A Connectionist Simulation of Attention and Vector Comparison: The Need For Serial Processing in Parallel Hardware[1]

Judith M. Shedden & Walter Schneider
Learning Research and Development Center[2]
University of Pittsburgh, Pittsburgh, PA 15260

Abstract

Given the massively parallel nature of the brain an obvious question is why are so many information processing functions serial? In particular, this paper addresses the issue of the comparison process. Behavioral data show that in perceptual matching tasks (such as memory scanning and visual search) performance is systematically affected by stimulus load, in that required processing time increases with each additional comparison item. It is arguable whether this indicates a processing system that performs serial comparisons, or a system for which comparisons are done in parallel but reaction time is affected by load because of other system limitations. In this simulation we show that in a modular connectionist system vector transmission is possible in parallel, but the comparison process within a module must be done serially unless accuracy is sacrificed.

This paper examines the question of the serial or parallel nature of the comparison process, and describes the implementation of a connectionist model designed to test the efficiency of parallel multiple comparisons. Despite the fact that neocortex is massively parallel in its architecture there are many tasks for which behavioral data illustrate serial processing. It is important to determine what processing limitations induce serial processing even in the presence of parallel hardware. In some cases this can be explained by limitations in the number of responses that can be made at one time (Deutsch & Deutsch, 1963), by crosstalk or vector transmission interference (Schneider & Detweiler, 1987), or by competition for limited processing resources or operators (Kahneman, 1973). Connectionist based modeling has emphasized the parallel nature of processing. However, even in connectionist based vector processing systems there is a need to serialize operations. With respect to the comparison process the limitation may be due, not to limitations of the system to *make* multiple comparisons, but to the increased error that results from multiple comparisons in the same comparator.

Theories of Comparison Processes: Background

Psychologists have been studying the perceptual comparison process for years using a multitude of tasks, for the most part analyzing measures of accuracy and reaction time to respond to a target in a field of distractors. In a memory scanning and visual search task subjects compare one or more items in memory to one or more items on visual display, looking for a match between items (Sternberg, 1969; Schneider & Shiffrin, 1977; see Shiffrin, 1988 for review). In a same/different response task subjects compare items in two lists looking for a mismatch between items (Proctor, Healy & Van Zandt, 1991; Ratcliff & Hacker, 1981). In a conjunction search task subjects search for a target consisting of a conjunction of particular features in a field of distractor items which consist of the same features but not the conjunction (Treisman & Gelade, 1980).

A robust finding of memory or display scanning is that reaction time increases in a nearly linear fashion as a function of the number of comparisons that must be performed (e.g., Sternberg, 1969, Schneider & Shiffrin, 1977). This linear increase typically occurs when there is a varied mapping between stimuli and responses (e.g., the subject's responses to the same stimuli change from trial to trial, see Schneider & Shiffrin, 1977) or under conditions of high accuracy and low discriminability. These data have been interpreted by some as indicating a serial repetition of the perceptual comparison process (Sternberg, 1969; Schneider & Shiffrin, 1977; Treisman & Gelade, 1980). Another interpretation of the linear increasing reaction time function is that all items are compared in parallel, and the effect of load on reaction time is due to other limitations of the parallel processing system (Pashler & Badgio, 1987; Proctor, Healy, & Van Zandt, 1991; Ratcliff, 1988). If there is a well practiced consistent mapping between stimuli and responses then comparisons can be performed in parallel (see Schneider & Shiffrin, 1977) indicating the hardware can support parallel comparisons. A reaction time function that does not increase with load indicates preattentive, or automatic parallel processing in which the target "pops out," and is thought to be independent of the comparison process (Schneider, 1985; Treisman, 1985). The modeling of consistent search is detailed elsewhere (Gupta & Schneider, 1991). This paper focuses on the varied mapping search in which serial processing occurs.

There have been a variety of modeling techniques used to explain the linear increase in reaction time (see Townsend & Ashby, 1983; Luce, 1986). The models take the form of either assuming the comparisons are performed sequentially, or they are performed in parallel but at a reduced rate due to the need to share the resources that enable parallel processing. In general, examining the mean data can not distinguish between the serial or parallel resource limited processing models. However, if the data (e.g., Schneider & Shiffrin, 1977, experiment 2) show a self-terminating scan in which the slope of the positive mean function increases at half the rate of the negative function, and the slope of the variance function increases faster for the positive than the negative responses,

1: This research was supported in part by ONR contract N00014-87K-0397 and N00014-86K-0678 and ARI contract MDA903-89K-0174, and in part by a Natural Sciences and Engineering Research Council of Canada Postgraduate Scholarship to the first author.

2: Email: shedden@vms.cis.pitt.edu schneider@vms.cis.pitt.edu

Shedden, J. M., & Schneider, W. (1991). A connectionist simulation of attention and vector comparison: The need for serial processing in parallel hardware , In Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society, Chicago (pp. 546-551). Hillsdale, NJ: Lawrence Erlbaum. .

then there is strong support for a serial comparison process (see Townsend & Ashby, 1983; Luce, 1986). In this paper we examine why the processing must sometimes be serial even with parallel hardware.

Existing models generally do not provide an interpretation of why processing should be serial or resource limited. From the physiological perspective there is little justification for a requirement that visual processing be serial. The retina and early levels of the visual system certainly operate in parallel with different retinal locations activating topographically distinct sections of tissue in multiple visual maps (Desimone & Ungerleider, 1989). It may be that these parallel channels must converge to a single comparator which becomes inaccurate when receiving multiple inputs. The present model explicitly models such a comparator in a connectionist simulation, and maps out accuracy as a function of the number of concurrent inputs. The simulations described in this paper provide evidence for serial comparisons within a single comparator module. Parallel comparisons are still conceivable if more than one comparator is available. However, the behavioral data supports the view that human visual and memory processing is serial, and is likely to represent processing by a single comparator in varied mapping search tasks.

Modular Organization of Cortical Anatomy

The present model utilizes a common modular architecture and parallel processing incorporating salient features of cortical processing. Cells throughout cortex (post the initial sensory areas such as visual area V1) show similar patterns of layering, types of cells, and local connections. The structure of cortex is modular, with processing occurring in identical columns, or hypercolumns, which are highly connected within and sparsely connected between (Mountcastle, 1979). Studies of V2 cortex show a structured layering system of cells and connections (Lund, Hendrickson, Ogren, & Tobi, 1981). Information is transmitted through a column in a feed forward direction through two layers of pyramidal cells. An excitatory signal is input to Layer 4 pyramidal cells, which project to layer 2-3 pyramidal cells, which in turn project out of the column. In addition, there are recurrent connections within a column, in which excitatory pyramidal cells feedback to themselves. Inhibitory interneurons are primarily local connections within a column, and it is thought they perform gating and modulatory functions. A special class of axon-axon inhibitory cell is the chandelier cell, which connects to the axon initial segments of sets of pyramidal output cells (Peters, 1984). Chandelier cells have fast inhibitory effects, and possibly function as attentional gating devices (see Shedden & Schneider, 1990, Douglas & Martin, 1990).

A Modular Connectionist Model

The simulation under discussion was implemented in the CAP2 computer simulation environment. The general model incorporates modules, units, layers, and control elements which can be combined in various architectural configurations. A module consists of an input layer of units, a recurrently connected auto-associative matrix, a feed-forward associative matrix, and an output layer of units (see Figure 1). Modules can be added to the system in breadth, so that several input modules (Figure 1: 1a and 1b) connect to one higher level module (Figure 1: 2a). In addition, modules can be added to the system in depth, creating several hierarchical layers. Associated with each module are control elements (gain and feedback) which manipulate signal strength within the system, and report elements (activity and priority) which manipulate attentional effects.

Each module effectively has three layers making up a connectionist network, including the module **input layer** (traditionally called the hidden layer), the module **output layer,** and the **data input** (which may be the output from the previous level of modules). The current simulation consists of two modules connected hierarchically, so that the output from one module feeds forward through an associative matrix to become the data input to the other module.

The model incorporates the recurrent nature of cortical within-column connections as seen on Layer 4 pyramidal cells. The modules are implemented differently from the standard three-layer connectionist network, in that the input layer of each module is recurrently connected through an auto-associative matrix to itself. In this way information input to the module on each iteration is a function of the external input plus internal feedback from previous transmissions. In a hierarchically organized architecture the external input is received from an input module (or modules) on the processing layer below, and the strength of the external signal is controlled by a scalar gain control element associated with the input module. The strength of the internal feedback signal in a module is controlled by a scalar feedback control element associated with that module. Thus, the gain and feedback control elements function to modulate the output of a population of units in an analogous fashion to the hypothesized function of the inhibitory chandelier cell discussed earlier. In general, the net input to a module is:

net input = (feedback * internal input) + (gain * external input),

where feedback and gain are scalar values and the internal and external inputs are vectors.

Simulations. The vector space for one simulation consists of ten pairs of input and target vectors, each having a length of 50 units, with correlations of 0.15 or below within the members of each input and target vector set. This is done by generating random vectors and discarding those with correlations above 0.15. Activation levels for each vector unit range from -1.0 to 1.0, with a resting activation of 0.0. Input vectors and target vectors have unit activations set randomly to -1.0 or 1.0, and the target vector units are then clipped to -0.9 or 0.9 respectively.

During training an input vector is presented to the system, activation is allowed to spread through the network, and the error is calculated between the output vector and the target vector. Activation of the hidden layer units are allowed to range freely between -1.0 and 1.0; activation of the output layer units are subjected to the nonlinear logistic function

$$activity = -1 + 2/(1 + e^{-activity \cdot 0.9}).$$

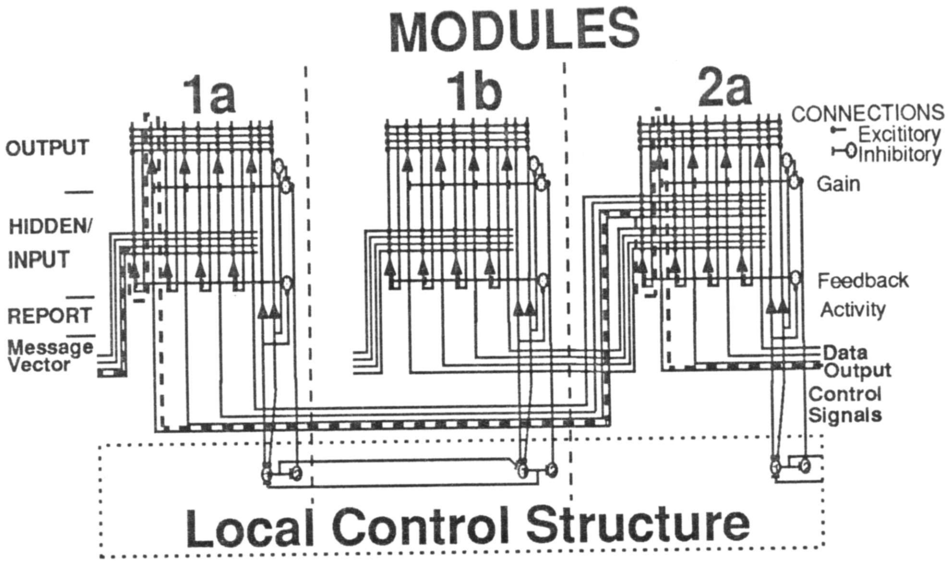

Figure 1: A connectionist model of cortical processing. This model consists of a two layer structure that parallels the cortical input layer 4 pyramidal cells, and the cortical output layer 2,3 pyramidal cells. Inhibitory modulation units control the strength of the internal feedback and output signals (feedback and gain). Activity and priority control information is carried by report units (layer 5,6 pyramidal cells) which transmit to central control structures as well as directly to the inhibitory modulation units. The thick dotted line illustrates the information flow from left to right through two modules. The CAP2 environment is described in more detail elsewhere (Schneider & Detweiler, 1987; Detweiler & Schneider, in press; Shedden & Schneider, 1990; Gupta & Schneider, 1991).

Connection weights are changed by the back propagation of the error after each presentation (Rumelhart, Hinton, & Williams, 1986). At the same time the recurrent connection weights of the auto-association matrix at the hidden layer are changed using the delta learning rule (see McClelland & Rumelhart, 1988). The network is trained for 20 epochs, each of which consist of one presentation of all input/target vector pairs in random order. At the completion of training the system has reached the criterion of 100% accuracy and correlations of 0.94 to 0.99 between the output and target vectors for each input pattern.

The Vector Comparison Task. The task set for the system is a simple matching task, in which vectors to be matched are presented along with 1, 2, 3, or 4 comparison vectors, thus requiring either 1, 2, 3, or 4 parallel comparisons. During testing every input vector is used in turn as the vector to be matched (sample vector), and the comparison vectors are chosen randomly without replacement from the remaining vectors in the set. On positive trials, one of the comparison vectors is identical to the sample vector. An equal number of positive and negative trials are presented. On each trial the comparison set of input vectors are added, and scaled as a function of the number of inputs (e.g., the addition of from 2 to 5 vectors are scaled from 0.5 to 0.2 of the single vector input). Activity is allowed to build in the system for 5 iterations of external input plus internal recurrent feedback. The data presented below are from 5 different sets of 10 input/target vector pairs, processed at 4 levels of feedback (discussed below), for a total of 20 simulations and 50 datum points per condition.

The measure of evaluation the network uses to determine a match is *vector activity*. The vector activity is the strength of the evoked vector following the summation of two or more input vectors. For example, in Figure 1, modules 1a and 1b output and activate vectors in module 2a. An input vector evokes a specific pattern of activity over the input and output layers of the module. When multiple vectors are presented their effect on the input layer is additive, thus when the vectors are similar the overall level of activity will be higher than when they are uncorrelated (see Schneider & Oliver, 1991).

One measure of vector activity is the average sum of the squared activity of each unit, which can be thought of in geometric terms as the length of the vector. When two vectors are added together, the length of the resultant vector is a measure of the similarity of the two vectors. The resulting activity is equal to:

$$activityreport=\sum_{i=1}^{n} x_i^2+\sum_{i=1}^{n} y_i^2+2|x||y|\cos\theta$$

where n is the number of vector units, theta is the angle between the vectors, and |x| and |y| are the Euclidian lengths of the vectors x and y.

In a matching task a system attempts to detect a match if the measure of activity (or vector length) is above a set criterion, and reject a match otherwise. This analysis examines the function of accuracy at this task with an increasing number of parallel comparisons. A measure of comparison accuracy is provided by the *d'* metric. The d' (from Signal Detection Theory) is a measure of the signal detection sensitivity of a system, and takes into account possible response biases (Tanner & Swets, 1954). There are two distributions of possible vector activity, one in which no match occurs (noise), and one in which a match does occur (signal plus noise). To achieve high accuracy the system must not only detect the signal, but must make a correct rejection of noise in which no signal occurs. Thus a match criterion must be chosen so that both the probability of missing a signal and the probability of making a false alarm to noise are low. This is only possible if sensitivity to the signal is high enough, that is, if there is enough distance between the two distributions of noise, and signal plus noise. The d' is a measure of the distance between the means of the two distributions in normal standard deviations, and is therefore a measure of sensitivity that is not affected by the possible positive or negative response biases for which humans are prone. From d', if one assumes a non-biased criterion it is possible to determine the probability for error, which is simply the area under the overlapping tails of the two distributions. A d' of 4, 3, 2, 1, or 0.5 normal standard deviations corresponds to an error probability of 0.02, 0.07, 0.16, 0.31, or 0.4 respectively, assuming the subject makes an equal number of misses and false alarms. In scanning experiments humans are generally expected to maintain accuracy above 95% and hence a d' of above 3 is expected.

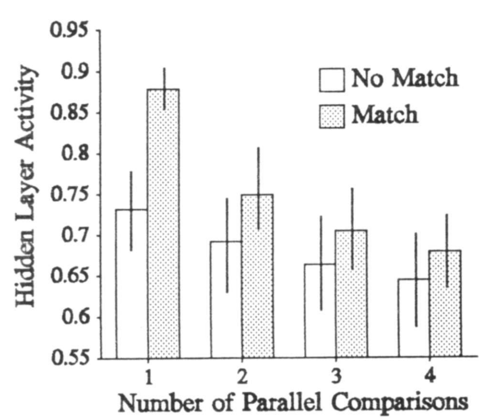

Figure 2: Activity on the hidden layer is presented as a function of the number of parallel comparisons. When only one comparison is necessary the difference between the Match and No Match distributions is large, but decreases dramatically if multiple comparisons must be made. The measure of activity is the average sum of the squared activity of each vector unit. Error bars indicate one standard deviation above and below the mean.

There is much more information contained in a vector than its length, and it would be possible to train a network specifically to distinguish between distributions of noise and signal plus noise. However, we are interested the human capacity to successfully perform comparisons on the first trial.

Simulation Results. Figure 2 shows the activity level of the hidden layer of the network as either 1, 2, 3, or 4 parallel comparisons are made (50 trials per condition). The measure of activity is the average sum of the squared activity of the vector units. When only one comparison is required there is clear separation between the Match and the No Match distributions (means of 0.88 and 0.73 respectively). With one comparison an activity criterion can be set which results in a d' of 3.48 and a 4% error rate. However when two or more comparisons are performed the difference between the distributions becomes much smaller and the region of possible error becomes much larger. For two comparisons the d' drops to 0.98 with an error rate of 31%, which would not be acceptable for most search tasks.

These severe decrements in accuracy with parallel comparisons are robust for different metrics of activity. Figure 3 shows d' values determined for distributions based on the average absolute value of vector activity as well as the average sum of squares, for comparisons based on the hidden and output layer.

An important issue relating to cortical architecture is whether comparisons can be performed on the hidden or output layer. There are three reasons to suggest that a module would monitor the hidden rather than the output layer. First, the output pyramidal cells often do not make synaptic connections within the layer. Second, if the output layer is gated to control the output to the next level, the comparison could not be performed until the vector is transmitted to the next level of modules. Third, when the output layer is transmitting, the transmission to the higher level of modules will interfere with any other potential signals. Performing the comparison within the module allows other modules to transmit to the higher level modules. This is analogous to the problem faced with data bus arbitration in computer architectures. Each device on the bus limits its transmission in order to allow other devices to transmit on the bus. Typically in computers, each device makes a priority assessment of its internal state without transmitting on the bus. Data is transmitted only after the device activates a bus request and is granted permission from the bus arbitration logic to transmit the data. For all of the above reasons it is important to determine if the match could be performed on the hidden versus the output layer.

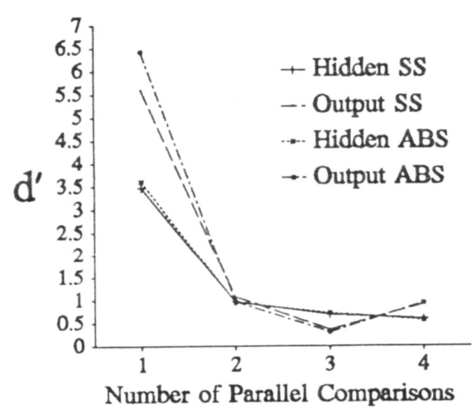

Figure 3: The d' for the hidden and output layers is shown as a function of the number of parallel comparisons. Two metrics of activity are graphed: The average sum of the squared activity of the vector units (SS), and the average sum of the absolute value of the vector units (ABS). In all cases, d' decreases dramatically if more than one comparison is made.

Figure 3 graphs the d' values for both the hidden layer and the output layer of the system. For both layers there is a robust deficit for multiple comparisons. The output layer does show better match sensitivity for single comparisons (for example, the d' of the average absolute value metric is 6.5 versus 3.62). It is unclear how relevant the increased detection sensitivity is because human sensitivity in search tasks is usually below 4. In future investigations we will determine how the hidden/output layer d' differences vary as a function of the nature of the squashing function (we used a logistic on the output layer and a step function on the hidden layer), vector size, and correlations among distractors and targets. In the current data, d' was higher on the output layer, but comparisons based on the hidden layer are in the range typical of human performance. Perhaps in cases where crosstalk can be managed and accuracy is extremely important, the more effortful comparison on the transmitted output is beneficial. In any case, as Figure 3 illustrates, both the hidden and the output layer show the harmful effects of multiple comparisons on the probability for error.

Levels of Feedback. The recurrent connections that are ubiquitous throughout cortex are represented by the auto-association on the hidden layer in the model. As described above, the feedback control element modulates the strength of the recurrent signal within a module. Four different levels of feedback were tested for each number of required parallel comparisons, and results for the different conditions are shown in Figure 4. When only one comparison is made, the d' is reasonably high for both the hidden and output layers at a feedback level of 0.1, but falls off at feedback levels above and below 0.1. The d' is below 1.6 for any case where multiple comparisons are made, and feedback has very little effect.

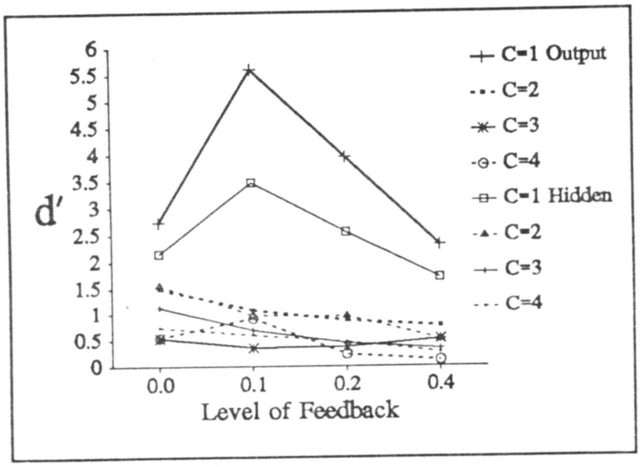

Figure 4: The d' for the hidden and output layers for each number of parallel comparisons is shown as a function of level of feedback. The feedback element controls the strength of the internal recurrent signal. For the optimum feedback level of 0.1 the d' measure is greatest, but it falls off with higher and lower levels of feedback.

There are several reasons to include feedback in the module making comparisons. First, input or hidden layer feedback is common in cortex. Second, feedback is critical for latching and holding signals, for the categorization of incoming information, and for signal buffering during concurrent transmissions (see Shedden & Schneider, 1990). Third, feedback may enhance d' because the associative feedback will strengthen previously learned vectors. Matching vectors have a close resemblance to previously learned vectors. In contrast, mismatching pairs represent a blending of features that have not been learned in the auto-associative matrix.

Figure 4 shows the effect of the strength of auto-associative feedback on detection sensitivity. Without feedback sensitivity levels are low (d' of 2.74 and 2.15 for output and hidden layers) and the error rates are higher (about 10%) than those typically observed in scanning experiments. Increasing feedback to 0.1 improves comparison sensitivity to d' measures of 5.60 and 3.48 respectively for the output and hidden layer for a single comparison. When feedback levels are too high distortion of signals begins to occur, and correlations between the actual and desired output vectors drop. The noise in the distributions increases and it becomes more difficult to detect a match. There appears to be an optimum level of internal feedback which is high enough to maintain signal strength and low enough to maintain signal accuracy. The 0.1 feedback range that provided the best comparison sensitivity in these simulations has been shown in previous simulations to be best for signal maintenance properties as well (see Shedden & Schneider, 1990 for other simulations dealing with feedback).

Serial and parallel processing. In the present architecture, to respond accurately, the system must serialize the comparisons. For example, if four display items must be compared to one memory item, the system inputs the first display item and the one memory item into a single comparator module. Then the second display item and the one memory item are input to the comparator, continuing until the fourth display item has been compared. In this way accuracy can be maintained although processing time increases linearly (a similar argument was made by Luce (1986, p.444) for serializing comparisons in a limited short term memory). The need for the serial processing of comparisons predicts the varied mapping search data.

How can the transition to parallel processing in consistent search (see Shiffrin, 1988) be explained in this architecture? We assume that each module can associate a priority tag for each learned vector at the hidden layer level. If there is a consistent relationship in which certain stimuli are always targets, they come to evoke a high priority relative to the distractors. Each module makes the priority assessment internally in parallel. If only one module has a high priority it transmits first. The reaction time for the first transmission does not increase with the number of stimuli. This model is detailed elsewhere (Gupta & Schneider, 1991) and provides a good fit to practice effects in consistent search tasks.

Conclusions and Summary

Although the architecture of cortex is very parallel there are operations that must be performed serially. One of these operations is the comparison process. This paper described simulations in a modular connectionist architecture incorporating central features of cortical structure. The results illustrate why a single comparator module cannot make multiple comparisons and still maintain high accuracy. Accurate performance requires high sensitivity (d') to the presence of a signal, and multiple comparisons generate too much noise for this to occur. In the present simulations matches and mismatches could only be discriminated at human performance levels if the comparisons were performed serially with modest auto-associative feedback (0.1) on the hidden layer. For single comparisons d' is high when measured on either the hidden layer or the output layer of the network. The d' measure is robust for different metrics of activity, and is sensitive to different levels of auto-associative feedback. None of the simulations involving parallel comparisons produced accuracy levels compatible with the human data. In the present architecture, even if all the modules could transmit in parallel to higher levels, the system would have to serialize the comparisons to maintain acceptable accuracy levels. The serialization necessitated in the current architecture matches the apparent serial processing of humans in varied mapping memory and visual search. This set of simulations shows the need for serial processing in parallel hardware.

References

Desimone, R. & Ungerleider, L.G. (1989). Neural mechanisms of visual processing in monkeys. In F. Boller and J. Grafman (Eds.), *Handbook of Neuropsychology*, Vol. 2:267-299. Netherlands: Elsevier.

Detweiler, M. & Schneider, W. (1991). Modeling the acquisition of dual task skills in a connectionist/control architecture. In D.Damos (Ed.) *Multiple-task performance: Selected topics.* London: Taylor & Francis.

Deutsch, J. & Deutsch, D. (1963). Attention: Some theoretical considerations. *Psychological Review*, 70:80-90.

Douglas, R.J. & Martin, K.A.C. (1990). Control of neuronal output by inhibition at the axon initial segment. *Neural Computation*, 2:283-292.

Gupta, P. & Schneider, W. (1991). Attention, automaticity, and priority learning. *Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society.*

Kahneman, D. (1973). *Attention and effort.* Englewood Cliffs, NJ: Prentice-Hall.

Luce, R.D. (1986). *Response times.* New York, NY: Oxford University Press.

Lund, J. S., Hendrickson, A. E., Ogren, M. P., & Tobin, A. E. (1981). Anatomical organization of primate visual cortex area VII. *Journal of Comparative Neurology*, 202:19-45.

McClelland, J. L., & Rumelhart, D. E. (1988). (Eds.), *Explorations in parallel distributed processing: A handbook of models, programs, and exercises.* Cambridge, MA: MIT Press.

Moran, J., & Desimone, R. (1985). Selective attention gates visual processing in the extrastriate cortex. *Science*, 229:782-784.

Mountcastle, V.B. (1979). An organizing principle for cerebral cortical function: The unit module and the distributed system. In F.O. Schmitt & F.G. Worden (Eds.), *The neurosciences: Fourth Study Program* (pp.21-42). Cambridge, MA: MIT Press.

Pashler, H. & Badgio, P.C. (1987). Attentional issues in the identification of alphanumeric characters. In M. Coltheart (Ed.), *Attention and Performance XII* (pp. 63-81). Hillsdale, NJ: Erlbaum.

Peters, A. (1984). Chandelier cells. In A. Peters & E. G. Jones (Eds.), *Cerebral cortex: Vol. 1: Cellular components of the cerebral cortex* (pp. 361-381). New York: Plenum.

Proctor, R.W. & Healy, A.F. (1985). Order-relevant and order-irrelevant decision rules in multiletter matching. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, 11:519-537.

Proctor, R.W. & Healy, A.F. (1987). Task-specific serial position effects in comparisons of multiletter strings. *Perception & Psychophysics*, 42:180-194.

Proctor, R.W., Healy, A.F. & Van Zandt, T. (1991). Same-different judgments of multiletter strings: Insensitivity to positional bias and spacing. *Perception & Psychophysics*, 49:62-72.

Ratcliff, R. (1988). Order information and distributed memory models. In *Proceedings of the Ninth Annual Conference of the Cognitive Science Society* (pp. 474-486). Hillsdale, NJ: Erlbaum.

Ratcliff, R. & Hacker, M.J. (1981). Speed and accuracy of same and different responses in perceptual matching. *Perception & Psychophysics*, 30:303-307.

Rumelhart, D.E., Hinton, G.E., & Williams, R.J. (1986). Learning internal representations by backpropagating errors. *Nature*, 323:533-536.

Schneider, W. (1985). Toward a model of attention and the development of automaticity. In M. Posner & O. S. Marin (Eds.), *Attention and performance XI* (pp. 475-492). Hillsdale, NJ: Erlbaum.

Schneider, W. & Detweiler, M. (1987). A connectionist/control architecture for working memory. In G. H. Bower (Ed.), *The psychology of learning and motivation, Volume 21* (pp. 54-119). New York: Academic Press.

Schneider, W. & Oliver, W.L. (1991) An instructable connectionist/control architecture: Using rule-based instructions to accomplish connectionist learning in a human time scale. In K. VanLehn (Ed.) *Architecture for intelligence.* Hillsdale, NJ: Erlbaum. Forthcoming.

Schneider, W. & Shiffrin, R.M. (1977). Controlled and automatic human information processing: I. Detection, Search, and Attention. *Psychological Review*, 84:1-66.

Shedden, J.M. & Schneider, W. (1990). A connectionist model of attentional enhancement and signal buffering. In *Proceedings of the Twelfth Annual Conference of the Cognitive Science Society* (pp. 566-573). Hillsdale, NJ: Erlbaum.

Shiffrin, R. M. (1988). Attention. In R. C. Atkinson, R. J. Herrnstein, G. Lindzey & R. D. Luce (Eds.), *Steven's handbook of experimental psychology, Volume 2: Learning and cognition* (739-809). New York: John Wiley & Sons.

Shiffrin, R. M., & Schneider, W. (1977). Controlled and automatic human information processing. II: Perceptual learning, automatic attending, and a general theory. *Psychological Review*, 84:127-190.

Sternberg, S. (1969). Memory-scanning: Mental processes revealed by reaction-time experiments. *American Scientist*, 57:421-457.

Tanner, W.P. & Swets, J.A. (1954). A decision-making theory of visual detection. *Psychological Review*, 61:401-409.

Townsend, J.T. & Ashby, F.G. (1983). *Stochastic modeling of elementary psychological processes.* New York, NY: Cambridge University Press.

Treisman, A.M. (1985). Preattentive processing in vision. *Computer Vision, Graphics, & Image Processing*, 31:156-177.

Treisman, A.M. & Gelade, G. (1980). A feature-integration theory of attention. *Cognitive Psychology*, 12:97-136.

Treisman, A. M., & Riley, J. G. A. (1969). Is selective attention selective perception or selective response? A further test. *Journal of Experimental Psychology*, 79:27-34.