AMERICAN PSYCHOLOGICAL ASSOCIATION

Psychological Review

ISSN: 0033-295X

2021, Vol. 128, No. 1, 1–44
http://dx.doi.org/10.1037/rev0000253

Serial Order in Perception, Memory, and Action

Gordon D. Logan
Vanderbilt University

This article asks whether serial order phenomena in perception, memory, and action are manifestations of a single underlying serial order process. The question is addressed empirically in two experiments that compare performance in whole report tasks that tap perception, serial recall tasks that tap memory, and copy typing tasks that tap action, using the same materials and participants. The data show similar effects across tasks that differ in magnitude, which is consistent with a single process operating under different constraints. The question is addressed theoretically by developing a Context Retrieval and Updating (CRU) theory of serial order, fitting it to the data from the two experiments, and generating predictions for 7 different summary measures of performance: list accuracy, serial position effects, transposition gradients, contiguity effects, error magnitudes, error types, and error ratios. Versions of the model that allowed sensitivity in perception and memory to decrease with serial position fit the data best and produced reasonably accurate predictions for everything but error ratios. Together, the theoretical and empirical results suggest a positive answer to the question: Serial order in perception, memory, and action may be governed by the same underlying mechanism.

Keywords: context updating, copy typing, serial order, serial recall, whole report

The problem of serial order is ubiquitous in perception, memory, and action. It challenges our ability to perceive structure in the world and to arrange objects, as in setting a table, understanding an equation, or seeing a word. It challenges our ability to remember what happened when and to recall sequences, such as conversations, the months of the year, or grocery lists. It challenges our ability to act coherently, to execute steps in the right order when cooking a meal, singing a tune, or typing a word. Serial order is the basis of language and STEM skills like logic, mathematics, and computing. It is an essential part of reading and literacy and a main culprit in dyslexia (Cowan et al., 2017) and dyscalculia (Fias, Menon, & Szucs, 2013). The broad purpose of this article is to ask whether these many phenomena are different manifestations of a single serial order mechanism or manifestations of different mechanisms attuned to each phenomenon. The broad purpose is addressed with two specific questions: One is empirical, asking whether serial order phenomena are the same in *whole report* tasks that tap perception (Sperling, 1960), *serial recall* tasks that tap memory (Conrad, 1964), and *copy typing* tasks that tap action (Sternberg, Monsell, Knoll, & Wright, 1978). These tasks maximize comparability across domains, as they all require sequential report of letter strings. The other question is theoretical, asking whether a single computational model can provide a coherent account of serial order phenomena in all three tasks. The effects may differ between tasks because of differences in overall accuracy. A computational model will be required to determine whether the effects are commensurate, that is, whether differences in effects can be accommodated by meaningful variation in model parameters. I generalize my Context Retrieval and Updating (CRU; Logan, 2018) model of skilled typing to account for encoding and retrieval of serial order in whole report, serial recall, and (novice) copy typing tasks, and test its ability to do so.

One Phenomenon or Many?

Serial order has been a burgeoning topic of research in perception, memory, and action, but there has been little integration across areas. Researchers in different areas study different tasks for different purposes and propose different models. Many studies of serial order in perception are aimed at "cracking the orthographic code," asking how order and identity information are bound together in letter strings in the course of reading words (Grainger, 2018). Researchers use whole and partial report tasks, identification, same-different judgments, and lexical decision tasks to address the binding process. Many studies of serial order in memory are aimed at memory for temporal order, using serial recall, cued recall, free recall, and recognition tasks to address these processes (Lewandowsky & Farrell, 2008). Many studies of serial order in action are aimed at producing sequences of action, using sequence learning, speaking, music, and typing tasks to address the under-

This article was published Online First August 17, 2020.

I am grateful to Jane Zbrodoff for inspiration and to Greg Cox, Jonathan Grainger, Alice Healy, Andrew Heathcote, Dora Matzke, Tom Palmeri, Sean Polyn, Philip Smith, and Trish Van Zandt for valuable discussion. I am grateful to Dakota Lindsey for programming the experiments and to Jana Ulrich for testing the participants, helping with the simulations, and managing heaps of data. Portions of this work were presented at the Context and Episodic Memory Symposium, the Control Processes meeting, and the annual meeting of the Psychonomic Society in 2019. Data, analyses, and the fitting, simulation, and data analysis programs are posted on the Open Science Framework at osf.io/f98kt.

Correspondence concerning this article should be addressed to Gordon D. Logan, Department of Psychology, Vanderbilt University, 301 David K. Wilson Hall, 111 21st Avenue South, Nashville, TN 37240. E-mail: gordon.logan@vanderbilt.edu

lying processes (Logan & Crump, 2011). Theories of serial order follow the empirical focus, addressing single tasks with little integration across research areas.

The literature has made it clear that it is possible to manipulate serial order in perception, memory, and action separately and selectively. A lexical-decision task requires perception of serial order in letter strings, which produces a single categorization (word or nonword) in memory, and requires a single keypress response (Ratcliff, Gomez, & McKoon, 2004). Cuing recall of an item with a marker that indicates its position in the list requires perception of a single item (the marker), serial ordering in memory to retrieve the item, and report of a single item (Oberauer, 2003). Typing the name of the color of an object requires perception of a single object, memory for a single attribute (color), and serially ordered action (keystrokes; Logan & Zbrodoff, 1998). This suggests that separate mechanisms are used in each task, but it remains possible that the results can be explained by a single mechanism operating under different constraints. The tasks require different decisions about different materials under different manipulations in different participants, which makes it hard to compare one effect to another. A stronger test of the single mechanism hypothesis would require comparing perception, memory, and action tasks that require the same decisions on the same materials under the same manipulations, and running the tasks in the same participants.

This article provides this stronger test in two experiments that compare whole report, serial recall, and copy typing tasks in the same participants using strings of random letters as materials. The three tasks require participants to report the strings in left-to-right order by typing them on a computer keyboard. The tasks differ primarily in conditions of exposure and instructions. The *whole report* task presents letter strings for 100 ms and requires immediate recall (Sperling, 1960); the *serial recall* task presents letter strings for 1000 ms to ensure accurate encoding and requires waiting until the display disappears to begin report (Conrad, 1964); and the *copy typing* task presents the strings throughout the response until the last letter is typed and requires immediate report (Sternberg et al., 1978).

I designed these tasks to be maximally similar to facilitate comparisons between them. I chose them because they are important tasks in their respective literatures. The whole report task was the control condition in studies of selective attention and partial report (Shibuya & Bundesen, 1988; Sperling, 1960). The random letter strings it employs are the "no-knowledge" controls in studies of word recognition and lexical decision (McClelland & Rumelhart, 1981) and the focus of some investigations of serial order (Adelman, 2011; Gomez, Ratcliff, & Perea, 2008). The serial recall task differs from most serial recall tasks in that the items are presented simultaneously instead of sequentially. However, many of the results are the same for simultaneous and sequential presentations and the theories are essentially the same. The copy typing task differs from everyday typing in requiring immediate responses to discrete stimuli instead of copying or composing text and it differs in using random letter strings instead of words and sentences. Nevertheless, it reveals important properties of serial ordering in typing (Sternberg et al., 1978). One might argue the three tasks are different versions of the same task under slightly different conditions. That is partly my point. Researchers study the same task for different purposes and there is not much communication between areas. My goal is to promote communication so each area can benefit from the others' insights into the common problem of serial order.

In each experiment, the tasks were compared on seven standard measures of serial report: list accuracy, serial position effects, transposition gradients, contiguity effects, error magnitudes, error types, and error ratios. Each experiment manipulated a theoretically important variable in each task. The first experiment manipulated list length, varying the number of letters from five to seven. List length is an important manipulation in all three tasks, because it bears on the presence and nature of capacity limitations. The second experiment manipulated within-list repetitions, so half of the strings contained repeated letters with lags of zero to three intervening letters between repetitions. Within-list repetitions are a major challenge for theories of serial order (Lashley, 1951), decisively ruling out some kinds of models.

One Theory or Many?

For a single theory to provide a unified account for serial order in perception, memory, and action, it must meet two criteria: First, it must provide adequate fits to data in each task and account for differences between tasks with meaningful variation in model parameters. An inadequate fit in any task would mean that the theory cannot provide a single account of serial order phenomena. Second, it must fit each task as well as or better than competing models. If one model fits perception better than another and the other model fits memory better, neither model can provide a single-theory account of serial order. Together, such fits would provide evidence on how serial order processes differ between tasks. Here I present a theory aimed at fulfilling the first criterion. I relate my theory to competing theories in the General Discussion but quantitative comparisons await future research.

The model fits cannot distinguish between a single mechanism that is applied to perception, memory, and action tasks and separate mechanisms that implement the same design principles—a single version of the model for all tasks or a different version of the model for each task. In both cases, a single set of computational principles governs serial order. The difference is in whether the computational principles are implemented in the same or different brain structures. I consider the question in the General Discussion in the section on Expertise, but definitive answers await future research.

A Context Retrieval and Updating Theory of Serial Report

The theoretical question, whether serial order relies on the same mechanism in perception, memory, and action, could be answered by applying any one of several existing theories of serial order from the different literatures. I chose to answer it by proposing and applying a *context retrieval and updating* (CRU) theory, which explains serial order in terms of an evolving context that contains fading records of the items experienced or recalled so far (Howard & Kahana, 2002; Logan, 2018). Items are associated with contexts made of previous items, so in effect, items are associated with each other This *item coding* perspective is underrepresented in studies of serial order, especially in serial recall and whole report where *position coding* is the dominant explanation (Brown, Preece, &

Hulme, 2000; Burgess & Hitch, 1999; Davis, 2010; Farrell, 2012; Farrell & Lewandowsky, 2002; Henson, 1998b; Houghton, 2018; Lewandowsky & Farrell, 2008). Good fits to data from all three tasks will support CRU as a candidate for a single mechanism of serial order, but it will not suggest that CRU is as good as or better than alternative theories of serial order. That would require comparative fitting of different theories of serial order, which is an important goal for future research. I address relations between CRU and other theories in the General Discussion.

CRU embodies my longstanding conjecture that cognitive control can understood as contextually driven memory retrieval. My instance theory of automaticity argued that contexts cue retrieval of past solutions, bypassing the need for thinking (Logan, 1988). My approach to task switching with Darryl Schneider argued that task cues act as contexts that combine with ambiguous targets to retrieve task-appropriate responses (Schneider & Logan, 2005, 2009). In each case, responses are associated with the contexts in which they occurred and retrieved when those contexts are reinstated. CRU also assumes that responses are associated with contexts. The new assumption is that the current response becomes part of the context that retrieves the next response. The context represents the history of past responses and that enables CRU to account for serial report (Logan, 2018).

The idea is new to me but familiar in the memory literature. The context retrieval and updating processes that are the core assumptions of CRU are taken from Howard and Kahana's (2002) *temporal context model* (TCM) and its intellectual descendants (Lohnas, Polyn, & Kahana, 2015; Polyn, Norman, & Kahana, 2009; Talmi, Lohnas, & Daw, 2019). TCM explains phenomena in free recall on the assumption that retrieval is driven by a current context that builds cumulatively across the list. Items are associated with the current context when they are first presented, and the associations guide retrieval of the items when a new current context matches the initial one. Howard and Kahana (2002) provide a formal analysis of the updating process, which CRU has adopted. Sean Polyn and his colleagues provided an important inspiration, fitting their *context maintenance and retrieval* (CMR) update of TCM to the sequence of individual responses participants made in free recall (Kragel, Morton, & Polyn, 2015; Morton & Polyn, 2016). This convinced me that serial report could be modeled at the level of individual responses, as in studies of attention and performance and studies of recognition memory, allowing the strong tests of theories that abound in those studies. CRU is focused on individual responses (Logan, 2018).

The CRU theory is implemented as a computer simulation that takes arbitrary lists of letters as input and retrieves them sequentially. Its computations are organized as four serial processing stages: encoding the items' identities, encoding their order, retrieving the items in order, and reporting them (see Figure 1). The probability of retrieving an item correctly in order is the product of the probabilities of succeeding at each of these stages. CRU models the computations in each stage, estimating the probability of success for each item in the list. My goal here is to extend the model to whole report, serial recall, and typing nonwords.

Encoding Items

CRU assumes that items are identified before their order is encoded. The item encoding stage (see Figure 1) takes the features of the letters in the string as input and identifies or gives a classification of each letter as output. CRU assumes that items are identified independently. This is a reasonable assumption for strings of random letters (Pelli, Farell, & Moore, 2003). Some models of orthographic processing assume letters are identified independently (Adelman, 2011), whereas others assume letters have already been identified before they are bound to locations and focus on the binding process (Grainger & Van Heuven, 2004; Houghton, 2018). Models of memory generally assume items are identified before order and identity are bound. Models of action generally extract movements from already-bound structures (Logan, 2018).

CRU assumes that letters are represented as points in a multidimensional feature space and similarity between letters is a negative exponential function of the Euclidian distance between them. When a letter is presented, its features activate the representations of all the letters in the alphabet in proportion to their similarity to the presented letter, the letter representations compete in proportion to their activation, and the winner of the competition becomes the encoded letter. The competition is implemented as a race between diffusion processes representing each letter, with the winner determining the identity of the encoded item. The racing diffusions allow estimates of the probability of encoding. How they do it is explained later.

I used a 25-dimensional feature representation for lowercase letters derived from response time measures of discrimination (Courrieu, Farioli, & Grainger, 2004) to represent the letters. The drift rates for the racing diffusion choice process are similarities calculated as exponential functions of distance. The drift rate v_i for letter i given that letter j is presented is:

$$v_i = \exp[-g \cdot d_{ij}] \tag{1}$$

where d_{ij} is the distance between letters i and j in feature space, and g is a sensitivity parameter scaling the effect of distance. Distance in feature space is Euclidian, defined as

$$d_{ij} = \sqrt{\left[\sum_{k=1}^{25} (x_{ik} - x_{jk})^2\right]}$$

where x_{ik} is the coordinate for letter i on feature (dimension) k. Formally, the result of item encoding is a unit vector that represents the encoded letter with a localist code: the element corresponding to the item is set to 1 and all other elements are set to 0. The subsequent context updating and retrieval processes require unit vector representations.

Defining item encoding in this way allows CRU to account for confusions among similar letters, which often occur in serial report tasks. The sensitivity parameter g modulates the amount of confusion (see Figure 2). Small values of g bring distant letters closer together and therefore increase confusions. Large values of g push nearby letters further away and therefore reduce confusions.

Encoding Serial Order

CRU encodes serial order by updating the *current context*, which represents the structure (list, word) containing the items and the items experienced so far (see Figure 1). CRU associates the items it encodes with the contexts in which they appear, and the

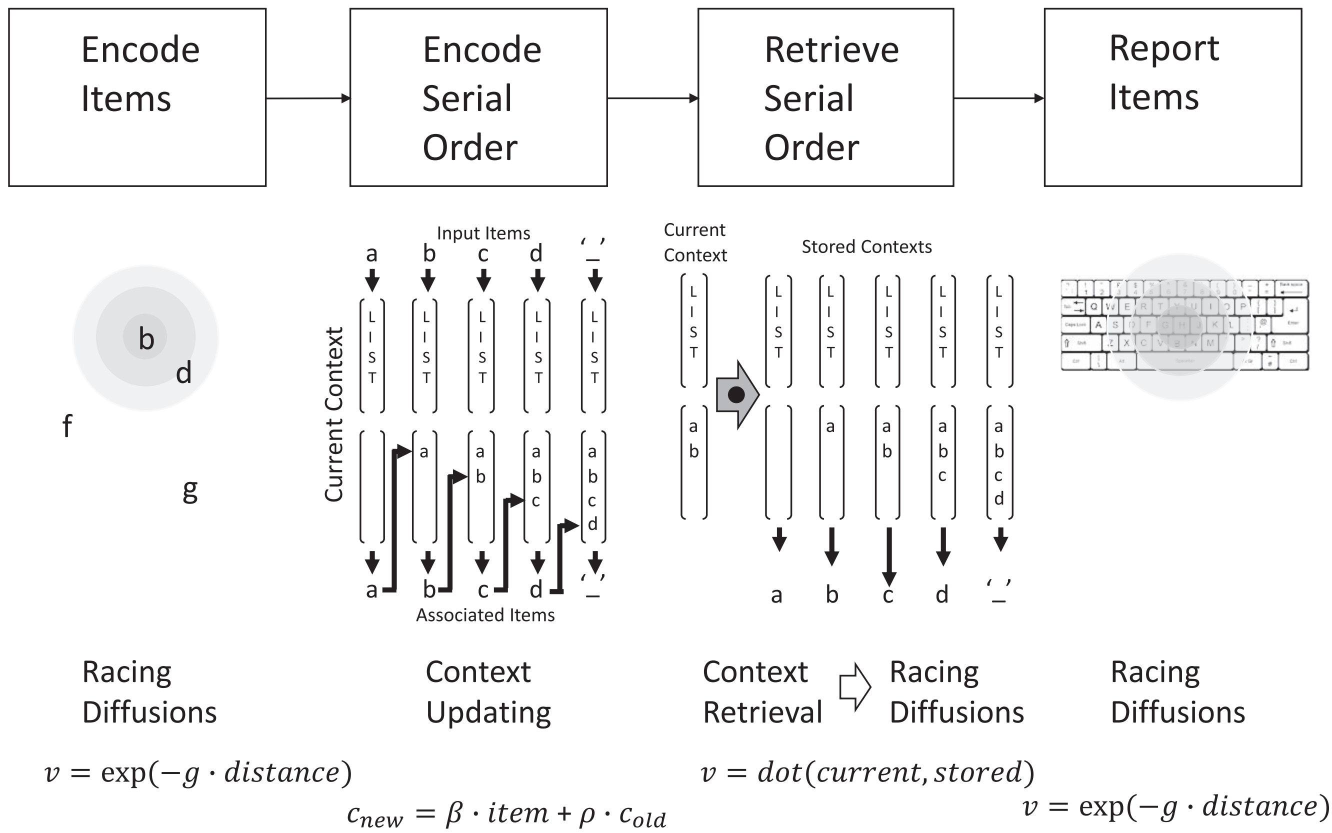


Figure 1. Context Retrieval and Updating model (CRU). The top row shows the flow of information through stages of processing. Items are encoded, then their serial order is encoded, then serial order is retrieved, and items are reported. The second row shows the representations that each stage processes. Item encoding and item report represent items as points in multidimensional space, so confusions are stronger among neighboring items than remote items. The gray circles represent the distance gradient set by the parameter *g* (Equation 1). Serial order is encoded by updating a context representation (vector). The second element in the second row depicts its evolution as the list is encoded. Its evolution is controlled by the parameter β (Equations 2–4). Each item is associated with the context in which it appears, producing a stored context that is accessed during retrieval. Serial order is retrieved by comparing the current context with all stored contexts and selecting the one that is most similar (with the highest dot product). The third row describes the computations in each stage and the fourth row gives the equations used to implement them.

contexts and their associations to items are stored for later retrieval. The *stored contexts* record the evolution of the current context, which builds cumulatively. Before the list is presented, the current context is initialized with a representation of the current list, becoming *List*. When list *abcd* is presented, the initial current context is associated with *a*, stored, and updated to become *List* + *a*. The new context is associated with *b*, stored, and updated to become *List* + *a* + *b*, which is then associated with *c*, and so on (see Figure 1).

More formally, items and structures (lists, words) are represented as unit vectors with 1 in the element corresponding to the item or structure and 0 in the other elements. The fits and simulations assume vectors with 1,032 elements. The first 26 elements represent the lowercase letters of the alphabet. Element 32 represents the space bar. Elements 33–1,032 represent the structure in which the items occur. In the typing model, the structures were words (Logan, 2018). In the current model, the structure is an abstract representation of the list the person experienced. I assume word representations and the contexts they contribute to exist prior to the experiment but list representations and the contexts they contribute to are generated anew for each different list. In principle, instance learning could turn list representations into permanent ones (Logan, 1988, 2018). I model the structures with localist codes, with 1 in the element representing the structure and 0 elsewhere, which does not capture similarities among the structures. However, the coding scheme could be extended to allow more than one structural element in each vector, thus representing similarity or hierarchy (Farrell, 2012; Henson, 1998b). For simplicity, only the current list is represented in the current implementation of CRU, so I use element 33 for every list, although in principle CRU could represent many different lists. Exploring such richer structures is an important goal for future research.

The current context, ***c***, is a 1032 element vector representing the list and the items that have occurred so far. It begins with all elements set to 0. Then the element representing the list is set to 1,

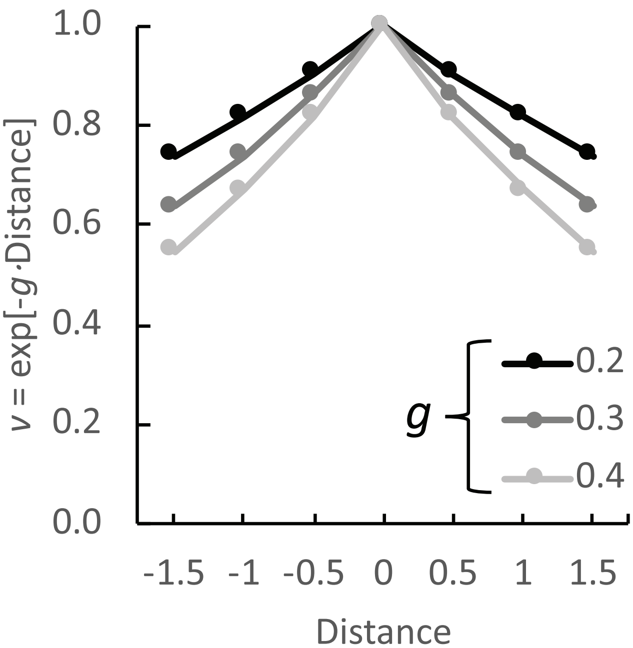


Figure 2. The effect of the sensitivity parameter *g* on the relation between distance and drift rate in the item encoding model (Equation 1).

producing a vector with length = 1. Then the first item is presented, associated with that context, and stored in memory. Then the context is updated by adding the vector representing the first item to the current context. The Nth input item $\boldsymbol{r}_N$ is weighted by β and the current context $\boldsymbol{c}_N$ is weighted by ρ, so

$$\boldsymbol{c}_{N+1} = \beta \cdot \boldsymbol{r}_N + \rho \cdot \boldsymbol{c}_N \tag{2}$$

(Howard & Kahana, 2002). The parameter β ranges from 0 to 1, reflecting the weight on the present; ρ also ranges from 0 to 1, reflecting the weight on the past. Following Howard and Kahana (2002) ρ is chosen so that the length of the new current context vector $\boldsymbol{c}_{N+1}$ is normalized to 1:

$$\rho = \sqrt{1 + \beta^2\left[(\boldsymbol{r}_N \cdot \boldsymbol{c}_N)^2 - 1\right]} - \beta(\boldsymbol{r}_N \cdot \boldsymbol{c}_N) \tag{3}$$

where $\boldsymbol{r}_N \cdot \boldsymbol{c}_N$ is the dot product of the item and current context vectors reflecting the similarity between them. With CRU's localist representations, a new item that is not already in the current context will have a dot product of 0 (will be orthogonal) because its nonzero element will not be occupied in the current context and all of its other elements equal zero. For unique items, the equation is simpler:

$$\rho = \sqrt{1 - \beta^2} \tag{4}$$

Equation 3 shows how CRU deals with repeated items. This will be important in accounting for within-list item repetitions in Experiment 2.

The effect of updating on the current context vector is illustrated in Figure 3. The left panel shows how element values change with updating. The List representation is initially set to 1.0 and then decreases by ρ with each updating step, so its value for serial position i is $\rho^{(i-1)}$. Each item representation is initially set to $1 \cdot \beta$ and then decreases by ρ with each updating step, so the value for serial position i for an item appearing in serial position j is $\beta \cdot \rho^{(i-j)}$ (for $i \geq j$ and 0 otherwise). This is illustrated in the left panels of Figure 3. The right panels present the same data as bar graphs to illustrate what the current context vector looks like at each serial position. Context vectors in adjacent positions are more similar than ones in remote positions.

CRU's serial order encoding mechanism requires the items to be presented to it sequentially, as updating is driven by new items from the item encoding mechanism. This is natural in studies of free recall and serial recall, where list items are typically presented one at a time. It is less obvious in studies of whole report and typing (and the present serial recall tasks), where list items are presented simultaneously. Theorists have long recognized that serial report of simultaneous lists poses a special problem of serial order—translating a parallel representation into a serial one (Bryden, 1967; Heron, 1957). A popular solution is to assume a serial scanning mechanism that operates on the simultaneous representation to produce a sequential representation (Bryden, 1967; Mewhort, Merikle, & Bryden, 1969). Serial processing is controversial (e.g., Coltheart & Rastle, 1994; Zorzi, 2000) and has not been resolved conclusively. Serial and parallel processing are notoriously hard to distinguish (Townsend, 1971, 1990; Townsend & Wenger, 2004). Some modern theories of orthographic processing assume serial scanning (Davis, 2010; Whitney, 2001), but they can be mimicked by a parallel model that assumes a gradient of activation that decreases over the list, so items earlier in the list are encoded before items later in the list (Adelman, 2011). CRU could implement serial scanning or a gradient of attention (Bundesen, 1990) to provide sequential input to the context updating process. I am currently exploring other alternatives.

There are two potential sources of error in CRU's serial order encoding mechanism. Noise could be added to input or current context vectors or to both, and the associations between contexts and vectors could vary in strength (Howard & Kahana, 2002). To simplify the modeling, I chose not to implement either source of error in CRU, forcing errors to come from item encoding or serial order retrieval.

Retrieving Serial Order

CRU retrieves serial order by comparing the current context to the set of stored contexts and reporting the item associated with the best-matching context (see Figure 1). The current context is initialized with the list representation, which is then matched to the stored contexts to retrieve the first item. The current context is then updated by adding the retrieved item, and matched to the stored contexts to retrieve the next item. Retrieving serial order uses the same updating process as encoding serial order (Equation 2), except that the retrieved item is added to the context instead of a presented item (see Figure 4). Updating the context with the retrieved item changes its similarities to the stored contexts, so a different stored context is selected. Ultimately, CRU retrieves an "end of list" marker, represented as a space bar response, which terminates report. For simplicity, CRU assumes the same β and ρ values for encoding and retrieving serial order, though in principle they

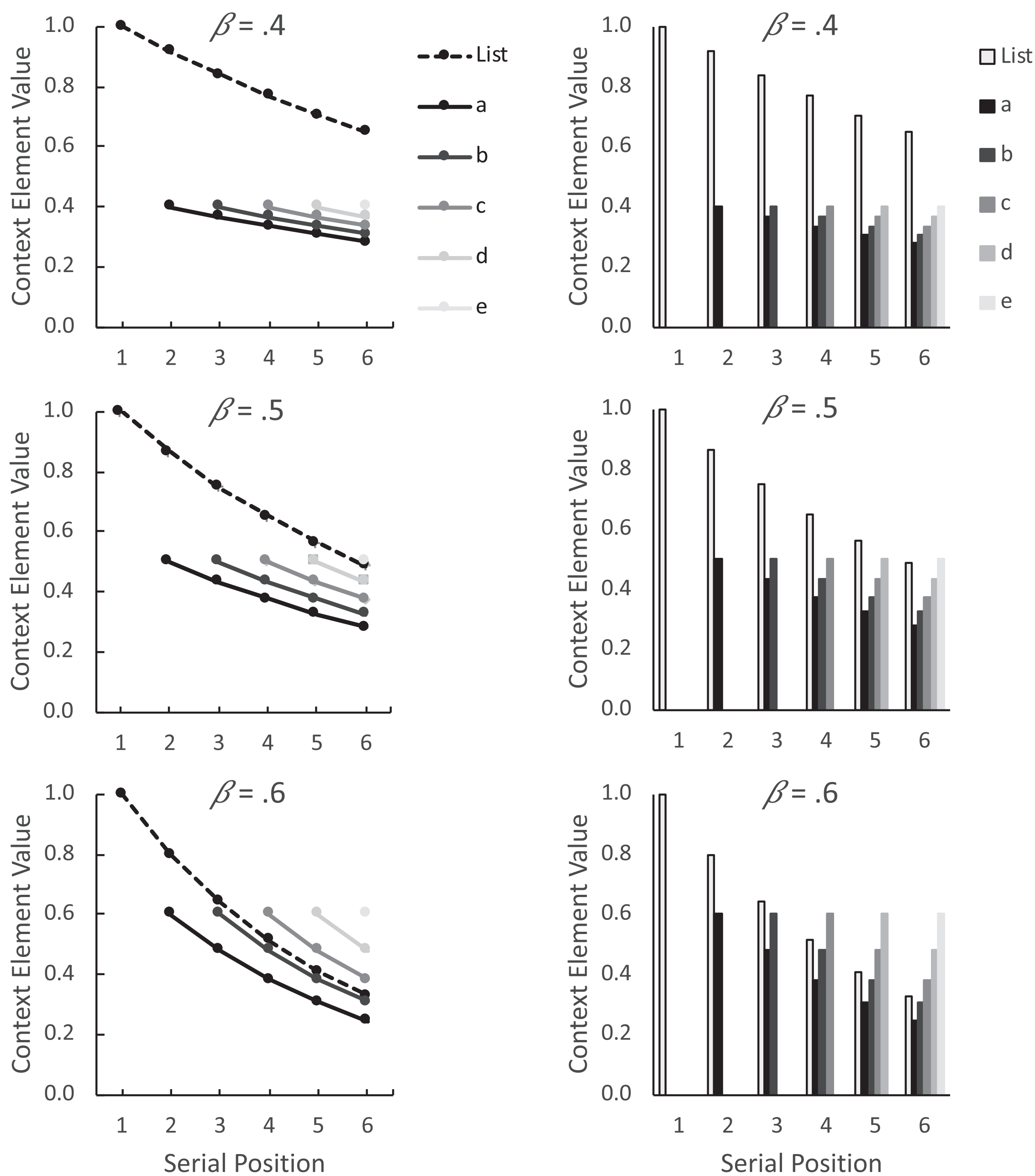


Figure 3. The evolution of element values over successive cycles of context updating. The left panel emphasizes the changes in value. The right panel emphasizes changes in the context vector.

could differ (Howard & Kahana, 2002; Lohnas et al., 2015; Polyn et al., 2009; Talmi et al., 2019).

Retrieval is based on similarity. The comparison process calculates similarity as the dot product (correlation) between current context ($\boldsymbol{c}_c$) and stored context ($\boldsymbol{c}_s$) vectors and chooses the stored context with the largest dot product using a racing diffusion decision process. The dot product ranges from 0 to 1 because the elements are positive and the vectors are normalized to length 1. The dot products serve as drift rates in the racing diffusion process,

$$v_i = \sum_{k=1}^{1032} \boldsymbol{c}_c(k) \cdot \boldsymbol{c}_{si}(k) \tag{5}$$

where $\boldsymbol{c}_c(k)$ is the kth element of the current context vector and $\boldsymbol{c}_{si}$ is the ith stored context vector. The racing diffusion process provides estimates of the probability of retrieval.

Context is updated by adding a single element to the current context, so adjacent contexts are more similar than remote ones. Context i contains one more element than context $i - 1$ and one less element than context $i + 1$. More remote contexts differ in two

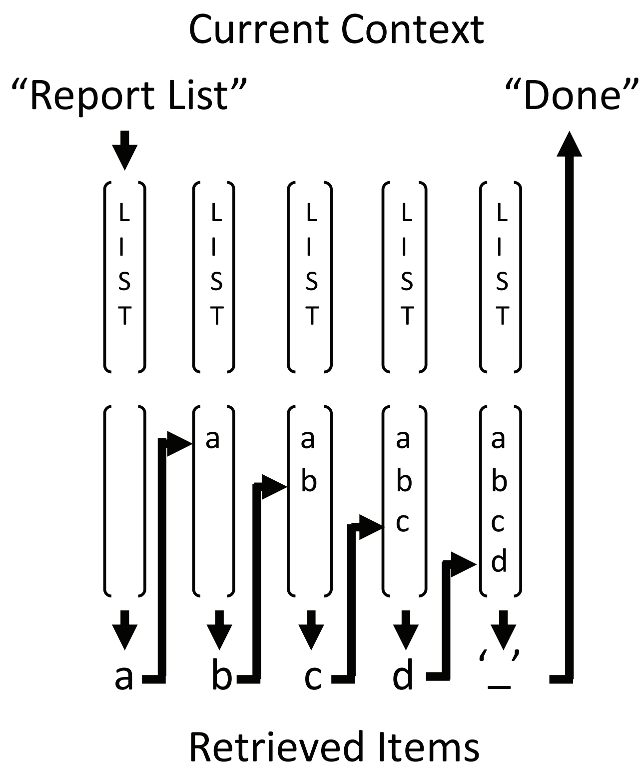


Figure 4. The evolution of the current context during retrieval. Retrieval begins when an element representing “list” is set in the current context vector (the rest of the vector is set to 0). The current context is matched to the stored contexts and an item is retrieved. That item is then added to the current context, and the cycle repeats until an “end marker” (the space character) is retrieved.

or more elements (see Figure 3). This relationship is captured in the dot products. Figure 5 shows dot products for each letter in a four-item list *abcd* and the space bar response that terminates it. Each line represents the similarity of the context associated with a serial position to stored contexts representing each position. The dot products are 1.0 for items in the correct position (*a* in position 1, *b* in position 2, etc.) and decrease smoothly with increasing distance from the correct serial position. The decrease is the same for items that precede and follow a given item: consider the symmetry of the line representing *c* in Figure 5.

The similarities depend on the elements of the context vectors, which in turn depend on β. The higher the value of β, the steeper the decline in element values over successive updates (see Figure 3), reducing the overlap between successive contexts. This is illustrated in Figure 5, which plots the dot products for different values of β. The higher the value of β, the steeper the decline in dot product over successive updates. The items become more distinct but the dot products are still graded symmetrically around the correct position.

Table 1 shows the stored contexts for list *abcdef* represented more abstractly in terms of β and ρ. The rows represent stored context vectors and the columns represent the values of the vector elements. The list element starts at 1.0 (i.e., as a unit vector) and decreases by a factor of ρ when each new item is encoded, so the value for the list element for context *i* is $\rho^{(i-1)}$. Each item starts at β (i.e., as a unit vector multiplied by β) and decreases by ρ as each subsequent item is encoded, so the value for the item presented in list position *j* is $\beta \cdot \rho^{(i-j)}$ for $i \geq j$ and 0 otherwise.

Imagine that CRU has retrieved *a*, *b*, and *c* and the current context has been updated so it now matches the 4th stored context. The dot products of this current context (4) with stored contexts

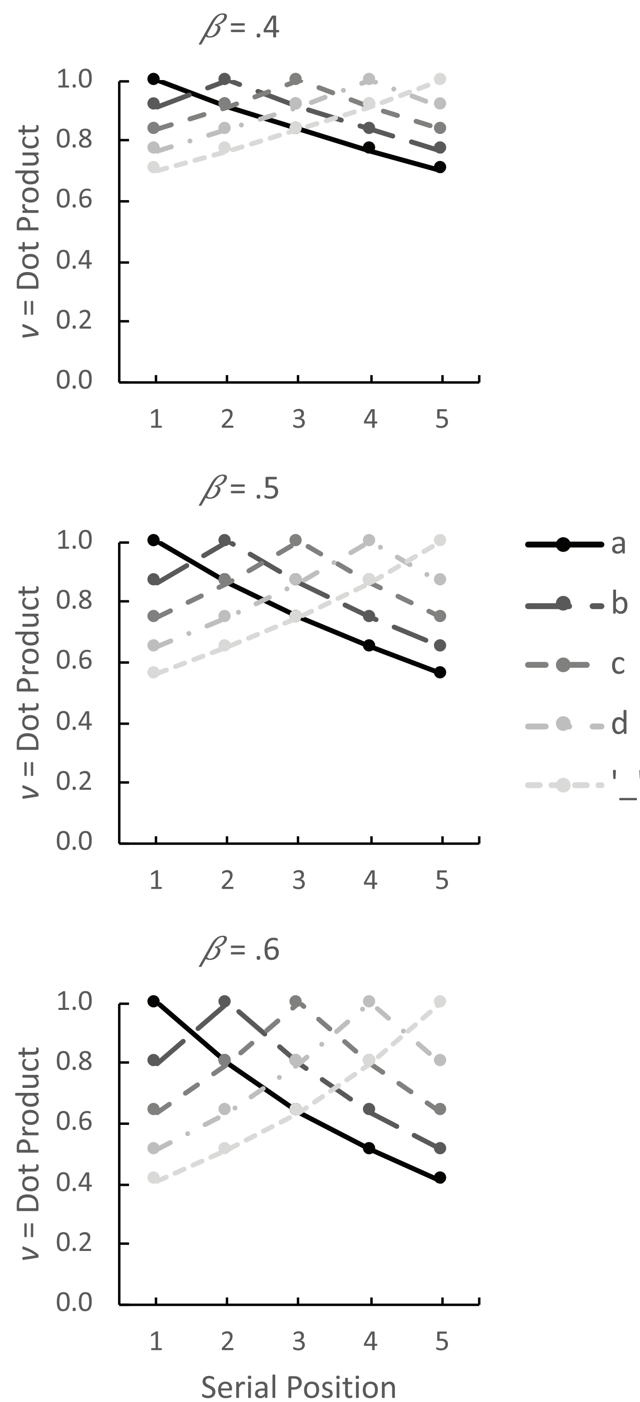


Figure 5. Similarities among stored context vectors for a four-item list, *abcd*, calculated as dot products, which serve as drift rates in the racing diffusions. The different panels show the effect of β on the similarities.

Table 1
Representations of Stored Context Vectors Encoded When Presented With List abcdef

	abcdef							
Position	a	b	c	d	e	f	List	Assoc.
1	0	0	0	0	0	0	1.0	a
2	β	0	0	0	0	0	ρ	b
3	$\beta \cdot \rho$	β	0	0	0	0	ρ^2	c
4	$\beta \cdot \rho^2$	$\beta \cdot \rho$	β	0	0	0	ρ^3	d
5	$\beta \cdot \rho^3$	$\beta \cdot \rho^2$	$\beta \cdot \rho$	β	0	0	ρ^4	e
6	$\beta \cdot \rho^4$	$\beta \cdot \rho^3$	$\beta \cdot \rho^2$	$\beta \cdot \rho$	β	0	ρ^5	f
7	$\beta \cdot \rho^5$	$\beta \cdot \rho^4$	$\beta \cdot \rho^3$	$\beta \cdot \rho^2$	$\beta \cdot \rho$	β	ρ^6	'_'

Note. Rows represent the context vectors for positions 1–7. Columns a–f and List represent the elements of the context vector. Column assoc. indicates the letter that is associated with the context vector.

1–7 (calculated as the sum of the products of corresponding columns in the two rows following Equation 5) are:

$$
\begin{aligned}
\text{dot}(4,1) &= \rho^3 && = \rho^3\\
\text{dot}(4,2) &= \beta^2 \cdot \rho^2 + \rho^4 && = \rho^2\\
\text{dot}(4,3) &= \beta^2 \cdot \rho^3 + \beta^2 \cdot \rho + \rho^5 && = \rho\\
\text{dot}(4,4) &= \beta^2 \cdot \rho^4 + \beta^2 \cdot \rho^2 + \beta^2 + \rho^6 && = 1.0\\
\text{dot}(4,5) &= \beta^2 \cdot \rho^5 + \beta^2 \cdot \rho^3 + \beta^2 \cdot \rho + \rho^7 && = \rho\\
\text{dot}(4,6) &= \beta^2 \cdot \rho^6 + \beta^2 \cdot \rho^4 + \beta^2 \cdot \rho^2 + \rho^8 && = \rho^2\\
\text{dot}(4,7) &= \beta^2 \cdot \rho^7 + \beta^2 \cdot \rho^5 + \beta^2 \cdot \rho^3 + \rho^9 && = \rho^3
\end{aligned}
$$

The dot products peak at 1.0 for the matching item and decrease symmetrically as distance from the matching item increases: dot(4, 3) = dot(4, 5), dot(4, 2) = dot(4, 6), and dot(4, 1) = dot(4, 7). More generally, the dot product between an item in position i and an item in position j is:

$$\text{dot}(i, j) = \rho^{|i-j|} \tag{6}$$

The $|i - j|$ term implies that the dot product declines symmetrically around the matching position. This relation was new to me but familiar to Howard and Kahana (2002) and Murdock (1997), who showed it was a property of TCM and a general property of evolving context models. I find it interesting that the drift rate depends only on ρ.

One may wonder why CRU assumes two context updating processes, one for encoding serial order and one for retrieving it. It would be simpler to assume that the items accessed during serial order encoding were reported without a subsequent retrieval process. The complexity is made necessary by the procedures of whole report and serial recall tasks: The items to be reported are no longer physically present at the time of report. Report must be based on some internal representation (Sperling, 1963). CRU's order encoding processes construct that representation and its retrieval processes extract reported items from it.

Reporting Items

CRU's serial order retrieval process retrieves abstract representations of items that must be translated into motor commands for responses. The item report process in Figure 1 fulfills this purpose. If the responses are typewritten, the abstract codes are locations on the keyboard, represented as points in two-dimensional space (Logan, 2018). The locations serve as cues for the retrieval of motor commands that select the fingers and movement trajectories required to strike keys in those locations (Rosenbaum, Loukopoulos, Meulenbroek, Vaughan, & Engelbrecht, 1995; Rosenbaum, Meulenbroek, Vaughan, & Jansen, 2001). CRU assumes key locations are activated in proportion to their distance from the target location. These activations become drift rates in a racing diffusion process that chooses the motor commands, so erroneous keystrokes are sometimes directed to adjacent keys (Logan, 2018). Drift rate for response i is a negative exponential function of distance from the target location j, using Equation 1, reproduced here for convenience:

$$v_i = \exp[-g \cdot distance_{ij}]$$

This formulation was natural for representing the goals for motor programs in skilled typing where the response apparatus is clearly two dimensional. Indeed, I chose to model item encoding as finding points in multidimensional space because of the success of that approach in modeling responding. The goals for motor programs in other modalities could be modeled as points in multidimensional feature space, such as speech or continuous report tasks (Ratcliff, 2018; Smith, 2016).

I did not include item report in any of the models I fit in this article. One reason was that motor errors (striking adjacent keys) were rare and not much different between tasks in the two experiments reported below. Both experiments had participants type nonwords, which is slower than typing words and perhaps less prone to motor errors. Another reason was to keep the models simple. Adding an item report stage would add at least one parameter and as many as three or four, and that would increase the time required to fit the models substantially. The responses were the same in all three tasks and I did not expect differences between tasks, so I assumed that the model reported the items it retrieved with no further errors in item report (i.e., the probability of reporting an item correctly = 1.0).

Racing Diffusion Decision Process

Each stage involves choosing one of several possible alternatives based on the similarities among the alternatives. I model the choice process as an *independent race between diffusions* (Logan, 2018; Logan, Van Zandt, Verbruggen, & Wagenmakers, 2014; Tillman, Van Zandt, & Logan, 2020). Each runner is a diffusion to a single bound governed by a rate parameter v and a threshold θ. The finishing time is characterized by the Wald distribution

$$f(t) = \theta(2\pi t^3)^{-1/2} \cdot exp\left[-\frac{1}{2t}(vt-\theta)^2\right] \tag{7}$$

The race between runners is characterized generally as

$$f(t, i) = f_i(t)\prod_{j \neq i}^{N}[1 - F_j(t)] \tag{8}$$

The probability that runner i wins the race is the integral of Equation 8:

$$P(i) = \int_0^\infty f(t, i)dt \tag{9}$$

The likelihood function for the racing diffusion model is obtained by substituting Equation 7 into Equation 8:

$$f(t,i) = \theta_i (2\pi t^3)^{-1/2} \cdot exp\left[-\frac{1}{2t}(v_i t - \theta_i)^2\right] \times \prod_{j \neq i}^{N} [1 - \Phi(t^{-\frac{1}{2}}(v_j t - \theta_j)) - \exp(2 v_j \theta_j)\Phi(-t^{-\frac{1}{2}}(v_j t - \theta_j))] \quad (10)$$

where v_i and θ_i are the drift rate and the threshold for alternative i and Φ is the cumulative normal distribution. The probability of choosing response i is obtained by integrating Equation 10 with respect to time (i.e., applying Equation 9). Note that the drift rate is not a free parameter in any stage of processing. It is determined by similarity—the distance in item encoding and report and the dot product in serial order retrieval—which is determined by the structure of the stimuli, context representations, and response representations and the sensitivity (g) and updating parameters (β) that modulate them. Although the notation allows threshold to vary among alternatives, I held it constant in all the fits.

I chose the racing diffusion process because it is a simple and tractable way to model immediate performance in serial report, which was the main focus of the modeling. Racing diffusions address response probability and response time distributions, which are the most common measures of immediate performance. Equation 10 provides the likelihood for choices among many alternatives (26 letters, 5–7 list items). Models of response time that fit two-choice data (somewhat) better than independent racing diffusions do not generalize easily to multiple choice responses (e.g., Ratcliff, 1978; but see Ratcliff, 2018). Many models of serial order violate independence, using lateral inhibition between response alternatives in the choice process, following Grossberg (1978), but likelihoods have to be simulated (Usher & McClelland, 2001) instead of calculated (Equation 10). That is less satisfying theoretically and it adds substantially to the cost of fitting the models. Since my purpose was not to test alternative models of the decision process, I opted for the simpler alternative.

Evaluating the Model

Fitting the model. Inspired by Polyn and colleagues (Kragel et al., 2015; Morton & Polyn, 2016), I fit CRU to the sequence of 3456 keystrokes from the 576 lists each participant reported in each experiment. The likelihood for each keystroke was calculated as the product of the probability that the letter was encoded, stored, retrieved, and reported correctly. That is

$$P_{Letter} = P_{Encode} \times P_{Store} \times P_{Retrieve} \times P_{Report}$$

For simplicity, I assume that letters are stored and reported with perfect accuracy (i.e., $P_{Store} = P_{Report} = 1$), so P_{Letter} reduces to

$$P_{Letter} = P_{Encode} \times P_{Retrieve}$$

where P_{Encode} and $P_{Retrieve}$ are obtained by integrating Equation 10 using drift rates from Equations 1 and 5, respectively. The likelihood of reporting all the letters in the list is the product of reporting each letter:

$$P_{List} = P_{Letter1} \times P_{Letter2} \times \cdots \times P_{LetterN}$$

and the likelihood of reporting all 576 lists is the product of the probability of reporting each list:

$$P_{All} = P_{List1} \times P_{List2} \times \cdots \times P_{List576}$$

I fit the data by maximizing the likelihood of reporting all lists for each participant. Because products of probabilities become small very quickly, I calculated the logs of the probabilities and maximized likelihood by minimizing the negative log likelihood of the data given the parameters using fmincon in Matlab. Model comparison was based on BIC scores to adjust for differences in the number of parameters, where

$$BIC = -2 \cdot \log(likelihood) + \ln(N) \cdot k,$$

N is the number of observations (roughly 3456 per participant), and k is the number of parameters in the model. The model with the lowest BIC is preferred.

To determine the effects of the parameters, I fit several versions of the model to the same data from each participant. The *baseline* (B) model required the same encoding (g) and retrieval (β) parameters for all three tasks. The *encoding* (E) model allowed different g parameters for each task but only one β for all three tasks. The *serial order* (SO) model allowed different βs for each task but only one g for all three tasks. The encoding + serial order (E+SO) model allowed different gs and βs for each task. Model comparisons reveal the importance of encoding (B vs. E) and serial order (B vs. SO) separately and jointly (B vs. E+SO) in accounting for differences between tasks.

Although the racing diffusion decision process can predict both response time and response probability, I chose to fit response probabilities (error data) and not response times at this stage in model development. Two major problems must be overcome to model response times. First, CRU models decision times in two or four stages, so the final response time distribution would be the convolution of the finishing time distributions of each stage (Equation 10), which would be very complex. Second and more fundamental, although it is clear when responses end in serial report tasks (i.e., when the key is pressed), it is not at all clear when responses begin. The "stimulus" for the response is internal and there is no way to measure when it occurs. I set these problems aside for future research and focused on response probability instead.

Without the constraint of response time, threshold and drift rate tended to trade off in the fitting. To focus the fits on drift rates, which are the parameters the model predicts (Equations 1 and 5), I fixed threshold at 200 for item encoding and serial retrieval. Consequently, the values of g and β estimated in the fits are conditional on $\theta = 200$. Different values of threshold produced values of g and β that differed somewhat in absolute magnitude but showed the same patterns across tasks. Details of the fitting process are described in Appendix A. A limited parameter recovery study is reported in Appendix B.

Evaluating model predictions. The BIC values from the fits indicate which of the alternative models fits best, but they do not indicate whether the fit to the data is good or bad. That has to be tested separately, by generating predictions and comparing them with the observed data (Palminteri, Wyart, & Koechlin, 2017). To generate predictions, I simulated the model for each participant, using the participant's best-fitting parameters, giving it the same sequences as the participant, and asking it to make the same responses—a series of 3,456 keystrokes—replicated 1,000 times. Then I scored the simulated sequences using the same calculations I performed on the participant's observed sequences to predict list accuracy, serial position effects, transposition gradients, contiguity effects, error magnitudes and error distances. I compared these predicted summary statistics with the observed ones with least squares measures—the *correlation* (r) and the *root mean squared deviation* (*rmsd*) between observed and predicted values—which reflect how closely they agree.

It is worth emphasizing that the parameters were fixed at the best-fitting values throughout the simulations that generated the data from which summary statistics were calculated. The same parameters predict all the effects.

Unit of replication. The models are fitted to data from individual participants, so the unit of analysis and the unit of replication is the individual participant (Smith & Little, 2018). All models are fit individually to all participants, so model comparisons are conducted within each participant. There were 576 lists per participant with an average of six letters per list, which resulted in (about) 3,456 keypress responses per participant. That should be sufficient to constrain the model fits, which are based on two to 11 parameters.

Experiment 1: List Length in Whole Report, Serial Recall, and Copy Typing

The first experiment addresses the empirical and theoretical relations between serial order phenomena in perception, memory, and action. I presented lists of random consonants in whole report, serial recall, and copy typing tasks, manipulating list length: five, six, or seven letters. The tasks differed in instructions and stimulus presentation. Responses were typed on the computer keyboard. In whole report, lists were exposed for 100 ms and reported immediately. In serial recall, lists were presented for 1,000 ms and reported after the list disappeared. In copy typing, the lists were reported immediately and presented until the last response was registered (a space bar press to indicate the end of the list). Each participant performed each task using the same materials and the same responses to maximize the comparability of the tasks.

The empirical answer to the question, "are serial order phenomena the same in perception, memory, and action," was obtained by comparing list accuracy, serial position effects, transposition gradients, contiguity effects, error magnitudes, error types, and error ratios between whole report, serial recall, and typing tasks. The theoretical answer was obtained by fitting models that differed in the way item encoding (g) and serial retrieval (β) parameters varied between conditions. The models followed a factorial structure, examining the effects of allowing versus not allowing parameters to vary between tasks to assess the role of each parameter in accounting for the data (Shen & Ma, 2019). The best-fitting model was simulated on the same lists as the participant and the simulated report sequences were analyzed in the same way as the actual data, producing predicted serial position effects, transposition gradients, contiguity effects, error magnitudes, error types, and error ratios that can be compared with the actual data. An adequate model should account for all these effects with one set of parameters whose values vary meaningfully between tasks.

Method

Participants. Twenty-four volunteers from the Vanderbilt community served in the experiment for course credit or $12 for serving in a single 1-hr session. Six identified as male, 17 as female, and one as male to female trans. They were selected for their self-professed ability to type 40 words per minute (WPM) or more. Their skill was confirmed on a typing test we have used for many years (Logan & Zbrodoff, 1998), which involved typing a paragraph of approximately 100 words espousing the many virtues of border collies. Their mean speed on the typing test was 64.36 WPM (SD = 14.80), and their mean accuracy was 91.72% (SD = 4.38).

Apparatus and stimuli. The experiment was run in E-Prime 2.0 (Psychology Software Tools, 2012) on ASUS M32BF desktop computers with BenQ XL2411Z flat screen monitors. Responses were taken from standard QWERTY keyboards with the backspace keys disabled. Lists were random strings of lowercase consonants, five, six, or seven letters in length.

Procedure. The experiment began with informed consent, the typing test, and the experimental trials. There were 576 experimental trials preceded by 15 practice trials in which each list length occurred equally often. The 576 trials were divided into three blocks of 192, one for each task (type, report, recall) and the order of tasks was counterbalanced across participants. List length (five, six, or seven letters) varied randomly within each block. Each trial began with a fixation cross (+) in the center of the screen for 500 ms. It was replaced with the list for that trial, which was presented for 100 ms in the whole report task, 1,000 ms in the serial recall task, and until the last response in the copy typing task. In whole report and serial recall, the lists were replaced by a blank screen when their exposure duration was complete, which was exposed until the last response was registered. In all three tasks, there was a 1,000-ms intertrial interval with a blank screen before the next trial began. Participants were run individually in separate testing rooms.

Participants were told that they would see lists of letters and their task was to report them in left-to-right order. They were told the sequence events on each trial and told that the lists would differ in duration for whole report, serial recall, and typing tasks. In whole report and typing, they were told to begin typing immediately. In serial recall, they were told not to begin typing until the list disappeared from the screen. They were told to type as quickly and accurately as possible but not to correct their mistakes as the backspace key had been disabled. They were told to guess if they were unsure of a letter. They were allowed breaks every 96 trials and each session took about 1 hr.

Results

Model fits. My main goal is to evaluate CRU's ability to account for differences between tasks in terms of meaningful variation in its parameters. To address this goal, I fit two sets of models, the *nondecrease* models described above, in which g and β were held constant across list position, and *decrease* models in which g and β decreased with list position.

Nondecrease models. Four nondecrease models were fit to the data for each participant separately. The details of the fitting process are presented in Appendix A. The first model was a *baseline* model in which g and β were not allowed to vary between tasks. The other three models allowed g or β or both to vary between tasks. The *encoding* model (E) allowed g to vary between tasks but held β constant. The *serial order* model (SO) allowed β

Table 2

Experiment 1 Nondecrease Models: Measures of Goodness of Fit and Best-Fitting Parameter Values for Baseline (B), Encoding (E), Serial Order (SO), and Encoding + Serial Order (E+SO) Models in Experiment 1

Measure	Likelihood	BIC	g_{Type}	g_{Recall}	g_{Report}	β_{Type}	β_{Recall}	β_{Report}
Base	4223.81	8454.69		.2513			.4687	
Encoding	4088.58	8165.52	.3395	.2542	.2147		.4687	
Serial Order	4075.68	8192.30		.2515		.5947	.4476	.4257
E+SO	3940.68	7902.59	.3395	.2542	.2148	.5947	.4477	.4257

Note. BIC = Bayesian information criterion.

to vary between tasks but held g constant. The *encoding plus serial order* model (E+SO) allowed both g and β to vary between tasks. Comparisons among these models allow inferences about the necessity of allowing g and β to vary in accounting for the data. The mean likelihoods, BIC scores, and parameter values across the 24 participants are presented for each model in Table 2.

The model comparisons in Table 3 show that the E model fit better than the baseline model in 23 participants, and the E+SO model fit better than the SO model in 23 participants. This indicates that variation in item encoding (g) between tasks is necessary to account for performance. Table 3 also shows that the SO model fit better than the baseline model in 23 participants and the E+SO model fit better than the E model in 23 participants. This indicates that variation in serial retrieval (β) between tasks is necessary to account for performance. Finally, the E+SO model fit better than the baseline model in 24 participants, better than the E model in 23 participants, and better than the SO model in 23 participants. This indicates that variation in both g and β are necessary to account for the data.

The likelihoods and BIC scores in Table 2 identify the best-fitting model, but the question remains whether the fits were produced by meaningful variation in the model parameters. The parameter values in Table 2 suggest the variation is meaningful. The encoding parameter g decreased from typing to serial recall to whole report, suggesting poorer encoding of items in tasks with shorter exposure durations. The serial order parameter β also decreased from typing to serial recall to whole report, suggesting more emphasis on maintaining context and less on distinguishing new items as exposure duration decreased. Thus, g and β both vary meaningfully between tasks. Interestingly, estimates of β were the same whether or not g varied between task, and estimates of g were the same whether or not β varied between tasks.

Decrease models. I generated predictions from the best-fitting nondecrease model (E+SO) and found that it mispredicted serial position curves and the distribution of error types across serial position. In the observed serial position curves, the first serial position was reported very accurately in each task and differences emerged across subsequent serial positions, with the curves fanning out from this position. The predicted serial position curves differed substantially in the first serial position due to differences in g and β between tasks. The observed error type distributions were clustered around the later serial positions, whereas the predicted functions were flat.

To accommodate these results, I developed decrease models in which g and β have the same initial values for all three tasks (g_{Max} and β_{Max}) but decrease across serial position. The decrease parameters ($g_{Decrease}$ and $\beta_{Decrease}$) either vary between tasks or are held constant. Thus, the encoding parameter for serial position i, g_i, is

$$g_i = g_{Max} \cdot g_{Decrease}^{i-1} \tag{11}$$

and the serial order parameter for serial position i, β_i, is

$$\beta_i = \beta_{Max} \cdot \beta_{Decrease}^{i-1} \tag{12}$$

with $g_{Decrease}$ and $\beta_{Decrease}$ ranging from 0 to 1. Thus, the decrease models were the same as the nondecrease models except that g_i (Equation 11) replaced g in Equation 1 and β_i (Equation 12) replaced β in Equation 2, and more parameters were fitted. I have no strong commitment to the form of the decrease in Equations 11 and 12. I wanted a monotonic decrease that would not go below 0. I could have used separate g and β parameters for each serial position but that would increase the number of parameters substantially to five to seven per task compared with eight for all three tasks in the decrease models (1 g_{Max} for all tasks and 1 g_{Decrease} for each task; 1 β_{Max} for all tasks and 1 $\beta_{Decrease}$ for each task).

I compared three decrease models with the baseline model from the nondecrease fits: an *encoding decrease* model (ED) in which g

Table 3

Experiment 1 Nondecrease Models: Model Comparisons Assessed by Differences in BIC Scores, Number of Participants Showing the Difference, and t Tests of the Difference in Experiment 1

Measure	Base – E	Base – SO	Base – E+SO	E – SO	SO – E+SO	E – E+SO
BIC difference	289.17	263.39	552.10	25.79	262.92	288.71
N different	23	23	24	16	23	23
t(23)	10.714	8.632	10.358	1.161	8.598	10.609

Note. Base = baseline model; BIC = Bayesian information criterion; E = encoding model; SO = serial order model; E+SO = encoding + serial order model.

Table 4
Experiment 1 Decrease Models: Measures of Goodness of Fit and Best-Fitting Parameter Values for Baseline (B), Encoding Decrease (ED), Serial Order Decrease (SOD), and Encoding Decrease + Serial Order Decrease (ED+SOD) Models in Experiment 1

Measure	Likelihood	BIC	g_{Max}	g_{DType}	$g_{DRecall}$	$g_{DReport}$	β_{Max}	β_{DType}	$\beta_{DRecall}$	$\beta_{DReport}$
Baseline	4223.81	8454.69	.2513		1.0000		.4687		1.0000	
ED	3907.14	7831.96	.3609	.9775	.8897	.8278	.4687		1.0000	
SOD	4045.20	8108.09	.2513		1.0000		.5743	.9988	.9251	.9061
ED+SOD	3728.46	7485.21	.3612	.9775	.8896	.8276	.5744	.9988	.9250	.9016

Note. BIC = Bayesian information criterion. Parameter values of 1.0000 were fixed, not fitted.

was allowed to decrease differently between tasks but β was held constant, a *serial order decrease* model (SOD) in which β was allowed to decrease differently between tasks but *g* was held constant, and an *encoding decrease plus serial order decrease* model (ED+SOD) in which both *g* and β were both allowed to decrease differently between tasks. The models were fit to each participant separately. The mean goodness of fit measures and parameter values are presented in Table 4. The parameter values and measures of goodness of fit for each participant for the best-fitting ED+SOD model appear in Table A1.

Model comparisons in Table 5 showed that all three decrease models fit better than the baseline model in all 24 participants, and the ED+SOD model fit better than the ED model and the SOD model in all 24 participants. These results indicate that variation in item encoding and serial retrieval parameters (*g* and β) between tasks and across serial positions is necessary to achieve good fits.

The parameter values in Table 4 changed meaningfully across tasks. Encoding (*g*) decreased more severely from typing to serial recall to whole report tasks, and so did emphasis on new items (β) in retrieving serial order. The reduction of *g* and β across serial position is consistent with left-to-right scanning and with differential attention to the first part of the list.

The decrease models fit better than the nondecrease models. Encoding decrease models fit better than encoding nondecrease models in 24 participants (E-ED BIC difference = 359.34, $t(23) = 9.956$); serial order decrease models fit better than serial order nondecrease models in 19 participants (SO-SOD BIC difference = 57.42, $t[23] = 3.703$); and encoding decrease plus serial order decrease models fit better than encoding plus serial order nondecrease models in 24 participants (E+SO—ED+SOD BIC difference = 417.38, $t[23] = 12.697$). Thus, decrease of *g* and β over serial position seems to be necessary to produce good fits.

Model predictions. I generated predictions for the best-fitting model (ED+SOD) for each participant individually, using their parameters to drive a simulation of CRU. The simulation took the same lists as the participant as input and produced a sequence of responses for each list as output, replicated 1,000 times. The simulated sequences were then scored with the same programs that were used to score the participant's data, producing seven sets of summary statistics: list accuracy, serial position effect, transposition gradient, contiguity effect, error magnitude, error type, and error ratio. The accuracy of CRU's predictions was assessed individually for each participant, with correlation (*r*) and root mean squared deviation (*rmsd*) between observed and predicted values.

Two points deserve emphasis: First, the predictions were based on the parameter values derived from the fits with no further adjustment. Second, for each subject, predictions for all seven summary measures were generated from a single set of eight parameters (g_{Max}, g_{DTupe}, $g_{DRecall}$, $g_{DReport}$, β_{Max}, β_{DType}, $\beta_{DRecall}$, and $\beta_{DReport}$). The summary measures provide different perspectives on serial order but they all derive from the same processes operating on the same representations, which is the behavior we measured in the experiments and modeled with variants of CRU.

List accuracy. The probability of reporting the entire list correctly in order (*list accuracy*) is a common measure in studies of serial recall. It is the basis for measuring the memory span, which is defined as the longest list that can be recalled in order 50% of the time. The mean observed and predicted list accuracies are plotted in Figure 6 as a function of list length and task. The observed accuracies decreased from typing to serial recall to whole report and decreased with list length in each task, although the decrease was steeper for serial recall than for the other tasks. The predicted accuracies also decreased with task and list length but underpredicted observed accuracy overall and missed the steeper decrease for serial recall. Correlation and *rmsd* were calculated for each participant. Across participants, the average correlation was .9238 (SD = .0883), reflecting the overall similarity in the pattern, and

Table 5
Experiment 1 Decrease Models: Model Comparisons Assessed by Differences in BIC Scores, Number of Participants Showing the Difference, and t Tests of the Difference in Experiment 1

Measure	B – ED	B – SOD	B – ED+SOD	SOD – ED	SOD – ED+SOD	ED – ED+SOD
BIC difference	622.73	346.59	969.47	276.13	622.88	346.75
N different	24	24	24	21	24	24
t(23)	9.749	17.064	12.558	5.020	9.751	17.086

Note. Base = baseline model; BIC = Bayesian information criterion; ED = encoding decrease model; SOD = serial order decrease model; ED+SOD = encoding decrease + serial order decrease model.

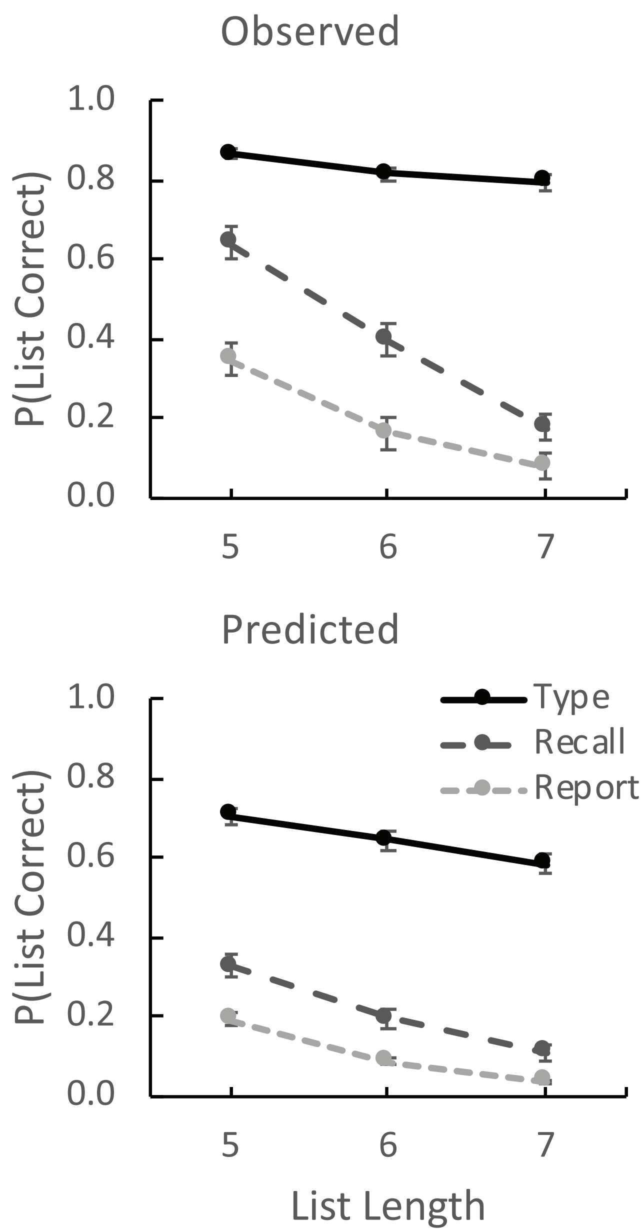


Figure 6. List accuracy in Experiment 1: Observed and predicted probabilities of reporting the entire list correctly as a function of list length and task. Predictions are from the ED+SOD model. Error bars are standard errors of the mean.

the average *rmsd* was .2104 (*SD* = .0669), reflecting CRU's underestimation of the observed data.

Serial position effects. The practice of plotting the probability of reporting an item correctly in its correct position—assessing the *serial position effect*—is common in studies of memory and perceptual report, dating to Nipher (1878) and Ebbinghaus (1885). The serial position effect is robust and highly replicable. It is a benchmark prediction for theories of memory in general and serial recall in particular (Lewandowsky & Farrell, 2008).

The observed and predicted serial position effects for each task and list length are plotted in Figure 7. The observed effects are typical. Accuracy decreased across serial position with some sparing of the last items. Accuracy was the same across tasks for the initial position but decreased across serial position at different rates. Typing hardly decreased while serial recall and whole report decreased more steeply. Serial recall and whole report showed recency effects for the last item. The predicted curves show similar declines from a common starting point without the recency effect. This shape follows from the parameterization of the ED+SOD model. It has one g_{Max} and β_{Max} parameter for all three tasks, which produce the common starting point, but it has different $g_{Decrease}$ and $\beta_{Decrease}$ parameters for each task, which produce the differential declines. Recency could be accommodated by allowing g or β to be larger for the last serial position, reflecting better perceptibility (g) or a sharper focus on the present in memory encoding (β). Agreement between observed and predicted values was assessed with r and *rmsd* for each participant. The average correlation across participants was .9195 (.0246) and the average *rmsd* was .1506 (.0245).

Transposition gradients. Transposition gradients show the probability that an item is recalled in its correct position or an

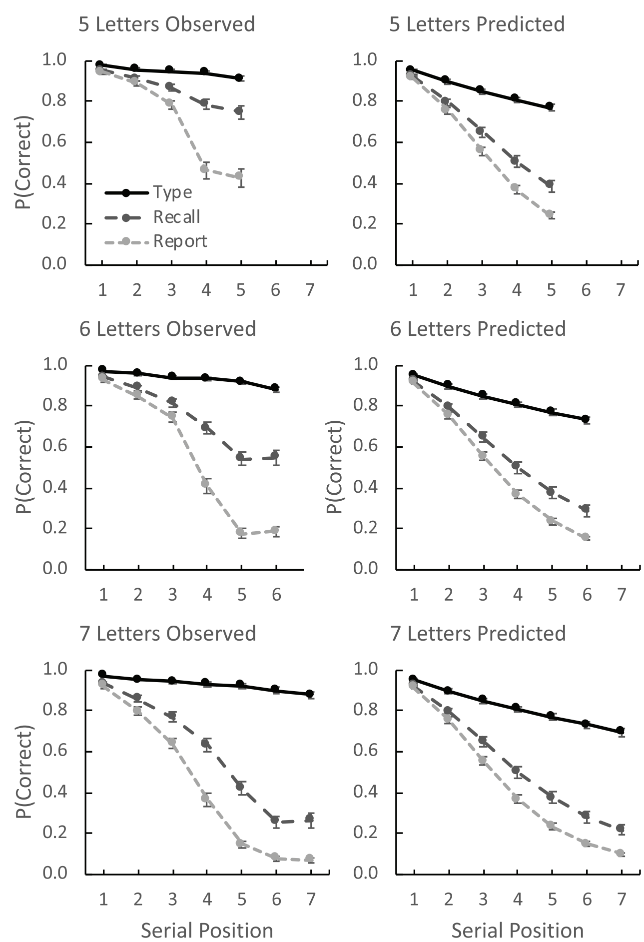


Figure 7. Serial position effects in Experiment 1: Observed and predicted probabilities of responding with the correct item in the correct position for typing, serial recall, and whole report tasks at each list length. Predictions are from the encoding decrease + serial order decrease (ED+SOD) model. Error bars are standard errors of the mean.

adjacent position earlier (−) or later (+) in the list, given that the item is recalled. Positive and negative shifts are *transpositions* and the data typically show a *gradient* decreasing symmetrically from the correct position (Lewandowsky & Farrell, 2008). The symmetry of transposition gradients is important theoretically. Strict chaining models predict only forward movements (+) through the list (Henson, Norris, Page, & Baddeley, 1996).

Transposition gradients were calculated for each task and list length for each participant, collapsed across serial position. The means of the observed and predicted gradients from the individual participants are plotted in Figure 8. The observed gradients were typical, being higher for adjacent than remote positions. They broadened with list length and task (typing, serial recall, whole report), reflecting the increase in error rate and showing it is largely accommodated by an increase in adjacent errors. The observed functions showed a negative asymmetry in which items are more likely to be transposed to earlier list positions. The asymmetry may result from the increase in order errors with serial position. Items from later positions can only move back toward the beginning of the list, producing a negative transposition.

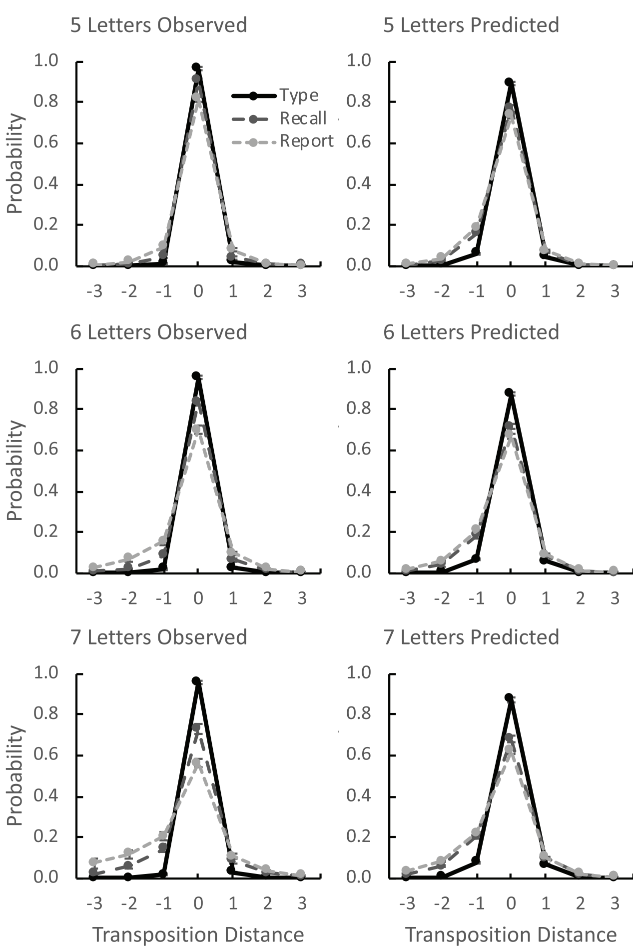


Figure 8. Transposition gradients in Experiment 1: Observed and predicted probabilities of reporting items in the correct position (0), positions before (negative values), and positions after (positive values). Predictions are from the encoding decrease + serial order decrease (ED+SOD) model. Error bars are standard errors of the mean.

The predicted gradients in Figure 8 closely resemble the observed gradients, showing the same preponderance of adjacent errors and the same broadening of the gradient with list length and task (typing, serial recall, whole report). They show the same negative asymmetry, reinforcing the idea that it reflects more order errors in late list positions. The correlation and *rmsd* between observed and predicted values were calculated for each participant. The average correlation across participants was .9744 (.0143) and the average *rmsd* was .0707 (.0170), reflecting the close fit.

Contiguity effects. Contiguity effects measure the tendency to move forward through the list, focusing on the *lag* or distance in the list between adjacent items in the report sequence (Healey, Long, & Kahana, 2019). The *lag conditional recall probability* (lag CRP) curve plots the probability of recalling item from position N + lag given that the item in position N has been recalled (conditional on opportunities for the lag to occur at position N; Kahana, 1996). It is usually asymmetric and peaked at lag = +1, reflecting the tendency to report lists in order.

Lag CRP curves were calculated for each task and list length for each participant's real and simulated data. The averages across participants are presented in Figure 9. The observed data showed a strong asymmetry with a large peak at lag = +1, reflecting compliance with the instruction to report the items in order. The peaks were lower and the curves were broader in serial recall than in typing and lower and broader in whole report than in serial recall, suggesting differential loss of order information across tasks. The predicted lag CRP curves were very similar to the observed ones for each participant, showing the same sharp peak at lag = + 1 and the reduction in the peak and broadening of the curve from typing to serial recall to whole report. The agreement between observed and predicted values was excellent. Correlation and *rmsd* were calculated for each participant. The averages across participants were r = .9889 (.0060) and *rmsd* = .0409 (.0072).

Error magnitude. Error magnitude is a measure of the number of errors in an erroneously reported string. It is useful because it measures the tendency for people (and models) to recover from errors. For example, skilled typists typing paragraphs typically make one error and recover from it immediately (and so does CRU; Logan, 2018). Following Logan (2018), I measured error magnitude by calculating *edit distance* between the correct list and the reported list for all reported lists that contained at least one error. Edit distance is the number of editing operations (changes) required to transform the error list into the correct list. Analysis of serial position effects relies on *Hamming distance*, which allows substitution as the only editing operation (Hamming, 1950). It does not capture the similarities between strings with insertions and deletions (e.g., correct list = *abcdef*; error lists = *abxcdef* or *abdef*), counting all responses after the initial error as substitution errors. Instead, I used *Damerau distance* to capture these errors. It allows substitution, insertion, deletion, and transposition as editing steps (Damerau, 1964), counting one error in *abxcdef* and one error in *abdef* given list *abcdef*.

I calculated Damerau distance for each participant for each list that contained at least one error using a Matlab algorithm provided by Schauerte and Fink (2010) and calculated the distribution of distances for each task and list length. I performed the same calculations on the participant's data and the CRU simulation of the participant. The observed and predicted distributions averaged over participants are presented in Figure 10.

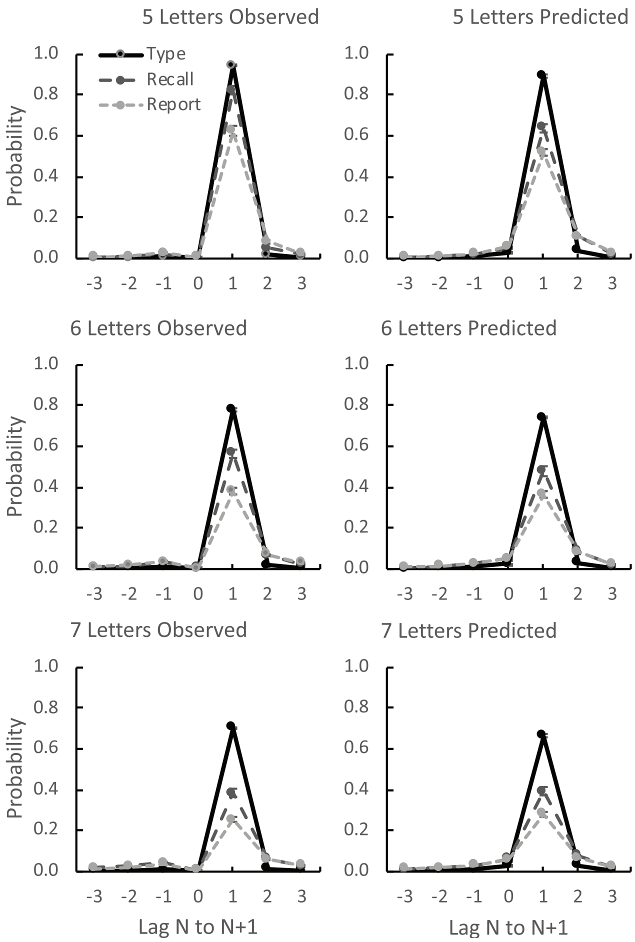


Figure 9. Contiguity effects in Experiment 1: Observed and predicted lag conditional recall probabilities as a function of task and list length. Predictions are from the encoding decrease + serial order decrease (ED+SOD) model. Error bars are standard errors of the mean.

The Damerau distance distributions differed between tasks and word lengths. The typing distributions replicated previous results with typing text (Logan, 2018). About 80% of the errors were one editing step from the correct list, indicating that typists made a single error and recovered from it. The distribution became a little broader as list length increased. The serial recall distributions had a broader version of the same shape for list lengths of five and six, but the mode shifted for list length of seven. The whole report distributions were even broader and peaked at two, three, and four for list lengths five, six, and seven, respectively. The predicted distributions captured these shapes well. The average correlation across participants was .9379 (.0602) and the average *rmsd* across participants was .0758 (.0294), indicating an excellent fit.

Error type. Much research on serial order in action has focused on error categories using various taxonomies (Dell, 1986; Logan, 2018; Norman, 1981; Salthouse, 1986). Lists containing multiple errors are harder to categorize, as the same error may fit different categories. Consequently, I focused on error categories that could be distinguished unambiguously. I defined *order errors* as reports of items that were in the list in a position other than the correct position. *Intrusion errors* were reports of items that were not presented in the list. *Omission errors* were failures to report items that were on the list. This taxonomy does not distinguish between intrusions that are insertions (*abxcdef* for *abcdef*) and intrusions that are substitutions (*abxdef* for *abcdef*). Substitutions were scored as an intrusion and as an omission. CRU produces order errors primarily in the retrieval stage, intrusion errors in the encoding stage, and omission errors either in the encoding stage or in the retrieval stage.

I calculated the proportion of trials on which order, intrusion, and omission errors occurred in each serial position relative to the total number of trials for each task and list length for each participant, analyzing both their actual and simulated data. The observed and predicted proportions of errors averaged across participants are presented in Figure 11. The observed proportions were lower for typing than for serial recall and lower for serial recall than for whole report. In all tasks, the observed proportions increased with list length and increased with serial position. For typing, order errors were always more likely than omission errors for all serial positions. For serial recall and whole report, order errors were

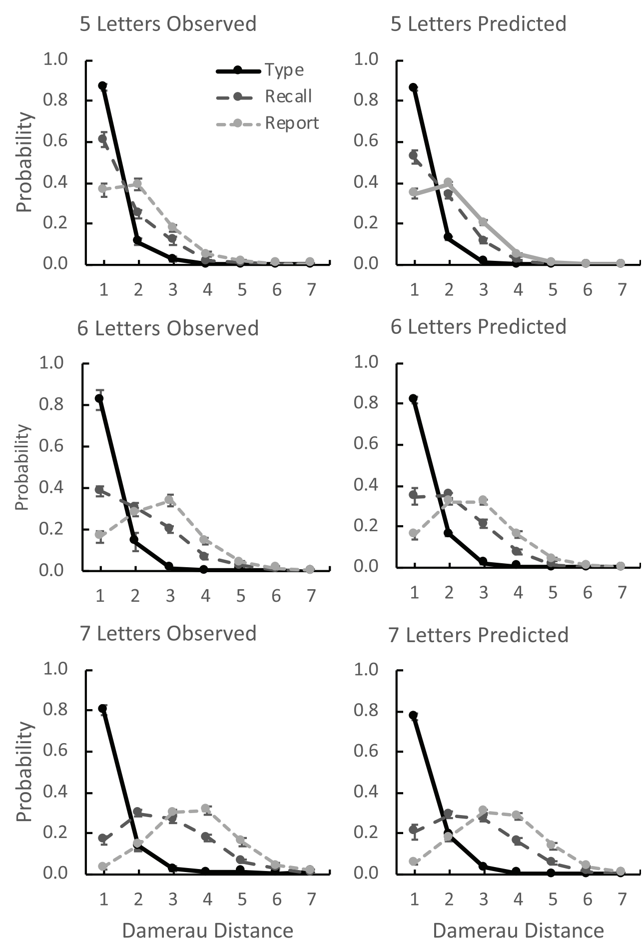


Figure 10. Error magnitudes in Experiment 1: Observed and predicted distributions of Damerau edit distance between correct and error strings for each task and list length. Predictions are from the encoding decrease + serial order decrease (ED+SOD) model. Error bars are standard errors of the mean.

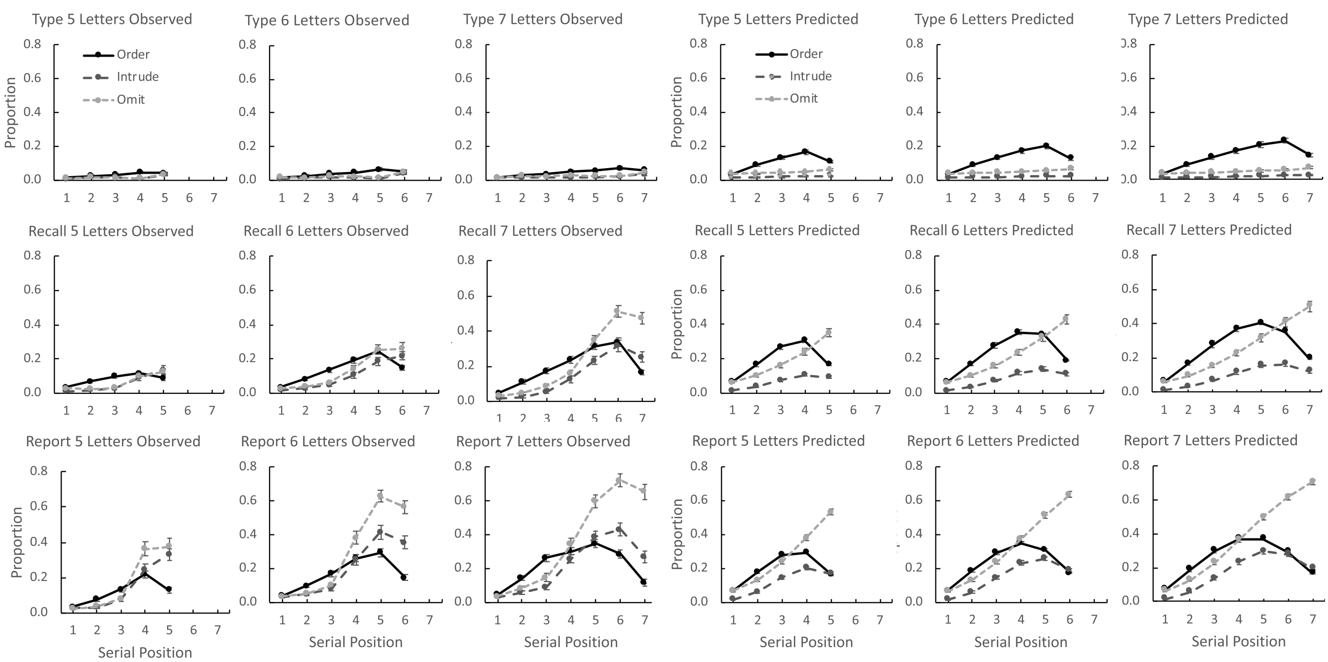


Figure 11. Error types in Experiment 1: Proportions of observed and predicted order, intrusion, and omission errors as a function of task, list length, and serial position. Predictions are from the encoding decrease + serial order decrease (ED+SOD) model. Error bars are standard errors of the mean.

more likely than omission errors at the earlier serial positions but the pattern reversed at later serial positions, where omissions were more likely than order errors.

The predicted proportions show a broadly similar pattern, with error proportions increasing from typing to serial recall to whole report, increasing with list length, and increasing with serial position. There were more order errors than omissions in typing. In serial recall and whole report, order errors were more frequent than omission errors at the early serial positions, but the pattern reversed at later serial positions. There were some systematic differences. The model underestimated the proportion of intrusion errors, especially at longer list lengths, and it did not show the reduction in omission error proportion for the last item on the list. The mean correlation across participants was .8081 (.0563) and the mean *rmsd* across participants was .1043 (.0222). Note that the model captured the ratio between order and omission errors and how it changed with serial position (see Figure 11).

Error ratio. Page and Norris (1998) proposed the *error ratio* as a critical test of serial order theories. It is the ratio of a subset of order errors to a subset of omission errors. It is calculated on trials in which participants skip one item in the sequence (e.g., recalling *abd* . . . given list *abcdef*), counting the frequencies with which the subsequent response is a transposition (*abdc* . . .) or *fill-in* error versus an omission (*abdef* . . .) or *in-fill* error, and taking the ratio of transpositions to omissions. Error ratios in serial recall tasks are typically greater than 1.0, reflecting a propensity to make transposition errors—to "fill in" the empty space. They are often described as "around 2.0" but there is great variability across participants (Farrell, Hurlstone, & Lewandowsky, 2013; Surprenant, Kelley, Farley, & Neath, 2005).

Page and Norris (1998) argued that the observed error ratios falsified simple chaining theories, because chaining theories can only move forward in the list. Having recalled *abd*, the only response associated with *d* is *e*, so the error will become an omission. By contrast, the error ratio uniquely supported their primacy model, in which serial order is controlled by a gradient of activation across the list, decreasing from the first position to the last. The item with the highest activation is retrieved and then suppressed so the item with the next highest activation can be retrieved. Normally, the item with the next highest activation is the next item in the list. However, when participants report *abd* . . . for *abcdef*, *d* is suppressed, and *c* has higher activation than *e* (because it occurs earlier in the list), so *c* is selected and the error will become a transposition.

Theories of serial recall developed after Page and Norris (1998) generally predict error ratios greater than 1.0, but the predictions generally rely on some ancillary mechanisms beyond the core serial order mechanism to produce the effect (Botvinick & Plaut, 2006; Farrell & Lewandowsky, 2002; Henson, 1998b). Evolving context models generally predict an equal tendency to move forward or backward in the list as the similarity function decreases symmetrically around the correct position (Equation 6; Howard & Kahana, 2002; Murdock, 1997), so something like a primacy gradient has to be added to break the symmetry in favor of earlier list positions. Thus, the error ratio may not be as diagnostic theoretically as Page and Norris suggested.

Observed and predicted proportions of transpositions and omissions were calculated by finding the first error in a list and determining whether it was a skipped letter (an omission followed by the item following the omitted item; given *abcded*, *abd* would count but *abe* or *abx* would not). The proportions of first error trials that resulted in transposition and omission errors were calculated for each participant and each simulation. The mean observed and predicted probabilities for each task are plotted as a function of list length in Figure 12.

The observed data were typical of the serial recall literature and extend the findings to whole report and copy typing. Averaged across tasks and list length, all 24 participants showed more transpositions than omissions. The predicted data showed the opposite pattern. Omissions were more frequent than transpositions for all 24 participants. The average error ratio was 3.8012 (2.7197) for the observed data and .5776 (.0762) in the predicted data. I calculated the correlation and *rmsd* between observed and predicted error ratios in 3 × 3 matrices defined by task and list length for each participant. The mean correlation was .3355 (.4500) and the mean *rmsd* was 2.4518 (1.1962), reflecting a poor fit.

I was surprised by the asymmetry in favor of omissions because I thought the similarity gradient was symmetrical (Equation 6), so I examined it more closely. The dot products after a correct response and an initial omission error are plotted in Figure 13 for the third position in the string. The plot for correct responses is symmetrical, following Equation 6, but the plot for initial omissions is asymmetrical, favoring later responses. The asymmetry can be understood by considering the set of context vectors in Table 1, which show the correct encoding of *abcdef*. A participant who recalled *abd* . . . would have the following current context vector after updating:

$$4' = [\beta \cdot \rho^2 \; \beta \cdot \rho \; 0 \; \beta \; 0 \; 0 \; \rho^3]$$

The dot products between 4′ and stored contexts 3, 4, and 5 are:

$$\mathrm{dot}(4', 3) = \beta^2 \cdot \rho^3 + \beta^2 \cdot \rho + \rho^5$$
$$\mathrm{dot}(4', 4) = \beta^2 \cdot \rho^4 + \beta^2 \cdot \rho^2 + \rho^6$$
$$\mathrm{dot}(4', 5) = \beta^2 \cdot \rho^5 + \beta^2 \cdot \rho^3 + \beta^2 + \rho^7$$

The dot product for position 4 is smaller than the dot product for position 3 without any response suppression:

$$\mathrm{dot}(4', 4) = \rho \cdot \mathrm{dot}(4', 3)$$

The dot product for position 5 is larger than the dot product for position 3, producing the asymmetry:

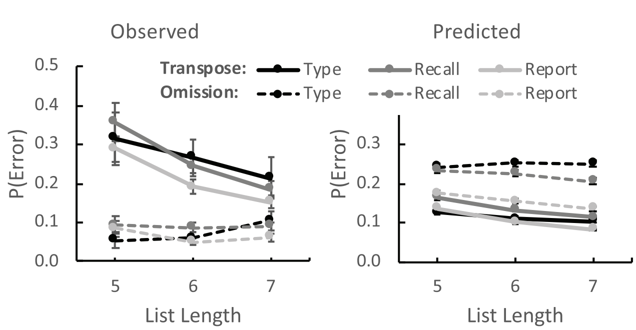


Figure 12. Error ratio components in Experiment 1: Proportions of observed and predicted transposition and omission errors following an initial skipped letter as a function of task and word length. The error ratio (fill-in/in-fill) is the ratio of transpositions to omissions. Predictions are from the encoding decrease + serial order decrease (ED+SOD) model. Error bars are standard errors of the mean.

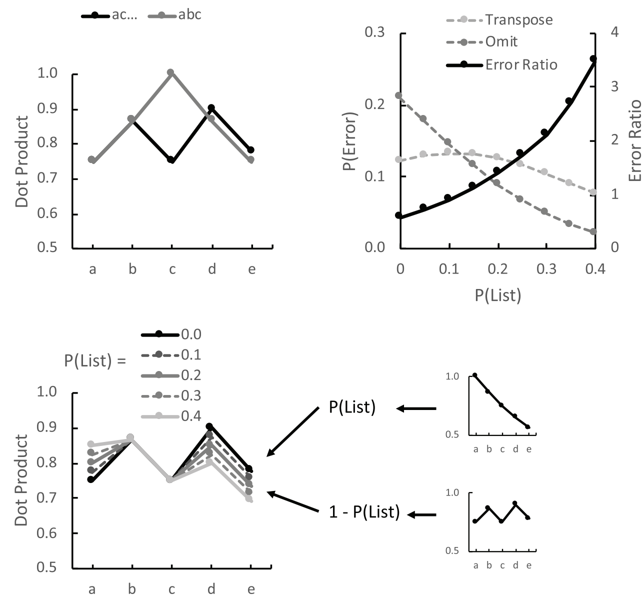


Figure 13. Dot products and error ratios. Top left: Dot products for correct responses (abc in gray) and skipped letter responses (ac . . . in black) showing symmetry for correct responses and asymmetry for errors. Bottom right: The components of the mixture model include the initial context and the current context, which are weighted by P(List) and 1 − P(List), respectively, to produce a retrieval cue. Bottom left: Dot products for skipped letter errors with different values of P(List).

$$\begin{aligned}\mathrm{dot}(4', 5) &= \rho^2 \cdot \mathrm{dot}(4', 3) + \beta^2\\ &= (1 - \beta^2) \cdot \mathrm{dot}(4', 3) + \beta^2\\ &= \mathrm{dot}(4', 3) + \beta^2 - \beta^2 \cdot \mathrm{dot}(4', 3)\end{aligned}$$

Thus, CRU predicts an asymmetry in favor of later positions following an initial omission. For CRU to predict the observed error ratio, this asymmetry must be reversed with something like a primacy gradient.

I explored one way to introduce a primacy gradient in CRU by assuming that the effective retrieval cue is a proportional mixture of the initial list cue (which produces a primacy gradient) and the current context. The bottom panel of Figure 13 shows how the asymmetry in the dot products is affected by varying the contribution of the list cue versus the current context (by varying P(List)). The top right panel shows how the probability of transpositions and omissions and the error ratio change as P(List) is varied in simulations of CRU. The error ratio is 1.0 when P(List) = .1210 and 2.0 when P(List) = .2839. Thus, in principle, CRU could account for the observed error ratios.

To determine whether this modification of CRU could account for the observed error ratios, I fit a version of the ED+SOD model in which there was a separate P(List) parameter for each task. Because P(List) affected retrieval and not encoding, and because encoding and retrieval parameters were the same whether or not the other varied, I fixed the encoding parameters to the best-fitting values from the ED+SOD fits. Thus, there were seven free parameters and four fixed parameters for a total of 11 parameters. Across participants, the mean likelihood (3727.88 ± 1224.16) was not much different from the mean likelihood from the ED+SOD fits (3728.46 ± 1224.32) but BIC was larger for the modified model in all 24 participants (mean BIC = 7494.67 ± 2448.31, $t[23] = 17.3333$), indicating that the modification did not improve the fit overall.

The mean β_{Max}, $\beta_{DecType}$, $\beta_{DecRecall}$, and $\beta_{DecReport}$ parameters across participants were .5760, .9984, .9252, and .9058, respectively, close to the values from the ED+SOD fits in Table 4. By contrast, the mean P(List) parameters were .0003, .0085, and .0069 for copy typing, serial recall, and whole report, respectively. All values were close to 0, at which point the model is equivalent to the ED+SOD model. The near-zero values were well below the values required for an error ratio above 1.0 (see Figure 13). Although in principle, adding the list cue allows CRU to predict the observed error ratios, in practice, fitting CRU to the entire data set did not favor this addition. Thus, the present version of CRU does not account for the error ratio. Interestingly, the model did a good job of accounting for the relative proportions of transpositions (order errors) and omissions when the errors were defined more inclusively to include all such errors, not just the ones that follow skipped-letter errors (see Error Types above).

Discussion

The empirical goal of the experiment was to compare serial order phenomena in perception, memory, and action to determine whether they result from the same mechanism acting under different constraints. The results were similar across tasks, differing quantitatively but not qualitatively (Figures 6–12), which is consistent with a single mechanism account. The theoretical goal was to ask whether a computational implementation of a single mechanism (CRU) could account for the quantitative differences between tasks in terms of meaningful variations in its parameters. CRU's ED+SOD model predictions agreed well with the observed data in six different summary measures (list accuracy, serial position effects, transposition gradients, contiguity effects, error magnitudes, and error types). CRU mispredicted the error ratio. Like other chaining models, it could produce error ratios greater than 1.0 but the parameters values it required did not fit the rest of the data (also see Solway, Murdock, & Kahana, 2012).

The model fits compared variants of CRU in which encoding (g) and serial retrieval (β) parameters were fixed or allowed to vary between tasks. The model comparisons indicated that better fits were obtained when g or β were allowed to vary either individually or jointly. The ED+SOD model, which fit best for all 24 participants, allowed g and β to decrease at differential rates across serial position for different tasks. There was almost no decrease in the typing task, consistent with previous CRU fits to skilled typing (Logan, 2018), a stronger decrease in serial recall, and the strongest decrease in whole report.

The theoretically important manipulation across tasks was list length. Participants generally did worse with longer lists and CRU largely accounted for these effects without having to add or adjust parameters. The same parameters were used regardless of list length. In the decrease models g and β reach lower values at the end of longer lists but only because same decrease propagates to more items ($g^{5-1} > g^{6-1} > g^{7-1}$). Thus, CRU predicts list length effects without any special assumptions. Longer lists provide more opportunity for decrease and more opportunity for errors. Those factors seem sufficient to explain list length effects without invoking assumptions about capacity limitations or limited numbers of slots.

Experiment 2: Within-List Repetitions in Whole Report, Serial Recall, and Copy Typing

The second experiment compared serial order effects in perception, memory, and action using six-letter lists and manipulating within-list repetition (*abcbef*). Within-list repetitions challenge serial order models based on chaining, activation gradients, and successor inhibition (Lashley, 1951). In strict *chaining models* in which each item is associated only with its successor, within-list repetitions create loops in the chain: given *abcbef*, *b* is associated with both *c* and *e* so the response following the repeated item is likely to be an error. Within-list repetitions create problems for models that represent order as a *gradient of activation* across items (Page & Norris, 1998) because repeated items disrupt the gradient: given *abcbef*, *b* is likely to have the highest activation and so is likely to be chosen first. Within-list repetitions also create problems for *successor inhibition* models in which each item inhibits every item that follows it, creating a gradient of activation that declines over serial position (Bryden, 1967; Estes, 1972; Rumelhart & Norman, 1982). Within-list repetitions have more activation and so disrupt the pattern of inhibition across the list. CRU can handle within-list repetitions without crashing (Logan, 2018). It remains to be seen whether it accounts for more subtle effects of within-list repetition.

Within-list repetitions challenge theories of memory and perception. Ranschburg (1902) showed that the second of two repeated items within a list is recalled less accurately than control items in the same serial positions if the lag between repetitions is greater than 0. Lag 0 repetitions are usually recalled better than control items. This *Ranschburg effect* has been replicated many times (Henson, 1998a; Jahnke, 1969). The impairment for the second item at lags > 0 is often attributed to response suppression, which makes a previously retrieved item less available. The benefit for lag 0 is often attributed to noticing immediate repetitions and tagging them as repetitions (Henson, 1998a). CRU handles within-list repetitions without either of these accounts. It remains to be seen how well it accounts for the Ranschburg effect and its modulation across tasks.

Kanwisher (1987) showed a deficit in reporting the second of two repeated items within a stream of rapid serial visual presentations, which has also replicated many times in different formats, including simultaneous presentations (also see Bjork & Murray, 1977; Santee & Egeth, 1980). She attributed this *repetition blindness* to perceptual processes involved in individuating types and tokens and found considerable support for that hypothesis, although there is some evidence for a retrieval effect (Fagot & Pashler, 1995). CRU handles repetition blindness and the Ranschburg effect in the same way, as described below. It remains to be seen whether CRU can account for within-list repetition effects in the three tasks without adding assumptions about perceptual interference.

CRU represents all letters, repeated or unique, with a localist code, setting the element in the input vector that represents the letter to 1 and setting all other elements to 0. When a letter is repeated, the same element is set to 1. The input vector is added to

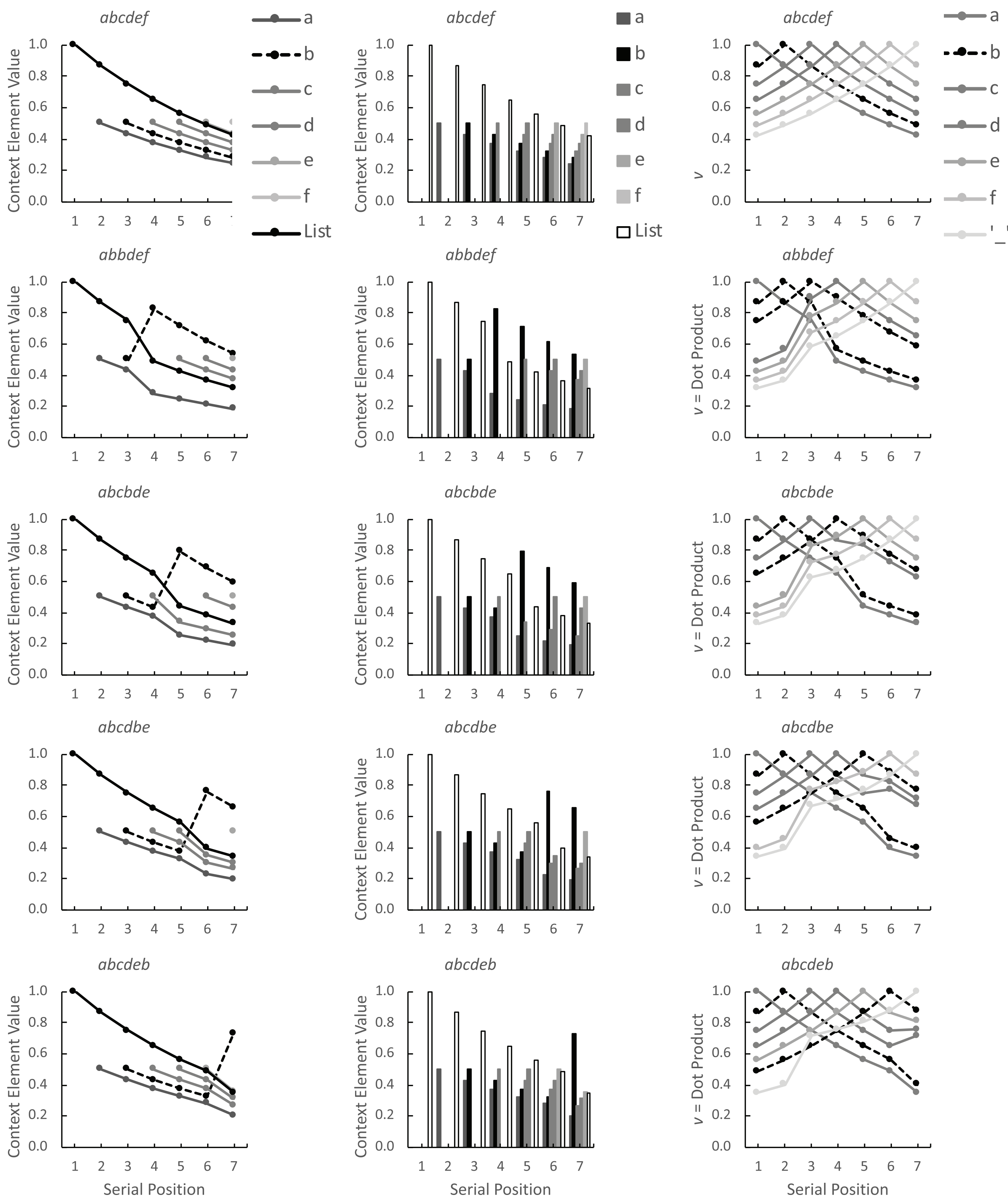


Figure 14. Context Retrieval and Updating model (CRU) context element values and dot products as a function of serial position for unique strings (top) and strings with repeated letters with lag 0–3 between repetitions (below). The repeated letter is represented by a dashed line in the left and right columns.

the current context using updating equations 3 and 4, reproduced here for convenience:

$$\rho = \sqrt{1 + \beta^2\left[(\boldsymbol{r}_N \cdot \boldsymbol{c}_N)^2 - 1\right]} - \beta(\boldsymbol{r}_N \cdot \boldsymbol{c}_N)$$

$$\rho = \sqrt{1 - \beta^2}$$

If the input letter is unique, there is no overlap between the input vector and the current context vector, so Equation 4 applies. If the input letter was presented earlier in the list, CRU sets the same element in the input vector to 1, and so the input vector overlaps with the current context vector. Equation 3 compensates for the overlap by reducing ρ in proportion to the dot product between the input vector and the context vector, normalizing the length of the updated context vector to 1.0 (Howard & Kahana, 2002). Figure 14 illustrates the effects of repeating letters at different lags on the representations of elements in the context vectors (left two panels) and on the dot products between stored context vectors (right panel). Repetition dramatically increases the element value for the repeated item, and that affects the other elements through the normalization process (Equation 3). Repetition changes the similarity structure: Items following the repetition compete less with items preceding the repetition and vice versa. Logan (2018) showed that CRU can retrieve sequences accurately despite the competition. It remains to be seen whether the nature of the competition can account for Ranschburg effects in whole report, serial recall, and copy typing tasks.

Experiment 2 asked the same empirical and theoretical questions as Experiment 1: Are the serial order phenomena in whole report, serial recall, and copy typing manifestations of a single serial order mechanism operating under different constraints? Experiment 2 answered the questions in the same way: Versions of CRU were fit to the data, the best-fitting model was selected with BIC, and the best-fitting parameters were used to drive simulations of the model to produce predictions for summary statistics (list accuracy, serial position effects, transposition gradients, contiguity effects, error magnitudes, error types, error ratios, and Ranschburg effects).

Method

Participants. Twenty-four participants were recruited for their self-professed ability to type 40 WPM. None had served in Experiment 1. Eight identified as male, 15 as female, and one as "gender." Their mean speed on the typing test was 76.74 WPM (21.03). Their mean accuracy was 93.66% (2.77).

Apparatus and stimuli. The apparatus, computer displays, and keyboards were the same as in Experiment 1. All lists were six letters in length. They were selected randomly from all 26 letters of the alphabet. There were 576 lists, 192 for each task. In each task, 96 lists contained no repetitions (*unique lists*) and 96 contained a single repeated item, selected at random. The number of letters intervening between repeated items (the *lag*) was 0, 1, 2, or 3, with 24 lists at each lag. Repeated items appeared in all possible positions in the list with equal frequency for lags 1, 2, and 3. For lag 0, four positions were tested five times and one (randomly selected) position was tested four times.

Procedure. Lists with unique and repeated items were mixed randomly. Otherwise, the procedure was the same as in Experiment 1.

Results

Model fits.

Nondecrease models. I fit the same four nondecrease models to the sequences of about 3456 responses generated by each participant individually, allowing g or β or both to vary between tasks. Measures of goodness of fit and best-fitting parameters are presented in Table 6 and model comparisons are presented in Table 7. For all 24 participants, fits were better than baseline when either g or β were allowed to vary between tasks and best when both were allowed to vary. The best fits were obtained with the E+SO model. Again, the estimates of β were the same whether or not g varied between tasks, and the estimates of g were the same whether or not β varied between tasks. Both g and β varied meaningfully between tasks, decreasing from typing to serial recall to whole report, reflecting the effect of limiting exposure duration.

Decrease models. I fit the same three decrease models to each participant's data, allowing $g_{Decrease}$ or $\beta_{Decrease}$ or both to vary between tasks, and comparing them to the nondecrease baseline model as before. Measures of goodness of fit and best-fitting parameters are presented in Table 8 and model comparisons are presented in Table 9. The parameter values and measures of goodness of fit for each participant for the best-fitting ED+SOD model appear in Table A2.

For all 24 participants, fits were better than baseline when either $g_{Decrease}$ or $\beta_{Decrease}$ was allowed to vary between tasks, and best when both were allowed to vary. The best fits were obtained with the ED+SOD model. Again, estimates of $\beta_{Decrease}$ were the same whether or not $g_{Decrease}$ varied between tasks, and estimates of $g_{Decrease}$ were the same whether or not $\beta_{Decrease}$ varied between

Table 6
Experiment 2 Nondecrease Models: Measures of Goodness of Fit and Best-Fitting Parameter Values for Baseline (B), Encoding (E), Serial Order (SO), and Encoding + Serial Order (E+SO) Models in Experiment 2

Measure	Likelihood	BIC	g_{Type}	g_{Recall}	g_{Report}	β_{Type}	β_{Recall}	β_{Report}
Base	3334.05	6675.17		.2724			.4959	
Encoding	3240.77	6495.69	.3435	.2862	.2372		.4961	
Serial order	3238.56	6491.28		.2724		.5861	.4934	.4500
E+SO	3145.41	6312.04	.3433	.2865	.2402	.5866	.4934	.4500

Note. BIC = Bayesian information criterion.

Table 7
Experiment 2 Nondecrease Models: Model Comparisons Assessed by Differences in BIC Scores, Number of Participants Showing the Difference, and t Tests of the Difference in Experiment 2

Measure	Base – E	Base – SO	Base – E+SO	E – SO	SO – E+SO	E – E+SO
BIC difference	179.48	183.89	188.64	4.41	179.24	95.36
N different	24	24	24	13	24	24
t(23)	5.669	7.787	4.060	0.142	5.655	4.034

Note. Base = baseline model; BIC = Bayesian information criterion; E = encoding model; SO = serial order model; E+SO = encoding + serial order model.

tasks. As before, both $g_{Decrease}$ and $\beta_{Decrease}$ decreased across tasks from typing to serial recall to whole report, accounting for differences between tasks with meaningful variation in parameters.

As in Experiment 1, the decrease models fit better than the nondecrease models. ED fit better than E in 24 participants (BIC difference = 253.38, t[23] = 9.04). SOD only fit better than SO in 15 participants (BIC difference = 10.87, t[23] = 1.17). ED+SOD fit better than E+SO in 22 participants (BIC difference = 216.03, t[23] = 7.64). Thus, decrease of g and β over serial position seems to be necessary to produce good fits.

Model predictions. As before, model predictions were generated for each participant by running a simulation of CRU on the same lists the participant experienced using the participant's best-fitting parameters from the best-fitting ED+SOD model.

List accuracy. The mean observed and predicted probabilities of reporting the entire list correctly are plotted as a function of task and lag between repetitions in Figure 15. As before, observed list accuracy decreased from copy typing to serial recall to whole report. For all three tasks, observed list accuracy was highest with lag 0 between repetitions (i.e., with doubled letters). Observed list accuracy was lower for lags 1–3 than for unique lists for whole report but not much different for serial recall and copy typing. Predicted list accuracy underestimated observed accuracy. It captured the differences between tasks but missed the effects of lag. The mean correlation between observed and predicted values across the 24 participants was .8891 (.1498) and the mean *rmsd* was .2542 (.0525), reflecting the poor fit.

Serial position effects. The mean observed and predicted serial position effects across participants are plotted for lists containing unique and repeated items in Figure 16. The observed serial position effects were much like the observed effects with list length 6 in Experiment 1. Accuracy was high for the first serial position in all three tasks and then declined at different rates. The decline was least for copy typing, intermediate for serial recall, and most for whole report. There was a recency effect for the last item in serial recall and whole report but not for copy typing. The predicted serial position curves captured the fanning out from a common beginning at the first serial position, but underestimated accuracy overall and did not capture the trend toward recency in the serial recall and whole report data. A larger g or β or both for the last position may accommodate this effect. Across participants, the mean correlation between observed and predicted values was .8996 (.0502) and the mean *rmsd* was 0.0811 (.0389), reflecting a decent fit.

Transposition gradients. The mean observed and predicted transposition gradients across participants are plotted for unique and repeated lists in Figure 17. The observed gradients for unique lists show the same peak at 0 that declines from copy typing to serial recall to whole report that was observed in Experiment 1. The gradients are negatively asymmetric, reflecting the tendency for errors to occur near the end of the list, so transpositions must come from earlier positions. The observed gradients for repeated lists are similar except for a tendency to recall items from positions −2 and −3. The predicted transposition matrices show very similar effects. The mean correlation between observed and predicted values was .9839 (.0091) and the mean *rmsd* was .0617 (.0147), reflecting a good fit.

Contiguity effects. The mean observed and predicted lag CRPs across participants for unique and repeated lists in each task are plotted in Figure 18. The lag CRPs for unique lists replicate Experiment 1. They peaked at lag + 1, reflecting a strong tendency to move forward through the list, and the peak decreased from copy typing to serial recall to whole report. The lag CRPs for lists with repeated items were similar, though the peaks were lower and the lower tail was thicker. The predicted lag CRPs captured these effects well. The mean correlation between observed and predicted values across participants was .9950 (.0039) and the mean *rmsd* was .0324 (.0086), reflecting an excellent fit.

Error magnitude. The mean observed and predicted distributions of Damerau distance across participants for unique and

Table 8
Experiment 2 Decrease Models: Measures of Goodness of Fit and Best-Fitting Parameter Values for Baseline (B), Encoding Decrease (ED), Serial Order Decrease (SOD), and Encoding Decrease + Serial Order Decrease (ED+SOD) Models in Experiment 2

Measure	Likelihood	BIC	g_{Max}	g_{DType}	$g_{DRecall}$	$g_{DReport}$	β_{Max}	β_{DType}	$\beta_{DRecall}$	$\beta_{DReport}$
Baseline	3334.05	6675.17	.2724		1.0000		.4959		1.0000	
ED	3112.31	6242.32	.3609	.9808	.9244	.8647	.4981		1.0000	
SOD	3231.36	6480.41	.2725		1.0000		.5582	.9995	.9647	.9350
ED+SOD	3033.85	6096.01	.3609	.9813	.9244	.8648	.5586	.9995	.9645	.9348

Note. BIC = Bayesian information criterion. Parameter values of 1.0000 were fixed, not fitted.

Table 9
Experiment 2 Decrease Models: Model Comparisons Assessed by Differences in BIC Scores, Number of Participants Showing the Difference, and t Tests of the Difference in Experiment 2

Measure	B − ED	B − SOD	B − ED+SOD	SOD − ED	SOD − ED+SOD	ED − ED+SOD
BIC difference	432.85	194.76	579.16	238.10	384.40	146.30
N different	24	24	24	21	24	24
t(23)	7.535	8.586	8.548	6.521	6.521	6.520

Note. Base = baseline model; BIC = Bayesian information criterion; ED = encoding decrease model; SOD = serial order decrease model; ED+SOD = encoding decrease + serial order decrease model.

repeated lists in each task are plotted in Figure 19. The distributions are very similar to the distributions from six-letter lists in Experiment 1 (see Figure 10). Copy typing peaked sharply at distance = 1, reflecting a strong tendency to recover from errors (Logan, 2018). Serial recall also peaked at distance = 1 but declined much more gradually. Whole report peaked at distance = 1 and distance = 2 and declined even more gradually, reflecting the preponderance of errors. There was little difference between the observed functions for unique and repeated lists and little difference between predicted and observed distributions. The mean correlation across participants was .9562 (.0392) and the mean *rmsd* was .0614 (.0175), reflecting a good fit.

Error type. The mean predicted and observed proportions of omission, order, and intrusion errors across participants for unique and repeated lists in each task are plotted as a function of serial position in Figure 20. The observed functions for unique lists show an increase in all error types with serial position with reduced growth for the last position. All error types increased with task, from copy typing to serial recall to whole report. The observed functions for repeated lists were similar except that omission errors were less frequent in repeated lists. Observed and predicted functions both showed more order errors than omission errors at early serial positions and a reversal at the later serial positions. The average correlation between observed and predicted values was

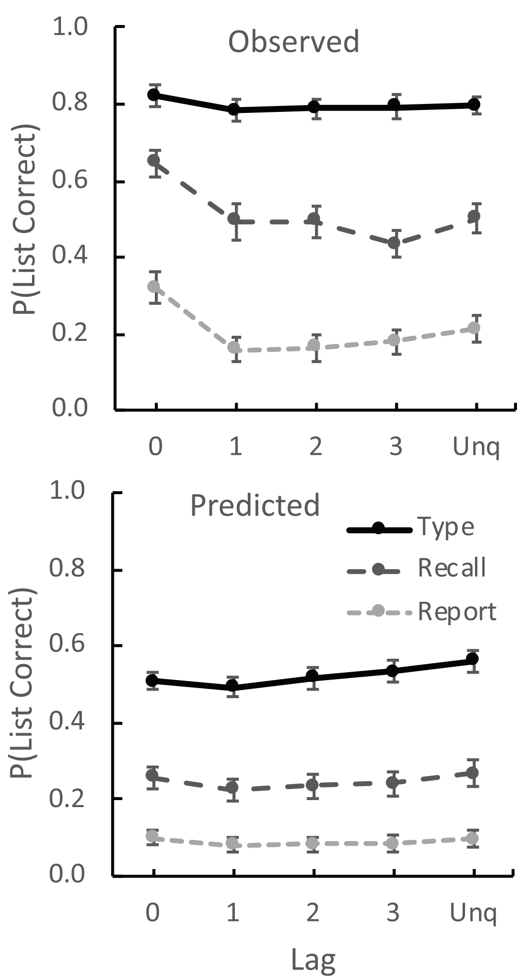


Figure 15. List accuracy in Experiment 2: Observed and predicted probabilities of reporting the whole string correctly as a function of task and lag between repetitions. Unq = unique. Predictions are from the encoding decrease + serial order decrease (ED+SOD) model. Error bars are standard errors of the mean.

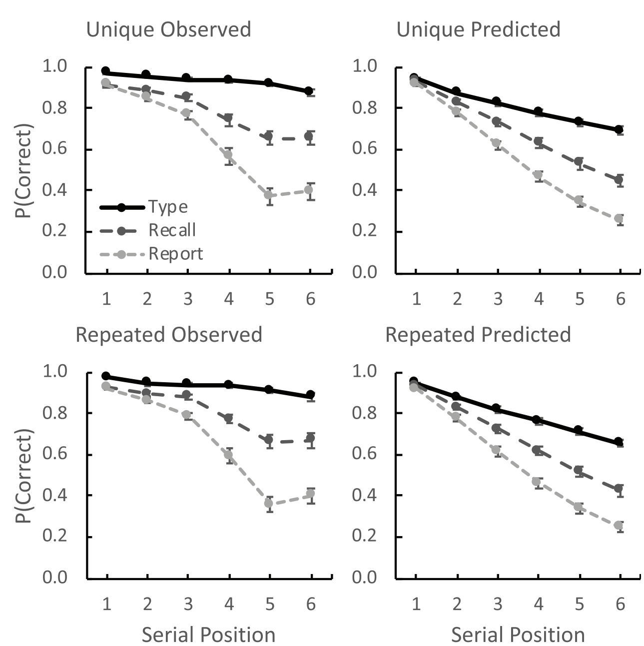


Figure 16. Serial position effects in Experiment 2: Observed and predicted probabilities of responding with the correct item in the correct position for typing, serial recall, and whole report tasks for lists with repeated and unique letters. Predictions are from the encoding decrease + serial order decrease (ED+SOD) model. Error bars are standard errors of the mean.

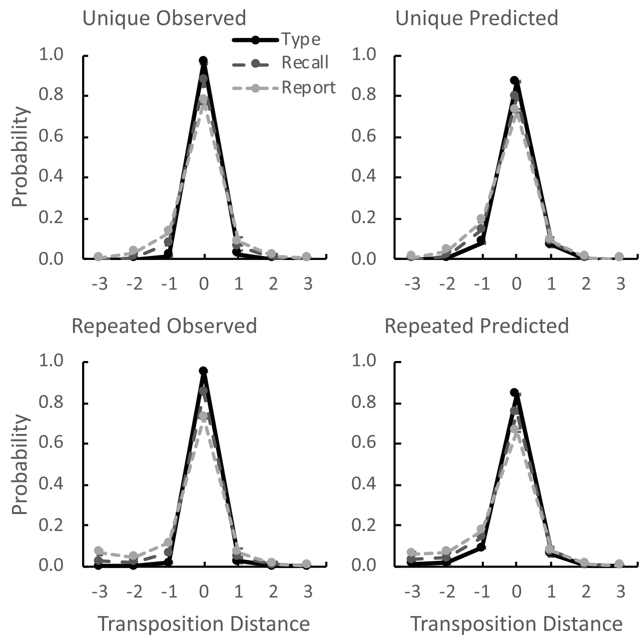


Figure 17. Transposition gradients in Experiment 2: Observed and predicted probabilities of reporting items in the correct position (0), positions before (negative values), and positions after (positive values). Predictions are from the encoding decrease + serial order decrease (ED+SOD) model. Error bars are standard errors of the mean.

.7704 (.0775) and the average *rmsd* was .0832 (.0224), reflecting CRU's ability to capture the general trends while missing the details.

Error ratio. The mean observed and predicted proportions of omissions and transpositions following a skipped item—the components of the error ratio—are plotted in Figure 21. In the observed data, transpositions were more frequent than omissions, producing error ratios >1.0 for all 24 participants. In the predicted data, transpositions were less frequent than omissions, producing error ratios <1.0 for all 24 participants. The average error ratio was 2.6330 (1.2638) for the observed data and .5787 (.0776) in the predicted data. I calculated the correlation and *rmsd* between observed and predicted error ratios in 3 × 2 matrices defined by task and unique versus repeated for each participant. The mean correlation was −.1400 (.4924) and the mean *rmsd* was 2.7109 (1.7780), reflecting a poor fit.

As in Experiment 1, I fit a modified model in which the retrieval cue was a mixture of the initial list context and the current context (see Figure 13). The model did not fit well. The likelihood was larger (3120.79 ± 918.32) than the likelihood of the ED+SOD model and BIC was larger (6280.51 ± 1836.63, $t(23) = 13.7162$) in all 24 participants.

The mean β_{Max}, $\beta_{DecType}$, $\beta_{DecRecall}$, and $\beta_{DecReport}$ parameters across participants were .5569, .9987, .9576, and .9277, respectively, close to the values from the ED+SOD fits in Table 8. The mean P(List) parameters were .0001, .0059, and .0047 for copy typing, serial recall, and whole report, respectively. When P(List) = 0, the modified model reduces to the ED+SOD model. Thus, CRU fails to account for the error ratio data, though it accounted for the relation between transposition errors and omission errors defined more broadly to include more than errors following skipped-letter errors (see Error Types above).

Ranschburg effects. The Ranschburg effect has two components: an advantage for both the first and second presentations for immediate repetitions (lag 0) and a disadvantage for the second presentation (but not the first) as other items intervene between repetitions (lags 1, 2 and 3). The components are calculated as difference scores, subtracting accuracy on control items from unique lists from accuracy on repeated items. The control items are selected to match the serial positions of the first and second presentations, which differ from each other (the second presentation is always later in the list) and differ with lag (at lag 0, the first item can appear in positions 1–5; at lag 3, the first item can only appear in positions 1–2). I calculated these differences for each participant. The means across participants appear in the top row of Figure 22.

The observed data showed Ranschburg effects that differed between tasks. The serial recall and whole report tasks showed typical effects. There was an advantage for first and second presentations at lag 0 and a disadvantage for second presentations at lags greater than 0, which was bigger for whole report than for serial recall. By contrast, copy typing showed neither effect very strongly. These findings replicated across participants. Table 10 shows the number of participants showing each component in each task. Most participants (>20) showed the lag 0 advantage and the lag > 0 disadvantage in serial recall and whole report, but the numbers showing the components in copy typing were closer to chance.

The bottom row of Figure 22 shows the Ranschburg effects predicted by the best-fitting ED+SOD model. CRU captured the basic shape—an advantage for lag 0 for both responses and a

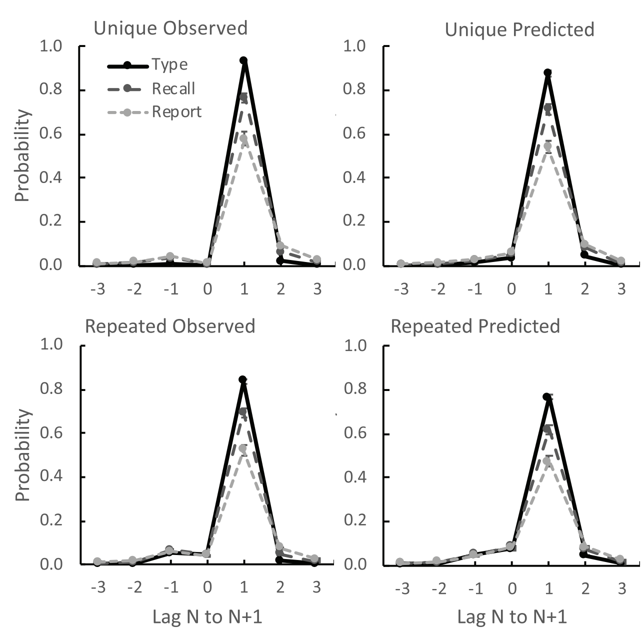


Figure 18. Contiguity effects in Experiment 2: Observed and predicted lag conditional recall probabilities as a function of task for unique and repeated lists. Predictions are from the encoding decrease + serial order decrease (ED+SOD) model. Error bars are standard errors of the mean.

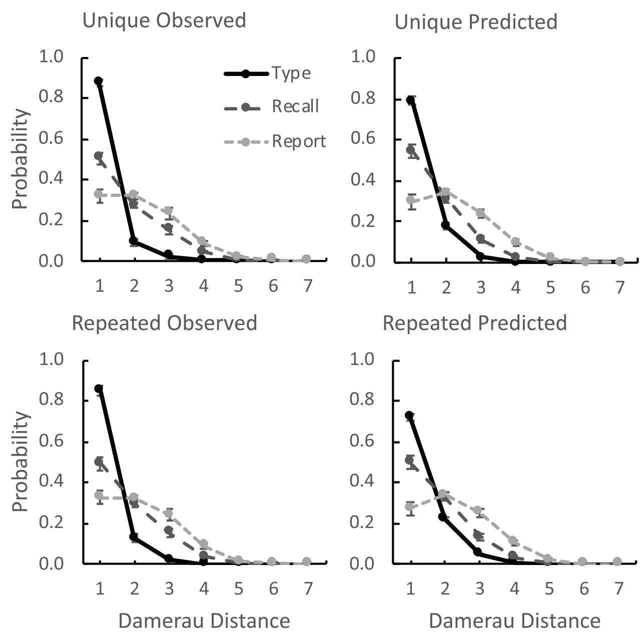


Figure 19. Error magnitudes in Experiment 2: Observed and predicted distributions of Damerau edit distance between correct and error strings for unique and repeated lists in each task. Predictions are from the encoding decrease + serial order decrease (ED+SOD) model. Error bars are standard errors of the mean.

disadvantage for lags > 0 for second presentations—but it greatly underestimated the magnitude of the disadvantage and failed to capture the observed differences between tasks. The mean correlation across participants between observed and predicted values was .5305 (.1275). The mean *rmsd* was .0857 (.0169). The poor fit seriously challenges CRU's ability to account for within-list repetitions, so I tried some modifications to see whether I could obtain a better fit.

Ranschburg mechanisms. Most theories of the Ranschburg effect propose special mechanisms to explain each component. There are three different explanations of the deficit for second presentations at lags greater than 0: *Perception accounts* attribute it to misperception of the second item (Bjork & Murray, 1977; Kanwisher, 1987; Santee & Egeth, 1980). Some memory studies rule out this possibility by having participants say the items out loud as they are presented (Jahnke, 1969) but it is a viable account of disadvantages with brief simultaneous presentations (Bjork & Murray, 1977; Santee & Egeth, 1980). *Memory accounts* attribute the deficit to weakened memory representations (Jahnke, 1969), although this can be ruled out to some extent by evidence of an advantage for within-list repetitions in recognition tasks, indicating that both items must be represented (Wolf & Jahnke, 1968). *Decision accounts* attribute the deficit to *response suppression*: Responses to items are suppressed after they are made, so the response to a repeated item will be less available the second time it is presented (Henson, 1998a; Jahnke, 1969). Response suppression is a popular assumption in models of serial recall as a way of preventing response repetitions in lists that do not contain repeated items (Farrell & Lewandowsky, 2012).

I implemented perception, memory, and decision accounts of the Ranschburg effect in CRU's ED+SOD model by allowing perceptual, memorial, and decisional parameters to change for second presentations of within-list repetitions. The *perception model* decreased the encoding sensitivity parameter g for second presentations, making letters more confusable and less likely to be encoded correctly. The reduction in g was implemented with a parameter g_{Red} that multiplies the current value of g, so g becomes $g \cdot g_{Red}$. The *memory model* decreased β in serial order encoding and retrieval of second presentations, making them less likely to be retrieved accurately. The reduction in β was implemented with a parameter β_{Red} that is subtracted from β, so β becomes $\beta - \beta_{Red}$, constrained so $\beta \geq 0$. The *decision model* implemented response suppression: It increased the threshold θ in the decision process (Equation 9), making previous responses less competitive in the race between diffusions. The increase in θ was achieved with a parameter θ_{Inc} that is added to θ, so θ becomes $\theta + \theta_{Inc}$. As in previous fits, the baseline threshold θ was fixed at 200 for all tasks and conditions. The memory model added θ_{inc} to the baseline threshold of 200. To capture the improved performance at lag 0, I did not apply these adjustments following the first presentation in immediate repetitions.

I ran exploratory simulations and found that each model could produce the observed pattern of the observed Ranschburg effect if the parameters varied between tasks. For the perception model, a monotonic reduction in g_{Red} from copy typing to serial recall to whole report produced the desired gradation across tasks. For the memory model and the decision model, monotonic increases in β_{Red} and θ_{Inc} across typing, recall, and report tasks produced the desired gradation.

Next, I fitted the three models to the data from all 24 participants individually to obtain the best-fitting parameters. To reduce the time it took to fit the models, I took advantage of the relative independence of encoding and memory updating parameters and fixed the parameters whose values were not adjusted when items repeated to the best-fitting values from the ED+SOD model. When fitting the perception model, which reduces g, I fixed the β parameters at their best-fitting values. When fitting the memory model, which reduces β, I fixed the g parameters at their best-fitting values. When fitting the decision model, which increases θ in the serial order retrieval process, I fixed the g parameters at their best-fitting values. I counted both the fixed and free parameters in calculating the BIC penalty term. The perception, memory, and threshold models each had 11 parameters.

The likelihoods, BIC scores and best-fitting parameters for the perception, memory, and decision models are presented in Table 11. For the perception model, the g_{Max}, $g_{DecType}$, $g_{DecRecall}$, and $g_{DecReport}$ parameters were similar to the values in the ED+SOD fits (see Table 7) and the $g_{RedType}$, $g_{RedRecall}$, and $g_{RedReport}$ parameters decreased monotonically, as in the exploratory simulations that produced the Ranschburg effect. For the memory model, the β_{Max}, $\beta_{DecType}$, $\beta_{DecRecall}$, and $\beta_{DecReport}$ parameters were close to the ED+SOD fits, but the $\beta_{RedType}$, $\beta_{RedRecall}$, and $\beta_{RedReport}$ parameters were close to the minimum value of 0, at which the model is no different from the ED+SOD model. For the threshold model, the β_{Max}, $\beta_{DecType}$, $\beta_{DecRecall}$, and $\beta_{DecReport}$ parameters were close to the previous values, but the $\theta_{IncType}$, $\theta_{IncRecall}$, and $\theta_{IncReport}$ parameters decreased monotonically from copy typing to serial recall to whole report instead of increasing monotonically, though the differences were small. To determine whether the differences mattered, I fit a version of the threshold

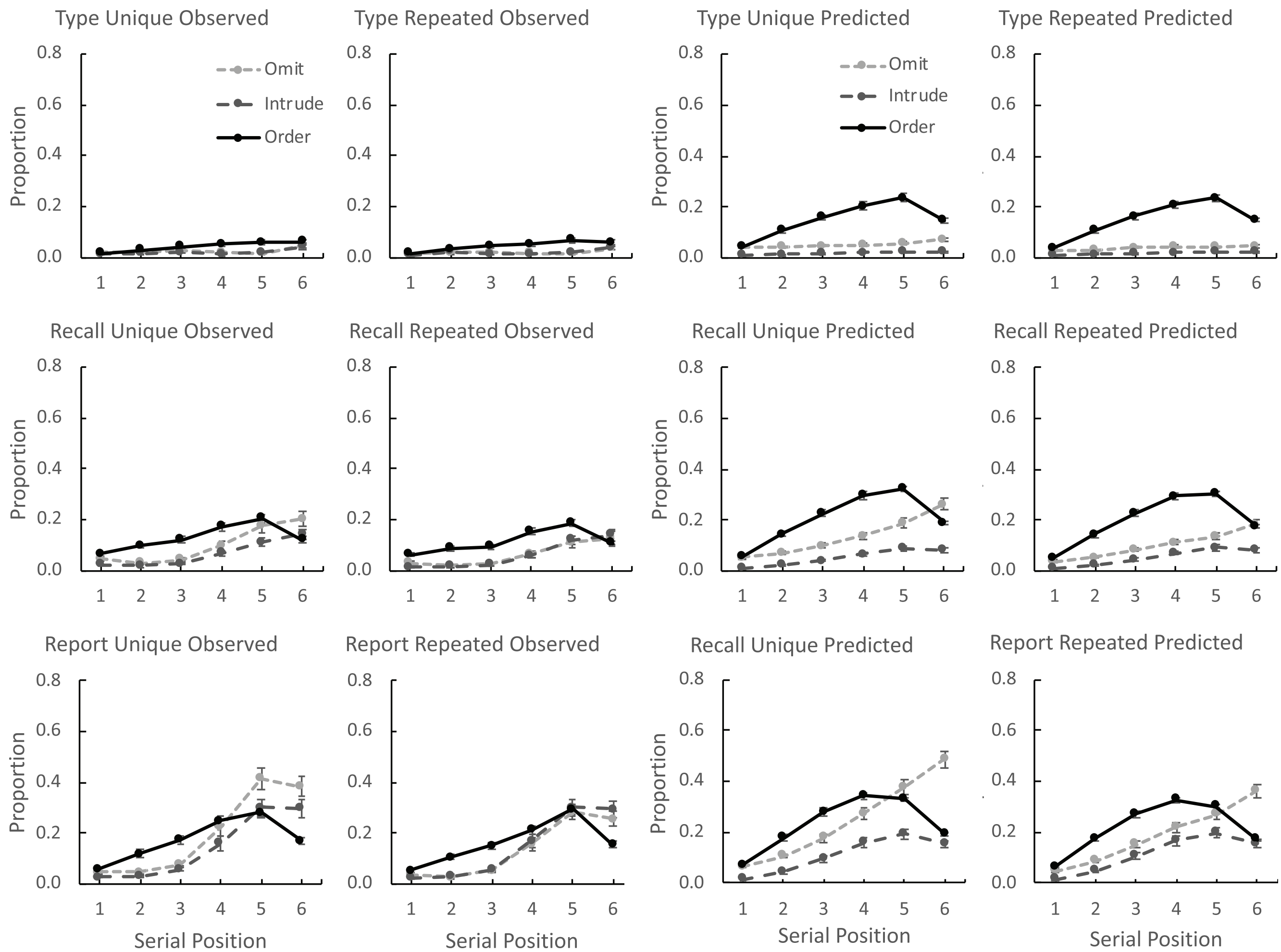


Figure 20. Error types in Experiment 2: Proportions of observed and predicted order, intrusion, and omission errors for unique and repeated lists as a function of task and serial position. Predictions are from the encoding decrease + serial order decrease (ED+SOD) model. Error bars are standard errors of the mean.

model in which there was only one value of θ_{Inc} for all three tasks. The likelihood, BIC, and best-fitting parameters for this *single threshold model* are also presented in Table 11.

Model comparisons are presented in Table 12. The perception model fit better than the ED+SOD model in eight participants and the BIC difference was essentially 0. The memory model fit worse than the ED+SOD model in all 24 participants, although the likelihoods were very similar and the BIC difference was about the same as the difference in the penalty term from adding the three extra parameters. The threshold model with separate parameters for each task fit better than the ED+SOD model in all 24 participants and the BIC difference was substantial. The separate threshold model fit better than the single threshold model in 19 participants, but the BIC difference was essentially 0. Variation in the threshold increment between tasks does not seem necessary to produce good fits. Overall, model selection based on BIC favors the decision models, which implement response suppression.

The next step was to generate predictions for the Ranschburg effect. I used the best-fitting parameters to simulate the perception, memory, and decision models on the same lists as the participants and scored the simulated data with the same routines used for the real data. The predicted Ranschburg effects are presented in Figure 23. All models captured the advantage at lag 0 but overestimated the magnitude for the typing task. The perception model captured the decrease in accuracy at lags > 0 and the modulation across tasks, though the predicted effects were somewhat smaller than the observed effects (cf. Figure 22). The correlation between observed and predicted values was .6035 (.1303) and the *rmsd* was .0791 (.0161).

The memory model estimates of β_{red} were essentially 0, so it makes the same predictions as the ED+SOD model, failing to capture the magnitude of the deficit at lags > 0 and failing to capture the modulation across tasks. The predictions of the ED+SOD model are replotted in the middle panel of Figure 23 for comparison. The correlation between observed and predicted values was .5304 (.1275) and the *rmsd* was .0857 (.0169).

The decision model predicted deficits for lags > 0 but failed to capture the modulation across tasks. The predicted values did not vary much between tasks and their variation was opposite to the observed data, predicting greater deficits for typing than for serial

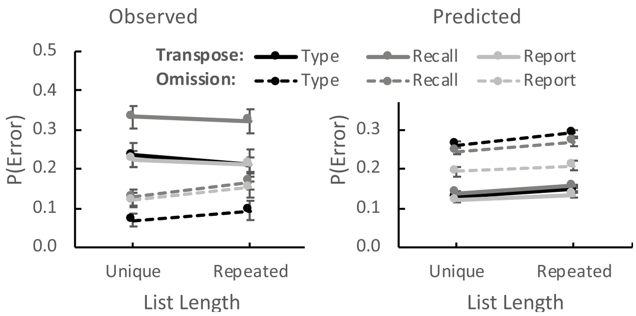


Figure 21. Error ratio components in Experiment 2: Proportions of observed and predicted transposition and omission errors following an initial omission for unique and repeated-item lists in each task. The error ratio (fill-in/in-fill) is the ratio of transpositions to omissions. Predictions are from the encoding decrease + serial order decrease (ED+SOD) model. Error bars are standard errors of the mean.

recall and greater deficits for serial recall than for whole report. The correlation between observed and predicted values was .4702 (.1296) and the *rmsd* was .0919 (.0135).

The Ranschburg effect correlations were higher for the perception model than for the memory model in 19 participants and higher than the separate threshold model in 23 participants. Correlations for the memory model were higher than the separate threshold model in 19 participants. The *rmsd*s for the perception model were lower than the memory model in 19 participants and lower than the separate threshold model in 23 participants. The *rmsd*s were lower for the memory model than for the separate threshold model in 18 participants. Overall, the predictions support the perception model over the others, suggesting that the Ranschburg effect results from a deficit in CRU's item encoding process.

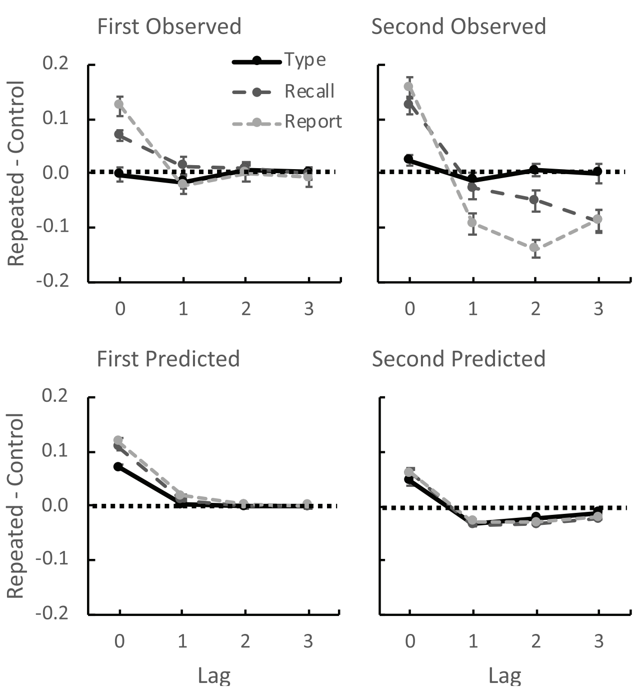


Figure 22. Ranschburg effects in Experiment 2: Observed and predicted differences between first and second repeated items and control items from corresponding serial positions as a function of task and lag. Predictions are from the encoding decrease + serial order decrease (ED+SOD) model. Error bars are standard errors of the mean.

Table 10

Number of Participants (of 24) Replicating Ranschburg Components (Advantage for Lag 0; Disadvantage for Lag > 0) for First and Second Presentations of the Repeated Item in Each Task

Presentation	Lag 0 advantage		Lag > 0 disadvantage	
	First	Second	First	Second
Copy typing	16	17	12	10
Serial recall	21	22	11	20
Whole report	22	22	17	23

It is hard to reach strong conclusions from these results because the fits and predictions favor different models. It occurred to me that the better fits of the decision model might reflect the benefits of increasing threshold on other aspects of performance besides repeated items. The threshold increases after every response, whether or not the item is repeated or the list contains repetitions. Increasing the threshold reduces competition from previous items, which may be beneficial, especially near the end of the list (Lewandowsky & Farrell, 2008). The perception model may have fit worse because it lacked these other benefits.

I decided to compare perception and decision models on more even ground, giving perceptual models the advantage of threshold adjustment after every response. I compared four models formed by factorially combining whether perceptual adjustment g_{Red} and threshold adjustment θ_{Inc} were fixed across tasks or allowed to vary. To reduce the number of parameters to be fitted, I fixed the g_{Max}, g_{Dec}, β_{Max}, and β_{Dec} parameters to the best-fitting values from the ED+SOD fits. The previous fits suggest they were affected little by adjustment in g_{Red} and θ_{Inc} (compare Tables 8 and 11). The likelihood, BIC, and best-fitting parameters for each model are presented in Table 13. Model comparisons are presented in Table 14.

All of the models fit about as well as the first version of the decision model and they fit better than the ED+SOD model. I attribute this to the benefit of adjusting thresholds after every response. The differences between the BIC values were small but they were consistent across participants. Compared with the baseline model in which g_{Red} and θ_{Inc} were fixed across tasks, the majority of the participants were fit worse when θ_{Inc} was allowed to vary between tasks, both by itself and in conjunction with between-task variation in g_{Red}. Compared with the baseline model, varying g_{Red} between tasks improved the fit for the majority of the participants. The model in which only g_{Red} varied between tasks fit better than the model in which only θ_{Inc} varied between tasks. Varying both g_{Red} and θ_{Inc} between tasks produced worse fits than the baseline model because the small improvement in likelihood was outweighed by the bigger BIC penalty for the extra 4 parameters. By these criteria, the perception varied model would be selected over the others. Given the small differences in BIC scores, it may be safer to accept the null hypothesis.

Table 11
Experiment 2 Ranschburg Models: Measures of Goodness of Fit and Best-Fitting Parameter Values for Perception (Reduces g), Memory (Reduces β), and Decision Models (Increases θ), Including Separate Thresholds for Each Task and a Single Threshold for All Three Tasks

Perception model								
Likelihood	BIC	g_{Max}	$g_{DecType}$	$g_{DecRecall}$	$g_{DecReport}$	$g_{RedType}$	$g_{RedRecall}$	$g_{RedReport}$
3029.26	6097.44	.3603	.9818	.9276	.8690	.9689	.8983	.8004
Memory model								
Likelihood	BIC	β_{Max}	$\beta_{DecType}$	$\beta_{DecRecall}$	$\beta_{DecReport}$	$\beta_{RedType}$	$\beta_{RedRecall}$	$\beta_{RedType}$
3033.62	6106.15	.5590	.9996	.9644	.9346	.0013	.0005	.0020
Decision model – Separate thresholds								
Likelihood	BIC	β_{Max}	$\beta_{DecType}$	$\beta_{DecRecall}$	$\beta_{DecReport}$	$\theta_{IncType}$	$\theta_{IncRecall}$	$\theta_{IncReport}$
2977.98	5994.87	.5334	.9994	.9673	.9385	10.03	8.10	7.67

Decision model – Single threshold						
Likelihood	BIC	β_{Max}	$\beta_{DecType}$	$\beta_{DecRecall}$	$\beta_{DecReport}$	θ_{Inc}
2981.10	5994.05	.5346	.9998	.9661	.9374	8.13

Note. BIC = Bayesian information criterion.

I calculated predictions as before, simulating each model with the best-fitting parameters for each participant and calculating the seven summary statistics. The predicted Ranschburg effects for each model are plotted in Figure 24. The correlations and *rmsd*s between observed and predicted values for each model are presented in Table 15. The four models make the same predictions for the first presentation of the repeated item. They differ in their predictions for the second presentation. Models in which *g* was allowed to vary produced higher correlations and smaller *rmsd*s than models in which *g* was fixed, but models in which θ was allowed to vary produced lower correlations and larger *rmsd*s than models in which θ was fixed. This can be seen in Figure 24: The predictions of models with *g* varied, presented in bottom two plots on the left, show the same modulation across tasks as the observed Ranschburg effect. The predictions of models with *g* fixed, presented in the top two plots on the left, do not show the modulation. Varying θ did not produce the desired modulation across tasks.

The models made very similar predictions for the other summary statistics. The mean correlations and *rmsd*s across participants for each model are presented in Figure 25, except for the error ratios. Apart from the Ranschburg effects, there are few differences between the models. This follows because every model used the same g_{Max}, g_{Dec}, β_{Max}, and β_{Dec} parameters that produce these effects.

In this set of models, the model that made the best predictions was the same as the model that produced the best BIC scores. The Ranschburg effect seems to be accounted for best by a model that assumes variation in perceptual confusability (g_{Red}) across tasks but no variation in decision threshold adjustment (θ_{Inc}) across tasks. Putting the models on an even ground by allowing them all to benefit from threshold adjustment seems to have resolved the ambiguity in the initial explorations of the Ranschburg effect.

Discussion

For the most part, Experiment 2 replicated the findings of Experiment 1. The observed results were much like the observed results for list length of six in Experiment 1. As in Experiment 1, the best-fitting ED+SOD model required decrease in both encoding sensitivity *g* and serial retrieval β across serial position that increased from copy typing to serial recall to whole report. Parameters varied meaningfully across tasks, with lower values associated with poorer performance. Model predictions were similar to Experiment 1. Transposition gradients, contiguity effects, and Damerau distances were predicted very accurately, whereas list accuracy, serial position effects, and error types were predicted less accurately, and error

Table 12
Experiment 2 Ranschburg Models: Model Comparisons Assessed by Differences in BIC Scores, Number of Participants Showing the Difference, and t Tests of the Difference in Experiment 2

Measure	ED+SOD – P	ED+SOD – M	ED+SOD – T3	T1 – T3
BIC difference	−1.43	−10.14	101.14	0.82
$N > 0$	8	0	24	19
$t(23)$	0.984	24.484	12.802	0.472

Note. BIC = Bayesian information criterion; ED+SOD = encoding decrease + serial order decrease model; P = perception model (reduces g); M = memory model (reduces β); T3 = Threshold model (increases θ) with separate thresholds for each task; T1 = Threshold model with the same threshold for all tasks.

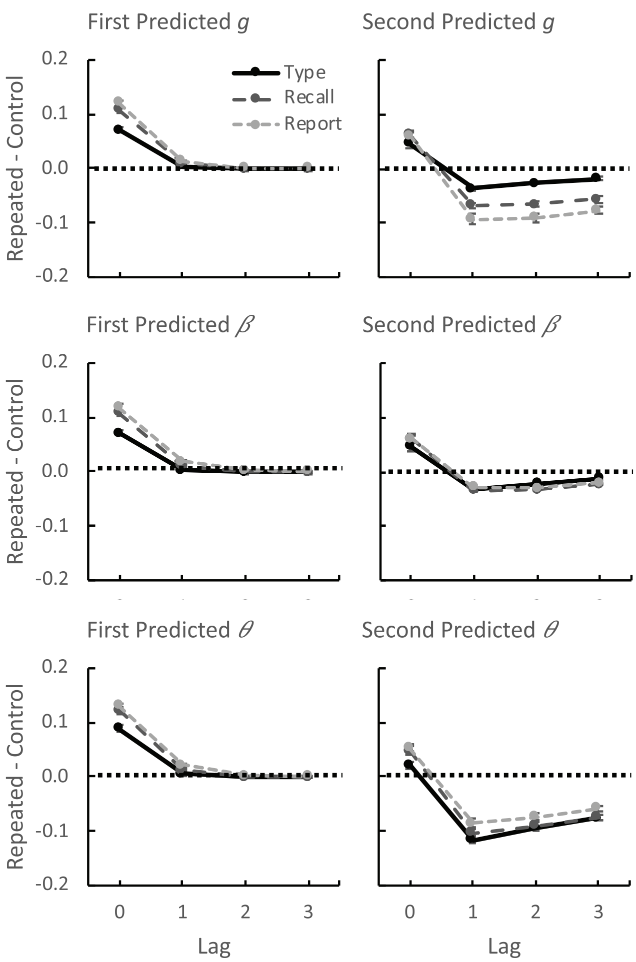


Figure 23. Predicted Ranschburg effects in Experiment 2: Predicted differences between first and second repeated items and control items from corresponding serial positions as a function of task and lag for the encoding adjustment model (g), the memory adjustment model (β), and the threshold adjustment model (θ) averaged over participants. Error bars are standard errors of the mean.

ratios were mispredicted. Overall, the observed results are consistent with the hypothesis that serial order phenomena in all three tasks are reflections of the same underlying process. The model fitting generally supports CRU but found difficulty accounting for the error ratio.

The main purpose of Experiment 2 was to manipulate within-list repetitions, which challenge theories of serial order (Lashley, 1951), memory (Ranschburg, 1902), and perception (Kanwisher, 1987). The observed results showed Ranschburg effects that increased from copy typing to serial recall to whole report. The effects with copy typing were small and inconsistent across participants, which might suggest there is no Ranschburg effect in typing. The best-fitting ED+SOD model underpredicted the magnitude of the observed Ranschburg effects and failed to capture the differences between tasks (see Figure 21), so I tried three modifications to CRU based on interpretations of Ranschburg and repetition blindness effects in the literature. The perception model decreased g (Bjork & Murray, 1977; Kanwisher, 1987; Santee & Egeth, 1980), the memory model decreased β (Jahnke, 1969), and the decision model increased θ (Henson, 1998a) for the second presentation of the repeated item. Ultimately, the data were fit best by a combination of the perception and decision models, in which g_{Red} varied between tasks but θ_{Inc} was fixed across tasks. The perception component accounted for the variation in the Ranschburg effect across tasks. The fits were far from perfect, which invites further exploration of the perception model and its alternatives.

General Discussion

Are serial order phenomena in perception, memory, and action manifestations of a single underlying mechanism? Are serial order phenomena the same in whole report, serial recall, and copy typing tasks? Can a single theory account for serial order phenomena in all three tasks with meaningful variation in its parameters? For the first two questions, the data suggest a clear "yes." In both experiments, the seven summary statistics (list accuracy, serial position, transposition gradient, lag CRP, error magnitude, error type, and error ratio) were similar across the three tasks, differing quantitatively rather than qualitatively. In Experiment 1, list length decreased performance in a similar manner in all three tasks. In Experiment 2, repeating items within lists produced a Ranschburg effect in serial recall and whole report but not in copy typing. These similarities encourage further investigation of empirical parallels between perception, memory, and action to identify other commonalities and differences.

The answer to the theoretical question is promising but less clear. In both experiments, the best-fitting model was the ED+SOD model, which assumed that encoding sensitivity (g) and the balance between new and old information in serial retrieval (β) decreased more across serial position in whole report than in serial

Table 13
Experiment 2 Perception and Decision Ranschburg Models: Measures of Goodness of Fit and Best-Fitting Parameter Values for Models in Which Perception (g) and Decision (θ) Are Fixed (F) or Varied (V) Between Tasks

Measure	Likelihood	BIC	g_{DecTyp}	g_{DecRec}	g_{DecRep}	θ_{incTyp}	θ_{incRec}	θ_{incRep}
gFθ F	2987.81	6003.92		0.9034			6.86	
gFθ V	2985.17	6005.73	0.9034			8.25	6.81	6.72
gVθ F	2986.00	5999.61	0.9712	0.9091	0.8135		6.86	
gVθ V	2983.37	6009.20	0.9712	0.9091	0.8135	8.25	6.81	6.72

Note. BIC = Bayesian information criterion.

Table 14
Experiment 2 Perception and Decision Ranschburg Models: Model Comparisons Assessed by Mean BIC Differences Across Participants, the Number (N) of Participants Showing a Positive BIC Difference, and a t Test Comparing the BIC Difference With 0

Measure	gFθF – gFθV	gFθF – gVθF	gFθF – gVθV	gFθV – gVθF	gFθV – gVθV	gVθF – gVθV
BIC difference	−1.81	4.31	−5.28	6.12	−3.47	−9.59
$N > 0$	5	24	4	23	4	1
$t(23)$	−1.277	6.489	−3.116	4.294	−5.218	−6.765

Note. BIC = Bayesian information criterion; g = perceptual adjustment; θ = decision (threshold) adjustment; F = fixed; V = varied.

recall and more in serial recall than in copy typing. The results replicated well across participants. Across the two experiments, 48 of 48 participants were fit better by the ED+SOD model than the baseline model and 48 were fit better by the ED+SOD model than the next best competitor (Tables 4 and 8). However, the model could not account for error ratios.

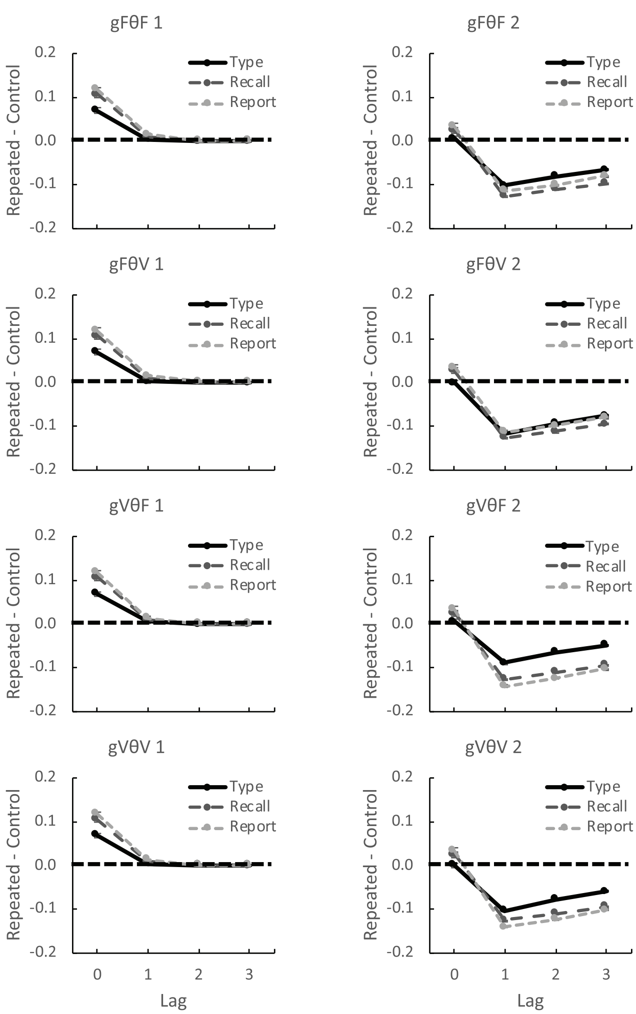


Figure 24. Predicted Ranschburg effects in Experiment 2: Predicted differences between first and second repeated items and control items from corresponding serial positions as a function of task and lag for the factorial combination of encoding adjustment (g) and threshold adjustment (θ) versus fixed (F) and varied (V) values across tasks, averaged over participants. Error bars are standard errors of the mean.

A model that unifies serial order phenomena in perception, memory, and action must fit the data for each task well and must fit as well as or better than competing models. The model fits and predictions address the first criterion. Comparisons with competing models await future research. The mean correlations and *rmsd*s (across participants) between the observed summary statistics and the ED+SOD model predictions are presented in Table 16 for each task in each experiment (excluding error ratio). The correlations were somewhat lower for typing than for serial recall and whole report, likely because the effects were smaller in typing in some measures, reducing correlation by restricting the range. The *rmsd*s are not affected by range and were about the same for all three tasks. Thus, the CRU ED+SOD model predicts each task (roughly) equally well, fulfilling the first criterion. However, the model predicted error ratios that were opposite to the observed ones and failed to capture the recency effect in serial recall and whole report serial position curves. Other supplementary assumptions may allow CRU to predict error ratios and recency. Successful prediction would allow CRU to be considered a viable model of serial order that could be compared with existing models built on different assumptions.

The model was tested in two experiments that manipulated important variables in the serial order literature: list length and within-list repetitions. The CRU ED+SOD model accounted for list length effects well, predicting their effects on all of the summary statistics without changing any parameters to accommodate list length. List length effects occurred because longer lists provide more competition in encoding and retrieval, more opportunities for error, and more updating steps in which g and β decrease. List

Table 15
Experiment 2 Perception and Decision Ranschburg Models: Correlations and rmsd Between Model Predictions and Observed Ranschburg Effects

Measure	Overall	Resp 1	Resp 2
Correlation			
gFθF	0.5087	0.4519	0.6232
gFθV	0.4841	0.4562	0.5618
gVθF	0.5456	0.4514	0.6782
gVθV	0.5241	0.4510	0.6324
rmsd			
gFθF	0.0897	0.0686	0.1051
gFθV	0.0922	0.0684	0.1094
gVθF	0.0875	0.0686	0.1013
gVθV	0.0900	0.0687	0.1054

Note. g = perceptual adjustment; θ = threshold adjustment; F = fixed; V = varied.

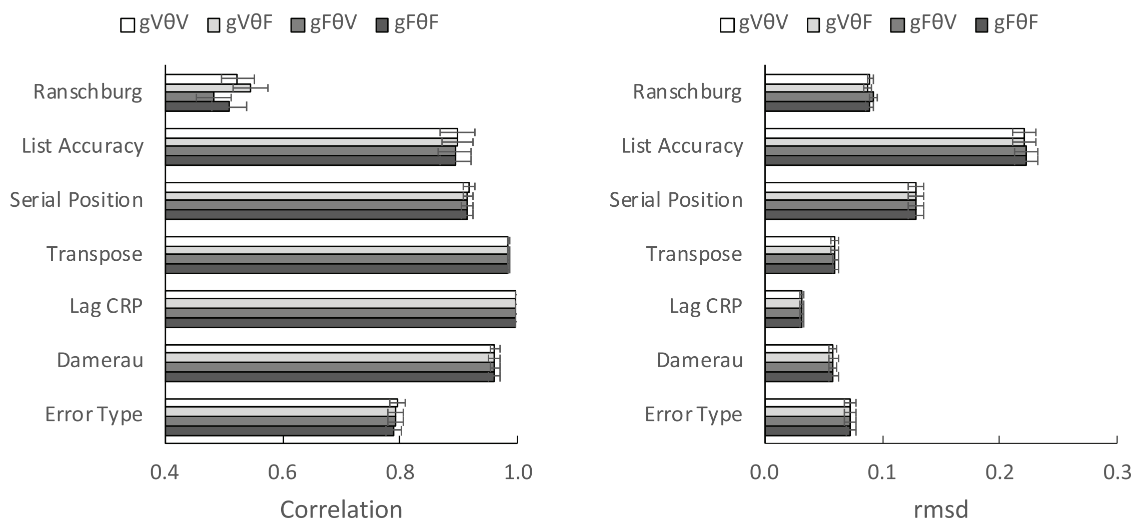


Figure 25. Predicted summary statistics in Experiment 2: Mean correlations and root mean squared deviations (rmsd) between observed and predicted summary statistics for the for the factorial combination of encoding adjustment (*g*) and threshold adjustment (θ) versus fixed (F) and varied (V) values across tasks, averaged over participants. Error bars are standard errors of the mean.

length effects reflect interference (Endress & Szabó, 2017; Oberauer & Lin, 2017) rather than fixed resources or fixed numbers of slots. In this respect, CRU is like models of serial recall (Burgess & Hitch, 1999; Farrell, 2012; Henson, 1998b; Lewandowsky & Farrell, 2008; Page & Norris, 1998), which also eschew assumptions about capacity or slot limitations.

The CRU ED+SOD model could not account for within-list repetitions without adding special mechanisms that reduced perceptual sensitivity differentially across tasks and increased decision thresholds after every response. The differential perceptual sensitivity accounted for the variation in Ranschburg effects across tasks and the threshold increase improved performance more generally. The extended model captured important aspects of the Ranschburg effect but left substantial room for improvement. Thus, CRU's account of the Ranschburg effect should be viewed as tentative.

Theory Development or Kludges?

CRU is built from core assumptions about the representations of items and contexts, associations between items and contexts, and the evolution of context through updating. These assumptions are expressed in three parameters, encoding sensitivity *g*, context updating β, and decision threshold θ, whose interactions are ex-

Table 16
Correlations and Root Mean–Squared Deviations (rmsd) Between Observed and ED+SOD Predicted Summary Statistics for Each Task in Experiments 1 and 2 Averaged Over Participants

Experiment	List accuracy	Serial position	Transpose gradient	Lag CRP	Damerau distance	Error type	*M*
Experiment 1 correlations							
Type	0.6024	0.6615	0.9931	0.9989	0.9613	0.4537	0.7785
Recall	0.9793	0.8572	0.9693	0.9821	0.9019	0.7104	0.9000
Report	0.9371	0.9374	0.9528	0.9780	0.9019	0.7956	0.9171
Experiment 2 correlations							
Type	−0.0078	0.6850	0.9904	0.9985	0.9777	0.5431	0.6978
Recall	0.3242	0.8703	0.9817	0.9947	0.9423	0.7486	0.8103
Report	0.6397	0.9414	0.9778	0.9901	0.9358	0.7621	0.8745
Experiment 1 *rmsd*							
Type	0.1932	0.1190	0.0487	0.0208	0.0729	0.0622	0.0861
Recall	0.2330	0.1794	0.0756	0.0533	0.0758	0.1086	0.1210
Report	0.1204	0.1362	0.0787	0.0395	0.0641	0.1269	0.0943
Experiment 2 *rmsd*							
Type	0.2826	0.1492	0.0605	0.0289	0.0697	0.0725	0.1106
Recall	0.2898	0.1529	0.0633	0.0358	0.0572	0.0763	0.1126
Report	0.1417	0.1298	0.0586	0.0307	0.0508	0.0957	0.0846

Note. CRP = conditional recall probability; ED = encoding decrease model; SOD = serial order decrease model; Type = copy typing; Recall = serial recall; Report = whole report.

pressed in Equations 1–10. Fitting CRU to experimental data requires ancillary assumptions about how these parameters vary across conditions. Ideally, the ancillary assumptions can be justified theoretically as predictions from a model, as a theory of attention might explain the decrease of g across serial position. Often, the justification is weaker and the ancillary assumptions are explorations of possibilities rather than deductions from first principles. In the worst cases, ancillary assumptions are made to fit the data without much justification. While the other cases may be viewed as theory development, the worst cases are kludges—patches on a weak theory—and should be avoided.

The first assumption in fitting CRU to the data is that differences between tasks might be expressed as differences in g and β (θ was fixed at 200). This is justified by the architecture of the model: changing g and β are the only ways to change accuracy. The factorial design that allowed g or β or both or neither to vary between tasks was planned a priori to assess the necessity and sufficiency of variation in g and β to account for the data (Shen & Ma, 2019). Modelers often use nested factorial model comparisons to understand which aspects of the model account for which effects.

The decrease in g and β across serial position is less principled. The data required small differences between tasks for the first serial position that grew progressively over serial position. One possibility would be to allow different values of g and β for each serial position and let the data say how they should change. This would require estimating 18 parameters on average (six for each task) compared with six for the nondecrease models and eight for the decrease models. Eighteen parameters would greatly increase the time required for model fitting, so I opted for the decrease expressed in Equations 11 and 12 to reduce the number of parameters. I chose the specific form of decrease without exploring many alternatives. Equations 11 and 12 produce a monotonic decrease, and I thought that would be sufficient. I have no theoretical commitment to the form of the decrease. The fits to the serial position curves suggest that g and β might increase for the last serial position to capture recency in serial recall and whole report, which may be justified as an edge effect (no competition from following items).

I view the decrease parameters as placeholders for future theories that explain the decrease. The reduction in g might be explained by Bundesen's (1990) theory of visual attention, which was developed to explain whole report and partial report (Shibuya & Bundesen, 1988). It could also be explained by serial scanning in which the number of items scanned varies from trial to trial (Davis, 2010; Heron, 1957; Mewhort et al., 1969). The memory literature offers several explanations of the reduction in β. Many theories of serial and free recall propose some kind of primacy advantage that is explained as extra attention, extra rehearsal, growing interference, or simply postulated without interpretation (for a review, see Lewandowsky & Farrell, 2008). Interfacing CRU with theories of attention and memory is an important goal for future research.

The extensions of CRU to account for the Ranschburg effect were made to accommodate the data, given the failure of the ED+SOD model to account for it. However, the extensions expressed existing positions in the literature (e.g., Henson, 1998a; Jahnke, 1969; Kanwisher, 1987), which mapped naturally onto CRU's existing parameters. The perception model changed g, the memory model changed β, and the decision model changed θ. Thus, the extensions increased the number of values of g, β, and θ required to fit the data but did not require new parameters (the adjustment parameters produced new values of g, β, and θ). The extensions do require more assumptions about the executive processes that change g, β, and θ (Logan & Gordon, 2001), which are implicit in my implementation of the models (and implicit in the existing positions in the literature). Specifying these processes will be difficult because they require the model to know whether the next item to be encoded will be a repetition before implementing the adjustments of g, β, and θ. Most models of serial recall assume response suppression is obligatory and automatic. Immediate repetitions would require control strategies to overcome suppression. Perhaps changes in g could result from perceptual interactions between dependent input channels (Bjork & Murray, 1977; Santee & Egeth, 1980) without executive control. I have assumed independent input channels for analytic convenience. An interactive channels model seems worth exploring.

Relations to Other Theories

Theories of serial order take one of three approaches to representation of order: item coding, position coding, and noisy coding. *Item coding* theories represent order in terms of associations between items (Abrahamse, Jiménez, Verwey, & Clegg, 2010; Ebenholtz, 1963; Ebbinghaus, 1885; Grainger & van Heuven, 2004; Helie, Roeder, Vucovich, Rünger, & Ashby, 2015; Hull, 1943; Lewandowsky & Murdock, 1989; Murdock, 1995; Solway et al., 2012; Whitney, 2001). *Chaining* theories are the simplest versions of item coding: each item is associated with its successor. *Position coding* theories represent order in terms of associations between items and codes that represent positions in the list or string (Brown et al., 2000; Burgess & Hitch, 1999; Davis, 2010; Farrell & Lewandowsky, 2002; Fischer-Baum, Charny, & McCloskey, 2011; Henson, 1998b; Ladd & Woodworth, 1911; Lewandowsky & Farrell, 2008; Tolman, 1948; Young, 1961). The position codes may be distances from the start and end of the list or contexts that are independent of the list items and evolve over the presentation of the list. *Noisy coding* theories represent order in terms of position codes that are inherently uncertain and so are distributed across time or space (Estes, 1972; Gomez et al., 2008; Lee & Estes, 1977, 1981; Ratcliff, 1981). The distributions for adjacent positions overlap more than the distribution for remote positions, and that accounts for a surprising amount of data (also see Compton & Logan, 1993, 1999; Logan, 1996; Logan & Bundesen, 1996).

CRU and its ancestors assume that items are associated with contexts, but the contexts are made of fading representations of items that were previously encoded or retrieved. Consequently, CRU is more closely aligned with item coding theories than with position or noisy coding theories. Lashley (1951) raised objections to simple chaining theories, which subsequent item coding theories have overcome by assuming remote forward and backward associations. CRU's evolving context mimics remote associations, in that more remote items are represented more weakly in the current context than more recent items. CRU implements a kind of behavioral chaining, in which the current response becomes part of the context that retrieves the next response. The current context controls retrieval in the same way working memory contents control the sequence of actions in production system models (Anderson, 2013; Newell, 1990). When a production fires, its action changes working memory, which changes the conditions productions must match to fire on the next cycle.

Thus, the chains in CRU are not the classic chains of Ebbinghaus (1885) and others. CRU's chains are built on information and similarity rather than association.

CRU in perception. Sperling's (1960) whole report task and the research that followed it involved horizontal strings of letters (Mewhort et al., 1969; Rumelhart, 1970). Modern studies of whole report use circular or random displays and are more concerned with capacity than order of report (Adam, Vogel, & Awh, 2017; Shibuya & Bundesen, 1988). Most of the current theories of serial order in perception address reading, describing the processes by which people form representations that bind letter identity and order and match them to representations of words. Early models used slots (McClelland & Rumelhart, 1981) or combinations of letters (Seidenberg & McClelland, 1989), but these models were rejected because they cannot handle transpositions and other errors (Grainger, 2018). Current models use position coding (Davis, 2010; Houghton, 2018), noisy coding (Gomez et al., 2008), and item coding (Grainger & Van Heuven, 2004; Whitney, 2001) and account for a complex array of data.

CRU has implications for theories of orthographic processing that address the binding of position and identity information in the first stages of reading. These theories assume that the bound representation is compared with lexical representations to identify words. CRU does not address that comparison, but its context representations contain information that could be used for that purpose. The comparison process could be based on CRU representations: The set of context representations for a presented letter string could be compared with sets of context representations for words. Alternatively, computations on the context representations may produce the codes that other theories use to compare with lexical representations. Figure 26 illustrates how CRU context vectors can support open bigram coding (Grainger & Van Heuven, 2004; Whitney, 2001) and position coding (Houghton, 2018).

Open bigram theories represent order in terms of ordered pairs of adjacent and nonadjacent (hence open) letters (Grainger & Van Heuven, 2004; Whitney, 2001). Thus, *cat* is represented as *c_a*, *c_t*, and *a_t*. CRU's stored contexts contain information about pairs of letters and their order that can be used to extract open bigrams. Each stored context contains representations of previous items whose values decrease with each successive update. The values within each context specify the order of the letters: the weaker letter came before the stronger letter. Figure 26 illustrates the open bigrams that can be extracted in this way from the context vectors representing DIET. Alternatively, open bigrams may be extracted from the sums of element values across contexts. Earlier letters appear in more contexts than later letters, so their sums are larger: the letter with the larger sum came before the letter with the smaller sum. The rightmost column of Figure 26 illustrates the open bigrams that can be extracted in this way from EDIT, TIED, TIDE, and DIET. The sums of element values across contexts may also be interpreted as position codes. The order of the sums indicates the order of the letters in the string, like position codes (Houghton, 2018). The graph in the bottom of Figure 26 plots the sums for each letter in EDIT, TIED, TIDE, and DIET. Thus, CRU may serve as the front end to models of orthographic processing. Of course, it needs to pass benchmark tests before it can be considered a competitor (e.g., Grainger, 2018).

CRU in memory. Although early theories of serial recall explored noisy coding (Estes, 1972; Lee & Estes, 1977, 1981), chaining (Lewandowsky & Li, 1994; Lewandowsky & Murdock, 1989; Murdock, 1995), and activation (Page & Norris, 1998), more recent theories have converged on position coding (Brown et al., 2000; Burgess & Hitch, 1999; Farrell, 2012; Farrell & Lewandowsky, 2002; Henson, 1998b; Lewandowsky & Farrell, 2008; Shiffrin & Cook, 1978), and chaining is generally viewed as beyond consideration (Henson et al., 1996). CRU is an outlier, representing order in terms of associations between items and contexts made from fading representations of past items, and implementing a kind of behavioral chaining in which each item that is retrieved becomes part of the context that retrieves the next item. Its fits to benchmark phenomena suggest it may be a viable theory of serial recall, but the existing models fit the benchmark phenomena as well and they account for many phenomena CRU has not addressed. The fits suggest CRU has promise but substantial work will be required to realize that promise.

Following Conrad (1964), much of the work on serial recall has focused on the acoustic or articulatory confusability of the items, finding consistent effects of manipulating confusability within and between lists (Hurlstone, Hitch, & Baddeley, 2014). Several theories account for these effects with two stage models, in which one stage represents serial order and the other generates responses from the retrieved items (Burgess & Hitch, 1999; Henson, 1998b; Page & Norris, 1998). Confusability effects occur in the second stage (but see Farrell & Lewandowsky, 2002; Lewandowsky & Farrell, 2008). CRU includes these two stages and attributes confusability among items to the second stage. Logan (2018) included this second stage in a model of skilled typing to account for errors that involved striking keys adjacent to the correct key. Responses were points in the two-dimensional plane of the keyboard and confusions were based on distance (Equation 1). This idea could be generalized to multidimensional representations of response alternatives, like phonological codes for spoken words or letter names.

In principle, CRU should extend beyond serial recall to other memory tasks, like recognition and free recall. CRU may be viewed as an extension of TCM (Howard & Kahana, 2002) and its descendants (Lohnas et al., 2015; Polyn et al., 2009; Talmi et al., 2019), which already address a broad range of phenomena in free recall. CRU's core assumptions about context updating say how representations are formed and those representations should be useful in other tasks. Extending CRU to other memory tasks is an important goal for future research.

CRU in action. In action tasks, CRU may be viewed as an alternative to strict chaining theories (Abrahamse et al., 2010; Helie et al., 2015). CRU may also be viewed as an extension of strict chaining theories that allows more than the most recent item to contribute to retrieval of the next and uses similarity rather than association to guide retrieval. CRU may provide new insights into speech (Dell, Burger, & Svec, 1997) and musical performance (Palmer & Pfordresher, 2003).

Expertise

CRU was inspired by my instance theory of automatization (Logan, 1988) as much as by TCM (Howard & Kahana, 2002), so I believe it extends naturally to skill acquisition and expertise. I have been exploring the possibility of storing traces of the current

Context Vectors

Open Bigrams from Sum

		E	D	I	T		Sum
List	1.0000	0.8660	0.7500	0.6495	0.5625		3.8280
E		0.5000	0.4330	0.3750	0.3248		1.6328
D			0.5000	0.4330	0.3750		1.3080
I				0.5000	0.4330		0.9330
T					0.5000		0.5000

E>D
E>I
E>T
D>I
D>T
I>T

		T	I	E	D		Sum
List	1.0000	0.8660	0.7500	0.6495	0.5625		3.8280
E				0.5000	0.4330		0.9330
D					0.5000		0.5000
I			0.5000	0.4330	0.3750		1.3080
T		0.5000	0.4330	0.3750	0.3248		1.6328

T>I
T>E
T>D
I>E
I>D
E>D

		T	I	D	E		Sum
List	1.0000	0.8660	0.7500	0.6495	0.5625		3.8280
E					0.5000		0.5000
D				0.5000	0.4330		0.9330
I			0.5000	0.4330	0.3750		1.3080
T		0.5000	0.4330	0.3750	0.3248		1.6328

T>I
T>D
T>E
I>D
I>E
D>E

		D	I	E	T		Sum
List	1.0000	0.8660	0.7500	0.6495	0.5625		3.8280
E				0.5000	0.4330		0.9330
D		0.5000	0.4330	0.3750	0.3248		1.6328
I			0.5000	0.4330	0.3750		1.3080
T					0.5000		0.5000

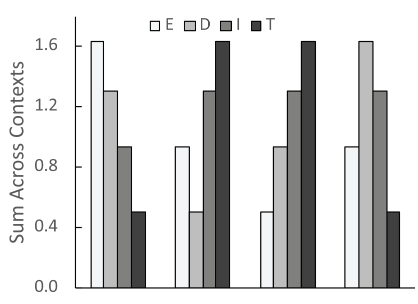
D>I
D>E
D>T
I>E
I>T
E>T

D<I

D<I
D<E
I<E

D<I
D<E
D<T
I<E
I<T
E<T

Open Bigrams from Vectors



Position Codes from Sums

Figure 26. Extraction of open bigrams and position codes from Context Retrieval and Updating model (CRU) representations. Open bigrams based on the sum of element (letter) values are in the leftmost column. Open bigrams for DIET are given under the table for DIET. Position codes based on sums of element values are plotted in the graph at the bottom.

contexts CRU generates when encoding and reporting a list and allowing all the stored traces of the same current context to race at retrieval. Simulations show a power function speedup in response time that is modulated by the probability of storing an instance. Further exploration of learning is an important goal for future research.

CRU was intended as a general model of serial order in perception, memory, and action with the idea that the same serial order process was engaged in all three domains. It is possible that a general model may only apply to novice performance with unfamiliar materials. The experiments used novel random letter strings that had to be encoded, remembered, and reported letter by letter because there was no higher-level structure to connect them. Expert processing of familiar letter strings (words) may not require such a general model. Much of the work on serial order in perception is aimed at understanding reading, where an ordered representation of letters is matched to lexical representations of single words to identify the word. Serial order is encoded but not retrieved. Once the word is identified, subsequent processes can access it without considering the letters. Similarly, skilled typing is driven by the word to be typed, not the letters that make it up. Serial order is retrieved but not encoded. The word is the initial cue for retrieving the letters (Logan, 2018; Yamaguchi & Logan, 2014). Thus, skilled typing might involve two serial order processes, a perceptual process that encodes letter order and allows the word to be identified, and a motor process that retrieves letters given a word. It may be faster and more accurate to use the word representation to communicate between perception and action than to use a sequence of letter representations.

The idea that skilled performance may involve separate serial order processes for perception and action and none for memory challenges the idea that CRU or any other model can account for perception, memory, and action with a single mechanism. Perhaps serial order is controlled differently in experts and novices. Novices dealing with unfamiliar materials may rely on a single version of CRU that controls serial order of individual items in perception, memory, and action. The idea that novice performance is governed by domain general processes has precedents in the literature (Anderson, 1982; Logan, 1988). Experts may develop domain-specific processes to exploit the structure of the specific tasks they perform and the materials specific to the domain. Although the same version of CRU may not be running in the perceptual system and the motor system, it is possible that the serial order mechanisms in each domain are structured like CRU. They may be built by context updating and retrieval, like the domain general mechanism. Thus, CRU may be a single metatheory that accounts for serial order in perception, memory, and action, though different versions CRU are implemented in different domains.

Control Processes

Atkinson and Shiffrin (1968) argued that memory theories should explain the control processes that operate on memory and not just the representations and storage systems. My longstanding interest in control processes leads me to emphasize this aspect of theory more than most (Logan, 2017). From this perspective, the simplicity of CRU's control processes is an important virtue. CRU is driven by an initial command to report or recall, which is set in the list or word part of the current context vector, then the current context vector is compared with stored contexts to retrieve the first item, which then updates the context and retrieves the next item, until the last item is retrieved. Control is required at the beginning to initiate retrieval and at the end to determine what to do next, but the retrieval process proceeds automatically and runs on to completion without top down control. Indeed, this feature of CRU makes it an appropriate model of automatic control, explaining how skilled typists type without thinking (Logan, 2018).

I think the simple control process that guides retrieval in CRU can also work in other chaining theories. The representations of the items control serial order, so the control process is specified by the associations or similarity relations between the items. Noisy coding and position coding theories generally assume that the control system steps through position codes in order without making mistakes. Errors come from confusing items that are associated with similar position codes. Often, the serial order process is not specified in detail. Some refer to a successor inhibition process for choosing the next item (e.g., Estes, 1972) but many are mute on the subject. Specifying the serial order process in greater detail would improve the theories and give a clearer idea of the demands they place on control processes.

Conclusions

The data and modeling suggest a tentative positive answer to the question, do serial order phenomena in perception, memory, and action result from a single underlying mechanism. Empirically, the data from the three tasks showed similar effects that increased in magnitude from typing to serial recall to whole report. Theoretically, CRU was able to fit most of the data from the three tasks with meaningful variation in its parameters, though it had some notable difficulties. This encourages further research on the question of how serial order phenomena are related to gather new data and test alternative models as well as developing CRU to connect it with theories of attention and other memory tasks.

References

Abrahamse, E. L., Jiménez, L., Verwey, W. B., & Clegg, B. A. (2010). Representing serial action and perception. *Psychonomic Bulletin & Review, 17,* 603–623. http://dx.doi.org/10.3758/PBR.17.5.603

Adam, K. C. S., Vogel, E. K., & Awh, E. (2017). Clear evidence for item limits in visual working memory. *Cognitive Psychology, 97,* 79–97. http://dx.doi.org/10.1016/j.cogpsych.2017.07.001

Adelman, J. S. (2011). Letters in time and retinotopic space. *Psychological Review, 118,* 570–582. http://dx.doi.org/10.1037/a0024811

Anderson, J. R. (1982). Acquisition of cognitive skill. *Psychological Review, 89,* 369–406. http://dx.doi.org/10.1037/0033-295X.89.4.369

Anderson, J. R. (2013). *The architecture of cognition*. New York, NY: Psychology Press. http://dx.doi.org/10.4324/9781315799438

Atkinson, R. C., & Shiffrin, R. M. (1968). Human memory: A proposed system and its control processes. *Psychology of Learning and Motivation, 2,* 89–195. http://dx.doi.org/10.1016/S0079-7421(08)60422-3

Bjork, E. L., & Murray, J. T. (1977). On the nature of input channels in visual processing. *Psychological Review, 84,* 472–484. http://dx.doi.org/10.1037/0033-295X.84.5.472

Botvinick, M. M., & Plaut, D. C. (2006). Short-term memory for serial order: A recurrent neural network model. *Psychological Review, 113,* 201–233.

Brown, G. D. A., Preece, T., & Hulme, C. (2000). Oscillator-based memory for serial order. *Psychological Review, 107,* 127–181. http://dx.doi.org/10.1037/0033-295X.107.1.127

Bryden, M. P. (1967). A model for the sequential organization of behaviour. *Canadian Journal of Psychology/Revue canadienne de psychologie, 21,* 37–56. http://dx.doi.org/10.1037/h0082960

Bundesen, C. (1990). A theory of visual attention. *Psychological Review, 97,* 523–547. http://dx.doi.org/10.1037/0033-295X.97.4.523

Burgess, N., & Hitch, G. J. (1999). Memory for serial order: A network model of the phonological loop and its timing. *Psychological Review, 106,* 551–581. http://dx.doi.org/10.1037/0033-295X.106.3.551

Coltheart, M., & Rastle, K. (1994). Serial processing in reading aloud: Evidence for dual-route models of reading. *Journal of Experimental Psychology: Human Perception and Performance, 20,* 1197–1211. http://dx.doi.org/10.1037/0096-1523.20.6.1197

Compton, B. J., & Logan, G. D. (1993). Evaluating a computational model of perceptual grouping by proximity. *Perception & Psychophysics, 53,* 403–421. http://dx.doi.org/10.3758/BF03206783

Compton, B. J., & Logan, G. D. (1999). Judgments of perceptual groups: Reliability and sensitivity to stimulus transformation. *Perception & Psychophysics, 61,* 1320–1335. http://dx.doi.org/10.3758/BF03206183

Conrad, R. (1964). Acoustic confusions in immediate memory. *British Journal of Psychology, 55,* 75–84. http://dx.doi.org/10.1111/j.2044-8295.1964.tb00899.x

Courrieu, P., Farioli, F., & Grainger, J. (2004). Inverse discrimination time as a perceptual distance for alphabetic characters. *Visual Cognition, 11,* 901–919. http://dx.doi.org/10.1080/13506280444000049

Cowan, N., Hogan, T. P., Alt, M., Green, S., Cabbage, K. L., Brinkley, S., & Gray, S. (2017). Short-term memory in childhood dyslexia: Deficient serial order in multiple modalities. *Dyslexia: An International Journal of Research and Practice, 23,* 209–233. http://dx.doi.org/10.1002/dys.1557

Damerau, F. J. (1964). A technique for computer detection and correction of spelling errors. *Communications of the ACM, 7,* 171–176. http://dx.doi.org/10.1145/363958.363994

Davis, C. J. (2010). The spatial coding model of visual word identification. *Psychological Review, 117,* 713–758. http://dx.doi.org/10.1037/a0019738

Dell, G. S. (1986). A spreading-activation theory of retrieval in sentence production. *Psychological Review, 93,* 283–321. http://dx.doi.org/10.1037/0033-295X.93.3.283

Dell, G. S., Burger, L. K., & Svec, W. R. (1997). Language production and serial order: A functional analysis and a model. *Psychological Review, 104,* 123–147. http://dx.doi.org/10.1037/0033-295X.104.1.123

Ebbinghaus, H. (1885). *On memory: A contribution to experimental psychology*. New York, NY: Teachers College, Columbia University.

Ebenholtz, S. M. (1963). Serial learning: Position learning and sequential associations. *Journal of Experimental Psychology, 66,* 353–362. http://dx.doi.org/10.1037/h0048320

Endress, A. D., & Szabó, S. (2017). Interference and memory capacity limitations. *Psychological Review, 124,* 551–571. http://dx.doi.org/10.1037/rev0000071

Estes, W. K. (1972). An associative basis for coding and organization in memory. In A. W. Melton & E. Martin (Eds.), *Coding processes in human memory* (pp. 161–190). Washington, DC: Winston.

Fagot, C., & Pashler, H. (1995). Repetition blindness: Perception or memory failure? *Journal of Experimental Psychology: Human Perception and Performance, 21,* 275–292. http://dx.doi.org/10.1037/0096-1523.21.2.275

Farrell, S. (2012). Temporal clustering and sequencing in short-term memory and episodic memory. *Psychological Review, 119,* 223–271. http://dx.doi.org/10.1037/a0027371

Farrell, S., Hurlstone, M. J., & Lewandowsky, S. (2013). Sequential dependencies in recall of sequences: Filling in the blanks. *Memory & Cognition, 41,* 938–952. http://dx.doi.org/10.3758/s13421-013-0310-0

Farrell, S., & Lewandowsky, S. (2002). An endogenous distributed model of ordering in serial recall. *Psychonomic Bulletin & Review, 9,* 59–79. http://dx.doi.org/10.3758/BF03196257

Farrell, S., & Lewandowsky, S. (2012). Response suppression contributes to recency in serial recall. *Memory & Cognition, 40,* 1070–1080. http://dx.doi.org/10.3758/s13421-012-0212-6

Fias, W., Menon, V., & Szucs, D. (2013). Multiple components of developmental dyscalculia. *Trends in Neuroscience and Education, 2,* 43–47. http://dx.doi.org/10.1016/j.tine.2013.06.006

Fischer-Baum, S., Charny, J., & McCloskey, M. (2011). Both-edges representation of letter position in reading. *Psychonomic Bulletin & Review, 18,* 1083–108.

Gomez, P., Ratcliff, R., & Perea, M. (2008). The overlap model: A model of letter position coding. *Psychological Review, 115,* 577–600. http://dx.doi.org/10.1037/a0012667

Grainger, J. (2018). Orthographic processing: A 'mid-level' vision of reading. *Quarterly Journal of Experimental Psychology: Human Experimental Psychology, 71,* 335–359. http://dx.doi.org/10.1080/17470218.2017.1314515

Grainger, J., & Van Heuven, W. J. B. (2004). Modeling letter position coding in printed word perception. In P. Bonin (Ed.) *Mental lexicon: "Some words to talk about words"* (pp. 1–23). Hauppauge, NY: Nova Science Publishers.

Grossberg, S. (1978). A theory of human memory: Self-organization and performance of sensory-motor codes, maps, and plans. In R. Rosen & F. Snell (Eds.), *Progress in theoretical biology* (Vol. 5, pp. 233–374). New York, NY: Academic Press. http://dx.doi.org/10.1016/B978-0-12-543105-7.50013-0

Hamming, R. W. (1950). Error detecting and error correcting codes. *The Bell System Technical Journal, 29,* 147–160. http://dx.doi.org/10.1002/j.1538-7305.1950.tb00463.x

Healey, M. K., Long, N. M., & Kahana, M. J. (2019). Contiguity in episodic memory. *Psychonomic Bulletin & Review, 26,* 699–720. http://dx.doi.org/10.3758/s13423-018-1537-3

Helie, S., Roeder, J. L., Vucovich, L., Rünger, D., & Ashby, F. G. (2015). A neurocomputational model of automatic sequence production. *Journal of Cognitive Neuroscience, 27,* 1412–1426. http://dx.doi.org/10.1162/jocn_a_00794

Henson, R. N. A. (1998a). Item repetition in short-term memory: Ranschburg repeated. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 24,* 1162–1181. http://dx.doi.org/10.1037/0278-7393.24.5.1162

Henson, R. N. (1998b). Short-term memory for serial order: The Start-End Model. *Cognitive Psychology, 36,* 73–137. http://dx.doi.org/10.1006/cogp.1998.0685

Henson, R. N. A., Norris, D. G., Page, M. P. A., & Baddeley, A. D. (1996). Unchained memory: Error patterns rule out chaining models of immediate serial recall. *Quarterly Journal of Experimental Psychology: Human Experimental Psychology, 49,* 80–115. http://dx.doi.org/10.1080/713755612

Heron, W. (1957). Perception as a function of retinal locus and attention. *The American Journal of Psychology, 70,* 38–48. http://dx.doi.org/10.2307/1419227

Houghton, G. (2018). Action and perception in literacy: A common-code for spelling and reading. *Psychological Review, 125,* 83–116. http://dx.doi.org/10.1037/rev0000084

Howard, M. W., & Kahana, M. J. (2002). A distributed representation of temporal context. *Journal of Mathematical Psychology, 46,* 269–299. http://dx.doi.org/10.1006/jmps.2001.1388

Hull, C. L. (1943). *Principles of behavior: An introduction to behavior theory*. Oxford, UK: Appleton-Century.

Hurlstone, M. J., Hitch, G. J., & Baddeley, A. D. (2014). Memory for serial order across domains: An overview of the literature and directions for future research. *Psychological Bulletin, 140,* 339–373. http://dx.doi.org/10.1037/a0034221

Jahnke, J. C. (1969). The Ranschburg effect. *Psychological Review, 76,* 592–605. http://dx.doi.org/10.1037/h0028148

Kahana, M. J. (1996). Associative retrieval processes in free recall. *Memory & Cognition, 24,* 103–109.

Kanwisher, N. G. (1987). Repetition blindness: Type recognition without token individuation. *Cognition, 27,* 117–143. http://dx.doi.org/10.1016/0010-0277(87)90016-3

Kragel, J. E., Morton, N. W., & Polyn, S. M. (2015). Neural activity in the medial temporal lobe reveals the fidelity of mental time travel. *The Journal of Neuroscience, 35,* 2914–2926. http://dx.doi.org/10.1523/JNEUROSCI.3378-14.2015

Ladd, G. T., & Woodworth, R. S. (1911). *Elements of physiological psychology: A treatise of the activities and nature of the mind from the physical and experimental point of view*. New York, NY: Charles Scribner's Sons.

Lashley, K. S. (1951). The problem of serial order. In L. A. Jeffress (Ed.), *Cerebral mechanisms in behavior* (pp. 112–136). New York, NY: Wiley.

Lee, C. L., & Estes, W. K. (1977). Order and position in primary memory for letter strings. *Journal of Verbal Learning & Verbal Behavior, 16,* 395–418. http://dx.doi.org/10.1016/S0022-5371(77)80036-4

Lee, C. L., & Estes, W. K. (1981). Item and order information in short-term memory: Evidence for multilevel perturbation processes. *Journal of Experimental Psychology: Human Learning and Memory, 7,* 149–169. http://dx.doi.org/10.1037/0278-7393.7.3.149

Lewandowsky, S., & Farrell, S. (2008). Short-term memory: New data and a model. In B. H. Ross (Ed.), *The psychology of learning and motivation* (Vol. 49, pp. 1–48). Burlington, MA: Academic Press.

Lewandowsky, S., & Li, S.-C. (1994). Memory for serial order revisited. *Psychological Review, 101,* 539–543. http://dx.doi.org/10.1037/0033-295X.101.3.539

Lewandowsky, S., & Murdock, B. B., Jr. (1989). Memory for serial order. *Psychological Review, 96,* 25–57. http://dx.doi.org/10.1037/0033-295X.96.1.25

Logan, G. D. (1988). Toward an instance theory of automatization. *Psychological Review, 95,* 492–527. http://dx.doi.org/10.1037/0033-295X.95.4.492

Logan, G. D. (1996). The CODE theory of visual attention: An integration of space-based and object-based attention. *Psychological Review, 103,* 603–649. http://dx.doi.org/10.1037/0033-295X.103.4.603

Logan, G. D. (2017). Taking control of cognition: An instance perspective on acts of control. *American Psychologist, 72,* 875–884. http://dx.doi.org/10.1037/amp0000226

Logan, G. D. (2018). Automatic control: How experts act without thinking. *Psychological Review, 125,* 453–485. http://dx.doi.org/10.1037/rev0000100

Logan, G. D., & Bundesen, C. (1996). Spatial effects in the partial report paradigm: A challenge for theories of visual-spatial attention. In D. L. Medin (Ed.), *The psychology of learning and motivation* (Vol. 35, pp. 243–282). San Diego, CA: Academic Press.

Logan, G. D., & Crump, M. (2011). Hierarchical control of cognitive processes: The case for skilled typewriting. In B. Ross (Ed.), *The Psychology of Learning and Motivation* (Vol. 54, pp. 1–27). Burlington, VT: Academic Press.

Logan, G. D., & Gordon, R. D. (2001). Executive control of visual attention in dual-task situations. *Psychological Review, 108,* 393–434. http://dx.doi.org/10.1037/0033-295X.108.2.393

Logan, G. D., Van Zandt, T., Verbruggen, F., & Wagenmakers, E.-J. (2014). On the ability to inhibit thought and action: General and special theories of an act of control. *Psychological Review, 121,* 66–95. http://dx.doi.org/10.1037/a0035230

Logan, G. D., & Zbrodoff, N. J. (1998). Stroop type interference: Congruity effects in color naming with typewritten responses. *Journal of Experimental Psychology: Human Perception and Performance, 24,* 978–992. http://dx.doi.org/10.1037/0096-1523.24.3.978

Lohnas, L. J., Polyn, S. M., & Kahana, M. J. (2015). Expanding the scope of memory search: Modeling intralist and interlist effects in free recall. *Psychological Review, 122,* 337–363. http://dx.doi.org/10.1037/a0039036

McClelland, J. L., & Rumelhart, D. E. (1981). An interactive activation model of context effects in letter perception: Part 1. An account of basic findings. *Psychological Review, 88,* 375–407. http://dx.doi.org/10.1037/0033-295X.88.5.375

Mewhort, D. J. K., Merikle, P. M., & Bryden, M. P. (1969). On the transfer from iconic to short-term memory. *Journal of Experimental Psychology, 81,* 89–94. http://dx.doi.org/10.1037/h0027467

Morton, N. W., & Polyn, S. M. (2016). A predictive framework for evaluating models of semantic organization in free recall. *Journal of Memory and Language, 86,* 119–140. http://dx.doi.org/10.1016/j.jml.2015.10.002

Murdock, B. B. (1995). Developing TODAM: Three models for serial-order information. *Memory & Cognition, 23,* 631–645. http://dx.doi.org/10.3758/BF03197264

Murdock, B. B. (1997). Context and mediators in a theory of distributed associative memory (TODAM2). *Psychological Review, 104,* 839–862. http://dx.doi.org/10.1037/0033-295X.104.4.839

Newell, A. (1990). *Unified theories of cognition*. Cambridge, MA: Harvard University Press.

Nipher, F. E. (1878). On the distribution of errors in numbers written from memory. *Transactions of the Academy of Science of St. Louis, 3,* ccx–ccxi.

Norman, D. A. (1981). Categorization of action slips. *Psychological Review, 88,* 1–15. http://dx.doi.org/10.1037/0033-295X.88.1.1

Oberauer, K. (2003). Understanding serial position curves in short-term recognition and recall. *Journal of Memory and Language, 49,* 469–483. http://dx.doi.org/10.1016/S0749-596X(03)00080-9

Oberauer, K., & Lin, H. Y. (2017). An interference model of visual working memory. *Psychological Review, 124,* 21–59. http://dx.doi.org/10.1037/rev0000044

Page, M. P., & Norris, D. (1998). The primacy model: A new model of immediate serial recall. *Psychological Review, 105,* 761–781. http://dx.doi.org/10.1037/0033-295X.105.4.761-781

Palmer, C., & Pfordresher, P. Q. (2003). Incremental planning in sequence production. *Psychological Review, 110,* 683–712. http://dx.doi.org/10.1037/0033-295X.110.4.683

Palminteri, S., Wyart, V., & Koechlin, E. (2017). The importance of falsification in computational cognitive modeling. *Trends in Cognitive Sciences, 21,* 425–433. http://dx.doi.org/10.1016/j.tics.2017.03.011

Pelli, D. G., Farell, B., & Moore, D. C. (2003). The remarkable inefficiency of word recognition. *Nature, 423,* 752–756.

Polyn, S. M., Norman, K. A., & Kahana, M. J. (2009). A context maintenance and retrieval model of organizational processes in free recall. *Psychological Review, 116,* 129–156. http://dx.doi.org/10.1037/a0014420

Psychology Software Tools, Inc. (2012). E-prime 2.0 [Computer software]. Pittsburgh, PA: Author.

Ranschburg, P. (1902). Uber hemmung gleichzeitiger reizwirkungen [On inhibition of simultaneous stimulus effects]. *Zeitschrift fiir Psychologic und Physiologic der Sinnesorgane, 30,* 39–86.

Ratcliff, R. (1978). A theory of memory retrieval. *Psychological Review, 85,* 59–108. http://dx.doi.org/10.1037/0033-295X.85.2.59

Ratcliff, R. (1981). A theory of order relations in perceptual matching. *Psychological Review, 88,* 552–572. http://dx.doi.org/10.1037/0033-295X.88.6.552

Ratcliff, R. (2018). Decision making on spatially continuous scales. *Psychological Review, 125,* 888–935. http://dx.doi.org/10.1037/rev0000117

Ratcliff, R., Gomez, P., & McKoon, G. (2004). A diffusion model account of the lexical decision task. *Psychological Review, 111,* 159–182. http://dx.doi.org/10.1037/0033-295X.111.1.159

Rosenbaum, D. A., Loukopoulos, L. D., Meulenbroek, R. G. J., Vaughan, J., & Engelbrecht, S. E. (1995). Planning reaches by evaluating stored postures. *Psychological Review, 102,* 28–67. http://dx.doi.org/10.1037/0033-295X.102.1.28

Rosenbaum, D. A., Meulenbroek, R. J., Vaughan, J., & Jansen, C. (2001). Posture-based motion planning: Applications to grasping. *Psychological Review, 108,* 709–734. http://dx.doi.org/10.1037/0033-295X.108.4.709

Rumelhart, D. E. (1970). A multicomponent theory of the perception of briefly exposed visual displays. *Journal of Mathematical Psychology, 7,* 191–218. http://dx.doi.org/10.1016/0022-2496(70)90044-1

Rumelhart, D. E., & Norman, D. A. (1982). Simulating a skilled typist: A study of skilled cognitive-motor performance. *Cognitive Science, 6,* 1–36. http://dx.doi.org/10.1207/s15516709cog0601_1

Salthouse, T. A. (1986). Perceptual, cognitive, and motoric aspects of transcription typing. *Psychological Bulletin, 99,* 303–319. http://dx.doi.org/10.1037/0033-2909.99.3.303

Santee, J. L., & Egeth, H. E. (1980). Interference in letter identification: A test of feature-specific inhibition. *Perception & Psychophysics, 27,* 321–330. http://dx.doi.org/10.3758/BF03206121

Schauerte, B., & Fink, G. A. (2010). Focusing computational visual attention in multi-modal human-robot interaction. *Proceedings International Conference of Multimodal Interaction*. Advance online publication. http://dx.doi.org/10.1145/1891903.1891912

Schneider, D. W., & Logan, G. D. (2005). Modeling task switching without switching tasks: A short-term priming account of explicitly cued performance. *Journal of Experimental Psychology: General, 134,* 343–367. http://dx.doi.org/10.1037/0096-3445.134.3.343

Schneider, D. W., & Logan, G. D. (2009). Selecting a response in task switching: Testing a model of compound cue retrieval. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 35,* 122–136. http://dx.doi.org/10.1037/a0013744

Seidenberg, M. S., & McClelland, J. L. (1989). A distributed, developmental model of word recognition and naming. *Psychological Review, 96,* 523–568. http://dx.doi.org/10.1037/0033-295X.96.4.523

Shen, S., & Ma, W. J. (2019). Variable precision in visual perception. *Psychological Review, 126,* 89–132. http://dx.doi.org/10.1037/rev0000128

Shibuya, H., & Bundesen, C. (1988). Visual selection from multielement displays: Measuring and modeling effects of exposure duration. *Journal of Experimental Psychology: Human Perception and Performance, 14,* 591–600. http://dx.doi.org/10.1037/0096-1523.14.4.591

Shiffrin, R. M., & Cook, J. R. (1978). Short-term forgetting of item and order information. *Journal of Verbal Learning & Verbal Behavior, 17,* 189–218. http://dx.doi.org/10.1016/S0022-5371(78)90146-9

Smith, P. L. (2016). Diffusion theory of decision making in continuous report. *Psychological Review, 123,* 425–451. http://dx.doi.org/10.1037/rev0000023

Smith, P. L., & Little, D. R. (2018). Small is beautiful: In defense of the small-N design. *Psychonomic Bulletin & Review, 25,* 2083–2101. http://dx.doi.org/10.3758/s13423-018-1451-8

Solway, A., Murdock, B. B., & Kahana, M. J. (2012). Positional and temporal clustering in serial order memory. *Memory & Cognition, 40,* 177–190. http://dx.doi.org/10.3758/s13421-011-0142-8

Sperling, G. (1960). The information available in brief visual presentations. *Psychological Monographs, 74,* 1–29. http://dx.doi.org/10.1037/h0093759

Sperling, G. (1963). A model for visual memory tasks. *Human Factors, 5,* 19–31. http://dx.doi.org/10.1177/001872086300500103

Sternberg, S., Monsell, S., Knoll, R. L., & Wright, C. E. (1978). The latency and duration of rapid movement sequences: Comparisons of speech and typewriting. In G. Stelmach (Ed.), *Information processing in motor control and learning* (pp. 117–152). New York, NY: Academic Press. http://dx.doi.org/10.1016/B978-0-12-665960-3.50011-6

Surprenant, A. M., Kelley, M. R., Farley, L. A., & Neath, I. (2005). Fill-in and infill errors in order memory. *Memory, 13,* 267–273. http://dx.doi.org/10.1080/09658210344000396

Talmi, D., Lohnas, L. J., & Daw, N. D. (2019). A retrieved context model of the emotional modulation of memory. *Psychological Review, 126,* 455–485. http://dx.doi.org/10.1037/rev0000132

Tillman, G., Van Zandt, T., & Logan, G. D. (2020). Sequential sampling models without random between-trial variability: The racing diffusion model of speeded decision making. *Psychonomic Bulletin & Review*. Advance online publication. http://dx.doi.org/10.3758/s13423-020-01719-6

Tolman, E. C. (1948). Cognitive maps in rats and men. *Psychological Review, 55,* 189–208. http://dx.doi.org/10.1037/h0061626

Townsend, J. T. (1971). A note on the identifiability of parallel and serial processes. *Perception & Psychophysics, 10,* 161–163. http://dx.doi.org/10.3758/BF03205778

Townsend, J. T. (1990). Serial vs. parallel processing: Sometimes they look like tweedeldum and tweedeldee but they can (and should) be distinguished. *Psychological Science, 1,* 46–54. http://dx.doi.org/10.1111/j.1467-9280.1990.tb00067.x

Townsend, J. T., & Wenger, M. J. (2004). The serial-parallel dilemma: A case study in a linkage of theory and method. *Psychonomic Bulletin & Review, 11,* 391–418. http://dx.doi.org/10.3758/BF03196588

Usher, M., & McClelland, J. L. (2001). The time course of perceptual choice: The leaky, competing accumulator model. *Psychological Review, 108,* 550–592. http://dx.doi.org/10.1037/0033-295X.108.3.550

Whitney, C. (2001). How the brain encodes the order of letters in a printed word: The SERIOL model and selective literature review. *Psychonomic Bulletin & Review, 8,* 221–243. http://dx.doi.org/10.3758/BF03196158

Wolf, T. M., & Jahnke, J. C. (1968). Effects of intraserial repetition on short-term recognition and recall. *Journal of Experimental Psychology, 77,* 572–580. http://dx.doi.org/10.1037/h0026070

Yamaguchi, M., & Logan, G. D. (2014). Pushing typists back on the learning curve: Contributions of multiple linguistic units in the acquisition of typing skill. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 40,* 1713–1732. http://dx.doi.org/10.1037/xlm0000026

Young, R. K. (1961). The stimulus in serial verbal learning. *The American Journal of Psychology, 74,* 517–528. http://dx.doi.org/10.2307/1419662

Zorzi, M. (2000). Serial processing in reading aloud: No challenge for a parallel model. *Journal of Experimental Psychology: Human Perception and Performance, 26,* 847–856. http://dx.doi.org/10.1037/0096-1523.26.2.847

(*Appendices follow*)

Appendix A

Modeling Procedures

All data, model fits, and analyses and all programs for fitting, simulation, and analysis are available on the Open Science Framework at osf.io/f98kt.

Fitting the Models

Models were fit to the data with the fmincon routine in Matlab, which allows parameters to be constrained within specified ranges (usually 0–1). Fitting each model involved comparing the observed sequences participants produced to the presented sequences, and calculating the probability of encoding and retrieving each of the letters in the participant's report. On each trial, a set of stored contexts was created for the correct string and the evolution of the current context was determined by the participant's responses. If the participant made an error, that erroneous response was added to the current context and compared with the stored contexts, to calculate the likelihood of that error given the correct stored contexts. For correct responses, the probability of encoding was set equal to the integral of Equation 10 with the drift rate equal to 1 (distance $= 0$ and $\exp[-0] = 1$). The probability of correct retrieval was set equal to the integral of Equation 9 with drift rate equal to the dot product between the correct item and the reported item equal to 1, as the current context matches the stored context exactly. For error responses, the fitting program identified the error as an order error (the erroneous item was in the string but not in the retrieved position) or an intrusion error (the erroneous item was not in the string). The probability of an order error was calculated by setting the drift rate equal to the dot product between the current context containing erroneous item and stored context representing the item in the correct position. I assumed that items that produced order errors were encoded correctly. The probability of an encoding error was calculated by setting the drift rate equal to $\exp(-g \cdot \text{distance})$, where distance is the euclidean distance between the error letter and the correct letter. If an item was encoded incorrectly, I assumed it became part of the stored context for encoding the item and was retrieved correctly in order. The probability of encoding and the probability of retrieving were converted to logs and summed over stages (encoding, serial retrieval), items within a list, and lists to produce the overall negative log likelihood, which fmincon minimized. Each participant was fit separately and independently.

The fits took between 2 and 4 hr per participant, so I used the same set of starting values for the parameters for all participants. In the nondecrease models, the starting value for β was .5 and the starting value for g was .2. In the decrease models, the starting values for β_{Max} and g_{Max} were both .5 and the starting values for $\beta_{Decrease}$ and $g_{Decrease}$ were both .9. The best-fitting parameters and measures of goodness of fit for the best-fitting ED+SOD model are presented in Table A1 for Experiment 1 and in Table A2 for Experiment 2.

Simulating the Models

The models were simulated using the best-fitting parameters for each participant. The program stepped through the 576 trials a participant performed, encoding and retrieving each item in each list. A set of correct stored contexts was generated for each list, and the current context evolved by adding what was retrieved on step N to the current context for step $N + 1$. Each list was simulated 1,000 times, producing 576,000 trials for analysis. Summary statistics (list accuracy, serial position effects, transposition gradients, contiguity effects, error magnitude, and error type) were calculated for the simulated data using the same routines I applied to the actual data.

(*Appendices continue*)

Table A1

Best-Fitting Parameter Values, Minimized Negative Log Likelihood, and BIC Scores for Each Participant for the Encoding Decrease Plus Serial Order Decrease (ED+SOD) Model in Experiment 1

Participant	β_{Max}	$\beta_{DecType}$	$\beta_{DecRecall}$	$\beta_{DecReport}$	g_{Max}	$g_{DecType}$	$g_{DecRecall}$	$g_{DecReport}$	Likelihood	BIC
1	0.5705	1.0000	0.9461	0.9122	0.3607	0.9663	0.9035	0.8349	3279.93	6588.17
2	0.5551	1.0000	0.9236	0.9150	0.3907	0.9712	0.8503	0.7603	4585.09	9198.49
3	0.5892	0.9928	0.9225	0.8949	0.3852	0.9619	0.8940	0.8409	2891.98	5812.27
4	0.6144	1.0000	0.9470	0.8982	0.3505	1.0000	0.9446	0.8853	1985.86	4000.02
5	0.5844	0.9825	0.9235	0.9126	0.3313	0.9570	0.9422	0.8727	2197.65	4423.60
6	0.5661	1.0000	0.9110	0.9131	0.3714	0.9716	0.8572	0.8035	4136.25	8300.81
7	0.5638	1.0000	0.9120	0.9086	0.3450	0.9787	0.8934	0.8235	4440.89	8910.09
8	0.5759	1.0000	0.9107	0.8767	0.3531	0.9658	0.8764	0.8203	4638.98	9306.27
9	0.5834	1.0000	0.8999	0.9159	0.3843	0.9833	0.8362	0.7741	4200.87	8430.04
10	0.5532	1.0000	0.9236	0.8987	0.3243	0.9786	0.9060	0.8451	4288.77	8605.85
11	0.6089	1.0000	0.9199	0.9020	0.3664	0.9965	0.8851	0.8445	3432.79	6893.89
12	0.5799	1.0000	0.9609	0.8928	0.3329	0.9712	0.9443	0.8305	3917.41	7863.12
13	0.5568	1.0000	0.9085	0.8985	0.3699	0.9641	0.8893	0.8041	3794.57	7617.45
14	0.5579	1.0000	0.9036	0.8831	0.3426	0.9863	0.8971	0.8323	4266.39	8561.08
15	0.5860	1.0000	0.9223	0.9164	0.3937	0.9774	0.8804	0.8204	3307.51	6643.33
16	0.5639	1.0000	0.9243	0.9250	0.3836	0.9756	0.8535	0.8349	3610.7	7249.69
17	0.5934	1.0000	0.9275	0.9165	0.3351	0.9778	0.8878	0.8458	3871.13	7770.57
18	0.5354	1.0000	0.9094	0.8979	0.3671	0.9630	0.8747	0.8208	3869.51	7767.33
19	0.5881	1.0000	0.9002	0.8962	0.3772	0.9624	0.8398	0.7723	4652.23	9332.76
20	0.6394	0.9955	0.9453	0.9165	0.4265	0.9837	0.9142	0.8271	1843.13	3714.57
21	0.6071	1.0000	0.9612	0.9215	0.3805	0.9906	0.9315	0.8740	2099.87	4228.04
22	0.5108	1.0000	0.9134	0.9051	0.3172	1.0000	0.8627	0.8537	6676.46	13381.18
23	0.5206	1.0000	0.9228	0.8992	0.3274	1.0000	0.8625	0.7952	5871.33	11770.94
24	0.5821	1.0000	0.9606	0.9291	0.3511	0.9766	0.9240	0.8463	1623.64	3275.58
M	0.5744	0.9988	0.9250	0.9061	0.3612	0.9775	0.8896	0.8276	3728.46	7485.21
SD	0.0291	0.0039	0.0189	0.0129	0.0267	0.0130	0.0324	0.0313	1224.32	2448.63

Note. BIC = Bayesian information criterion.

(*Appendices continue*)

Table A2
Best-Fitting Parameter Values, Minimized Negative Log Likelihood, and BIC Scores for Each Participant for the Encoding Decrease Plus Serial Order Decrease (ED+SOD) Model in Experiment 2

Participant	β_{Max}	$\beta_{DecType}$	$\beta_{DecRecall}$	$\beta_{DecReport}$	g_{Max}	$g_{DecType}$	$g_{DecRecall}$	$g_{DecReport}$	Likelihood	BIC
1	0.5718	0.9997	0.9673	0.9300	0.3758	1.0000	0.8870	0.8380	3169.46	6367.23
2	0.6013	0.9979	0.9807	0.9445	0.4083	0.9687	0.9560	0.8702	1705.51	3439.31
3	0.5454	0.9923	0.9626	0.9531	0.3066	0.9533	0.9556	0.8939	2555.88	5140.06
4	0.5716	1.0000	0.9838	0.9275	0.3526	1.0000	0.9407	0.8375	3021.26	6070.83
5	0.5548	1.0000	0.9659	0.9602	0.3268	0.9855	0.9298	0.8974	3091.18	6210.66
6	0.6040	1.0000	0.9560	0.9130	0.3685	1.0000	0.8635	0.7305	4081.27	8190.85
7	0.5926	1.0000	0.9422	0.8962	0.3430	0.9693	0.9400	0.8794	3105.55	6239.41
8	0.5427	1.0000	0.9622	0.9318	0.3751	0.9463	0.8697	0.7709	4407.43	8843.17
9	0.5837	1.0000	0.9420	0.9126	0.3419	0.9999	0.9346	0.8676	3247.73	6523.77
10	0.5197	1.0000	0.9365	0.9198	0.3252	0.9860	0.8855	0.8832	4392.26	8812.83
11	0.5430	0.9999	0.9799	0.9997	0.3709	0.9922	0.9688	0.9422	1738.90	3506.10
12	0.5167	1.0000	0.9594	0.9152	0.3669	0.9893	0.9051	0.8564	3833.64	7695.59
13	0.5959	1.0000	0.9874	0.9495	0.3827	0.9858	0.9943	0.9223	1342.40	2713.11
14	0.5495	1.0000	0.9811	0.9450	0.3766	0.9579	0.9306	0.8876	2604.41	5237.13
15	0.5303	1.0000	0.9756	0.9487	0.3433	0.9449	0.9181	0.8758	3875.36	7779.03
16	0.5755	0.9999	0.9701	0.9408	0.3667	0.9634	0.9695	0.8846	2271.18	4570.67
17	0.5387	1.0000	0.9240	0.9030	0.3111	1.0000	0.9089	0.8897	4263.29	8554.89
18	0.5767	1.0000	0.9615	0.9351	0.3540	1.0000	0.9483	0.9067	2237.30	4502.91
19	0.5643	1.0000	0.9855	0.9619	0.4081	1.0000	0.9105	0.8783	1908.26	3844.83
20	0.5719	1.0000	0.9767	0.9530	0.4019	0.9716	0.9151	0.8476	2409.74	4847.78
21	0.4968	1.0000	0.9679	0.9255	0.3420	0.9805	0.9127	0.8074	4447.47	8923.25
22	0.5726	1.0000	0.9330	0.8698	0.3739	1.0000	0.9184	0.8635	3140.96	6310.23
23	0.5625	1.0000	0.9657	0.9390	0.3668	0.9909	0.9082	0.8407	2988.79	6005.89
24	0.5237	0.9985	0.9805	0.9601	0.3737	0.9668	0.9145	0.8829	2973.27	5974.84
M	0.5586	0.9995	0.9645	0.9348	0.3609	0.9813	0.9244	0.8648	3033.85	6096.01
SD	0.0285	0.0016	0.0177	0.0264	0.0274	0.0186	0.0318	0.0458	917.57	1835.14

(*Appendices continue*)

Appendix B

Parameter Recovery

I performed a limited parameter recovery study to evaluate the possible use of the model as a measurement tool. For each experiment, I simulated CRU with the best-fitting parameters from the (best fitting) ED+SOD model for each participant, running it on the same 576 trials as the participant (one run of the simulation per trial) to generate a sequence of about 3,456 responses. Then I fit the model to the simulated sequence to estimate parameters, just as I fit the actual data. I compared the parameters from fits to the data with parameters from fits to the simulations across participants in each experiment, assessing the agreement with correlation and *rmsd*. I analyzed the experiments separately because their procedures were different (Experiment 1 manipulated list length; Experiment 2 manipulated within-list repetition). The same ED+SOD model is evaluated in both experiments, so the two analyses are separate replications of parameter recovery for that model.

Figure B1 shows that the mean values from fits to the data and the fits to the simulations were very close for all eight parameters in each experiment. It also shows scatterplots for the eight parameters, with each parameter falling in an appropriate region of the scale. Figure B2 shows scatterplots for each parameter separately, focusing on a smaller range of values so the patterns can be seen more clearly. The mean correlations and *rmsd*s for each parameter in each experiment are presented in Table B1.

In both experiments, the agreement between fits to the data and fits to the simulations was reasonably good. The correlations in Table B1 are respectable, and the parameters whose correlations were low in Experiment 1 produced higher correlations in Experiment 2 ($\beta_{DReport}$ and g_{DType}). The *rmsd*s were small. However, the fits to the simulations underpredicted β_{Max} and g_{Max} from the fits to the data. β_{DType} and g_{DType} clustered tightly at values near 1.0. A broader range may give a clearer picture of their recovery. The fits to the simulations overestimated g_{DType}, $g_{DRecall}$, and $g_{DReport}$ from the fits to the data. These misfits were relatively small but deserve further attention.

The parameter recovery study is limited in that it used a relatively narrow range of parameter values (the range produced in fits to participants' data) and a relatively small number of simulations (24 per experiment). However, it does assess the CRU's ability to recover parameters given the range of parameters and number of participants one would see in a typical experiment and in the present experiments.

(*Appendices continue*)

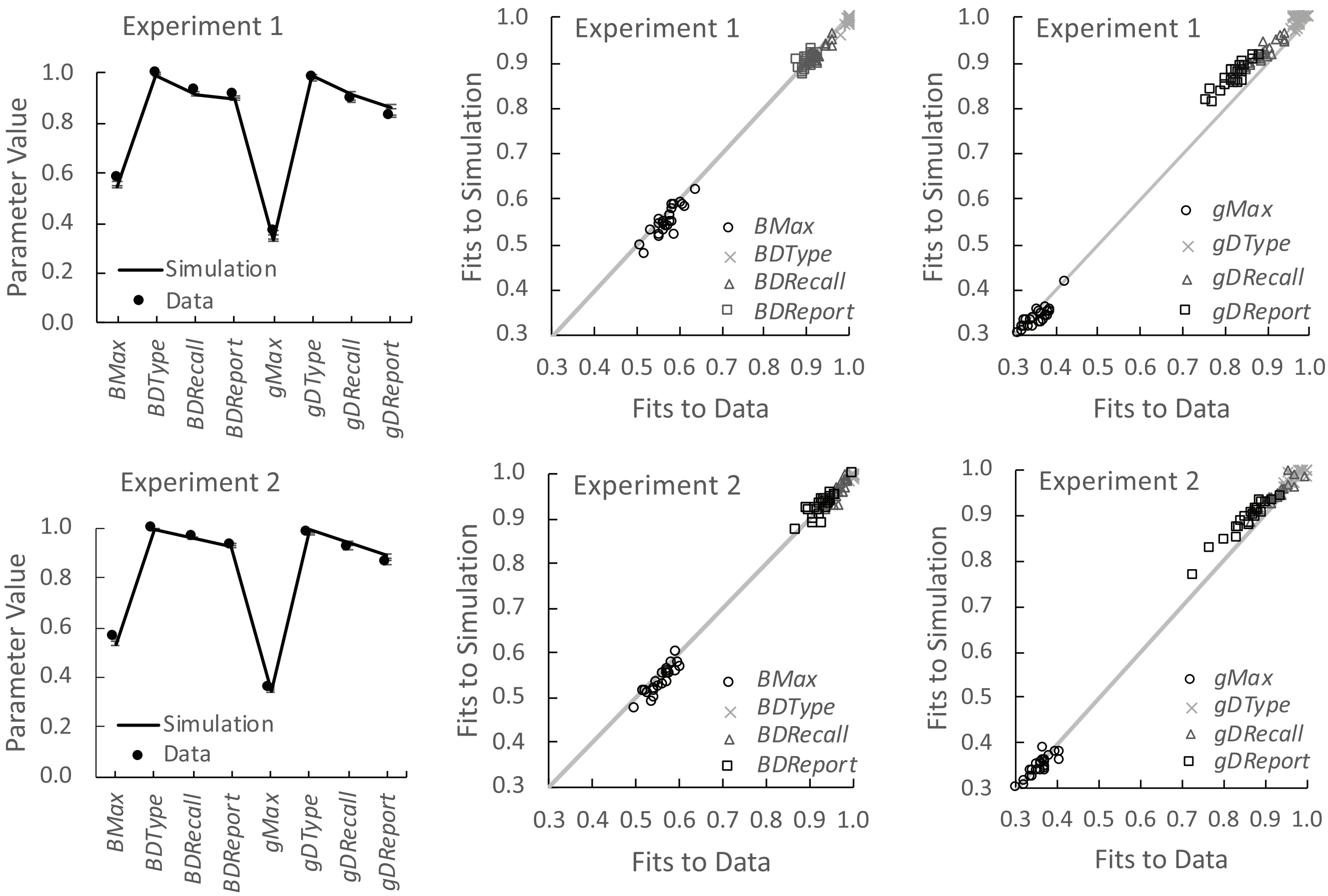


Figure B1. Top panels: Experiment 1. Bottom panels: Experiment 2. Left panels: Mean parameter values across participants for fit to the data and fits to the simulation. Middle panels: Plots of the serial order parameters (β . . .) from fits to the data versus fits to the simulations. Right panels: Plots of the perceptual encoding parameters (*g* . . .) from fits to the data versus fits to the simulations.

(*Appendices continue*)

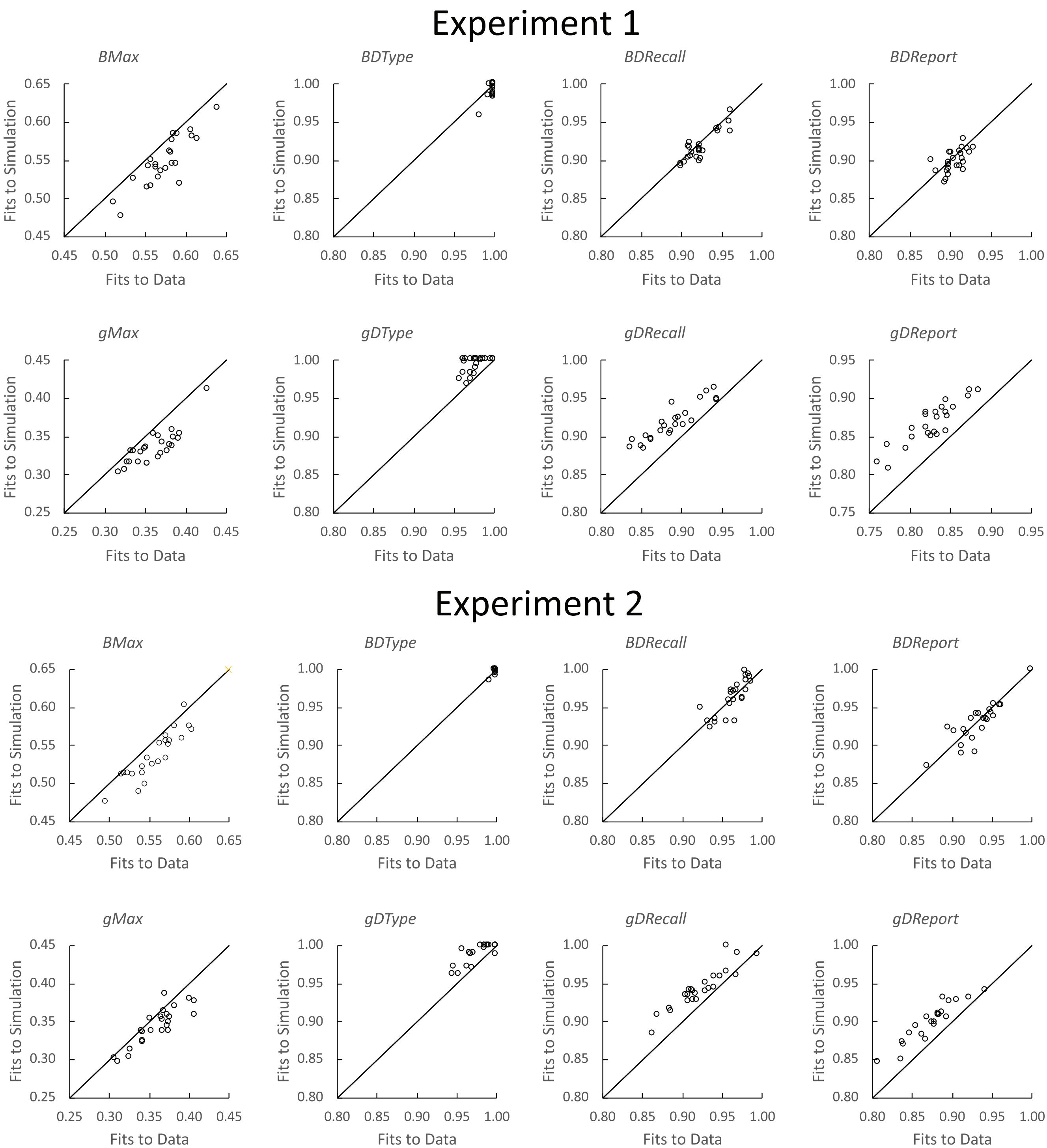


Figure B2. Plots of parameters estimated from fits to the data versus fits to the simulations. The top eight panels are from Experiment 1. The bottom eight panels are from Experiment 2. Each panel in the set of eight for each experiment represents a different panel. Axis values are scaled to a common range of 0.20 to provide more detail than is available in Figure B1.

(*Appendices continue*)

Table B1
Correlations and Root Mean–Squared Deviations (rmsd) Between Parameters Fitted to Participants' Data and Parameters Fitted to Simulations

Measure	β_{Max}	β_{DType}	$\beta_{DRecall}$	$\beta_{DReport}$	g_{Max}	g_{DType}	$g_{DRecall}$	$g_{DReport}$
Experiment 1								
r	0.8589	0.7782	0.8804	0.5680	0.8291	0.4665	0.9108	0.8887
rmsd	0.0317	0.0086	0.0119	0.0148	0.0300	0.0200	0.0313	0.0410
Experiment 2								
r	0.8956	0.7527	0.8031	0.8659	0.8715	0.8126	0.9222	0.9672
rmsd	0.0259	0.0028	0.0134	0.0144	0.0215	0.0146	0.0224	0.0286

Received October 2, 2019
Revision received June 15, 2020
Accepted June 23, 2020 ■