Journal of Experimental Psychology:
Learning, Memory, and Cognition
1983 Vol 9. No 3. 462–477

Structure and Process in Alphabetic Retrieval

David Klahr and William G. Chase
Carnegie-Mellon University

Eugene A. Lovelace
University of Virginia

Subjects were shown letters of the alphabet and asked to name the letter in the alphabet immediately before or after the probe letter. Subjects were also asked to describe how they accessed the alphabet and from what point in the alphabet they began their search. The structure of the reaction times (RTs) and subjects' reports on alphabetic entry points are accounted for by a model of alphabetic storage and retrieval. The model is a particularization of well-established general theories of the structure of long-term memory. The alphabet is represented as a two-level hierarchical list structure composed of six chunks that are in turn composed of from two to seven letters. Probe letters have direct access only to the name of the chunk in which they are embedded, and alphabetic access consists of serial, self-terminating search at each level. A model using only a single parameter for the time required to access the next element at either the chunk or letter level accounts for about 50% of the variance in RTs for our two experiments. A two-parameter model accounts for over 80% of the variance in previously published studies of covert and overt alphabet recitation.

In this article we propose a specific internal representation for order information of the alphabet, a detailed model of the processes used to access the representation, and an estimate of the speed of the model's basic processes. In order to address the general question of how familiar, long lists are stored and accessed in memory, we have focused on the alphabet. It is a common long list, with little explicit structure, learned very early and used throughout life.

How is the alphabet structured? There is evidence in the literature that long serial lists are stored hierarchically as subgroups in long-term memory (Anderson & Bower, 1973; Broadbent, 1975; Chase & Ericsson, 1982). The standard theoretical interpretation is that subgroup size is determined by the capacity of short-term memory because storage operations as well as retrieval and search operations are performed on chunks in short-term memory. Therefore, following Anderson and Bower (1973) and many others, we assume that long-term memory lists are stored as hierarchical sublists in a link-node structure such that sublists do not exceed the capacity of short-term memory.

How is the alphabet searched? Evidence originating with the seminal work of Sternberg (1967) strongly suggests that search for the location of an item in short-term memory is a serial, self-terminating process. Furthermore, due to constraints on short-term memory capacity, large lists are searched hierarchically: A serial, self-terminating search is first performed at the top level in the hierarchy followed by a serial search at the subgroup level (Naus, 1974; Naus, Glucksburg, & Ornstein, 1972).

Three major experimental procedures have been used in previous investigations of alphabet storage and access.

1. *Order.* The subject must decide whether a presented letter pair is in the correct alphabetical order (Lovelace & Snodgrass, 1971). Reaction time (RT) is measured from the onset of the letter pair until the subject presses a response button.

This work was supported in part by National Science Foundation Grant BSN 81-12743 to David Klahr and by Office of Naval Research Grants N00014-18-C-0335 and N00014-79-C-0661 to William G. Chase. Computer simulation was supported by National Science Foundation Grant BNS80-13051. We thank C. M. Powell, R. J. Brooks, V. Matsko, L. Swedko, M. R. Robinson, and M. Kinsky for assistance with data collection and statistical analysis and John Anderson and Jeff Shrager for comments on an earlier version of the article. A preliminary report on Experiment 1 was presented at the 19th annual meeting of the Psychonomic Society, San Antonio, Texas, November 1978.

Requests for reprints should be sent to David Klahr, Department of Psychology, Carnegie-Mellon University, Pittsburgh, Pennsylvania 15213.

2. *Recitation.* The subject is presented with a pair of letters and must recite the alphabet (covertly or overtly) from the first letter to the second letter (Landauer, 1962; Lovelace, Powell, & Brooks, 1973). RT (also called *recitation time*) is measured from the onset of the letter pair (at which point the subject starts alphabetic recitation) until the subject recites the final letter and pushes a response button.

3. *Forward or backward search.* The subject must say what comes *n*th after or before the presented letter. For example, "what comes *before* P," or "what comes four letters *after* K" (Lovelace & Spence, 1972). RT is measured from the onset of the probe letter until the subject's verbal response triggers a voice-actuated switch.

Regardless of the procedure used, if one looks at RTs as a function of alphabetic position of the stimulus, two findings consistently emerge: (a) At the aggregate level, stimuli near the end of the alphabet tend to require more processing time than stimuli near the beginning. (b) RTs are definitely nonmonotonic across the alphabet, and the fine structure of the RT pattern is similar across a variety of procedures.

Lovelace and Spence (1972) used a forward-search procedure and found an irregularly increasing RT as a function of alphabetic position (see bottom curve of Figure 1). The increase from the early portion of the alphabet to the final was substantial: RT for the first six letters, A–F, averaged 890 msec, and for the last six letters, T–Y, averaged 1,180 msec. Furthermore, there are several prominent and reliable "peaks" and "valleys." For example, it takes subjects almost a half a second longer to name the letter that follows K than to name the letter that follows M. Also shown in Figure 1 are the results from an earlier study by Lovelace and Snodgrass (1971) using the order procedure. Notice that even with this procedural change the RT patterns are very similar (average $r = .83$).

Lovelace and his colleagues proposed two possible explanations for the increasing RTs across the alphabet. One possibility is that there are lower associative strengths between adjacent letters near the end of the alphabet and these weaker associative strengths lead to longer RTs. The other possibility is that the

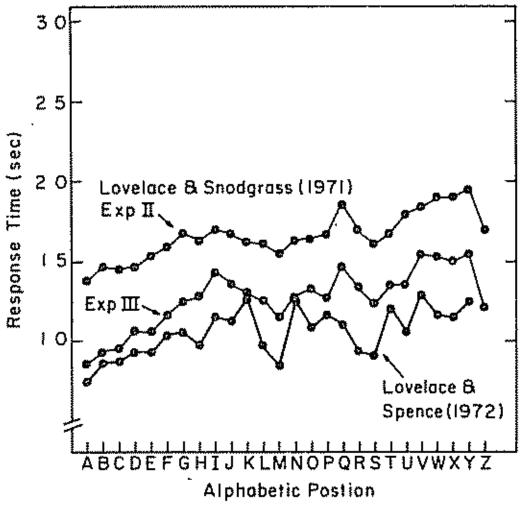

Figure 1. Mean response times in forward search and order decisions (The bottom curve is from "Reaction Times for Naming Successive Letters of the Alphabet" by E. A Lovelace and W. A. Spence, *Journal of Experimental Psychology,* 1972, *94,* 231–233 Copyright 1972 by American Psychological Association. Reprinted by permission. The top two curves are results of two experiments from "Decision Times for Alphabetic Order of Letter Pairs" by E. A. Lovelace and R. D. Snodgrass, *Journal of Experimental Psychology,* 1971, *88,* 258–264. Copyright 1971 by American Psychological Association. Reprinted by permission.)

interletter associative strengths are equal throughout the alphabet but there is differential access to particular letters. That is, there might be *preferred entry points* in the alphabet, with fewer entry points and/or slower access to such entry points toward the end of the alphabet. This would lead, on the average, to longer search sequences (and higher RTs) for probes near the end of the alphabet.

In order to discriminate between these two possibilities, Lovelace et al. (1973) used the recitation procedure and varied the number of letters processed after accessing the probe letter. They found that the recitation speed did not decrease toward the end of the alphabet (see Figure 2), and they concluded that the longer RTs at the end of the alphabet do not come from lower associative strengths. Rather, these longer RTs come from a greater difficulty of *entering* the alphabet near the end (cf. Hovancik, 1975).

Several questions remain concerning the structure and processing of the alphabet.

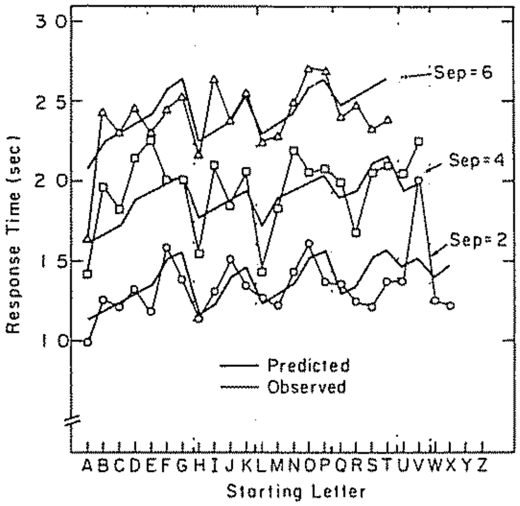

Figure 2. Mean response times for forward recitation at separations of two, four, and six letters from initial to final letter (From "Alphabetic Position Effects in Covert and Overt Alphabetic Recitation Times" by E. A. Lovelace, C. M. Powell, and R. J. Brooks, *Journal of Experimental Psychology,* 1973, *99,* 405–408. Copyright 1973 by American Psychological Association. Reprinted by permission. Also shown are predictions from a model with chunk boundaries shown as vertical dotted lines.)

1. Although the Lovelace et al. (1973) study supports the notion of preferred entry points, it does not provide any *direct* evidence. The present investigation demonstrates such entry points.
2. The model, as stated thus far, is largely intuitive, with no specification of the representation or processes involved. This article describes a detailed model written as a computer simulation, with model parameters estimated from the data.
3. The fine structure of the RT patterns has never been accounted for. The present model is designed to predict the RT for each alphabetic position in both forward and backward search tasks.

A Theory of Alphabetic Access

In this section, we propose an information-processing theory of these alphabetic search tasks. The theory is consistent with generally accepted ideas about the structure of long-term memory. In a subsequent section, we will describe a particular computer simulation model derived from the theory.

On the *after* task, in which the letter following the probe is sought, the theory works as follows:

First, search from the beginning of the alphabet for a subgroup containing the probe item.

Having found the appropriate subgroup, search from the beginning of the subgroup until the probe is found.

Once the probe is found, get the next item in the subgroup and output that value.

If the end of the subgroup is encountered, then get the next subgroup and output the first item.

On the *before* task, in which the letter preceding the probe is sought,

Start by searching for the subgroup containing the probe.

When the appropriate subgroup is located, search through the subgroup while keeping track of the prior position.

When the probe is found, get the prior item and output it.

If the probe is the first letter of the subgroup, so that there is no prior item, get the prior subgroup, search to the end of that subgroup and output the last item.

This theory makes several specific predictions about search times in the alphabet. It handles the well-known effect of increasing RTs across the alphabet by assuming a serial, self-terminating search of the subgroups at the top level in the hierarchy. The fine-grain structure of peaks and valleys in RTs, according to this theory, is due to the relative accessibility of letters within subgroups: Earlier items within a subgroup are detected faster by the serial, self-terminating search, and the really slow times should occur when the probe and the next (or prior) letter are separated by a subgroup boundary. Thus, according to this theory, local maxima in RTs should occur at the end of a subgroup for the *after* task and at the beginning of a subgroup for the *before* task. Local minima should occur at the beginning of a subgroup for the *after* task and at the second element in a subgroup for the *before* task.

Since none of the previous alphabetic search studies used the *before* task,[1] we conducted an experiment to test these predictions, using both *before* and *after* tasks.

[1] Weber, Cross, and Carlton (1968) used both forward and backward search, but the only letter sequence they used consisted of the five letters E through I.

Experiment 1

Method

Subjects Twelve adult subjects from introductory psychology courses at Carnegie-Mellon University participated in this experiment.

Materials The stimulus letters were all uppercase, helvetica medium, 38.1 mm high, black on a white 5 × 8 in. (12.7 × 20.3 cm) card. At a viewing distance of 45.7 cm, the letters subtended a visual angle of 4.78°.

Procedure. Each subject received each of two experimental conditions. In the *before* condition, there were five successive presentations of a randomized set of 25 stimulus letters B to Z, for which the subject was to name the preceding letter. In the *after* condition, there were five successive presentations of a randomized set of 25 stimulus letters A to Y, for which the subject was to name the following letter. The order of conditions was counterbalanced across subjects.

On each trial, the subject looked into a tachistoscope and pressed a button to start the trial when he or she was ready. After 500 msec, the probe letter appeared in the center of the field, and the subject named the preceding or following letter in the alphabet as quickly as possible. A standard timer, connected to a voice-actuated relay, recorded RT to the nearest 100th of a second. Error trials were rerun randomly within the remaining trials.

Results

For each subject, the median RT (out of five trials) for correct responses to each letter was determined for the *before* and *after* tasks. Figure 3 presents the means (over the 12 subjects) for these median RTs as a function of the alphabetic position of the stimulus letter.

The most striking feature of these curves is their agreement with the predicted peaks and valleys of the *before* and *after* curves. For example, the first prominent local maximum on the *after* curve occurs at G and is followed by a local minimum at H. For the *before* curve, the maximum and minimum points occur one letter later, at H and I, respectively. Strong boundary effects also occur between K and L and between P and Q. Other boundary effects are a bit weaker, and there are some anomalies toward the end of the alphabet. The model suggests that we should get high correlations between *after* times and lagged *before* times. If we exclude (Y, Z), the product-moment correlation for these lagged times is .79. (For unlagged times, it is only .2.)

It is also worth noting that the peaks and valleys of the *before* curve are much more pronounced than those of the *after* curve, and

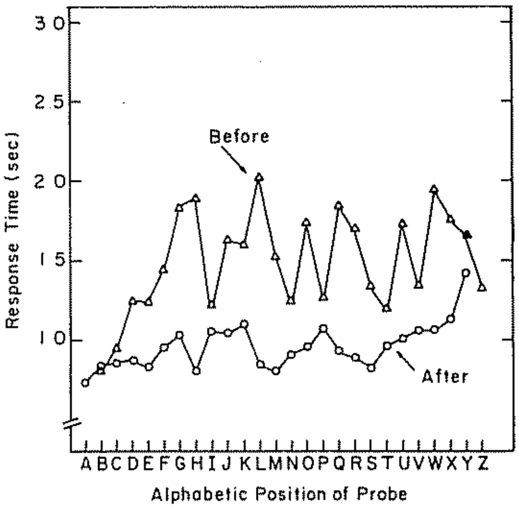

Figure 3 Mean of median response times for forward (*after*) and backward (*before*) search (Experiment 1).

the *before* RTs are longer than the *after* RTs. This result is consistent with the subjective impression that the *before* task is the more difficult of the two, and it is a direct consequence of the postulated alphabetic structure: Elements are linked only to their successors. Thus, locating the prior letter in a list requires keeping track of both the prior letter and the prior chunk. In a subsequent section of the article, we make more precise processing distinctions between the *before* and *after* tasks when we develop parameter estimates for the model.

Experiment 2

Although these results are consistent with the predictions from the model, our decision about how to segment the alphabet is based on an informal post hoc analysis of local extreme points. The fact that this segmentation is consistent with the phrasing in the common nursery school "Alphabet Song"[2] pro-

[2] There are two principal variants of the "Alphabet Song" commonly used in nursery schools in the United States. Both of them are sung to the tune of "Twinkle, Twinkle, Little Star," and are in agreement with respect to the G–H, K–L, P–Q, and V–W boundaries (cf. Betail, 1947). However, they differ in how they segment the letters QRSTUV: one version sings "QRS and TUV," while the other sings "QRST, U and V" (corresponding to the "up above the world so high" part of "Twinkle")

vides some additional basis for believing it to be correct, but we have no direct independent evidence that it is the segmentation used by our subjects. In the second experiment, we asked subjects to report directly their "entry points," if any. The independent assessment of the segmentation allowed us to perform a more rigorous evaluation of the model.

Method

Subjects. Thirty students from introductory psychology classes at the University of Virginia participated in this experiment.

Materials. Each subject was presented six sets of slides. Each set contained one slide of each of the 26 letters of the alphabet (Artype No. 1407 capitals). Sequencing of the letters within a slide set was random with the restriction that no letter followed the same letter in any two sets. On half the occurrences of a given letter the subject was required to name the preceding letter of the alphabet; on the remaining occasions the task was to name the following letter. (On all six occurrences of the letter A the task was to name the following letter, and on all occurrences of Z, to name the preceding letter.)

Procedure. Each subject was seated before a translucent screen in a sound-deadened chamber; the experimenter and all apparatus for stimulus presentation and response timing were outside the chamber. Single letters were back projected onto the translucent screen, and the subject was to say aloud, as quickly as possible, either the preceding or the following letter of the alphabet. Each stimulus letter was preceded by a warning buzzer and by one of the two different colored lights marked PRECEDING and FOLLOWING, which informed the subject of the type of decision required on that trial. A photocell on the back of the screen activated a Lafayette Model 5721 digital timer when the letter came on the screen. The subject's spoken response activated a voice key that stopped the timer and advanced the projector to an opaque slide. The experimenter initiated each trial manually; the buzzer and light preceded the stimulus letter by approximately 1.5 sec. Error trials were not rerun. The stimulus letters were presented at a rate of about 12 per min.

The subjects were also instructed to tell the experimenter what they had done to think of the correct response after they had named the appropriate letter. These verbal reports were classified into three categories: (a) didn't have to do anything, the letter just occurred to me; (b) had to covertly recite a specifiable portion of the alphabet; or (c) had to do something, but not explicitly described as a recitation of a specific portion of the alphabet. Whenever subjects reported covert recitation of part of the alphabet, they were asked to indicate, if possible, the letter at which they began that recitation.

Results

Most subjects were able to maintain very high accuracy levels while operating with a speed set; overt errors of naming the wrong letter occurred on less than 1% of the trials. Voice-key equipment malfunctions and other errors account for data lost on about 1.5% of the trials.

For each individual the median RT for correct responses to each letter was determined separately for the *before* and *after* tasks. Figure 4 presents the means of these median RTs as a function of the stimulus letter presented. There is a high correspondence between the Lovelace and Spence (1972) times (see bottom curve in Figure 1) and the *after* condition in the present study ($r = .81$), although the times were both longer and more variable in the present study, perhaps because *before* and *after* tasks were mixed or perhaps because subjects were assigned the additional task of attending to and reporting their thought processes on each trial. Even so, the RTs from Experiment 2 were highly correlated with those from Experiment 1 ($r = .75$ for *after*; $r = .94$ for *before*).

Figure 5 shows the proportion of trials on which subjects reported having to do something (combined response categories b and c) in order to respond to each stimulus letter for *before* and *after*. This occurred much more frequently on *before* trials than on *after* trials, but the frequency in the latter case was substantial, indicating that even when required to name the next letter of the alphabet, subjects frequently could not immediately access that item. In addition, it can be seen that the relative frequency with which individuals needed to "do something" was related to position in the alphabet; this need occurred more frequently for letters near the end of the alphabet than for those near the beginning. These relative frequencies correlated highly with RTs on both *before* ($r = .80$) and *after* trials ($r = .90$) and with after RTs in the earlier data of Lovelace and Spence (1972; $r = .88$).

In most cases where individuals had to "do something," they reported covert recitation from a specifiable letter (90% for *after* decisions, and 95% for *before*). Figure 6 shows the relative proportions with which various letters of the alphabet were reported as the beginning point on those trials when covert recitation of a specific portion of the alphabet occurred. This plot provides clear

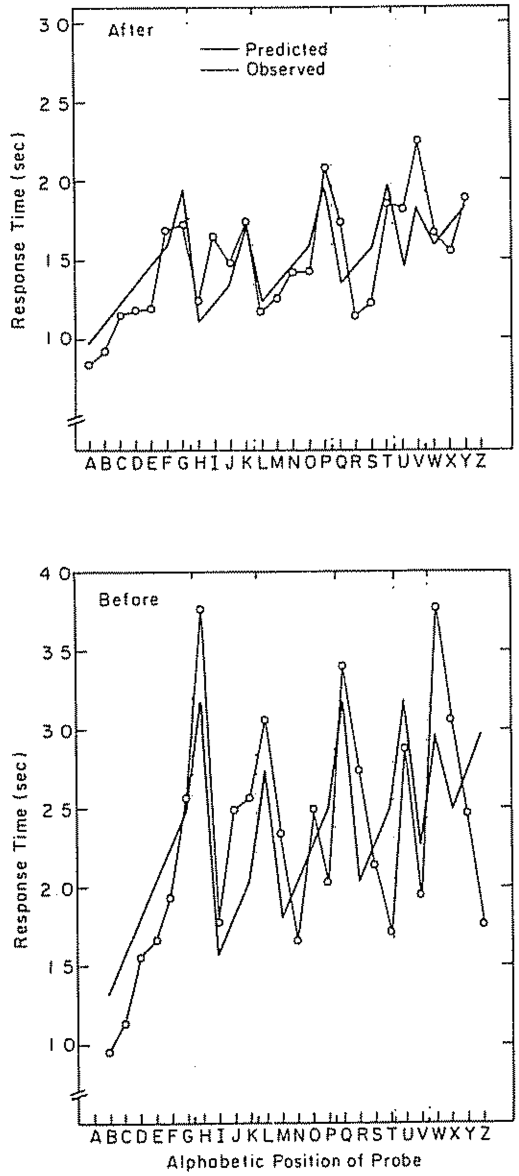

Figure 4 Observed and predicted response times from Experiment 2 as function of alphabetic position for *after* and *before* conditions (Chunk boundaries are shown as vertical dotted lines.)

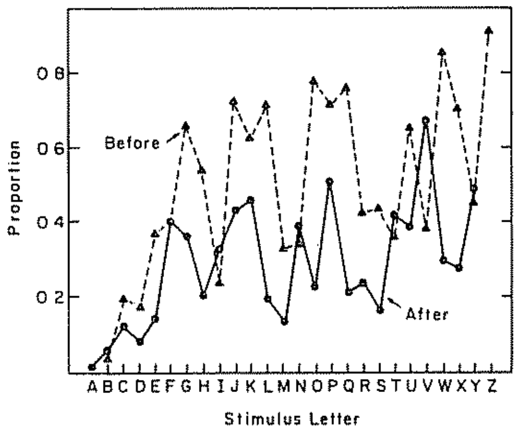

Figure 5 Proportion of trials on which subjects reported having to "do something" in order to respond to the probe in *before* and *after* conditions (Experiment 2).

evidence that there are preferred points of entry into the alphabet and that entry points are, to a considerable extent, shared by individuals. The deviation from a rectangular distribution (which would denote no preferred entry points) is clearly greater in early portions of the alphabet than in later portions. There are at least two plausible interpretations for this deviation. Stable preferred entry points may be less prevalent later in the alphabet, or alternatively, the location of stable preferred entry points may be more variable between subjects later in the alphabet.

The maximum values of this distribution provide an empirically based segmentation of the alphabet. Based on the peaks of the *before* curve in Figure 6, the alphabet appears to be segmented into chunks starting with the following letters: A, H, L, Q, U, X. This segmentation corresponds to one of the principal versions of the "Alphabet Song" (see Footnote 1), except that the last chunk of the song begins with W rather than X.

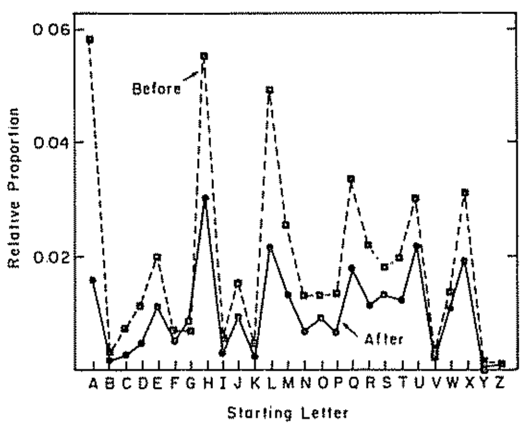

Figure 6 Relative proportion of trials on which specific letters were reported as starting points for covert recitation in Experiment 2.

Description and Evaluation of ALPHA: A Model of Alphabetic Access

In order to generate RT predictions from the model, we need to be more specific about its component processes and about how each process contributes to the overall RT. In this section we describe a computer simulation model, ALPHA, for the *before* and *after* tasks used in this study. First, we describe the data structure for the representation of the alphabet. Then we discuss the processes that operate on this structure and parameters associated with these processes. Next, we estimate the value of a single, key parameter for the *after* and *before* tasks. Finally, our model is extended to fit the recitation data of Lovelace et al. (1973) and Browman and O'Connell (1976).

Representation

Figure 7 illustrates the semantic memory representation used by ALPHA. This figure depicts two components of semantic memory separately: (a) The concept node, shown in the lower right and labeled "Letter Recognition," and (b) the *order* information of the alphabet, shown at the top of Figure 7 as a link-node structure. The concept node of a letter contains all the information directly associated with that letter, including its graphemic and phonemic properties, words beginning with the letter, and so on. Included in the concept node is the alphabetic location, represented here as the name of the chunk in the alphabet containing that letter.

We assume that when a letter of the alphabet is recognized, its concept node in semantic memory is activated. However, only a few properties of that node are normally activated, such as the graphemic and phonemic descriptions and other associated properties that are primed by the context. Upon recognition, a letter's alphabetic location is not normally activated. However, in the context of an alphabet-search task, we assume that the alphabetic location is primed,

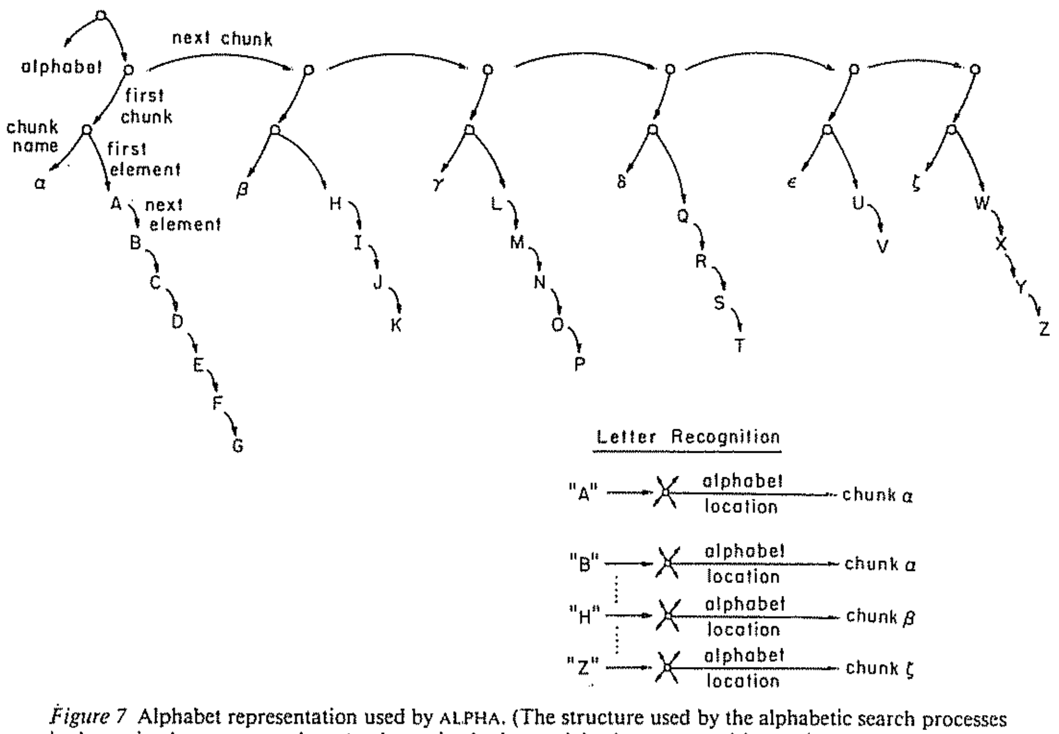

Figure 7 Alphabet representation used by ALPHA. (The structure used by the alphabetic search processes is shown in the upper portion. As shown in the lower right, letter recognition activates the letter node, which has links to many nodes associated with the letter. The only association explicitly represented here is the link to "alphabet location." The node for "A" is connected via the "alphabet location" link to "chunk α." "B" is also associated with α, "H" with β, and so on.)

so that when a probe letter is recognized, the name of the chunk containing its location is activated. The name of the chunk is then used to search the alphabetic structure.

In the hierarchical structure depicted at the top of Figure 7, the alphabet is stored as a list of chunks, with each chunk containing its name and a list of letters in their proper order. This list structure has the property that each element is linked to its successor; thus, each chunk is linked to the next chunk, and within each chunk, each letter is linked to the next. Lists are accessible only through their beginnings, and only forward search is possible.

The most important feature of this representation is that probe letters do not have direct access to their *nexts* or *priors*, instead, they have direct access only to the *name* of the chunk in which the probe is located. In this list representation, the alphabet can be entered directly only at chunk boundaries, at the head of each sublist within the hierarchy. In the alphabet-search task, when people use these entry points, they do not choose one at random. Rather, they tend to choose the one that is "just ahead" of the probe letter.

While this aspect of the model may, at first, seem nonintuitive, it is simply a formalization of the notion of preferred entry points. It has been long known that access to parts of well-learned series (songs, poems, telephone numbers) generally requires access to the beginning of the series or to the beginning of a subpart of the series (Müller & Pilzecker, 1900). This entry-point phenomenon is modeled in our theory by means of a hierarchical list representation in conjunction with a simple forward-search process.

The name of a chunk should not be thought of as a verbal label but rather as an *internal address* that links a letter's concept node to its alphabetic location This characterization is similar to the "control elements" postulated by Estes (1972) to represent the hierarchical nature of order information in serially learned lists. Simple chaining of associations is inadequate to account for chunking, for the types of serial order errors that occur and, in general, for the hierarchical organization of serial behavior (Lashley, 1951). Estes (1972) found it necessary to postulate the existence of a higher level node in order to account for the various phenomena of serial order within the framework of an association theory. Estes' theory gives a detailed proposal for building lists out of associations and is specifically designed to account for chunking and serial order errors. Our theory differs from Estes' in that we simply assume a list structure representation and that the heart of our theory involves search mechanisms for lists.

The Search Process

The basic mechanism underlying the search process is a simple NEXT operation that takes an element in semantic memory and activates its successor. Thus, in the alphabetic structure of Figure 7, search begins at the head of the alphabet, and each successive chunk is activated until the chunk containing the probe is found. At this point, each successive letter in the chunk is activated until the probe is located. Then, depending upon the task, the next or prior letter is located. In our model, the interesting complexities arise in how to get the prior letter with only a NEXT operator and how to handle NEXT operations across chunk boundaries.

Figure 8 provides flowcharts of ALPHA on the two tasks,[3] showing the basic steps of the program, with some of the detail suppressed for clarity of exposition. The *after* task (Figure 8[A]) is the simpler process because there is no need to keep track of the prior chunk or prior letter. In this flow diagram, the top loop characterizes the serial, self-terminating search at the top level for chunk C containing probe P. There are two ways to characterize this rather nonintuitive part of the model. One way is to assume that once a chunk is activated, a high-speed scan for presence is performed on the chunk to test for the presence of the probe (Sternberg, 1967). An alternative is to assume that each letter is associated with its location within the alphabet. Specifically, we assume that the probe activates the name of its associated chunk, and

[3] The program is written in FRANZ-LISP, a variant of MACLISP used at Carnegie-Mellon University. Listings of the program and sample runs are available from the first author. People with access to the ARPA network can contact KLAHR@CMUA

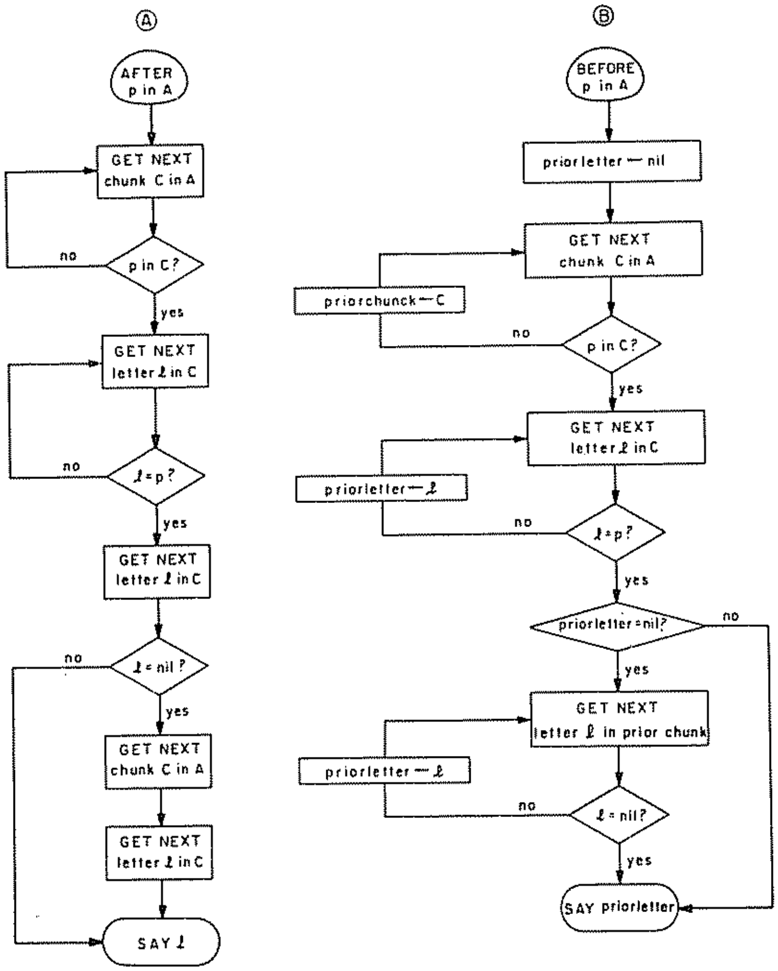

Figure 8 Flowchart for ALPHA, a model for searching the two-level hierarchical alphabetic structure shown in Figure 7. (A = *after*; B = *before*.)

the top-level search is a serial, self-terminating scan for the name. In our model, we have used the latter process, although in either case the important assumption is that letters of the alphabet are associated with their relative locations within the alphabet; exact location requires a hierarchical search process.

The next loop in the flowchart characterizes the serial, self-terminating search within the chunk, after the chunk containing the probe is found. When the probe is located, ALPHA gets the next letter and says it. The one complication occurs when the probe is the last letter in the chunk (depicted at the bottom of the flowchart). When this happens, ALPHA gets the next chunk and then gets the first letter of that chunk and says it.

The logic of the *before* task (Figure 8[B]) is very similar except that the search is slower because the prior chunk and prior letter are saved at each step. Also notice that when the probe is the first letter of a chunk, the process must scan from the beginning to the end of the prior chunk. Thus, the model predicts extreme peaks for these points.

Model Evaluation and Parameter Estimation

Model evaluation requires several steps:

1. Choose a segmentation for the alphabet.

2. Present ALPHA with each letter of the alphabet for the *before* and *after* tasks, and have it search the hierarchical structure with the chosen segmentation.

3. Count the number of executions of the basic processes in ALPHA for both the *before* and *after* tasks with the chosen segmentation.[4]

4. Estimate, via regression analysis, the duration of the basic processes by regressing RT against number of basic operations predicted by ALPHA in both the *before* and *after* tasks

The predicted RTs from ALPHA depend on both the assumed segmentation of the alphabet and the parameters used to estimate the duration of each subprocess. Our first problem was to decide what segmentation to use. As noted earlier, the subject-defined segmentation of Experiment 2 (the peaks of the entry point frequency function of Figure 6) corresponds to the "Alphabet Song" segmentation, except for the last chunk boundary. (The subject-defined segmentation indicates that there is a W–X chunk boundary, whereas the "Alphabet Song" has a V–W chunk boundary.) To determine which chunk boundary to use in our model, we fit regression equations to RTs of Experiments 1 and 2 using both segmentations, and we consistently obtained slightly better fits with a V–W chunk boundary (by about 5% more variance accounted for in all our data sets). This better fit is due to the presence of consistently longer RTs for V in the *after* task and for W in the *before* task, clearly indicating a V–W chunk boundary. The more frequent use of X, rather than W, as a starting letter in the verbal reports of Experiment 2 (Figure 6) may be due to the existence of a special XYZ chunk in semantic memory. That is, ABC and XYZ may be special high-frequency phrases in peoples' lexical memories that intrude in their verbal reports, but the semantic memory structure actually used in the search task may contain the WXYZ chunk of the "Alphabet Song" segmentation, with chunks starting with A, H, L, Q, U, and W.[5]

Our next problem was to decide what parameters of ALPHA to use in estimating RTs. We finally decided on a linear regression on the number of executions of a single internal process: doing a NEXT operation on all the internal list structures in ALPHA.[6] That is, for each of the 25 probes of the *after* and *before* task, we computed the number of times that the model did a NEXT operation on any of its internal structures in order to find a response to the probe. We fitted the *after* and *before* tasks separately because we expected that the NEXT process in the *before* task would be slower due to the increased memory load imposed by saving prior chunks and prior letters. We made the simplifying assumption that the time required to do a NEXT at the chunk level is the same as that required to do a NEXT within a chunk. A two-parameter regression that fits separate parameters for NEXTs at the chunk level and NEXTs within chunks accounted for only about 5% more variance than the one-parameter fit. Thus, we felt that the simplest and most elegant regression was the one-parameter fit of NEXTs because, as a first approximation, it seems to capture the essential aspect of searching the alphabet, namely, getting the next element in a list structure.

Figure 9 shows a plot of RT versus the number of NEXT operations for the *before* and *after* tasks of Experiments 1 and 2, along with the straight-line predictions of the model. Table 1 contains the regression equations and the amount of variance accounted for by the model for each condition, as well as predictions for the data (*after* only) from the Lovelace and Spence (1972) experiment. The model, with a single parameter,[7] accounts for

[4] ALPHA is a completely deterministic model that actually carries out the postulated processes on a specific data structure. It represents errorless performance of the idealized subject. Note that this is not a Monte-Carlo simulation, in which repeated runs are made in order to discover the distributional properties of a complex stochastic model.

[5] Another decision we faced was which version of the "Alphabet Song" to use, the one with the T beginning the next-to-last chunk or the one with U. We settled on the latter one because it consistently gave slightly better fits to the data (about 5% more variance accounted for)

[6] This is a simple count of all CDRs used by all the functions in the LISP program. A CDR is the most basic LISP function for processing lists: It takes a list as input and outputs a new list consisting of the old list minus the first element.

[7] The intercept can be viewed as a "fitting" parameter, or as a "base" time that contains all the other processes of the model that are executed only once, independent of the number of NEXT operations. The intercept does not enter into the analysis of variance; the variance accounted for by the model is the variance about the mean of each condition that is accounted for by the *slope* of the regression line. Note that much more variance is accounted for by fitting a three-parameter model (two slopes and an intercept difference) to the *before* and *after* conditions combined: 76% of the variance in Experiment 1 and 69% of the variance in Experiment 2. The three-parameter models are identical to those shown in Figure 9 and Table 1; the difference lies in whether variance is

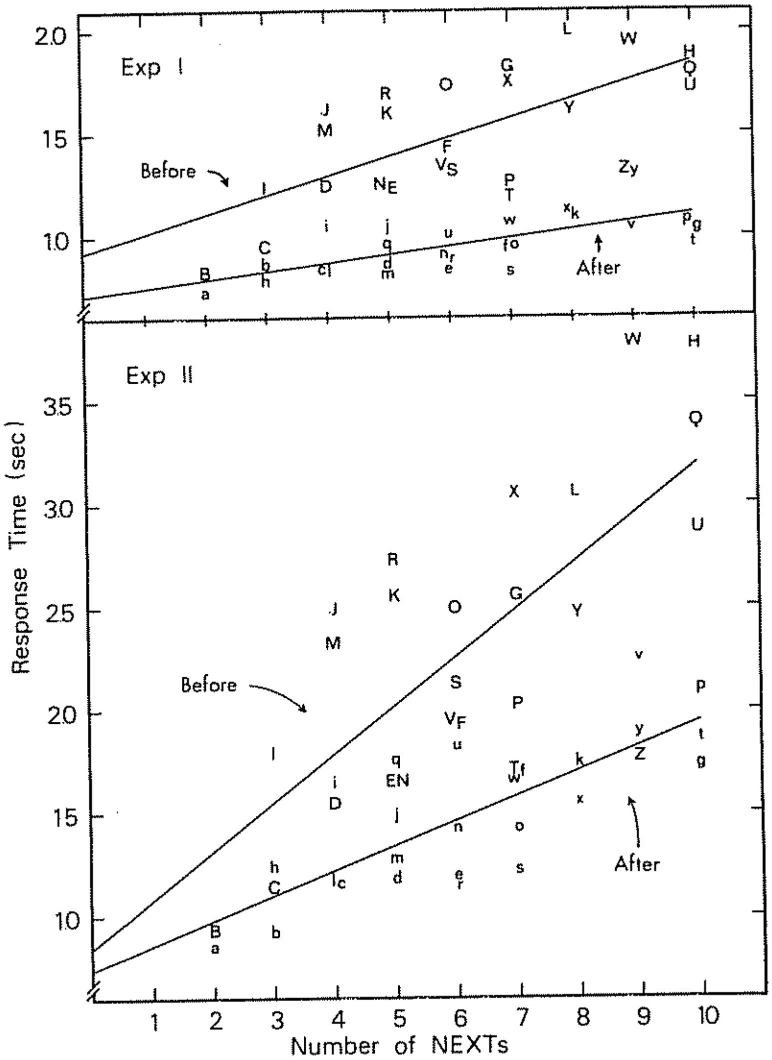

Figure 9. Observed and predicted response times (RTs) for the *before* and *after* conditions of Experiments 1 and 2 as a function of number of executions of ALPHA's basic processes (NEXTs). (Observed RTs are indicated for each probe letter, with capital letters for the *before* conditions and small letters for the *after* conditions. Predicted RTs are depicted by straight lines.)

about half the variance in each condition by assuming that search time is a linear function of the number of NEXT operations.

When RT is plotted as a function of alphabetic position, as in Figures 1 through 4, the most prominent features are the local extreme points caused by chunk boundary crossings. Predicted RTs and chunk boundaries have been inserted in Figure 4, which shows the results from both the *before* and *after* tasks of Experiment 2 (the results for Experiment 1 and Lovelace and Spence, 1972, are similar). ALPHA appears to capture these peaks and valleys quite well on both *after* and *before* tasks, although the first part of the alphabet is in closer correspondence with the model than the end of the alphabet.

---

computed about the mean of each condition separately (the one-parameter fit) or about the mean of the whole experiment (the three-parameter fit). Finally, it should be noted that if only alphabetic position is used as the independent variable without postulating a hierarchical search process (thus capturing just the linear trend across alphabetic position in Figures 3 and 4 without taking into account the peaks and valleys), much less variance is accounted for: Experiment 1 *before* = 12.6%, *after* = 35.5%; and Experiment 2 *before* = 15.6%, *after* = 37.9%.

Table 1
Best Fitting Equations (in msec) for ALPHA

Experiment	Equation	% variance accounted for
Experiment 1		
After	RT = 716 + 39.1 × N	46.3
Before	RT = 925 + 92.5 × N	45.4
Experiment 2		
After	RT = 743 + 120 × N	60.0
Before	RT = 863 + 233 × N	51.6
Lovelace & Spence (1972)		
After	RT = 779 + 43.5 × N	43.6
Lovelace, Powell, & Brooks (1973)		
Covert recitation	RT = 405 + 55.0 × N + 185 × S	86.5
Browman & O'Connell (1976)		
Overt recitation	RT = 1,067 + 66.3 × N + 313 × S	86.6
Covert recitation	RT = 575 + 53.9 × N + 220 × S	81.1

Note RT = reaction time; N = number of NEXTS; S = number of SAYS.

Another interesting deviation from ALPHA's predictions are the slight but consistent overpredictions for the first chunk, particularly for the *before* times for the first few letters. This might result from the alternative representation (mentioned earlier) for these items (the ABCs) that provide more direct access than the full process modeled by ALPHA.

At this point, we should comment on the magnitude of the parameters we have obtained. First, as expected, there is a substantial difference in the NEXT time for the *before* and *after* tasks. In the model (cf. Figure 8), this extra time corresponds to the additional step of saving the prior chunk and the prior letter. However, it is also plausible that the extra memory load involved in the *before* task also increases the time to do a NEXT. A second point worth noting about these parameters is that the time to do a NEXT is substantially longer for Experiment 2 than for either Experiment 1 or for the Lovelace and Spence (1972) experiment. This slower processing could be caused by either or both of two procedural differences: (a) the *after* and *before* tasks were mixed together in Experiment 2, and (b) subjects were asked to give rather extensive retrospective reports in Experiment 2.

To what extent do individual subjects have the same chunk boundaries as those determined by the aggregate analysis? At each chunk boundary, the model predicts the RT pattern shown in Figure 10. For the *after* condition, the RTs should reach a local maximum for the last item in a chunk (Figure 10, A_2). In addition, the first item in a chunk (A_3) should be faster than the penultimate item in the preceding chunk (A_1)—unless the

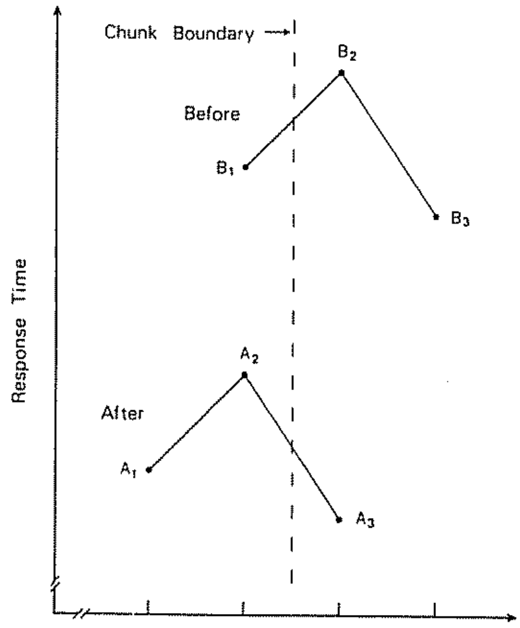

Figure 10 Hypothetical reaction time patterns at chunk boundaries for *after* and *before* conditions

Table 2
Frequency of Subjects in Experiments 1 and 2 Passing Strong and Weak Tests

Tests	No. of times passing 0	1	2	3	4	5	6	7	8	9	10	Total
Strong												
Experiment 1	2	1	3	3	3							12
Experiment 2	1	2	10	9	4	2	1	1				30
Total	3	3	13	12	7	2	1	1				42
Weak												
Experiment 1				1	2	2	1	4	1	1		12
Experiment 2					4	3	2	8	10	2	1	30
Total				1	6	5	3	12	11	3	1	42

Note. For the strong test there are 8 chances to pass; expected chance frequency is 1.33. For the weak test there are 10 chances to pass; expected chance frequency is 5.

preceding chunk has only two items. Thus a "strong" test for a postulated chunk boundary is that a subject's RTs at the boundary can be ordered: $A_2 > A_1 > A_3$. A somewhat weaker test is that at the chunk boundary there is a local dip in the RT curve: $A_2 > A_3$. A similar argument for the *before* case leads to corresponding strong ($B_2 > B_1 > B_3$) and weak ($B_2 > B_3$) tests.

Patterns of median RTs for individual subjects in both experiments were analyzed in order to determine the extent to which they observed the postulated boundaries. We computed the frequency with which each of the 12 subjects in Experiment 1 and the 30 subjects in Experiment 2 passed the strong and weak tests at the five postulated chunk boundaries.[8] We considered both *before* and *after* conditions, so each subject had 8 strong tests and 10 weak tests. Table 2 shows the frequency with which subjects in each experiment passed the strong and weak tests. For example, 3 subjects from Experiment 1 and 10 subjects from Experiment 2 passed the strong test two out of eight times. For the strong test, there is one chance in six that a random set of RTs would have the predicted order, and for the weak test the probability is 1/2. The null hypothesis is that the frequencies shown in Table 2 come from binomial distributions ($n = 8$, $p = 1/6$ for the strong test, and $n = 10$, $p = 1/2$ for the weak test). To test this hypothesis, a chi-square for lack of fit was computed, and extreme categories were collapsed such that all expected frequencies were greater than one. In all cases, the null hypothesis can be rejected with reasonable certainty: (a) strong test: Experiment 1, $\chi^2_3 = 14.8$, $p < .002$; Experiment 2, $\chi^2_3 = 54.8$, $p < .00001$; and (b) weak test: Experiment 1, $\chi^2_4 = 9.4$, $p < .06$; Experiment 2, $\chi^2_6 = 95.7$, $p < .0001$. Thus, individual subjects do indeed generate the predicted RT patterns at chunk boundaries at a frequency well above chance.

Another way to assess the reality of individual chunk boundaries is to examine the frequency with which each boundary shows the predicted pattern. Table 3 shows the proportion of subjects in both experiments (out of 42) who passed the strong and weak tests at each chunk boundary for the *before* and *after* conditions. For example, 45% of the subjects passed the strong test in the *before* condition at the G–H boundary. The boundary effects are very clear at the first two boundaries, and they are generally somewhat stronger for the *before* condition than for the *after* condition. Even the two smallest boundary effects (T–U and V–W) were detected by the weak test at the .05 level.

As a final test of the model, we have extended it to fit the alphabet recitation data of Lovelace et al. (1973) shown in Figure 2,

[8] Because the UV chunk has only two elements, there are only four boundaries at which to make the strong test.

Table 3
Proportion of Subjects from Experiments 1 and 2 Passing Strong and Weak Tests at Each Postulated Chunk Boundary in Before and After Conditions

	Boundaries					
Tests	G–H	K–L	P–Q	T–U	V–W	
Strong						
After	28*	.46***	29*	.09	—	.27***
Before	.45***	.46***	22	.17	—	.29***
Combined	.37***	46***	26*	.13	—	.28***
Weak						
After	.67*	81***	.64*	.52	.63*	.66***
Before	.84***	.71**	.61	.69**	.57	.68***
Combined	.75***	.76***	.62*	.60*	.60*	.67***

* $p < .05$. ** $p < .01$. *** $p < .0001$.

as well as a replication of the Lovelace et al. (1973) experiment by Browman and O'Connell (1976). In this task, subjects are presented with a pair of letters; the right-hand letter of the pair is either two, four, or six letters later in the alphabet than the left-hand letter. The subjects' task is to press a button as soon as they have recited all the letters between and including the starting and ending letter. Subjects performed both an overt and a covert recitation; the difference is that subjects did not say anything aloud in the covert recitation condition. (Only the data from the covert condition were available from the Lovelace et al., 1973, study.)

The interesting thing about these data, on close inspection, is the existence of prominent peaks at points in the alphabet where subjects have to recite across one or two chunk boundaries and the valleys at the beginning of chunks. Our model of this task is straightforward. We assume that subjects first search the alphabet until they find the left-hand letter. This search procedure is already modeled in the *after* task, and it is characterized by the top two loops in the flow diagram of Figure 8(A). When the left-hand letter is found, we assume that subjects say it and then do a series of NEXTs and SAYs until the right-hand letter is located. When subjects encounter a chunk boundary, we assume that they do a NEXT operation to retrieve the next chunk and then another NEXT operation to get the first element of the chunk. When subjects find the right-hand letter in the probe, they SAY it, press the RT button, and that completes the task.

To fit our model to the data of Lovelace et al. (1973), we simply counted the number of NEXTs and number of SAYs for each of the 68 data points of Figure 2, and we ran a multiple linear regression to estimate the time to do a NEXT and the time to do a SAY. The results of this analysis reveal the best fit of our model so far. Figure 2 illustrates the fit of the model to the data, with the predicted times shown as heavy lines; the two-parameter regression equation is given in Table 1. It accounted for 86.5% of the variance among the 68 data points, $F(2, 63) = 201$, $p < .001$. The amount of variance accounted for by each parameter alone was 65.4% for the number of NEXTs, $F(1, 64) = 121$, $p < .001$, and 78.1% for the number of SAYs, $F(1, 64) = 229$, $p < .001$, and the correlation between number of NEXTs and number of SAYs was .67. Thus, the partial correlation between RT and number of NEXTs with number of SAYs partialed out was .62, and between RT and number of SAYs with number of NEXTs partialed out was .78. There were only two significant outliers (greater than 2 standard errors from predicted): the probe with A as a first letter for Separation 6 was faster than predicted, and there was a large peak at the V–W boundary with Separation 2. This outlier was one reason for our use of the "Alphabet Song" segmentation.

We also fit our model to the data of Browman and O'Connell (1976), which is a replication of the Lovelace et al. (1973) experiment. We were able to fit our model to both overt and covert recitation conditions, since Browman and O'Connell published both sets of data. These data were virtually identical to the Lovelace et al. (1973) data ($r = .92$ and .94 for covert and overt, respectively), and our two-parameter model achieved good fits to both sets of data, accounting for 86.6% of the variance for overt recitation, $F(2, 63) = 204$, $p < .001$, and 81.1% of the variance for covert recitation, $F(2, 63) = 135$, $p < .001$ The regression equations are given at the bottom of Table 1.

Compared to the covert condition, the overt condition causes about a half-second delay in the button-push RT (the intercept), it slows the NEXT operation down slightly (from 53.9 to 66.3 msec, a 23% difference), and it produces a substantial increase in the SAY operation (from 220 to 313 msec, a 42% increase).[9] A close examination of the recitation equations reveals that in other respects the parameters are about the right magnitude. The NEXT parameter is in the 50–60 msec range, which is well within the limits of the other estimates of Table 1. The 200–300 msec range of the SAY parameter and the slower rate for overt recitation are both in close agreement with other published estimates of rehearsal rates for letters of the alphabet. Chase (1977), for example, measured rehearsal rates for random lists of letters that varied from 170 to 310 msec per letter, depending on the size of the list, and overt rehearsal was about 30 msec slower than covert rehearsal. The fact that our model obtains sensible parameter estimates is additional converging evidence for the model.

General Discussion

In summary, ALPHA has provided a satisfactory fit both to our own data of Experiments 1 and 2 and to the data in the literature of Lovelace and Spence (1972), Lovelace et al. (1973), and Browman and O'Connell (1976). An important feature of the model is its ability to replicate the fine structure of the peaks and valleys associated with presumed entry points. Another good feature is its simplicity in achieving satisfactory fits with a single parameter, the time to do a NEXT operation on a list. In the Lovelace et al. (1973) and the Browman and O'Connell (1976) data, an additional parameter was needed to model the recitation time. Finally, the magnitude of the NEXT parameter seems to be in the right range relative to other studies in the literature on the speed of list search processes, which vary from around 50 msec per item for the fastest type of search—scanning for the presence of an item in active memory—to around 250 msec per item for the relatively slow process of scanning for the location of an item within an arbitrary list in active memory (Sternberg, 1967).

The present model provides a detailed account of processes involved in the retrieval of alphabetic order information. The basic structure of ALPHA (Figure 7)—a hierarchical or multilevel model of serial search in which access occurs in a top-down manner—is similar to those previously advocated by a number of researchers (Lesgold & Bower, 1970; Martin, 1974; Seamon, 1973; Seamon & Chumbley, 1977).

As noted earlier, the preferred points of entry and individual chunk boundaries are shared in common by many individuals. These shared boundaries may derive in part from the "Alphabet Song," but this raises the issue of why the song has this particular structure. We speculate that the structure of the "Alphabet Song" derives from two basic mnemonic principles: chunk size and rhyming. First, Chase and Ericsson (1982) have recently proposed that order information for long serial lists takes the form of a hierarchical structure and that the size of sublists can not exceed the capacity of working memory because both storage and retrieval operations on chunks must be carried out in working memory. Thus, according to this

[9] One would a priori suppose that overt recitation should influence only the SAY parameter, although in retrospect there are plausible explanations why overt recitation might influence the other parameter as well. Overt recitation might induce the subject to change position on the speed–accuracy trade-off curve. Another possibility is that the required monitoring of overt vocal responses might compete for limited processing capacity, thereby slowing down the NEXT process

principle, alphabet sublists should contain seven or fewer items, with the optimum size being four items (Chase & Ericsson, 1982). The second mnemonic principle states that the phrases of the song should rhyme; that is, all the sublists should end with the same phonemic sound. The only exception to this rule is the second chunk, which ends with K because there is no alternative if chunk size is to be held to seven or fewer items. Thus, we are suggesting that the alphabetic structure revealed both by the behavior of our subjects and by the "Alphabet Song" is the direct result of the properties of the human memory system.

References

Anderson, J. R., & Bower, G. H. *Human associative memory.* Washington, D.C.: Winston, 1973.

Betail, I. (Ed.). *Complete nursery song book.* New York: Lothrop, Lee & Shepard, 1947.

Broadbent, D. E. The magic number seven after fifteen years. In R. A. Kennedy & A. Wilkes (Eds.), *Studies in long term memory.* New York: Wiley, 1975.

Browman, C. P., & O'Connell, D. C. Sequential phonological effects in recitation times. *Bulletin of the Psychonomic Society,* 1976, *8.* 37–39.

Chase, W. G. Does memory scanning involve implicit speech? In S. Dornic (Ed.), *Attention and Performance VI.* Hillsdale, N.J.: Erlbaum, 1977.

Chase, W. G., & Ericsson, K. A. Skill and working memory. In G. H. Bower (Ed.), *Advances in learning and motivation* (Vol. 16). New York: Academic Press, 1982.

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

Hovancik, J. R. Reaction times for naming the first and second next letters of the alphabet. *American Journal of Psychology,* 1975, *88.* 643–647.

Landauer, T. K. Rate of implicit speech. *Perceptual and Motor Skills,* 1962, *15.* 646.

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

Lesgold, A. M., & Bower, G. H. Inefficiency of serial knowledge for associative responding. *Journal of Verbal Learning and Verbal Behavior,* 1970, *9.* 456–466.

Lovelace, E. A., Powell, C. M., & Brooks, R. J. Alphabetic position effects in covert and overt alphabetic recitation times. *Journal of Experimental Psychology,* 1973, *99,* 405–408.

Lovelace, E. A., & Snodgrass, R. D. Decision times for alphabetic order of letter pairs. *Journal of Experimental Psychology,* 1971, *88,* 258–264.

Lovelace, E. A., & Spence, W. A. Reaction times for naming successive letters of the alphabet. *Journal of Experimental Psychology,* 1972, *94,* 231–233.

Martin, E. Serial Learning: A multilevel access analysis. *Memory & Cognition,* 1974, *2.* 322–328.

Müller, G. E., & Pilzecker, A. Experimentelle Beiträge zur Lehre vom Gedächtnis. *Zeitschrift für Psychologie. Ergänzungsband 1.* 1900.

Naus, M. J. Memory search of categorized lists: A consideration of alternative self-terminating search strategies. *Journal of Experimental Psychology,* 1974, *102.* 992–1000.

Naus, M. J., Glucksburg, S., & Ornstein, P. A. Taxonomic word categories and memory search. *Cognitive Psychology,* 1972, *3.* 643–654.

Seamon, J. G. Retrieval processes for organized long-term storage. *Journal of Experimental Psychology,* 1973, *97.* 170–176.

Seamon, J. G., & Chumbley, J. I. Retrieval processes for serial order information. *Memory & Cognition.* 1977, *5.* 709–715.

Sternberg, S. Retrieval of contextual information from memory. *Psychonomic Science.* 1967, *8.* 55–56.

Weber, R. J., Cross, M., & Carlton, M. Searching circular sequences. *Journal of Experimental Psychology,* 1968, *78.* 588–592.

Received March 1, 1982
Revision received September 6, 1982 ■