Behavior Research Methods, Instruments, & Computers
1994, 26 (4), 421–426

SYNWORK1: A PC-based tool for assessment of performance in a simulated work environment

TIMOTHY F. ELSMORE
Walter Reed Army Institute of Research, Washington, D.C.

SYNWORK1 is a computer-based performance task that requires subjects to work simultaneously on four distinct subtasks involving memory, arithmetic processing, and visual and auditory monitoring. Difficulty levels, the payoff matrix, feedback levels, and component subtask mix are user selectable. Detailed data are automatically collected, and a suite of data analysis programs is available. SYNWORK1 is being used in sleep-deprivation and circadian desynchronization experiments and in a variety of clinical research applications. Representative data from a sleep-deprivation experiment are presented to demonstrate the sensitivity of the technique. The strategy used for programming concurrent tasks on a PC is described.

The usual approach to performance assessment has been to break performance into component abilities required to perform tasks and then assess effects of variables of interest on these component abilities. Numerous computerized performance assessment batteries (PABs) of this sort have been devised (AGARD, 1989; Anger, 1990; Englund et al., 1986; Kennedy et al., 1981; Perez, Masline, Ramsey, & Urban, 1987; Thorne, Genser, Sing, & Hegge, 1985), and they are currently being used in a variety of situations. Kane and Kay (1992) provide an excellent review of the use of computerized performance tests in neuropsychology, as well as in more general applications. It has been suggested that the component-abilities approach fails to deal adequately with situations that require simultaneous attention to multiple tasks. Chiles (1982) pointed out that the area of multiple tasks has been largely ignored in performance assessment:

> It is an unfortunate, but inescapable, fact that a body of established, reliable data does not exist in this area of the behavioral sciences. Most of the published laboratory research findings on tasks like those found in operational systems have little or no potential relevance, because the tasks were performed alone, as single tasks by themselves without time-sharing requirements. (Chiles, 1982, p. 51)

Typical test batteries are designed to *isolate* aspects of performance, thus losing most of the dynamism and complexity found in job situations.

Another approach is to build performance-testing systems that more closely approximate actual tasks or systems. The most prevalent of these are full-blown system simulators (e.g., flight simulators) or "part" simulators that yield highly specific data for the system that is being simulated. However, the generality of these data may be questioned. In addition, systems such as these are usually not designed for research purposes; they are expensive to acquire and use, and existing systems are usually dedicated to training rather than to research.

A somewhat more manageable research approach is one in which researchers use "synthetic work tasks" (Alluisi, 1967; Chiles, 1982) in an attempt to simulate functional aspects of a situation, sacrificing the structural fidelity that is sought in simulators. These tasks typically require simultaneous attention to a montage of tasks that appear to require behavior similar to that of real jobs. Chiles, Alluisi, and Adams (1968) described a Multiple Task Performance Battery (MTPB), in which various combinations of six tasks could be simultaneously presented. The original MTPB was a one-of-a-kind hardware device (as have been other synthetic-work consoles; see Hartman, Benes, & Storm, 1980) and was therefore limited in its application. The ready availability of powerful, standard personal computers in recent years has enabled the construction of PC-based synthetic-work tasks with the potential for broader application. In this article one such program is described, and representative data from a laboratory study on sleep deprivation are presented to illustrate its application.

The author was supported by the Office of Naval Technology, under work units 63706NM0096.002-6201 and 62233N MM3P30-002-6005, and Program Executive Officer Surface Ship ASW Systems Task SSAS-91-77A01R2. The views expressed in this article are those of the author and do not reflect the official position of the Department of the Navy, Department of the Army, Department of Defense, or the U.S. Government. This program was written while the author was a visiting scientist at the Naval Health Research Center. The author wishes to acknowledge the advice and guidance of F. W. Hegge and P. Naitoh during this project. Reprints may be obtained from T. F. Elsmore, Division of Neuropsychiatry, Walter Reed Army Institute of Research, Washington, DC 20307-5100. Software may be obtained at no cost by sending the author a formatted, floppy disk of any size, or by downloading it from the electronic bulletin board system maintained by the Office of Military Performance Assessment Technology (OMPAT). Contact the author at (202)576-3006 to become a registered OMPAT BBS user.

THE SYNWORK1 PROGRAM

SYNWORK1 was designed as a prototype for the implementation of synthetic work tasks on personal computers. It contains two characteristics of real work environments that are commonly lacking in other computer-based tests of performance: the presentation of concurrent tasks and explicit assignment of outcomes for component task performance. The program was not designed to simulate any particular job. It does, however, contain elements common to a number of watch-standing jobs. The program allows researchers to readily manipulate the difficulty of component tasks and the payoff matrix for component task performance. It also allows component tasks to be presented singly or in any combination. The program is entirely mouse based, which permits subjects' attention to focus on the display screen and eliminates variability due to intersubject differences in typing skill.

During a test session, the PC screen is divided into four quadrants, or "windows," each assigned to a different subtask. The screen's background is dark blue, and double magenta lines delineate the windows. Boxes within windows are drawn in light blue, and informational text is displayed in light red. Figure 1 illustrates the screen.

Feedback

The small window in the center of the screen is used for displaying a composite "score" for performance on all of the subtasks. The subject is instructed to maximize this score. In addition, at the user's option, a "score bar" may be displayed in each window to illustrate graphically the cumulated score on the subtask for that window. Throughout the session, auditory feedback is provided. Correct responses (i.e., those producing points) are followed by a high-pitched "squeaking" sound, and errors are followed by a low-pitched "burping" sound.

Component Tasks

In the following text, default values for adjustable parameters will be indicated in square brackets—[parameter]—when describing component tasks. These parameters can be changed at run time through the use of command-line arguments.

Upper left window: Memory subtask. For each session, a list [6] of letters (the "positive list") is chosen from a subset of the alphabet (the letters C, D, M, Q, and V are not used since they are difficult to distinguish from other letters on some systems) and displayed in bright, white uppercase letters in a box at the top of the window. The positive list is displayed for only 5 sec, after which it is replaced by the words "Retrieve List." When this message is displayed, clicking the mouse on the list box results in display of the list for another 5 sec. A point penalty [10] is charged for each list retrieval. An equal-sized list (the "negative list") is also selected at the start of each session. Following each intertrial interval [20 sec], a sample letter is displayed in the box in the center of the window. The subject's task is to indicate, by clicking the mouse on either the "Yes" or the "No" box at the bottom of the window, whether or not the letter is a member of the positive list. The sample disappears as soon as either a correct response or an error is made. Points [10] are awarded for each correct response and are deducted [10] for each error.

Upper right window: Arithmetic subtask. An addition task presents two or three [2] randomly selected numbers less than 1,000. The numbers are displayed in bright, white characters. The subject's task is to adjust the answer by clicking on "+" and "−" boxes below each character of the answer, which is initially set to "0000." Clicking on a "Done" box at the bottom of the window results in the presentation of a new problem, as well as the addition of points [10] for correct answers, and the deduction of points [10] for errors. There are no time limits for completion of this task.

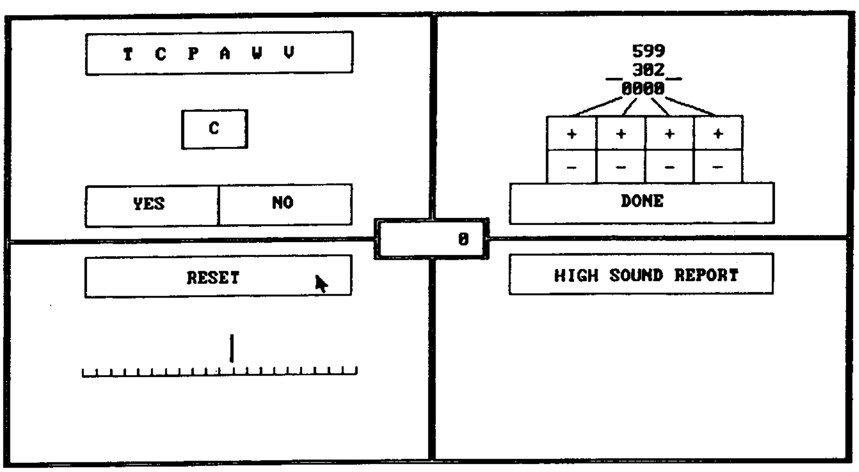

Figure 1. The SYNWORK1 screen display.

Lower left window: Visual monitoring subtask. A bright, white pointer moves from the center of a 200-pixel scale, toward either end, at a steady rate [5 pixels/sec]. Clicking the mouse on a box labeled "Reset" at the top of the window resets the pointer to the center. The subject's task is to prevent the pointer from reaching the end of the scale. Points are awarded for each reset according to the following formula:

$$\text{PointsAwarded} = \text{INT}\left(\text{MaxPoints} \cdot \frac{\text{DistFromCenter}}{\text{MaxDistance}}\right)$$

The default maximum point value is 10. Thus, the closer the pointer is to the end of the scale at reset, the more points are awarded. Points [10] are deducted when the pointer reaches the end of the scale and remains there for 1 sec, and additional points [10] are deducted for each additional second that the pointer remains at the end of the scale. Thus, the penalty for allowing the pointer to reach the end is severe.

Lower right window: Auditory monitoring subtask. Periodically [5 sec], a brief tone is sounded over the PC speaker, or over headphones or external speakers if a Sound Blaster card is being used. The tone is either low [931 Hz] or high [1234 Hz] frequency. The subject's task is to click the mouse in a box at the top of the window labeled "High Sound Report" following a high tone. High tones occur with a given probability [.2]. Correct responses are those that occur after a high tone, prior to the next scheduled tone. All other responses are incorrect. Points are awarded [10] for each correct response and are deducted [10] for each error.

Training

Initial training is accomplished by first exposing the subject to each of the four subtasks independently for 2–4 min; each is preceded by a single screen of explanation. Immediately following individual subtask presentations, all four subtasks are presented simultaneously for a brief, 5-min session. A batch program is provided to present this training sequence. During this training, an experimenter is present to answer questions regarding program operation. Usually, no instructions are given regarding strategies to be used in performing the task. Following initial training, no additional instructions are usually required. In most studies, 6 15-min sessions are sufficient to achieve near-maximal performance, although performance typically shows gradual improvement for up to 20 sessions.

Technical Details

Programming. The SYNWORK1 program was programmed using Microsoft QuickBASIC 4.5 and augmented with shareware subroutines for mouse and file handling[1] and public domain routines for timing (Graves & Bradley, 1988, 1991). These timing routines provide accuracy to ± 0.1 msec, though actual timing precision may be as poor as ± 3 msec, depending on the processing speed of the computer. Even the worst case, however, is more than an order of magnitude better than the ± 55 msec that is obtainable with the BASIC "timer" command. The random-number generator is seeded with the session number at the beginning of each session, and to assure repeatability for different subjects, all stimulus sequences are generated prior to the start of the session. To maximize its utility as a research tool, SYNWORK1 allows user modification of many task parameters through the use of command-line parameters.

The structure of SYNWORK1 is based on the idea of parallel sequential networks, or "state diagrams," as employed in the SKED programming language for behavioral experiments (Snapper & Kadden, 1973). Thus, within the program, each subtask operates as an independent subroutine in a polling loop, maintaining its own timers, counters, and pointers. Elapsed times are obtained by subtracting values obtained from sequential calls to the timing routine, which returns time, in tenths of a millisecond, from midnight. For example, in the visual monitoring subtask, the time of a pointer movement is saved in a variable. On subsequent calls to the visual monitoring subroutine, the current time is subtracted from the saved value to determine whether it is time for the next pointer movement. If it is, the pointer is moved, and the current time is saved. Thus, any number of independent timers may be implemented.

Each subtask is structured as a sequence of states; one state is for subtask initialization (screen, task parameters, and stimulus sequences) and there are separate states for each definable condition within the subtask. Again using the visual monitoring subtask as an example, separate states exist when the pointer is in motion and when it is at the end of the scale. Each time a subtask subroutine is called, on the basis of the value of the state pointer for that subtask only the code for the current state is executed. The main decisions to be made on each pass must take into account answers to the following questions: (1) Has a critical time expired? and (2) Has a mouse click occurred in a critical area? In most cases, the answer to both of these questions is "no," resulting in rapid completion of subtask processing on each pass through the loop. When the answer to either of these questions is "yes," additional code is executed to handle the situation. This strategy could be implemented in most programming languages and should be generally useful for programming other types of parallel tasks.

During sessions, SYNWORK1 collects data by time stamping all critical events (stimulus changes, mouse clicks) and storing the data in a large memory array. At the end of sessions, data are written to data files coded with the subject's ID and session numbers.

Hardware requirements. SYNWORK1 requires an IBM-compatible personal computer, a Microsoft-compatible mouse, and an EGA or better display. Since all timing is done with hardware that is standard on all IBM-compatible PCs, no add-in timer board is required. Auditory stimuli may be presented either with the standard PC speaker, or over headphones or external speakers using a Sound Blaster sound board (Creative Labs Inc., Santa Clara, CA).

Data Analysis Programs

Three programs are provided for SYNWORK1 data analysis. SYN1ANAL provides data summaries for the program as a whole and for the component tasks. Results may be displayed on the screen, written to summary data files, or printed on the system printer if one is available. A second program, SYN1SS, creates or appends selected summary data to an ASCII data file in a format suitable for importing into statistical packages for analysis or graphing. A third program, SYN1CUM, generates cumulative response records of individual sessions, permitting inspection of minute-to-minute changes in performance. This program provides output either to the screen or as a file of HPGL (Hewlett Packard Graphics Language) commands, which can be used with a variety of graphics programs to produce hard-copy output.

SLEEP-DEPRIVATION EFFECTS ON SYNTHETIC WORK

As part of a broadly based performance assessment system, SYNWORK1 was incorporated into a battery of performance tests that were administered to control subjects to study the effects of naps and stimulant administration during 64 h of sleep deprivation.[2] The subjects received no treatment during the course of the experiment. The data from this study demonstrate SYNWORK1's performance sensitivity in studies of fatigue.

Method

Subjects and Apparatus. The subjects were 9 male enlisted personnel in either the Navy or the Marine Corps, recruited from a variety of sources. Most had completed high school, but none had completed 4 years of college. Four IBM/AT- compatible computers (Unisys 386, 20-MHz, color VGA monitor) were used in this study. The computers were located in a single room, with testing stations separated by low partitions. One to 3 subjects were tested simultaneously.

Testing Protocol

Two training sessions were conducted on the subjects' first day in the laboratory. Beginning at 0900 on the next day (Day 1) and continuing until the end of the experiment, a battery of computer-based performance tests was administered every 3 h. The battery required approximately 70 min to complete, and SYNWORK1 was the last test of the battery. SYNWORK1 sessions were 15 min in duration.

Results

In Figure 2, the average group composite score (i.e., the overall score for the session), ±*SEM*, is plotted for each of the 23 sessions in the study. Performance continued to improve for the first 6 sessions, reaching a maximum by the 1800 h session on Day 1. Performance dipped markedly on Days 2 and 3, with minimum scores during the 0600 h session on both days. A one-way repeated measures (day and time of day as repeated factors) analysis of variance for scores on Days 2 and 3 (Sessions 8–23) showed significant effects for both day [$F(1,8) = 10.98, p < .02$] and time of day [$F(2.3,18.2) = 3.44, p < .05$].[3] Thus, performance continued to deteriorate throughout the entire period of sleep deprivation, with the lowest scores appearing in the early morning hours and recovery evident later in the day.

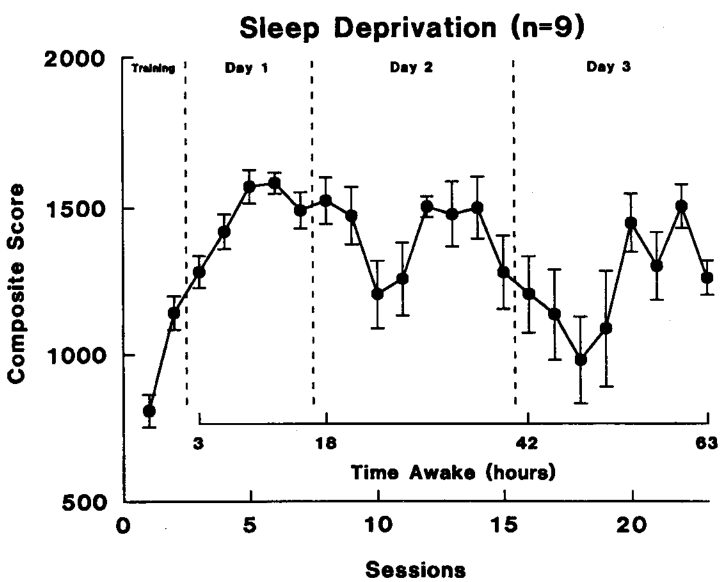

Figure 2. Mean composite score for 9 subjects, ±*SEM*, for each session of the experiment. Subjects were not permitted to sleep from the beginning of Day 2 until the end of the experiment.

The rate at which the mouse button is clicked may be taken as an index of the subject's level of exertion. This measure is plotted in Figure 3 in terms of responses per second. There was a very strong correlation (.929) between the response rate curve and the composite score curve. However, an analysis of response rates from Days 2 and 3 did not show a statistically significant effect of day on rate, but did show an effect of time of day [$F(2.4,19.1) = 3.60, p < .40$]. Thus, although the circadian rhythm continued to exert its effect on response rate throughout the sleep-deprivation period, overall response rate did not deteriorate as much as the composite score. In fact, the highest average response rate occurred after 60 h of sleep deprivation.

DISCUSSION

The data presented here demonstrate that performance on the SYNWORK1 multitask synthetic work program is sensitive to sleep deprivation. The composite score declined as time without sleep increased; the decrement was most evident in morning sessions, and recovery followed later in the day.

More detailed measures of performance appeared to be less sensitive to sleep deprivation than the composite score. However, this seems to be a reflection of differences between subjects in terms of how they coped with the task—that is, individual differences in strategy. SYNWORK1 was explicitly designed to permit observation of this type of variation, since this is an aspect of real-world job performance that is not measured in traditional PABs. Of the four subtasks within SYNWORK1, response omission was penalized on only one subtask—visual monitoring. Thus, when severely sleep deprived, 3 of the 9 subjects concentrated their efforts on this subtask and devoted fewer resources to the other component tasks. These effects were manifested as local breakdowns in performance. Presumably, more consistent performance could be maintained by providing tighter contingencies for all subtasks (e.g., point penalties for response omissions).

The reliability of SYNWORK1 cannot easily be assessed in the present study, since there was a relatively small number of subjects as well as continually increasing sleep deprivation. Correlations between composite scores on pairs of successive sessions were computed for the period of time when performance was near maximal and before significant sleep deprivation. The correlations for Session Pairs 5–6, 6–7, and 7–8 (from 1500 h to midnight) on Day 1 were, respectively, .68, .47, and .83. In another study involving adaptation to night-shift work, 5 college students were tested once every hour during the day in an extremely well-controlled, residential laboratory setting. On the second day of this procedure, session-to-session correlations of composite scores ranged from .76 to .99 for 5 subjects during a block of 10 sessions, averaging .92. Thus, given stable testing conditions, SYNWORK1 can produce highly reliable data.

Using SYNWORK1, Savu (1991) demonstrated that there were clear interactions among the component tasks when subtask difficulty was manipulated. In particular, he demonstrated a significant deterioration in performance on the arithmetic task when pointer speed was in-

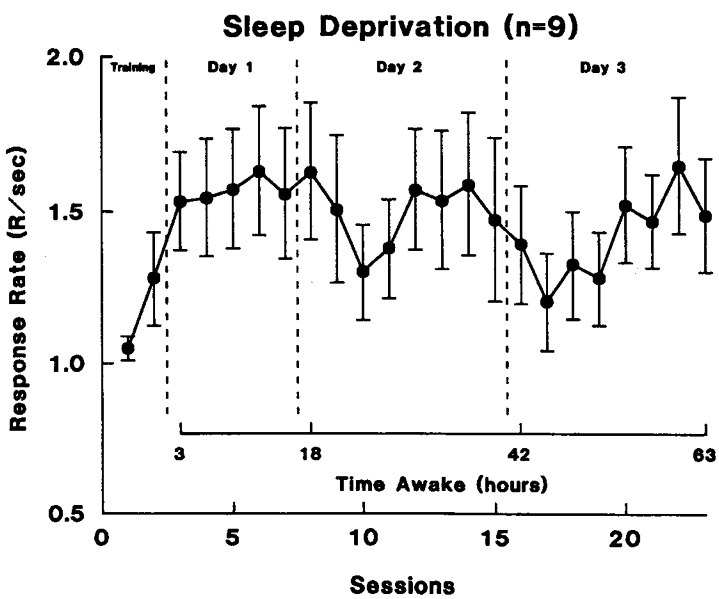

Figure 3. Mean rate of mouse clicks for 9 subjects, ±*SEM*, for each session of the experiment.

creased in the visual monitoring task. Similarly, as suggested by the substantial literature on multiple-task performance (e.g., Chiles, 1982), it is likely that other manipulations, such as an increase in the difficulty of the math problems or changes in the payoff matrix, might produce similar interactions.

Because of its game-like quality, subject acceptance of SYNWORK1 is excellent. For example, the subjects in the sleep-deprivation experiment universally preferred SYNWORK1 over other tests in the performance assessment battery. In a setting in which data were collected during a military operation in the field, this feature was useful for maintaining subject interest, thus aiding compliance with the data collection schedule (Elsmore, Naitoh, & Linnville, 1992).

SYNWORK1 can serve as a research framework for investigating variables that may degrade performance of complex, multicomponent jobs. Clearly, if system-specific information is desired, other, more targeted synthetic work tasks, part-task simulators, or high-fidelity simulators will be required. Traditional PAB tests tell us about the effects of independent variables on the elements of job performance. For many purposes, particularly those situations in which the emphasis is on the individual rather than on a system or job, PAB tests may be the method of choice. Synthetic work tests such as SYNWORK1 provide a viable supplementary or alternative approach, focusing on total job performance by adding the common features of real-world tasks, contingency and concurrency, to the testing situation. The program is currently being used in a variety of settings to assess the effects of drugs, the status of HIV-infected individuals, treatments to minimize temporal desynchronization (jet lag), and levels of fatigue in medical and transportation personnel. The procedure should also be useful for other settings requiring repeated assessment of complex performance.

This approach to programming concurrent tasks on a PC can be easily adapted to other computer systems and performance testing situations. The technique simplifies the development of programs for assessment of fitness for duty and is useful for a variety of clinical and educational applications.

REFERENCES

AGARD. (1989). *Human performance assessment methods* (AMP Working Group 12 and AGARD Lecture Series 163, AGARDograph No. 308, NATO).

Alluisi, E. A. (1967). Methodology in the use of synthetic tasks to assess complex performance. *Human Factors*, **9**, 375-384.

Anger, W. K. (1990). Worksite behavioral research: Results, sensitive methods, test batteries, and the transition from laboratory data to human health. *Neurotoxicology*, **11**, 629-720.

Chiles, W. D. (1982). Workload, task, and situational factors as modifiers of human performance. In E. A. Alluisi & E. A. Fleischman (Eds.), *Human performance and productivity: Stress and performance effectiveness* (pp. 11-29). Hillsdale, NJ: Erlbaum.

Chiles, W. D., Alluisi, E. A., & Adams, O. S. (1968). Work schedules and performance during confinement. *Human Factors*, **10**, 143-196.

Elsmore, T. F., Naitoh, P., & Linnville, S. (1992). *Performance assessment in sustained operations using a computer-based synthetic work task* (Naval Health Research Center Tech. Rep. No. 92-30). San Diego, CA: Naval Health Research Center.

Englund, C. E., Reeves, D. L., Shingledecker, C. A., Thorne, D. R., Wilson, K. P., & Hegge, F. W. (1986). *The Unified Tri-Service Cognitive Performance Assessment Battery (UTC-PAB): Design and specification of the battery.* (JWGD3 MILPERF Rep. No. 86-1). Frederick, MD: U. S. Army Medical Research and Development Command.

Graves, R. [E.], & Bradley, R. (1988). More on millisecond timing and tachistoscope applications for the IBM PC. *Behavior Research Methods, Instruments, & Computers*, **20**, 408-412.

Graves, R. E., & Bradley, R. (1991). Millisecond timing on the IBM PC/XT/AT PS/2: A review of the options and corrections for the Graves and Bradley algorithm. *Behavior Research Methods, Instruments, & Computers*, **23**, 377-379.

Hartman, B. O., Benes, R. A., & Storm, W. F. (1980). *A review of USAFSAM studies employing multiple-task performance devices* (Rep. No. SAM-TR-8016). Brooks Air Force Base, TX: USAF School of Aerospace Medicine.

Kane, R. L., & Kay, G. G. (1992). Computerized assessment in neuropsychology: A review of tests and test batteries. *Neuropsychology Review*, **3**, 1-117.

Kennedy, R. R., Bittner, A. C., Carter, R. C., Krause, M., Harbeson, M., McCafferty, D. B., Pepper, R. L., & Wilker, S. F. (1981). *Performance Evaluation Tests for Environmental Research (PETER): Collected papers* (Rep. No. NBDL-80R008). New Orleans, LA: Naval Biodynamics Laboratory.

Perez, W. A., Masline, P. J., Ramsey, E. G., & Urban, K. E. (1987). *Unified Tri-Services Cognitive Performance Assessment Battery: Review and methodology* (Rep. No. AAMRL-TR- 87-007). Dayton, OH: Armstrong Aerospace Medical Research Laboratory.

Savu, T. G. (1991). *A synthetic work environment experiment.* Unpublished master's thesis, University of Oklahoma.

Snapper, A. G., & Kadden, R. M. (1973). Time sharing on a small computer based on a behavioral notation system. In B. Weiss (Ed.), *Digital computers in the behavioral laboratory* (pp. 41-97). New York: Appleton-Century-Crofts.

Thorne, D. E., Genser, S. G., Sing, H. C., & Hegge, F. W. (1985). The Walter Reed Performance Assessment Battery. *Neurobehavioral Toxicology & Teratology*, **7**, 415-418.

NOTES

1. Mouse and file-handling routines were used from the ADVBAS shareware library (advanced function library for the BASIC compiler) by Thomas Hamlin, available on many computer bulletin boards or from the author.

2. Appreciation is expressed to Paul Naitoh and Tamsin Kelly, who are the principal investigators in the sleep-deprivation studies.

3. Degrees of freedom were adjusted by using the Greenhouse-Geisser correction.

(Manuscript received May 10, 1993;
revision accepted for publication March 8, 1994.)