Quantal Synaptic Failures Improve Performance in a Sequence Learning Model of Hippocampal CA3

D.W. Sullivan and W. B Levy

Department of Neurological Surgery

University of Virginia Health Sciences Center

Charlottesville, VA 22908, USA

email: dws3t@virginia.edu,wbl@virginia.edu

Keywords*:* neural network, fluctuations, noise, configural learning, multiplicative noise

Abstract

Quantal synaptic failures are random, independent events in which the arrival of an action potential fails to release transmitter. Although quantal failures destroy information, we have discovered conditions where such failures enhance learning by a neural network model of the hippocampus. By suitably adjusting network parameters and failure rate, the model can learn a hippocampally dependent task: transverse patterning. Usefully, synaptic failures lead to robust performance at lowered activity levels. Because lowered activity levels produce higher memory capacity, the synaptic failure mechanism is an example of a random fluctuation that improves neural network computations.

1. Introduction

Examined in isolation, noisy communications channels are less efficient than noise-free communications channels, as they require more energy to produce equivalent capacity. Although noisy communication is bad for the users of a single communications channel, it can have a positive effect on the performance of a network of intercommunicating computational devices. Previous studies have demonstrated a positive role for noisy communication in simulated neural network computation. Jim et al. [2] demonstrated that noisy synaptic weights improve generalization performance in a recurrently connected network employing the backpropagation learning algorithm. However, generalization is not the issue addressed here.

Wu & Levy [7] and Shon et al. [6] investigated random fluctuations in a recurrently connected, Hebbian learning, neural network model of hippocampal CA3. Both studies found better simulated performance on transverse patterning when the initial firing state of the simulation is randomized before each training trial. While

Wu & Levy demonstrated the benefits of initial state randomness for the transitive inference task, Shon et al. shows that random initial firing states drive a search-like process that finds robust neural codes, aiding simulations in solving the transverse patterning task.

In this paper, we investigate the effects of introducing a biologically accurate, multiplicative form of noise, quantal synaptic failures, at the recurrent excitatory synapses of our hippocampal CA3 model. In the neocortex, when a neuron fires, an action potential always travels down the neuron's axon and its branches. However, while an action potential is a necessary condition for presynaptic transmitter release, transmitter is not always released in response to an arriving action potential. An event in which transmitter release does not follow an action potential is termed a *quantal synaptic failure* [3], or more briefly, a *synaptic failure*. Here, we show that synaptic failures can improve network learning on the cognitive task of transverse patterning (TP).

To summarize the results, there are two primary beneficial effects of synaptic failures. First, proper adjustment of the synaptic failure rate results in model parameterizations that produce more robust performance of TP. The second benefit of synaptic failures is that the model is capable of robust performance of TP at lower activity levels when a nonzero synaptic failure rate is used. Lower activity levels are desirable because they allow more memory capacity and because these levels are closer to values observed in biology.

2. Methods

Our model of hippocampal CA3 consists of a sparsely connected (10%) network of McCulloch-Pitts neurons. External inputs represent projections from the dentate gyrus and entorhinal cortex to CA3.

In previous incarnations of our model, a neuron's excitation was defined as the sum of the weights of its recurrent inputs that were active on time step (t -1). However, with the addition synaptic failures to the model, a neuron's excitation is the sum of the weights of its active inputs that did not undergo synaptic failure. We model synaptic failures as independent, random, binary events. For ($y_j(t)$), the internal excitation of neuron j on time step t,

$$y_j(t) = \sum_{i=1}^{N} C_{ij} W_{ij}(t-1) Z_i(t-1) \Phi_{ij}(t),$$

where N is the total number of neurons. A synaptic connection from neuron i to neuron j is represented by C_{ij} (1 if a connection is present, 0 if a connection is not present). The weight of the synapse from neuron i to neuron j on time step t is represented by W_{ij}(t). The variable $Z_i(t\text{-}1)$ represents the binary, $[Z_i(t\text{-}1) \in \{0,1\}]$, firing of neuron i

on time step ($t - 1$). The binary variable $\Phi_{ij}(t)$ implements the failure process in the synapse from neuron *i* to neuron *j*. All $\Phi_{ij}(t)$ are randomly and independently determined on each trial such that $\Phi_{ij}(t) = 1$ with probability (1- *f*), where *f* is the probability of synaptic failure.

Once internal excitation has been calculated, a deterministic binary (0/1) output firing decision occurs, representing the absence or presence of an action potential respectively. On every time step, firing is determined competitively. That is, only the top *A*% of neurons, ordered according to excitation, fire (where *A* is percent neurons active per time step). Any neuron that is part of the external input code at the current time step is automatically included in this top *A*%, regardless of its internal excitation.

Simulations include two types of trials: test and training. During training, the network is presented with a pre-specified input firing sequence, and synaptic modification is allowed to occur. A training trial begins at time step 1; only those externally activated neurons that are part of the first pattern of the input sequence fire on time step 1. No synaptic modification takes place during a test trial.

This model learns via locally driven synaptic modification, using a time-dependent associative synaptic modification rule. On each time step of each training trial, the weight of every synapse in the network is updated according to the following equation:

$$W_{ij}(t) = W_{ij}(t-1) + \mu Z_j(t)\left(Z_i(t-1)\Phi_{ij}(t-1) - W_{ij}(t-1)\right),$$

where the synaptic modification rate, μ, is equal to 0.05. Synaptic modification occurs only if postsynaptic neuron *j* fires. At the beginning of a simulation, all weights have a value of 0.4. With enough training, synaptic weights converge towards (1- *f*)*E[$Z_i(t\text{-}1) \mid Z_j(t) = 1$].

Transverse patterning is a hippocampally dependent cognitive task that requires the subject to assign meaning to stimulus pairs that cannot be derived from the individual stimuli [1]. To succeed at transverse patterning, the subject must learn the correct choice for each of three stimulus pairs. Thus, if we denote the stimulus items as A, B, and C, then the subject must learn the relationships A > B, B > C, and C > A. The circularity of these relationships prevents any single stimulus item from being consistently correct or incorrect, forcing the subject to derive meaning from the pairings, rather than the individual stimuli.

In our simulations, we train the model on transverse patterning as a sequence learning problem [8]. For each stimulus pair (AB, BC, CA), we train the model on two input sequences, one representing the correct choice, and one representing the incorrect choice. Each sequence is constructed from three patterns: first, a pattern representing

the stimulus pair; then, a pattern representing the choice of one of the stimulus items; and lastly, a pattern representing whether the choice resulted in a positive or negative outcome. Each pattern is presented for three time steps, making each input sequence nine time steps long. Training uses the progressive schedule [5], and lasts 300 trials.

Three test trials (one for each stimulus pair) occur after each of the last 30 training trials. During a test trial, the stimulus pattern is presented along with the first third of the positive outcome pattern neurons. The test trial is successful if more recurrent firing occurs in the correct decision pattern than in the incorrect decision pattern during time steps 4-6. A simulation is deemed to have successfully learned TP if it can give the correct answer for all stimulus pairs in at least 26 out of the 30 series of test trials. Figure 1 shows a TP test trial before and after training has occurred. In the left diagram of figure 1 (before training), neural firing is randomly distributed between each of the decision patterns, as the network has not been trained on TP. In the right diagram of figure 1 (after training), the network's correct response to the input is clearly not a, nor b, but c. These c neurons are activated by recurrent firing in the rest of the network (neurons 176-2048 not shown).

3. Results

Figure 1 demonstrates the model's sensitivity, at a particular set of parameters, to the rate of synaptic failure. As rate of synaptic failure is increased from 0 to 10%, TP learning increases from 0 to 80% of the simulations. For a wider range of failure rates, 30% to 70%, five out of five simulations successfully learn TP. However, when synaptic failure rates exceed 70%, performance drops abruptly. Thus, very small differences in synaptic failure rate can lead to very large differences in TP performance rate.

Synaptic failure rate interacts with the other simulation parameters, such as number of neurons, activity level, learning rate, initial weight, and size of the input code. The range of synaptic failure rates that are beneficial changes according to the other parameters. The number of neurons in the network is especially important. Although synaptic failures do not benefit networks smaller than 2000 neurons, the effects of synaptic failures become more robust as network size increases. This effect is likely due to an insufficient number of active connections per neuron in the smaller networks.

The interaction of synaptic failure rate with activity level is, perhaps surprisingly, beneficial. Although one might expect that lowered activity together with synaptic failures would lead to too few active inputs per neuron

(and this can happen when failure rate is too high and activity level is too low), it is not always so. Figure 3 shows TP performance rates for simulations run at four activity levels and two synaptic failure rates. Without synaptic failures, simulations require activity levels of 10% in order to achieve good (>80%) TP performance. But when synaptic failures are present (here, set to 70%), simulations can learn TP with 100% success at each of the four activity levels plotted.

A lowered activity level not only moves this network property toward a more biological value; it also has the beneficial effect of increasing sequence length capacity. Previous studies have shown that lowering activity level can increase the model's memory capacity by decreasing the number of recurrent neurons used to re-code the input sequence [4]. The present study supports those findings; as activity is lowered, fewer neurons are used. Moreover, this trend persists when synaptic failures are used, and seems to be enhanced. For example, at 10% activity and 70% probability of synaptic failure, 7375 out of 8192 neurons are used to learn TP (on average); whereas at 7% activity and 70% probability of synaptic failure, 4310 out of 8192 neurons are used to learn TP (on average). Thus number of unused neurons goes from 817 to 3882 when the synaptic failure process is included and activity is dropped from 10% to 7%.

4. Discussion

Synaptic failures can benefit computation via greater memory capacity and more robust performance. Apparently, the rate of synaptic failure is a powerful parameter that can have profound effects on the model's ability to learn transverse patterning. Specifically, simulations with low activity levels can learn TP only if synaptic failures are present in the model. While information is lost in synaptic transmission, the overall computation is more robust. Future studies will investigate the basis of this effect, but the idea in Shon et al. (in press) concerning fluctuation driven code and search during training seems to be a good candidate.

5. Acknowledgements

This work was supported by NIH MH48161 to WBL and MH57358 to N. Goddard (with subcontract # 163194-54321 to WBL).

6. References

[1] M. C. Alvarado and J. W. Rudy, 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.

[2] K. Jim, C. L. Giles, B. G. Horne, An analysis of noise in recurrent neural networks: convergence and generalization, IEEE Trans. Neural Networks 7 (1996) 1424-1438.

[3] B. Katz, Nerve, Muscle, and Synapse (McGraw-Hill, New York, 1966).

[4] W. B Levy and X. B. Wu, The relationship of local context codes to sequence length memory capacity, Network 7 (1996) 371-384.

[5] A.P. Shon, X.B. Wu, W. B Levy, Using computational simulations to discover optimal training paradigms, Neurocomputing 32-33 (2000) 995-1002

[6] A. P. Shon, X.B. Wu, D.W. Sullivan, and W. B Levy, Initial state randomness improves sequence learning in a model hippocampal network, Physical Review E, in press

[7] X. B. Wu and W. B Levy, Enhancing the performance of a hippocampal model by increasing variability early in learning, Neurocomputing 26-27 (1999) 601-607

[8] X. B. Wu, J. Tyrcha, and W. B Levy, A neural network solution to the transverse patterning problem depends on repetition of the input code, Biol. Cybern. 79 (1998) 203-213

Figure Legends

Figure 1: Training is required for a simulation to make a correct decision on the transverse patterning problem. Here, we plot firing diagrams in response to a CA stimulus pair, both before training (left) and after training (right). Before training, neural firing is random, and the network shows no particular preference for either of the three decisions, {a, b, c}. After training, recurrent firing is restricted exclusively to neurons representing decision 'c', which is the correct decision when the CA stimulus pair has been presented. In both diagrams, the A and C stimulus patterns are externally activated on time steps 1-3, and the positive outcome pattern is externally activated on all nine time steps. Each column of the diagrams represents one time step, and each row represents one neuron. Firing neurons ($Z_j(t) = 1$) are indicated by large dots, and non-firing neurons ($Z_j(t) = 0$) are indicated by small dots. Only neurons 1-175 are plotted here. Since the connectivity of the network is random, the spatial juxtaposition of any two neurons is irrelevant. Simulation parameters: $N = 2048$, $A = .07$, $f = 0.3$.

Figure 2: Synaptic failure improves TP performance. Running at a relatively low activity level (7%), simulations are incapable of successfully completing the TP problem without synaptic failures. However, as the probability of synaptic failures is increased, TP performance rate rises quickly. While simulations display robust performance when synapses fail between 30% and 70% of the time, a synaptic failure rate that is too high (>80%) destroys performance. Simulation parameters: $N = 8192$, $A = 0.07$, # of external inputs = 172, $\mu = 0.05$.

Figure 3: Synaptic failures allow better TP performance, with the greatest improvement found at lower activity levels. When synaptic failure rate is set to zero, performance at 7%, 8%, and 9% activity is poor, with less than three out of five simulations successfully completing the TP task. However, when the probability of synaptic failure is set to 70%, five out of five simulations at each of the activity levels plotted learned TP. Each point is the average success rate of 5 simulations, each simulation using a different random connectivity matrix. For each simulated activity level, the external input comprises 30% of the active neurons. Simulation parameters: $N = 8192$, $\mu = 0.05$.

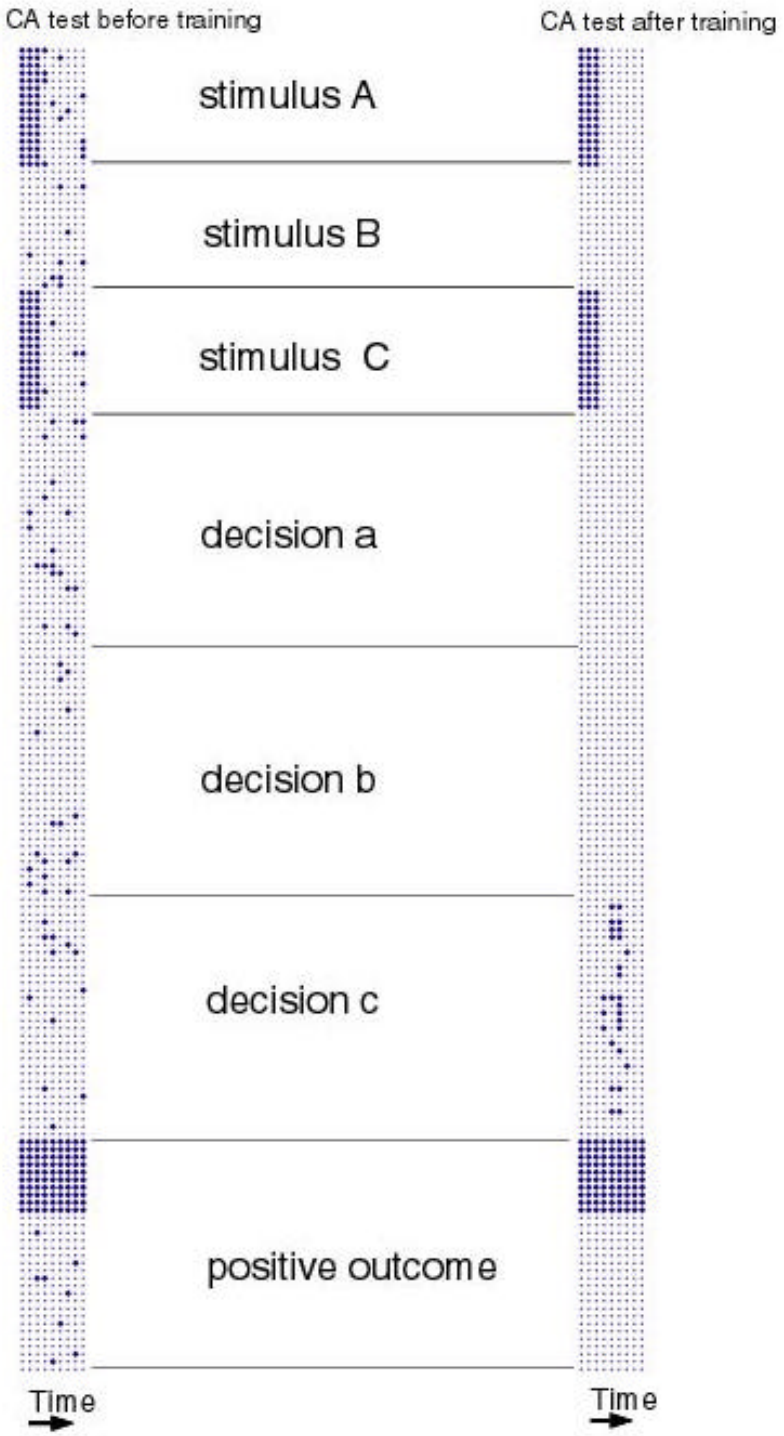
CA test before training
CA test after training
stimulus A
stimulus B
stimulus C
decision a
decision b
decision c
positive outcome
Time
Time

Figure 1

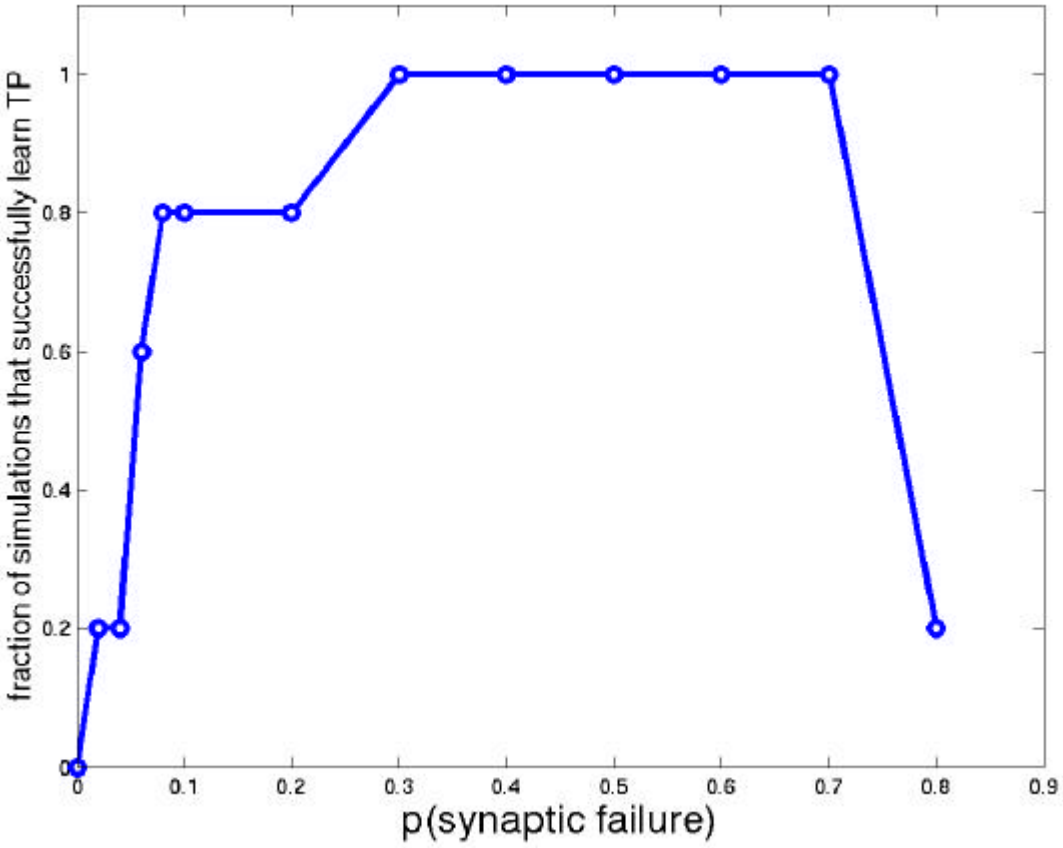
fraction of simulations that successfully learn TP
p(synaptic failure)
0
0.2
0.4
0.6
0.8
1
0.1
0.3
0.5
0.7
0.9

Figure 2

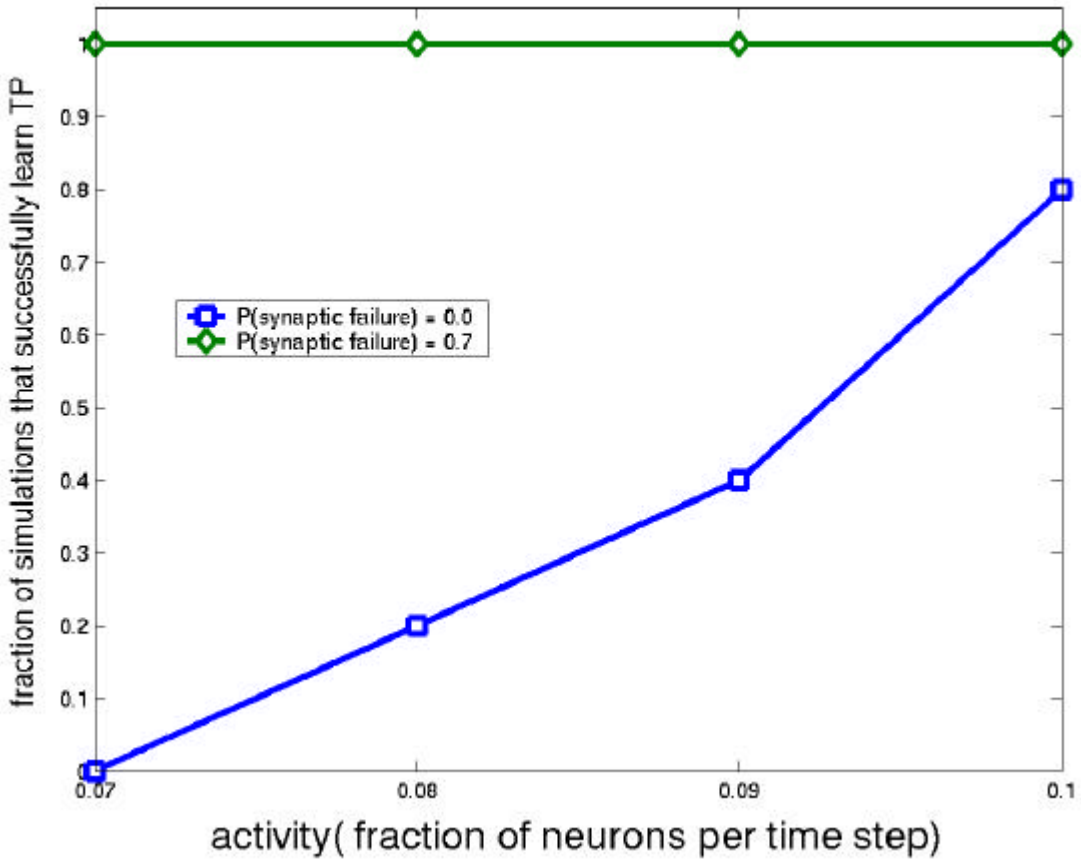
fraction of simulations that successfully learn TP
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
P(synaptic failure) = 0.0
P(synaptic failure) = 0.7
0.07
0.08
0.09
0.1
activity(fraction of neurons per time step)

Figure 3