of behavioral science models of all kinds. One version or another of MESS (formerly known as Expersim) is now in use at more than 20 universities, colleges, and community colleges in the U.S. and other countries in undergraduate and graduate courses in experimental design, statistics, and several content areas of psychology.

When MESS was designed, five principal considerations were used in making decisions concerning the program specifications: (1) It should be possible for students with all degrees of lack of sophistication to learn to use the system quickly; and students should spend as much time as possible on the design and analysis of their experiments and as little time as possible puzzling over how to get the computer to do them. In particular, the program should (a) require as few lines of input as possible, (b) allow students to describe conditions and groups using terms derived directly from the language of the problem area, (c) be highly tolerant of minor errors in spelling and syntax,

(d) provide helpful error messages, and (e) provide output which is formatted and labeled in such a way as to be maximally intelligible. (2) The system should be capable of handling models from any area of psychology, and of any desired structure or degree of complexity. The program should also allow a large number of independent and dependent variables, including nonnumeric dependent variables, and every kind of model structure, from static analysis of variance and regression models to dynamic, highly structured models such as finite automata or cognitive models like Newell and Simon's General Problem Solver. (3) The system should allow any general class of experimental design, including multivariate designs, repeated measures, confounded designs, and correlational experiments. (4) The system should provide a command language with as many options as possible to facilitate use of the system and provide pedagogical flexibility. (5) It should be as simple as possible to implement and modify models in the system.

The versions of MESS that have been produced so far have all been strongly student-oriented; that is, extraordinary effort was expended to make sure that

MESS was easy for students to use, while leaving the simulation designer and programmer plenty of work to do, although less than they would have under other circumstances. But for students, the system provides a number of desirable features; only five lines of input are required to specify a factorial experiment having (for the current implementation) up to 32 conditions involving up to 24 independent variables and 6 dependent or concomitant variables. Independent variables may take on numeric values, or keyword values, or both. Extra lines of input are needed for specifying repeated measures designs with up to 16 replications on a given group. A special context-sensitive keywork processor with some string-processing capabilities borrowed from SNOBOL substantially reduces failures due to mistyping or syntax errors. In no case is rigidly formatted input required. The MESS command language offers students several options for requesting special output, obtaining statistics, abbreviating printouts, or requesting information. Commands can be given to MESS at any time that the computer is waiting for input, even when the program is expecting a response to a request for, say, the number of conditions in the experiment. Thus, a student running the program from a terminal can ask for such information as the range of legal values of the variables while he is in the middle of setting up his experiment. MESS can be run with equal facility in batch or from a terminal.

There is still plenty of room for improvement in the student-to-model-to-student interface, but the favorable reactions of most of the hundreds of students who have used MESS at the University of Michigan over the past 2 years indicate that the system as it stands is generally satisfactory.

The program allows model builders a completely open field as far as the kind of model that is allowed. One implements a model by writing a FORTRAN subroutine and preparing one or two data decks. Aside from the conventions required for the MESS system to communicate parameters to the model subroutine, there are no restrictions on the model. The model need not produce simulated data at all; it may simply select data from a file stored on disk or tape. It might be interesting in some educational or research applications for a model to provide either simulated or actual raw data at the student's or instructor's option. Although most models so far implemented use the dynamic numeric output formatting features of MESS, the use of those features is optional. Models could be implemented whose output was nonnumeric-spatial configurations, drawings, or sentences.

In our work at the University of Michigan, we have attached considerable importance to having models produce data that mimic genuine raw data in as many ways as possible. Consequently, few of our models comply with the assumptions of analysis of variance, and the output sheets of our students are frequently strewn with missing data in the form of simulated dead rats or fired workers. Some models also calculated simulated costs of experimentation.

For the instructor, MESS provides a spectrum of options which make it possible to employ it in a variety of classroom situations, and to use it with different pedagogical techniques. In addition to the basic option of running experiments on the terminal or in batch, a command is available to have the system accept input from the terminal and print all results on a remote high-speed printer; error messages and prompting messages are still printed on the terminal. An instructor may also decide the descriptive statistics that a student should see, or he can allow the students to decide individually what they want, or make it impossible for the students to obtain statistics. MESS itself calculates only simple descriptive statistics—means, variances, product-moment correlations, etc.—but a command is available to direct MESS to place results in a file on disk, tape, or cards in a format that is suitable to input to most statistical analysis programs, such as the UCLA BMD series or any of several interactive statistical programs. Again, the instructor has the option of requiring use of the feature, allowing students to use it at their option, or making the feature unavailable to the students. An instructor can also alter a simulation model in a number of useful ways without reprogramming. Among the changes that can be made are altering the default value of a variable, its range of legal values, and its name and set of possible abbreviations. One can also change the status of a variable from "external" to "internal." "External" variables are visible to and manipulable by the students; "internal" variables might as well not exist from the students' point of view—the program will refuse to recognize the variable name and will not mention the variable in any printouts. By using this feature, it is possible to hold back certain critical variables until one is ready to introduce them. Also, in some models, such parameters as error variances are controlled by internal variables, so that by changing the default value for these variables the instructor may change the error variances without reprogramming. There are other sorts of changes an instructor can make, but they are generally of less importance.

The model designer and programmer benefit least from the current version of the MESS system. The relative difficulty of implementing a model makes the MESS system less effective as an educational tool in two ways: by reducing the variety of models available for use, and by making it effectively impossible for students who are not sophisticated in computer techniques to construct and explore models of their own. On the other hand, the system does not hinder the simulation designer by unduly restricting the way models can be constructed, and it does relieve the programmer of the burden of input-output programming, which is frequently the most onerous aspect of programming. It is obvious, however, that the principal weakness of the MESS system is that it does require the programming in

FORTRAN of all models. It would be far more desirable to provide a special-purpose interpretive simulation language on the order of SIMSCRIPT. Unfortunately, our resources at this time are insufficient to support the very substantial programming effort that would be involved in providing such a language. A complete reprogramming of the entire MESS system in some language other than FORTRAN would be a minimum first step on the way to an interactive interpreter.

I believe that behavioral scientists should support the development of simulation tools of this sort for use not only in teaching, but in research as well. In many areas of psychology and such allied disciplines as sociology, geography, and anthropology, investigators are forced to deal with complex and ill-understood phenomena, and in such situations simulation offers a reasonable and systematic way of exploring the implications of alternative theories. Furthermore, in complex situations, a properly constructed and documented simulation model would be the clearest, most concise, and most rigorous way of expressing the state of one's knowledge about a given area. Clear and rigorous thinking is essential when dealing with a maze of interacting factors, and I think simulation technology can substantially increase our ability to deal with these difficult situations. Lamentably, simulation technology is not doing us much good right now. Only a tiny fraction of all behavioral scientists have ever used the computer to do more than calculate statistics or run rats. Even on large-scale computer systems, where a variety of high-level languages are available, few investigators attempt to take advantage of the opportunities available. One of the major factors which maintains this state of affairs is that a major investment of time and energy is required for a behavioral scientist to become sufficiently expert with the computer system and the language he has to use to express his ideas. Learning a computer language is a formidable task for the average psychologist, and so much time is required to attend to the details of input, output, and program flow that little

time is left to work on the essence of the model. For these reasons, I believe that a high-level simulation language, convenient though that would be, is only a small fraction of what systems like MESS need to have to be fully accessible to the whole community of behavioral scientists. An interactive interpreter or compiler, even a sophisticated one with laissez-faire input conventions and intelligible error messages, is still nothing more than a passive instrument. It may be a powerful tool in the hands of an expert, but we cannot expect most scientists who are primarily interested in behavior to become experts in programming, and we cannot afford to provide an expert programmer-mathematician-systems-scientist for every research psychologist. We can, on the other hand, make the computer into something more than just a passive tool. We can construct programs which will not only eliminate many burdensome programming tasks, but also actively assist a modeler in the construction of his model by interrogating for necessary details, outlining alternatives, and supplying supplemental information such as sensitivity analyses for incomplete models or submodels as well as the complete model. Programs are now available that construct probability distributions by interrogating relatively naive persons, mostly businessmen, concerning their opinions about the variable in question. I propose that the same sort of thing be done on a much larger scale for model builders. Computer systems of this nature are possible with current computer technology, and I think we should begin to take steps in this direction, even though funding for a thorough project may be years in the future.

The experience that we have had with computer data-generating systems in the classroom demonstrates that simulation systems can be of substantial benefit to our students. We should continue to develop these systems so that they may help to improve the intellectual capabilities, not only of our students, but of ourselves and our colleagues as well.