Behavior Research Methods & Instrumentation
1979, Vol. 11 (2), 192-198

SESSION VI
SPECIAL PARADIGMS

JOHN HALE, *University of California, Presider*

Data acquisition software for high data rate experiments

PETER VAN GELDER
Psychiatry Department, New York University Medical Center, New York, New York 10016

JAMES TODD
University of Connecticut, Storrs, Connecticut 06268

and

WAI HON TSUI
New York University Medical Center, New York, New York 10016

A computerized perceptual laboratory operates a variety of stimulus devices, including static and dynamic graphics displays, and records both discrete and analog data in a diverse and complex set of experimental paradigms. The software requirement to accommodate this variety of paradigms and high input and output data rates has been met by a single multitasked acquisition program that interprets unitary event commands to display graphics or nongraphics stimuli, record responses, control timing, or provide appropriate feedback. Event lists are created and modified with a text editor, then assembled into binary experiment definition files by dialogue with a parsing program. Graphics stimuli are referenced by a file name that contains stimulus attributes for later data extraction. All protocols and stimulus files are loaded prior to the block of trials, so no disk accesses that would delay events are required during a block. The resulting data file contains a record of all variable stimulus and timing information, as well as the discrete and analog responses, in a uniform format which facilitates data extraction.

The Information Processing Laboratory of the Psychiatry Department, New York University Medical Center, was established to conduct a wide variety of perceptual experiments. Graphics stimuli are presented on a plasma display screen which provides a matrix of 512 by 512 neon dots in an 8-in-square viewing area; the screen is constructed in a glass sandwich that allows back-screen slide projection. A relay register operates the slide advance and tachistoscopic shutter of the slide projector. Auditory tone stimuli are presented by a voltage-controlled oscillator driven by a digital/analog (D/A) channel, and auditory speech stimuli are presented on a prerecorded audio tape. Discrete responses are made on the plasma display keyboard and by switch closures from telegraph keys and a voice key. A 24-channel amplification system allows a wide variety of analog inputs, although only two channels are currently utilized for oculomotor (EOG) responses.

The first project of the laboratory is a battery of 10 experiments designed to measure as many different components of attention as possible. Reaction time and oculomotor measures are emphasized to maximize response information. One task, for example, is the Stroop color-word test, to measure the competing influence of an irrelevant stimulus dimension. Our version of this popular test is a discrete trials reaction time procedure, where countdown and feedback information is presented as graphics stimuli on the display screen, but the colored stimulus words are projected onto the same screen from slides. The subject presses one of four buttons to indicate the color of the stimulus word, which may be a competing color name, the same color name, or a neutral nonword. Preliminary training conditions first provide time uncertainty, then event uncertainty using only the neutral nonword stimuli.

Supported in part by the Schizophrenia Research Program, The Supreme Council, 33° A:. A:. of the Scottish Rite, Northern Masonic Jurisdiction, USA. The many thoughtful suggestions of Gail Ellen Levy are appreciated. Send correspondence to Peter Van Gelder, Psychiatry Department, New York University Medical Center, 550 First Avenue, New York, New York 10016.

0005-7878/79/020192-07$00.95/0

Another task in the battery is the continuous performance test, where letters are presented on the screen for 100 msec each, with 1 sec between letters. As a vigilance task of maintained attention, subjects press a key whenever a target letter follows a priming letter. Letters are blocked in lists of 50, and the target letter changes every three lists to test for the ability to shift to a new target. Reaction time and error feedback are presented for each response. A speeded condition allows only 130 msec between letters to provide a more continuous visual task. Here the task is simply to count the number of occurrences of a target in a list of 50 letters, with a second guessing opportunity provided upon an incorrect response.

As a truly continuous attention task that has shown clinical sensitivity in differentiating persons at risk for schizophrenia, a pursuit eye tracking task provides a moving single letter on the display, swinging back and forth as a simulated pendulum, with a cycle time of 2.5 sec. The target letter is moved to the next displacement frame every 10 msec, while two channels of EOG data are sampled every 4 msec. In one condition the letter remains constant for 16 movement cycles. In another condition the letter is replaced by another letter every .5 sec, for a reading task concurrent with the tracking task. In a final condition, the continuous performance task is used with the moving target, but without response feedback.

A fourth task from the attention battery is a choice reaction time task using two graphics geometric figures and two auditory tone stimuli, randomly intermixed, in a cuing condition. A brief sample of one of the four stimuli is presented at the start of each trial to indicate which stimulus is most likely to be presented on that trial. A detailed exposition of this experiment will be given as a concluding example; its data analysis will be presented similarly in the following article.

The four experiments outlined above should serve to indicate the diversity of experimental paradigms that guided the development of the data acquisition software. This diversity, coupled with the high data rate requirements of static and dynamic graphics output, analog input, and tachistoscopic real-time constraints, dictated a single acquisition progam that could interpret a pregenerated event protocol. This program is now the nucleus of an integrated data acquisition and analysis software system. Supporting programs generate stimuli and event protocols, and the single output file format has enabled the development of powerful analysis programs for discrete and analog data.

DESIGN CRITERIA

High Data Rate

Graphics stimuli must be presented easily and with precise control of timing. In the limiting case of moving stimuli, selected stimuli are erased and rewritten with some specified displacement every 10 msec. Analog data sampling must proceed concurrently with other events, making a sweep of all relevant channels at an appropriately programmed rate. Discrete stimulus events and response recording must proceed as directed, with millisecond time resolution, to allow tachistoscopic paradigms and the recording of reaction times.

The resulting acquisition progam is a multitasking set of assembly language modules. All timing occurs on an interrupt-driven basis from a set of external timers. An experiment is organized into blocks of trials, so that all event protocols are loaded prior to the block. Stimulus files for the entire experiment are loaded prior to the first block. The result is that no disk accesses are required for stimulus events during a block of trials. Discrete and analog data are sent to the disk as they are acquired, on a low-priority buffered basis.

Flexible

There must be as few paradigm constraints as possible. The acquisition program has therefore been conceived as a paradigm-free collection of modules, interpreting an event protocol that has been generated in advance.

Minimize Required Programming Skills

Although it is obvious that psychologists should be able to understand the operation of their own experiments to the extent of programming the apparatus, the inclusion of this point as an explicit design criterion has proved useful in preserving several principles. First, a simple syntax is employed. Only a single experimental event is specified for each typed input line. The number of arguments to each event specification is thereby minimized. Second, a simple experiment requires a correspondingly simple program. All of the more complex features employed in a more detailed experiment are optional. Third, the system is easy to use. Experimental protocols are composed and modified by using the text editor, and an indirect file reference capability allows large sequences of experimental events to be included by named reference.

Powerful

The system must have the capability of presenting complex experimental paradigms and complex stimulus displays. Our experiments, for example, routinely present instructions, training examples, and reaction time feedback after each response to provide an active dialogue with the subject. The system is also powerful in preserving the complete protocol of variable stimuli and responses for later analysis. This occurs automatically, preserving a record of everything that changes from trial to trial without explicit command to do so.

Easy to Implement and Change

The concept of a collection of assembly language modules activated interpretively at run time by a binary event protocol prepared in advance by a high-level language program, introduced primarily to support paradigm flexibility and real-time constraints, has served also to facilitate program development and evolution.

Transportable

Although the run-time program must reside on a Data General Corporation computer, an effort has been made to facilitate the substitution of different peripheral device handlers. A particular problem is the graphics display device, because different experiments and different laboratories may legitimately require the mutually exclusive properties of different displays. We utilize the back-screen projection capability and spatial and temporal precision of display that our inherently digital plasma display provides. Another application may prefer a video display for color and gray scale capability and the rapid alternation of complex displays. A stroke-generated CRT display may better suit still another application for its greater capability in presenting complex dynamic displays.

In service of this need, we adopted a universal stimulus file format for graphics stimuli; it is free of device-specific instruction formats but flexibly allows the inclusion of any display information that a device may be able to use. Then a new device handler must be written for each new type of display in the system to translate display information into the requirements of the instruction set of that device. Any stimulus file will then be presentable on any device. Information such as color will simply be ignored by a device without that capability. At present, device handlers have been written for a stroke-generated CRT (Megatek) and a fast video alphnumeric display (Lexiscope), as well as the plasma display (Applications Group).

PROCEDURAL OVERVIEW

Figure 1 illustrates the procedural flow of both data acquisition and analysis. Data acquisition and analysis are related more formally than is customarily the case because the single acquisition program, with its automatic generation of an output file containing all variable stimulus and response information, provides a universal, paradigm-independent, data file format for analysis. This has facilitated the writing of a powerful data analysis capability mated to the data structure of the acquisition program. The analysis program is described by Tsui and Van Gelder (1979)

There are four main items in the experimental procedure: editing event sequences, parsing them into binary experimental protocols, running the experiment, and analyzing the data.

EDIT

The experimenter's input to the data acquisition system is a set of event lists generated via Data General's text editor. Modifications to an experiment are thus made by using the text editor to make appropriate changes in event lists.

PARSE

The various event lists for each block are collected by a parser that assembles a binary experiment definition file. The lists can be typed in directly or exist as separate named disk files, or be some combination of both. For example, major units of events may exist separately, to be combined interactively by the parser via indirect file references. The parser also generates a dialogue file, preserving a complete record of the experimental events that can be typed or printed.

RUN

The data acquisition program reads the experiment definition file as input and produces an output file. An experiment is run by typing

RUN inputfilename outputfilename.

The experiment definition file contains references to all stimulus and movement files, which are loaded by RUN before the start of the first block. A movement file is a list of successive horizontal and vertical displacements which defines one cycle of movement. If the displacements all sum to zero then the movement may proceed for any number of cycles. Simple FORTRAN programs may be written to generate these files. Text stimuli are generated by the text editor. The experimenter need not differentiate between text and graphics stimuli when including stimulus file references in event lists.

Graphics stimuli are generated by a program that produces and displays stimuli interactively. They also may contain indirect file references to enable the building of complex pictures out of subpictures. Alternatively, subpictures may be referenced directly in the event lists, but the capability of specifying a complex graphics stimulus with a single file reference allows that single

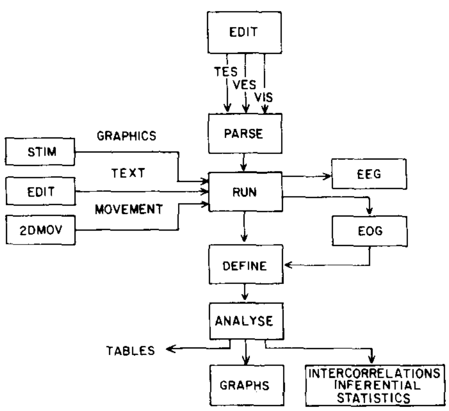

Figure 1. Logical flow of the data acquisition and analysis process.

stimulus file name to indicate the experimenter-defined attributes that may be useful for later data extraction. For example, one of our pilot experiments used trigrams as stimuli. The three constituent characters of each trigram were generated as graphics stimuli so they would be larger than letters generated by the hardware character generator of the plasma display. In one condition, subjects pressed one button if a vowel was present and another if not. In another condition, subjects pressed one button if the trigram was symmetrical about a vertical axis and another if not. Rather than using three stimulus file references to present the separate characters, the trigrams themselves were made into stimulus files, with a four-character file name convention. The first character indicated whether a vowel was present, the second indicated whether the trigram was symmetrical, the third indicated whether the stimulus was a word or a nonword, and the fourth character of the stimulus file name was a digit that indicated the instance of that class of trigrams.

The output file contains a reference to the input file name for recoverability of fixed events. The bulk of the file consists of all variable stimuli and variable timing events, discrete responses and correct response specifications, and analog data. Analog data must be analyzed by applications-oriented programs, but discrete data are passed to the generalized data analysis program.

ANALYSE

After passing through the DEFINE program, which strips the analog data and enters the resulting discrete data file into the data base explicitly, the ANALYSE program operates to extract descriptive statistics. Its output may be printed directly in tabular form or used as input for further processing into graphic form or for inferential statistics.

DATA ACQUISITION LANGUAGE CONVENTIONS

For each block of trials, the parser asks for the name of a file containing a Trial Event Sequence (TES) which defines the structure of the trial. The TES contains all events that remain constant across trials. It also directs where variable stimulus and timing events should be inserted in the sequence of events and may include event groups that are executed under certain contingencies.

The parser then requests the name of the disk file containing a Variable Event Sequence (VES). Each entry on this list of events includes either a graphics stimulus file name and screen location or a nongraphics stimulus name and specification, and an associated correct response specification. Each reference to a variable stimulus accesses the next list element. Therefore, the length of the list determines the length of the block of trials by automatically terminating the block when the list is exhausted. Randomization is accomplished by shuffling this list.

Stimulus Presentation

Stimuli presented in a TES are not associated with a correct response, since they remain constant throughout the block of trials. A display command presents a picture at some starting screen location with the format

DISPLAY, filename, x, y.

Nongraphics stimuli are presented as a 16-bit octal number setting a pattern in a relay register with the command

SET, pattern [, register]

or a voltage level on a D/A converter with the command

VOLTS, level [, channel].

Brackets enclose optional specifications. The default case operates all relay registers or all D/A channels. The display command is also used to present the next VES entry by the format

DISPLAY, VARIABLE.

That stimulus may be graphics, relay, or D/A output.

Movement

Some predefined movement can be imparted to the last stimulus file displayed by the command

MOVE, filename, cycles.

The file that defines one cycle of movement contains a list of X and Y offsets. Each X and Y offset pair thus defines one frame of the stimulus movement. Every 10 msec the stimulus is erased and then immediately presented in its next location, providing the next movement frame. Since only the most recently displayed stimulus event is moved, some information on the screen may be stationary while other information moves. Any stimulus presented after the onset of movement becomes the moving stimulus. In all other respects, movement proceeds concurrently with other ongoing events.

Timing

Fixed delays are introduced into the TES by the command

WAIT, msec.

Variable delays are introduced by the command

WAIT, VARIABLE.

For each block that calls for variable delays, the parser asks for a Variable Interval Sequence (VIS), a list in which each element is a time interval in milliseconds. The list determines the distribution of variable delays and may be shuffled in the same manner as the VES.

In subject-paced experiments, presentation of instructions, and the beginning of blocks of trials, it is desirable to introduce indeterminate delays. The PAUSE command delays the experiment until interrupted by any key on the graphics device keyboard or a special pushbutton input line. The latter is wired to a voice-actuated relay and is used to synchronize a prerecorded audio tape with the TES in a dichotic monitoring task.

Analog Data Input

Sampling some consecutive set of channels of the A/D converter for some number of sweeps of those channels at a specified rate is initiated with the command

SAMPLE, type, rate, sweeps, 1st channel, channels.

The type field allows different sample events in a TES to be uniquely identified for analysis. Sampling of analog data proceeds concurrently with other events, with the constraint that discrete data output must not occur while the sample event is still active. The constraint is necessitated by the output data file structure, which starts each array of sweeps with a header that indicates the length and structure of the array. Discrete data would intrude into this array, making the data file irrecoverable. A run-time error message to this effect serves to avoid unpleasant surprises at analysis time.

Analog data sampling for an indeterminate period, consistent with the above data structure constraint, is provided by inserting the word CONT in the type field of the sample command. Sampling then proceeds continuously until a

SAMPLE, STOP

command is received. More precisely, when the sample event specified by the other arguments of the sample command has been completed, a new, identical sample event is initiated indefinitely unless a SAMPLE, STOP command has been received in the meantime.

Discrete Data Input

A response interval, during which the display device keyboard or telegraph keys will record a response, is initiated by the command

BUTTON, START

and terminated by the command

BUTTON, STOP.

Reaction timing begins with the start command. Any combination of stimulus and timing events may occur during the response interval, but a DISPLAY, VARIABLE command must have been received prior to the interval to provide a correct response, and implicitly a response device, specification.

Discrete Data Output

Correct and actual response, if any, along with any variable stimulus and timing information occurring since the previous data command, are sent to the output file by the command

DATA, type.

The type field allows different data commands to be inserted in the TES and to be so identified. This is useful for data extraction and is most useful when completely generalized event-contingent branching is implemented. At that time the necessity for the linear flow of events down the TES to define each trial will cease. The concept of the trial is then reduced to a special-case vestige of its former self; accordingly, trials are not included in the structure of the data file and are not mentioned in the documentation of data analysis.

Contingency-Driven Event Groups

Some special-case branching is presently implemented. A group of events, labeled as to which of 11 contingencies causes the group to be executed, is bracketed between the command

GROUP, grouptype

and the command

GROUP, END.

The contingencies can be an experiment boundary (INITIATE, TERMINATE), a block boundary (FIRST, LAST), response error feedback (CORRECT, INCORRECT), reaction time feedback (RT), reaction time category feedback (FAST, OK, SLOW), and no response feedback (NORESP). The seven feedback contingencies in the last four classes of event groups are activated by the joint occurrence of the response event itself and a feedback command of the format

FEEDBACK, n

in the TES. The n field is a 4-bit octal number, each bit of the number enables one of the four classes of feedback event groups. The feedback command is placed between the start of the response interval and the subsequent DATA command. If a feedback command occurs before a response is made, its action is suspended until the response occurs. Different subsets of the four feedback classes may be initiated at different points in the TES by the use of multiple feedback statements. Reaction time feedback is always the reaction time displayed in milliseconds. Its event group serves to identify the stimulus files containing the individual digits, to allow for different character sets.

Variable Event Sequence

Any stimuli, graphics or nongraphics, may be listed in order of presentation in the VES. Each includes a correct response specification. If more than one DISPLAY, VARIABLE command is encountered prior to a response, then only the most recently encountered correct response specification is applicable for response feedback and for entry into the data file. Graphics stimuli are specified with the format

filename, rc, x, y.

Relay-operated stimuli are presented with the command

stimname, rc, SET, pattern [, register]

and analog voltage output stimuli are specified with the command

```
;CUED CONDITION

;CUEING INSTRUCTIONS FOR BEGINNING OF BLOCK
GROUP,FIRST
DISPLAY,QINSTR,75,100
PAUSE,
GROUP,END
;INDIRECT FILE FOR STANDARD CONTINGENCY GROUPS
@GROUPS@

;PARAMETERS AS MNEMONIC COMMAND ARGUMENTS
•HCENTER = 252
•VCENTER = 260
DISPLAY,ERASE,0,0
;DISPLAY LONG LINES & FIXATION POINT
DISPLAY,LINE,85,85
DISPLAY,LINE,256,85
DISPLAY,LINE,85,427
DISPLAY,LINE,256,427
DISPLAY,FIXPT,HCENTER,VCENTER
WAIT,700
DISPLAY,ERASE,0,0
;DISPLAY THE CUE
DISPLAY,VARIABLE
;DISPLAY SHORT LINES AND FIXATION POINT
DISPLAY,LINE,171,171
DISPLAY,LINE,171,342
DISPLAY,FIXPT,HCENTER,VCENTER
WAIT,700
VOLTS,0,0
DISPLAY,ERASE,0,0
;DISPLAY FIXATION POINT ONLY
DISPLAY,FIXPT,HCENTER,VCENTER
WAIT,700
DISPLAY,ERASE,0,0
;DISPLAY THE STIMULUS
DISPLAY,VARIABLE
BUTTON,START
WAIT,1000
;TURN OFF TONE, IF ANY
VOLTS,0,0
BUTTON,STOP
;PRESENT FEEDBACK IN PROPER ORDER:
;RT FIRST, THEN OTHERS, SO RT WILL BE ERASED IF WRONG
FEEDBACK,4
FEEDBACK,13
WAIT,1500
DISPLAY,ERASE,256,256
WAIT,1500
DATA,0
```

Figure 2. Trial event sequence example for visual and auditory cuing experiment.

```
AL,<3>,VOLTS,631
AL,<3>,VOLTS,631
;
AL,<3>,VOLTS,631
AL,<3>,VOLTS,631
;
AL,<3>,VOLTS,631
AL,<3>,VOLTS,631
;
AR,<5>,VOLTS,3777
AR,<5>,VOLTS,3777
;
AR,<5>,VOLTS,3777
AR,<5>,VOLTS,3777
;
AR,<5>,VOLTS,3777
AR,<5>,VOLTS,3777
;
VL,<3>,240,235
VL,<3>,240,235
;
VL,<3>,240,235
VL,<3>,240,235
;
VL,<3>,240,235
VL,<3>,240,235
;
VR,<5>,240,235
VR,<5>,240,235
;
VR,<5>,240,235
VR,<5>,240,235
;
VR,<5>,240,235
VR,<5>,240,235
;
AL,<3>,VOLTS,631
ERASE,<2>,256,256
;
AR,<5>,VOLTS,3777
ERASE,<2>,256,256
;
VL,<3>,240,235
ERASE,<2>,256,256
;
VR,<5>,240,235
ERASE,<2>,240,235
;
AL,<3>,VOLTS,631
AR,<5>,VOLTS,3777
;
AR,<5>,VOLTS,3777
AL,<3>,VOLTS,631
;
VL,<3>,240,235
AL,<3>,VOLTS,631
;
VR,<5>,240,235
AL,<3>,VOLTS,631
```

Figure 3. Twenty trials of cue-stimulus pairs for visual and auditory cuing experiment.

stimname, rc, VOLTS, level [, channel] .

In both cases, "stimname" is the reference name of the stimulus that is added to the output file for data extraction purposes, but is not an actual disk file name as for graphics stimuli.

All randomization is performed in this list (and in the VIS) by the command

SHUFFLE, first, last, count,

where "first" and "last" are unit numbers in the list and "count" is the number of list elements considered as a unit for randomization purposes. For example, if each trial contains three ordered DISPLAY, VARIABLE commands, taking an ordered triple of stimuli from the VES, but the order of trials is to be randomized, then intact triples of the sequence are shuffled by the command

SHUFFLE, 1, 30, 3

for 30 trials or 90 list elements. Shuffle commands may be nested to shuffle within subgroups, then shuffle the subgroups.

AN EXAMPLE

Figure 2 presents the TES for the cuing experiment where two visual and two auditory stimuli are presented. Each trial begins with a three-phase countdown consisting of two long lines, then two short lines, then no lines bracketing a central fixation point. In the second phase of the countdown the cue is presented. Then a 1-sec response interval is followed by appropriate feedback of all four classes; feedback remains on the screen for 1.5 sec.

In Figure 3 the auditory (AL and AR) and visual (VL and VR) stimuli are listed in cue-stimulus pairs for 20 trials of a representative 40-trial block. Catch trials are indicated by ERASE stimuli. Three sets of the 20 stimuli are required to produce all noncued combinations in the appropriate frequencies, so they are combined two sets per block, for a total of three blocks with 120 trials. Figure 4 presents the VES, which indirectly references two sets of 20 trials each and randomizes the first half and last half of the block separately.

```
@20TRITG1@
SHUFFLE,1,20,2
@20TRITG2@
SHUFFLE,21,40,2
```

Figure 4. Variable event sequence example for visual and auditory cuing experiment. Incorporates two 20-trials lists, as in Figure 3, as indirect files.

REFERENCE

Tsui, W. H., & Van Gelder, P. A general-purpose data extraction language. *Behavior Research Methods & Instrumentation*, 1979, **11**, 199-204.