PROGRAM ABSTRACTS/ALGORITHMS

A subroutine for counterbalanced assignment of stimuli to conditions

FRANCIS T. DURSO
University of Oklahoma, Norman, Oklahoma

In numbers of cognitive psychology experiments dealing with verbal materials, the experimenter attempts to control for the effect of the particular linguistic stimulus (see Clark, 1973) by making certain that the stimulus appears in each of the experimental conditions. One solution, of course, is to place every stimulus in every experimental condition for every subject. This, however, introduces several sequencing confounds (Campbell & Stanley, 1963) that result from a subject's responding to the same stimulus twice. Most experimenters have chosen instead to counterbalance the assignment of stimuli to conditions across subjects, so that by the end of the experiment a particular stimulus has appeared in each experimental condition equally frequently, but any particular subject has responded to the stimulus only once (e.g., Schvaneveldt, Durso, & Mukherji, 1982).

For much experimentation, this has been accomplished by creating a different stimulus file for each subject (or set of subjects receiving the same counterbalancing). Alternatively, experimenters may sometimes write a counterbalancing routine idiosyncratic to the particular experiment. These approaches are often tedious, and become increasingly intractable as the number of factors to be simultaneously considered increases.

This note describes a subroutine capable of performing these counterbalancing operations for any multilevel, factorial design. The subroutine's output is a vector of codes representing levels of the independent variables that can then be used by the experimenter to control on-line presentation; as examples, these codes can be used to subscript variables (e.g., to choose the type of priming stimulus) or to control logical branching (e.g., choose a display duration).

Input Requirements. The version of the routine in Table 1 is in MBASIC, although it can be translated easily into any other language supporting integer division (backslash, \, in MBASIC) and the MOD function. For example, it has been successfully incorporated into the Pascal-based PSYCHLAB system (Maxwell & Schvaneveldt, 1983). The essential code of the subroutine is contained in Line 310.

This work was conducted while the author was a visiting scientist at the Computing Research Laboratory, New Mexico State University. The author's address is: Department of Psychology, University of Oklahoma, Norman, OK 73019.

Table 1
MBASIC Code for the Counterbalance Subroutine and the Globals Required From the Main Program With Illustrative Values

Main Program	Counterbalance Routine
The following must be entered or initialized in the Main Program:	
SN = 543 NFACTORS = 3 LEVELS = 2 3 2 NTRIALS = 24 ITEMNUM = usually randomly selected	
In addition the variable CONS must be created in the main program:	
CONS=1: FOR F=1 TO NFACTORS: CONS=CONS*LEVELS(F): NEXT F	
prior to the subroutine call:	
GOSUB 250	
	250 'CTRBAL
	260 IPL=NTRIALS
	270 REPS=NTRIALS\CONS:IF (REPS<CONS) THEN
	PRINT "UNBALANCED":STOP
	290 FOR IV=1 TO NFACTORS
	300 IPL=IPL\LEVELS(IV)
	310 CONDS(IV)=((((ITEMNUM-1)\IPL) MOD LEVELS(IV)) +
	((SN-1)\(IPL\REPS)) MOD LEVELS(IV))) MOD LEVELS(IV)) + 1
	330 NEXT IV
	350 RETURN
	IPL= items per level
	REPS= the number of stimuli within a stimulus set

Table 1 also provides examples of the variables required in the main program calling the counterbalance subroutine. Much of the input required by the routine will be available in the main body of typical experimental control programs. For example, the routine must have access to the variables for subject number (SN), number of trials (NTRIALS), and number of independent variables (NFACTORS). In addition, the main program must explicitly represent the design of the experiment as a vector of integers (LEVELS), where each integer is the number of levels for the independent variables. The total number of conditions in the design (CONS) must also be created in the main program. This is used to determine the number of repetitions (REPS) or the size of the set of stimuli to be treated identically by the counterbalance subroutine.

Output. For each stimulus item, the routine returns a vector (CONDS) specifying the appropriate level of each of the independent variables for that stimulus for that subject. If the number of trials in the experiment is a multiple of the number of conditions, then the routine assumes that it is counterbalancing sets of several stimuli rather than sets consisting of only a single stimulus. If the number of trials in the experiment is not a multiple of the number of conditions, the routine returns an "Unbalanced" error.

Table 2 shows the output for a hypothetical experiment with three independent variables of levels 2, 3, and 2. The total number of trials was 24, indicating that each stimulus set consisted of two items. To counterbalance

Table 2
Sample Output From the Counterbalance Routine for Selected
Subjects (Example Parameters Are From Table 1)

Subject #	1	2	3	4	6	12
Stimulus Number						
1	1 1 1	1 1 2	1 2 1	1 2 2	1 3 2	2 3 2
2	1 1 1	1 1 2	1 2 1	1 2 2	1 3 2	2 3 2
3	1 1 2	1 1 1	1 2 2	1 2 1	1 3 1	2 3 1
4	1 1 2	1 1 1	1 2 2	1 2 1	1 3 1	2 3 1
5	1 2 1	1 2 2	1 3 1	1 3 2	1 1 2	2 1 2
6	1 2 1	1 2 2	1 3 1	1 3 2	1 1 2	2 1 2
7	1 2 2	1 2 1	1 3 2	1 3 1	1 1 1	2 1 1
8	1 2 2	1 2 1	1 3 2	1 3 1	1 1 1	2 1 1
9	1 3 1	1 3 2	1 1 1	1 1 2	1 2 2	2 2 2
10	1 3 1	1 3 2	1 1 1	1 1 2	1 2 2	2 2 2
11	1 3 2	1 3 1	1 1 2	1 1 1	1 2 1	2 2 1
12	1 3 2	1 3 1	1 1 2	1 1 1	1 2 1	2 2 1
13	2 1 1	2 1 2	2 2 1	2 2 2	2 3 2	1 3 2
14	2 1 1	2 1 2	2 2 1	2 2 2	2 3 2	1 3 2
15	2 1 2	2 1 1	2 2 2	2 2 1	2 3 1	1 3 1
16	2 1 2	2 1 1	2 2 2	2 2 1	2 3 1	1 3 1
17	2 2 1	2 2 2	2 3 1	2 3 2	2 1 2	1 1 2
18	2 2 1	2 2 2	2 3 1	2 3 2	2 1 2	1 1 2
19	2 2 2	2 2 1	2 3 2	2 3 1	2 1 1	1 1 1
20	2 2 2	2 2 1	2 3 2	2 3 1	2 1 1	1 1 1
21	2 3 1	2 3 2	2 1 1	2 1 2	2 2 2	1 2 2
22	2 3 1	2 3 2	2 1 1	2 1 2	2 2 2	1 2 2
23	2 3 2	2 3 1	2 1 2	2 1 1	2 2 1	1 2 1
24	2 3 2	2 3 1	2 1 2	2 1 1	2 2 1	1 2 1

assignment of materials to the 12 conditions requires 12 subjects, and thus Subject 13 would receive the same assignments as Subject 1. The routine binds the functional

subject number to the appropriate range regardless of the nominal subject number input by the experimenter.

With this routine incorporated into an experimental control program, an experimenter can create one stimulus file, present these stimuli in a random order, recover the data in an order consistent with the stimulus file (along with accompanying codes), and yet be certain that the assignment of stimuli to experimental conditions is counterbalanced to assure that the peculiarities of the stimuli do not contribute (or detract) from the experimental effects in an undesirable way.

REFERENCES

- CAMPBELL, D. T., & STANLEY, J. C. (1963). *Experimental and quasi-experimental design for research*. Chicago: Rand McNally.
- CLARK, H. H. (1973). The language-as-fixed-effect fallacy: A critique of language statistics in psychological research. *Journal of Verbal Learning and Verbal Behavior*, 12, 335-359.
- MAXWELL, K. J., & SCHVANEVELDT, R. W. (1983). The PSYCHLAB programming system: A tool for developing experimental control programs. *Behavior Research Methods & Instrumentation*, 15, 49-56.
- SCHVANEVELDT, R. W., DURSO, F. T., & MUKHERJI, B. R. (1982). Semantic distance effects in categorization tasks. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 8, 1-15.

(Manuscript accepted for publication September 21, 1984.)