

A program that simulates random choice in radial arm mazes and similar choice situations

MARCIA L. SPETCH and DONALD M. WILKIE

University of British Columbia,
Vancouver, British Columbia V6T 1W5, Canada

The radial arm maze is one of several choice procedures used in psychology in which a subject is presented a set of items from which to choose with replacement. The maze has a central area from which a number of arms radiate, like the spokes on a wheel. This apparatus is being used with increasing frequency in studies of animals' working memory (Honig, 1978) for spatial cues (cf. Olton, 1978, 1979). In a typical experiment, a food pellet is placed out of sight at the distal end of each arm, and then a food-deprived rat is placed in the central area and allowed to choose among the arms. The rat sometimes behaves optimally: It chooses each arm only once and thereby obtains all the pellets. At other times, such as early in training or in mazes with a large number of arms, animals make errors by reentering a previously chosen (and now empty) arm. Then it is important to know how the animal is performing relative to simple random selection of arms. For example, when the number of choices permitted on the trial is limited to the number of arms in the maze, how many different arms might the animal enter as the result of random selection? Or if the animal is permitted to choose among the arms until all pellets have been obtained, how many entries will be required before each arm is entered at least once, if the animal responds randomly?

Calculation of either of these values is tedious, especially when the number of arms is large. It is possible to readily determine these and other measures with a simple computer program that simulates an animal making random arm choices. The program is written in the ACT language (Millenson, 1973), but any of the widely used state notation process control languages that have an adjustable probability generator (e.g., MANX, Gilbert & Rice, 1979; SKED, Snapper & Inglis, 1979) and run on Data General's NOVA and Digital Equipment Corporation's PDP-8 and PDP-11 would serve equally well.

Programming is straightforward. For example, in a program written to estimate chance accuracy when n selections are made in a maze with n arms, the beginning state (S_B) represents the central area of the maze. After a period of time, control passes from S_B to one of n other states ($S_1, S_2, \dots, S_i, \dots, S_n$) that represent the different arms. Control passes to a particular one of these states with a probability set equal to $1/n$. Each state has a variable associated with it, the value of which is incremented when that state is entered. After a S_i state has been entered, control passes back to S_B .

After n exits from S_B , control passes to a state that sets a variable (number of correct arm entries) equal to the number of variables, associated with the S_i states, that had been incremented (i.e., that had been entered at least once). The variables associated with the S_i states are then zeroed, and control passes back to S_B for another trial of n exits. After numerous trials, division of the number of correct arms variable by the number of trials variable yields the expected number of correct choices per trial, assuming random arm choices.

Because most state notation process control languages allow transitions from one state to another each .01 sec, several trials per second can be simulated. Thus a run of several minutes produces several thousand trials and stable statistical values. Using a single program, we have simulated random performance in mazes having 2 to 20 arms. Table 1 shows the expected number of correct choices per trial. Each value is based on at least 15,000 simulated trials.

The program can be modified easily to take arm biases into account or to simulate the number of choices required by chance to enter all arms of a maze at least once. With other modifications, the program can be elaborated to assess other aspects of chance performance, such as sequential patterns of errors.

Availability. Listings of sample programs can be obtained free of charge from either author: Department of Psychology, 2075 Wesbrook Place, University of British Columbia, Vancouver, British Columbia V6T 1W5, Canada.

Table 1
Estimated Chance Performance for Radial Arm Mazes
With 2 to 20 Arms

Number of Arms	Estimated Number Correct by Chance
2	1.50
3	2.11
4	2.73
5	3.36
6	3.99
7	4.61
8	5.25*
9	5.87
10	6.52
11	7.14
12	7.76
13	8.41
14	9.04
15	9.66
16	10.28
17	10.93†
18	11.55
19	12.19
20	12.81

*Olton (1978) reports a value of 5.3; Eckerman (1980) reports a value of 5.28. †Olton (1978) reports a value of 10.9.

REFERENCES

- ECKERMAN, D. A. Monte Carlo estimation of chance performance for the radial arm maze. *Bulletin of the Psychonomic Society*, 1980, 15, 93-95.
- GILBERT, S. G., & RICE, D. C. NOVA SKED II: A behavioral notation language utilizing the Data General Corporation real-time disk-operating system. *Behavior Research Methods & Instrumentation*, 1979, 11, 71-73.
- HONIG, W. K. Working memory in pigeons. In S. H. Hulse, W. K. Honig, & H. Fowler (Eds.), *Cognitive aspects of animal behavior*. Hillsdale, N.J: Erlbaum, 1978.
- MILLENSON, J. R. On-line sequential control of experiments by an automated contingency translator. In B. Weiss (Ed.), *Digital computers in the behavioral laboratory*. New York: Appleton-Century-Crofts, 1973.
- OLTON, D. S. Characteristics of spatial memory. In S. H. Hulse, W. K. Honig, & H. Fowler (Eds.), *Cognitive aspects of animal behavior*. Hillsdale, N.J: Erlbaum, 1978.
- OLTON, D. S. Mazes, maps, and memory. *American Psychologist*, 1979, 34, 583-596.
- SNAPPER, A. G., & INGLIS, G. B. *Supersked: The Sked software system*. Kalamazoo, Mich: State Systems, 1979.

(Accepted for publication April 14, 1980.)