SESSION VIII USERS' EXPERIENCES WITH COMPUTER-ASSISTED INSTRUCTION

What's happening in CAI: A survey of ongoing projects

DORIS AARONSON

New York University, New York, New York 10003

As there are more CAI papers in this year's "Computers in Psychology" program than in any past year, the membership was surveyed to provide a broad overview of current trends. This is a report of the obtained data (1) on what courses use CAI and the hardware supporting those courses, (2) on the software and "courseware" involved, (3) on CAI record-keeping and evaluation procedures, and (4) on the financial costs and educational benefits of the various projects.

The use of computers in education has increased markedly during the past few years (Psotka, 1982). In a national survey of psychology departments taken last year by Castellan (1982), 62% of the faculty members reported that instructional computing at their universities had increased during the last 2 years, and 75% anticipated that it would increase further during the next 2 years. Consistent with this trend, this year's "Computers in Psychology" program has more papers on computer-aided instruction (CAI) than in any past year. Thus, we decided to survey the membership on the details of their CAI projects, to provide a broad perspective of what was happening. I will report here (1) on the uses of CAI and the computer hardware supporting those uses, (2) on the software and the "courseware" being implemented, (3) on the procedures being used to evaluate the effectiveness of CAI, and (4) on the cost/benefit economics of these projects.

Survey forms were sent to authors of about 30 articles on CAI that appeared in *Behavior Research Methods & Instrumentation* during the past 5 years, and to a few others also involved in CAI. Responses were received from 12 people working on CAI projects at seven different universities [University of Kentucky, Indiana University, Notre Dame University, University of Pittsburgh, New York University (NYU), Brown University, and the University of California at Santa Barbara]. The graphs to be presented are based on only

a subsample of 10, as 2 respondents sent letters that could not be interpreted within the survey format (but which are reflected in the text). Although the sample is small, it is fairly representative of the wide variation among CAI projects currently in progress.

CAI USE AND HARDWARE

Figure 1 shows frequency distributions for the types of courses in which CAI is being used and the length of time it has been used in those courses. All of the courses included in the survey represent graduate or undergraduate psychology courses. The top graph shows that most of the applications are in quantitative courses, which include statistics, logic, and computer applications. In accord with this trend, the first session of this year's conference was devoted entirely to "Computer-Assisted Instruction in Quantitative Methods" (Fazio

Figure 1. Frequency distributions for (A) courses using CAI and (B) their duration of use, in years.

For reprints, write to D. Aaronson, NYU Psychology Department, 6 Washington Place, New York, New York 10003. I thank Edmund Acker, Charles Chubb, Mira Divinsky, Martin Gizzi, Alina Hsu, and Misha Pavel for helpful comments on an earlier draft.

Figure 2. Frequency distributions of (A) CAI hardware and (B) number of student stations.

& Backler, 1983; Levy, 1983; Pavel, Marcovici, Sherman, & Falmagne, 1983; Schaffner, 1983). The frequencies in this graph sum to 12 rather than 10 because 1 respondent is using CAI in three different courses. The bottom graph shows that half of the projects are rather new, using CAI for 2 years or less.

Figure 2 shows frequency distributions for the types of hardware supporting these CAI projects and the number of student stations employed by each respondent. The left graph shows that the Apple II microcomputers are in the lead. In this sample, they were generally configured as a CORVUS network (Lowe, 1983). The DEC-based projects include two PDP-11/34s and two VAX-780s. A few projects are connected to university computer centers, using large IBM and Control Data Corporation (CDC) machines. Again, the frequencies sum to more than 10 because some respondents are using more than one type of computer. This graph illustrates a change in machine type that has occurred in research applications, as well as educational uses of computers. Today, the microcomputers dominate; 5-10 years ago the minicomputers were the most frequent, and 10-15 years ago the large university-owned mainframes were the only machines available for most computer-based projects.

Note the bimodality in the right graph on the number of student stations: The modes are at 4 and 12 stations. The respondents indicated that 60% of the terminals at these stations are programmed as "smart" and 40% as "dumb" peripherals. Further, half of the CAI stations are run "stand alone" and half under "timesharing" software. Spelt (1981) suggests that the important differences in instructional systems are no longer those between micro- and minicomputers, but rather, those between stand-alone and timesharing software, often correlated with machine type.

SOFTWARE AND COURSEWARE

The data in Figure 3 concern CAI "software." These graphs are frequency distributions for the author languages being used in the projects and the programming languages involved. The top graph shows that the most frequently used author language is PILOT. PILOT, which is programmed in PASCAL, is the author language developed for the Apple system. It sells for only \$150 and is the major reason that so many CAI projects use the Apple (Spivey, 1983). It is interesting that three

projects have no author language at all but are using their systems' algebraic programming languages directly, generally FORTRAN and BASIC. Only one project, the NYU logic course, was developed with an author language programmed by the project sponsors themselves, "ARIS," programmed in C (Pavel et al., 1983). ARIS is an entire CAI system, including components to control the teaching process, monitor student stations, organize class records, manage files, and so on. Embedded in this system is a limited-scope "author language." Likewise, TUTOR is one component of the PLATO CAI system (E. Holt, 1983; R. Holt, 1983; Simutis & Barsam, 1983). A consideration of the nature of existing author languages and their relations to other components of the CAI system would provide useful information for future CAI development.

Next, let us turn to the most time-consuming aspect of CAI development, the "courseware." Figure 4 shows frequency distributions for both instruction and the testing modes. Although the various "modes" indicated along the X-axes in these graphs are not mutually

Figure 3. Frequency distributions for (A) author languages and (B) programming languages used in CAI.

Figure 4. Frequency distributions for (A) instruction modes and (B) testing modes used in CAI.

exclusive, particular course segments can usually be conceptualized as focusing predominantly on one or another of these educational procedures. The frequencies in Figure 4 (and in later graphs) sum to more than 10 because a given CAI project may use several different modes. The average number of instruction modes used by a given project is 3, with a range from 1 to 6. The average number of testing modes is 2.8, with a range from 2 to 4. The most frequent instructional mode is "simulation," because of its combination of versatility and programming ease. Useful discussions of simulation systems are presented by Levy (1981), Main and Head (1971), Spivey and Jackson-Smith (1983), and Williams and Hollan (1981). The least used instructional mode is "dialogue," because of the inherent programming complexities and the difficulties in orienting students to such a system. For example, a dialogue mode would need to establish a common vocabulary between the student and the system in which interactive conversation could occur. Further, it would need a good, and thus probably large and complex, syntactic parser. Finally, it would need interpretive and branching facilities that are more sophisticated than those needed for other modes. As this mode is closest to the usual teacher-student interaction, it would be useful to have more research and development in this direction. In the next paper of this session, Castellan (1983) describes in more detail the uses of several different instructional modes.

The frequency pattern for the testing modes is somewhat disappointing, but again, it is understandable in terms of programming ease. Simple recall (or, as the students say, "spit-back") of facts is the most frequent. And, the most educationally creative mode, "applications" of ideas in new situations, is the least frequent mode of testing. Lesgold (Note 1) has written an interesting technical report for the National Institute of Education (NIE) describing criteria for future authoring environments that would make creative and educationally sophisticated use of various instruction and testing modes. Interesting papers on currently used modes have been written by Eckerman (1981), Lesgold (1982), and Psotka (1982).

One of the most important aspects of a CAI system is its flexibility in instructional and testing situations. Traditional classroom teaching generally has a fixed lecture, reading, and testing structure that is the same for every student. The teacher cannot afford the time and does not have the information to develop a course that is tailor made for every individual student. But the computer-based system can vary many parameters of the learning situation, contingent on individual student performance. Figure 5 shows how current CAI projects are making use of such flexibility. This slide shows frequency distributions for the use of some learning variables and of variation in item selection for those learning variables.

These data suggest that current CAI projects are not taking maximum advantage of the flexibility permitted

Figure 5. Frequency distributions for (A) learning variables and (B) item-selection modes used in CAI.

by computers. On the average, respondents are using only two of the learning variables indicated, and two of the projects use none. The top graph (Figure 5) indicates that the most used dimension of flexibility is the "learning pace," that is, whether the student can repeat or skip over items. Variation in instructional modes and the use of "help" facilities tie for second place in the types of flexibility incorporated in CAI systems. The least used dimension is variation in the learning sequence. Optimal use of this option requires a sophisticated, generally hierarchical organization of the information to be learned and a programming structure that can move students from one node to another contingent on the knowledge they demonstrate. Articles in cognitive psychology and psycholinguistics have been giving us hierarchical theories of text structure and of text processing for the past 5 years or so. But, when we move from experimental stimuli to natural text in the learning environment, those theories are not being, or cannot be, implemented.

Likewise, the bottom graph (Figure 5) shows that the CAI projects are often not making full use of the possible flexibility in the order in which items are selected. The most frequently used order is "teacher determined": again, not taking advantage of the computer's strong points. However, the next most used option is "contingent on student performance" and is generally based on accuracy rather than on time measures. It is interesting that one of the least used options is "student selection from a menu." As teachers, we do not permit, or trust, students to select for themselves what to learn next. This contradicts some principles of active, self-paced learning and of "intrinsic" motivation.

RECORD KEEPING AND EVALUATION

The most tedious and least creative parts of the teacher's work are record keeping and the evaluation of

Figure 6. Frequency distributions for information used in (A) student records, (B) class records, (C) feedback to students, and (D) comparisons of CAI and traditional control classes.

students based on those records. As these jobs are easily specified and rather routine in nature, they provide an ideal way for the computer to help out. The top two panels of Figure 6 are frequency distributions of the types of information used in record keeping, respectively, for the individual student and for the class as a whole. The bottom two panels are frequency distributions of the types of information on which evaluations are based, again respectively, for the individual student and for the class as a whole.

Surprisingly, only 60% of the projects have student record keeping as part of their CAI system. But, for those that do keep records, data are maintained on an average of 4.7 aspects of performance, such as those indicated across the X-axis in the top left panel (Figure 6). The most frequent type of data used are overall accuracy data, that is, the total number of items correctly answered. Most of the other data sources are also accuracy related. For example, the number of items attempted and the number of items studied to reach some accuracy criterion are tied for second place in frequency. The least frequent performance measures are time based, such as the amount of time per item and the total time spent learning. When it comes to records for the class as a whole, the imbalance between time and accuracy information becomes even more extreme. The top right panel shows that 40% of the projects keep records on class accuracy, but none maintains time-based records. Thus, 60% of the projects do not maintain any summary class records of CAI performance.

Not indicated in Figure 6 are data on item-based, as opposed to student-based, records. Such record keeping is almost nonexistent. For example, only two of the projects keep records on relative item difficulty, and none keeps information on correlations among items. As the bulk of the CAI development time is spent on "courseware" for teaching and testing, it is indeed sad that the respondents are not evaluating whether or not their particular learning items are any good. The lack of

richness in record keeping was surprising, as 80% of these projects were supported by the National Science Foundation (NSF), which requires record keeping and evaluation as a condition for funding. But, many of these projects are less than 2 years old and include plans to enhance these aspects in future years.

Although a large amount of information on time and accuracy is not permanently maintained, students are given an immediate evaluation of their performance in terms of feedback on these attributes. The bottom left panel of Figure 6 shows that all 10 projects provide students with displays of the correct answers and with information on their own accuracy. Eighty percent of the projects also provide feedback on the student's own speed. However, this graph indicates that far fewer projects (1) provide feedback on a student's time or accuracy relative to the rest of the class or to some absolute standard or (2) an explanation of why the answer was right or wrong.

The lower right panel indicates that relatively little has been done to evaluate CAI in comparison to traditional classroom teaching (but see Belmore, 1983). Ninety percent of the projects claim to have subjective information on student motivation, but it is not clear how to interpret such information, as it is based on an experimental design that is "between classes" and generally also "between teachers." The survey form requested respondents to indicate sample sizes for their evaluation data, but over half of the respondents left these slots for traditional classes blank, thus questioning a real comparison. Six of the projects claim to have compared exam scores for CAI and traditional classes, but two of those filled in the sample sizes only for the CAI students and one filled in no sample sizes. In summary, Figure 6 suggests that some future work is needed on both record keeping and evaluation of CAI. A fine start in this direction is provided by the PROCTOR system (Crowell, Quintanar, & Grant, 1981).

CAI ECONOMICS

Next, let us turn to the economics of CAI. Figure 7 provides information on the time and money invested in CAI hardware and software. As these costs vary tremendously among projects, the graphs indicate the

Figure 7. (A) Cost of CAI hardware. (B) Time for CAI course development. (C) Cost of CAI course and courseware development.

minimum and maximum in addition to the mean. Note that the Y-axes are scaled differently for the three graphs in this figure.

The left graph shows that the average hardware cost is about \$100,000. The range is from \$25,000 for an Apple project to \$250,000 for a VAX. The hardware costs for the university computer center IBM and CDC machines are obviously not included in this graph.

The middle panel of Figure 7 shows the amount of time invested in developing the courseware and the course itself. Again, there is substantial variability among projects, the range for getting a course on the road being from 1 month to 6 years. Most of the Apple-based projects took about 1 year to run smoothly. The 6-year project had a pilot course running in 2 years and incorporated many improvements thereafter. That project exceeded all the others in the variety of instructional and testing modes used, in the flexibility of learning parameters and item selection, and in evaluation procedures. As one would expect, a high-quality system requires a substantial investment of time and effort.

The rightmost graph (Figure 7) shows data on the costs of developing the course and its courseware. These data indicate a mean of about \$12,000, with a range from \$3,000 to \$20,000. These data are problematic as they are quite out of line with past professional estimates of about \$150,000 to develop a CAI course. However, the discrepancy can be reconciled by considering the way in which all of these academic respondents probably did their accounting. The typical project involves one graduate student programmer, one teaching assistant (TA), and half-time of one faculty member. The costs of the faculty member and the teaching assistant, as well as fringe benefits and overhead, were probably not included, as they are paid by the dean's office. The only "direct" costs are those of the student programmer. As the middle panel indicates that the mean development time is 22 months, and as a typical graduate student research assistant (RA) earns about \$550/month, one obtains a mean cost of a little over \$12,000, in agreement with the rightmost panel.

Alternatively, one might compute the total cost as follows: (1) salary at \$550/month for the TA as well as the RA for 22 months, (2) half-time for a \$40,000/year faculty member for 28 months (i.e., adding an extra 6 months to plan the project, write grant proposals, and purchase equipment before the students are hired), plus (3) conservative estimates of 15% fringe benefits and 35% overhead on salaries. This assessment procedure yields \$105,300, much closer to other estimates of \$150,000. As we will see in the next two figures, most of the projects surveyed are still incomplete and plan more documentation and evaluation, which would increase the total costs. Finally, I have not included salary for time from the departmental engineer, who typically helps to install the hardware, interface the peripherals, and maintain the system when it breaks down. Indeed, the expectation of \$150,000 appears to be correct.

As additional financial information, note that most of these projects received funding from multiple sources. One hundred percent were supported by the university, and an additional 80% received NSF funds. Thirty percent had private foundation support, and one project had NIE and military funds in addition to all of the other sources.

After looking at CAI costs, let us turn to CAI benefits, in developing a more complete economic picture. As most of these projects are quite new, it is somewhat premature to judge their benefits. But, the data in Figure 8 provide a start in that direction. The top left panel shows that, on the average, CAI accounts for 50% of the total course instruction. But again, the range is quite wide, from 10% to 100%. The resultant reduction in traditional teaching duties, such as grading and record keeping, will hopefully free the teacher to invest time in qualitatively superior ways, providing a teaching benefit.

The top right panel (Figure 8) shows that the average student spends about 19 h during the course with the computer terminal. The range among projects is from 10 to 40 h/student, hopefully providing a learning benefit for these students. The bottom left panel shows that about 60 students/term are enrolled in the average CAI course. The range is from 9 for a graduate course in logic to 240 for an undergraduate experimental psychology course. This individualized instruction can improve the learning environment for classes with a wide range of student abilities.

Finally, the lower right panel (Figure 8) provides some data on comparing CAI with traditional instruc-

Figure 8. (A) Percent of total instruction time that is based on CAI. (B) Number of hours students spend working on the computer. (C) Enrollment for CAI courses. (D) Frequency with which CAI is better than traditional control course on four types of measures.

tional procedures. All of the people surveyed reported CAI to be better than traditional techniques on all attributes that were evaluated. But, note that most of this evaluation so far is on student motivation rather than performance, as mentioned earlier, and, that we have some reservations about the missing sample sizes from the traditional course "control groups." Further, as observed in Figure 6, only one project reported that it was collecting data on study time, but Figure 8 shows that four reported CAI classes to exceed controls on that attribute. Thus, some of these comparisons may be based on the subjective opinions of the project directors rather than on data-based evaluations. In summary, the four panels of Figure 8 suggest that CAI can benefit the students, the teachers, and the class environment in psychology departments. But, more time is needed for a complete appraisal.

One last benefit of the CAI projects that can be examined is their benefit to other institutions: namely, their transportability. Figure 9 provides information on several factors related to transportability. The top left graph indicates that a high percentage of the respondents feel their system is transportable; 100% indicated hardware transportability and 70% indicated software transportability. But the other data in this figure suggest some strong reservations about transportability.

First, the top middle panel shows that most of the CAI projects are not self-sufficient but need human monitoring. Two projects reported that no monitoring was necessary, but four reported monitoring the CAI 100% of the time. The top right panel indicates the reasons for monitoring the system. Sixty percent of the projects are monitored because of computer crashes and bugs. In 30%, a human being makes educational decisions required for running the CAI system (e.g., see Figure 5B). Finally, 60% felt monitoring was necessary to help "confused" students use the system, to correct their mistakes, to answer their questions, and to provide

a "security blanket." The fact that 80% of the systems require human interaction places some restrictions on transportability.

The bottom panels (Figure 9) indicate another type of constraint on transportability: namely, nonoptimal documentation. If prospective CAI project directors cannot determine from the documentation they have received how the CAI system works, it is not very transportable. The bottom left panel indicates that 40% of the projects have no documentation for programmers to use in getting the system up or in making minor modifications needed in a new environment. The bottom middle panel indicates that 30% of the projects have no documentation for new teachers to consult, and another 40% indicate only "fair" documentation. In contrast, the bottom right panel shows that 100% of the projects have student-level documentation, with 70% of it rated as good or excellent. Maybe the students can help the teachers and programmers when the system fails to work. As CAI projects involve so much time and money for development, it is too bad that others will not be able to benefit easily or at all. It would be worthwhile for NSF and NIE to invest additional funds in the best of these projects to make them fully transportable. The costs would be small, relative to the large benefits to other schools for such an investment in transportability.

WORDS TO THE WISE

In addition to the structured and constrained survey items, eight open-ended questions were listed on the last page of the form. These dealt primarily with problems that the respondents had in getting their project running and with advice they would give to new project directors. Figure 10 provides a summary table of the replies that were given by two or more respondents. They could be divided into two sections: technical problems and resource problems.

The technical problems occurred at all levels: hardware, software, and courseware. I will discuss only the most severe problem, as the other problems are pretty much self-explanatory from Figure 10. Seventy percent of the projects use Apple/CORVUS computer systems and many of the users complained bitterly about the hardware and software support from Apple (e.g., see Lowe, 1983; Spivey, 1983). In past years, DEC users solved some of these types of problems by approaching high-level company officials. Regional sales managers and company vice presidents care more about their corporate image and reputation than do the local sales and service people. Thus, they will often order the local people to provide better help or send in people from the regional and national offices. Thus, I advise owners of microcomputers, particularly Apple, to use the same approach. Further, some respondents advised allocating funds to pay hardware and software consultants, instead of depending upon Apple. Probably, both the use of consultants and an attack on Apple executives would be worthwhile.

Figure 9. (A) Frequency of hardware and software transportability. (B) Frequency distribution for rate of monitoring CAI. (C) Frequency of reasons given for monitoring CAI. Frequency distributions of documentation quality for (D) programmers, (E) teachers, and (F) students.

- I. TECHNICAL PROBLEMS
 - A. HARDWARE
 - APPLE BREAKDOWNS AND REPAIRS
 - MORE GRAPHICS AND SPECIAL CHARACTERS
 - B. SOFTWARE
 - NO APPLE SUPPORT ON CORVUS OR BASIC
 - PROGRAMMERS NEED MORE HARDWARE KNOWLEDGE
 - ADD MORE SYSTEMS FLEXIBILITY EARLIER
 - C. COURSEWARE
 - BETTER DOCUMENTATION TO HELP TEACHERS
 - DIDN'T ADEQUATELY ANTICIPATE STUDENT ERRORS
- II. RESOURCE PROBLEMS
 - A. MONEY
 - UNDERESTIMATED PROGRAMMING & EVALUATION COSTS
 - B. TIME
 - UNDERESTIMATED TIME NEEDED FOR DIRECTOR, PROGRAMMERS, EVALUATORS, TEACHERS
 - C. COMMITMENT
 - INSUFFICIENT COMMITMENT OF UNIV. ADMINISTRATORS, OTHER FACULTY INVOLVED, GRADUATE ASSISTANTS

Figure 10. List of CAI problems reported by two or more respondents.

The resource problems indicated in the lower half of Figure 10 are probably much more critical to the future of CAI than are the technical problems. The three types of problems listed in the figure, money, time, and commitment, will be markedly intensified during the next few years because of the federal government cut-backs in funds for research and education. In government and university proposals for CAI development, the project directors spelled out the scientific, educational, and technical goals and plans in terms of logic and content. But, once the detailed work was underway, it became clear to all of the respondents in our survey that they had grossly underestimated the needed resources. The advice they have for prospective CAI directors is to increase their time and money estimates by a factor of 2 to 4 for every stage of development.

The less tangible resource of human commitment was even more strongly misjudged than time and money. For example, departmental and university administrators responsible for personnel replacements and for inter-departmental interactions failed in their duties. The supporting faculty resented the time taken from research and the time needed to reorganize their courses, and many had an honest case of "computerphobia." The graduate student assistants often lost interest in the project and, after a semester or a year, requested that their financial support come from a different assistantship, creating a high rate of personnel turnover. Also, course TAs often did not have the technical background to handle a CAI job.

Thus, the advice to prospective CAI directors is to extract strong commitments ahead of time, in writing at the proposal stage, not just promises and good wishes. Other sincere advice from a younger respondent was "to get tenure first." Closely related advice to administrators is to make CAI development a strong part of the "reinforcement program" used by the personnel and tenure committee. At most institutions, salary levels and

promotions are based on contributions to research, but not to teaching. In dealing with this problem, perhaps the most important line of advice from two respondents is to incorporate the CAI into an ongoing research program. As psychologists, we have been doing research on the psychological processes involved in learning, cognition, problem solving, and text processing for years. The CAI student performance records could provide further data on such processes. But unfortunately, the CAI projects are generally part of curriculum enhancement programs, and not research programs. If the CAI projects were planned to serve both functions, many of the personnel commitment, time, and money problems could be alleviated.

In closing, let me quote the advice of one respondent that should drastically reduce both technical and resource problems: "Look at what is already available elsewhere: Old wheels may squeak, but new wheels are often square." Hopefully, the user-group discussion to follow will help carry out this advice.

REFERENCE NOTE

1. Lesgold, A. M. *Authoring environments for computer-based instruction* (Tech. Rep. NIE-P-81-189). Washington, D.C.: National Institute of Education, 1981.

REFERENCES

- BELMORE, S. M. RELEASE FROM PI: Comparison of traditional and computer modules in an experimental psychology laboratory. *Behavior Research Methods & Instrumentation*, 1983, 15, 191-194.
- CASTELLAN, N. J., JR. Computers in psychology: A survey of instructional applications. *Behavior Research Methods & Instrumentation*, 1982, 14, 198-202.
- CASTELLAN, N. J., JR. Strategies for instructional computing. *Behavior Research Methods & Instrumentation*, 1983, 15, 270-279.
- CROWELL, C. R., QUINTANAR, L. R., & GRANT, K. L. PROCTOR: An on-line student evaluation and monitoring system for use with PSI format courses. *Behavior Research Methods & Instrumentation*, 1981, 13, 121-127.
- ECKERMAN, D. A. Developing a laboratory course using PASCAL on the Apple II. *Behavior Research Methods & Instrumentation*, 1981, 13, 204-208.
- FAZIO, R. H., & BACKLER, M. H. Computer lessons for a social psychology research methods course. *Behavior Research Methods & Instrumentation*, 1983, 15, 135-137.
- HOLT, E. M. Problems in the evaluation of PLATO lessons. *Behavior Research Methods & Instrumentation*, 1983, 15, 155-157.
- HOLT, R. W. Applications of the PLATO computer system to research in social psychology. *Behavior Research Methods & Instrumentation*, 1983, 15, 145-147.
- LESGOLD, A. M. Computer games for the teaching of reading. *Behavior Research Methods & Instrumentation*, 1982, 14, 224-226.
- LEVY, C. M. Microcomputer-based tutorials for teaching statistics. *Behavior Research Methods & Instrumentation*, 1983, 15, 127-129.
- LEVY, M. C., & DURNIN, M. W. Modeling on microcomputers. *Behavior Research Methods & Instrumentation*, 1981, 13, 201-203.

- LOWE, D. C. A Winchester hard-disk integrated computer-assisted instructional laboratory: Hardware and data management considerations. *Behavior Research Methods & Instrumentation*, 1983, **15**, 181-182.
- MAIN, D. B., & HEAD, S. Computer simulation in the elementary psychological laboratory. In *Proceedings of the Conference on Computers in the Undergraduate Curricula*. Hanover, N.H.: Dartmouth College, June 1971.
- PAVEL, M., MARCOVICI, S., SHERMAN, A., & FALMAGNE, J.-C. ARIS: A computer-assisted instruction system. *Behavior Research Methods & Instrumentation*, 1983, **15**, 138-141.
- PSOTKA, J. Computers and education. *Behavior Research Methods & Instrumentation*, 1982, **14**, 221-223.
- SCHAFFNER, P. E. A self-paced CAI package for selecting statistical methods. *Behavior Research Methods & Instrumentation*, 1983, **15**, 130-134.
- SIMUTIS, Z. M., & BARSAM, H. F. PLATO: Army applications. *Behavior Research Methods & Instrumentation*, 1983, **15**, 148-154.
- SPELT, P. B. Bridging the gap between micros and minis in the teaching laboratory. *Behavior Research Methods & Instrumentation*, 1981, **13**, 198-200.
- SPIVEY, J. E. Software development for computer-assisted instruction in experimental psychology. *Behavior Research Methods & Instrumentation*, 1983, **15**, 183-186.
- SPIVEY, J. E., & JACKSON-SMITH, P. Computer simulation: A case of conflict. *Behavior Research Methods & Instrumentation*, 1983, **15**, 187-190.
- WILLIAMS, M., & HOLLAN, J. An overview of STEAMER: An advanced computer-assisted instruction system for propulsion engineering. *Behavior Research Methods & Instrumentation*, 1981, **13**, 85-90.