Random without replacement is not random: Caveat emptor

CHRIS BLAIS

University of California, Berkeley, California

The vast majority of psychology labs rely on prepackaged software applications (e.g., E-Prime) for the programming of experiments. These programs are often used for stimulus selection, and many use a selection method referred to as *random without replacement*. We demonstrate how random without replacement deviates from random selection, and we detail selection biases that result. We also demonstrate, in a simple experiment, how these selection biases, if left unchecked, can influence behavior. Recommendations for reducing the impact of these biases on performance when random without replacement is used are discussed.

Prepackaged software applications such as PsyScope (Cohen, MacWhinney, Flatt, & Provost, 1993), Presentation (www.neurobs.com), and E-Prime (Psychology Software Tools, 2002) have immensely simplified the programming of experiments in psychology. Stimulus selection is one of the tasks often delegated to these software packages. Most modern designs present participants with multiple repetitions of exemplars from a given condition. For example, in the Stroop task, participants would see each congruent item (e.g., BLUE in blue, RED in red, GREEN in green) a number of times in a given experimental session. This common practice requires a mechanism with which a particular stimulus can be selected for each trial. Prepackaged software applications provide a number of different selection methods for the user (e.g., sequential selection, random with replacement, random without replacement). Here, we demonstrate two biases associated with the (mis)use of the most popular of these methods: random without replacement.

Random without replacement randomly selects one stimulus from a list and then tags it so that it cannot be chosen again (i.e., it is not replaced in the list) until some specific condition is satisfied (e.g., all of the other stimuli in the list have been selected). This form of stimulus selection is preferred by psychologists because it ensures the fixed exposure to each stimulus for a participant. The problem this introduces is that the selection of trial n is not independent of previous selections. In other words, random without replacement is not random selection. Here, we discuss two related biases associated with the random without replacement selection method and report an experiment demonstrating the influence that these biases have on behavior.

The Biases

Progressive determination. We refer to the first bias as *progressive determination*. When one uses random

without replacement, each selection in which a particular stimulus is *not selected* increases the likelihood that that particular stimulus will be selected next. For example, imagine that one has a bag filled with four different colored marbles (e.g., blue, yellow, red, and green). The task is to reach inside the bag and pull out one marble at a time. When all four marbles are in the bag, the probability of selecting any marble of a particular color (e.g., blue) is equal to the number of marbles of that particular color (i.e., one) divided by the total number of marbles in the bag (i.e., four; 25%). Using a random without replacement strategy, one selects a marble—for example, the blue marble—and then removes that marble from the bag before the next selection. After the first selection, the probability that an unselected marble (i.e., the yellow, red, or green marble) is selected increases, in this case to 33%. After the second selection, that probability increases to 50%, and after the third selection, the probability of selecting the unselected marble increases to 100%. In Figure 1, we plot the function relating the probability that an unselected stimulus is selected as a function of how many stimuli have yet to be selected from the list (i.e., the number of unique marbles in the bag).

It is instructive to compare the random without replacement selection strategy with another method: random with replacement. *Random with replacement* randomly selects one stimulus from a list and then returns that item to the list before the next selection. Thus, selection n is independent of previous selections. That is, random with replacement is true random selection. In our marbles example, a marble that has been selected would be placed back in the bag before the next selection. Random with replacement removes the progressive determination bias. Each selection in which a particular stimulus is not selected does not increase the likelihood that a particular stimulus will be selected next. However, random with replacement

Figure 1. Probability of selecting a stimulus as a function of list length. The solid line portrays the probability when random without replacement is used. The dotted line portrays the probability when random with replacement is used.

is not considered an ideal selection strategy for research psychologists, because there is no (easy) way to ensure that stimuli are sampled an equal number of times. If one is selecting marbles from the bag using a random with replacement strategy, there is no guarantee that one will select one marble of each color. Presenting stimuli a fixed number of times is typically a mainstay of experimental psychology.

Transitional bias. The second bias introduced by the random without replacement selection strategy is a transitional bias. Random without replacement alters the likelihood that two stimuli will follow each other. Returning to the marbles example, imagine that one wants to select a total of 12 marbles using the same bag of 4 colored marbles. According to random without replacement, 4 marbles would be selected, then all marbles returned to the bag (i.e., the stimulus list is repopulated); 4 marbles are selected again, then all marbles are returned to the bag; then, finally, 4 marbles are selected again. In such a design, an immediate repetition can occur only on the first selection after all the marbles have been returned to the bag. Thus, the probability of an immediate repetition in this design is $1/4$ (an immediate repetition can occur on the first selection only after all the marbles are returned to the bag) $\times 1/4$ (the probability of selecting the same marble as the previous selection) = 6.25%. With

four stimuli, a repetition should occur randomly on 25% of the trials. Thus, random without replacement leads to a decrease in the number of repetitions and an increase in the number of nonrepetitions relative to what would be expected by chance. The likelihood of a repetition depends on the number of stimuli and the number of instances of each stimulus in the list and is given by the formula

$$P(\text{repetition}) = \frac{xn - n + 1}{xn^2},$$

where n is the number of unique stimuli that are being sampled from (e.g., the number of different colored marbles) and x is the number of instances of each of those stimuli (see the Appendix). Figure 2 shows the probability of a repetition when random without replacement is used, given various list lengths.

Are These Biases a Problem?

The existence of the progressive determination and transitional biases may not be sufficient to motivate a researcher to attempt to eliminate them without strong evidence that these biases could threaten the validity of their results. It would be a Herculean task to empirically demonstrate when, where, and how these biases influence results in the myriad paradigms available. Instead,

Figure 2. The probability of a repetition as a function of the number of unique items in the list (n) and the number of instances of each item in the list (x). ∞ represents true random selection.

we describe previous results suggesting that these biases could influence behavior and present a single experiment unequivocally demonstrating that these biases indeed do so.

A number of studies suggest that the progressive determination and transitional biases induced by random without replacement could influence the results of a given study. Generally, both biases provide information about upcoming stimuli, and prior knowledge can have appreciable effects on performance (e.g., Haber, 1966). Probably the most direct body of evidence for the influence of these biases on performance can be found in the sequence learning literature. In these experiments, participants are asked to respond to a stimulus, and the order of the stimuli either follows a fixed sequence or is random. Participants respond faster and more accurately to the stimuli that are embedded within a fixed sequence than to the stimuli embedded within random sequences despite the fact that the participant is unaware of the sequence information (e.g., Nissen & Bullemer, 1987). Both the progressive determination bias and the transitional bias institute a structure in the presentation of the stimuli that participants are likely to explicitly or implicitly pick up on and use.

This literature clearly suggests that the progressive determination and transitional biases could influence behavior. We also wanted to demonstrate their influence in a simple choice response time (RT) experiment, as well as to assess a solution that would minimize the effects of the biases generated by sampling randomly without replacement. The proposed solution involves using random without replacement (thus ensuring equal presenta-

tion of each stimulus) in conjunction with a stimulus list (i.e., the list from which the stimulus on each trial will be drawn) that consists of the maximum number of stimuli within the constraints of the experiment. For example, if there are four unique stimuli to be presented within a 64-trial experiment, then the list should consist of 64 entries (i.e., 16 repetitions of each of the four unique stimuli). A list that contains the maximum number of stimuli within the experiment's constraints will best approximate true random selection when random without replacement is used.

METHOD

Participants

Sixteen students from the University of California, Berkeley, served as participants and received compensation of either \$3 or partial course credit.

Stimuli and Design

Participants were asked to identify one of four letters (A, B, C, or D) displayed on the screen by means of a keypress. There were 64 trials in each block and the stimulus selection method varied across blocks. There were four different conditions representing the manipulation of list length. See Figure 3 for a schematic of the list length manipulation. A fifth condition in which the stimuli were randomly selected with replacement was used to get a sense of baseline.

List length = 4. The four stimuli (A, B, C, or D) were placed in a list with each unique stimulus represented once (i.e., list length = 4). After four selections (i.e., list length), the list was repopulated until 64 trials had been selected.

List length = 8. The four stimuli (A, B, C, or D) were placed in a list with each unique stimulus represented twice (i.e., list length = 8). After eight selections (i.e., list length), the list was repopulated until 64 trials had been selected.

Figure 3. Schematic illustration of the implementation of the list length manipulation. Below each heading are enough stimuli for 64 trials. With a list length of four, 4 selections are made; then the list is repopulated until 64 trials have been selected. With a list length of eight, 8 selections are made; then the list is repopulated until 64 trials have been selected. With a list length of 16 selections are made; then the list is repopulated until 64 trials have been selected. With a list length of 64 selections are made. The list does not have to be repopulated because the list length is equal to the number of trials.

List length = 16. The four stimuli (A, B, C, or D) were placed in a list with each unique stimulus represented four times (i.e., list length = 16). After 16 selections (i.e., list length), the list was repopulated until 64 trials had been selected.

List length = 64. The four stimuli (A, B, C, or D) were placed in a list with each unique stimulus represented 16 times (i.e., list length = 64). Because the list length was equal to the number of trials in the block, the list did not need to be repopulated.

Note that list length does not represent the number of unique stimuli in the experiment. List length represents the length of the list from which the stimuli are chosen. It is determined by the number of unique stimuli (n) and the number of repetitions of each of these unique stimuli (x) in the list. List length could, therefore, be increased by increasing either n (e.g., ABCD vs. ABCDEFGH) or x (e.g., ABCD vs. ABCDABCD). List length was manipulated by increasing x , because adding unique stimuli introduces various problems (e.g., different levels of difficulty across blocks). We opted for a task with four unique stimuli and assessed performance across various implementations (i.e., list lengths) of that design. We believe that this is sufficient to make our point.

In addition to the manipulation of list length using random with replacement, we also included a block of trials using random without replacement in order to determine performance when true random selection is used.

Random selection. The four stimuli (A, B, C, and D) were placed in a list with each unique stimulus represented once. The stimulus on each trial was selected using the random with replacement selection strategy.

Procedure

A four-choice task was utilized in which the participants were asked to respond as quickly and accurately as possible to the presentation of one of four letters (A, B, C, or D) using spatially con-

gruent mappings (i.e., the “d,” “f,” “j,” and “k” keys on a standard keyboard). The interval between the participants’ responding to a stimulus and the appearance of the next stimulus was 500 msec. The participants received two blocks of 64 trials selected from the four stimuli randomly with replacement both to learn the stimulus–response mappings and to have their RTs approach asymptote. Next, the participants received eight blocks of 64 trials, ordered using a partial Latin square design (see Table 1). The 64 trials within each block were sampled from lists containing 4, 8, 16, or 64 stimuli. Each item from the list was selected once before it was available for reselection. The final two blocks of 64 trials, sampled randomly with replacement, were presented to each participant. Thus, for Blocks 1 and 2 and for Blocks 11 and 12, stimuli were sampled randomly with replacement, and for Blocks 3–10, stimuli were sampled randomly without replacement according to one of the four list-size conditions (4, 8, 16, or 64). Self-paced breaks were available after every two blocks. The experiment lasted approximately 15 min.

Table 1
Order List Lengths Received by
Participants in Each Counterbalance

Counterbalance	Order							
	1	2	3	4	5	6	7	8
1	4	8	16	64	64	16	8	4
2	8	16	64	4	16	8	4	64
3	16	64	4	8	8	4	64	16
4	64	4	8	16	4	64	16	8
5	64	16	8	4	4	8	16	64
6	16	8	4	64	8	16	64	4
7	8	4	64	16	16	64	4	8
8	4	64	16	8	64	4	8	16

RESULTS

Correct RTs greater than 1,500 msec and less than 200 msec were excluded from the analysis (0.4% of the data).

Analysis 1: Progressive Determination

In order to test for an effect of progressive determination, we analyzed RT as a function of selection number and list length. If progressive determination does not have an effect on performance, there is no a priori reason why RT should be affected by its time of selection. However, if progressive determination does influence performance, we should observe a significant effect of selection number on RT.

Data from the four list lengths (4, 8, 16, and 64) were analyzed separately using one-way repeated measures ANOVAs with selection number (1–4 for a list length of 4, 1–8 for a list length of 8, 1–16 for a list length of 16, and 1–64 for a list length of 64) as the only factor. For list lengths of 8, 16, and 64, there was no effect ($F_s < 1.4$, $\eta_p^2 < .083$). However, when the list length was 4, there was a very strong linear effect [$F(1,15) = 12.71$, $MS_e = 550$, $p = .003$, $\eta_p^2 = .459$] such that RT was slowest on the first selection ($M = 559$ msec, $SE = 71$ msec) and got progressively faster from selection two ($M = 556$ msec, $SE = 87$ msec), to three ($M = 546$ msec, $SE = 76$ msec), to four ($M = 532$ msec, $SE = 83$ msec). The RT data when list length was 4 are shown in Figure 4.

Thus, with a list length of 4, progressive determination has a clear effect on performance. Progressive determination did not have any appreciable effect with the longer list lengths. The absence of this effect with list lengths of 8, 16, and 64 likely reflects the difficulty in maintaining a running count of which stimuli have and have not been presented. It is important to emphasize that the loss of the progressive determination effect once list length exceeded 4 may not represent an effect of list length per se but might reflect the fact that any length beyond 4 (here) necessitates the inclusion of a repetition (or repetitions) of the same stimulus. Thus, with a list length of 8, with eight unique stimuli, one may observe a progressive determination effect. Indeed, Boyer, Destrebecqz, and Cleeremans (2005) provided evidence consistent with this claim in a six-choice task where they used a list length of 6 with an additional constraint that the last stimulus in a given list could not be the same as the first stimulus in the list that followed it (i.e., there were no repetitions). This created a situation where, on each trial, the probability of a stimulus being chosen increased as the number of intervening items (i.e., its lag) since its last presentation increased. Consistent with our findings, Boyer et al. found that the longer lags were associated with shorter RTs; they referred to this association as a *negative recency bias*. Critically, when selection was random, this negative recency bias was eliminated. What is clear from the present experiment and previous work (e.g., Boyer et al., 2005) is that progressive determination can influence behavior, and this influence is reduced as the method of sampling approaches true randomness.

Figure 4. Response time as a function of selection number within the list. Error bars represent $\sqrt{(MS_e/N)}$.

Analysis 2: Transitional Bias

In order to test for an effect of the transitional bias, we assessed the effect of list length on the magnitude of the repetition effect in choice RT. Participants respond faster and more accurately when trial $n-1$ is the same as trial n (see, e.g., Bertelson, 1965). This repetition advantage is one of the most reliable effects in psychology. Random without replacement reduces the probability of a repetition, and the magnitude of this reduction depends on list length (see Figure 2).

The data were analyzed using a 5 (list length: 4, 8, 16, 64, or random) $\times$ 2 (repetition: repeated or not repeated) repeated measures ANOVA. There was no main effect of list length ($F < 1.02$, $\eta_p^2 = .063$). There was a large repetition effect [$F(4,60) = 75.65$, $MS_e = 255$, $p < .001$, $\eta_p^2 = .835$] that was qualified by the linear component of the list length $\times$ repetition interaction [$F(1,15) = 14.66$, $MS_e = 517$, $p = .002$, $\eta_p^2 = .494$]. Figure 5A shows RT as a function of list length and repetition.

Figure 5B highlights the striking relation between the size of the repetition effect and the probability of a repetition. The dotted line in Figure 5B is an RT-scaled curve based on the probability of a repetition at each list length, similar to Figure 2. Note that the observed data lie virtually atop this line; the probability of a repetition accounts for over 94.6% of the variance in the size of the repetition

Figure 5. (A) Response time (RT) for repeated items (open circles), non-repeated items (closed circles), and their average (gray dotted line). Error bars represent the standard errors of the means. (B) Size of the repetition effect (nonrepeated minus repeated RTs; closed circles) plotted as a function of the probability of a repetition (Figure 2, $n = 4$) scaled for these RT data [$RT = 2.136P(\text{rep}) + 5.037$; gray dotted line]. Light gray circles represent individual data points for each participant for each condition. Error bars represent the standard error of the difference. List length = xn ; ∞ represents the random selection condition. When xn is greater than the list length, it approximates random sampling, because the constraint of a fixed number of observations per cell is lost.

effect. This finding is important, because it shows that the magnitude of the repetition effect can be changed simply by manipulating the probability of a repetition.

Also of interest is the magnitude of the repetition effect in the random with replacement blocks. These blocks use what can be considered true random selection, and therefore, the repetition effects represent what can be expected when random selection is used. It is clear from Figure 5B that the transitional bias leads to an underestimation of the magnitude of the repetition effect. This underestimation is particularly evident with

a list length of 4, where the repetition effect is approximately 20 msec, compared with approximately 60 msec when random with replacement is used. Also important is that a list length of 64 using random without replacement approximates the “true” repetition effect. This is expected, given that the probability of a repetition with a list length of 64 (i.e., the four stimuli repeated 16 times each) is approximately 24%, whereas with true random selection, one would expect a repetition on 25% of trials. This result demonstrates that the transitional bias can have a marked effect on behavior.

DISCUSSION

The results of the experiment reported here demonstrate that both the progressive determination bias and the transitional bias, resulting from using random without replacement, have an appreciable effect on behavior. We next discuss a potential solution to these biases and possible implications for condition selection as opposed to stimulus selection.

Dealing With These Biases

The results of the present experiment suggest that a good strategy for reducing the impact of the progressive determination and transitional biases on performance is to increase the list length. In analyses of the impact of the progressive determination and transitional biases, their influence was most prevalent at the shorter list lengths. In the introduction, we suggested using a list length that was populated by as many instances of each stimulus as were required for the total number of stimuli to match the number of trials in the experiment. This experiment confirms that this is a safe route. An alternative would be to systemically determine the list length at which each bias no longer influences behavior for each experiment. For example, in this experiment the progressive determination analysis clearly indicates that the bias influenced behavior only at a list length of four. Thus, a list length greater than four appears sufficient to eliminate the progressive determination bias on choice RT in this experiment. However, the appropriate list length will change as a function of context, and the most general advice that can be given is to use the maximum list length (see also Brysbaert, 1991).

It is important to note that the strategy suggested here does not eliminate the biases. The progressive determination and transitional biases are direct by-products of random without replacement and will exist whenever that selection strategy is used. The suggested strategy merely reduces the impact of these biases in practice, by making random without replacement behave as closely as possible, given experimental constraints (e.g., equal selection of different stimuli, a set number of trials), to true random selection.

There are algorithms that generate sequences of trials such that each stimulus occurs equally often and each type of stimulus follows every other type equally often (Emerson & Tobias, 1995; Remillard & Clark, 1999). However, none of these are directly implemented into the major software packages and would involve the use of an external program to generate a number of lists (see Emerson & Tobias, 1995). The researcher would then need to present those lists sequentially to the participants, which defeats one of the main purposes of the software packages.

Stimulus Selection Versus Condition Selection

We have thus far focused on stimulus selection, ignoring the issue of condition selection. Nevertheless, conditions, like stimuli, have to be selected on each trial, and their selection is typically done in the same manner as is

stimulus selection. As such, the progressive determination and transitional biases will influence condition selection as well. For example, in a Stroop task, the researcher may program an experiment to select a condition (i.e., congruent, incongruent, or neutral) and then to select a particular item (e.g., the word RED in red) to present. Here, if the conditions are selected using random without replacement, both progressive determination and transitional bias will operate at the condition level. Each selection in which a particular condition is not selected increases the likelihood that that particular condition will be selected next (i.e., progressive determination). In addition, the number of condition repetitions will be less than that expected by chance (i.e., the transitional bias). There is no reason to expect that instituting a predictable sequence in terms of condition selection would be any less likely to influence behavior than it would in terms of stimulus selection. For example, in a Stroop-like task, the congruency of the previous trial can influence the current trial (e.g., Gratton, Coles, & Donchin, 1992), and if a participant knows the congruency of the next trial, he/she can modulate performance in accordance with that knowledge (Logan & Zbrodoff, 1982).

Balancing the Biases and Experimental Constraints

There may be situations in which one wants to avoid too many repetitions of the same stimuli or conditions. It is evident that such a deviation should be considered carefully. At a minimum, one should use the largest possible list that satisfies the experimental constraints. Ideally, a few participants would also be run through a fully randomized list, at least during the piloting phase, in order to assess whether these biases are contaminating the effect of interest.

Conclusion

Random without replacement is not random selection. This selection method induces at least two selection biases that demonstrably affect behavior. We have suggested a strategy that reduces the impact of these biases on performance and hope that this work will help improve the design of psychological experiments in the future.

AUTHOR NOTE

This work was supported by a Natural Sciences and Engineering Research Council of Canada (NSERC) postdoctoral fellowship to C.B. The author expresses considerable gratitude to one anonymous reviewer for the proof contained in the Appendix and to several anonymous reviewers for constructive comments on earlier drafts of this article. Supplemental materials, including a detailed demonstration of its application in E-Prime, can be found at bungalow.berkeley.edu/BRM/index.html. Correspondence concerning this article should be addressed to C. Blais, University of California, Berkeley, Helen Wills Neuroscience Institute, 132 Barker Hall MC 3190, Berkeley, CA 94720 (e-mail: cblais@berkeley.edu).

REFERENCES

- BERTELSON, P. (1965). Serial choice reaction-time as a function of response versus signal-and-response repetition. *Nature*, **206**, 217-218.
- BOYER, M., DESTREBECQZ, A., & CLEEREMANS, A. (2005). Processing

- abstract sequence structure: Learning without knowing, or knowing without learning? *Psychological Research*, **69**, 383-398.
- BRYSHAERT, M. (1991). Algorithms for randomness in the behavioral sciences: A tutorial. *Behavior Research Methods, Instruments, & Computers*, **23**, 45-60.
- CASTELLAN, N. J. (1992). Shuffling arrays: Appearances may be deceiving. *Behavior Research Methods, Instruments, & Computers*, **24**, 72-77.
- COHEN, J. D., MACWHINNEY, B., FLATT, M. R., & PROVOST, J. (1993). PsyScope: A new graphic interactive environment for designing psychology experiments. *Behavior Research Methods, Instruments, & Computers*, **25**, 257-271.
- EMERSON, P. L., & TOBIAS, R. D. (1995). Computer program for quasi-random stimulus sequences with equal transition frequencies. *Behavior Research Methods, Instruments, & Computers*, **27**, 88-98.
- GRATTON, G., COLES, M. G., & DONCHIN, E. (1992). Optimizing the use of information: Strategic control of activation of responses. *Journal of Experimental Psychology: General*, **121**, 480-506.
- HABER, R. N. (1966). Nature of the effect of set on perception. *Psychological Review*, **73**, 335-351.
- LOGAN, G. D., & ZBRODOFF, N. J. (1982). Constraints on strategy construction in a speeded discrimination task. *Journal of Experimental Psychology: Human Perception & Performance*, **8**, 502-520.
- NISSEN, M. J., & BULLEMER, P. (1987). Attentional requirements of learning: Evidence from performance measures. *Cognitive Psychology*, **19**, 1-32.
- PSYCHOLOGY SOFTWARE TOOLS (2002). E-Prime (Version 1.2) [Computer software]. Pittsburgh, PA: Author.
- REMILLARD, G., & CLARK, J. M. (1999). Generating fixed-length sequences satisfying any given n th-order transition probability matrix. *Behavior Research Methods, Instruments, & Computers*, **31**, 235-243.

APPENDIX

What follows is a formal proof followed by two properties of the resulting equation.

Let n be the number of unique stimuli, x the number of instances of each unique stimulus, and y the number of orderings of the xn elements, where each ordering is established using random selection without replacement.

Now, there are $(xn)!$ possible ways of ordering the xn elements, and random selection without replacement ensures that each of the $(xn)!$ orderings is equally likely to be generated (Castellan, 1992).

Let S be one of the unique stimuli.

For $1 \leq i \leq xn$, the probability that S appears in position i in a generated ordering is

$$P_S(i) = \frac{(x)(xn-1)!}{(xn)!} = \frac{x}{xn}.$$

This follows from the fact that of the $(xn)!$ possible orderings, $(x)(xn-1)!$ orderings have S in position i .

For $1 \leq i \leq xn-1$, the probability that S appears in position $i+1$ in a generated ordering, given that it appears in position i (i.e., that S repeats), is

$$P_S(i+1|i) = \frac{(x)(x-1)(xn-2)!}{(x)(xn-1)!} = \frac{x-1}{xn-1}.$$

This follows from the fact that of the $(x)(xn-1)!$ orderings with S in position i , $(x)(x-1)(xn-2)!$ orderings have S in position $i+1$.

For $i = xn$, the probability that S appears in Position 1 in a generated ordering, given that it appears in position xn of the preceding ordering, is clearly

$$P_S(1|xn) = P_S(1) = \frac{x}{xn}.$$

Now, S appears yx times in the sequence created by concatenating the y orderings.

For $1 \leq i \leq xn$, the number of times (out of yx) that S is expected to appear in position i of an ordering is

$$y \cdot P_S(i).$$

Therefore, the probability that S will repeat in the sequence is given by the following equation, where the numerator is the number of times (out of yx) that S is expected to repeat in the sequence.

$$\begin{aligned} PR_S &= \lim_{y \rightarrow \infty} \frac{y \cdot P_S(xn) \cdot P_S(1|xn) + \sum_{i=1}^{xn-1} y \cdot P_S(i) \cdot P_S(i+1|i)}{yx} \\ &= \lim_{y \rightarrow \infty} \frac{y \cdot \frac{x}{xn} \cdot \frac{x}{xn} + (xn-1) \cdot y \cdot \frac{x}{xn} \cdot \frac{x-1}{xn-1}}{yx} \\ &= \frac{xn-n+1}{xn^2}. \end{aligned}$$

This completes the proof.

The equation for PR_S has two properties. First, for $n > 1$, PR_S increases as x increases, and PR_S asymptotes at n^{-1} . This follows from the fact that the derivative of PR_S with respect to x is positive for all values of $x > 0$ and that the limit as x goes to infinity of PR_S is n^{-1} . Second, for $x > 0$, PR_S decreases as n increases, and PR_S asymptotes at 0. This follows from the fact that the derivative of PR_S with respect to n is negative for all values of $n > 0$ and that the limit as n goes to infinity of PR_S is 0.