JOURNAL OF MEMORY AND LANGUAGE 31, 429–460 (1992)

Toward a Network Model of the Articulatory Loop

NEIL BURGESS AND GRAHAM J. HITCH

The University of Manchester, Manchester, United Kingdom

The basic features of verbal short-term memory for serially ordered lists are reviewed. A feed-forward network model based on Baddeley's concept of an "articulatory loop" is presented. One of its aims was to explore mechanisms for the storage of serial order information in the articulatory loop. Information is represented locally, learning is by "one-shot" Hebbian adjustment of weighted connections, corresponding to item–item and item–context associations, which decay with time. Items are modeled at the level of phonemes and phonemic output is fed back to the next phonemic input. At recall, items are selected serially by "competitive queuing." Noisy activation values are used, resulting in errors during recall. Simulations of recall showed good agreement with human performance with respect to memory span, phonemic similarity, word length, and patterns of error. There was good but incomplete agreement on the shape of the serial position curve and on the effects of articulatory suppression. A simple modification is shown to produce the correct serial position curve. However, the model was unable to simulate human memory for sequences containing mixtures of phonemically similar and dissimilar items. A suggested modification which retains the central idea of using competitive queuing to select among noisy activation values is described.

A considerable amount of empirical evidence suggests that human short-term memory is mediated by a relatively simple system with a highly restricted storage capacity. For example, immediate recall of random sequences of verbal stimuli such as words, letters, or digits is inaccurate if the sequences are more than only a few items long (Miller, 1956). This limited span of short-term memory contrasts with the very much larger storage capacity of long-term memory. The limited capacity system responsible for span appears to be involved in a wide range of more general cognitive tasks. This assumption is implicit in the inclusion of memory span in many tests of general intelligence (see, e.g., Terman & Merrill, 1961; Wechsler, 1955). It is explicit in a number of functional models which identify short-term memory with a general-purpose working memory (see, e.g., Atkinson & Shiffrin, 1971; Baddeley & Hitch, 1974; Broadbent, 1984).

Psychological studies of short-term memory have established that it involves multiple subsystems (see, e.g., Baddeley, 1986). The most important storage system in short-term memory for verbal stimuli appears to be a phonological system which holds information about serial order. Baddeley (1986) has presented a simplified model of this component which describes it as an *articulatory loop*. The main purpose of the present investigation was to construct and evaluate a network model of the articulatory loop. We begin by describing the concept of the articulatory loop and summarizing some of the behavioral evi-

This work was done while Neil Burgess was supported by an S.E.R.C. Ph.D. studentship and, subsequently, by the European Scientific Exchange Programme of the Royal Society and the hospitality of the Institute of Psychology of the C.N.R., Rome. We acknowledge many useful discussions we have had with M. A. Moore and J. L. Shapiro. We also thank G. Houghton for helpful private communications, D. Hintzman for trenchant comments on our original manuscript, and D. Bowie for help in preparing the manuscript. Requests for reprints should be addressed to Graham Hitch, Department of Psychology, Lancaster University, Lancaster LA1 4YF, UK or Neil Burgess, Department of Anatomy, University College, London WCIE 6BT, UK.

0749-596X/92 $5.00

dence supporting it. Having done this, we go on to consider the rationale for modeling and the particular approach we adopted.

The Articulatory Loop

Following Miller's (1956) paper it became standard to assume that memory span reflects the limited capacity of a single system for attention and short-term memory. However, it has gradually become apparent that more than one system is involved (see, e.g., Baddeley & Hitch, 1974) and that an important limitation on span arises from the contribution of a specifically speech-based subsystem. Early evidence was the observation that substitution errors in recall of visually presented sequences of letters tend to share phonemes in common with the correct items (Conrad, 1964). Phonemically similar items are also particularly difficult to recall (Baddeley, 1966) and Murray (1968) found that this effect could be abolished by requiring subjects to perform articulatory suppression, the repetition of redundant words such as "the–the–the," during the memory task. Suppression therefore appears to disrupt phonological short-term memory. However, suppression does not reduce recall to chance levels (Estes, 1973; Peterson & Johnson, 1971), confirming that a second, non-phonological component contributes to performance. In the working memory model of Baddeley and Hitch (1974), this second component was seen as a limited capacity central executive, which could control the operation of the articulatory loop.

The articulatory loop is thought to be particularly concerned with retaining information about serial order. In the most recent account (Baddeley, 1986) it is described as a limited capacity phonological store coupled with a control process of subvocal rehearsal. Memory traces in the phonological store are assumed to decay in 1 or 2 s unless refreshed serially by rehearsal. This simple model can give a coherent account of the effects of word length, phonemic similarity and articulatory suppression on short-term recall, as follows.

The word length effect is the tendency for short-term recall to be poorer for longer than shorter words. Baddeley, Thompson, and Buchanan (1975) observed a linear relationship between the number of words recalled and the time taken to articulate them such that the amount subjects could recall equalled the number of words they could say in about 1 or 2 s, plus a constant. This relationship was interpreted in terms of the longer time taken in subvocal rehearsal of longer words allowing more trace decay. A simple metaphor is that of a closed tape loop which can store a 1 or 2 s sequence of "inner speech." Baddeley et al. (1975) confirmed this interpretation in subsequent experiments in which they demonstrated that the word length effect could be abolished by articulatory suppression. The phonemic similarity effect and its sensitivity to articulatory suppression can also be explained in terms of the loop by assuming that it reflects confusions among similarly coded items.

The empirical evidence about the loop is not, however, quite as straightforward as has been described. In particular, it was discovered that whereas suppression removed the word length effect for spoken or visual stimuli, it removed the phonemic similarity effect only when stimuli were presented visually and not when they were spoken (Baddeley, Lewis, & Vallar, 1984). These findings suggested that although the phonemic similarity and word length effects are closely related, they must be carefully distinguished from one another. The model of the articulatory loop has been refined to accommodate these effects of presentation modality. It does so by maintaining that whereas the word length effect is due to rehearsal processes, the phonemic similarity effect arises within the phonological store. The modality-specific effect of articulatory suppression can then be explained by assuming that visual stimuli, but not au-

ditory stimuli, have first to be recoded in order to enter the phonological store, and that this extra process involves subvocalization. Further evidence for the complementarity of word length and phonemic similarity effects has come from analysis of the temporal limit associated with memory span (Schweickert, Guentert, & Hersberger, 1990).

The articulatory loop is still a relatively simple theoretical concept even when it is modified to account for the role of presentation modality and differences between the phonemic similarity and word length effects. However, despite its simplicity, it has been shown to be a remarkably robust concept with considerable generality and applicability. The basic empirical effects on which it is based have been replicated several times. Furthermore, the model has been shown to account for a number of new findings. Foremost among these is the tendency for short-term recall to be disrupted by exposing subjects to unattended speech (Salamé & Baddeley, 1982). This effect is explained by the assumption that unattended speech gains obligatory access to the phonological store, where it competes with traces of the memory items. The articulatory loop has also been surprisingly successful in accounting for cross-linguistic differences in digit span. When the rate at which digits can be spoken in different languages is taken into account and the underlying temporal capacity of the loop is calculated, such differences largely disappear (Ellis & Hennelly, 1980; Naveh-Benjamin & Ayers, 1986). The developmental increase in the memory span of children has also been analyzed in terms of the articulatory loop. For example, it has been found that the temporal capacity of the loop remains constant during development and that the improvement in span is predictable from an increase in speech rate as children grow older (Hulme, Thompson, Muir, & Lawrence, 1984; Hitch, Halliday, & Littler, 1989). There have also been applications to routine cognitive skills such as reading and to the analysis of neuropsychological impairments (see, e.g., Baddeley, 1986). To sum up, the properties of simplicity, robustness, generality, and applicability underpin the usefulness of the articulatory loop as a theoretical concept.

Limitations of the Articulatory Loop Model: The Problem of Serial Order

Although the articulatory loop is clearly a useful concept, it has some important limitations as an explanatory account. For example, given that one of its major functions is the preservation of order information, surprisingly little is said about how this is achieved, and some of what is said is clearly incorrect. As has been seen, the model accounts for the phonemic similarity effect in terms of increased difficulty in discriminating among the memory traces of similar items. This explanation is consistent with the well established finding that phonemic similarity tends to disrupt memory for the serial order of the items rather than memory for the items themselves (Wickelgren, 1965a). However, there is only a hint of a mechanism here, and the model fails to explain important observations such as the tendency for phonemically similar items to be involved in paired transpositions—where one item is interchanged with another in recall (Conrad, 1964). More generally, the model does not attempt to explain other aspects of order errors such as the tendency for paired transpositions to involve adjacent items (Conrad, 1964), nor the characteristic bowed shape of the curve relating probability of error to serial position (Murray, 1966). Where the model is explicit about a mechanism for storing order, as in the tape loop metaphor, it is clearly unable to explain any of the above effects without additional elaboration. However, it remains an open issue whether the model can be extended to give an account of order errors.

The problem of how serial order is re-

tained in short-term memory is one aspect of the general problem of how serial order is remembered at all. This problem has a long history which includes Hull's ideas about interitem associations, Lashley's (1951) powerful critique of associative "chaining," the idea of position–item or contextual associations, and various more complex proposals (see, e.g., Young, 1968). There have been numerous models which attempt to deal with serial order in short-term memory (see, e.g., Lee & Estes, 1981; Lewandowsky & Murdock, 1989; Shiffrin & Cook, 1978), by making different assumptions about the associations which encode serial order, but they fail to deal fully with the pattern of phonemic similarity, word length, and articulatory suppression effects that are so well explained by the articulatory loop account.

There is some evidence for interitem chaining from observations of "associative intrusions" in recall (Wickelgren, 1965b). These are errors involving switching the elements following repeats, as in the sequence a x b a y being recalled as a y b a x. However, simple chaining (that is, the recall of an item being prompted only by the recall of the previous item) would predict that sequences containing repeated items would be extremely difficult to recall and this is evidently not so (Jahnke, 1969). The existence of position–item associations is indicated by the observation of "serial order intrusions" in recall (Conrad, 1960). These are intrusions where the error is the item that was correct (at that position) in the immediately preceding list. However, any purely positional account would of course have difficulty explaining associative intrusions. It seems probable therefore that an extended model of the articulatory loop will have to incorporate some mechanism for storing both interitem and position–item associations if it is to account for behavioral data on the storage of order information in short-term memory.

Rationale for Modeling

We have argued that the Baddeley (1986) model of the articulatory loop captures some important regularities about short-term recall but fails to give a realistic account of the problem of serial order. The simplicity of the model holds the attraction of making it easy to understand and apply, but it has the disadvantage of restricting its explanatory value. The primary motivation for constructing a network model of the articulatory loop was to specify a system which would retain the advantages of the present simplistic model, in accounting for the word length, phonemic similarity, and articulatory suppression effects, but which would deal more effectively with the problem of serial order and with patterns of error in recall. A successful model might also be expected to make some novel predictions and to generate insights into other related phenomena.

Approach to Modeling

Any attempt to build a network model of the articulatory loop must adopt an approach falling somewhere between two extremes. At one end is a top-down approach which attempts to implement the current concept of the articulatory loop. At the other is a bottom-up approach which attempts to identify the computational problem that is solved by phonological short-term memory. We rapidly rejected an entirely top-down approach on the grounds that it would be impractical given the underspecification of the articulatory loop as described by Baddeley (1986). In any case, our network model was intended not merely as an implementation of the articulatory loop, but as a development from it which would include a solution to the problem of serial order. To satisfy this goal a bottom-up approach seemed more appropriate. This takes the computational problem of storing information about the serial order of an unpredictable series of verbal stimuli during a single presentation as its starting point. It then goes on to ask if the solution to this computational problem is a system which behaves similarly to what is

already known about the articulatory loop. However, it is important to appreciate the practical reality that no bottom-up approach to modeling can proceed without the intrusion of insights from known human data and without influence from previous thinking. This was certainly the case here, as will be clear in the selection of constraints to guide model building and, in particular, in an early decision to include item–item and context–item associations.

Empirical Constraints on Modeling

In the following section, we describe the empirical constraints from human data which were considered important in determining the architecture of the model. These include effects that are currently attributed to the articulatory loop and, in addition, others we judged basic to people's ability to remember the order of series of items immediately after a single presentation.

a. Decline in immediate recall with increasing list length. Although any model ought to predict the limit on the span of short-term memory to about seven items, this is only a relatively crude index of performance. The mechanism for storing serial order has an even more limited capacity than is implied by span, as shown by the function relating the probability of correct recall to sequence length. Figure 1 illustrates this function for sequences of auditorily presented digits using data reported by Guildford and Dallenbach (1925). It is clear that recall is virtually perfect up to about five digits and that the proportion correct drops dramatically thereafter to reach zero by sequence length 10. One of the targets for modeling was to reproduce the general form of this function.

b. Phonemic similarity, word length, and articulatory suppression. One of the successes of the current articulatory loop model is its ability to account for the disruption to short-term recall arising from increasing phonemic similarity of the items (Baddeley, 1966) and from increasing word length (Baddeley et al., 1975), and the sensitivity of these effects to articulatory suppression. It was considered fundamental that the network should show these effects.

c. Shape of the serial position curve. The serial position curve is one of the best known aspects of short-term recall. It characterizes performance for sequences of intermediate length, where performance is breaking down for part but not all of the list. An important determinant of the shape of the serial position curve is whether items are presented visually or auditorily (Crowder, 1972). As can be seen in Fig. 2, there is a clear primacy effect for both methods of presentation, corresponding to a decline in accuracy from the start of the sequence onwards. However, superimposed on this tendency is a recency effect which is markedly greater for auditorily presented items, and which is restricted to the final item. The

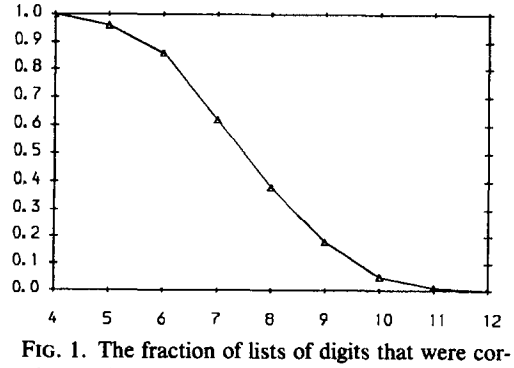

FIG. 1. The fraction of lists of digits that were correctly recalled versus list length, adapted from Guildford and Dallenbach (1925).

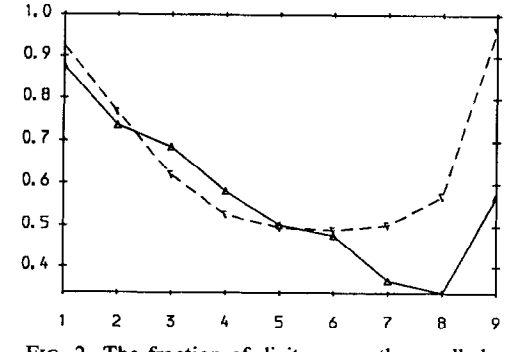

FIG. 2. The fraction of digits correctly recalled at each serial position in lists of nine digits. Visual presentation: full line, simultaneous visual and oral presentation: dashed line, adapted from Crowder (1972).

present account of the articulatory loop has very little to say about this phenomenon (Baddeley, 1986). Therefore, an immediate problem is whether to attempt to simulate both forms of presentation or just one, and if the latter, which.

Conventional accounts of the influence of presentation modality on the shape of the serial position curve have maintained that it reflects the operation of a speech-specific auditory input store (Crowder, 1972). However, this explanation has been seriously challenged by more recent data (Gardiner, 1983) and there is at present no generally accepted account. Given that the articulatory loop is assumed to be a mechanism that is fed by visual or auditory input, the obvious choice here was to model features of performance common to both presentation modalities. Therefore, the initial target for simulation was the serial position curve for visual presentation. Since the recall data suggest that whatever is responsible for auditory recency is additive to this common system, it seemed reasonable to regard extension of the model to account for the modality effect as a longer-term goal.

d. Types of recall error. An error in recalling an individual item from a sequence must be either an order error, where an item drawn from elsewhere in the presented sequence is reported, or an item error, where an item from outside the presented sequence is reported, or an omission. By far the greater proportion of errors are order errors (see, e.g., Aaronson, 1968; Bjork & Healey, 1974), and these often involve paired transpositions where the positions of two items from the sequence have been interchanged. Conrad (1964) presented data showing that the majority of paired transpositions involve adjacent items, e.g., the sequence a b c d e f recalled as a b d c e f. He found that the frequency of paired transpositions declines with the number of intervening items. Paired transposition errors are impossible to explain in terms of a simple chain of interitem associations, and so accounting for their occurrence was a key target for the present simulation.

e. Phonemic confusion and order errors. It has already been noted that errors where one item is substituted for another tend to share phonemes in common. Conrad (1964) published a confusion matrix showing that, for example, f is more likely to be substituted by s than t. It has also been noted that short-term recall is disrupted when the presented items are phonemically similar and that this effect is associated with an increase in order errors (Wickelgren, 1965c). It was considered fundamental that the simulation should have these properties.

Note that constraints c and d refer to empirical data outside the scope of the simple concept of the articulatory loop, while the others refer to effects for which the loop does provide some form of explanation. The goal of simulation was to build a model satisfying the above constraints and to explore its ability to account for other aspects of short-term recall.

A Network Model of the Articulatory Loop

Connectionist Modeling Background

Part of the problem of modeling short-term memory is that it requires a network with unusual characteristics. The net must be able to encode a novel sequence of stimuli in a single presentation and then recall the stimuli immediately in the correct order. Most network models of memory attempt to solve the problem of storing large numbers of patterns without regard to their order, and the application of network modeling techniques to problems involving temporally ordered behavior is relatively new (see, e.g., Jordan, 1986).

Most of the work on creating explicit "neural network" models can be divided into two areas:

1. Networks in which units are arranged in layers; activity in the units of one layer feeding forward to the next layer. The input and output information is often (but not necessarily) represented locally, i.e., each item is represented by the activity of a single unit.

2. Networks in which each unit is connected homogeneously to a fraction of all the other units. Information is distributed throughout the pattern of activity of all the units. The archetypal example of this kind of network is the Hopfield model (Hopfield, 1982).

There have been some models of short-term memory based on modified Hopfield models (Hopfield, 1982). The Hopfield model can be changed in a variety of ways so that a limited number of the most recent patterns of activity can always be stored (Parisi, 1986; Nadal, Toulouse, Changeux, & Dehaene, 1986; Mezard, Nadal, & Toulouse, 1986). Of these only a few have been compared with psychological phenomena (e.g., Nadal, 1987; Virasoro, 1989) or with human data (e.g., Burgess, Moore, & Shapiro, 1989; Burgess, Shapiro, & Moore, 1991). However, actual models of recall within the Hopfield paradigm are very scarce (see Amit, Sagi, & Usher, 1990, for a model of Sternberg fast scanning) and none model the serially ordered recall of whole lists.

A temporal sequence of patterns can be stored in a Hopfield network by using Hebbian learning (see below) that partly stores each pattern and partly stores the association between previous patterns and subsequent patterns. The network can be made to cycle through the various stored patterns in order. This has been done in many slightly different ways (see, e.g., Kleinfeld, 1986; Buhmann & Schulten, 1987; Amit et al., 1990). However, although order information can be stored as a temporal sequence of patterns of activity, the kinds of errors typically made by humans denies this type of system. It is hard to see how such a system would make order errors on recall such as paired transpositions. If errors were made in the temporal sequence of patterns of activity it is unlikely that each pattern would still be visited once or that the temporal positions of a pair of patterns could be transposed. The same difficulty applies to models which involve error back-propagation through time (Rumelhart, Hinton, & Williams, 1986) and back-propagation with feedback (Jordan, 1986; Elman, 1990), added to the necessity of repetitive learning with such models.

There is a longer history of memory modeling using layered networks and local representations, particularly with "on-center off-surround" models in which connections are excitatory between nearby units and inhibitory between distant units. There are many interesting properties shown by networks with this type of architecture (see, for example, Kohonen, 1984; Grossberg, 1987). Given the speech-like nature of the articulatory loop one of these, Houghton's (1990) "competitive queuing" model of speech production, is particularly interesting. In his model of the production of a serially ordered string of sounds, each phoneme is represented by a single node. The main feature of the model is that nodes can be active before their articulation, the most active node being selected by a "competitive filter" (see below). This, together with the way that nodes are temporally cued, results in the coarticulation effects necessary in a realistic model of speech production. Although the model does not use "one shot" learning, this is mainly to prevent order errors. The pronunciations of words are learned gradually and are stored in long-term memory. It is Houghton's method of temporal cuing that necessitates repetitive learning and does not seem appropriate here. However, the selection and subsequent suppression of each item by a competitive filter seems a natural choice for our model of the articulatory loop.

Brief Outline of the Model

We use a layered architecture (see Fig. 3) in which information is locally represented by nodes taking activation values between ± 1. Positive activation values represent activity above background or resting levels, and only positive activation values are propagated forward between layers via weighted connections. The activation value of each node depends on the weighted sum

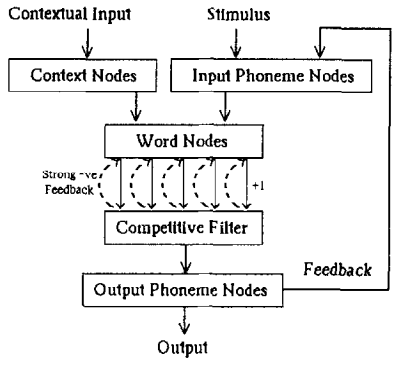

FIG. 3. An outline of the architecture of the model.

of the activations of nodes to which it is connected. The activation values of nodes are "noisy": they do not respond perfectly but include a small random element. Learning takes place by "one-shot" Hebbian adjustment of the weighted connections.

Presentation and articulation of items is modeled at the level of phonemes. The presentation of an item excites a layer of "input phoneme" nodes, and the articulation of an item involves exciting a separate layer of "output phoneme" nodes. The phonemic output for one item is fed back to excite the phonemic input for the next. That is, there is a set of feedback weights which store item–item associations or links. It is by this feedback that the model forms a basic "articulatory loop." A second set of weights stores context–item associations; the relative influence of these two mechanisms can be varied and is an important free parameter of the model. The "context" for each item is represented by a random pattern of activation which alters progressively with the passage of time. These two sets of "temporary" weights are "learned" during a single presentation of a series of stimuli. They also decay with time.

The associations between a word or letter and its constituent phonemes necessary for its recognition and articulation are stored in "permanent" weighted connections. Presenting the "input phoneme" and "context" states of a word excites the node corresponding to that word which, if selected, excites the relevant "output phoneme" nodes. The articulation of each word in the list is achieved by "competitive queuing": a "competitive filter" (in which there is strong lateral inhibition) selects the most active word node and then suppresses it before recall of the next word.

The model can perform rehearsal in this way: the temporary weights that store associations for a particular item are automatically refreshed (by Hebbian adjustment) each time an item is rehearsed. Thus temporary weights do not decay to zero and the associations that they store may change. Thus "recall" of the list can be triggered many times. Errors in recall (that is, selecting the wrong item nodes, or selecting the right nodes in the wrong order) occur according to the amount of noise that is introduced into the activation levels.

It should be noted that the use of permanent connections to store associations between words or letters and their constituent phonemes is an obvious simplification since unfamiliar nonsense words can also be stored in phonological short-term memory (see, e.g., Gathercole & Baddeley, 1989; Hintzman, 1967). However, this simplification was regarded as a reasonable starting point given that most data on human behavior are concerned with familiar stimuli and given that the model could be simply modified to allow weights connecting the phoneme and word layers to be learned at presentation.

DESCRIPTION OF THE MODEL

Four layers of nodes feed directly from one to the next: a layer representing input or presented phonemes, a layer representing each word, a related "competitive filter" layer that selects which word to articulate and a layer representing output or articulated phonemes.

Feedback occurs in two places. There is strong inhibitory feedback from the competitive filter to the word nodes (to suppress a word node after the word has been articulated) and excitatory feedback from the output phoneme nodes to the input phoneme nodes. Excitatory feedback allows the articulation of a word to excite the input

phoneme nodes corresponding to the next word in the sequence during rehearsal or recall (we refer to this as "chaining"). For simplicity the model does not distinguish between subvocal rehearsal of a list and actual vocal recall of a list.

The "Context" Layer

The context layer feeds forward to the word nodes and is used to represent all the nonphonological information in the input, including temporal information. As far as possible, the context and the input phoneme layer (see below) were made to have similar characteristics. There were thus 50 context nodes. During the presentation of each word a random two-thirds of these are updated and the remainder are left unaltered. Of the updated nodes, a randomly chosen subset of six are given a nonzero activation comparable to the average activation in the phoneme layers. The remainder are given an activation of zero. In this way, an average of only nine context nodes are active during the presentation of any item. Importantly, the fact that some context nodes are not updated when a new word is presented gives some time correlation to the patterns of activation for successive inputs. Thus the similarity of the activation patterns for two words will on average vary monotonically with their temporal separation in the input sequence.

How Activations Propagate

The activity of all other (i.e., noncontext) nodes is determined by the activities of the nodes to which they are connected. The activity $a_j(t)$ of a node onto which arrive connections (of "weight" W_{ij}) from c nodes each of positive activity $\{a_i(t) > 0, i = 1 \text{ to } c\}$ is updated at time t according to the weighted sum of activities,

$$a_j(t + 1) = f\left(\sum_{i=1}^{c} W_{ij}a_i(t)\right), \qquad (1)$$

where f is a squashing function which keeps the value of a_j between -1 and $+1$. Note that only positive activation values are propagated and updating is done in parallel (i.e., synchronously).

This equation shows the activity of an immediately reactive node, i.e., one which responds to the signals it receives as soon as it is updated. If a node has been suppressed to a very inactive state (denoted by a negative activation level), or if it is in the competitive filter (which is updated on a much shorter time scale than other layers) its activity does not change so immediately—but depends partly on its previous value according to:

$$a_j(t + 1) = f(a_j(t) + \sum_{i=1}^{c} W_{ij}a_i(t)). \qquad (2)$$

Selecting a Word in Recall

The output of each word involves a cycle in which the most active word node is selected, exciting its corresponding output phoneme nodes, and it is then suppressed. This is done by "competitive queuing" (Houghton, 1990), as follows. Each word node connects to a competitive filter node with an excitatory connection. Thus the competitive filter initially reproduces the pattern of activation of the word layer. However, there is strong lateral inhibition between nodes in the competitive filter (*cf*) which, after relatively few iterations of the activations in this layer, results in the *cf* node connected to the most active word node suppressing all the other *cf* nodes to negative activation levels. The winning *cf* node then excites the output phoneme nodes corresponding to the selected word node.

There are also strongly inhibitory connections from each competitive filter node back to the corresponding word node. Thus the next time the word layer is updated the selected word node is suppressed by the winning competitive filter node (see Fig. 4).

Connections, Weights and Learning

There are three types of connections between nodes: "hard wired" connections, prelearned permanent connections, and

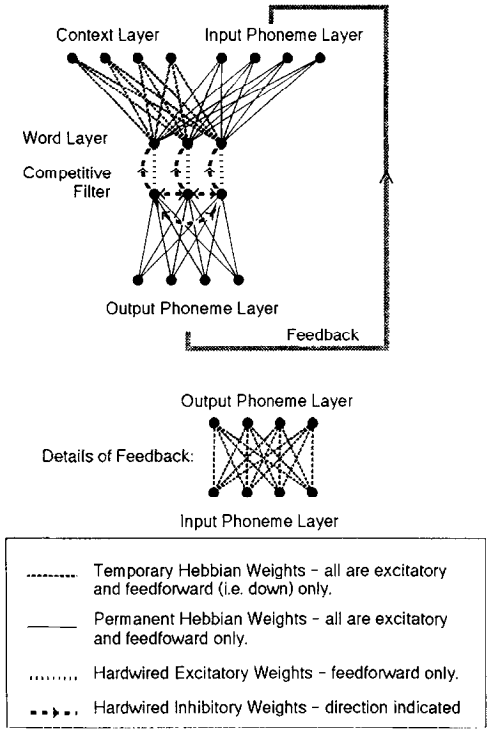

FIG. 4. A schematic representation of the model of the articulatory loop—in reality there are more nodes.

temporary connections which are learned, and decay, during an experiment (see Fig. 4).

The lateral inhibition in the competitive filter layer, the one-to-one excitatory connections from word nodes to competitive filter nodes and the one-to-one inhibitory feedback to the word nodes are hard wired. The values of the connection weights do not change and are the same for all connections involved in the same task.

Learning (i.e., weight modification) only occurs in excitatory connections and is achieved by "one-shot" Hebbian increment of connections between two nodes with positive activation.

Permanent connections store the associations enabling the "recognition" and "articulation" of individual words. These are the connections which excite the relevant word node when the corresponding input phoneme nodes are active or excite the relevant output phoneme nodes when a particular word has been selected by the competitive filter. The weights for these connections are prelearned before the model is presented with immediate recall tasks.

Temporary weighted connections are used to learn (i) the association between the state of the context nodes when a word is presented and the corresponding word node, and (ii) the association between the output phonemes of a word and the input phonemes of the next word.

If a connection of weight W_{ij} connects two nodes of activity a_i and a_j then learning implies

$$\text{if } a_i > 0 \text{ and } a_j > 0 \quad \text{then } W_{ij} = \epsilon a_i a_j, \tag{3}$$

where ϵ is a parameter governing the size of the weights (see implementation).

The temporary weights decay with elapsed time. We (crudely) take the amount of time that elapses during the presentation or rehearsal of a word to be proportional to the total number of phonemes output. Thus the decay of a temporary weight during the presentation or rehearsal of a word containing n phonemes is

$$W_{ij} \mapsto \Delta^n W_{ij},$$

where Δ is the decay factor per (phonemic) time step.

The Word Layer

For most of our simulations we use the letters of the alphabet as the vocabulary of the system. Accordingly there are 26 "word" nodes, one for each letter. To investigate the word length effect systematically we sometimes use a vocabulary of "random words," i.e., "words" made from random selections of phonemes, with different numbers of phonemes per word.

As described above, the weighted connections between the input phoneme layer and the word layer for the recognition of all the words in the vocabulary are prelearned, and are permanent. For example, to learn to recognize the letter c the input phoneme nodes for "s" and "ee" and the word node

for c are activated and connection weights are incremented according to Eq. (3).

Thus activation of input phoneme nodes "s" and "ee" will cause activation of the word node c; it will also cause lesser activation of the word nodes b d e g p t v s x which will also have excitatory connections to either "s" or "ee." We opted to see how much could be achieved by making the simplest possible choice of phonological coding. Thus, the phoneme nodes were not ordered in any way, and so the system would treat "s-ee" and "ee-s" as equivalent. For most stimuli and in particular those used in testing the model, this simplification does not lead to any errors of recognition.

There are also excitatory connections from the context layer to the word layer. These learn the association between each word and the context state for each word. Thus, if at recall the nodes in the context layer have the pattern of activation that coincided with presentation of the first word, the corresponding word node will be activated. As with the input phoneme nodes, there will be nodes that are active in the context state for more than one word. Because of the way context is assumed to vary, the overlap will be greatest for words that are adjacent in the list. Thus, if the context nodes are in the pattern of activation for the first word, the corresponding word node will be the most active, but others will also be active (particularly the second word node)–but to a lesser degree.

The connection weights from the context layer are learned on the presentation of each word and decay with time (see above). Each time a word node is selected during rehearsal the temporary weights between the current context state and that word node are relearned.

A fraction F_{ph} of the activation of a word node is determined by the input phoneme layers and $1 - F_{ph}$ by the context layer (see "Setting Correct Weight Strengths"). During rehearsal or recall some word nodes will be active before their time of recall, irrespective of which layer is driving the word layer. The competitive queuing (see above) serves to select the most active node each time a word should be recalled and then to suppress it in time for the next word to be recalled.

The Phoneme Layers

There are 53 input and output phoneme nodes, one for each possible phoneme. For simplicity, no attempt is made to represent the similarity space of phonemes. Most of the phoneme nodes are not used in lists of letters of the alphabet, although they are all used to make up "random words." Output phoneme nodes are excited by activation in a competitive filter node. The excitatory connections required to articulate a word are prelearned. For example, to learn to articulate c, the competitive filter node corresponding to word node c and the output phoneme nodes "s" and "ee" are activated, and the connection weights are updated according to Eq. (3).

The input phoneme nodes are excited by temporary connections from the output phoneme layer. These connection weights store item–item associations, referred to here as chaining weights. They are learned during presentation and decay with time. During the presentation of a word, the corresponding input phoneme nodes are activated while output phoneme nodes corresponding to the phonemes in the preceding word are still active. The chaining weights are incremented according to Eq. (3). Thus, during recall or rehearsal, the "saying" of each word will excite the input phoneme nodes for the next word.

There is opportunity for erroneous activation of input phoneme nodes (as there is for word nodes) if words in a list share phonemes. For example in the list a b f c q, excitatory weights will be learned from output phoneme node "ee" to input phoneme nodes "e," "f," "kh," and "uu" (in the associations b $\mapsto$ f and c $\mapsto$ q). Thus when letter b is output the input phoneme nodes for f will be excited, but so will (to a lesser extent) the input phoneme nodes for q.

During recall/rehearsal the chaining weights are relearned each time a word is articulated.

These decaying chaining weights which store phonemic item–item associations and must be relearned during rehearsal are our interpretation of a decaying phonological store. For example, we interpret articulatory suppression as putting these weights out of action (i.e., preventing them propagating activations).

There are two immediate constraints on the size of connection weights in the phonological store:

(i) When a word is presented (i.e., its input phoneme nodes are excited) the corresponding word node should become equally excited whether the word contains many or few phonemes.

(ii) Nodes corresponding to words that share phonemes with an input word should be less activated by the input phoneme nodes than the node for the inputted word itself.

These are made harder to satisfy by the fact that words within a list may be of different lengths and may have phonemes in common.

The prescription we have used is to ensure that the phoneme nodes corresponding to the input or output of a word of n phonemes have activity proportional to $1/\sqrt{n}$. The two most obvious choices of having activity independent of n or proportional to $1/n$ (so that the total phonemic activity for a word is independent of n) will not satisfy (i) and (ii), whatever the choice of strengths for the other weights in the model.

Noise

Up to this point we would have a system which could recall perfectly. However, the occurrence of errors in humans is both fundamental and instructive: a realistic model must also show this type of unreliability. The behavior of microscopic biological systems tends to be probabilistic, i.e., while behavior averaged over time or over many systems depends on an external influence, the behavior of, say, an individual neuron at a particular time is erratic. So a natural choice for our model is to put noise equally into the rule determining the activation level of nodes. The success of this choice should be judged by how humanlike the resulting errors are. Accordingly we adapted Eq. (1) to become

$$a_j(t+1) = f\left(\sum_{i=1}^{c} W_{ij}a_i(t)\right) + \eta, \qquad (4)$$

where η is a random variable taking values uniformly between $\pm\sigma$. Equation (2) changes similarly. The strength of the noise is determined by σ. This also introduces noise into the connection weights through Eq. (3). Note that the bounds ± 1 on the activity of a node are not rigid. If a node has an activation level of very nearly 1 in Eq. (1), the addition of noise as in Eq. (4) could make its activation greater than 1.

The introduction of noise ensures that the decay of the temporary weights has an effect. Without noise, decay would result in the typical levels of activation in the system continually decreasing, but not necessarily in any change of behavior. With noise the level of activation of the nodes in a layer must be large in comparison to the level of noise or the information held by the layer will be lost.

If the system is not making mistakes then a temporary weight will decay by Δ^M between each relearning, where M is the total number of phonemes in the list. The activation levels of nodes in the word layer will decrease by the same factor. Errors will occur when the activation level of word nodes is low enough for the noise term in Eq. (4) to be significant in determining which one is the most active. Thus the length of a list that the model can recall without error should be determined by M, i.e., by the time taken to articulate a list (within our approximation that all phonemes take the same length of time to articulate).

Rehearsal and Relearning

During recall or rehearsal the temporary weights are relearned to prevent them from decaying to zero. With the output of each word there occurs Hebbian relearning of both sets of temporary weights. The temporary weights storing the (context–word or output phoneme–input phoneme) associations for a particular word are only incremented if that word is selected in rehearsal. For the chaining weights between output and input phoneme layers this cannot be otherwise; the only output phoneme nodes that are excited at any one time are those corresponding to the most recently selected word. In the context–word weights, relearning occurs only in those weights connected to the word node corresponding to an active competitive filter node (i.e., the most recently selected word node). Note that if a word is omitted in recall its temporary weights will continue to decay, and when an error occurs the erroneous context–word and phonemic associations will be learned.

We want temporary weights only to decrease by means of decay, so if the connection weight W_{ij} is already greater than $\epsilon a_i a_j$ before learning, no change should take place. Hence Eq. (3) becomes

$$W_{ij}(t+1) = \begin{cases} \epsilon a_i(t) a_j(t) & \text{if } W_{ij}(t) < \epsilon a_i(t) a_j(t), \\ W_{ij}(t) & \text{otherwise.} \end{cases} \quad (5)$$

Thus temporary weights can only be increased by relearning; they are only decreased by decaying. This change only affects the temporary weights; permanent weights are incremented once each from zero so that, for them, Eqs. (3) and (5) are equivalent. In this paper we only consider immediate recall, involving a single rehearsal, in any detail.

Setting Correct Weight Strengths

The size of the connection weights will determine the typical activation values of excited nodes in the various layers. The size of a weight is set by the value of ϵ during learning in Eq. (3). We can calculate the values of ϵ that will result in the desired activation values. If the desired running activation levels for word, input phoneme, output phoneme, and context nodes are A_{wd}, $1/\sqrt{n}$, $1/\sqrt{n}$, and A_{ct} respectively, then the correct learning strength ϵ for weights can be calculated from Eq. (1). For example, the learning strength for weights from the context to word layers is: $\epsilon = (1 - F_{ph}) f^{-1}(A_{wd}) / A_{wd} A_{ct}^2 N_{act}$. Thus weights learned according to Eq. (3) when context and word nodes have activation A_{ct} and A_{wd} will be of the correct strength to ensure that these nodes will have activations A_{ct} and A_{wd} when they are excited during rehearsal/recall.

For the hard wired connections all weights sharing the same function have the same value; e.g., weights from the word layer to the competitive filter are all +1.0.

The decay of temporary weights means that the activation values of word nodes will also be lower by a factor of Δ^M after each rehearsal. Thus during Hebbian relearning the learning factor ϵ must be increased to ϵ/Δ^M if relearned weights are to have the same magnitude as before decay (otherwise all activation values will decay by Δ^M for each rehearsal until they are lost in the noise).

Parameter Values

To determine the weight strengths we use in the model we must decide on the activation values we want for excited nodes in layers other than the phoneme layers. These are as follows:

$A_{ct} = 0.50$ so that the context layer is as comparable to the input phoneme layer as possible.

$A_{cf} = 0.50$ for convenience—this value is not important.

$A_{wd} = 0.25$—any small value will do, so that weight strengths to the word layer are relatively low and previously suppressed word nodes will not often become reexcited

and get repeated (repetition errors are rare in the human data).

$\Delta = 0.95$—this cannot be much smaller than 1 or else at the beginning of recall the temporary weights for early items in the list will be very much smaller than those for later items. If this happens later items will be recalled in place of earlier ones, but not because of either of the confusion mechanisms intentionally in the model (confusion in the context or phoneme layers).

It is hard to see what the remaining two parameters in the model should be without reference to the human data; these are:

F_{ph}—the fraction of input to the word nodes from the phonological store rather than from the context nodes; this will determine the strength of effects dependent on phonemic similarity and item–item association (i.e., "chaining").

σ—the amount of noise in the activation levels of nodes; this will determine the capacity of the model.

Thus we left only two independent parameters to vary during simulations of the model.

Simulation Procedure

The procedure to model presentation and rehearsal/recall is as follows (see also Figs. 3 and 4):

- All nodes and connection weights are set to zero (excluding the "hard wired" connections which are fixed).
- Prelearning. The association between the input phoneme nodes corresponding to a word and the relevant word node is learned. Similarly the association between a competitive filter node (corresponding to a selected word) and the relevant output phoneme nodes is learned.
- Presentation. 1. Initialization. A sequence of words to be learned is chosen. The context nodes are set to a random initial state.

2. Presentation and learning of the ith word.

i. The input phoneme nodes are set to the phonemic activity of the i_{th} word. Two-thirds of the context nodes are updated.

ii. The "chaining" weights from the output phoneme layer to the input phoneme layer are learned using Hebbian learning as in Eq. (5); the output phoneme nodes are still active from the articulation of the last word. Note that for the first word in a sequence the output phoneme nodes are zero and no weights are changed.

iii. The word nodes (particularly the ith word node) are excited by the input phoneme nodes and suppressed by the competitive filter nodes (generally only the $(i - 1)^{th}$ word node is suppressed). The weights from the context layer to the ith word node are learned as in Eq. (5). See Eqs. (1) and (2) for how activation levels are updated. Note that for the first word in a sequence this is the only learning that takes place.

iv. The competitive filter nodes are updated several times. The hard wired connections from the word nodes and strong lateral competition result in the node connected to the most active word node (the ith) dominating and suppressing all the others.

3. The output phoneme nodes are excited by the winning competitive filter node (i.e., the only active competitive filter node) to show the phonemic activity of the ith word.

4. Decay. The temporary weights (between the output and input phoneme layers and the context and word layers) decay by factor Δ for each output phoneme excited in stage 3.

5. Return to 2(i) for the next word.

- Recall/rehearsal (deliberately similar to presentation):

1. Initialization. The word nodes are reset to zero, the context nodes are reset to

the initial state they had in step (1) of the presentation.

2. Recall/rehearsal of the ith word:

i. The input phoneme nodes are excited by the chaining weights from the output phoneme nodes. There is "cuing" for the first word in the list; i.e., the input phoneme nodes are set to the phonemic activity of the first word. The context nodes replicate the state of activity they had in step 2(i) of the presentation. (The rationale for cuing the first word and the context states is that these processes occur outside the articulatory loop and without significant error. That is, we assume the phenomena we are trying to explain are not due to errors in cuing the loop at recall, but in the temporary associations and in the mechanism we have chosen for the selection of each item within the loop itself).

ii. The "chaining" weights from the output phoneme layer to the input phoneme layer are relearned. The output phoneme nodes are still active from the output of the $(i - 1)^{th}$ word. (For the first word the output phoneme nodes are still active from the last word in the presented list).

iii. The word nodes are excited by the input phoneme nodes and context nodes and suppressed by the competitive filter nodes (generally only the $(i - 1)^{th}$ word node is suppressed).

iv. The competitive filter nodes are updated several times.

3. The output phoneme nodes are excited by the winning competitive filter node (i.e., the only active competitive filter node) to show the phonemic activity of the ith word.

4. Decay. The temporary weights (between the output and input phoneme layers and the context and word layers) decay by a factor Δ for each phoneme outputted in stage 3.

5. Return to 2(i) for the next word.

6. Return to 1 for the next rehearsal.

Performance and Evaluation

In this section we compare the performance of the model with the main features of human performance outlined earlier. Note that the simulations repeated in this section are examples of the behavior of the model for certain values of F_{ph}, Δ, etc. We have a more general understanding of the dependence of its behavior on these and other parameters than is shown in these few examples. Thus we have derived approximate mathematical expressions for the probability of an error between two items and for span as a function of the parameters. Extensive simulations showed good agreement with these mathematical expressions (see Burgess, 1990, for a full description).

The most basic property of a model of short-term memory is its capacity. We begin by examining the word span of the model (how long a list can be correctly recalled), giving regard also to the length and similarity of the words used. We then consider the frequencies of errors made at the different serial positions in the list. This also depends on the phonemic similarity of the words. We also examine the types and relative proportions of errors made by the model. We consider chiefly only the errors made in the first rehearsal, i.e., in "immediate recall." The occurrence of errors during further rehearsals is briefly discussed at the end of this section.

The decay of temporary weights in the model depends on the total number of phonemes in the list (M). We expected this to determine the model's span, in accordance with evidence that the span for human short-term memory is determined by the time taken to articulate the list (given our approximation that all phonemes take the same time to articulate). Forgetting will occur when the difference in activation be-

tween the correct word node and competing word nodes is of the same order of magnitude as the noise.

In the simulations that are reported, the noise parameter, σ, was adjusted to give a span of seven letters (of length two phonemes) so as to establish a point of correspondence with the general level of human performance. This was done for $F_{ph} = 0.5$, i.e., equal influence of chaining and contextual associations to the word layer. As a result, the noise parameter was set at $\sigma = 0.03$. Thereafter, simulations were run across the whole range of values of F_{ph} to explore the influence of chaining on the performance of the model. In the following sections, data are in general presented for simulations with $F_{ph} = 0.5$, with additional data for simulations with $F_{ph} = 0.02$ (i.e., almost no chaining) and $F_{ph} = 0.98$ (almost "pure" chaining), where the differences in performance are considered important.

Capacity: Effects of List Length, Word Length, and Item Similarity

Figure 5 shows the probability of a whole list being correctly recalled (on the first rehearsal) as a function of list length. Lists were random selections from a set of "dissimilar" letters (b f j i o u r t n h y q v s), or "similar" letters (b c d g p v t f l m n s x z). The prelearned vocabulary was the set of all the letters of the alphabet. The curves were calculated from 200 trials of each list length; parameter values were $F_{ph} = 0.5$ and $\sigma = 0.03$.

We see that the "span" of the model (the list length at which half of the lists are correctly recalled) is greater for dissimilar letters than for similar letters. The reduction due to similarity is 33% which appears to be slightly greater than in humans. For example, in a similar but not identical comparison, Schweickert et al. (1990) reported a difference of 20%. Note also that the dissimilar letters are "shorter" than the similar letters. The 14 dissimilar letters contained a total of 24 phonemes, whereas the similar letters contained 30. However, this small difference in word length is too slight to provide an adequate explanation of the phonemic similarity effect (cf. also Schweickert et al., 1990).

The span for dissimilar words of exactly two phonemes is best shown in Fig. 6. Because we are approximating the time taken to articulate a word by the number of phonemes it contains, the phonemic word length is important in terms of model span. Figure 6 shows that word span does have this type of dependence on phonemic word length. It shows the probability of correctly recalling an item from a list of words as a

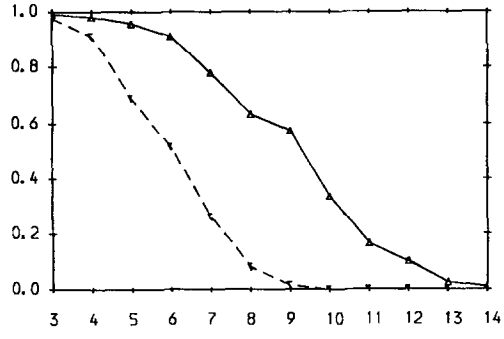

FIG. 5. The probability of correctly recalling (on the first rehearsal) a list of N letters versus list length N. Lists were selected from the letters b f j i o u r t n h y q v s (full line) and b c d g p v t f l m n s x z (dashed line). There were 200 trials of each list length, $F_{ph} = 0.5$ and $\sigma = 0.03$.

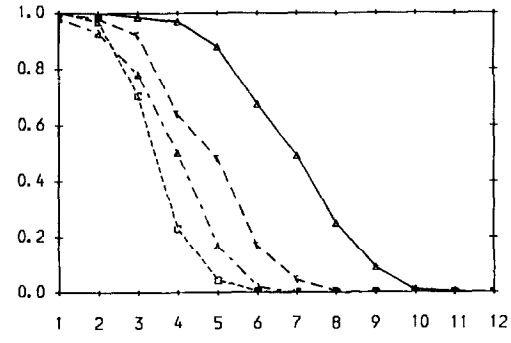

FIG. 6. The probability of correctly recalling a list of N "random words" versus list length N. A "random word" is a random selection of n phonemes. The phonemic word lengths n are 2 (full line), 3 (dashed line), 4 (dash-dot-dash line), and 5 (dotted line). There are 200 trials for each word length, $F_{ph} = 0.5$ and $\sigma = 0.03$.

function of word length. Each list was comprised of "words" made from n randomly selected phonemes. Lists were chosen from a pool of 14 of these "random words," the prelearned vocabulary consisted of 26 "random words" (including the 14 used in lists). The number of phonemes per word was varied between two and five. Performance declines noticeably with word length in a way that looks remarkably similar to human data (Baddeley et al., 1975, Fig. 1).

The data in Fig. 6 could be interpreted in terms of articulation rate: simulations involving five-phoneme words correspond to an articulation rate 2.5 times slower than those involving two-phoneme words. Figure 7 shows the relationship between span, defined as the list length at which 50% of recalls are correct, and articulation rate using the data in Fig. 6. The linearity of this relationship corresponds well with equivalent plots of psychological data (Baddeley et al., 1975; Hulme et al., 1984; Hitch et al., 1989). Figure 7 shows that the model's span s increases proportional to the speech rate r. Assuming a speech rate of about 12 phonemes per second as a rough estimate, the constant of proportionality is approximately 1, which compares well with values of between 1 and 2 from the psychological studies. In fact the simulations show $s \approx r + 1$, in close agreement with the equation reported by Hitch et al. (1991).

Note that the total number of phonemes in the list is not the sole factor that determines span. Figure 8 shows the same curves as Fig. 6 plotted as a function of the total number of phonemes in the list. The performance on long lists of short words is a little worse than that on short lists of long words even where both lists contain the same total number of phonemes. This is because more words means a greater number of excited word nodes, which increases the possibility of a noise-induced error.

Interestingly, there may be a parallel for such effects in human data. Zhang and Simon (1985) found that while span was related to the time to articulate individual items, there was a second factor (which they interpreted as the time taken to access each "chunk"). The direction of this effect was the same as in Fig. 7. However, our model suggests an alternative interpretation of the mechanism responsible.

In summary the model reproduces the psychological data on memory span and its susceptibility to the effects of phonemic similarity and word length. In all three cases the model shows effects of the right type with the same order of magnitude as the human data. Note that the span of the model is arbitrary and is determined by choosing a suitable level of noise (i.e., σ). However, once this parameter has been set, the phonemic similarity and word

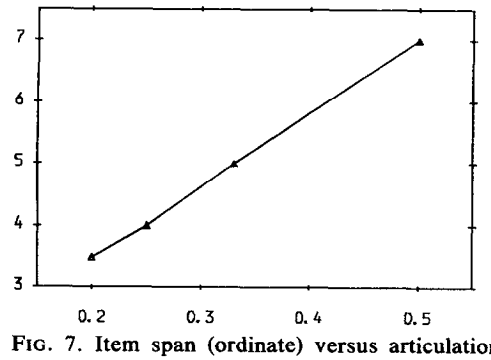

FIG. 7. Item span (ordinate) versus articulation rate. The data for random words in Fig. 6 is replotted to show the effect on span of the number of words that can be articulated per time step (= 1/the number of phonemes per word).

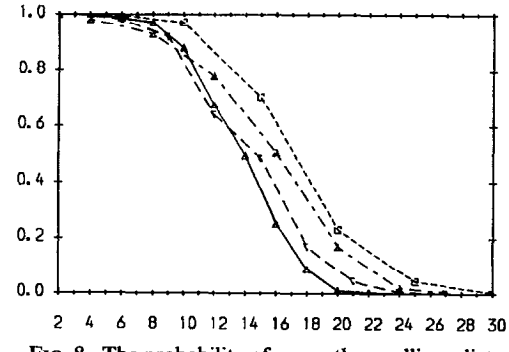

FIG. 8. The probability of correctly recalling a list of N "random words" (each of n phonemes) versus the total number of phonemes in the list $N \times n$. The data is the same as in Fig. 6: full line, $n = 2$; dashed line, $n = 3$; dash-dot-dash line, $n = 4$; dotted line, $n = 5$.

length effects, particularly the encouraging dependence of recall on speech rate, are emergent features of the model.

Serial Position and Item Similarity

Figure 9 shows the probability of correct recall (on the first rehearsal) as a function of serial position. The letters b f j i o u r t n h y q v s were used, and the data is from the simulation shown in Fig. 5. The serial position curves clearly show primacy and no recency.

The model also shows primacy when it is running with $F_{ph} = 0.02$ (i.e., the word nodes are driven almost entirely by the context layer) or with $F_{ph} = 0.98$ (i.e., the word nodes are driven almost entirely by the phoneme layers). This is because an error near the beginning of a list (i.e., a later item being recalled too soon) tends to increase the chance of subsequent errors. Thus even if spontaneous errors occurred uniformly throughout the list the increased chance of "knock on" errors would ensure primacy. This effect is very robust when the word nodes are driven purely by the phoneme layers ($F_{ph} = 0.98$). The chaining of the phoneme layers results in serial position curves that decrease progressively in nearly all trials. This is a fundamental problem with excessive chaining since an error anywhere in the sequence will throw out everything that follows. When $F_{ph} = 0.02$ there is much greater individual variation in serial position curves, although the averaged curve also shows primacy of similar downward slope. In a later section we show how a simple alteration of the way contextual information is encoded can result in recency when the model is run without chaining.

We also ran the model with the same parameters as in Fig. 9, but with lists chosen from the letters b c d g p t v f l m n s x z. Half of these letters share the phoneme "ee," and half the phoneme "e," but there are also other shared phonemes. Thus the phonemic similarity encountered by the model was much greater than in Fig. 9. Consistent with the span data (see Fig. 5), performance is much decreased, as shown in Fig. 10. The reduction is about 14% for five-item lists, 24% for six items, and somewhat more for longer lists. Typical human data show reductions of the order of 22% for five-item lists (Peterson & Johnson, 1971, Experiment 1) and 37% for six items (Baddeley, 1968, Experiment 5). The agreement is therefore quite close here, bearing in mind that the model was run with a slightly lower degree of phonemic similarity

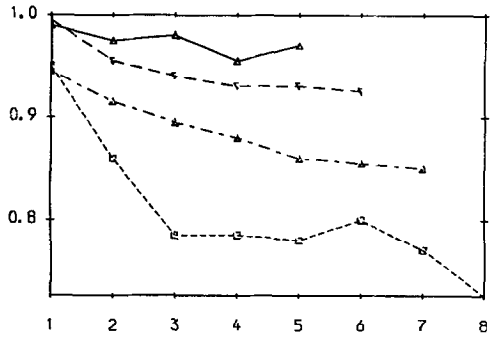

FIG. 9. The probability of correctly recalling an item from a list of N of the letters: b f j i o u r t n h y q v s, versus the serial position of the item. The values of N are 5 (full line), 6 (dashed line), 7 (dash-dot-dash line), and 8 (dotted line). The data are the same as in Fig. 5.

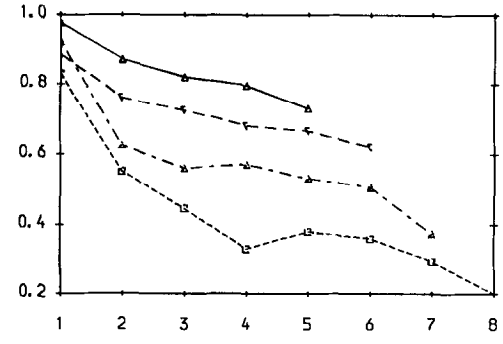

FIG. 10. The probability of correctly recalling an item from a list of N of the letters: b c d g p t v f l m n s x z versus the serial position of the item. Note the phonemic similarity of these items. The list length N is 5 (full line), 6 (dashed line), 7 (dot-dash-dot line), and 8 (dotted line). There were 200 trials for each list length, $F_{ph} = 0.5$ and $\sigma = 0.03$.

than was experienced by the human subjects. It is interesting to note the similarity in the shape of serial position curves for lists of similar and dissimilar letters. This is also a feature of human performance (Baddeley, 1968).

Table 1 shows an analysis of substitution errors and indicates the proportion of times the substitutions shared a phoneme with the correct letter. This was done for the simulations shown in Fig. 10. The expected proportions of errors if substitution occurs at chance are also shown (i.e., assuming that errors are independent of phonemic similarity). It can be seen that, when errors are made, similar items tend to be confused with each other. Note that c shares phoneme "s" with s and x, also d shares phoneme "d" with z. The letter e occurred as an intrusion error almost as frequently as letters from the list, accounting for the greater difference in proportions of "similar" and "different" errors shown by the letters containing phoneme "ee" (b c d g p t v) than those containing "e" (f l m n s x z).

The similarity effect is clearly shown. However, the effect of chaining is that the similarity of two items has as much effect on their immediate successors in the list as on themselves. This partially obscures the similarity effect in the confusion matrix (Table 1). It also prevents the modeling of experiments involving alternating similar and different items (see below).

Types and Proportions of Errors Made

We counted the number of various types of errors made by the model in simulations with F_{ph} = 0.02, 0.5, and 0.98 to illustrate the different effects of the context and phoneme layers on the word layer. The results for lists of phonemically similar and dissimilar items are shown in Table 2. Apart from a very small number of repetition errors all the errors made were order errors or extra-list intrusion errors. There were no "don't knows" since the competitive filter always selected an item for output.

Note first that performance for F_{ph} = 0.5 is much better than for either F_{ph} = 0.02 or F_{ph} = 0.98. This is interesting in that it demonstrates a natural advantage of having two different mechanisms controlling recall. An error in either system alone may be tolerated; the incidence of two different types of error occurring coincidentally for the same item is lower than for either error alone. Table 2 also shows that errors increase with both list length and phonemic similarity for all values of F_{ph}. Furthermore, order errors consistently form the majority of errors, ranging between 74% and 88% for F_{ph} = 0.5. This is in agreement with estimates of around 75% for human

TABLE 1

THE PROPORTION OF TIMES A LETTER WAS SUBSTITUTED BY A SIMILAR LETTER (ONE WITH A PHONEME OR PHONEMES IN COMMON) IN THE ERRORS MADE IN THE RECALL OF LISTS OF 5 TO 8 OF THE LETTERS: b c d g p t v f l m n s x z

Type of error	Input letter													
	b	c	d	g	p	t	v	f	l	m	n	s	x	z
Similar	.59	.73	.73	.65	.63	.61	.55	.55	.53	.53	.59	.71	.63	.62
Different	.41	.27	.27	.35	.37	.39	.45	.45	.47	.47	.41	.29	.37	.38
Chance														
Similar	.46	.62	.54	.46	.46	.46	.46	.46	.46	.46	.46	.54	.54	.54
Different	.54	.38	.46	.54	.54	.54	.54	.54	.54	.54	.54	.46	.46	.46

Note. Also shown are the proportions expected if substitution occurred by chance. The data are from the simulations in Fig. 10, F_{ph} = 0.50, σ = 0.03.

TABLE 2
THE DIFFERENT TYPES AND PROPORTIONS OF ERRORS PRODUCED BY THE MODEL AS A FUNCTION OF LIST LENGTH, ITEM SIMILARITY, AND F_{ph}

		Phonemically dissimilar letters list length					Phonemically similar letters list length				
F_{ph}	Error type	5	6	7	8	Mean	5	6	7	8	Mean
0.02	Item	1.2	3.0	5.9	9.4	4.9	4.6	9.8	11.7	15.6	10.4
	Order	60.8	71.3	95.5	107.7	83.8	85.6	101.5	108.9	117.7	103.4
0.50	Item	0.6	2.7	6.0	10.1	4.9	4.8	12.5	17.3	21.7	14.1
	Order	4.6	8.1	16.9	28.5	14.5	27.6	43.0	66.0	93.6	57.6
0.98	Item	7.6	11.0	18.0	23.9	15.1	19.6	26.0	27.0	25.0	24.4
	Order	8.0	20.8	33.0	50.7	28.1	38.8	66.3	99.0	118.3	80.6

Note. The values shown are the number of errors/list length during 200 trials, i.e., the number of times that an item is incorrectly recalled averaged over serial position, $\sigma = 0.03$.

recall under similar conditions (Aaronson, 1968; Bjork & Healy, 1974). The main effect of phonemic similarity is to increase order errors, again in good agreement with the psychological data (Wickelgren, 1965c). However, there is also a slight increase in item errors, which is unlike the data from human subjects. As is to be expected the effect of phonemic similarity increases as the amount of chaining, given by the parameter F_{ph}, increases. However, phonemic similarity has a slight effect even for $F_{ph} = 0.2$ because the effect of the phoneme layer on the word layer is of the same order of magnitude as (although smaller than) the noise.

The absence of "don't know" responses (omissions) is not characteristic of human behavior, it is a consequence of the fact that when the competitive filter has reached a steady state, it has always chosen a word node (the most strongly activated). A simple way of producing omissions is to apply a threshold to the output from the competitive filter such that the response is "don't know" if the activation of the winning word node is below the threshold. Further simulations illustrate the effects of this modification for different thresholds (see Table 3). Interestingly, while omissions tend to increase towards later serial positions, commission errors (which are mainly order errors) show some evidence of recency. This suggests that the model is capable of producing recency given suitable modification (see below). Note that for very high thresholds the overall accuracy of recall drops sharply because the activation of correct winners sometimes falls below threshold.

We can also investigate the difference in serial position of items that are confused in recall. In human data the majority of order errors involve an item being substituted by an adjacent or very near item (Healy, 1974). In Fig. 11 we show the distribution of the separations between pairs of items whose positions are transposed in simulations with $F_{ph} = 0.02$, 0.5, and 0.98. For example, recall of the sequence a b c d e f g h as a b c f e d g h would correspond to a separation of two positions. Figure 11 gives us an indication of the range in serial position over which paired transposition errors occur.

When the context layer alone is driving the word layer ($F_{ph} = 0.02$), paired transpositions tend to be between adjacent items because of the temporal correlation in the context layer. However, when the input phoneme layer alone is driving the word layer ($F_{ph} = 0.98$) there is no such bias towards nearby items. The model running with both types of input to the word layer

TABLE 3
EFFECT OF APPLYING DIFFERENT THRESHOLDS TO THE OUTPUT OF THE COMPETITIVE FILTER TO PRODUCE OMISSION ERRORS

Threshold		Serial position 1	2	3	4	5	6	7	Proportion of lists correct
0.38	Omissions	—	—	—	.01	.01	.01	.03	
	Commissions	.03	.10	.18	.17	.16	.17	.16	.81
0.40	Omissions	—	—	—	.02	.02	.04	.06	
	Commissions	.03	.10	.18	.16	.16	.15	.13	.81
0.42	Omissions	.02	.01	.01	.05	.05	.07	.12	
	Commissions	.03	.11	.18	.14	.14	.13	.08	.76
0.44	Omissions	.24	.09	.08	.10	.14	.17	.22	
	Commissions	.01	.09	.17	.13	.09	.11	.03	.46

Note. The values shown are mean probability correct from 200 trials (F_{ph} = 0.05, σ = 0.03).

(F_{ph} = 0.50) shows only a slight increase in the incidence of paired transpositions between adjacent items.

The root cause of the large proportion of well separated paired transposition errors and the higher incidence in omissions towards later serial positions is decay. If there are decaying weights or activities in the phoneme or context layers then there will be a tendency for a word wd_{err} late in the list to be substituted for a word wd_{cor} early in the list. At the time of recall for wd_{cor} the temporary weights that store the associations necessary for its excitation will have decayed by much more than those for wd_{err}. Thus if any of the context or phoneme nodes that excite wd_{cor} also excite wd_{err} (because of correlations in the states of the context layer or item similarity) they will do so via much stronger weights. In fact for very strong decay (Δ = 0.5, say) this effect is so strong that lists can be recalled in reverse order! Clearly decay must be in the model, as the information stored is temporary. Furthermore, it is interesting to note that in free recall of very long lists the subject can often remember only the last few items (see, e.g., Murdock, 1962). However, some modification of the model is indicated, as described later.

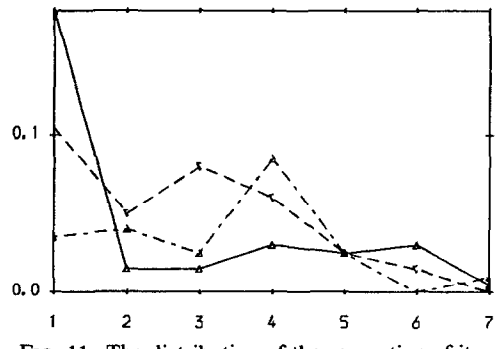

FIG. 11. The distribution of the separation of items involved in paired transpositions. The data are from the same simulation as Table 2. Lists were of eight dissimilar letters, F_{ph} = 0.02 (full line), F_{ph} = 0.50 (dashed line), and F_{ph} = 0.98 (dot-dash line).

In summary, the model is capable of producing the same types of errors as humans. It does well in that there are typically more order errors than item errors and in that phonemic similarity has its effect chiefly on order errors. Both of these are robust features of the model. Paired transpositions are also relatively common, but they tend to involve adjacent items only when the amount of chaining is minimal. A major problem is the unnaturally large separation of items involved in some paired transpositions even when there is little chaining.

Articulatory suppression. We interpret

interference with subvocal rehearsal as preventing the use of the chaining weights. Thus only the context layer will affect the word layer, as input phoneme nodes will no longer be excited. We simulated suppression by running the model with the excitation from context as if $F_{ph} = 0.5$, but with $F_{ph} = 0.02$. Figure 12 shows the probability of correctly recalling lists of either phonemically similar or dissimilar letters as a function of list length. In this simulation the parameters of the model were the same as for the equivalent "no suppression" simulations illustrated in Fig. 5. Span is reduced to less than half its normal value because excited word nodes are typically at half their normal activation and because the effect of using only the context to drive the word layer also leads to lower performance (as illustrated in Table 2). The phonemic similarity effect largely disappears and this is because of the virtual inactivity of the input phoneme layer. These results closely resemble human data for the recall of visually presented sequences under suppression (Estes, 1973; Murray, 1968; Peterson & Johnson, 1971).

Figure 13 shows the probability of correct recall versus list length for lists of items of different word lengths. These "suppression" results can be directly compared with the equivalent "no suppression" simulations shown in Fig. 6. The word length effect is reduced under suppression but persists because weights from the context layer decay according to the time taken to articulate the list. Human data show that suppression eliminates the word length effect for visually presented stimuli (Baddeley et al., 1975).

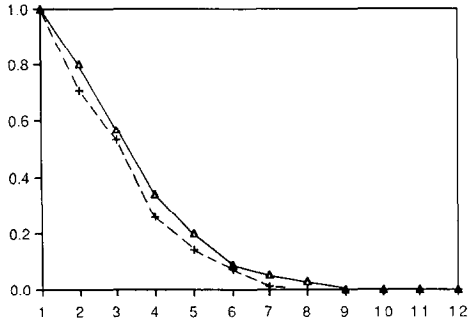

FIG. 12. The probability of correctly recalling a list of N phonemically similar letters (dashed line) or dissimilar letters (full line) under articulatory suppression. (Letters as in Fig. 5.) There were 200 trials at each list length, $\sigma = 0.03$. For other parameters see text.

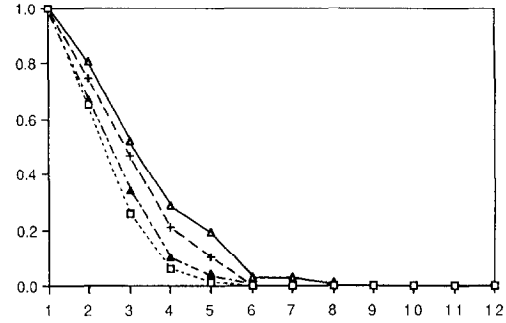

FIG. 13. The probability of correctly recalling a list of N random words under articulatory suppression as a function of word length. The phonemic word lengths are 2 (full line), 3 (dashed line), 4 (dash-dot-dash line), and 5 (dotted line). There were 200 trials for each list length, $\sigma = 0.03$. For other parameters see text.

The model is therefore partially, but not completely, successful in accounting for the effects of articulatory suppression on recall. As in humans, performance is disrupted, the phonemic similarity effect almost disappears, and the word length effect is markedly reduced. However, unlike in humans, a residual word length effect remains. It would be a simple matter to improve the performance of the model by altering the way that weights from the context layer decay. However, more realistic behavior under articulatory suppression can be expected from changing the architecture of the model as part of a set of general improvements, as described later in this paper.

Further Properties of the Model

In this section we briefly examine the performance of the model in tasks beyond those set out as its primary objectives.

More subtle similarity experiments. There is an interesting experiment by Bad-

deley (1968) in which subjects were tested on lists of alternating phonemically similar and dissimilar items (e.g., a b f c q d). Performance was worse only for the similar items in the list giving a serial position curve that zigzags between the serial position curves for all similar or all dissimilar items (see Fig. 14). We performed the equivalent simulation for 500 trials of lists of six dissimilar, similar, and alternating items (see Fig. 15). The effect seen in Baddeley's experiment was not shown; the serial position curve for alternating similar and dissimilar items lies between those for all similar or all dissimilar items but does not zigzag.

The reason for this limitation of the model is that when two items are phonemically similar there is as much chance of error between the two items following them in the list as between themselves (see Burgess, 1990). For example, with the list a b f c q, the similarity of b and c results in the same increased chance of error in recalling f and q as in recalling b and c. This happens because all phonemic information is stored in the phoneme layers which are a self-contained *chaining* system—there is no contextual input into the phoneme layers, only to the word layer.

In the above example, at the recall of b the word node b is fully activated (to activation A_{wd}) by the input phoneme nodes

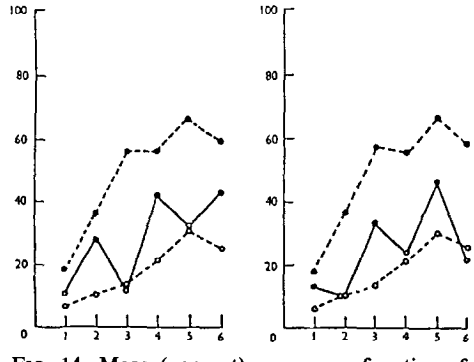

FIG. 14. Mean (percent) errors as a function of serial position. Filled circles represent acoustically similar letters, open circles dissimilar letters. Presentation was visual, taken from Baddeley (1968).

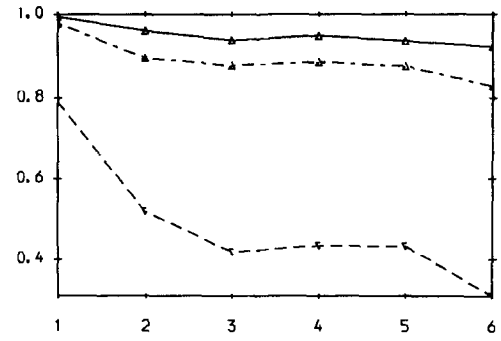

FIG. 15. Serial position curves for six similar (full line), dissimilar (dashed line), or alternately similar and dissimilar items (dot-dash line). There were 500 trials, $F_{ph} = 0.5$ and $\sigma = 0.03$.

"b" and "ee" but word node c is also activated (to $A_{wd}/2$) by "ee." At the recall of f the input phoneme nodes "e" and "f" will be fully activated (to A_{ph}) by the chaining weights from output phoneme nodes "b" and "ee." However, input phoneme nodes "kh" and "yu" will also be excited (to $A_{ph}/2$) by output phoneme node "ee." Thus word node f is fully activated (to A_{wd}) and word node q is activated to $A_{wd}/2$. Hence the presence of noise will be as likely to produce errors where q is substituted for f as errors where c is substituted for b.

This view is supported by the separations of items in paired transposition errors. A large proportion of errors in the simulations with alternating phonemically similar and dissimilar items were transpositions of items 2 or 4 positions away from each other, i.e., transpositions between similar items or between dissimilar items. For the simulation of 200 lists of six letters there were 67 paired transpositions in total, 49 of which were between items of separation two. In a similar simulation of 200 lists of eight letters, 165 out of 194 paired transpositions were between items of separation two or four.

Rehearsal. We did not systematically investigate the behavior of the model after many rehearsals in this work. Suffice it to say that to a first approximation we expect errors to occur in each rehearsal with approximately the same frequency as in the

first rehearsal. After enough rehearsals have been made for several errors to have occurred, performance will probably decline more quickly. Table 4 shows 10 rehearsals of lists of five and eight letters.

To be more compatible with human rehearsal we would like to make the occurrence of errors decrease after the first rehearsal (e.g., see Heffernan, 1991) and curb the tendency of w to appear as an intrusion error. The letters e and w are the most frequent extra-list intrusions because they so often share a phoneme with other letters (see Table 1). Furthermore, we know that rehearsal in short-term memory tasks can involve complex strategies such as grouping and cumulative repetition. In its present form the model can only rehearse an entire list an integer number of times.

Conclusions

In this section we evaluate the performance of our model with respect to the aims we had when starting. Successes and limitations are noted; simple modifications are suggested to address each limitation individually. Finally we identify and discuss two or three major flaws in our model, some of which necessarily arise from deficiencies in the articulatory loop idea. Further changes in the model are proposed to allow these limitations to be addressed simultaneously.

TABLE 4
An Example of 10 Rehearsals of the Lists of 5 and 8 Letters

o	s	b	n	r	o	s	b	n	r	f	h	t
o	s	b	n	r	o	s	b	n	r	f	h	t
o	s	b	n	r	o	s	b	n	r	f	h	t
o	s	r	n	b	o	s	b	n	r	f	h	t
o	s	b	n	r	h	t	o	r	f	b	n	c
o	s	b	n	r	h	t	o	c	n	b	f	w
o	s	b	n	r	h	t	n	b	w	x	f	c
o	s	b	n	w	b	w	h	c	x	n	l	f
o	s	w	n	g	b	w	h	n	x	c	t	w
o	s	w	n	g	w	t	x	p	k	c	w	n
o	s	w	n	b	w	t	x	k	z	c	w	n

Note. $F_{ph} = 0.5$, $\sigma = 0.03$.

Successes

Points 1–4 are robust features of the model, found regardless of the amount of chaining.

1. Span. The probability of successfully recalling a list shows the right dependence on list length showing a dramatic fall-off around span (cf. Guildford et al., 1925).

2. Phonemic similarity effect. The use of similar items reduces performance as in humans (see, e.g., Baddeley, 1968; Peterson & Johnson, 1971; Schweickert et al., 1990). Further, when an error occurs, there is a tendency for the erroneous item to be phonemically similar to the (correct) item it replaces. This is also a characteristic of human performance (Conrad, 1964).

3. Word length. We made the approximation that the time taken to articulate a word is proportional to the number of phonemes it contains. Within this approximation the model shows performance decreasing with increasing word length. It is interesting to note that the simulations show a linear dependence of span on articulation rate (as determined by the number of phonemes per letter) with a slope of about 1, thus modeling the human data quite closely (Baddeley et al., 1975; Hulme et al., 1984; Hitch et al., 1989). It was noted that capacity is not entirely determined by the total articulation time of a list. However, this unanticipated effect is also found in psychological data (Zhang & Simon, 1985).

4. Serial position curve. The model shows a strong primacy effect. This is a common feature of the serial position curves for visual and auditory presentation (Crowder, 1972).

5. Types of error. The model produces the same categories of error as humans (i.e., order errors, item errors and, with a simple modification, omissions). Most of the errors are order errors (Aaronson, 1968; Bjork & Healy, 1974), and the effect of phonemic similarity is chiefly on order errors, as in human data (Wickelgren, 1965c). The model produces paired transpositions in re-

call and, provided there is not too much chaining, these tend to involve adjacent items, as in human subjects (Conrad, 1964).

6. Articulatory suppression. We interpret the prevention of subvocal rehearsal as preventing the use of the chaining weights. As with human data (at least for visual presentation) the effect of suppression is that span is reduced and the phonemic similarity effect disappears. However, the word length effect is not completely lost when there is articulatory suppression because the weights from the context layer decay per phoneme, and not, for example, per word.

We take these successes as suggesting that, despite its many simplifying assumptions, our network model of phonological short-term memory is a reasonable first approximation. We interpret these successes as confirming the utility of competitive queuing as a mechanism for serial output, and our assumption that short-term memory phenomena can be captured in terms of the effects of noise on a system of rapidly decaying temporary associations.

Limitations of the Model

1. Absence of recency. Human data shows a small recency effect for visual presentation (Crowder, 1972), but our model shows none at all.

2. Articulatory suppression. The model does not fully simulate the effects of suppression on memory for visually presented sequences. Furthermore, since there is no difference between oral and vocal presentation in the model the effects of articulatory suppression are necessarily the same for both modes of presentation. As described in the introduction, articulatory suppression effects depend on presentation modality. To model these differences we must explicitly model the two modes of presentation.

3. Lists of alternating similar and dissimilar words. Our model cannot show the zigzag in levels of recall observed by Baddeley (1968). This is because the chaining nature of the phoneme nodes means that the acoustic similarity of two items is as likely to cause an error in the recall of words immediately after them as in their own recall.

4. The separation of items involved in order errors. Because of the decaying nature of the temporary weights there is a tendency for items late in the list to replace items early in the list. If there is any correlation in the state of the context layer for early and late items (or if the items preceding them share phonemes) then, even early on in recall, later word nodes may be more excited than early ones purely because their temporary weights have decayed less.

5. Order errors in structured lists. In experiments using lists of items that are grouped in some way there is a tendency for paired transpositions to be between items at similar positions within each group (see, e.g., Estes, 1985). Also, where subjects have had to learn many lists, extra-list intrusions tend to be an item from the same serial position in a previous list (Conrad, 1960). In our model all lists are homogeneous and all temporary information decays to zero between list presentations.

Suggested Modifications

The main lessons that we draw from the limitations of this first attempt to model the articulatory loop are as follows:

1. We have stored the short-term associations necessary for the serial recall of a list in connections whose strength decays with time. (These temporary associations were between the context and item layers and the output and input phoneme layers.) Because of this, any correlation between temporally well separated states in the context or output phoneme layers can lead to items from the end of the list replacing those at the beginning, purely because they are excited via connection weights that have decayed less. This defect shows itself in the unusually wide separation of items involved in order errors. This in turn makes it

unlikely that the model could ever show recency; if errors tend to be long-range then there is little chance that recall of the last item could remain undisturbed by errors earlier in the list.

To remedy this behavior we can use context states in which there is zero correlation between temporally well separated states. Further, because correlations in the output phoneme layer (determined solely by items' phonemic similarity) are independent of item separation, this layer should not be used to excite the next item in the list (as occurs through the phonemic chaining mechanism we used).

We can test the effect of these changes using the present model by putting F_{ph} = 0.02 and using activation states in the context layer with nonzero correlation only for temporally adjacent or very nearby states. A simple set of states with these properties is shown in Table 5.

For the context states shown in Table 5, four context nodes out of six remain active at successive times. Thus the correlation with the state at time 1 is 0.66 for the state at time 2, 0.33 for the state at time 3, and 0.0 for states at all other times. Note that (similarly to the context states used before) the context states for the first and last items have nonzero correlation with half as many other states as do the context states for the items in between; i.e., they are twice as "distinctive." Murdock (1960) demonstrated how the distinctiveness of different

TABLE 5
A SET OF CONTEXT STATES WITH NONZERO CORRELATION ONLY FOR STATES AT SIMILAR TIMES

Time	Activation of context nodes
1	* * * * * * 0 0 0 0 0 0 0 0 0 0 . . .
2	0 0 * * * * * * 0 0 0 0 0 0 0 0 0 . . .
3	0 0 0 0 * * * * * * 0 0 0 0 0 0 0 . . .
4	0 0 0 0 0 0 * * * * * * 0 0 0 0 0 . . .
.	.
.	.
.	.

Note. * indicates an activation of Act = 0.5.

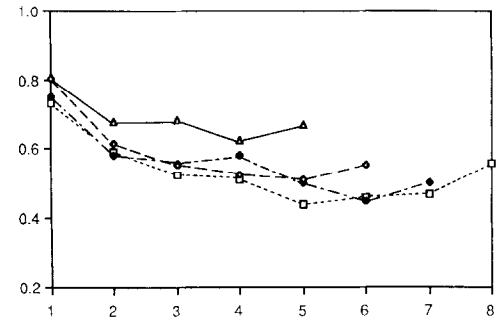

FIG. 16. The probability of correctly recalling an item from a list of N of the letters: b f j i o u r t n h y q v s, versus the serial position of the item. The context states shown in Table 5 were used. The values of N are 5 (full line), 6 (dashed line), 7 (dash-dot-dash line), and 8 (dotted line). There were 200 trials for each list length, F_{ph} = 0.02 and σ = 0.03.

serial positions could contribute to serial position effects in recall.[1]

The serial position curves (for F_{ph} = 0.02 and the context states described above and the set of "dissimilar" letters: b f j i o u r t n h y q v s) show modest recency for the last item (see Fig. 16) reminiscent of psychological data for visual presentation (see Fig. 2). The recency effect is caused by the extra discriminability of the context state of the last item, the effect of which is not swamped by long-range disturbance from errors earlier in the list. The majority of order errors occur between adjacent items (see Fig. 17).

Simulations with F_{ph} = 0.5 also show slight recency for the last item (i.e., less than in Fig. 16) and a majority of order errors between adjacent items (see Fig. 17). When F_{ph} = 0.5 and the set of "similar" letters: b c d g p v t f l m n s x z are used, recency is no longer shown—the number of errors due to phonemic similarity swamps the effect of the extra discriminability of the last item.

2. The experiments involving alternating similar and dissimilar words provide evidence that the phonemic component of an item should not be used to cue the recall of

[1] We are grateful to D. Hintzman for bringing this to our attention.

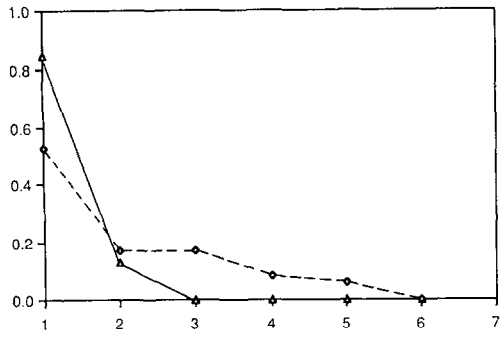

FIG. 17. The distribution of the separation of items involved in order errors. The context states as shown in Table 5 were used, $F_{ph} = 0.02$ (full line), $F_{ph} = 0.50$ (dashed line).

the next item. While it seems natural to use the output of one item to trigger the recall of the next, this should be independent of (a) the acoustic properties of the outputted item and (b) whether it was the correct item or a mistake. Acoustic similarity effects indicate that we must retain a phonemic component to the model, and the excitation of item nodes by the corresponding input phoneme nodes still seems natural.

Thus the architecture we propose for future modeling would have a "loop" in which activation spreads from context layer to input phoneme layer to item layer to competitive filter which then triggers the activation of the next context state, see Fig. 18. Note that we would take the output phoneme layer out of the loop and put the context layer in (removing explicit item–item chaining). The output phoneme layer would be concerned only with the articulatory details of output, either vocal or subvocal. Note also that the only temporary associations learned would be those between the context and input phoneme layers.

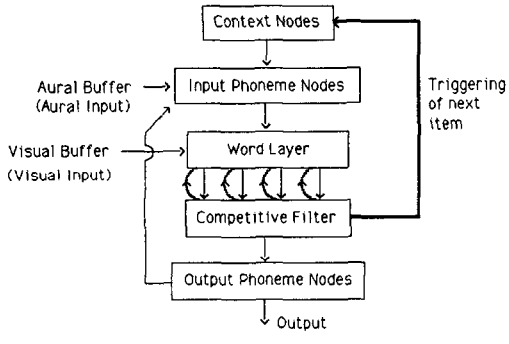

FIG. 18. The modified architecture proposed for an improved model of the articulatory loop.

3. Our first step towards modeling aural and visual presentation separately would be to include an aural input buffer that serves to excite the input phoneme layer and a visual input buffer that serves to excite the item layer directly (see Fig. 18). The motivation for this is for aural input to be "heard" by the input phoneme layer and visual input to be "recognized" by the item layer without being heard.

Visually input items must excite the input phoneme layer for the temporary weighted connections from the context layer to be learned (so that the sequence can be recalled by the loop). This could occur by one-to-one excitatory connections from output phoneme nodes to input phoneme nodes (i.e., output is also "heard" by the input phoneme layer). Thus, by exciting an item node and, in turn, the corresponding competitive filter node, input phoneme nodes corresponding to a visually presented item can be excited. We would then interpret the effect of articulatory suppression as occupying the output phoneme layer with a redundant articulatory task—preventing visually presented items from entering the loop at all and lessening the activation input phoneme nodes (and, hence, connection weights from the context layer) for aurally presented items.

The effect of connections from output to input phoneme layers during recall and rehearsal will need investigating. It may be that we would have to assume that input from the context layer has preferential access to the input phoneme layer for the occurrence of an error not to affect the recall of the next item.

Relationship to Other Models

Our present simulation differs in interesting ways from other models of memory span and related tasks. Some of these mod-

els share the assumption that both chaining associations between successive items and associations between items and their contexts are stored (see, e.g., Shiffrin & Cook, 1978). However, models of a chaining nature have been proposed (e.g., Lewandowsky & Murdock, 1989) as have models which reject chaining and rely entirely on some equivalent of context–item associations (e.g., Lee & Estes, 1981). We consider pure chaining of items to be implausible. Our simulations provide good evidence that if a model contains chaining, the occurrence of errors must rely on some other mechanism, i.e., the way a chained activation vector is used to determine which item is actually output. We also know that chaining models ought to have great difficulty recalling sequences containing repeated items, whereas repetitions lead to only a slight disruption in human performance (Jahnke, 1969; Baddeley, 1990, personal communication).

An important feature of the present model is that the relative contributions of chaining and contextual associations can be varied. This has made it clear that simulations with little or no chaining come closest to reproducing human behavior, particularly in relation to order errors and the shape of the serial position curve. This outcome is of course fully consistent with Lashley's (1951) original arguments against chaining.

Our simulation differs from other models in a number of other ways. Most notably it attempts to model only short-term memory and is therefore of more restricted scope than, for example, TODAM (Lewandowsky & Murdock, 1989), which attempts to explain serial order phenomena in both short-term and long-term memory. Unlike TODAM, our model uses local, as opposed to distributed, representations of information. Of course the local units in our model would not necessarily represent local objects at a more detailed level of implementation.

Finally, it is interesting to note that our model has a certain similarity with that of Lee and Estes (1981). For example, errors in both models occur because of the effects of noise or perturbation. However, Lee and Estes (1981) use reverberatory loops to preserve order information and introduce noise into the timing of these loops, whereas in our model noise is introduced into activation levels. It seems that Lee and Estes favored noisy timing in order that the characteristic errors of their model would be order errors. The present simulation shows that this is not a necessary assumption.

The recent explosion of interest in parallel distributed processing and neural network modeling has led to much work in related fields, inspired by the discovery of the error back-propagation algorithm (see, e.g., Rumelhart, Hinton, & Williams, 1986) and the Hopfield model (Hopfield, 1982). While there is not room to review more of this vast and growing area of research than those bits we have already mentioned, the present approach (and that of some of the above models) holds a relevant lesson.

There are many possible ways to incorporate serial order into standard connectionist models (see, e.g., Jordan, 1986; Elman, 1990) using multilayer perceptrons and error back-propagation, or using Hopfield type networks (e.g., Kleinfeld, 1986). However, a model whose architecture is chosen, ad hoc, simply because it can address the problem of serial order is most unlikely to solve this problem in a human-like way. In the present paper we have a connectionist model whose architecture (i.e., its structure and dynamics) is inspired by psychological thinking—representing full consideration of the lessons to be drawn from experimental constraints. The successes of the model so far reflect the validity of these psychological ideas; its limitations indicate the need for further development of the architecture used to implement them.

Discussion

In the section "Rationale for Modeling" we said that a successful model should re-

produce both the detailed pattern of errors observed in short-term recall, which the Baddeley (1986) model of the "articulatory loop" cannot explain, and the more global effects of word length and similarity and articulatory suppression, which it can explain. To a first approximation then, our model is successful in that it shows word length, similarity, and suppression effects as well as the correct types of error. In light of this it might be fruitful to apply the model to some of the other phenomena for which the concept of the articulatory loop has proved valuable, e.g., developmental increases in memory span in children and cross-linguistic differences in digit span. It may also be informative to try to simulate neuropsychological disorders of short-term memory that have been attributed to impairment of the articulatory loop (e.g., Vallar & Baddeley, 1984). Progress in any of these areas can be regarded as testing the generality and applicability of the model.

However, the major motivation for our present work is that it should go some way toward providing an *explanatory* account of the core phenomena associated with the articulatory loop. The simple concept of the loop (Baddeley, 1986) is a useful model of many of the behavioral effects observed in short-term memory experiments but does not specify exactly how it could operate. In this paper we have provided a relatively simple candidate mechanism for the articulatory loop at a much more detailed level than the current conceptual account.

We also noted that a successful model might be expected to make some novel predictions and to generate insights into other activities involving the temporary storage of phonemes. The model does show behavior beyond the basics that we set out to reproduce. However, unsurprisingly in a first attempt, the further behavior of the model that we can explore has tended to correspond to more complex experiments that have already been done, rather than to novel predictions. In some cases the model shows the right behavior and in others it does not. Where the model does not match human data we suggest the modifications we would make for a second attempt. Clearly we will not really be at the stage of making novel predictions to test until we have a model that reflects the considerable amount that is already known about human behavior in short-term memory tasks.

Finally, we consider whether we have generated any insight into activities involving the temporary storage of phonological information. First, the articulatory loop should not be thought of as a chain of item–item associations, because the types of errors observed (particularly order errors) would not be natural to such a system. If we use the loop metaphor (or any chaining mechanism) it should perhaps be in terms of a loop of labels or contexts with associated items which are selected for recall by a mechanism, such as competitive queuing, in which item selection is prone to error.

Second, the limited capacity of short-term memory should not be thought of in terms of a box with a limited number of slots, or a loop of tape of finite length. It would seem to be more useful to think of it as a noisy system in which the opportunity for confusion increases with the number of items until the system cannot reliably select the correct one.

Third, if short-term associations (e.g., context–item associations) are stored in connections of decaying strength then the cues that excite temporally well separated items for recall via such connections must have no significant correlation.

Fourth, it seems to be useful to distinguish between input and output phoneme layers in the "phonological store." However, the Baddeley (1968) experiments indicate that triggering of the next item only requires that the recall of the previous item is finished—it should not depend on the phonemic composition of the recalled item.

The final point we note is the natural way in which the idea of competitive queuing solves the problem of choosing items in order. The way in which item nodes are active

before they are recalled, but are suppressed after recall is naturally prone to order errors and also accounts for the relative lack of repetition errors. We believe that experiments involving repeated items (in which omission of the repeated item tends to be a common error) will also be relatively easy to model using competitive queuing. The activation of items before recall is mentioned by Houghton (1990) as natural in the context of modeling speech production. It is surely no coincidence that the same mechanism seems natural in the articulatory loop, given that it is based on the idea of using "inner speech" or "subvocal rehearsal" to prolong the life of items in a short-term phonological store.

References

Aaronson, D. (1968). Temporal course of perception in an immediate recall task. *Journal of Experimental Psychology*, **76**, 129–140.

Amit, D. J., Sagi, D., & Usher, M. (1990). Architecture of attractor neural networks performing cognitive fast scanning. *Network*, **14**, 189–216.

Atkinson, R. C., & Shiffrin, R. M. (1971). The control of short-term memory. *Scientific American*, **225**, 82–90.

Baddeley, A. D. (1966). Short-term memory for word sequences as a function of acoustic, semantic and formal similarity. *Quarterly Journal of Experimental Psychology*, **18**, 362–365.

Baddeley, A. D. (1968). How does acoustic similarity influence short-term memory. *Quarterly Journal of Experimental Psychology*, **20**, 249–264.

Baddeley, A. D. (1986). *Working memory*. Oxford: Clarendon Press.

Baddeley, A. D., & Hitch, G. J. (1974). Working memory. In G. H. Bower (Ed.), *Recent advances in the psychology of learning and motivation* (Vol. VIII, pp. 47–90). New York: Academic Press.

Baddeley, A. D., Lewis, V. J., & Vallar, G. (1984). Exploring the articulatory loop. *Quarterly Journal of Experimental Psychology*, **36**, 233–252.

Baddeley, A. D., Thomson, N., & Buchanan, M. (1975). Word length and the structure of short-term memory. *Journal of Verbal Learning and Verbal Behavior*, **14**, 575–589.

Bjork, E. L., & Healy, A. F. (1974). Short-term order and item retention. *Journal of Verbal Learning and Verbal Behavior*, **13**, 80–97.

Broadbent, D. E. (1984). The Maltese cross: A new simplistic model for memory. *Behavioral and Brain Sciences*, **7**, 55–94.

Buhmann, J., & Schulten, K. (1987). Noise-driven temporal association in neural networks. *Europhysics Letters*, **4**, 1205–1209.

Burgess, N. (1990). *Neural networks, human memory and optimisation*. Ph.D. thesis, Faculty of Science, Manchester University.

Burgess, N., Moore, M. A., & Shapiro, J. L. (1989). Human-like forgetting in neural network models of memory. In W. K. Theumann & R. Koeberle (Eds), *Neural networks and spin glasses, proceedings, Porto Alegre, 1989*. Singapore: World Scientific, 1990.

Burgess, N., Shapiro, J. L., & Moore, M. A. (1991). Neural network models of list learning. *Network*, **2**, 399–422.

Conrad, R. (1960). Serial order intrusions in immediate memory. *British Journal of Psychology*, **51**, 45–48.

Conrad, R. (1964). Acoustic confusions in immediate memory. *British Journal of Psychology*, **55**, 75–84.

Conrad, R., & Hull, A. J. (1964). Information, acoustic confusion and memory span. *British Journal of Psychology*, **55**, 429–432.

Crowder, R. G. (1972). Visual and auditory memory. In J. F. Kavanagh & I. G. Mattingly (Eds.), *Language by ear and by eye* (pp. 251–276). Cambridge, MA: MIT Press.

Ellis, N. C., & Hennelly, R. A. (1980). A bilingual word length effect: Implications for intelligence testing and the ease of mental calculation in Welsh and English. *British Journal of Psychology*, **71**, 43–51.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, **14**, 179–211.

Estes, W. K. (1973). Phonemic coding and rehearsal in short-term memory for letter strings. *Journal of Verbal Learning and Verbal Behavior*, **12**, 360–372.

Estes, W. K. (1985). Memory for temporal information. In J. A. Michon & J. L. Jackson (Eds.), *Time, mind and behaviour*. Heidelberg: Springer-Verlag.

Gardiner, J. (1983). On recency and echoic memory. *Philosophical Transactions of the Royal Society London, Series B*, **302**, 267–282.

Grossberg, S. (1987). Competitive learning: From interactive activation to adaptive resonance. *Cognitive Science*, **11**, 23–63.

Guildford, J. P., & Dallenbach, K. M. (1925). The determination of memory span by the method of constant stimuli. *American Journal of Psychology*, **36**, 621–628.

Healy, A. F. (1974). Separating item from order information in short-term memory. *Journal of Verbal Learning and Verbal Behavior*, **13**, 644–655.

HEBB, D. O. (1961). Distinctive features of learning in the higher animal. In J. F. Delafresnaye (Ed.), *Brain mechanisms and learning*. Oxford: Blackwell.

HEFFERNAN, T. (1991). Ph.D. thesis, Manchester University.

HINTZMAN, D. L. (1967). Articulatory coding in short-term memory. *Journal of Verbal Learning and Verbal Behavior*, **6**, 312–316.

HITCH, G. J., HALLIDAY, M. S., & LITTLER, J. E. (1989). Item identification time and rehearsal rate as predictors of memory span in children. *Quarterly Journal of Experimental Psychology, Series A*, **41**, 321–338.

HOPFIELD, J. J. (1982). Networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Science USA*, **79**, 2554–2558.

HOUGHTON, G. (1990). The problem of serial order: A neural network model of sequence learning and recall. In R. Dale, C. Mellish, & M. Zock (Eds.), *Current research in natural language generation* (pp. 287–319). London: Academic Press.

HULME, C., THOMSON, N., MUIR, C., & LAWRENCE, A. L. (1984). Speech rate and the development of short-term memory. *Journal of Experimental Child Psychology*, **38**, 241–253.

JAHNKE, J. C. (1969). The Ranschburg effect. *Psychological Review*, **76**, 592–605.

JORDAN, M. I. (1986). *Serial order: A parallel distributed approach*. ICI Report 8604, Institute for Cognitive Science, University of California, San Diego, La Jolla, CA.

KLEINFELD, D. (1986). Sequential state generation by model neural networks. *Proceedings of the National Academy of Science USA*, **83**, 9469–9473.

KOHONEN, T. (1984). *Self-organisation and associative memory*. Berlin: Springer-Verlag.

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

LEE, C. L., & ESTES, W. K. (1981). Item and order information in short-term memory: Evidence for a multilevel perturbation process. *Journal of Experimental Psychology: Human Learning and Memory*, **7**, 149–169.

LEWANDOWSKY, S., & MURDOCK, B. B., JR. (1989). Memory for serial order. *Psychological Review*, **96**, 25–57.

MEZARD, M., NADAL, J. P., & TOULOUSE, G. (1986). Solvable models of working memories. *J. Physique*, **47**, 1457–1462.

MILLER, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, **63**, 81–97.

MURDOCK, B. B. (1960). The distinctiveness of stimuli. *Psychological Review*, **67**, 16–31.

MURDOCK, B. B. (1962). The serial position effect in free recall. *Journal of Experimental Psychology*, **64**, 482–488.

MURRAY, D. J. (1966). Vocalization-at-presentation and immediate recall with varying recall methods. *Quarterly Journal of Experimental Psychology*, **18**, 9–18.

MURRAY, D. J. (1968). Articulation and acoustic confusability in short-term memory. *Journal of Experimental Psychology*, **78**, 679–684.

NADAL, J. P. (1987). *Neural networks: A path from neurobiology to psychology?* Paper presented at the proceedings of "Chaos and Complexity," Turin, Italy.

NADAL, J. P., TOULOUSE, G., CHANGEUX, J. P., & DEHAENE, S. (1986). Networks of formal neurons and memory palimpsests. *Europhysics Letters*, **1**, 535.

NAVEH-BENJAMIN, M., & AYRES, T. J. (1986). Digit span, reading rate, and linguistic relativity. *Quarterly Journal of Experimental Psychology, Series A*, **38**, 739–752.

PARISI, G. (1986). A memory which forgets. *Journal of Physics A*, **19**, L617.

PETERSON, L. R., & JOHNSON, S. T. (1971). Some effects of minimizing articulation on short-term retention. *Journal of Verbal Learning and Verbal Behavior*, **10**, 346–354.

RUMELHART, D. E., HINTON, G. E., & WILLIAMS, R. J. (1986). Learning internal representations by error propagation. In D. E. Rumelhart, J. L. McClelland, and the PDP Research Group (Eds), *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Vol. 1. Foundations*. Cambridge, MA: MIT Press.

SALAMÉ, P., & BADDELEY, A. D. (1982). Disruption of short-term memory by unattended speech: Implications for the structure of working memory. *Journal of Verbal Learning and Verbal Behavior*, **21**, 150–164.

SCHWEICKERT, R., GUENTERT, L., & HERSBERGER, L. (1990). Phonological similarity, pronunciation rate, and memory span. *Psychological Science*, **1**, 74–77.

SHIFFRIN, R. M., & COOK, J. R. (1978). A model for short-term item and order retention. *Journal of Verbal Learning and Verbal Behavior*, **17**, 189–218.

TERMAN, L. M., & MERRILL, M. A. (1961). *Stanford-Binet intelligence scale—Manual for the third revision form L-M*. London: Harrap.

VALLAR, G., & BADDELEY, A. D. (1984). Fractionation of working memory: Neuropsychological evidence for a phonological short-term store. *Journal of Verbal Learning and Verbal Behavior*, **23**, 151–161.

VIRASORO, M. A. (1989). Categorization in neural networks and prosopagnosia. *Physics Reports*, **184**, 301–306.

WECHSLER, D. (1955). *The Wechsler adult intelligence scale*. New York: Psychological Corporation.

WICKELGREN, W. A. (1965a). Short-term memory for phonemically similar lists. *American Journal of Psychology*, **78**, 567–574.

WICKELGREN, W. A. (1965b). Short-term memory for repeated and non-repeated items. *Quarterly Journal of Experimental Psychology*, **17**, 14–25.

WICKELGREN, W. A. (1965c). Acoustic similarity and intrusion errors in short-term memory. *Journal of Experimental Psychology*, **70**, 102–108.

YOUNG, R. K. (1968). Serial learning. In T. R. Dixon & D. L. Horton (Eds.), *Verbal Behavior and General Behavior Theory* (pp. 122–148). Englewood Cliffs, NJ: Prentice-Hall.

ZHANG, G., & SIMON, H. A. (1985). STM capacity for Chinese words and idioms: Chunking and acoustic loop hypotheses. *Memory and Cognition*, **13**, 193–202.

(Received November 15, 1990)
(Revision received August 30, 1991)