In: Computational Neuroscience: Trends in Research, 1997. Proceedings of the Annual Computational Neuroscience Conference. (J. M. Bower, Ed.) New York:Plenum Press, pp. 885-889, 1997.

Note:
Fig. 1 was improperly turned sideways by the publisher.

A SPECIAL ROLE FOR INPUT CODES IN SOLVING THE TRANSVERSE PATTERNING PROBLEM

Xiangbao Wu, Joanna M. Tyrcha, and William B. Levy

Department of Neurological Surgery
University of Virginia Health Sciences Center
Charlottesville, Virginia 22908
xw3f@virginia.edu
joan@sans.kth.se
wbl@virginia.edu

1. ABSTRACT

Rats require a hippocampus to solve the transverse patterning problem. Here, a hippocampal model also solves this configural learning problem. The problem is hard: A learning paradigm, called progressive learning, is required. It is required by rats, humans, and the model. Second, input patterns within a sequence must be repeated. Such repetition increases the statistical dependence, a surprising observation if you assume statistical dependence is undesirable. Such repetition of the same patterns in a sequence facilitates the formation of local context neuronal firings. These neuronal firings are critical, and we hypothesize that they are analogous to place cells found in behaving animals.

2. INTRODUCTION

2.1. Configural Learning and the Problem of Transverse Patterning

Recently, Alvarado and Rudy (1995) demonstrated that the hippocampus is necessary for a configural learning problem called transverse patterning. In this problem, the meaning of a particular stimulus depends upon the context supplied by the other stimuli that are simultaneously present.

Consider three atomic stimuli W, X, and Y. Let these stimuli be presented as pairs including (WX), (XY), and (WY). Then, reinforce behavior in the following manner: for the (WX) pair, W is the correct answer; for the (XY) pair, X is the correct answer; and for the (YW) pair, Y is the correct answer. Thus, each individual stimulus is equally rewarded and punished, and the only way to solve such a problem is to consider the stimulus complex. In this sense the transverse patterning problem requires a system that can learn and use the context provided by the configuration of stimuli themselves. To say it another way, meaning and predictability come by virtue of the context that arises from the pairing of stimuli.

Because an intact hippocampus is necessary for a rat to learn this problem, it is important that a hypothesized model of the hippocampus also solve the problem. Because our hippocampal model is a sequence learning system, we turned the configural learning problem into a very simple sequence. The first pattern in the sequence is the configured stimulus such as we have just described above (e.g., XY). The second pattern in the sequence is a randomly selected motor response that represents a response that chooses one of the two atomic stimuli (e.g., response x chooses X), and the third pattern in the sequence would be either the positive or negative reinforcement as appropriate (e.g., reward =+). Figure 1a shows one such input sequence and some of the recurrently activated neurons. The system is tested as suggested by the method of goal finding (Levy et al. 1995; Wu & Levy 1996; Levy & Wu, 1997). That is, the desired outcome (reward =+) is partially turned on along with a test configuration of paired stimuli.

3. THE NETWORK

The hippocampal model is essentially a model of region CA3 (see Fig. 1 in Levy & Wu, 1997). The input layer corresponds to a combination of the entorhinal cortex and dentate gyrus. To make the system's operation as transparent as possible, decoding is performed by similarity comparisons rather than a CA1-subiculum-entorhinal decoding system. The CA3 model is a sparsely (10%) interconnected feedback network of 512 neu-

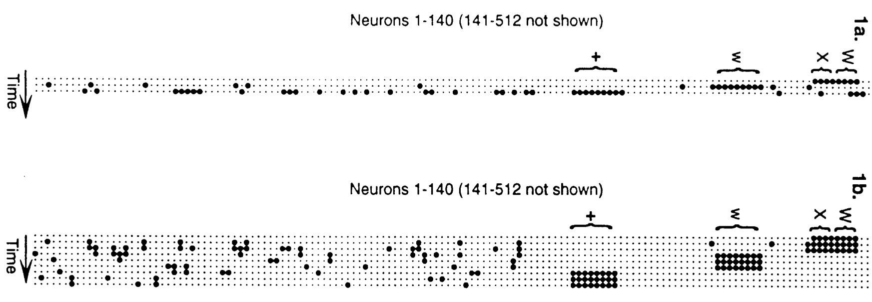

Figure 1. Configural learning requires the development of local context units. CA3 activities at the end of training for two types of inputs. The simple orthogonal input sequence of 1a does not produce local context neurons or learning while the same input sequence "stuttered," in 1b, produces local context neural firings and the appropriate learned "behavior." The two sequences of neural firing illustrated here are for a sequence that is part of the configural learning problem. Here we illustrate an WX trial (W = neurons 3–6, X = 7–10, the response of choice W is represented by neurons 19–26, and the + reinforcement given for this correct response is represented by externally activating neurons 43–50). All other neuronal firings are driven by recurrent connections. In 1b, the externally driven inputs are repeated three times while in 1a just a single pattern of each is given. As a result, in 1b, some of the recurrently driven neurons fire repetitively and selectively in time. Such repetitive firing can be asynchronous relative to all the externally driven input patterns. Note also that these local context neurons overlap with one another so that they can efficiently pass their information on from one time step to another. Only 140 neurons of the 512 are illustrated due to space limitations. Neuron 1 is at the top of the page, neuron 140 is at the bottom.

rons where all direct connections are excitatory and the network elements are McCulloch-Pitts neurons. Inhibition is of the divisive form, but the system is not purely competitive because of a slight delay. Synaptic modification is a postsynaptic rule that includes both potentiation and depression aspects (Levy & Steward 1979; Levy 1982). For details see equations in Levy & Wu, 1997.)

4. RESULTS

We found that this model of the hippocampus could not learn the transverse patterning problem when transverse patterning is coded as described above and when each input just activated neurons orthogonal to all other inputs and patterns. However, when we repeated inputs (as in Fig. 1b) so that instead of giving the input WX, then the response "w", then +, we gave inputs WX, WX, WX, response "w", response "w", response "w", followed by +, +, +, then the network was able to correctly perform the transverse patterning problem with one proviso — the training paradigm effects learnability.

Alvarado and Rudy (1992) noted a controversy — results between labs are contradictory on the learnability of this problem — and they explained its basis. When learning trials totally intermix all stimuli pairs, then even college sophomores, as well as rats, are largely incapable of discovering the correct solution to the transverse patterning problem. However, when a special learning paradigm called, progressive learning, is used, then humans and rats are able to learn and solve the transverse patterning problem. This is exactly the same observation for our network. When the network is taught all three pairs totally intermixed, then the system fails. But when the task is learned gradually, progressively (see Table 1), the network performs the task as successfully as rats do. In the progressive learning paradigm, there is the progression from training on one pair, to training on two pairs, to training on all three pairs. Because our model reproduces this requirement for progressive learning, i.e., it fails when learning is randomized, we are further encouraged to believe in the usefulness of this computational model of the hippocampus. We return now to consider the importance of the input code for controlling the development of local context neurons.

4.1. Local Context Neurons

In studying the transverse patterning problem the network sees inputs that are only sequences made up of orthogonal patterns. Apparently, without capacitive elements and with the narrow time span of synaptic associative modification of this study, such orthogonal sequences do not allow the formation of local context neurons. (Presumably, these

Table 1. The progressive learning paradigm of transverse patterning

	Stage 1.	
		Learn W is correct when WX is the stimulus pair
then	Stage 2.	
		Learn W is correct when WX is the stimulus pair and Learn X is correct when XY is the stimulus pair
then	Stage 3.	
		Learn W is correct when WX is the stimulus pair and Learn X is correct when XY is the stimulus pair and Learn Y is correct when WY is the stimulus pair

neurons are an analogy of place cells that have been discovered in hippocampus (O'Keefe & Nadel 1978). Like place cells, local context neurons fire in response to places in a sequence.)

4.2. Sequence Completion and Local Context Firing with Stuttered Inputs

In general, orthogonal input sequences do not promote local context firing unless patterns are repeated. We have systematically investigated this "preprocessing" and quantified the average length of local context neuron firing and the amount of "stuttering" (i.e., repetition of elements) within the input sequence. In fact, the network can produce local context neuron firing time spans that exceed the length of input stuttering. It is this greater time spanning ability of some recurrently activated neurons that accounts for the appropriate performance by the network in the transverse patterning problem.

We studied the effect of repeating inputs directly on simple sequence completions. For example, we compared CA3 codes for the sequences wxyz and the sequence wwwwxxxxyyyyzzzz. With an activity level of 15%, for instance, such a four-fold stuttering gives an average context neuron firing time span of seven to eight time steps. Such stuttering of the input and creation of context neurons is not without its problems. Sometimes stuttering reduces the sequence length memory capacity (when this capacity is a count of the number of different patterns). For example, at 7.5% activity and a quadrupling of inputs, sequence length capacity goes down approximately 25%.

5. DISCUSSION

Preprocessing an input by repetitively presenting it to the network increases the statistical dependency of the input. In contrast to a prevalent theme of neural computing concerning the importance of lowering statistical dependency — a theme which we support in general — there are certain, very clear situations where statistical dependency should be retained or even increased in order to improve performance (see Levy & Adelsberger-Mangan 1995 for example). Here we have another example where added redundancy helps a neural computation.

6. ACKNOWLEDGMENTS

This work was supported by NIH MH48161 and MH00622, by EPRI RP8030-08, and Pittsburgh Supercomputing Center Grant BNS950001 to WBL, and by the Department of Neurosurgery, Dr. John A. Jane, Chairman. The authors also thank Dr. A. Lansner of the SANS Research Group, Department of Numerical Analysis and Computing Science, Royal Institute of Technology, Stockholm, Sweden for his graciousness and generosity in allowing us access to the SANS computing facilities.

7. REFERENCES

Alvarado, M.C. & Rudy, J. W. Some properties of configural learning: An investigation of the transverse-patterning problem. J. Exp. Psychol.:Animal Behav. Processes 18, 1992, 145–153.

Alvarado, M. C. & Rudy, J. W. Rats with damage to the hippocampal-formation are impaired on the transverse-patterning problem but not on elemental discriminations. Behav. Neurosci. 109, 1995, 204–211.

Levy, W. B Associative encoding at synapses. Proceedings of the Fourth Annual Conference of Cognitive Science Society, 1982, 135–136.

Levy, W. B & Adelsberger-Mangan, D. M. Is statistical independence a proper goal for neural network preprocessors? INNS World Congress on Neural Networks, 1995, I-527–531.

Levy, W. B & Steward, O. Synapses as associative memory elements in the hippocampal formation. Brain Res. 175, 1979, 233–245.

Levy, W. B & Wu, X. B. Predicting novel paths to goals by a simple, biologically inspired neural network. CNS*96, 1997, this proceedings.

Levy, W. B, Wu, X. B. & Baxter R. A. Unification of hippocampal function via computational/encoding considerations. In: Proceedings of the Third Workshop: Neural Networks: from Biology to High Energy Physics. International J. of Neural Sys. 6 (Supp.), 1995, 71–80.

O'Keefe, J & Nadel, L. The Hippocampus as a Cognitive Map. Oxford: Oxford Univ. Press, 1978.

Wu, X. B. & Levy W. B, Goal Finding in a Simple, Biologically Inspired Neural Network. INNS World Congress on Neural Networks, 1996, 1279–1282.