Software Review

The Sorcerer's Apprentice: A Review of
Hardy's *Behavior Analysis: A Computer-based Tutorial*

Stephen A. Graf
Youngstown State University

In Walt Disney's 1940 animated production of Paul Dukas's *The Sorcerer's Apprentice* (Culhane, 1983), one sees how much trouble can ensue from attempts to program an easier way to complete a repetitive task. Reed Hardy has attempted to get a computer program to take over the repetitive task of teaching behavioral concepts and terminology to individuals who need such a background in work or educational environments. The three units he has created so far cover basic history/ethics, respondent conditioning, and operant conditioning.

The endeavor is heroic, as anyone who has attempted to author courseware will attest (e.g., McDade et al., 1985; Chute, 1987; West et al., 1985). Spurned by several software publishers, Hardy has offered the program to the behavioral community as "shareware." "Shareware" means that the author will send the program to those that request it and users contribute what they think the program is worth after trying it out.

As Hardy denotes in the documentation, computer courseware tutorials have several advantages over printed texts. First, learners can't continue the program until a certain level of performance is achieved on embedded content questions. Next, visual and auditory sound effects can be used as stimuli to precede or follow responding. Also, the names of the learner and friends can be incorporated into the program to personalize the interactions between program and person, and finally, courseware can ensure that the student has "mastered" its contents.

With the advantages, however, comes another feature. When you use the program, you commit the learners to the philosophy and approach of the author. With a book, one can choose to ignore or disagree with the author on certain points. With courseware, you no longer control what the learner must learn; control resides with the program and the author. Our idiosyncratic behavioral histories may thus leave us unwilling to accept the wares of another sorcerer's apprentice. We may want our own.

This software review focuses on the content, instructional design, and program utility of *Behavior Analysis*, following closely the guidelines suggested for evaluating software detailed by Vachon and Carnine (1983). The comments reflect my own experiences on the tutorial as well as the comments of ten undergraduates at Youngstown State University who critiqued the program as part of a course requirement.

Content

Is the content accurate and of educational value? I reacted to some word choices that reflect my own behavioral history. I cringed when I read ". . . the very definition of reinforcement helps us to keep an open mind about the effects of different consequences." I think behaviorists should follow the suggestion of Malott and Whaley (1976) and avoid reifications such as "mind," even in everyday speech.

Hardy's use of the labels "positive and negative reinforcement and punishment" clash with Lindsley's more compelling terms "reward, relief, punishment, penalty" (Lindsley, 1983), but the former are still in common use.

The value of respondent conditioning as one of the three units covered might be questioned, in view of the target audience inclusion of mental health staffs. Overall, accuracy and educational value seemed acceptable.

Instructional Design

Objectives. Are objectives clearly stated? No, the objectives are identified by Hardy as content areas for deciding whether learners may proceed, but these objectives are not differentiated from the rest of the text.

Individualization. Is a pretest provided? Is there a method of record keeping? Are options for exiting and returning to the menu available? A placement test would allow a variety of entry points into the program based on the learner's repertoire. In the current version, everyone starts at the same point. Also, record-keeping features are still under development and not available, so the instructor has no information available on where learners are in the program, nor on which items learners are having problems. Options for exiting or returning to the menu are available but didn't work reliably according to the some of the people who used the program.

Presentation. Does the tutorial provide review? Review was provided but much of the time as an option with restrictive alternatives. One learner pointed out that in the operant unit, one could not skip the shaping game when going through a review of the unit. The game had been done the first time and the individual judged it a waste of time the second time through.

Concepts (discriminations and relationships). While the program provided examples, it lacked non-examples and range variations through examples as described by Engelmann & Carnine (1982). Minimally different non-examples sequenced to demonstrate differences would have been useful but did not appear. The preskills required as entry competencies could have served to select users ready to succeed in the program from users who needed other skills before attempting the program.

Strategies. The program does not list component skills, does not present a strategy as a series of steps, and does not guide the learner in responding to each step. While less specific direction in the form of hints and clues occur to some extent as a learner progresses, the program doesn't provide sufficient independent practice with a range of appropriate tasks.

Simulations. A shaping simulation occurs in the operant conditioning unit in which the learner attempts to succcessfully reward a graphic rat's approximations to pushing the picture of a bar. While I found the program acceptable, an unfortunate latency occurs between the pushing of the space bar by the person at the keyboard and the translation of that into the conditioned rewarding stimulus.

Feedback. Errors on the embedded questions are done again until corrected, with the prompt to read more carefully if several errors occur. Errors on the unit tests are not remediated specifically. No correction occurs on the unit tests unless a critical percentage of items are missed. If this occurs, the correction doesn't fit the context in which the errors occurred; the learner must repeat the entire unit up to the point of the quiz. So informative feedback does not occur to any quiz items. Several people criticized this feature, e.g.: "You couldn't page back and get feedback or relearn what you didn't know in a certain area. Essentially you didn't know what you didn't know!"

Review. Review occurs for newly acquired skills, but usually in the form of quizzes and less often as a part of the program. The quizzes detract from the program, rather than enhance it. I suggest it would be better to incorporate embedded review items and remediate specific errors of those items, rather than to withhold feedback from the learner for the sake of protecting quiz items. This feature interacts with the current unfinished status of record-keeping. Ideally, items producing high error counts would signal the author that something about the teaching sequence was faulty, and that

that portion of the program needed revision. The alternative, that the student's failure to learn was his or her fault, typifies what Engelmann (1968) has called the "floating standard" in education: if the learner succeeds, the school and teacher (or in this case the tutorial program) receive the credit; if the learner fails, the school and teacher (or tutorial) bear no responsibility. This fault is assumed to lie within the learner.

The ideal tutorial or teacher would provide a "faultless communication" in the teaching sequence such that the learner can form only the appropriate interpretation of what is being taught (Engelmann & Carnine, 1982).

Motivation. The level of difficulty of *Behavior Analysis* will likely challenge anyone to some extent, but the lack of entry-level assessment requires one to assume that the learners are alike in the status of their repertoires. The tutorial is self-paced, except for the quizzes. On the quizzes the learner is told that only a certain amount of time (unspecified) will be allowed for each item and failure to answer in the allotted time will result in a miss for that item. Three of ten student reviewers mentioned this as a feature that should be changed. As one user put it, "I felt 'rushed' on quizzes, not knowing how much time one had and may have misanswered thinking I would run out of time before I could answer. Not knowing how much time I had, I felt my first objective was to beat the clock, not understand the concept presented."

I suggest that the program combine the latency and duration on each embedded question, labeling this as the time taken by the learner. Correct counts and error counts for items and concepts divided by the time would provide the frequency correct and the frequency incorrect (Pennypacker, Lindsley, & Koenig, 1972). Such frequencies could be used to gauge both the pace of the learner and the difficulty of any particular item or concept.

Program Utility

Is the program easy to operate? It could be made considerably more "user com-

fortable." The introduction could be shortened. Four pages of information about the system serve no function past the first viewing. No user will read it on subsequent times through, so what happens is that the return key is pressed repeatedly until one is forced to make a more careful response. Too much time is consumed getting started. The particular system is slow enough without making the situation worse. Better to have a quick entry into where one left off than to make the startup aversive. Overcome the inertia and get the learner on the system. Then maybe they'll find the rewards. They won't when they're not plugged in, and they're not likely to plug in for a quick five minute run when it takes five minutes to get logged on.

Is the program reliable under normal use? The program requires that DOS be booted first, and this may require an additional disk for the student besides the two program disks. Several unreliable aspects of the program appear to be 1) the password feature 2) the optional printout near the end of the operant conditioning unit. Over half the users in my sample, including myself, experienced problems with the password. This was a hassle which, if experienced, would occur each time one tried to log on. The log-on time was consequently in the five-to-eight minute range. This appeared to be a "bug" and is unrelated to the type of program design considerations that have been considered to this point. At first I thought that this was probably an example of a faulty communication in which something not explicitly forbidden by the instructions turns out not to work if selected, but subsequent efforts to produce "perfect" passwords still produced the problem.

The printout option allows one to receive a hardcopy of definitions and explanations of terms. The hardware requirements mentioned do not include a printer, but if one is not attached (as is the case in the networking arrangement in use at Youngstown State University), the program crashes regardless of whether you request a printout or not. Consequently, you lose credit for your pro-

gression through the program from the point at which you logged on, and you must redo the admittedly long unit again if that's where you started. Several students tried unsuccessfully to cope with this problem and never reached the quiz on the operant unit which follows the printout option.

Some type of graphics card is also necessary, either a Color Graphics Adaptor (CGA) or Extended Graphics Adaptor (EGA). If not installed, the graphics displays do not appear and the simulation is a jumble of text characters instead of an animated rat. Also, the text reduces to regular size, is underlined, and occupies only the left half of the screen.

At spots throughout the program, the screen will display "press return to continue" and then the program changes the screen before you press the key. I found this mildy disconcerting.

Can the program analyze a variety of responses? Limitations occur. For example, when a correct answer was FR10, the program did not accept FR as partially correct. This could be the particular program rather than the authoring system used. The preponderance of responses are simply pressing the return key. In the classical conditioning unit, one presses the return key 26 times before any other response is necessary.

Are information displays attractive? Yes, but I see why form should follow function. Besides having seen the text in its attractive colorful, big-letter format, I also saw it in somewhat more normal-looking text—this is what happens if you run the program on a machine without a Color Graphics Adaptor card. My hunch is that the latter is easier on the eyes and can be read more quickly as well.

Additional Comments

The tutorial received the following commendations. Unless otherwise noted, each comment represents one and only one student.

Instructions were clear and easy to follow. The graphical representations were mentioned as being helpful. Several students reported enjoying the overall experience. One individual mentioned lik-

TABLE 1

Courseware description

Program Name: *Behavior Analysis: A Computer-based Tutorial*

Publisher: Reed Hardy, St. Norbert College, De Pere, WI 54115-2099

Required Equipment:
Hardware: IBM PC or compatible computer with minimum 256 K of RAM and two disk drives
Software: Version 2.3 comes on two 5.25″ floppy disks
Peripherals: Color Graphics Card required (CGA or EGA); color monitor optional

Package Materials: Instruction sheets, disks

Unit Price: As "Shareware," user sends contribution directly to developer

Subject Area: Behavior Analysis
Topics: Ethics, respondent conditioning, operant conditioning

Target Audience: High school and college students, parents, hospital and mental-health staffs, clients, business managers

Mode of Interaction: Tutorial __×__; Simulation __×__; Drill & Practice/Gaming ____

Program Description: *Behavior Analysis* covers basic definitions found in classical and operant conditioning literature, all of the basic schedules of reinforcement and graphic presentation of behavioral data. An animated simulation of shaping by successive approximations is included. Progress is based on the student's ability to answer multiple-choice and fill-in questions on unit quizzes. The program has been designed for people who need to quickly acquire a working knowledge of behavioral techniques and the specialized vocabularay that covers those techniques.

ing the personalized aspects such as name, height, name of friend, etc., but would have liked for the author to have provided the same information about himself! Another student felt that these questions for personal information were neither necessary nor useful. Two people reported that they found the unit on ethics interesting and a "good background."

The main criticism of the program is that certain instructional design features reduce its potential effectiveness.

Some problems in the functioning of the program exist independently of program design. Complexities in software authoring invariably lead to "bugs" which an author has not uncovered but can ameliorate if alerted to them.

Behavior Analysis is not a "slick" prod-

uct. Rather, it represents the first step in an effort to provide a needed service: effective teaching in environments where competency with key concepts is crucial to the success of the program. Reed Hardy has already demonstrated a commitment to effort and he has offered the rest of us the product of his effort at a very low cost. As a "sorcerer", he has worked hard in producing this "apprentice". As a fellow sorcerer, I have attempted to provide some directions that I'd go in if I were wearing his felt boots. Godspeed and high frequency, Reed!

REFERENCES

Chute, D. (1987). Glimpses of the darker side of academic computing. *Academic Computing, 2* (1), 40–41, 58–59.

Culhane, J. (1983). *Walt Disney's Fantasia.* New York: Abrams.

Engelmann, S. (1975). *Your child can succeed.* New York: Simon & Schuster.

Engelmann, S., & Carnine, D. (1982). *Theory of instruction: principles and applications.* New York: Irvington.

Lindsley, O. R. (1983). Say reward, relief, punishment, or penalty. *Journal of Precision Teaching, 3* (4), 100–101.

Malott, R. W., & Whaley, D. M. (1976). *Psychology.* New York: Harper & Row.

McDade, C. E., Austin, D. M., & Olander, C. P. (1985). Technological advances in precision teaching: a comparison between computer-testing and SAFMEDS. *Journal of Precision Teaching, 6* (3), 49–54.

Pennypacker, H. S., Lindsley, O. R., & Koenig, C. H. (1972). *Handbook of the Standard Behavior Chart.* Kansas City, Kansas: Precision Media.

Vachon, V., & Carnine, D. (1983). Evaluation of computer software. *Association for Direct Instruction News, 3* (2), 11–15.

West, R. P., Young, K. R., West, W. J., Johnson, J. I., & Freston, C. W. (1985). *AC CEL: An expert system for student performance evaluation and instructional decision-making.* Great Falls, Mont.: Precision Teaching Materials and Associates.