

UC Merced

Proceedings of the Annual Meeting of the Cognitive Science Society

Title

The Role of Neurocomputational Principles in Skill Savings

Permalink

<https://escholarship.org/uc/item/5m63x7v1>

Journal

Proceedings of the Annual Meeting of the Cognitive Science Society, 27(27)

ISSN

1069-7977

Authors

Gupta, Ashish
Noelle, David C.

Publication Date

2005

Peer reviewed

The Role of Neurocomputational Principles in Skill Savings

Ashish Gupta & David C. Noelle

({ASHISH.GUPTA, DAVID.NOELLE}@VANDERBILT.EDU)

Department of Electrical Engineering and Computer Science
Vanderbilt University, Box 1679 Station B Nashville, TN 37235, USA

Abstract

Humans exhibit *savings* in skills. A skill is rarely forgotten completely, even if it remains unused for long periods. Also, the reacquisition of a skill to its previous level of competence is faster than initial skill learning. Traditional artificial neural network models of skill learning have been unable to exhibit savings comparable to that seen in humans because they suffer from *catastrophic interference*. They are commonly trained to perform only one specific task, and when trained on a new task, they forget the original task completely. A number of specialized connectionist architectures and learning rules have been suggested as means to avoid catastrophic interference. Instead of introducing such a new mechanism, we have investigated the degree to which the foundational neurocomputational principles embodied by the Leabra cognitive modeling framework are sufficient to ameliorate catastrophic interference. In particular, this framework includes both fast lateral inhibition and a local synaptic plasticity model that incorporates both Hebbian and error-based dynamics, grounded in known properties of cortical circuits. In this paper, we provide evidence that these fundamental computational properties of neural circuits can support savings during sequential learning of multiple motor skills.

Introduction

Learned motor skills are central to many activities, from bicycle riding to piano playing, from typing to playing ping-pong. To maintain and improve our motor skills, we need practice. If a ping-pong player does not play for a while, it is likely that his proficiency will wane, but the skill will not be forgotten completely. Furthermore, reacquisition of proficiency will typically be rapid, as compared to the period of initial learning. This retention of skill knowledge, sometimes in a latent form, is called *savings*.

Why do skills degrade when unpracticed? What is the neural basis of skill savings? A common explanation of proficiency loss involves interference. The initial acquisition of a skill is driven by synaptic plasticity shaped by experience. Plasticity continues once practice on that skill ceases. Thus experience with other activities continues to shape neural circuits, often in a way that interferes with the proper performance of the original skill. The interfering task experience may modify the response properties of neurons that are directly involved in the performance of the original skill, or the interference might take the form of a strengthening of the response of competing neurons that encode the new

behaviors. From this perspective, savings arise when the interference produced by experience with new behaviors is insufficient to completely erase the synaptic modifications introduced by the initial learning of the skill. This lack of “unlearning” could be due to neural specialization, with some neurons being employed by the first skill but not by subsequent activities. To the degree that the sets of neurons associated with different skills are disjoint, learning one skill will not affect the synapses associated with another. When neurons are shared between skills, savings could be due to sub-threshold residual synaptic weights associated with the initial skill — weights that have been driven down by interfering experiences to below the threshold for neural firing, but not all the way down to their initial values. Finally, skills may share components or “sub-tasks”. To the degree that such components have isolated neural representations, learning a new skill may actually reinforce portions of a previously learned skill.

Traditional artificial neural network models of skill acquisition fail to display savings when skills are learned sequentially. Instead, these networks exhibit “catastrophic interference”, where the later learning of a second related skill obliterates essentially all knowledge of an initially acquired skill (McCloskey and Cohen, 1989). Researchers have proposed a number of specialized neural network architectures and learning algorithms designed to reduce catastrophic interference (French, 1994; Brashers-Krug et al., 1995; McClelland et al., 1995). Most of these proposed mechanisms involve isolating the sets of neurons associated with different skills, either through some form of explicit architectural modularization, or through the use of learned *sparse representations*, where only a few neurons in some internal layer of the network are highly active at any one time.

We have explored the possibility that computational models of skill acquisition need not posit dedicated mechanisms for shielding from catastrophic interference. Instead, it is possible that biological constraints imposed by the structure of cortical circuitry may embody the necessary properties to promote skill savings. Specifically, we have examined the neurocomputational principles forming the Leabra cognitive modeling framework (O’Reilly and Munakata, 2000), and we have found that these biologically motivated principles give rise to savings without the need for auxiliary mechanisms. We trained a Leabra network to produce motion trajectories

for a three joint planar arm. After an initial trajectory was mastered, an interfering trajectory was taught, and savings was assessed on the retention of knowledge concerning the initial trajectory. Our findings suggest that Leabra’s implementation of fast acting lateral inhibition acts in concert with its synaptic plasticity mechanism in order to produce adequately sparse representations to support skill savings.

In the next section, we provide a brief overview of related work. We follow that with a description of our model simulation experiments. Then, we offer the results of our experiments, and we close with a general discussion.

Background

Leabra

The Leabra framework offers a collection of integrated cognitive modeling formalisms that are grounded in known properties of cortical circuits but are sufficiently abstract to support the simulation of behaviors arising from large neural systems (O’Reilly and Munakata, 2000). It includes dendritic integration using a point-neuron approximation, a firing rate model of neural coding, bidirectional excitation between cortical regions, fast feedforward and feedback inhibition, and a mechanism for synaptic plasticity. Of particular relevance to skill savings are Leabra’s lateral inhibition formalism and its synaptic learning rule.

The effects of inhibitory interneurons tend to be strong and fast in cortex. This allows inhibition to act in a regulatory role, mediating the positive feedback of bidirectional excitatory connections between brain regions. Simulation studies have shown that a combination of fast feedforward and feedback inhibition can produce a kind of “set-point dynamics”, where the mean firing rate of cells in a given region remains relatively constant in the face of moderate changes to the mean strength of inputs. As inputs become stronger, they drive inhibitory interneurons as well as excitatory pyramidal cells, producing a dynamic balance between excitation and inhibition. Leabra implements this dynamic using a *k-Winners-Take-All (kWTA)* inhibition function that quickly modulates the amount of pooled inhibition presented to a layer of simulated cortical neural units based on the layer’s level of input activity. This results in a roughly constant number of units surpassing their firing threshold. The amount of lateral inhibition within a layer can be parameterized in a number of ways, with the most common being the percentage of the units in the layer that are expected, on average, to surpass threshold. A layer of neural units with a small value of this *k* parameter (e.g., 10-25%) will produce sparse representations, with few units being active at once.

With regard to learning, Leabra modifies the strength of synaptic connections in two primary ways. An error-correction learning algorithm changes synaptic weights so as to improve network task performance. Unlike the backpropagation of error algorithm, Leabra’s error-correction scheme does not require the biologically implausible communication of error information backward

across synapses. In addition to this error-correction mechanism, Leabra also incorporates a Hebbian correlational learning rule. This means that synaptic weights will continue to change even when task performance is essentially perfect. This form of correlational learning allows Leabra to capture certain effects of overlearning.

We have investigated the degree to which the sparse representations enforced by Leabra’s lateral inhibition mechanism, in conjunction with Leabra’s synaptic learning rule, cause Leabra simulations of cortical circuits to escape the pitfalls of catastrophic interference when those circuits are required to sequentially learn multiple temporally-extended motor trajectories.

Catastrophic Interference

Many past studies have shown that artificial neural networks suffer from catastrophic interference in a manner uncharacteristic of human performance. The seminal example involves an AB-AC paired-associate list-learning task, in which the learning of a second list of paired-associates was shown to interfere with memory for an initially studied list in a much more mild way than predicted by a backpropagation network model (McCloskey and Cohen, 1989).

Since this observation was made, a number of computational mechanisms have been proposed for avoiding catastrophic interference. Some of these involve segregating the neural units associated with different skills in order to avoid the damage caused by “reuse” of synaptic weights (French, 1999). For example, forcing layers of neural units to form sparse representations reduces the probability that a given unit will be active while performing multiple skills and thereby reduces the probability of interference when learning the skills in sequence. Leabra offers a biologically justified mechanism for producing sparse representations. With a low *k* parameter, Leabra’s kWTA lateral inhibition implementation limits the overlap between the neural representations used for different skills. This has been shown to improve performance on the AB-AC list-learning task (O’Reilly and Munakata, 2000). We have found that the benefits of kWTA inhibition extend to the learning of motor sequences, and we have systematically studied the effects of varying sparsity and layer size on savings.

One extreme form of segregation between neurons devoted to different skills involves isolating them into discrete modules. Modular artificial neural network architectures have been proposed in which differences between skills are explicitly detected during learning, and a “fresh” module of neural units is engaged to learn the skill, protecting previously trained modules from interference (Brashers-Krug et al., 1995). Importantly, overlearning of a skill can strengthen its consolidation in a module, increasing resistance to interference, as is observed in humans (Brashers-Krug et al., 1996; Shadmehr and Holcomb, 1997). While such modular models can exhibit robust savings and appropriately limited forms of interference, we question the biological plausibility of a reserve of untrained neural modules awaiting assignment when a new skill is to be learned.

Modular approaches of this kind should be distinguished, from the hypothesis that the hippocampus and the neocortex form distinct learning systems (McClelland et al., 1995). This hypothesis asserts that catastrophic interference is alleviated through the use of a fast hippocampal learning system that uses sparse representations. While neocortical systems are assumed to use a less sparse representations, making them more vulnerable to interference, problems are avoided through a hippocampally mediated process of consolidation, where neocortical networks receive interleaved “virtual” practice in multiple skills. In addition to explicit hippocampal models, this strategy has also been embodied in *pseudo-pattern* models, in which savings is facilitated by a process of knowledge transfer between multiple separate networks (Robins, 1995). While we see this approach as extremely promising, there is evidence that humans can continue to learn new motor skills even after complete removal of the hippocampus (Jenkins et al., 1994). From our perspective, this suggests that neocortical representations may be sufficiently sparse to support savings in motor skills. Thus, we report the results of simulations exploring the effects on savings of varying sparsity of representation. We have also tested the ability of Leabra’s learning rule to account for overlearning effects without recourse to a separate memory consolidation mechanism.

While sparsity may play an important role in savings, other neurocomputational mechanisms may also contribute. It is possible that synaptic changes during the learning of an interfering skill may drive certain neurons associated with a previously learned skill below their firing threshold — but just below — allowing them to recover quickly once practice of the previous skill is resumed. This is exactly the mechanism posited for savings after extinction in a biophysically detailed model of the role of the cerebellum in eye blink conditioning (Medina et al., 2001). Savings through subthreshold responding is consistent with the Leabra learning rule, and it will be the focus of future analysis.

Lastly, it is worth noting that savings might be facilitated if the multiple skills to be learned share some common structure, such as a shared sub-task. In this case, training in a skill may reinforce components of a previously learned skill. Artificial neural networks trained in an interleaved manner to produce multiple motor sequences have been found to generate internal representations that reflect common sub-sequences, allowing knowledge of those sub-tasks to be generalized across tasks (Botvinick and Plaut, 2004). We have found similar generalization of sub-tasks when skills are learned sequentially by a Leabra network, and we have found that this has a positive effect on savings. Thus, Leabra’s support for sparse representations does not prevent neuron sharing across skills when doing so is appropriate.

Methods

The Tasks

We have performed simulation experiments involving the learning of motion trajectories of a three joint planar arm

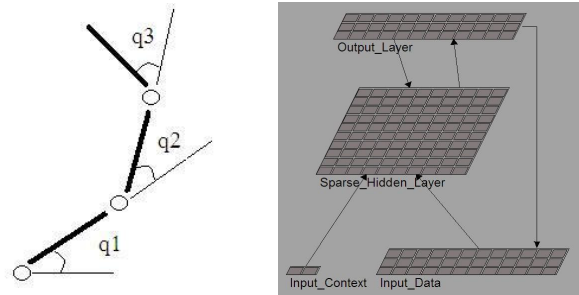


Figure 1: Left: A three joint planar arm. The state of the arm at any point in time is given by the vector of joint angles: (q_1, q_2, q_3) . Right: The Leabra network.

Task 2	Nothing in common with Task 1.
Task 3	Joint 1 matches Task 1 at all steps.
Task 4	Joints 1 & 2 match Task 1 at all steps.
Task 5	For steps 6–10, all joints match Task 1.
Task 6	For steps 6–15, all joints match Task 1.

Table 1: Similarities between various tasks as compared to Task 1, over the 18 time steps that make up each task.

by a Leabra network (Figure 1). The state of the arm at any point in time is represented by the three joint angles. The position of a joint can range from 0° to 90° . Six different motion trajectories were used in our simulations: Task 1 to Task 6. Each trajectory is discretized into 18 time steps. Thus, motion trajectories are represented as a sequence of arm states at successive points in time. Each of the six trajectories were non-Markovian with regard to individual joint angles but were Markovian with regard to the complete state of the arm. In other words, it is not possible to reliably predict the future position of a joint given only its current position, but the set of three joint angles is always sufficient to predict the arm configuration at the next time step.

Task 1 was used as the primary task in all the simulations. The network was always trained on this task first. Then, the network was trained on one of the other tasks. Finally, we measured the extent to which the network remembered Task 1. Table 1 describes the similarities between tasks.

Each of the joint angles was encoded in the Leabra network over a pool of 12 neural units. Each of the 12 units had a preferred angle, ranging from -10° to 100° in 10° increments. If the angle to be encoded was a multiple of ten, the corresponding unit, as well as its two neighbors, were set to their maximal firing rates. Otherwise, the two units with preferred angles that straddle the angle to be encoded were set to fire maximally, and their neighbors were set to an intermediate activation level. Similarly, patterns of activity over the 12 units were decoded by locating the three or four adjacent units that were all active and computing the weighted sum of the preferred angles of those units, weighted by their activity (i.e., normalized firing rate). Other patterns of

activity were considered ill-formed. With each joint angle encoded over 12 units in this way, the complete arm configuration could be encoded over a layer of 36 units.

The Network

Figure 1 shows the Leabra network used in our simulations. On each time step, the network was provided with a 36 unit input that encodes the current state of the arm. Complete interconnections from this input layer to a hidden layer produced an internal representation of the current arm state, with the sparsity of this representation controlled by lateral inhibition within the hidden layer. Complete bidirectional excitatory connections map this internal representation to an output layer that is intended to encode the next arm state in the current trajectory. Lateral inhibition in the output layer was set to encourage well-formed angle codes (i.e., approximately 9 units highly active out of the 36). During training, the output layer was also provided with a target signal, indicating the correct arm configuration for the next time step. The arrow on the right side of Figure 1 indicates that the output on a given time step became the network’s input on the subsequent time step, matching other recurrent network architectures (Jordan, 1986). The context layer contained two units, each corresponding to one of the two learned tasks, indicating which trajectory was to be produced by the network. This context information was not initially included in our simulations and is described later.

Most of the parameters used in our simulations were Leabra default values. Hebbian learning was strengthened in our simulations, contributing to 1% of synaptic weight changes rather than the default 0.1%. An error tolerance of 0.25 was used, treating outputs within 0.25 normalized firing rate of their targets as correct. A small amount of activation noise was also added to the input layer, sampled uniformly from $[-0.05, +0.05]$.

There are two common measures of savings: *exact recognition* and *relearning* (French, 1999). The exact recognition measure assesses the percentage of the original task that the network performs correctly after it has learned a second task. The relearning measure examines how long it takes the network to relearn the original task. The two measures are usually correlated. We used an exact recognition measure to assess savings. In particular, we measured the sum-squared error (SSE) of the network output on the first task after training the second task. In order to contrast this SSE value with “complete forgetting” of the first task, the SSE was also recorded prior to the first task training, and we report the ratio of SSE after interference training to SSE of the untrained network. A value of one or more for this ratio indicates complete forgetting of the initial task, while lower values indicate savings. We repeated each experimental condition five times in order to deal with stochastic variations in our simulations. We report the average of these repetitions. For comparison, we have also reported the results of running all the experiments on a traditional backpropagation (BP) artificial neural network.

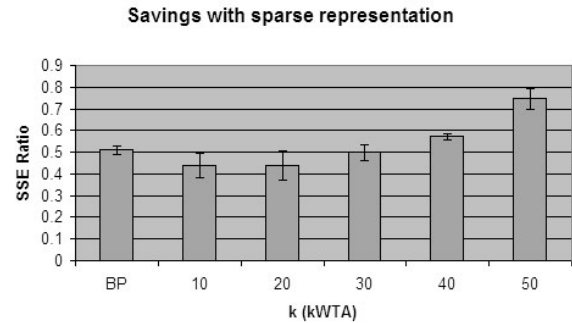


Figure 2: Savings as a function of sparsity. An SSE ratio of one or more indicates no savings, while lower values indicate retention of Task 1 knowledge. The k parameter roughly equals the percentage of active units in the hidden layer. Error bars display standard errors of the mean.

Results

Sparse Representations

In this set of experiments, we explored the contribution of sparse representations to savings. For this set of experiments, the size of the hidden layer was set to 100 units, but the amount of lateral inhibition was varied. Tasks were trained until a zero SSE (within error tolerance) was achieved for three successive trajectory executions. Using Task 2 as the second task, the SSE Ratio as a function of the hidden layer kWTA parameter is shown in Figure 2.

Savings were greater (lower SSE ratio) when sparser representations were used (lower k value). The likely reason for this effect is a decrease in the overlap between Task 1 and Task 2 hidden layer activation patterns as representations become more sparse. To test this hypothesis, we counted the number of hidden layer units that were active (at least 0.05 activation) during one task but *not* during the other. This number of discriminating units was high for sparse representations (e.g., about 30 for $k = 10$) and very low for dense representations (e.g., about 2 for $k = 50$). Thus, increasing inhibition produced more distinct internal representations between the tasks and resulted in improved savings. The BP network performed worse than the sparse Leabra network. This was as expected, since there was no explicit mechanism to facilitate non-overlapping hidden layer representation in the BP network.

We also manipulated sparsity by fixing the number of active units in the hidden layer to 10 while varying the absolute number of units in the layer to 25, 100 and 1000. Once again, savings, as measured by the SSE ratio, increased substantially with sparsity in the Leabra network. With a layer size of 25 units, the SSE ratio was about 0.6, but it dropped to less than 0.2 with 1000 hidden units. The SSE ratios for the BP network dropped from 0.63 to 0.28 as hidden layer size increased.

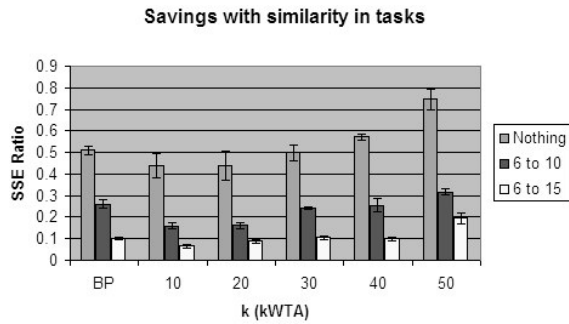


Figure 3: Savings as a function of sparsity and task similarity. The three shaded bars correspond to the use of Task 2, Task 5, and Task 6, respectively, for the interfering task. Low SSE ratio values indicate increased savings. Error bars display standard errors of the mean.

Generalization Due To Shared Sub-Tasks

We also explored the effects of similarity between the initial task and the interfering second task. In particular, we considered second tasks that shared a common sub-task with Task 1, assessing the contribution of this common task component to savings. We examined three different second tasks — Task 2, Task 5, and Task 6 — which varied in the number of time steps for which their trajectories exactly matched that of Task 1. We expected shared task components to improve savings, as the shared sub-task would be reinforced by training on the second task. Using a network with 100 hidden units produced the confirmatory results shown in Figure 3.

We also counted hidden units whose activity discriminated between the tasks, and found that the number of such units dropped substantially during the time steps corresponding to shared motion between the tasks. For the maximally sparse networks, the number of discriminating units fell from as many as 27 units during time steps involving differing motion to as few as 2 units during shared sub-tasks. Thus, the same neural units were used to encode shared sub-tasks, even when tasks were learned sequentially.

Next, we examined the case in which the first and second tasks share common motion for only a subset of the joints. This is another way in which two tasks might be seen as sharing a common sub-task. We compared savings when the interfering second task was Task 2 (nothing in common), Task 3 (joint 1 in common), or Task 4 (joints 1 and 2 in common). We were surprised to find that there were no reliable effects of this form of task overlap (Figure 4). It is interesting to note that since our tasks are non-Markovian with regard to individual joint angles, the network is forced to integrate information about all joint angles in order to produce correct output. This could be the reason for the lack of savings in this case. The control of a joint having common motion across the two tasks had to be learned differently for the two tasks, because its motion depended on the position of other joints in different ways for the two tasks.

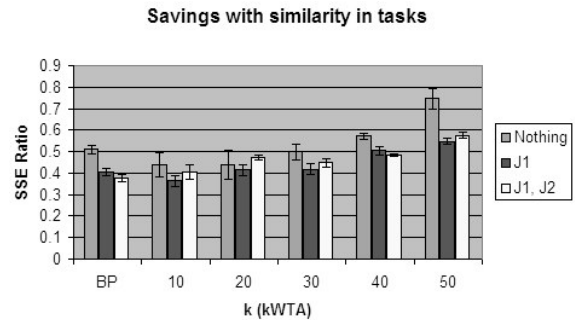


Figure 4: Savings as a function of sparsity and shared joint motion. The three shaded bars correspond to the use of Task 2, Task 3, and Task 4, respectively, for the interfering task. Low SSE ratio values indicate increased savings. Error bars display standard errors of the mean.

Contextual Cues

In the simulations presented so far, the network received no information about the appropriate trajectory to produce except for the current position of the arm. In most real-world situations, however, there are distinct sensory or internal control cues that are associated with different skills. The presence of such cues may not only assist in the selection of a known skill, but they may help shape internal representations so as to separate the representations for different skills. This could improve savings. In order to investigate this possibility, we included a two-unit context layer (Figure 1). One unit in this layer was active for each of the two tasks that were learned. These two units were randomly connected to the units in the hidden layer, with an 80% probability of any particular connection being formed. The magnitudes of the synaptic weights were determined by standard Leabra learning mechanisms. The use of this contextual cue greatly increased savings, though savings remained sensitive to sparsity. The results of using Task 2 as the interfering task are shown in Figure 5. Analysis of hidden layer activation patterns found many more units whose activity discriminated between the tasks when the cue was present. For $k = 10$, the number of discriminating units rose from about 30 to over 60 when a contextual cue was incorporated. The BP network showed no improvement in savings due to the incorporation of a contextual cue.

Overlearning

Humans display increased savings in motor skills when the initial skill is overlearned (Shadmehr and Holcomb, 1997). In order to assess if this effect is captured by Leabra’s biologically-based learning rule, we performed a set of experiments in which training time on the two tasks was varied. Specifically, we varied the number of consecutive task executions that had to be performed by the network with zero SSE (within error tolerance) in order for the task to be considered mastered. In order to simulate overlearning, this number of successes was increased to 10. We also examined “weak learning” by re-

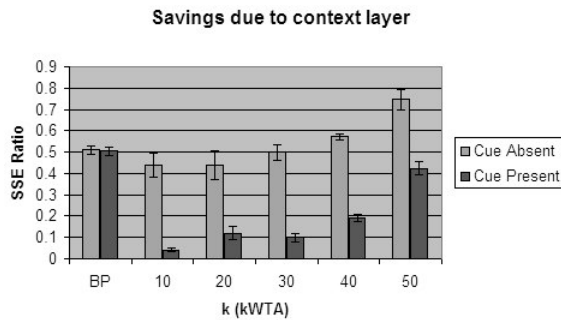


Figure 5: Savings as a function of sparsity and inclusion of a contextual cue. Low SSE ratio values indicate increased savings. Error bars display standard errors of the mean.

quiring only one successful execution. We expected that Leabra’s Hebbian learning mechanism would strengthen synaptic weights during the overlearning period, making them more difficult to perturb during the learning of the second task. Using Task 2 as the interfering task and $k = 10$, we found that savings improved in the Leabra network when both tasks were overlearned, and it improved even more substantially when the first task was overlearned and the second was “weakly learned”. Thus, the effect of overlearning on savings falls out of Leabra’s learning mechanism. As expected, overlearning did not improve savings in a BP network.

Conclusion

We have shown that the neurocomputational principles embodied by the Leabra modeling framework are sufficient to exhibit substantial savings in the sequential learning of temporally-extended motor skills. No auxiliary computational mechanisms are needed in order to avoid catastrophic interference. Savings was found to be sensitive to the amount of lateral inhibition in internal network layers, with sparser representations encouraging skill savings. Interestingly, our data actually show noteworthy savings even for internal representations that aren’t very sparse, suggesting that some amount of motor skill savings may be directly supported by dense representations in neocortex. We found generalization to sub-sequences of motor actions, but not to individual joint motions, but this has been in the context of tasks that require a tight interdependence between joints. It is likely that a similar lack of generalization would be seen in humans who are learning skills that involve many synchronized component motions, like swimming or typing. Contextual cues were found to greatly benefit savings in Leabra. Also, the general pattern of overlearning effects observed in humans were reproduced.

We have focused on an error ratio measure of savings in this work, but retraining time in Leabra would also be interesting to assess. Initial simulation experiments have found savings in the form of reduced retraining times, but we have found this measure to be insensitive to the sparsity of internal representations unless

training is made more stringent by reducing the error tolerance. Future simulation experiments will focus on understanding the relationship between retraining time and lateral inhibition in Leabra, with the goal of collecting additional evidence concerning the suitability of Leabra’s biologically-based modeling framework for explaining skill acquisition and skill savings.

References

- Botvinick, M. and Plaut, D. C. (2004). Doing without schema hierarchies: A recurrent connectionist approach to normal and impaired routine sequential action. *Psychological Review*, 111(2):395–429.
- Brashers-Krug, T., Shadmehr, R., and Bizzi, E. (1996). Consolidation in human motor memory. *Nature*, 382:252–255.
- Brashers-Krug, T., Shadmehr, R., and Todorov, E. (1995). Catastrophic interference in human motor learning. In *Advances in Neural Information Processing Systems*, volume 7, pages 19–26.
- French, R. M. (1994). Dynamically constraining connectionist networks to produce distributed, orthogonal representation to reduce catastrophic interference. In *Proceedings of the Fifteenth Annual Conference of the Cognitive Science Society*, pages 335–340.
- French, R. M. (1999). Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, 3(4):128–135.
- Jenkins, I. H., Brooks, D. J., Nixon, P. D., Frackowiak, R. S. J., and Passingham, R. E. (1994). Motor sequence learning: A study with positron emission tomography. *The Journal of Neuroscience*, 14(6):3775–3790.
- Jordan, M. I. (1986). Attractor dynamics and parallelism in a connectionist sequential machine. *Proceedings of the fifteenth Annual Conference of the Cognitive Science Society*, pages 531–546.
- McClelland, J. L., McNaughton, B. L., and O’Reilly, R. C. (1995). Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory. *Psychological Review*, 102(3):419–457.
- McCloskey, M. and Cohen, N. J. (1989). Catastrophic interference in connectionist networks: The sequential learning problem. In Bower, G. H., editor, *The Psychology of Learning and Motivation*, volume 24, pages 109–164. Academic Press, New York.
- Medina, J. F., Garcia, K. S., and Mauk, M. D. (2001). A mechanism for savings in the cerebellum. *The Journal of Neuroscience*, 21(11):4081–4089.
- O’Reilly, R. C. and Munakata, Y. (2000). *Computational Explorations in Cognitive Neuroscience*. MIT Press.
- Robins, A. (1995). Catastrophic forgetting, rehearsal, and pseudorehearsal. *Connection Science*, 7:123–146.
- Shadmehr, R. and Holcomb, H. H. (1997). Neural correlates of motor memory consolidation. *Science*, 277:821–825.