Altmann, E. M. (2000). *Proceedings of the twenty second annual meeting of the Cognitive Science Society*. Mahwah, NJ: Erlbaum. pp 31-36.

Memory in Chains: Modeling Primacy and Recency Effects in Memory for Order

Erik M. Altmann (altmann@gmu.edu)
Human Factors & Applied Cognition
George Mason University
Fairfax, VA 22030

Abstract

Memory for order is fundamental in everyday cognition, supporting basic processes like causal inference. However, theories of order memory are narrower, if anything, than theories of memory generally. The memory-in-chains (MIC) model improves on existing theories by explaining a family of order memory effects, by explaining more processes, and by making strong predictions. This paper examines the MIC model's explanation of primacy and recency effects, and the prediction that primacy should dominate recency. This prediction is supported by existing data sets, suggesting that Estes's (1997) perturbation model, dominant among theories of order memory, is incorrect. Fits to data are presented and compared with fits of other models.

Introduction

When EgyptAir Flight 990 crashed off the coast of Massachusetts last year, the co-pilot had been recorded commending his life to God shortly before the plane went down. Did he do this because he had decided to crash the plane? Or did he do this because the plane was already crashing? The correct causal inference depends on knowing more than the key events — it also depends on knowing the *order* in which they occurred. If there were a living eyewitness, that person's memory for order would be immensely valuable, assuming it were correct. A theory that would help to predict the accuracy of order memory would thus be important in many applied domains.

Despite the importance of order memory, current theories are, if anything, narrower than is typical of memory theories generally. For one thing, they are only descriptive, in that they reproduce empirical phenomena once the analyst has encoded the appropriate underlying memory representation. For example, a widely cited model of order memory is the perturbation model (Estes, 1997). This model takes as input an array of items indexed by the dimension along which order confusion can occur (in the example above, time). Every so often, two cells in this array have some chance of swapping with one another. Over time, elements drift away from their original position, producing an "uncertainty gradient". However, the assumption that memory is organized as an array suggests that memory is an immense multi-dimensional array, with a dimension for each different kind of confusion. A representation this complex would place a heavy burden on the encoding process that creates it, and yet the perturbation model fails to address encoding at all. Two other models of order memory, the primacy model (Henson, Norris, Page, & Baddeley, 1996; Page & Norris, 1998) and the partial matching model (Anderson & Matessa, 1997), fail to address the encoding question, as well.

This paper presents a model of order memory that not only explains the underlying encoding processes, but also fits existing data better than the other models cited above.[1] The memory-in-chains (MIC) model accounts for a family of effects, but the focus here is on the theoretical prediction that primacy should dominate recency in memory for order.

Encoding Memory for Order

The model presented here is built on the ACT-R/PM cognitive theory, which combines perceptual-motor constraints (Byrne, 1998) with an analysis of memory as adapted to the structure of the environment (Anderson, 1990). The three theoretical mechanisms underlying the MIC model are a dual-code representation of attended objects, associative learning, and noisy communication between cognition and attention.

Dual-Code Representation

The main assumption shaping the representation of items in the MIC model is that cognition and attention are different processes that must communicate.[2] This assumption is fleshed out by what we know about the functional roles of the two processes. For example, we know that cognition can program attention in a top-down manner, and we know that attention communicates relatively low-level information to cognition for complex processing.

This analytical framework allows us to specify generic processes involved in processing sequential stimuli. For a given stimulus, cognition must first tell attention to attend to the stimulus. Then, attention must send the attended object back to cognition for further task-related processing. Thus, processing one stimulus requires two acts of communication — one to direct attention and one to receive the contents of the attended location.

In terms of representation, this communication model implies that processing a single stimulus involves two codes. One code, representing the item's location or position, is passed from cognition to attention. Another code representing the item's semantic or post-categorical identity, is passed back from attention. This need for two codes per item converges with broad support in the literature for dual-

[1] Executable and documented code for the model is available at http://hfac.gmu.edu/people/altmann/nairne-rpm.txt

[2] I use "attention" here to mean attention to external stimuli, and will use "the focus of mental attention" to refer to ACT-R's internal goal focus. The latter maps roughly to the task-related contents of the central executive (reviewed in Baddeley, 1992).

code representations (e.g., Logan, 1996; Paivio, 1971; Whiteman, Nairne, & Serra, 1994).

The communication model is illustrated in Figure 1A, which shows codes for three hypothetical items (X, Y, and Z). Time moves from left to right, and arrows mark the sequence in which codes appear in the focus of mental attention within cognition. (This interpretation of the arrows is elaborated below.) To process stimulus X, cognition sends a location code (X_L) to attention, from which it receives an object code (X_O). This is followed by whatever further task-related processing (not shown in the figure) might be required of the stimulus. The cycle then repeats for the next stimulus, Y.

An additional constraint on the model is that the channel through which cognition and attention communicate is the memory system itself. That is, when cognition sends a message to attention, it places a location code in memory for attention to retrieve. Similarly, attention sends a message back by placing an object code in memory for cognition to retrieve. This implementation of the communication channel is specified by the underlying theory, ACT-R/PM, but the tight functional integration of communication and memory can be traced to the earliest information-processing models of the cognitive system (e.g., Broadbent, 1958). The general implication is that functional descriptions of memory can also serve as functional descriptions of communication within the cognitive system as a whole. Two specific implications for the MIC model, concerning associative learning and noisy communication, are addressed in the next two subsections.

Associative Learning

Evidence suggests that associative links between temporally proximal codes are acquired incidentally by the cognitive system (e.g., Altmann & John, 1999; Crowder, 1976; Hasher & Zacks, 1979; Mandler & Mandler, 1964; Nairne, 1983). Like other unified cognitive theories, ACT-R contains an *associative learning* mechanism to explain and predict the corresponding behavioral phenomena (Anderson & Lebiere, 1998). Associative learning in ACT-R creates a link between two codes if one (the target) is retrieved from memory while the other (the cue) is already in the focus of mental attention within the cognitive system. As in Soar (Newell, 1990), this association is a new, permanent element of long-term memory. In the future, if the cue again enters the focus of mental attention it will prime (spread activation to) the target, increasing the chance that the target will be the next item retrieved to the focus of attention. Associative links therefore allow chained retrieval, in which each retrieved item cues retrieval of the next item.

Applied to the memory-based communication protocol described above, associative learning produces a linked structure in which location codes are interleaved with object codes. Figure 1A illustrates such a structure after the model has studied and encoded the three hypothetical items (X, Y, and Z) introduced earlier. An important assumption in the model, based on standard associationist principles, is that each code remains in the focus of attention long enough to still be there when the next code is retrieved. The consequence is that the first code becomes the cue for the second code, and the associative-learning mechanism links the two codes permanently in memory. In Figure 1 (and later figures), links created by associative learning are represented by arrows.

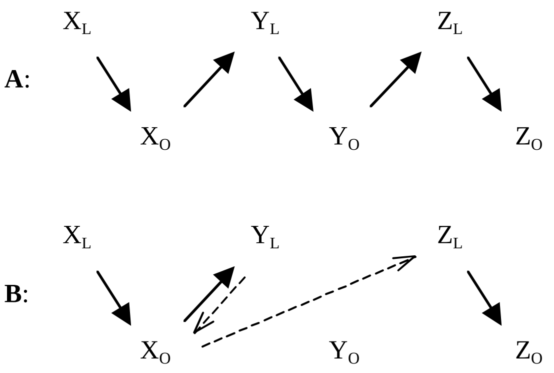

Figure 1: Memory representations encoded by the MIC model after studying items X, Y, and Z. An item has a location code (subscript L) and an object code (subscript 0). Panel A: Error-free representation. Panel B: Representation with two branches (incorrect links), in dashed ink, caused by noisy processing at study time.

Noisy Communication

If communication between cognition and attention were free of noise, then, subject to associative learning, it would produce a memory structure that allowed perfect sequential retrieval of items (Figure 1A). However, a memory system without noise would be unrealistic, and indeed sub-optimal (Anderson & Lebiere, 1998). In ACT-R as in other theories, items in memory have activation levels that determine their availability — items high in activation are less vulnerable to interference from other items. Noise in the memory system is expressed as transient fluctuations in individual activation levels, introducing the possibility of memory-retrieval error.

In the MIC model, noise can critically affect communication between attention and cognition at study time and produce incorrect links between codes. For each item processed, two memory retrievals are involved, one of a location code and one of an object code. Both retrievals are subject to activation noise. Specifically, when attention attempts to retrieve the location code most recently placed in memory, it may retrieve an old location code instead. Similarly, when cognition attempts to retrieve the object code most recently placed in memory, it may retrieve an old object code instead.[3] In terms of an everyday example, suppose that a newcomer is being introduced to a number of people, one at a time but perhaps too rapidly. While looking at the current person, the newcomer might “fall behind” and retrieve a previous, incorrect name. The result of such an

[3] I assume that errors occur within a code type only, and that a retrieval attempt always produces an item. These assumptions imply, for example, that an attempt to retrieve a location code will always produce a location code, though it may produce the wrong location code.

error would be that the newcomer might associate the wrong name with the wrong face. This kind of associative error is what the MIC model can encode at study time when there is noisy communication between attention and cognition.

Associative learning implies that a retrieval error during encoding produces an incorrect link in memory. I will refer to an incorrect link as a *branch*, because it branches off the correct temporal path through the codes of the list. The creation of a branch is illustrated in Figure 1B. There, a retrieval error occurs as Y_L is in the focus of mental attention and cognition tries to retrieve Y_O. This code was just placed in memory by attention, but due to noise in activation levels, X_O is transiently more active and hence is retrieved instead. This incorrect retrieval causes an association to be encoded between Y_L and X_O. This branch, shown as a dashed arrow, means that X could be mistakenly placed in Y's position at test time, producing an order error. This possibility is explored below in a discussion of the model's order-reconstruction process.

A second branch is also created in the scenario in Figure 3. When the model is presented with Z, it correctly retrieves Z_L, but X_O is still in the focus of mental attention (because of the retrieval error that occurred while processing Y). Therefore, associative learning creates a branch from X_O to Z_L, bypassing Y_O. This branch, however, need not produce an order error at test time, a possibility I also explore below.

A critical constraint on the communication model is the *near-miss* constraint, which is that incorrect codes temporally proximal to the correct code are more likely to intrude (and cause a branch). This constraint follows directly from the dynamics of activation in ACT-R. A code's activation depends on the lag since it was last retrieved — the longer the lag, the lower the activation. Therefore, a presented item will be more active than its predecessor (more precisely, the item's codes will be more active than its predecessor's codes), because the lag since presentation is smaller. The implication is that most branches created at study will be like those in Figure 1B – near misses, rather than far misses. This explains the uncertainty gradient, as I describe next.

Reconstructing Memory for Order

In order-memory experiments, items themselves are usually shown at test as well as at study — participants are asked simply to reconstruct their original order. Because items and positions are available at test, an assumption I represent in the model is that people randomly choose an initial item or position to start the reconstruction process. This assumption means that the model can take many paths through the representation in Figure 1B. In particular, one of these paths produces a positional swap of the kind that underlies uncertainty gradients (Nairne, 1992), and a second path produces a correct reconstruction.

The model will make an order error if the first cue it uses is location code Y_L. This code was linked incorrectly to X_O at encoding time, because of a retrieval error then. The result now is that the model will infer that X_O was the object that originally appeared in location Y_L, producing an order error. Next, the model might use X_O as a cue for which location to focus on next, in which case it would focus on location Z_L. Using Z_L as a cue, the model would most likely retrieve Z_O, which is correct. Thus, of two items placed, one was placed incorrectly and one correctly. The environment now indicates one remaining position and one remaining item. (Participants are typically instructed in the one-to-one nature of the reconstruction task, namely that every item maps to one position, with no items or positions left over.) The model will therefore infer that object Y_O occurred at location X_L. That is, the model will have swapped the order of the neighboring items X and Y. This is precisely the swap assumed (but not explained) by the perturbation model (Estes, 1997; Nairne, 1992).

Despite the encoding error, the structure in Figure 1B can also produce a correct reconstruction. If the model begins with location code X_L, for example, then it will most likely retrieve X_O, which is correct. Used as a cue, X_O will then prime two location codes, Y_L and Z_L. Suppose, first, that Z_L is retrieved. Used as a cue, Z_L will likely retrieve Z_O, which is correct. At this point, because only one item and one position remain, the model can place Y_O at Y_L, and the reconstruction will be correct. Suppose, instead, that when X_O is the cue, Y_L is retrieved. Used as a cue, Y_L will likely retrieve X_O, but this is now a dead end — X_O has been placed already. The model might now decide to place X_O elsewhere, but it might also decide simply to abandon Y_L as a cue and use Z_L instead. This would also produce a correct reconstruction.

How Primacy and Recency Arise

A standard empirical finding is that items at either end of a list are remembered more accurately than items in the middle. To explain these primacy and recency effects in order memory, we first need to revisit how the model generates order errors from an incorrect representation like the one in Figure 1B. Suppose, again, that the model initially focuses on Y_L at test time (essentially asking itself, "What item was in the second location?"). This cue will prime retrieval of X_O, causing the model to place X second instead of first. In contrast, given the correct representation of Figure 1A, Y_L would correctly prime Y_O. Thus, the frequency of branches, in aggregate data, is an important factor in determining the frequency of order errors. This relationship between branches and order errors means that we can examine branching patterns in the representation created at study time to predict error patterns at test time.

Primacy and recency effects arise in the MIC model because branch frequency is higher for middle items than for end items. Support for this claim comes from analyzing the interaction of branch frequency, branch *length*, and the distribution of branch lengths across a list. The notion of branch length is illustrated in Figure 2. Panel A shows two branches out of Y_L. Each branch is of length 1, meaning that the code at the head of the branch is temporally off by one from the correct code. Panel B shows two branches out of Z_L. One branch is of length 1, but the other is of length 2 because the code at the head of the branch is off by two from the correct code.

Two important points are illustrated in Figure 2. First, branch frequency varies inversely with branch length. That is, in aggregate data, branches to nearby codes are more

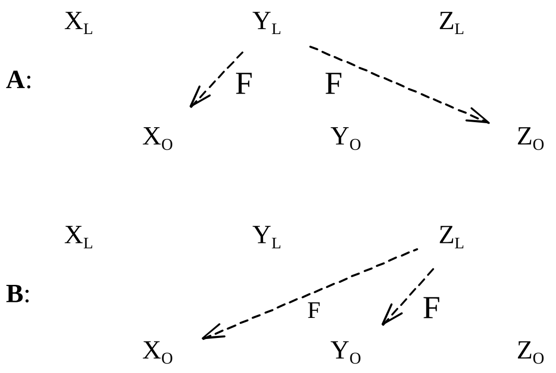

Figure 2: Middle codes have greater branch frequency than end codes. Panel A: A middle code with two short branches. Panel B: An end code with one short branch (bigger F) and one long branch (smaller F).

frequent than branches to far-away codes. This relationship follows directly from the near-miss constraint at encoding time: Temporally near codes are more likely than temporally remote codes to intrude on communications between cognition and attention and thereby cause branches. In Figure 2, branch frequency is indicated by the size of the "F" label. The branch of length 2 has a smaller F, meaning that it occurs less frequently in aggregate data.

The second point is that branch lengths are distributed unevenly across a list: Middle items have more short branches than end items. This distribution is also illustrated in Figure 2. Panel A shows all possible branches out of a middle code, where by "all possible" I mean that there is one branch to each possible incorrect code in the list. Similarly, Panel B shows all possible branches out of an end code. The middle code in Panel A has two short branches, whereas the end code in Panel B has only one. Because short branches are more frequent in aggregate data, the middle code will produce more order errors at test time.

In sum, primacy and recency effects in the MIC model reflect error patterns during encoding, in that middle items suffer branches more frequently than end items. At test, these extra branches produce more order errors.

Prediction: Primacy Dominates Recency

Models of order memory make conflicting predictions about the relationship between primacy and recency. The perturbation model, for example, predicts that primacy and recency should be symmetrical. In contrast, the primacy model was constructed to account for the common result that primacy is greater than recency (Henson et al., 1996).

The MIC model predicts that primacy should be greater than recency, an effect I refer to as *primacy dominance*. This prediction is a logical consequence of interactions between the task and constraints on the cognitive system (as specified by ACT-R/PM). In contrast, the primacy model (Henson et al., 1996; Page & Norris, 1998) accounts for primacy dominance with ad hoc mechanisms that are not constrained by task structure or independent theory.

Primacy dominance in the MIC model is a consequence of three interacting constraints. The first constraint is sequential processing at study — participants see one item at a time. The second constraint is related to branch *direction*. Every branch has a direction in that it points either forward or backward in time. A forward branch points to a code newer than the correct one (in Figure 3, from X_L to Y_O). A backward branch points to a code older than the correct one (in Figure 3, from X_O to X_L instead of to Y_L). As I elaborate below, branch direction interacts with sequential processing to make forward branches less likely to be taken at test time as the model is reconstructing order. The third constraint is the distribution of branch directions across a list. The early (not-recent) end of the list systematically involves more forward branches than the late (recent) end. Because forward branches are less likely to be taken at test time, early items suffer fewer order errors.

To see why forward branches are less likely to be taken at test time than backward branches, we need to consider the contingent nature in which forward branches are encoded at study. The encoding of forward and backward branches is illustrated in Figure 3. In that scenario, the model correctly processes X_L and transitions to X_O (creating link 1). A retrieval error then occurs — with X_O still in the focus of attention, the model retrieves X_L instead of Y_L. This creates a backward branch from X_O to X_L (link 2). The next step (assuming no further retrieval errors) creates a forward branch from X_L to Y_O (link 3). Thus, one retrieval error has produced two branches, one backward and one forward.

Two important points are illustrated in Figure 3. First, link 3 (the forward branch) is *contingent* on link 1 (the correct link). That is, a forward branch can only occur if a correct link out of the same code already exists. This contingency simply reflects sequential processing – X is already linked into the chain when Y is processed. The effect of this contingency is that at test time, if the model uses X_L as a cue, link 3 and link 1 prime competing targets. Thus the potential for taking a forward branch (link 3) is mitigated by the existence of the correct alternative (link 1). (By "taking a branch" I mean that the code at the tail end successfully primes the code at the head end, causing the latter code to be retrieved next.) The second important point in Figure 3 is that no such contingency accompanies a backward branch. Link 2 is the only link leading from X_O. At test time, if the model uses X_O as a cue, the backward branch will prime only X_L, with no correct alternative. Thus,

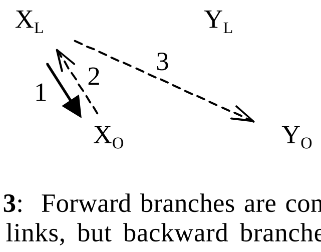

Figure 3: Forward branches are contingent on correct links, but backward branches are not. (1) Cognition retrieves X_O, creating a correct link. (2) Cognition retrieves X_L instead of Y_L, creating a backward branch. (3) Cognition retrieves Y_O, creating a forward branch.

backward branches are more likely than forward branches to be taken at test time, in the sense that they prime only incorrect target codes. Put another way, backward branches have a higher *effective* branch frequency than forward branches. If a given forward branch and a given backward branch have the same frequency over multiple trials, the backward branch will be taken more often, making it effectively more frequent.

The third constraint leading to primacy dominance is that forward and backward branches are distributed unevenly across a list. Both kinds of branch occur with equal frequency overall, because a single retrieval error at study produces one branch in each direction. However, earlier items have more forward branches than later items. In the extreme cases, the first item can have only forward branches, and the last item can have only backward branches. Thus, earlier items have a lower effective branch frequency. That is, branches from earlier items, though as frequent as branches from late items, are effectively less frequent because they are less likely to be taken during order reconstruction.[4]

In sum, primacy dominates recency as a natural consequence of task structure interacting with cognitive constraints. Sequential processing makes forward branches contingent on correct links, and because forward branches are more frequent for early items, these items suffer fewer order errors. In graphical terms, the serial position curve in order memory is rotated slightly clockwise.

Comparing Model to Data

To test whether the model reproduces the serial position effects predicted by the analysis above, I simulated data from Nairne (1992). In that study, memory for order was tested implicitly. Participants were asked to give pleasantness ratings of words, with words presented in lists of five for three seconds a word. In a between-subjects manipulation, participants were given a surprise order-reconstruction test after 30 seconds of distraction, after 4 hours, or after 24 hours.

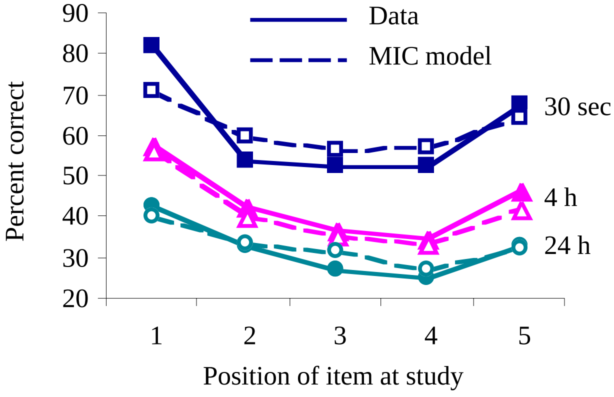

Figure 4: Accuracy data for order memory (Nairne, 1992) and fits of the MIC model.

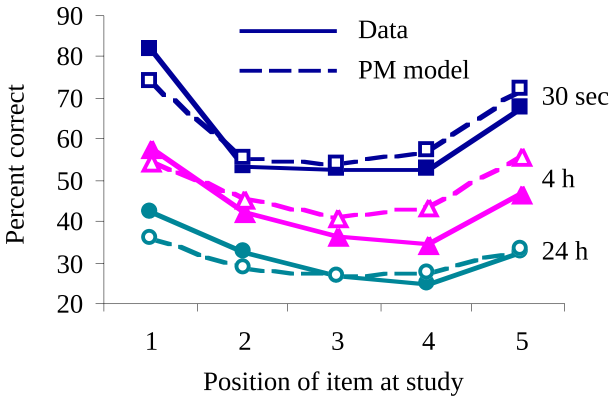

Figure 5: Accuracy data for order memory (Nairne, 1992) and fits of the partial-matching (PM) model.

Data from Nairne (1992) are shown in Figure 4, fit to data from the MIC model. In all three conditions, primacy appears to dominate recency, and the model captures this pattern, accounting for 93% of the variance over 15 data points (RMSE = 4.2%). The close fit of the MIC model to complex data is strong support for its assumptions.

Moreover, the fit of the MIC model improves slightly on that of the perturbation and partial-matching models of the same data. The partial matching model, which fits better than the perturbation model (Anderson & Matessa, 1997), accounts for 90% of the variance over the same 15 data points (RMSE = 5.0%).[5] These fits are close, but Figure 5 shows that in all three conditions the model under-predicts primacy and over-predicts recency. This mis-alignment is systematic, according to the MIC model, because the partial matching model (like the perturbation model) mistakenly predicts that primacy and recency should be the same.

Many important details about the MIC model are omitted here. For example, only 15 data points, or those for correct responses, are shown in Figure 4; the total number of points fit by the model is 75. In addition, I have not described the time parameter that causes the model's serial position curve to shift downwards with longer retention intervals. These issues will be addressed in a subsequent report.

Discussion

The MIC model explains a family of phenomena in memory for order. This paper has described how the model explains primacy and recency effects — why they occur, and how they are related. Primacy and recency effects occur because middle items suffer more branches (incorrect links) than end items and thus are more vulnerable to order errors. In addition, primacy should dominate recency because early items suffer fewer backward branches than early items. Backward branches cause more order errors than forward branches, offsetting the benefits of recency and rotating the

[4] Specifically, the first item has a lower effective branch frequency than the last item, the second item has a lower one than the second-last, and so on.

[5] The 15 data points given here are a subset of the 75 data points found in Nairne (1997). Fits of the perturbation and partial matching models to the complete data set are given in Anderson and Matessa (1997). The fit of the partial matching model to the 15 data points used here was determined by running the model available on the Web at http://act.psy.cmu.edu.

bowed serial-position curve slightly clockwise. In addition to these serial position effects, the MIC model also explains positional uncertainty (Altmann, 2000), and thus is a step toward an integrated and executable theory of memory for serial order.

The MIC model is important for several reasons. First, it extends an existing cognitive theory to incorporate an additional set of effects. The model inherits a representation, a learning mechanism, and a communication channel from ACT-R/PM. The model's explanations follow directly from the integration of these mechanisms, illustrating (again) the explanatory power of unified theories (Newell, 1973; 1990).

Second, the MIC model goes beyond existing models of order memory to explain study-time processes as well as test-time processes. Of existing models, the perturbation model is the best known, and has been advanced as a generalized model of memory loss and distortion (Estes, 1997). However, the perturbation model has nothing to say about how memory for order is encoded at study time, begging the question of how the information-rich, array-like memory representation input to the perturbation model comes about in the first place.

Third, the MIC model is behaviorally distinguishable from the perturbation and partial-matching models. Both models predict that primacy and recency should be symmetrical, but several data sets suggest otherwise. The primacy model (Henson et al., 1996; Page & Norris, 1998) accommodates this primacy dominance, but like the others fails to explain how order information is encoded in the first place. The MIC model, in which primacy dominance is a logical consequence of the underlying memory theory, may also be the most accurate and complete explanation, as well.

Rigorously testing the prediction of primacy dominance will be the next important step in this research. Because this prediction flows from architecture-level premises (about representation, learning, and cognitive noise), primacy dominance should be found pervasively in empirical studies. A second important step will be to extend the model to account for the "sawtooth" pattern arising when confusable and non-confusable items are interleaved (Henson et al., 1996). Finally, order memory is a strong constraint on memory theory generally. As we build toward unified theories of cognition, it will be important to integrate order memory with related models (e.g., Anderson & Matessa, 1997; Burgess & Hitch, 1999) and with the rich theoretical history of serial learning (see, for example, Crowder, 1976).

Acknowledgments

This work was supported by grant F49620-97-1-0353 from the US Air Force Office of Scientific Research. Thanks to Melanie Diez, Wai-Tat Fu, Wayne Gray, Margaret Peterson, Lelyn Saner, Wolfgang Schoppek, Chris Schunn, Greg Trafton, Richard Young, and an anonymous reviewer for their insights and comment.

References

Altmann, E. M. (2000). Memory in chains: A dual-code associative model of positional uncertainty. In N. Taatgen & J. Aasman (Eds.), *Proceedings of the 3rd international conference on cognitive modeling*. Groningen, The Netherlands.

Altmann, E. M., & John, B. E. (1999). Episodic indexing: A model of memory for attention events. *Cognitive Science, 23*(2), 117-146.

Anderson, J. R., & Lebiere, C. (Eds.). (1998). *The atomic components of thought*. Hillsdale, NJ: Erlbaum.

Anderson, J. R., & Matessa, M. (1997). A production system theory of serial memory. *Psychological Review, 104*(4), 728-748.

Baddeley, A. D. (1992). Is working memory working? The fifteenth Bartlett lecture. *Quarterly Journal of Experimental Psychology, 44A*, 1-31.

Broadbent, D. E. (1958). *Perception and commucation.* London: Pergamon Press.

Burgess, N., & Hitch, G. J. (1999). Memory for serial order: A network model of the phonological loop and its timing. *Psychological Review, 106*(3), 551-581.

Byrne, M. D. (1998). Perception and action. In J. R. Anderson & C. Lebiere (Eds.), *Atomic components of thought* (pp. 167-200). Hillsdale, NJ: Erlbaum.

Crowder, R. G. (1976). *Principles of learning and memory.* Hillsdale, NJ: Erlbaum.

Estes, W. K. (1997). Processes of memory loss, recovery, and distortion. *Psychological Review, 104*, 148-169.

Hasher, L., & Zacks, R. T. (1979). Automatic and effortful processes in memory. *Journal of Experimental Psychology: General, 108*(356-388).

Henson, R. N. A., Norris, D. G., Page, M. P. A., & Baddeley, A. D. (1996). Unchained memory: Error patterns rule out chaining models of immediate serial recall. *Quarterly Journal of Experimental Psychology, 49A*(1), 80-115.

Logan, G. D. (1996). The CODE theory of visual attention: An integration of space-based and object-based attention. *Psychological Review, 103*(4), 603-649.

Mandler, J. M. & Mandler, G. (1964). *Thinking: From association to Gestalt.* New York: John Wiley & Sons.

Nairne, J. S. (1983). Associative processing during rote rehearsal. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 9*(1), 3-20.

Nairne, J. S. (1992). The loss of positional certainty in long-term memory. *Psychological Science, 3*(3), 199-202.

Newell, A. (1973). You can't play 20 questions with nature and win: Projective comments on the papers of this symposium. In W. G. Chase (Ed.), *Visual information processing* (pp. 283-308). New York: Academic Press.

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

Page, M. P. A., & Norris, D. (1998). The primacy model: A new model of immediate serial recall. *Psychological Review, 105*(4), 761-781.

Paivio, A. (1971). *Imagery and verbal processes*. New York: Holt, Reinhart, and Winston.

Whiteman, H. L., Nairne, J. S., & Serra, M. (1994). Recognition and recall-like processes in the long-term reconstruction of order. *Memory, 2*(3), 275-294.