LETTER

Connectionist Selectionism: A Case Study of Parity

Rio B. T. Lowry and Michael R. W. Dawson

Department of Psychology, University of Alberta
Edmonton, Alberta, Canada T6G 2E9
E-mail: mdawson@ualberta.ca

(Submitted on August 31, 2005)

Abstract—There is a general view that instructionist and selectionist theories of adapting to an environment are mutually incompatible [1-4]. Below, we report the results of a series of computer simulations that demonstrate that this is not necessarily the case. Our simulations show that the main ideas of selectionism can be incorporated into a standard instructionist framework, which can benefit both perspectives.

Keywords— Instructionism, Selectionism, PDP networks, Parity, Networks of value units

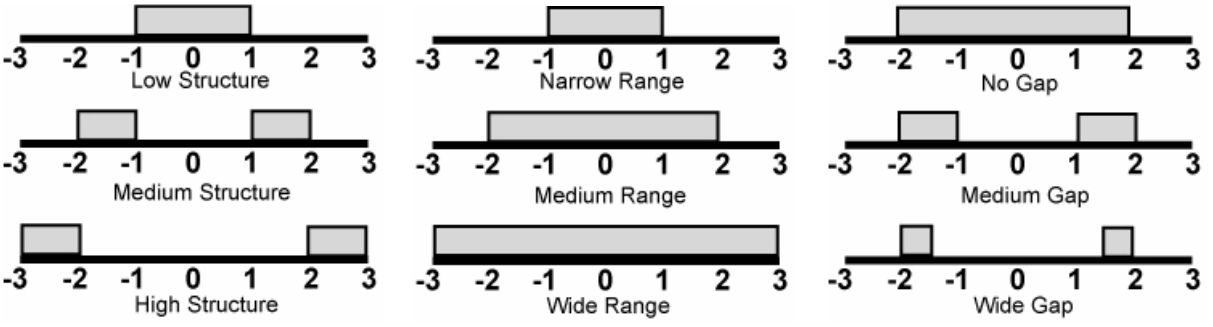
1. Introduction

Instructionist or epigenetic theories view cognition as the ultimate product of neuronal growth [5]. In its most extreme form, the developing brain is viewed as initially being a *tabula rasa*. As the result of interactions with an environment, neural structure emerges via the growth and/or strengthening of neurons and synapses. Parallel distributed processing (PDP) models represent one influential variant of instructionism.

Instructionist theories have an advantage of being highly formalized which has allowed them to be explored in detail using computer simulation methods, and to be linked to well established theories of pattern recognition and machine learning [6]. But this formalization may have been purchased at the expense of their biological relevance. Many neuroscientists have raised serious questions about the neural plausibility of instructionist theories like PDP networks [7]. Some have argued that PDP networks are as functionalist in nature as the symbol-based cognitive science theories that they have been reacting against [8-10].

In contrast to instructionist theories, selectionist theories of cognition deny the extreme epigenetic claim that the brain is a structureless *tabula rasa*. Instead they assume that the initial stages of brain development involve the generation of a large and varied amount of structure. This structure provides a preexisting repertoire of responses to be elicited by the environment. The interaction between the environment and preexisting structure selects some responses as being more appropriate than others, which modifies the underlying neural architecture. "After initial selection, certain cell groups in the repertoire have a higher probability than others of being selected by a similar or identical signal pattern" [11]. This change in the probability of a response being elicited can either be created by a positive process (e.g., an increase in the population of neural circuits that have been selected by environmental signals) or by a negative process (e.g., a decrease in the population of the neural circuits that have not been selected by environmental signals). In short, selectionist theories provide a "use it or lose it" perspective on brain structure.

Selectionist theories maintain a high degree of biological plausibility. For instance, they attempt to be extremely consistent with measurements of neural development. Several researchers have observed that in the first year of human life there is a dramatic increase in both the number of neurons and in synaptic density, but that this is followed by a longer period of time in which both of these factors demonstrate substantial declines [12, 13]. This is predicted by selectionist theories in which early neuronal growth provides a large repertoire of neural circuits that is later pruned by environmental exposure. However, the strong biological nature of selectionist theories has also worked against their formalization. While computer simulations have been used to study some selectionist predictions [14], they have not successfully modeled some of the higher-order phenomena that the more functional PDP models have been used to study. As a result, selectionist theories have not had a strong impact on cognitive science in general. ”The crucial issue remains to find a learning rule coherent with such a Darwinian picture” [15].



(a) (b) (c)

Figure 1. The sampling distribution used to randomly initialize weights in (a) Simulation 1, (b) Simulation 2, and (c) Simulation 3. The grey boxes indicate ranges from which weights could be randomly sampled.

Our hypothesis was that the learning rule sought by selectionist researchers might in fact be the kind of rule that has already been established in instructionist models. Specifically, there is no reason in principle why procedures used to train PDP models, such as the generalized delta rule [16], cannot be used in a selectionist paradigm. We explore this possibility in the simulations reported below.

2. Simulation 1: Increasing Structure and Hidden Unit Numbers

Consider what would have to be done in order for a PDP learning rule to alter a network in accordance with selectionist assumptions. If a) initial connection weights were randomly selected in such a fashion that they were much more structured than in typical PDP practice, and b) if many more hidden units were used than would ordinarily be required in a PDP network, then it might be possible to use a rule like the generalized delta rule to select useful, preexisting processing units from a pre-structured network.

One of our independent variables concerned the distribution from which connection weights were randomly sampled prior to the training of the network. This manipulation was used to insert initial structure into the PDP networks prior to training. In the control condition, all of the weights were initialized by randomly sampling values from a rectangular distribution that ranged from -1 to +1. In the experimental conditions, structure was added to initial weights by changing the variability (but not the mean) of this distribution. This was accomplished by inserting a "gap" in the distribution from which weights were randomly initialized, as is detailed below (see also Figure 1(a)). The rationale underlying this manipulation was that structure would be supplied to the network by ensuring that all weights began at values that were much more extreme than in the control condition. Indeed, this is the rational behind the VARIMAX method for rotating factors into an orientation that produces simple structure: the higher the variance, the greater the likelihood that cases cluster into distinct, structured, groups [17].

A second independent variable was the number of hidden units in the network. In one condition, there were as many hidden units as there were input units. In a second condition, there were twice as many hidden units as there were input units. In a third condition, there were three times as many hidden units as there were input units. As the number of hidden units is increased, so does the potential number of different internal responses to stimulus patterns. This is particularly true when this manipulation is combined with a condition in which the initial connection weights are highly structured.

Our hypothesis was that in networks in which initial connection weights were highly structured, and in which there was also a large number of preexisting hidden units, the application of the generalized delta rule would essentially serve as a selectionist mechanism. In other words, rather than "growing" a network for solving the task -- which is the selectionist view of PDP modeling -- the learning rule would select the appropriate hidden units from the large number that were available. One consequence of this should be a dramatic increase in learning speed. However, this should only occur under the appropriate combination of the two independent variables – high structure and a large number of hidden units. Our first simulations attempted to determine whether this interaction between independent variables would appear.

The first simulation was designed to test whether the selectionist approach to PDP networks would provide any benefits for the learning of a particularly difficult pattern recognition problem, the parity problem. In the

parity problem, a network has a single output unit, and it has N input units. Each input unit is a bit that can either be on or off. The network is trained to detect when an odd number of its input bits are active. If this is the case, then the network is to turn its output unit on. Otherwise, the network is to turn its output unit off. The parity problem is an extremely difficult benchmark for a PDP network [11]. This is because patterns that neighbor one another in the pattern space require the network to make opposite responses. We were interested in whether the performance of a network on this difficult problem could be improved by training it from a selectionist perspective. In other words, we hypothesized that a combination of "high" structure in the starting state of the network and a large number of hidden units would lead to fast learning of the parity problem, even in cases when N was large.

Network Architecture. Each network had one output unit, which was trained to activate when an odd parity problem was presented to the input units, and to fail to activate when an even parity problem was presented to the input units. The output unit was a value unit [18]. Such a unit is trained with a variant of the generalized delta rule, because it employs a Gaussian activation function ($G(net_i) = \exp(-\pi(net_i - \mu_i)^2)$, where net_i is the net input to unit i, and μ_I is the mean of the Gaussian, which is similar to the bias of a standard unit).

To train networks of such units, the standard error term for output units that is used in the generalized delta rule is changed to: $E = \frac{1}{2} \Sigma\Sigma(t_{pj} - a_{pj})^2 + \frac{1}{2} \Sigma\Sigma\ t_{pj}\ (net_{pj} - \mu_j)^2$, where E is error, t_{pj} is the desired value for unit j presented pattern p, a_{pj} is the activity, and net_{pj} is the net input, of unit j when presented pattern p, and μ_j is the mean of the unit's Gaussian. With this change, the algorithm for training the network is exactly the same as the generalized delta rule (with the exception that the derivative of the Gaussian is used in the new equations). Dawson has provided details about the mathematical relationships between the two rules, and software for training networks of value units [9, 20]. Networks of value units were selected for this study because we have considerable experience in how they behave (e.g., the many examples in [9] and [20]). This experience was useful in exploring whether the behavior of the networks was affected by the selectionist manipulations that we were exploring.

Three different versions of the parity problem were examined. In the 5-parity problem, the network had 5 input units, and the training set consisted of all of the 32 binary patterns that could be represented by these units. In the 7-parity problem the network had 7 input units and a training set of 128 possible binary inputs. In the 9-parity problem, the network had 9 input units and a training set of 512 possible binary inputs. For each network, the "off" state of an input unit was represented with the value 0, and the "on" state of an input unit was represented with the value 1.

Hidden Unit Manipulation. For each version of the parity problem, three different sizes of networks were trained. One had the same number of hidden units as there were input units (i.e., N hidden units, where N is the number of input units). A second had twice as many hidden units as there were input units (i.e., 2N hidden units). A third had three times as many hidden units as there were input units (i.e., 3N hidden units). In all simulations, every hidden unit was also a value unit.

Structure Manipulation. For each network trained on a parity problem, three different starting conditions were examined (Figure 1(a)). The first was a "low structure" condition. In this condition, all of the connection weights in the network were initialized by randomly sampling from the range -1 to +1. As this is typical practice, this represented a control condition. The second was a "medium structure" condition. In this condition, all of the connection weights were initialized by randomly sampling from the range -2 to -1 and 1 to 2, but not from the range between -1 and 1. In other words, a "gap" was inserted into the sampling distribution for weight initialization. The third was a "high structure" condition. In this condition, all of the connection weights were initialized by randomly sampling from the range -3 to -2 and 2 to 3, but not from the range between -2 and 2. This had higher structure than the previous condition because the weights were more extreme (leading to higher variance, which can be equated with higher structure [17], and there was a larger gap in the sampling distribution. In all three of these conditions, the bias of each processing unit was initialized with a value of 0.

With this structure manipulation, the mean of the sampling distribution was held constant, but the variance of the distribution was increased. In general, we increased structure by changing the sampling distribution to make the initial network weights more extreme.

Experimental Design. This simulation can be described as a 3 X 3 X 3 factorial design. The first factor was size of problem (5-parity, 7-parity, and 9-parity). The second factor was the number of hidden units (N, 2N, and 3N). The third factor was the structure in the sampling distribution used to initialize connection weights (low

Table 1. Mean epochs to convergence (and standard deviations) for the various conditions of Simulation 1. Rows correspond to different levels of the "gap" manipulation of structure, while columns correspond to different levels of the number of hidden units (expressed as multiples of the number of input units).

	5 Parity			7 Parity			9 Parity		
	N	**2N**	**3N**	**N**	**2N**	**3N**	**N**	**2N**	**3N**
Low Structure	928.8 (371.1)	234.4 (103.1)	123.6 (25.1)	4633.1 (3997.6)	460.5 (161.3)	225.5 (51.6)	7176.9 (3841.2)	3791.6 (3241.7)	1977.8 (1413.6)
Medium Structure	2065 (4182.4)	57.1 (67.5)	15 (10.3)	9017.9 (3105.7)	63 (32.4)	46.5 (33.1)	10000 (0.0)	1158.0 (3108.9)	30.5 (8.5)
High Structure	4015.7 (5150.5)	1032 (3152.3)	2.5 (2.3)	9005.7 (3144.3)	8.1 (7.3)	2.4 (0.8)	9039.3 (3038.0)	2017.2 (4207.3)	28.5 (29.2)

structure, medium structure, and high structure). In this design, there are 27 different cells. In each cell, 10 different networks were trained, each randomly initialized in accordance with the constraints imposed by the structure manipulation. Each of these different networks (270 in total) represented a different "subject" in the experiment. The dependent measure for the study was the number of training epochs required for a network to solve the parity problem.

Network Training. Each network was trained with a variant of the traditional generalized delta rule [16] that was developed for networks of value units by Dawson and Schopflocher [18]. Network connections were updated after every pattern presentation. One epoch involved the presentation of every possible input pattern to the network. The order of pattern presentation was randomized every epoch. Networks were said to have converged on a solution to the problem when a "hit" was recorded for the output unit for every pattern presented during the epoch. A "hit" was defined as output unit activity of 0.9 or greater when the desired output was 1.0, or as output unit activity of 0.1 or less when the desired output was 0.0. If convergence was not achieved after 10,000 epochs, then training was stopped, and the value of 10,000 was entered as the value for the dependent measure.

Results. As can be seen from Table 1, for each version of the parity problem there appears to be an interaction between the number of hidden units and the amount of initial structure in the network. In general, when there were many hidden units and high degrees of initial structure, fast convergences were observed. However, decreases in either the number of hidden units or in the amount of initial structure resulted in networks that had difficulty in learning the solution to the problem. Indeed, in many cells of the experiment all 10 "subjects" failed to converge upon a solution to the problem after 10,000 training epochs.

In order to examine the pattern of results in Table 1, analysis of variance (ANOVA) was conducted. The ANOVA involved three independent factors (input units, structure, hidden units) that each had three levels. The ANOVA revealed a significant main effect of the number of input units ($F_{2,243} = 35.118$, $p < 0.0001$). An inspection of Table 1 indicates that this effect is due to the fact that in general a parity problem involving fewer input units is easier to solve than is one involving a larger number of input units. There was also a significant main effect of the number of hidden units ($F_{2,243} = 166.437$, $p < 0.0001$). The results in Table 1 suggest, not surprisingly, that increasing the number of hidden units on average leads to faster learning of solutions to the parity problem. There was also a significant interaction between the number of input units and the number of hidden units ($F_{2,243} = 35.118$, $p < 0.0001$). An inspection of Table 1 indicates that the effect of increasing the number of hidden units had a greater magnitude for larger versions of the parity problem than it did for smaller versions of this problem. Importantly, there was also a significant interaction between the number of hidden units and the structure manipulation ($F_{4,243} = 8.186$, $p < 0.0001$). Table 1 indicates that this interaction is exactly of the type to be expected if the selectionist hypothesis guiding the current study was correct: there was a dramatic speeding of learning in those conditions that had both a high number of hidden units and a high degree of structure. However, if structure was high, but the number of hidden units was smaller, the result was slower learning than in conditions with both low structure and a low number of hidden units. No other significant effects were revealed in the ANOVA.

Table 2. Mean epochs to converge (and standard deviations) for the various conditions of Simulation 2. Rows correspond to different widths of the distribution from which weights were initialized, while columns correspond to different levels of the number of hidden units (expressed as multiples of the number of input units).

	5 Parity			7 Parity			9 Parity		
	N	**2N**	**3N**	**N**	**2N**	**3N**	**N**	**2N**	**3N**
Narrow Range	928.8 (371.1)	234.4 (103.1)	123.6 (25.1)	4633.1 (3997.6)	460.5 (161.3)	225.5 (51.6)	7176.9 (3841.2)	3791.6 (3241.7)	1977.8 (1413.6)
Medium Range	3515.2 (4483.1)	430.7 (441.2)	272.3 (191.1)	10000 (0.0)	2147.2 (2819.5)	367.9 (186.0)	10000 (0.0)	10000 (0.0)	8604.9 (1994.6)
Wide Range	10000 (0.0)	1063.4 (1546.5)	196.4 (125.4)	10000 (0.0)	6950.1 (3969.7)	1034.4 (923.3)	10000 (0.0)	10000 (0.0)	10000 (0.0)

3. Simulation 2: Manipulating the Width of the Sampling Distribution

The results of Simulation 1 were consistent with the hypothesis that selectionist approaches to training PDP networks can lead to improved learning. The significant interaction between the structure manipulation and the number of hidden units occurred because the fastest learning conditions were those in which high structure was combined with a large number of hidden units. However, alternative accounts are possible, and need to be explored. In Simulation 1, the creation of the different structure conditions also produces a confound: not only is a gap introduced into the distribution from which weights are initialized, but the range of the distribution is also changed. It is possible that the effects observed in Simulation 1 were simply due to the greater range of weights, and not due to the presence of any gap in the distribution at all. Simulation 2 tested this possibility.

Method. The method for Simulation 2 was identical to the method for Simulation 1, with one key exception: the structure manipulation of Simulation 1 was replaced with a manipulation of the range of the sampling distribution (Figure 1(b)). In the "narrow range" condition, the connection weights were sampled from the range -1 to 1. In the "medium range" condition, the connection weights were sampled from the range -2 to 2. In the "wide range" condition, the connection weights were sampled from the range -3 to 3. In other words, these conditions were the same as those in Simulation 1 with the key exception that there was no gap inserted in any of the sampling distributions.

Results. The results of Simulation 2 are presented in Table 2 below. An inspection of these results indicates that they are quite different than those of Simulation 1. In particular, it is the narrow range of weights that appears to provide the fastest learning, which was not evident in Table 1. Furthermore, the average number of epochs in the fastest cells in Table 2 is at least one order of magnitude slower than the fastest cells in Table 1. Indeed, for the 9-parity problem convergences were never achieved in the wide range condition, regardless of the number of hidden units.

Statistical analysis of the data confirms the conclusions drawn from an inspection of Table 2. The fact that the smaller parity problems were easier to solve than the larger ones was reflected in a main effect of the number of input units ($F_{2,243} = 244.982$, $p < 0.0001$). An increase in the number of hidden units also tended to speed up learning, and a significant main effect of the number of hidden units was present ($F_{2,243} = 158.907$, $p < 0.0001$). On average, increasing the width of the sampling distribution for the weights also increased learning speed ($F_{2,243} = 128.458$, $p < 0.0001$). However, there were significant two-way and three-way interactions amongst all of these variables. For instance, increasing the number of hidden units and the width of the sampling distribution increased learning speed for the 5-bit and 7-bit parity problems, but slowed learning down for the 9-bit parity problem, producing a significant interaction between the number of input units, the number of hidden units, and the width of the sampling distribution ($F_{8,243} = 15.633$, $p < 0.0001$). There were also significant interactions between the number of input units and the number of hidden units ($F_{4,243} = 18.680$, $p < 0.0001$), the number of input units and the width of the sampling distribution ($F_{4,243} = 10.012$, $p < 0.0001$), and between the number of hidden units and the width of the sampling distribution ($F_{4,243} = 4.272$, $p < 0.002$).

These results point out two things. First, the significant interactions between all the manipulations in Simulation 2 indicate that solutions to the parity problem can be influenced by the size of the problem, the number of hidden units, and the width of the sampling distribution from which weights are initially randomized. Second, these influences are quite different than was the case in Simulation 1 in which a gap was inserted into

Table 3. Mean epochs to convergence (and standard deviations) for the various conditions of Simulation 3. Rows correspond to different levels of the width of the gap in the sampling distribution used to initialize weights, while columns correspond to different levels of the number of hidden units (expressed as multiples of the number of input units).

	5 Parity			7 Parity			9 Parity		
	N	**2N**	**3N**	**N**	**2N**	**3N**	**N**	**2N**	**3N**
No Gap	3515.2 (4483.1)	430.7 (441.2)	272.3 (191.1)	10000 (0.0)	2147.2 (2819.5)	367.9 (186.0)	10000 (0.0)	10000 (0.0)	8604.9 (1994.6)
Medium Gap	2566.25 (4182.4)	57.1 (67.5)	15.0 (10.3)	7150.8 (4595.8)	63.0 (32.3)	46.5 (33.1)	10000 (0.0)	1158 (3108.9)	30.5 (8.5)
Wide Gap	5016.1 (5261.1)	2012.2 (4209.9)	6.3 (6.5)	8009.0 (4197.4)	1017.4 (3156.2)	5.3 (2.8)	7039.7 (4766.7)	26.3 (22.4)	13.5 (9.3)

the sampling distribution. In other words, the presence of the gap in Simulation was a crucial factor in speeding up learning. The results of Simulation 1 were not simply due to increasing the range from which weights could be sampled.

4. Simulation 3: Manipulating Gap Size in the Sampling Distribution

The results of Simulation 2 indicated that the effects observed in Simulation 1 were not just due to an increase in the range of the distribution from which weights were sampled, but were also due to the presence of a gap in this distribution. A second confound in Simulation 1 was that when the width of the sampling distribution was increased, so too was the width of the gap. Simulation 3 explored the behaviors of networks when the width of the gap was manipulated, but the range of the sampling distribution was held constant.

Method. The method for Simulation 3 was identical to the method for Simulation 1, with one key exception: the structure manipulation of Simulation 1 was replaced with a manipulation of the gap in the sampling distribution. At the same time, the range of the sampling distribution was held constant (Figure 1(c)). In the "no gap" condition, the connection weights were sampled from the range -2 to 2. In the "medium gap" condition, the connection weights were sampled from the range -2 to –1 and 1 to 2. In the "wide gap" condition, the connection weights were sampled from the range –2 to –1.5 and to 1.5 to 2.

Results. The results of Simulation 3 are presented in Table 2 below. An inspection of these results indicates that they are similar in many respects to those of Simulation 1, but are not quite as regular. For all three parity problems, the fastest learning occurred in the condition with the widest gap and the largest number of hidden units, as was the case in Simulation 1. In both simulation studies, these cells correspond to the conditions that we hypothesized were most consistent with selectionism (i.e., a large amount of pre-existing internal structure). However, the gap effect was moderated by the number of hidden units in different ways for different sizes of the parity problem. For both the 5- and 7-parity problems, a medium gap with a medium number of hidden units produced faster learning than a wide gap with a medium number of hidden units. However, the reverse was true for the 9-parity problem. Clearly, when gap size is manipulated in conjunction with a manipulation of the sampling range (Simulation 1, Figure 1(a)), the results are more regular than when gap size is manipulated while holding the sampling range constant (Simulation 3, Figure 1(c)).

An ANOVA of the Simulation 3 data reflects the regularities revealed in an inspection of Table 3. There was a significant main effect of the number of input units ($F_{2,243} = 47.818$, $p < 0.0001$) due to the fact the smaller parity problems were learned faster than were large ones. A significant main effect of the number of hidden units ($F_{2,243} = 142.273$, $p < 0.0001$) indicated that faster learning was achieved by increasing the number of hidden units in the network. Increasing the width of the gap in the sampling distribution also increased learning speed, as was indicated by a significant main effect of gap size ($F_{2,243} = 31.541$, $p < 0.0001$). However, further complexities in Table 3 also resulted in many significant interactions amongst these factors. There was a significant interaction between the number of input units and the number of hidden units ($F_{4,243} = 8.565$, $p < 0.0001$), and between the number of input units and gap size ($F_{4,243} = 20.928$, $p < 0.0001$). However, the interaction between the number of hidden units and gap size did not quite reach statistical significance ($F_{4,243} =$

2.161, $p < 0.074$). Given our earlier discussion of Table 3, it should be no surprise that the three-way interaction between the number of input units, the number of hidden units, and gap size was also significant ($F_{8,243} = 5.012$, $p < 0.0001$).

5. Discussion

According to selectionism, when agents encounter an environment, they are already armed with a wide variety of pre-existing internal structures [19]. Adaptation is accomplished when environmental pressures select some of these structures over others because they are (already) better suited to solving a particular problem.

The purpose of our first simulation was to explore the possibility that an artificial neural network, trained with a standard learning rule, could benefit from this selectionist perspective. We employed two manipulations to investigate this possibility. First, we manipulated the distribution from which connection weights were initially randomized. We increased the variability of this distribution (but not its mean) by inserting gaps of various widths into it, and increasing its range (Figure 1(a)). Second, we manipulated the number of hidden units in the network. The condition in which there were a very large number of hidden units, and the connection weights of these units were initialized by randomly sampling from the distribution of highest variation, represented an operationalization of the selectionist hypothesis. The networks in this condition started with the greatest amount (in terms of numbers of hidden units) of pre-existing structure. We found that this condition uniformly produced the fastest learning for three different sizes of the parity problem. For example, this condition resulted in the 9-bit parity problem being solved incredibly quickly, after approximately 29 epochs of training. This result was two orders of magnitude faster than was observed in networks that had high structure, but a small number of hidden units, and in networks that had a high number of hidden units, but low structure. In short, this result demonstrated that the selectionist approach could be incorporated into artificial neural networks, which are usually viewed as being examples of instructionist simulations.

One possibility was that this main result was merely due to the fact that the range of connection weights was being increased. However, when the range of the sampling distribution of weights was increased without inserting a gap in the distribution (Figure 1(b)), the results were quite different. The distributional gap – our main source of structure – was essential to the results of Simulation 1. A third simulation held the range of the initial connection weights constant, and varied the size of the gap in the distribution (Figure 1(c)). Although the pattern of results varied somewhat from one version of the parity problem to another, the results were again consistent with selectionist theory – fast learning was only observed in conditions that included structure (i.e., some size of gap) combined with a large number of hidden units.

These preliminary results all point to a potential reconciliation between selectionist and instructionist theorizing. However, they also raise a number of issues that require further study.

First, the current results were all obtained with networks of value units that use a Gaussian activation function [9, 18, 20]. One question to ask is whether these results are peculiar to this architecture, or whether they can be obtained with other types of artificial neural networks as well. Second, the current results were also all obtained using versions of the parity problem. This problem was selected because of its notorious difficulty. However, it is important to determine whether similar results can be obtained for other difficult classification problems. Third, the current results were obtained with a relatively crude manipulation of internal structure – coarse manipulations of the sampling distributions used to initialize connection weights. Many other more sophisticated manipulations of initial structure need to be formulated and studied. Fourth, one of the main advantages to the use of value unit networks is the ability to interpret their internal structure [9, 21-24]. This ability needs to be exploited in order to determine the precise mechanism by which learning is accelerated in Simulation 1. Presumably, the learning rule is able to either accentuate just those hidden units that are well-suited to solving the parity problem, or to prune away the behavior of those hidden units that are not. Early studies of the value unit architecture revealed networks in which hidden units were essentially pruned, because value units can be easily manipulated to be turned off to all stimuli (e.g., by modifying a unit's bias). It is essential that the "selectionist networks" that have been described in this paper have their internal structure interpreted in order to discover whether this type of mechanism is at work in them as well. We are in the process of investigating all of these different issues.

Acknowledgment

This research was supported by NSERC and SSHRC research grants to MRWD.

References

[1] J.-P. Changeux, *Neuronal Man*. Princeton, NJ: Princeton University Press, 1985.

[2] G. M. Edelman, *Neural Darwinism*. New York: Basic Books, 1987.

[3] M. S. Gazzaniga, *Nature's mind*. New York: Basic Books, 1992.

[4] M. Piattelli-Palmarini, "Evolution, selection and cognition: From "learning" to parameter setting in biology and in the study of language," *Cognition*, vol. 31, pp. 1-44, 1989.

[5] S. Pinker, *The Blank Slate*. New York, NY: Viking, 2002.

[6] B. D. Ripley, *Pattern Recognition And Neural Networks*. Cambridge, UK: Cambridge University Press, 1996.

[7] R. J. Douglas and K. A. C. Martin, "Opening the grey box," *Trends In Neuroscience*, vol. 14, pp. 286-293, 1991.

[8] M. R. W. Dawson, *Understanding Cognitive Science*. Oxford, UK: Blackwell, 1998.

[9] M. R. W. Dawson, *Minds And Machines : Connectionism And Psychological Modeling*. Malden, MA: Blackwell Pub., 2004.

[10] G. M. Edelman, *Bright Air, Brilliant Fire*. New York: Basic Books, 1992.

[11] G. M. Edelman and V. B. Mountcastle, *The Mindful Brain*. Cambridge, MA: MIT Press, 1978.

[12] P. Seeman, "Brain development, X - Pruning during development," *American Journal of Psychiatry*, vol. 156, pp. 168-168, 1999.

[13] P. R. Huttenlocher, "Morphometric study of human cerebral cortex development," in *Brain Development and Cognition: A Reader*, M. H. Johnson, Ed. Oxford: Blackwell, 1993.

[14] G. N. Reeke, "Selection versus instruction: Use of computer models to compare brain theories," *International Review of Neurobiology*, vol. 37, pp. 211-242, 1994.

[15] J.-P. Changeux and S. Dehaene, "Neuronal models of cognitive functions," in *Cognition and brain development: A reader*, M. H. Johnson, Ed. Oxford: Blackwell, 1993, pp. 363-402.

[16] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, pp. 533-536, 1986.

[17] H. R. Kaiser, "The VARIMAX criterion for analytic rotation in factor analysis," *Psychometrika*, vol. 23, pp. 187-200, 1958.

[18] M. R. W. Dawson and D. P. Schopflocher, "Modifying the generalized delta rule to train networks of nonmonotonic processors for pattern classification," *Connection Science*, vol. 4, pp. 19-31, 1992.

[19]N. K. Jerne, "Antibodies and learning: Selection versus instruction," in *The neurosciences: A study program*, G. C. Quarton, T. Melnechuk, and F. O. Schmitt, Eds. New York: Rockefeller University Press, 1967, pp. 200-208.

[20] M. R. W. Dawson, *Connectionism : a hands-on approach*, 1st ed. Oxford, UK ; Malden, MA: Blackwell Pub., 2005.

[21] I. S. N. Berkeley, M. R. W. Dawson, D. A. Medler, D. P. Schopflocher, and L. Hornsby, "Density plots of hidden value unit activations reveal interpretable bands," *Connection Science*, vol. 7, pp. 167-186., 1995.

[22] M. R. W. Dawson and P. M. Boechler, "An artificial neural network that uses coarse allocentric coding of direction to represent distances between locations in a metric space," *Spatial Cognition and Computation*, vol. Under editorial review, 2005.

[23] M. R. W. Dawson, D. A. Medler, and I. S. N. Berkeley, "PDP networks can provide models that are not mere implementations of classical theories," *Philosophical Psychology*, vol. 10, pp. 25-40, 1997.

[24] M. R. W. Dawson, D. A. Medler, D. B. McCaughan, L. Willson, and M. Carbonaro, "Using extra output learning to insert a symbolic theory into a connectionist network.," *Minds And Machines*, vol. 10, pp. 171-201, 2000.

Michael R.W. Dawson received his Ph.D. in psychology from the University of Western Ontario in 1986, and is currently a full professor in the Psychology Department at the University of Alberta. His research interests include pure and applied research on artificial neural networks and the relationship of this research to empirical and theoretical issues in Cognitive Science. (Homepage: http://www.bcp.psych.ualberta.ca/~mike/)

Rio B.T. Lowry is currently a fourth year undergraduate in Psychology Department at the University of Alberta. His research interests include the synthetic approach to building models and selectionist theories and models.