A Computational Analysis of Conservation

Thomas R. Shultz

McGill University

Abstract

An approach to modeling cognitive development with a generative connectionist algorithm is described and illustrated with a new model of conservation acquisition. Among the conservation phenomena captured with this model are acquisition, the problem size effect, the length bias effect, and the screening effect. The simulations suggest novel explanations for sudden jumps in conservation performance (based on new representations of conservation transformations) and for the problem size effect (based on an analog representation of number). The simulations support the correlation-learning explanation of length bias (that length correlates with number during number altering transformations). Some conservation phenomena that so far elude computational modeling attempts are also discussed along with their prospects for capture. Suggestions are made for theorizing about cognitive development as well as about conservation acquisition. A variety of classic puzzles about cognitive development are addressed in the light of this model and similar models of other aspects of cognitive development.

One of the mainstays of research on cognitive development is conservation. Conservation involves a belief in the continued quantitative equivalence of two physical quantities over a transformation that only appears to alter one of the quantities. A well known conservation problem presents the child with two identical rows of evenly spaced objects. An example of such a row is portrayed in the first line of Table 1. Once the child agrees that two such rows have the same number of objects, the experimenter transforms one of the rows, e.g., by spreading out the items thereby elongating the row, as shown in the fourth line of Table 1. Then the experimenter asks the child whether the two rows still have the same amount or whether one of them now has more. Piaget (1965) and many subsequent researchers found that young children, below about six or seven years of age, respond that one of the two rows, usually the longer row, now has more than the other. This seemed somewhat surprising given that quantities of items in the two rows were still identical. In contrast, children older than six or seven years respond that the two rows still have equal amounts, i.e., they conserve the equivalence of the two amounts over the elongating transformation.

Table 1
Example Transformations with Constant Density

Transformation	Length	Density	Row
Pre-transformation	2	2	o o o o
Add	2.5	2	o o o o o
Subtract	1.5	2	o o o
Elongate	4	1	o o o o
Compress	1.33	3	oooo

For Piaget, the transition from nonconservation to conservation on such tasks measured a major, pervasive shift from the pre-operational stage to the stage of concrete operations. Indeed, conservation became a major unifying idea in Piagetian theory. As Flavell put it, " It was an act of creative inspiration when Piaget hit upon the idea that a wide variety of cognitive areas -- number, quantity, time, etc. -- are in certain crucial respects mastered according to a common procedure: to discover what values do and do not remain invariant (are and are not conserved) in the course of any given kind of change or transformation . . . (Flavell, 1963, p. 415)."

Although most contemporary researchers might disagree with anything as pervasive as concrete operational thought, it is clear that very few developmental phenomena have generated as much continuing interest as has conservation. A library search uncovered over 1000 conservation publications, marking it as one of largest areas in psychology. This popularity is probably the result of conservation's counter-intuitiveness and robust replicability, in addition to it's theoretical importance.

In spite of these many empirical studies, the cognitive mechanisms underlying conservation development remain obscure. How do children come to distinguish those transformations that conserve number from those that alter number? One way to explore such cognitive mechanisms is with computer simulations where the details of knowledge representations, processing mechanisms, learning methods, and environmental information must be fully specified. The size and current stability of the conservation literature, signaled by the fact that its growth rate has decreased in recent years, along with its continued theoretical importance, make it an attractive target for computer simulations.

Because human cognition often appears to be rule governed, some computational psychologists have created rule-based cognitive models (J. R. Anderson, 1993; Newell, 1990). Rule-based models of conservation captured basic nonconservation and correct performance, but either did not develop (Klahr & Wallace, 1976; Siegler, 1981), or modeled a single training experiment with a considerable amount of prior domain-specific knowledge having been built in (Simon & Klahr, 1995). Because rule-based models often have an ad hoc quality and are sometimes too rigid for the variability in human cognition, a number of developmental researchers have turned to neural network techniques (Bates & Elman, 1993; Elman, Bates, Johnson, Karmiloff-Smith, Parisi, & Plunkett, 1996; McClelland, 1995; Plunkett & Sinha, 1992). Inspired by how biological neurons compute, artificial neural networks use graded distributed knowledge representations, pass activation among units, adjust connection weights, and in some cases recruit new units (Hertz, Krogh, & Palmer, 1991).

One of the more successful modeling techniques for cognitive developmental phenomena is the cascade-correlation algorithm. Cascade-correlation is a generative algorithm for learning in feed-forward neural networks. It was invented to correct the slowness and inability to learn some difficult problems that plagued the leading connectionist learning algorithm, back-propagation (Fahlman & Lebiere, 1990). Like other generative algorithms, cascade-correlation builds its own topology as it learns. It does this by recruiting new hidden units into the network as it needs them.[1] In contrast to cascade-correlation, many neural networks are static. That is, they do not

---

[1]Hidden units are those that neither receive input from the environment nor send output to the environment. In this sense, hidden units are like inter-neurons, as opposed to sensory or motor neurons. Hidden units are required for

change their topology as learning progresses; they merely adjust their connection weights. Generative algorithms like cascade-correlation undergo not only quantitative adjustments in connection weights but also qualitative adjustments in network topology through the recruitment of new hidden units. These two network learning techniques have been compared to assimilative and accommodative learning, respectively (Shultz, Schmidt, Buckingham, & Mareschal, 1995), a point that is elaborated in the General Discussion.

So far, cascade-correlation has produced successful models of balance scale phenomena (Shultz, Mareschal, & Schmidt, 1994), causal predictions of potency and resistance (Shultz et al., 1995b), seriation (Mareschal & Shultz, 1993), integration of velocity, time, and distance cues (Buckingham & Shultz, 1994), and acquisition of personal pronouns (Shultz, Buckingham, & Oshima-Takane, 1994a). Cascade-correlation networks produce either the only model or arguably the current best model in each of these domains. They are functional models, designed to cover the relevant cognitive and behavioral phenomena in a principled way, but not at a detailed neural level.

Although none of these cascade-correlation models were designed to simulate specific neural circuits, they do function according to established principles of neural functioning. Such neural principles include graded thresholded activation functions that sum across multiple inputs (J. A. Anderson, 1995; Crick & Asanuma, 1986), a layered topology with both direct and cascaded pathways (Crick & Asanuma, 1986; Thompson, 1986; Yeckel & Berger, 1990), and learning-driven synaptogenesis throughout life (Greenough & Bailey, 1988; Greenough, Withers, & B. J. Anderson, 1992; Quartz & Sejnowski, in press). Recruitment of new hidden units can be compared to the establishment of new neural circuitry through synaptogenesis, giving networks the ability to engage in constructivist development (Quartz, 1993; Mareschal & Shultz, 1996). In this comparison, it can be assumed that candidate hidden units are available for recruitment, but that they are either not yet connected or their connections are not yet functional.

There are four main regularities in the conservation literature that successful models would need to capture: (1) there is a shift from nonconservation to conservation beliefs regarding large quantities starting around the age of 6 to 7 years (hereafter, this is referred to as acquisition), and this shift can be rather abrupt, (2) correct conservation judgments emerge for small quantities before larger quantities (hereafter, the problem size effect), (3) nonconservers tend to choose the longer row as having more items than the shorter row (hereafter, the length bias effect), and (4) younger children (3-6 years) conserve only until they actually see the results of the transformation (hereafter, the screening effect because the effects of the transformation are temporarily screened from view). This article reports on attempts to qualitatively capture these four conservation phenomena, which are among the most well replicated and widely cited conservation phenomena, with neural networks constructed by the cascade-correlation algorithm.

Qualitative fits are sufficient because current modeling is so far from the actual phenomena, in terms of simplicity and abstraction, that precise fitting of data points would merely be an exercise in parameter tweaking. Cascade-correlation is used to qualitatively simulate many different developmental phenomena with default parameter settings, changing only the training and test sets. This generality is considered more important than adjusting parameters to produce exact fits.

---

neural networks to learn nonlinear functions, i.e., those functions that cannot be expressed by simple linear combinations of inputs.

Simulation Methods

Despite the many hundreds of studies of conservation acquisition, the nature of the natural experiences relevant to this acquisition remains unclear. Consequently, networks were trained in an environment with the fewest possible constraints. The simplest relevant environment has equivalence conservation problems in which rows of objects are described in terms of their perceptual characteristics, namely length and density. The rows are not described in terms of their number because that would make the conservation task trivially easy and would not conform to Piaget's (1965) efforts to limit the child's estimation abilities on these tasks. Target feedback supplied to the network concerns relative equality judgments comparing the two rows: Which row has more, or do they both have the same number of items? This is similar to the feedback employed in experiments that attempt to train conservation in children (Brainerd & Allen, 1971; Curcio, Kattef, Levine, & Robbins, 1972; Hamel & Riksen, 1973; Sheppard, 1974). The transformations used are those common to the psychological literature, including both those that alter number (addition and subtraction) and those that preserve number (elongation and compression). Conservation acquisition is not merely a matter of learning to conserve, but rather learning to distinguish those transformations that alter quantities from those that preserve quantities (Klahr, 1984; Siegler, 1981). To maintain problem difficulty, addition and subtraction transformations each alter a row by one item, and elongation and compression transformations decrease or increase the density of the row by one level, respectively. Examples of the four transformation types are presented in Table 1.

In these conservation problems, there are five levels of length and five levels of density, each ranging from 2-6 in the initial rows.[2] The inputs for each conservation problem are coded on 13 input units: two for each of two rows before and after the transformation, one for the identity of the transformed row, and four for the transformation. Each row is described in terms of its length and density, using a real number for each aspect. The identity of the transformed row is coded as -1 for one row, and 1 for the other row. Transformations are coded as 1 -1 -1 -1 for addition, -1 1 -1 -1 for subtraction, -1 -1 1 -1 for elongation, and -1 -1 -1 1 for compression. Note that the coding of transformation information is essentially arbitrary, on the assumption that the network must learn how this information is related to numeric comparisons.

Outputs reflecting a numerical comparison of the post-transformation rows are coded on two units with sigmoid activation functions.[3] If row 1 has more items than row 2, this is coded on the output units as 0.5 -0.5; if row 2 has more items than row 1, the output code is -0.5 0.5; and if the two rows have an equal number of items, the output code is -0.5 -0.5. The quantities for these numerical comparisons are computed as number = length x density.[4] In this way, it is possible

---

[2] The units of length and density measure are undefined.

[3] Such S-shaped activation functions ensure that the unit's level of activity is a graded threshold function of its input. Somewhat like biological neurons, the input must exceed a threshold in order to change the resting level of activation of the unit. Further increases beyond this threshold do not boost the unit's activation very much. Both hidden and output units in our simulations have sigmoid activation functions.

[4] For Piaget, conservation is based on what he called Grouping VII: The co-univocal multiplication of relations (Flavell, 1963). This is one of seven logical groupings that Piaget assumed to underlie concrete operational thought. Grouping VII in particular concerns the coordination of two asymmetrical series, in this case, longer and denser. Multiplication of coordinate values in these two series reveal compensation between them. When number = length x density and number is held constant, increases in length are in a sense compensated for by decreases in density. Conversely, increases in density are compensated for by decreases in length. In the simulations, these compensating

for a network to learn about number from the perceptual characteristics of items arranged in rows with a constant within-row density.

An example of this coding scheme for inputs and outputs is presented in Table 2. The example problem starts with initially unequal rows, the first of which receives one additional item during the transformation. Initially, the first row has six items, with a length of 3.0 and density of 2.0; the second row starts with seven items, with a length of 3.5 and density of 2.0. Because the transformation adds an item to the first row while keeping density constant, the first row comes to match the second row, with seven items and length of 3.5 and density of 2.0. The target output is -0.5 on each output unit because the two rows are now equal in number.

Table 2
Inputs and Outputs of a Network Along with an Example of the Coding Scheme for Initially Unequal Rows, the First of Which Receives One Additional Item

Input units	Output units
Pre-transformation row 1 length: 3.0	
Pre-transformation row 1 density: 2.0	
Pre-transformation row 2 length: 3.5	
Pre-transformation row 2 density: 2.0	Output 1: -0.5
Post-transformation row 1 length: 3.5	Output 2: -0.5
Post-transformation row 1 density: 2.0	
Post-transformation row 2 length: 3.5	
Post-transformation row 2 density: 2.0	
Identity of transformed row: -1	
Type of transformation:	
Transformation unit 1: 1	
Transformation unit 2: -1	
Transformation unit 3: -1	
Transformation unit 4: -1	

In this representation of conservation problems, space is traded for time in that the two rows are represented twice, once before and once after the transformation, in two different pairs of input units. The psychological idea underlying this scheme is that an image of the pre-transformation rows is maintained in active memory and processed along with perception of the post-transformation rows. The fact that children often mention pre-transformation rows in justifications of their conservation judgments supports the presence of those memories. Such trading of space for time is common in neural network simulations, although time-dependent problems can alternatively be represented in networks with recurrent connections (Hertz et al., 1991). Of course, with the passage of time represented spatially as here, the network must learn this aspect of the representation.

relations are contained in the structure of conservation problems in the environment. The assumption is that, through learning, cognition adapts to the structure of these problems.

With five levels of length and five levels of density, there are 5 x 5 = 25 initial rows. A pre-transformation partner row is created for each initial row by adding one item. Three pairs of pre-transformation rows are created as two identical rows, row 1 having one more item than row 2, and row 2 having one more item than row 1. Although such initial inequalities are rare in the conservation literature, it seems clear that skilled conservers should be able to deal with them. Multiplying the number of initial rows by these three initial relations between the two rows yields 25 x 3 = 75 initial pairs of rows. Either row can be transformed in any of the four possible ways mentioned earlier, yielding 2 x 4 = 8 transformations. Multiplying these 75 pairs of rows by the number of transformations yields 75 x 8 = 600 possible conservation of equivalence problems.

For each network, 420 training problems and 100 test problems are randomly selected from these 600 problems. Test problems are used in connectionist research to assess a network's ability to generalize to problems on which it has not been trained. All assessments of the effects of interest are performed on test problems rather than training problems. This distinction is important because it insulates network performance a bit from the particulars of training.

Each of the simulations contains 20 networks in each condition. Each of these 20 networks begins training from a unique set of initial randomly selected connection weights. As well, because of random selection of training and test problems, networks develop in somewhat different environments. In general, different networks were run in each simulation because different data were being recorded for particular purposes. However, the basic model remains constant across all of the simulations.

Networks stop learning when they master all of the training patterns, meaning that both output units are within a threshold (.4) of their target values on all of the training patterns.[5] An epoch is a sweep through the entire set of training problems. The epoch at which mastery of the training problems occurs is called the victory epoch. The simulations ran up to a maximum of 1500 epochs. Typically about one-half of the networks would have reached victory by then, and the other one-half of the networks would have been very close to victory, as indicated by the fact that these terminated networks would have outputs very close to their target values on all training patterns.

There are a number of parameters in the cascade-correlation algorithm, documented in the program code (Fahlman, 1991). Default values of these parameters were used, with the exception of input-epsilon and output-epsilon, which were each lowered to 0.01. These two parameters control the amount of linear gradient descent in updating input and output weights, respectively. Input weights are those going into hidden units; they are updated only while a hidden unit is being recruited. Output weights are those going into output units; they are updated when no recruitment of new units is occurring. The default values for these two parameters are 1.0 and 0.35, respectively, but there is nothing a priori about these default settings. Smaller steps, such as used here, are often helpful in reducing network error fluctuations during learning. Pilot simulations suggest that the basic simulation results hold up over a wide range of epsilon values. Mathematical and computational details of cascade-correlation can be found elsewhere (Fahlman & Lebiere, 1990; Shultz et al., 1994, 1995b).

[5]With output units having activation ranges from -.5 to .5, and a score-threshold of .4, any output activation below -.1 would count as correct for a target of -.5; any output activation above .1 would be correct if the target is .5.

Simulation 1: Acquisition

The first requirement of a successful conservation model would be to learn to perform equivalence conservation accurately, on both training and test problems. There is also evidence from a longitudinal study that children undergo a major sudden jump in acquiring conservation (van der Maas, 1993). One technique for analyzing sudden jumps in performance is to use multiple regression with two predictors of performance, continuous time and a dichotomous variable representing before or after the time of the major jump. When individual performance curves were matched on the time of the major jump in performance, this technique predicted conservation performance means for 24 children with a significant coefficient for the before-after variable (Raijmakers, van Koten, & Molenaar, 1996).

Results

The mean number of epochs required to master the training patterns was 1305, with a standard deviation of 298 epochs, and a range between 649 and 1500 epochs (the termination point). The mean number of hidden units recruited was 8.3, with a standard deviation of 2.0, and a range between 4 and 11.[6]

Proportions of correct training and test problems are plotted over 100 blocks of output epochs for two representative networks in Figure 1. Output epochs are those accumulated during training of output connection weights. Results for the epochs accumulated during hidden unit recruitment phases are not plotted because network performance does not change during the recruitment phase. There are 698 output epochs in the training of network 1 and 381 for network 16. In order to reduce clutter in these plots, epochs for each network are grouped into 100 blocks of equal size that span the entire training period. The proportions are averaged within each block. The blocks at which a hidden unit was recruited are marked with a triangle. It is typical in these simulations for performance to increase markedly after particular hidden units are recruited. For the two networks in Figure 1, recruitment of the second hidden unit led to large increases in conservation performance.

Performance on untrained test problems closely tracks that of trained problems indicating good generalization. At the end of training, the mean proportion correct on test problems was .95, with a standard deviation of .05, and a range from .82 to 1.0.

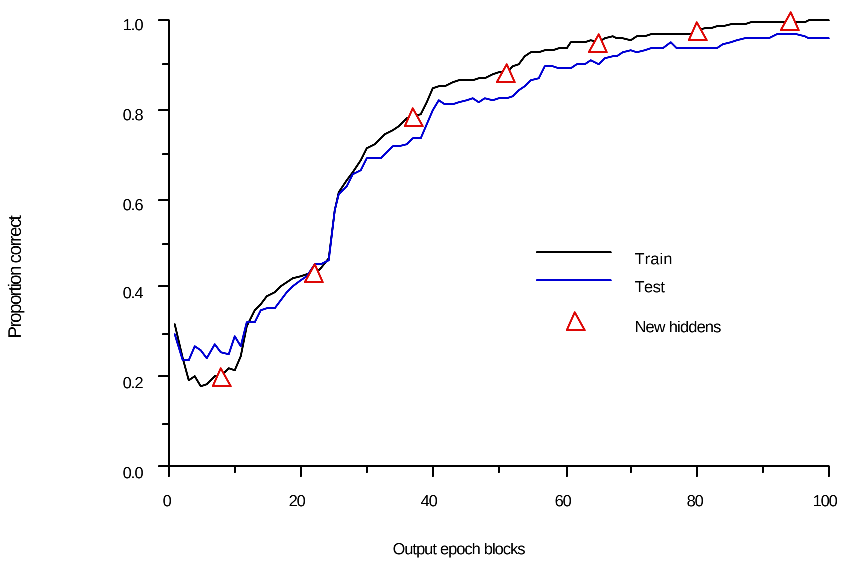
Sudden jumps in performance after the installation of new hidden units, such as are evident in Figure 1, were present in all 20 networks. The presence of a major sudden jump was tested with multiple regression. The key epoch for such jumps was identified for each network by the installation of the hidden unit that enabled the largest increase in proportion correct. This was the second hidden unit in 12 networks, the third hidden unit in six networks, the fourth hidden unit in one network, and the eighth hidden unit in one network. Nine epochs before and nine epochs after the key epoch were selected from each network, evenly spaced, with a spread to the beginning and to the end of training, respectively. These 19 epochs matched the number of testing sessions used in the analysis of children's longitudinal data (Raijmakers et al., 1996). The proportions correct for each of the 20 networks, as well as the mean proportions correct across

---

[6]The ten networks that terminated at 1500 epochs had a mean of 8.2 bits wrong (out of a possible 840 bits), with a standard deviation of 6.0 bits. These ten networks had a mean training error at 1500 epochs of 6.0, with a standard deviation of 4.0. To put this in context, at the first epoch, the mean training error for all 20 networks was 360.0, with a standard deviation of 86.7.

all 20 networks, were subjected to multiple regression. In each regression, there were two predictors, continuous time from 1 to 19 and a dichotomous variable with 0 representing times 1-10 and 1 representing times 11-19.

a. Network 1



b. Network 16

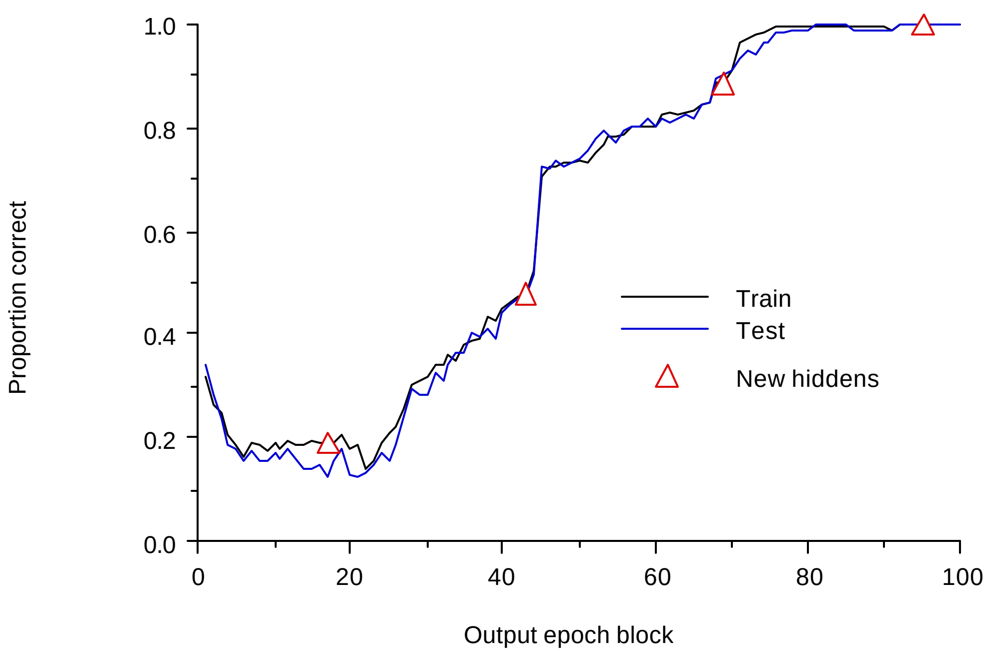


Figure 1. Learning and generalization in two networks.

Results of the multiple regressions are presented in Table 3 in terms of multiple R^2 and semipartial r^2 values for the time and before-after predictors. Semipartial r^2 values are computed as

$$sr_i^2 = \frac{t_i^2}{df_{res}}\left(1 - R^2\right) \qquad \text{Equation 1}$$

where t_i is the t-value assessing the significance of the regression coefficient for predictor i, and df_{res} are the residual degrees of freedom used in the F test of the significance of the multiple R^2 value (Tabachnick & Fidell, 1983). The p values in Table 3 are from these F and t-tests. The total proportion of variance predicted in conservation performance is given by the multiple R^2 values. The proportions of unique variance predicted in conservation performance are given by the semipartial r^2 values. These proportions of unique variance are rather small because of the high correlation between the two predictors, $r(17) = .866$, $p < .0001$. The coefficients and significance levels in Table 3 indicate that conservation acquisition in the networks was produced by both continuous growth and a sudden major spurt. All networks except network 20 had a significant regression coefficient for the time predictor; and all networks except network 17 had a significant regression coefficient for the before-after predictor. The before-after coefficient was insignificant for network 17 because this network had two large jumps in performance, after installation of the second and eighth hidden units. Coefficients for both the time and before-after predictors were significant when performance was averaged over all 20 networks. These means, along with standard deviation error bars are presented in Figure 2, where the major sudden jump at time 10 is evident.

Discussion

Successful learning of the training problems results from error reducing connection weight adjustments plus occasional increases in representational power supplied by new hidden units to deal with nonlinearities in the problems. An example nonlinearity is that adding one item does not always make a row more numerous than its partner; it depends on the initial numerical relations between the rows. For example, adding one to a row that has one less than the standard row makes the two rows equal in number.

Not only do these networks learn the training problems, but they also generalize well to the test problems. This means that they have not merely memorized the training patterns, but have abstracted an underlying function that allows correct responding to untrained problems.

The sudden jumps in conservation performance in these networks shortly after recruitment of particular hidden units mirror those found in children (Raijmakers et al., 1996). Although those authors argue that static connectionist networks (e.g., McClelland, 1995) cannot simulate such sudden jumps in performance, the present results suggest that generative networks can. Later, in Simulation 6, the source of these network jumps is examined. The significance of continuous time effects in the networks reflects the numerous smaller jumps in conservation performance that were enabled by installation of other hidden units (see Figure 1). The multiple regression technique for analyzing a single sudden jump is too crude to capture all of the jumps that occur. Finally, it is worth noting that results from the multiple regression technique for analyzing sudden jumps depend on the density and span of data sampling. Briefly, increasing the density and decreasing the span of data sampling are likely to minimize sudden jumps in performance.

Table 3
Results of Multiple Regression of Proportion Correct Onto Time and Before-after Major Jump

		Semipartial r^2	
Network	R^2	Time	Before-after
Mean	.995[a]	.036[a]	.045[a]
1	.989[a]	.058[a]	.070[a]
2	.965[a]	.102[a]	.035[c]
3	.955[a]	.029[c]	.112[a]
4	.985[a]	.139[a]	.021[b]
5	.982[a]	.065[a]	.067[a]
6	.976[a]	.064[a]	.066[a]
7	.958[a]	.109[a]	.031[c]
8	.989[a]	.003[d]	.206[a]
9	.978[a]	.120[a]	.029[b]
10	.979[a]	.105[a]	.033[a]
11	.969[a]	.023[c]	.125[a]
12	.977[a]	.022[c]	.131[a]
13	.994[a]	.011[a]	.157[a]
14	.984[a]	.018[b]	.145[a]
15	.963[a]	.059[a]	.069[a]
16	.955[a]	.045[c]	.086[a]
17	.899[a]	.143[b]	.012
18	.989[a]	.037[a]	.109[a]
19	.973[a]	.021[c]	.132[a]
20	.984[a]	.002	.213[a]

[a] $p < .0001$
[b] $p < .001$
[c] $p < .01$
[d] $p < .05$

Simulation 2: The Problem Size Effect

The problem size effect refers to the idea that children develop conservation with small numbers before large numbers (Cowan, 1979a, b; P. H. Miller & Heller, 1976; Siegler, 1981; G. A. Winer, 1974). In the simulations, a problem was considered to be small if the number of the smaller row was less than 12; large if the number of the smaller row was greater than 24. These values split the patterns so that small and large problems are about equally numerous.

Results

Proportions of small and large problems correct for all 20 networks are plotted in Figure 3 over ten blocks of epochs. For a given network, these blocks are of equal size and span the entire training period. These proportions were transformed to arcsins to uncorrelate the relation between means and variances (B. J. Winer, 1962):

$$transformation = 2\arcsin\sqrt{proportion} \qquad \text{Equation 2}$$

Arcsin values were subjected to an ANOVA in which epoch block, with ten levels, and problem size, with two levels, served as repeated measures. There are main effects of block, $F(9, 171) = 140$, $p < .001$, and size, $F(1, 19) = 5.93$, $p < .05$, and an interaction between block and size, $F(9, 171) = 5.00$, $p < .001$. The simple effect of size at each block, $F(1, 19)$, reaches $p < .01$ at blocks 2-5. At each of these blocks, performance is better on small problems than on large problems. Thus, the expected problem size effect is evident at intermediate blocks of epochs, but not at block 1, where the networks presumably have not learned enough to show the effect, nor at the final blocks, where networks approach a performance ceiling on all problem sizes.

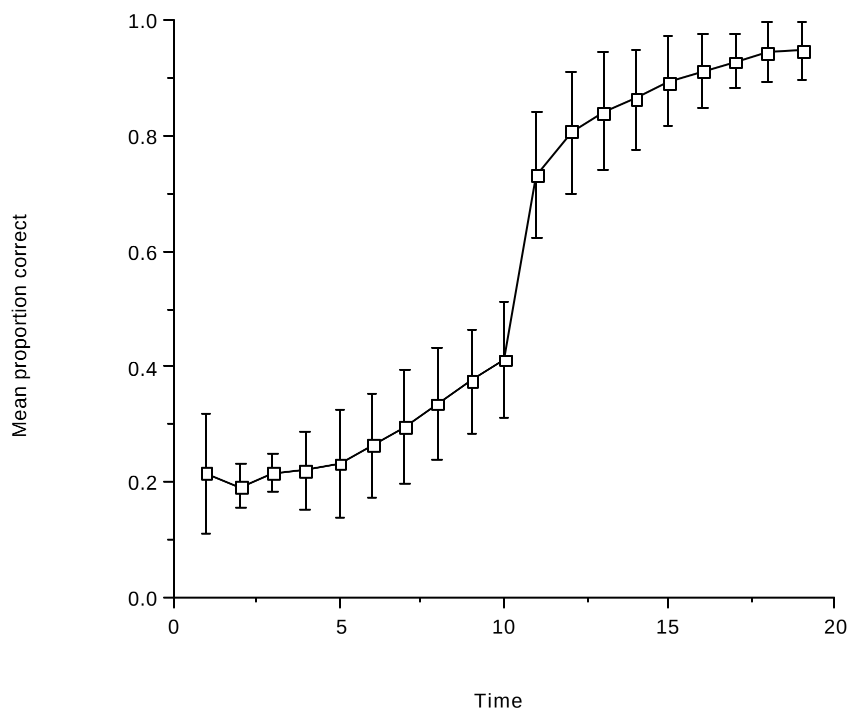


Figure 2. Mean proportion correct across 20 networks matched for the time of the major jump in performance.

Discussion

Problem size effects are pervasive in human quantitative judgments. For example, in tasks in which adults are asked to compare the size of two numbers, they are quicker with small numbers than with larger numbers (Moyer & Landauer, 1967). Children performing similar number comparison tasks show similar latency results and are more accurate with small numbers than with larger numbers (Sekuler & Mierkiewicz, 1977; Siegler & Robinson, 1982). It is possible to simulate such effects with connectionist networks such as cascade-correlation as long as the inputs are coded in an analog fashion (Hashmi & Shultz, 1997). In analog representation of number, the representation grows in intensity with the size of the number being represented.

With analog representation, small sizes are naturally easier to discriminate than large sizes. Probably this has to do with the fact that proportion differences between numbers are greater for small than for large numbers of the same absolute difference. For example, 3 is 50% greater than 2, but 8 is only 14% greater than 7. In contrast, arbitrary representations of number fail to simulate size effects on the same tasks.

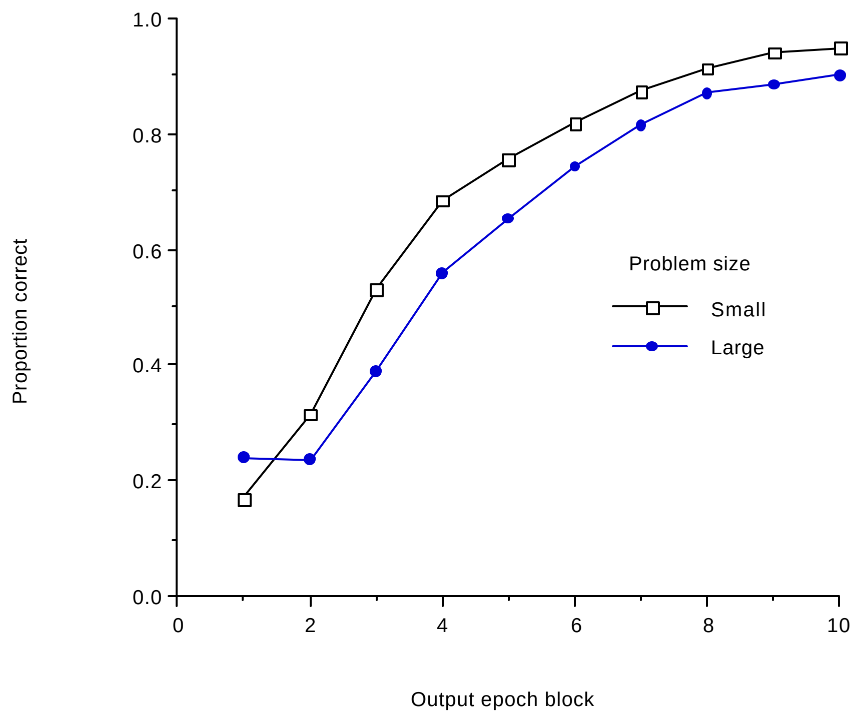


Figure 3. Problem size effect over blocks of epochs.

Past explanations of the problem size effect have emphasized either that children get more practice on smaller numbers (Ashcraft, 1992) or that children's estimation techniques are more accurate with small numbers (Silverman & Briga, 1981). Either of those factors could serve as the basis for neural network simulations, but the results would be entirely obvious because networks tend to learn better with more training and with more accurate feedback. The present simulations illustrate a more subtle point, that neural networks with analog representations of number show a problem size effect even though they do not have more experience with or better estimations of small numbers. Problem size effects in children may be over-determined in the sense that they could be caused by practice, estimation accuracy, and analog representations.

It is somewhat difficult to compare the sizes of the problem size effect between simulated and psychological data. Different studies use different criteria for defining small and large problems, different age levels, and different performance criteria, all of which could affect the results of such comparisons. Cowan (1979a) contrasted rows of 2, 5, or 15 items in 5-year-olds and found proportions of correct conserving answers to be 1.0, .78, and .62, respectively. P. H. Miller and Heller (1976) obtained proportions of correct conserving answers in 5-year-olds of

.83 with four items and .73 with eight items. Both of these studies used number preserving transformations with initially equal rows, whereas the present simulations contain a wider variety of conservation problems, with both equal and unequal initial rows and both number preserving and number altering transformations. The definition of small and large number problems in the simulations also differs substantially from those used in these psychological studies. Finally, it is difficult to match particular age levels against epochs of network training. Nonetheless, the maximum mean differences of about .10 in the simulations (as revealed in Figure 3) are not seriously out of line with the size of the differences reported in children. It is possible that the size of the problem size effect in networks could be modulated by various parameter adjustments, but precise quantitative fits done on an ad hoc basis are not the purpose of these simulations. What is important is qualitative coverage of phenomena for principled reasons.

Simulation 3: The Length Bias Effect

Length bias refers to nonconservers tending to choose a longer row as having more items than a shorter row in conservation of discrete number tasks (P. H. Miller, Grabowski, & Heldmeyer, 1973; Piaget, 1965; Siegler, 1995). In conservation of continuous quantity tasks, length bias might take the form of choosing a taller beaker as having more liquid than a shorter beaker, or a longer, thinner sausage as having more to eat than a shorter, fatter sausage (P. H. Miller, 1973; P. H. Miller et al., 1973). As with Piaget and other investigators, the test for length bias uses only elongation and compression problems that have initially equal rows. Thirty of the 100 such equality conserving problems are randomly selected as test problems. The remaining 70 equality conserving problems enter the training set, conforming to a pattern of training on 70% of the relevant problems, along with 350 other randomly selected problems. Choosing the shorter, denser row as having more in these equality conserving problems is considered a density bias.

The following procedure was used to determine which row a network chose as having more items: If the first output activation exceeded 0 and the second output activation was less than 0, then the network was deemed to have chosen the first row as having more; if the first output activation was less than 0 and the second output activation exceeded 0, then the network was deemed to have chosen the second row as having more; if both output activations were less than 0, then the network was deemed to have decided that the rows were numerically equal.

Results

Numbers of test problems on which each network showed either length or density bias each output epoch were collapsed into mean problems showing each bias in each of ten equal sized blocks of epochs. These mean numbers of biased answers were subjected to an ANOVA in which epoch block, with ten levels, and type of bias, with two levels, served as repeated measures. Mean problems with length and density bias are shown in Figure 4. There are main effects of block, $F(9, 171) = 15.29$, $p < .001$, and bias, $F(1, 19) = 28.75$, $p < .001$, and an interaction between them, $F(9, 171) = 5.72$, $p < .001$. The simple effect of bias, $F(1, 19)$, reaches $p < .05$ at blocks 1-8, reflecting more length bias than density bias. Length bias decreases in the later blocks as the networks come to master conservation problems.

Length and density biases are expressed in Figure 5 in terms of proportions of errors, where an error is a nonconservation response.

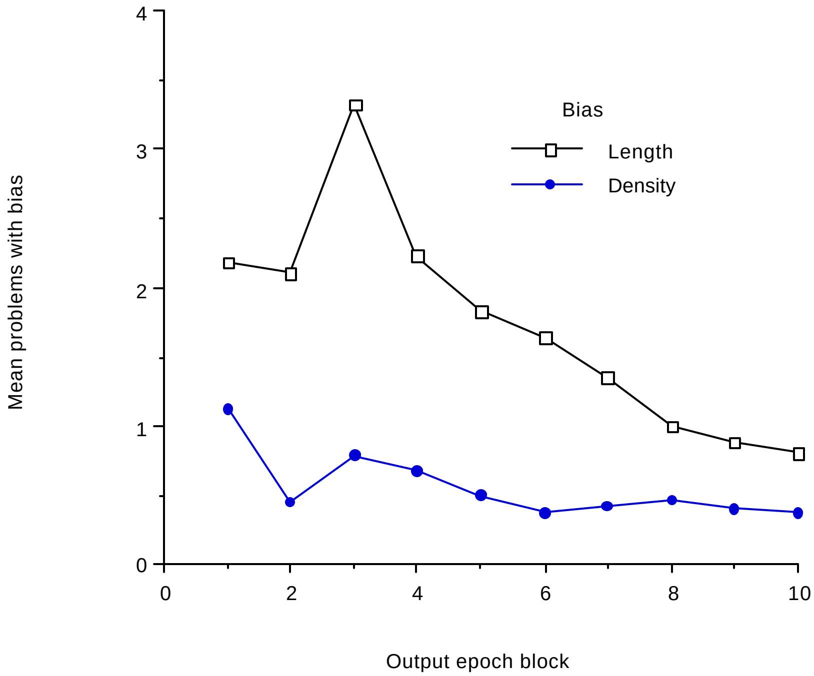


Figure 4. Length and density bias over blocks of epochs under constant density.

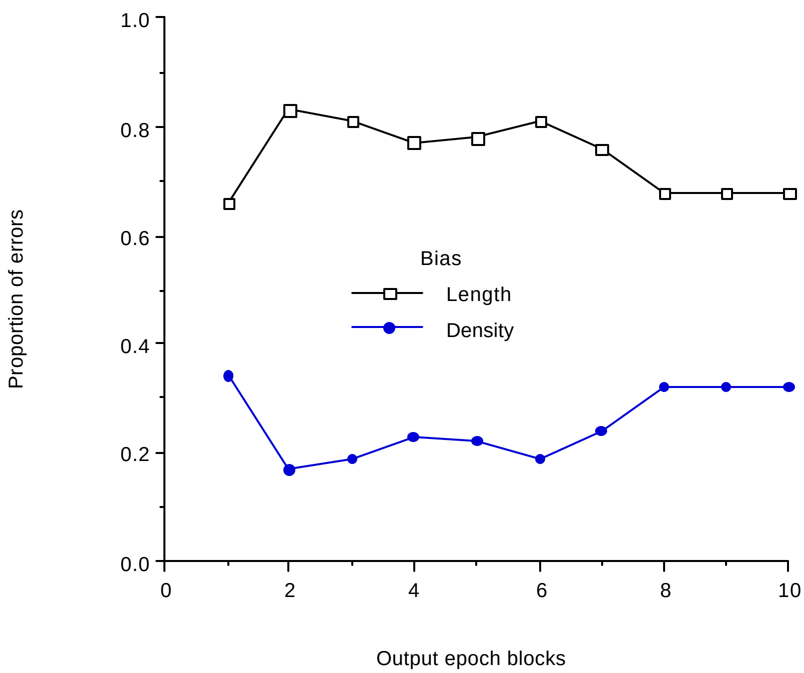


Figure 5. Bias over blocks of epochs under constant density expressed as proportion of errors.

Discussion

In this environment of constant density during transformations, there is clearly more length bias than density bias up until the final blocks of epochs. Although there is some length bias even in the first epoch block, length bias increases in the third block, underscoring that it is, to a considerable extent, a learned bias. As learning continues and conservation is mastered, the length bias diminishes. The absolute amount of length bias in these networks is less than that typically observed with children, partly because the large numbers of errors in early epochs are swamped by the greater success of later epochs in the same block, but the length bias is strong and statistically reliable. When expressed as a proportion of errors, as in Figure 5, length bias in the networks approximates the value of .80 found in pilot research in our laboratory with 5-year-old children, values of .69 - .86 found in various conditions by Miller et al. (1973) with clay sausages, and the value of .77 found in Siegler's (1995) pre-test of number conservation. That is, in both children and networks, approximately 80% of the errors on initially equal conserving problems result from choosing the longer row. Subsequently conducted pilot simulations indicate that the absolute amounts of length bias can be increased by various manipulations, such as increasing the range of length values relative to density values, or increasing the relative proportion of number altering transformation problems in the training set. Again, because the present simulations strive for qualitative and not quantitative coverage of human data, there was no interest in using such parameters to improve quantitative fits.

Simulation 4: Explaining the Length Bias Effect

Length bias on conservation tasks has been attributed to either perceptual salience (length is more salient than number) or learning that longer rows often have more items than shorter rows (Bryant, 1972; P. H. Miller, 1973). Although the issue has not been settled by psychological research, it has been noted that very young children do not show a length bias (P. H. Miller et al., 1973). This is consistent with the idea that some time is required for children to learn that length is a somewhat reliable cue to number. It is not definitive, however, because salience might also change with experience or development.

The learning explanation seems reasonable in the context of transformations that change number, namely addition and subtraction. These transformations are capable of introducing a correlation between number and length, as long as density is held constant during the transformation. With constant density, whenever an item is added to a row, the row becomes longer as well as more numerous; whenever an item is subtracted, the row becomes shorter as well as less numerous.

The impact of this length-number correlation on conservation performance was tested by creating an alternate environment in which length, rather than density, was held constant during number altering transformations. Example transformations of this sort are provided in Table 4. Here, when an item is added to a row, the row is compressed so that density increases and length does not change. Similarly, during subtraction, the row is elongated so that density decreases and length remains the same. In order to preserve symmetry with Simulation 3, length rather than density is changed by one in elongation and compression transformations. Other than these changes, the simulation was performed in an identical manner with Simulation 3.

Table 4
Example Transformations with Constant Length

Transformation	Length	Density	Row
Pre-transformation	2	2	o o o o
Add	2	2.5	o o o o o
Subtract	2	1.5	o o o
Elongate	3	1.33	o o o o
Compress	1	4	oooo

Results

Mean problems with length and density bias are shown in Figure 6. There is a main effect of block, $F(9, 171) = 12.53$, $p < .001$, and bias, $F(1, 19) = 20.23$, $p < .001$, and an interaction between block and bias, $F(9, 171) = 7.30$, $p < .001$. The simple effect of bias, $F(1, 19)$, reaches $p < .01$ at blocks 1-4, reflecting more density bias than length bias.

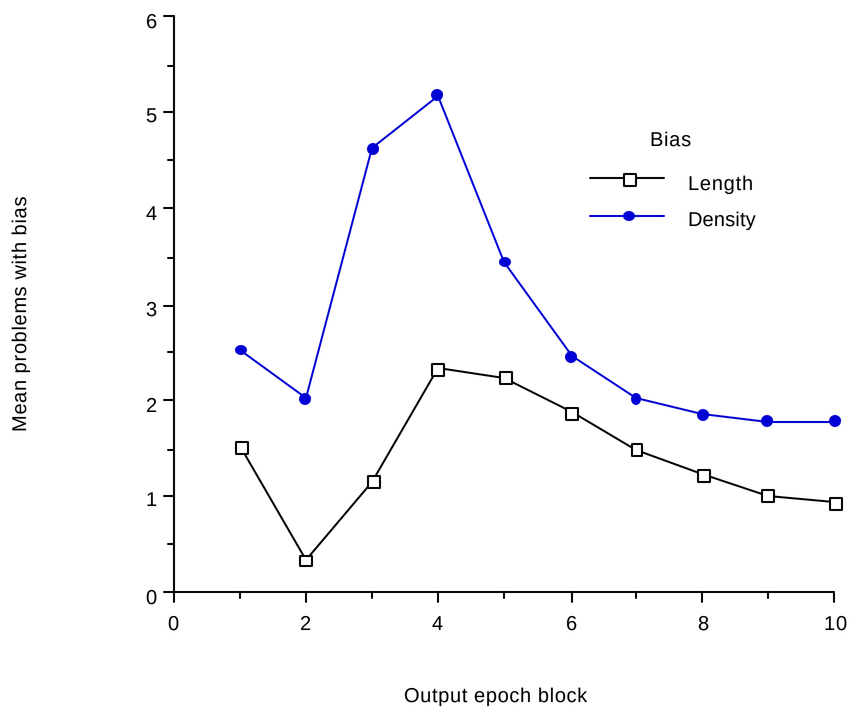


Figure 6. Length and density bias over blocks of epochs under constant length.

Discussion

Nonconservation biases are strongly influenced in these simulations by the correlation of length or density with number during number altering transformations. When density is held constant during number altering transformations, number correlates better with length than with density, and so length becomes a good predictor of number. When length is held constant, number correlates better with density than with length, and so density becomes a good predictor of number. The default condition would appear to be that of constant density because it is more efficient to leave density alone when adding or subtracting an item from a row.

These results on perceptual bias underscore the tension between perception and cognition in conservation tasks. Piaget designed these tasks to create a somewhat artificial conflict between perception and cognition. What the child knows (e.g., that a transformation does not alter a quantity) appears to conflict with what she sees (e.g., that one row is longer, and thus appears more numerous than the other).

To put this another way, there are two methods for solving conservation problems, one perceptual and the other cognitive (Shultz, Dover, & Amsel, 1979). In number conservation, the perceptual solution employs the cues of length and density of the rows after the transformation. Early in development, this leads to nonconservation responses attributable to length bias. Later on, perceptual solutions can yield correct conservation judgments as length is coordinated with density to estimate quantity. Recall that, in the simulations, number = length x density. A more cognitive solution to conservation involves reasoning about the effects of the transformation, given the initial relation between the two rows. If the transformation is one that does not alter number, then that initial relation is preserved. But if the transformation alters number, then the equivalence relation is also altered. Perceptual solutions dominate early in children's conservation development, but eventually are subordinated to cognitive solutions (Halford & Boyle, 1985; Piaget, 1965; Siegler, 1981). Early, but diminishing length bias in Simulation 3 conforms to this account.

Simulation 5: The Screening Effect

The screening effect refers to younger children (3-6 years) conserving only until they see the results of a transformation (Bruner, Olver, & Greenfield, 1966; Piaget & Inhelder, 1971; P. H. Miller & Heldmeyer, 1975). As long as the effects of the transformation are screened from view, they are conservers. As soon as the screen is removed, and the results of the transformation become apparent, they revert to nonconservation.

The screening effect was simulated by coding test problems with zeroes in the length and density inputs of the transformed row after the transformation. For these so-called screened problems, the networks had information about the appearance of the rows before the transformation, and the non-transformed row after the transformation, and which transformation was being applied, but not the way the transformed row looked after it was transformed. Networks were trained in the standard way, without screening, as in Simulation 1, but were tested on both screened and unscreened versions of the test problems.

Results

Because it became apparent that the screening effect would appear only very early in learning, the focus is on the first 10 of 40 blocks of output epochs. Proportions of correct test problems were transformed to arcsins as in Equation 1. Arcsin values were subjected to an ANOVA in which blocks of output epochs, with ten levels, and presentation, with two levels, served as repeated measures factors. Mean proportions correct over the first eight of 40 blocks of output epochs are plotted in Figure 7. There are main effects of block, $F(9, 171) = 44.54$, $p <$

.001, and an interaction between block and presentation, $F(9, 171) = 20.89$, $p < .001$. Simple effects of presentation, $F(1, 19)$, reach $p < .05$ at blocks 2-7 and 9-10. At blocks 2-7, performance is better on screened problems than on unscreened. By block 9, the superiority of unscreened problems emerges, because that is what the networks are being trained on.

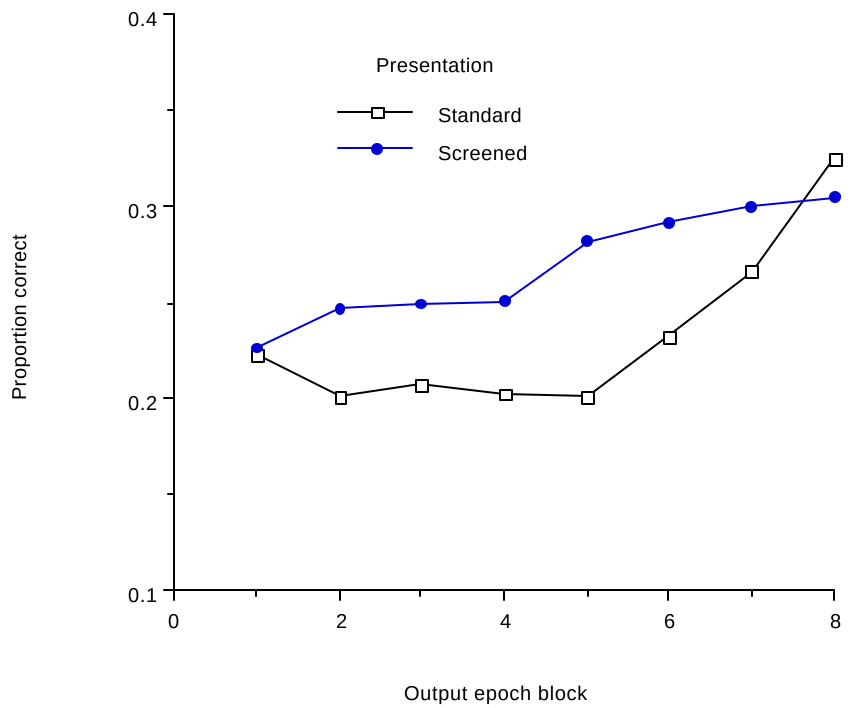


Figure 7. Screening effect over the first eight of 40 blocks of output epochs.

Discussion

These results again reflect the tension between perception and cognition in conservation problems. When misleading perceptual information is removed, the networks are more likely to conserve, at least early in development. The reason that the screening effect only appears early in acquisition is that, later on, the networks are not so easily misled by perceptual information. As the networks come to master conservation, their length bias drops as noted in Simulation 3, as does their use of perceptual inputs as seen in the next simulation.

Simulation 6: Network Analysis

Although conservation stages and associated perceptual phenomena have been well documented in children and simulated in cascade-correlation networks, the knowledge representations achieved at various stages are still unknown. Because emerging knowledge representations can be so difficult to study in children, it may be worthwhile to see how networks, being more accessible and easier to analyze, learn to conserve.

A variety of techniques have been developed for network analysis, including visual examination of connection weight diagrams (e.g., Hinton & Sejnowski, 1986) and unit activation patterns (e.g., Elman, 1989, 1990), statistical analysis of network contributions (Shultz, Oshima-Takane, & Takane, 1995), rule diagnosis (e.g., Bochereau & Bourgine, 1990; Hayashi, 1991), and graphical analysis of how closely a network approximates the function it is trying to learn (Takane, Oshima-Takane, & Shultz, 1994).

In general, network analysis can be difficult because of the distributed nature of network knowledge representations and the nonlinearity of both target functions and unit activation functions. These difficulties are compounded in the present model by the relatively high dimensionality created by multiple units, the presence of cross-connections, and the redundancy of input information and thus solutions. Nonetheless, it is possible to examine the insides of networks, which is, of course, impossible with children.

Method

Contributions are products of sending unit activations and connection weights going into output units. They are particularly informative regarding networks with cross-connections that bypass hidden units, as in cascade-correlation networks (Shultz et al., 1995a). In 20 networks, network contributions were obtained at the end of each output phase, and once again at the end of training. At those points, the network would have adapted as well as possible to the newest hidden unit (or to the absence of any hidden units in the first output phase). Error on the training and test problems was also recorded at those epochs to provide an indication of the network's performance on conservation problems. The matrix of contributions x 100 test patterns was subjected to a Principal Components Analysis (PCA) in order to reduce the dimensionality of the contributions. Each PCA was performed on the covariance matrix and included a varimax rotation to improve interpretability. A scree test was performed to determine how many components to include in the analysis (Cattell, 1966). Rotated loadings were standardized by dividing them by the standard deviation of the respective contribution scores.

Results

Standardized rotated loadings from PCAs of contributions for a single representative network are presented in Tables 5-10 for the end of each output phase in that network's development. The highest loadings are highlighted in bold to aid in interpretation of the components. Each contribution is described in terms of the sending and receiving units, and input units are further identified by the information they encode.

Table 5 shows the standardized rotated loadings at the end of the first output phase before any hidden units had been recruited. Two components account for 92% of the variance in contributions. Component 1 has heavy loadings from the length inputs and accounts for 54% of the variance in contributions. Component 2 has heavy loadings from the density inputs and accounts for 38% of the variance. At this point, no other contributions load on any components, suggesting that the network is primarily representing length and density of the rows and ignoring information on transformations. Interpretation of the loadings in Tables 5-10 is facilitated by the strong vertical symmetry in these Tables. Any contribution with a high loading flowing into output unit 1 has a mirror image contribution flowing into output unit 2 with an equally high loading, and vice versa.

Table 5
Standardized Loadings from PCA of Contributions for Network 13 before Any Hidden Units

Contribution	Input unit	Component and % variance 1	2
		54%	38%
Bias-Output1		.00	.00
Input1-Output1	length row 1 pre	**.98**	-.01
Input2-Output1	density row 1 pre	-.04	-**.99**
Input3-Output1	length row 2 pre	-**.99**	-.00
Input4-Output1	density row 2 pre	.04	**1.00**
Input5-Output1	length row 1 post	**.89**	.10
Input6-Output1	density row 1 post	.08	.94
Input7-Output1	length row 2 post	-**.85**	-.14
Input8-Output1	density row 2 post	.07	**.96**
Input9-Output1	transformed row	.03	-.03
Input10-Output1	add	.10	-.10
Input11-Output1	subtract	.07	-.02
Input12-Output1	elongate	-.09	-.10
Input13-Output1	compress	-.19	-.08
Bias-Output2		.00	.00
Input1-Output2	length row 1 pre	-**.98**	.01
Input2-Output2	density row 1 pre	-.04	-**.99**
Input3-Output2	length row 2 pre	**.99**	.00
Input4-Output2	density row 2 pre	.04	**.99**
Input5-Output2	length row 1 post	-**.89**	-.10
Input6-Output2	density row 1 post	.08	**.94**
Input7-Output2	length row 2 post	**.85**	.14
Input8-Output2	density row 2 post	-.07	-**.96**
Input9-Output2	transformed row	.00	.00
Input10-Output2	add	.13	-.08
Input11-Output2	subtract	.06	.00
Input12-Output2	elongate	.08	.12
Input13-Output2	compress	.19	.06

Table 6 displays standardized loadings at the end of the second output phase after the network has adjusted to a hidden unit. Now there are three components, together accounting for 85% of the variance in contributions. The first and third components deal with length and density, respectively, while the second component deals with the identity of the transformed row and the first hidden unit. Both training and test error have decreased over this second output phase, from 152.3 to 107.3 and from 40.2 to 30.1, respectively, indicating marked improvement in conservation performance.

Table 6
Standardized Loadings from PCA of Contributions for Network 13 after One Hidden Unit

		Component and % variance		
Contribution	Input unit	1 42%	2 28%	3 15%
Bias-Output1		.00	.00	.00
Input1-Output1	length row 1 pre	-**.96**	.11	-.04
Input2-Output1	density row 1 pre	.05	.00	-**.99**
Input3-Output1	length row 2 pre	**.97**	-.11	.02
Input4-Output1	density row 2 pre	-.05	.00	**.99**
Input5-Output1	length row 1 post	-**.87**	.23	.10
Input6-Output1	density row 1 post	-.09	.05	**.95**
Input7-Output1	length row 2 post	**.90**	.20	-.12
Input8-Output1	density row 2 post	-.11	-.19	**.96**
Input9-Output1	transformed row	.08	**.78**	.10
Input10-Output1	add	-.11	.38	-.08
Input11-Output1	subtract	-.07	.22	.02
Input12-Output1	elongate	.15	.29	-.15
Input13-Output1	compress	.23	.17	-.07
Hidden1-Output1		-.07	-**.99**	-.03
Bias-Output2		.00	.00	.00
Input1-Output2	length row 1 pre	**.96**	-.11	.04
Input2-Output2	density row 1 pre	.05	.00	-**.99**
Input3-Output2	length row 2 pre	-**.97**	.11	-.02
Input4-Output2	density row 2 pre	-.05	-.00	**.99**
Input5-Output2	length row 1 post	**.87**	-.23	-.10
Input6-Output2	density row 1 post	-.09	.05	**.95**
Input7-Output2	length row 2 post	-**.90**	-.20	.12
Input8-Output2	density row 2 post	.11	.19	-**.96**
Input9-Output2	transformed row	-.08	-**.78**	-.10
Input10-Output2	add	.11	-.38	.08
Input11-Output2	subtract	.07	-.23	-.02
Input12-Output2	elongate	-.15	-.29	.14
Input13-Output2	compress	-.23	-.17	.07
Hidden1-Output2		.07	**.99**	.03

After recruiting another hidden unit, things have not changed very much, as revealed in Table 7. Again, the three components account for a total of 85% of the variance in contributions. It would appear that the second hidden unit has not moved the solution to conservation problems very far along. This is supported by the fact that error on the training problems has decreased only a little, from 107.3 to 95.7 on the training problems and from 30.5 to 27.3 on the test problems. There is, in other words, no important change in knowledge representation and very little improvement in performance during this third output phase. Occasionally, cascade-correlation networks do recruit hidden units that are not very helpful, at least initially. As will be seen momentarily, this particular second hidden unit does appear to play an important role later on.

Table 7
Standardized Loadings from PCA of Contributions for Network 13 after Two Hidden Units

		Component and % variance		
Contribution	Input unit	1 41%	2 29%	3 15%
Bias-Output1		.00	.00	.00
Input1-Output1	length row 1 pre	**.97**	.07	-.04
Input2-Output1	density row 1 pre	-.05	-.03	-**.99**
Input3-Output1	length row 2 pre	-**.97**	-.07	.02
Input4-Output1	density row 2 pre	.05	.03	**.98**
Input5-Output1	length row 1 post	**.88**	.20	.10
Input6-Output1	density row 1 post	.09	.07	**.95**
Input7-Output1	length row 2 post	-**.89**	.23	-.13
Input8-Output1	density row 2 post	.10	-.17	**.96**
Input9-Output1	transformed row	-.05	**.79**	.08
Input10-Output1	add	.12	.37	-.09
Input11-Output1	subtract	.08	.22	.01
Input12-Output1	elongate	-.14	.30	-.16
Input13-Output1	compress	-.22	.18	-.08
Hidden1-Output1		.03	-**.99**	-.00
Hidden2-Output1		-.06	.16	.05
Bias-Output2		.00	.00	.00
Input1-Output2	length row 1 pre	-**.97**	-.07	.04
Input2-Output2	density row 1 pre	-.05	-.03	-**.99**
Input3-Output2	length row 2 pre	**.97**	.07	-.02
Input4-Output2	density row 2 pre	.05	.03	**.99**
Input5-Output2	length row 1 post	-**.87**	-.20	-.10
Input6-Output2	density row 1 post	.10	.07	**.95**
Input7-Output2	length row 2 post	**.89**	-.23	.13
Input8-Output2	density row 2 post	-.10	.16	-**.96**
Input9-Output2	transformed row	.05	-**.79**	-.08
Input10-Output2	add	-.13	-.37	.09
Input11-Output2	subtract	-.08	-.22	-.01
Input12-Output2	elongate	.14	-.29	.16
Input13-Output2	compress	.22	-.18	.08
Hidden1-Output2		-.03	**.99**	.00
Hidden2-Output2		.06	-.16	-.05

The relative stagnation of the third output phase stands in marked contrast to the fourth output phase, the loadings for which are portrayed in Table 8. There are now four components, accounting for 84% of the variance in contributions. The density component has disappeared and the length component now accounts for only 9% of the variance, suggesting that pure perceptual information is not very important in the network's current representation of conservation problems. Component 1, accounting for 33% of the variance, has heavy loadings from the second and third hidden units. Component 2, with 31% of the variance, continues to focus on the identity of the transformed row and hidden unit 1. In component 3, the network begins for the first time to focus on transformation information, in this case elongation. Subtraction contributions load in a distributed way on two of the components, 2 and 3. Interestingly, there is a substantial decrease in test error across this fourth output phase, from 95.7 to 16.0 on the training problems and from 27.3 to 3.8 on the test problems. The major representational changes that have taken place allow for major increases in conservation performance.

In Table 9, after the fourth hidden unit, there are five components, accounting for a total of 90% of the variance in contributions. The principal changes can be traced to components 3 and 4, with component 3 focusing on the compression transformation and the new hidden unit, and component 4 emphasizing addition and elongation transformations. Error continues to drop, although not as dramatically as in the previous phase, from 16.0 to 3.3 on the training patterns and from 3.8 to 1.5 on the test patterns.

Table 10 presents the loadings on five components, accounting for 90% of the variance, at the end of training, after recruitment of a fifth hidden unit. There is not much change from the previous output phase. Component 1 is still associated with the second and third hidden units, component 2 with the identity of the transformed row and the first hidden unit, component 3 with compression and hidden unit 4, and component 5 with length. All of the transformations are well represented, addition and elongation on component 4, compression on component 3, and subtraction distributed across components 2 and 4. The fifth and final hidden unit loads in a distributed way on components 1, 2, 4, and 5, suggesting that it is perhaps playing a mopping-up role, rather than providing any dramatically new representation of knowledge. Error has dropped to 1.86 on the training problems and to 1.0 on the test problems. Even in victory, there can be some residual error on the training patterns, because output activations do not need to match their targets exactly; they only need to be within score-threshold (0.4) of their targets.

Discussion

The general picture gained from PCA of network contributions in all networks so far examined is characterized by an early focus on perceptual information (i.e., length and density), an intermediate attention to the identity of the transformed row, and a later focus on the impact of particular transformations. Hidden units are required for the second and third representation stages, and all of the critical input information is represented, at least at some point in development. Until all of the critical input information is represented, conservation problems are not fully mastered. Dramatic increases in conservation performance coincide with major changes in knowledge representation. A major burst in performance typically occurs after recruitment of the second or third hidden unit, coinciding with the first representations of transformation information.

Recruitment of critical hidden units is responsible for both the major changes in knowledge representation and the sudden jump in performance. These critical hidden units supply additional

Table 8
Standardized Loadings from PCA of Contributions for Network 13 after Three Hidden Units

		Component and % variance			
Contribution	Input unit	1	2	3	4
		33%	31%	11%	9%
Bias-Output1		.00	.00	.00	.00
Input1-Output1	length row 1 pre	-.09	.00	-.16	**-.94**
Input2-Output1	density row 1 pre	.07	-.04	.15	.09
Input3-Output1	length row 2 pre	-.06	-.01	.16	**.94**
Input4-Output1	density row 2 pre	.07	-.04	.16	.09
Input5-Output1	length row 1 post	-.10	.17	-.48	**-.74**
Input6-Output1	density row 1 post	.05	-.11	.35	.07
Input7-Output1	length row 2 post	-.21	.21	.39	**.81**
Input8-Output1	density row 2 post	-.00	-.11	-.34	-.09
Input9-Output1	transformed row	-.13	**.80**	-.16	.09
Input10-Output1	add	-.19	.34	.24	-.38
Input11-Output1	subtract	-.11	-.34	.41	-.04
Input12-Output1	elongate	.26	-.14	-.93	.20
Input13-Output1	compress	.01	.15	.44	.22
Hidden1-Output1		.05	**-.98**	-.16	.06
Hidden2-Output1		**-.99**	.06	-.05	.03
Hidden3-Output1		**-.88**	.08	-.03	-.01
Bias-Output2		.00	.00	.00	.00
Input1-Output2	length row 1 pre	.09	-.00	.16	**.94**
Input2-Output2	density row 1 pre	.07	-.04	.15	.09
Input3-Output2	length row 2 pre	.06	.01	-.16	**-.94**
Input4-Output2	density row 2 pre	-.07	.04	-.15	-.09
Input5-Output2	length row 1 post	.10	-.17	.48	**.74**
Input6-Output2	density row 1 post	-.05	.11	-.35	-.07
Input7-Output2	length row 2 post	.21	-.21	-.39	**-.81**
Input8-Output2	density row 2 post	.00	.11	.34	.09
Input9-Output2	transformed row	.13	**-.80**	.16	-.09
Input10-Output2	add	-.19	.34	.24	-.38
Input11-Output2	subtract	-.11	-.35	.41	-.04
Input12-Output2	elongate	.26	-.14	**-.93**	.20
Input13-Output2	compress	.01	.15	.44	.22
Hidden1-Output2		-.05	**.98**	.16	-.06
Hidden2-Output2		**.99**	-.06	.05	-.03
Hidden3-Output2		**.88**	-.08	.03	.01

Table 9
Standardized Loadings from PCA of Contributions for Network 13 after Four Hidden Units

		Component and % variance				
Contribution	Input unit	1	2	3	4	5
		31%	25%	14%	15%	5%
Bias-Output1		.00	.00	.00	.00	.00
Input1-Output1	length row 1 pre	.09	.02	.07	-.12	**.94**
Input2-Output1	density row 1 pre	-.06	.05	-.17	.00	-.05
Input3-Output1	length row 2 pre	.06	-.01	-.06	.11	**-.94**
Input4-Output1	density row 2 pre	-.06	.05	-.16	.00	-.05
Input5-Output1	length row 1 post	.12	-.16	.28	.13	**.85**
Input6-Output1	density row 1 post	-.03	.12	-.34	-.12	-.09
Input7-Output1	length row 2 post	.23	-.20	-.32	-.04	**-.76**
Input8-Output1	density row 2 post	-.02	.11	.34	.09	.08
Input9-Output1	transformed row	.11	**-.74**	.22	-.07	-.06
Input10-Output1	add	.05	-.31	.30	-.67	.10
Input11-Output1	subtract	.05	.43	-.18	-.42	.02
Input12-Output1	elongate	-.19	.08	.52	**.82**	.06
Input13-Output1	compress	.12	-.21	**-.77**	.17	-.20
Hidden1-Output1		-.06	**.97**	.12	.20	-.06
Hidden2-Output1		**.97**	-.03	.16	-.14	-.05
Hidden3-Output1		**.89**	-.08	.00	-.02	.04
Hidden4-Output1		-.05	-.37	**.81**	.02	-.28
Bias-Output2		.00	.00	.00	.00	.00
Input1-Output2	length row 1 pre	-.09	-.02	-.07	.12	**-.94**
Input2-Output2	density row 1 pre	-.06	.05	-.17	.00	-.05
Input3-Output2	length row 2 pre	-.06	.01	.06	-.11	**.94**
Input4-Output2	density row 2 pre	.06	-.05	.17	-.00	.05
Input5-Output2	length row 1 post	-.12	.16	-.28	-.13	**-.85**
Input6-Output2	density row 1 post	.03	-.12	.34	.12	.09
Input7-Output2	length row 2 post	-.22	.20	.32	.04	**.77**
Input8-Output2	density row 2 post	-.02	.11	.33	.09	.07
Input9-Output2	transformed row	-.11	.74	-.22	.07	.06
Input10-Output2	add	.05	-.31	.30	-.67	.10
Input11-Output2	subtract	.05	.43	-.18	-.41	.02
Input12-Output2	elongate	-.18	.08	.52	**.82**	.06
Input13-Output2	compress	.12	-.21	**-.77**	.17	-.20
Hidden1-Output2		.06	**-.97**	-.12	-.20	.06
Hidden2-Output2		**-.97**	.03	-.16	.15	.05
Hidden3-Output2		**-.89**	.08	-.00	.02	-.04
Hidden4-Output2		.05	.37	**-.81**	-.02	.28

Table 10
Standardized Loadings from PCA of Contributions for Network 13 after Five Hidden Units

		Component and % variance				
Contribution	Input unit	1 32%	2 24%	3 15%	4 14%	5 5%
Bias-Output1		.00	.00	.00	.00	.00
Input1-Output1	length row 1 pre	.09	.03	.07	-.14	**.92**
Input2-Output1	density row 1 pre	-.06	.05	-.16	.01	-.04
Input3-Output1	length row 2 pre	.06	-.02	-.06	.13	**-.92**
Input4-Output1	density row 2 pre	-.06	.05	-.16	.01	-.04
Input5-Output1	length row 1 post	.11	-.16	.30	.10	**.84**
Input6-Output1	density row 1 post	-.04	.13	-.34	-.09	-.09
Input7-Output1	length row 2 post	.23	-.20	-.33	-.02	**-.73**
Input8-Output1	density row 2 post	-.02	.11	.34	.08	.07
Input9-Output1	transformed row	.11	**-.74**	.21	-.12	-.05
Input10-Output1	add	.05	-.27	.25	**-.71**	.09
Input11-Output1	subtract	.05	.45	-.21	-.37	.05
Input12-Output1	elongate	-.19	.04	.57	**.79**	.07
Input13-Output1	compress	.12	-.22	**-.75**	.21	-.22
Hidden1-Output1		-.06	**.95**	.13	.25	-.07
Hidden2-Output1		**.97**	-.02	.15	-.15	-.05
Hidden3-Output1		**.88**	-.08	.00	-.03	.04
Hidden4-Output1		-.05	-.37	**.81**	-.05	-.31
Hidden5-Output1		.28	-.30	.10	-.33	.34
Bias-Output2		.00	.00	.00	.00	.00
Input1-Output2	length row 1 pre	-.09	-.03	-.07	.14	**-.92**
Input2-Output2	density row 1 pre	-.06	.05	-.16	.01	-.04
Input3-Output2	length row 2 pre	-.06	.02	.06	-.13	**.92**
Input4-Output2	density row 2 pre	.06	-.05	.16	-.01	.04
Input5-Output2	length row 1 post	-.11	.16	-.30	-.09	**-.84**
Input6-Output2	density row 1 post	.03	-.13	.34	.09	.09
Input7-Output2	length row 2 post	-.22	.18	.33	.00	**.73**
Input8-Output2	density row 2 post	-.02	.10	.34	.08	.08
Input9-Output2	transformed row	-.11	**.74**	-.21	.12	.05
Input10-Output2	add	.05	-.27	.25	**-.71**	.09
Input11-Output2	subtract	.05	.45	-.21	-.37	.05
Input12-Output2	elongate	-.19	.04	.57	**.79**	.07
Input13-Output2	compress	.12	-.22	**-.75**	.21	-.23
Hidden1-Output2		.06	**-.95**	-.13	-.25	.07
Hidden2-Output2		**-.97**	.03	-.15	.15	.05
Hidden3-Output2		**-.88**	.08	-.00	.03	-.04
Hidden4-Output2		.05	.37	**-.81**	.05	.31
Hidden5-Output2		-.28	.30	-.10	.33	-.34

computational power to the network, but it seems that for radical changes to occur there must be a good match between the network's current status and the error detection abilities of the new hidden unit. Not all hidden units have this dramatic impact. The final few hidden units typically do not alter knowledge representations in any fundamental way, perhaps merely refining representations established earlier. This coincides with the leveling off in performance curves as conservation performance reaches asymptote (see Figures 1 and 2). The transition from perceptual to cognitive solutions in networks mirrors Piaget's account of conservation acquisition in children, conforms to evidence that understanding transformations is critical in conservation acquisition (Halford & Boyle, 1985; Siegler, 1981), and suggests concrete ways in which such knowledge can be learned, represented, and integrated.

This interpretation assumes that both perception and reasoning are implemented in the brain by the firing of neurons that alter the momentary levels of neuronal activations. Perception and reasoning may well employ different neuronal circuits, but one way they surely differ is in the nature of their inputs. Perception deals with inputs reflecting the way things look, sound, feel, etc., whereas cognition deals at least in part with various actual or hypothetical transformations that can be applied to objects.

Summary of Simulations

These simulations capture a variety of well-replicated effects in the conservation literature: acquisition, including learning, generalization, and sudden jumps; the problem size effect; length bias; and the screening effect. The simulations suggest a novel explanation of sudden jumps in conservation performance in terms of the recruitment of new hidden units that represent transformation information. In addition, the simulations suggest a novel explanation of the problem size effect, based on the analog nature of number representation, rather than practice or estimation accuracy. Moreover, the simulations provide support for the correlation-learning explanation of length bias. Length correlates fairly well with number because density is typically held constant during addition and subtraction transformations. These findings constitute better and more comprehensive coverage of natural conservation phenomena than has been achieved in previous computational approaches to conservation, although admittedly such simulations are still in their infancy.

Uncaptured Phenomena

There are, of course, many other conservation phenomena in a literature with over 1000 publications, far too many to cover in a single paper. A few of the well known conservation phenomena that so far escape simulation attempts concern estimation techniques, justifications, pragmatic effects, training studies, readiness to learn, the discrete advantage effect, and early additive rules for integrating length and density.

Estimation Techniques

The principal estimation techniques used by children are subitizing, counting, and one-to-one correspondence (Chi & Klahr, 1975; Fuson, Secada, & Hall, 1983; Gelman & Gallistel, 1978), none of which are directly simulated in the present model. An assumption of the model is that some method of estimating quantities is available to provide the information required for target comparisons of *greater than*, *less than*, or *equal to* (Klahr & Wallace, 1976; K. F. Miller, 1989; Sophian, 1995). A more complete simulation would implement one or more of these estimation techniques, rather than simply assume their presence. It might be possible to implement connectionist models of estimation techniques and integrate them into equivalence conservation models like this one.

Justifications

In addition to obtaining conservation judgments from children, Piaget (1965) asked children to justify their judgments. Indeed, Piaget and other European researchers were typically more interested in children's verbal justifications than in their judgments. Justifications for correct conservation judgments often focus on issues of identity ("It is the same water"), reversibility ("If I squeeze the pennies together, it will be the same as before"), or compensation ("It is longer, but farther apart").

Because verbal justifications usually lag behind judgments, North American researchers typically favor using judgments rather than justifications (Brainerd, 1973). They argue that inadequate justifications might reflect inability to justify, rather than inability to reason. In any case, present computational models, lacking language capabilities, do not capture justifications.

Pragmatic Effects

Another conservation phenomenon, concerning pragmatic effects, also escapes simulation in part because of a lack of language capabilities in neural networks. There is some psychological evidence that nonconservation results from the conversational rules of the typical conservation experiment. In most conservation experiments, the child is asked the same question about the equality of two quantities twice, once before and once after the transformation. If children assume that the transformation conserves the initial equality, they may view repetition of the question as pragmatically odd. They might explain this oddity by inferring that the experimenter wants to know about changes in the display rather than about numerical equality. Such reasoning could produce incorrect nonconservation responses. Support for this idea comes from studies that omit the pre-transformation question (Rose & Blank, 1974) or use accidental, rather than deliberate, transformations (McGarrigle & Donaldson, 1974). In both cases, the procedural modification produced more correct conservation responses than did the standard technique.

However, subsequent research casts doubt on the robustness of these results. The question omission result held for 6-year-olds, but not for 4.5-year-olds (Neilson, Dockrell, & McKechnie, 1983). And the accidental transformation effect held for small, but not for large, arrays, and for elongation, but not for addition, transformations (Moore & Frye, 1986). Because these pragmatic effects are not robust, it might be unwise for computational modelers to try to simulate them. Or, if simulations are attempted, all of the various nuances in the data should probably be accommodated. In any case, the learning required for the relevant pragmatic knowledge would be far afield from the essentials of conservation itself.

Training Studies

Although the present simulations focus on natural conservation acquisition, there is a vast sub-literature on the training of conservation in children. Typically, training techniques emphasize one of the three principles that children mention in their justifications of conservation judgments, namely identity, reversibility, or compensation. Identity training tries to convince the child that a quantity has not changed over a number preserving transformation. Reversibility training emphasizes that the effects of a transformation can be undone with no change in quantity. Compensation training focuses on the fact that, with number preserving transformations, changes in length and density compensate for each other. Empirical support has been found for each sort of training (Brainerd & Allen, 1971; Curcio et al., 1972; Hamel & Riksen, 1973; Sheppard, 1974). Simulation of such training studies should not pose insurmountable problems for the present network model. Identity and reversibility training could be regarded as providing evidence that particular transformations such as elongation and compression conserve number and thus equivalence. Analogously, compensation training could be regarded as providing evidence that changes in length or density compensate for each other in a multiplicative fashion. Because number = length x density, and number is held constant, increases in length are compensated for by decreases in density, and vice versa. Thus, traditional training techniques could be regarded as implementing parts of the conservation training received by the present networks.

Readiness to Learn

It has been demonstrated, both in conservation acquisition (Curcio et al., 1972) and in other cognitive domains (Siegler, 1976) that training is more effective when children possess knowledge that renders them ready to be trained. McClelland (1995) showed that such readiness effects in the balance scale literature can be simulated by training networks either a little or a lot, and then providing equal amounts of new training. Perhaps not surprisingly, networks with lots of previous training learned more from the new training than did networks with only little previous training. The same kind of simulation would likely capture readiness effects in conservation acquisition.

The Discrete Advantage Effect

The discrete advantage effect refers to the finding that children conserve discrete quantities before continuous quantities (Elkind & Schoenfeld, 1972; Siegler, 1981). It has been assumed that such an effect occurs because it is easier to estimate discrete quantities than continuous quantities (Simon & Klahr, 1995). The principal numeric estimation techniques that have been attributed to children are subitizing, counting, and setting items into one-one correspondence (Chi & Klahr, 1975; Fuson et al., 1983; Gelman & Gallistel, 1978; Sophian, 1995). All of these estimation techniques can be more accurately applied to discrete quantities than to continuous quantities.

It might be possible to simulate continuous quantities in the present model by randomizing target outputs in the equivalence training and test patterns by a small amount. That is, output targets for discrete quantities could be represented accurately, but those for continuous quantities could be represented with some degree of error. To capture the discrete advantage effect in this way, a computational system would have to be sensitive to noise, but not fatally so. That is, noise should make learning more difficult, but not impossible. Even with some noise, the system should have the ability to abstract fundamental regularities from the training problems. Connectionist networks have this property (Hertz et al., 1991) and thus might be able to simulate the discrete advantage effect.

Another difference between continuous and discrete quantities is that continuous quantities are often quantified as volumes rather than as countable integers. This suggests that the underlying functions relating continuous quantity to dimensions like length and diameter might be more complex than the relatively simple number = length x density that applies to rows of discrete items. For example, the amount of a cylindrical shaped quantity might be represented as ð x radius^2 x length. Perhaps such function differences could be exploited in simulations of the discrete advantage effect.

Early Additive Rules for Integrating Length and Density

An early stage of numerical estimation of rows has been reported in which children view number as the sum, rather than the product, of length and density: number = length + density (Cuneo, 1982). This research requires the child to give a quantitative estimate of a row, instead of a qualitative conservation judgment. On such tasks, as children develop, the additive rule gives way to the definitive multiplicative rule, number = length x density. Similar developmental trends from additive to multiplicative rules have been reported in a number of other areas such as the integration of velocity, time, and distance cues, where velocity = distance / time (Wilkening, 1981).

Such stage transitions have been successfully simulated by cascade-correlation networks (Buckingham & Shultz, 1994). What is critical in these simulations is the gradual increase in computational power achieved in generative networks. Static back-propagation networks, for example, are unable to capture the full range of stages in velocity, time, and distance integration (Buckingham & Shultz, 1995). They are either too powerful and miss the additive stage, or too weak to reach the final multiplicative stage.

In summary, phenomena such as justifications or pragmatic effects would be relatively difficult to capture with connectionist techniques because of their heavy reliance on language capacity and background knowledge. But other conservation phenomena, such as estimation, training, readiness to learn, the discrete advantage effect, and early additive rules ought to be feasible for cascade-correlation models.

General Discussion

The present results indicate that cascade-correlation networks trained on conservation of equivalence judgments can simulate a variety of basic psychological phenomena in the conservation acquisition literature. This is the first computational model to simulate a range of conservation phenomena and to simulate any conservation phenomenon without extensive domain-specific knowledge having been built in. Coupled with other successful cascade-correlation simulations, these results suggest a theory of cognitive development and begin to answer some traditional developmental puzzles.

Suggestions for a Theory of Cognitive Development

Contemporary standards for cognitive theory require specification of how knowledge is both represented and processed (Thagard, 1996). Adding a developmental perspective requires further specification of how the knowledge representations and processes change with development. Issues of knowledge representation may be further broken down into active memory and long-term memory categories. Active memory refers to the momentary contents of consciousness or particular conclusions or judgments. Long-term memory refers to relatively permanent knowledge, typically acquired through learning.

Cascade-correlation modeling suggests that active memory knowledge is represented by transitory fluctuations in patterns of activation across units. This is roughly analogous to activity patterns in groups of neurons. The modeling suggests that long-term memory knowledge is represented by connection weights among units in the particular network module dealing with the domain of interest. This is roughly analogous to synaptic strengths that get built up and modified through learning.

Processing of active memory knowledge is achieved in the networks by feed-forward spreading activation. A pattern of activation on the input units describes a conservation problem in terms of the length and density of the rows and the particular transformation that is applied. This input representation is then transformed into a comparative judgment about the equivalence of the two rows by the spreading of activation to layers of hidden units and eventually to the output units. The spreading of activation is governed by activation functions that sum their input and pass that net input through a nonlinear squashing function. This process is roughly equivalent to the way that the content of active memory changes in brains. Processing of long-term knowledge is involved in these changes of active memory in the sense that the connection weights modulate via multiplication the activations of the sending inputs. Presumably, this roughly mirrors the way that synaptic strengths modulate neural activity in the brain.

Developmentally, there are progressive changes in both network topology and connection strengths. First, network connection weights are adjusted to reduce error at the output units. Error is essentially the discrepancy between the system's expectations and the results actually obtained. Such obtained results in the case of conservation are presumably provided by the system's own estimation techniques or by the comparative claims of other people. When such connection weight adjustments fail to solve the problem, new computational power is generated by the recruitment of new hidden units that are good at representing the network's current error. The brain analogies here are to synaptic adjustments and the creation of novel circuits through synaptogenesis.

Suggestions for a Theory of Conservation

Of more specific relevance to conservation, the present modeling suggests that quantitative representations of row length and density information are in analog form, and that this perceptual information is supplemented by information about the nature and object of the transformation, both of which are coded in some arbitrary, non-analog form. Some type of feedback is required about numerical comparisons, although it is not presently clear whether such feedback comes from the system's own estimation techniques, or from those used by other people, or from both. It is also currently unclear whether this feedback comes from conservation problems or from similar experiences that allow the system to generalize to conservation problems.

Because of correlations between number of items and row length during number altering transformations, a bias emerges toward picking the longer row as having more in equivalence preserving problems. This length bias disappears as the system comes to master conservation problems by analyzing the transformations that get applied. As this conflict between different information sources gets played out, the system shifts from perceptual to cognitive representations and solutions. In both solutions, however, the nature of the active memory processing remains the same, characterized by feed-forward spreading activation -- only the nature of input emphasized in the knowledge representation changes with development.

Many of these theoretical suggestions about conservation can be found in either the Piagetian or the subsequent conservation literature (e.g., the shift from perceptual to cognitive solutions, early length bias, the importance of feedback on number comparisons), but a few of these suggestions appear to be novel, at least as applied to conservation (e.g., the explanation of sudden jumps in terms of the recruitment of new hidden units representing transformation information, the analog representation explanation of the problem size effect, the documentation of the correlation-with-number explanation of the length bias effect, and the nature of knowledge representation and processing).

Some Traditional Puzzles about Cognitive Development

The present modeling, along with similar cascade-correlation modeling, also provides some tentative answers to some traditional puzzles about cognitive development. Foremost among these puzzles is the issue of transition mechanisms. How does cognitive developmental change occur? The modeling suggests that development results from an error reduction process characterized by quantitative adjustment of connection weights and occasional recruitment of hidden units to re-conceptualize the problem when needed. Piaget's vague, but interesting notions of assimilation and accommodation can be interpreted in terms of mathematically specified weight adjustment and unit recruitment (Shultz et al., 1995b). Accommodation can be viewed as a rather drastic change in network topology introduced by the installation of a new hidden unit that has been trained to detect the current network error. At this point, the network might be able to achieve representations that were impossible prior to recruitment. The extent to which this process can capture the phenomena of conceptual change (Carey, 1991) and representational re-description (Karmiloff-Smith, 1992) should be explored. Assimilative learning, which Piaget never fully explained, can be understood as the quantitative adjustment of connection weights in a network without a change in topology. This is gradual learning without drastic change. But, just as hidden unit recruitment does not guarantee substantial accommodation, neither does mere weight adjustment preclude substantial accommodation. In either case, knowledge representations need to be assessed directly, e.g., with contribution analysis, to determine whether knowledge changes have been qualitative or merely quantitative. Finally, pure assimilation can be viewed as more or less accurate generalization to problems the system has not been trained on, without any further learning.

Another traditional developmental puzzle concerns the representation of knowledge within a particular stage. Such representations have usually been characterized in terms of logical groupings (Piaget, 1966) or condition-action rules (Klahr, 1984). Groupings have not stimulated computational models of development, but rules certainly have (Klahr, 1984; Siegler, 1981; Simon & Klahr, 1995). For example, nonconservation can be expressed in terms of a rule such as "If two rows of objects are initially equal, and one row is now longer than the other, then pick the longer row as having more than the other." If the conditions of this rule match those of the problem, this rule might be selected and used, leading to a nonconservation judgment. Such computational approaches are typically described as serial and symbolic; *serial* because rules are selected and used one at a time and are selected by examining their conditions one at a time, and *symbolic* because the parts of a rule refer to particular objects or events in the world.

In contrast, connectionist approaches are typically described as parallel and sub-symbolic; *parallel* because unit activations are updated simultaneously and *sub-symbolic* because a given unit may symbolize nothing in particular but instead typically participates in the representation of many different objects and events (Smolensky, 1988). Rulelike behavior is said to emerge from the sub-symbolic parallel processes of neural networks when they learn to encode the regularities in their environment, as conveyed by their training patterns, and respond accordingly. Rules are not represented explicitly anywhere in the network, but the network may come to behave as if it were following rules.[7] Of course, the same may be true of children. In other words, rules may be more epiphenomena than computational mechanisms.

Another developmental puzzle, not so often recognized, is the existence of a variety of perceptual effects in cognitive tasks (P. H. Miller, 1978). An example for conservation is the problem size effect, small number problems being easier to solve than large number problems. Other perceptual examples treated in the present simulations are the length bias and screening effects. It has always been unclear how such perceptual effects can be integrated with cognitive phenomena that are often of more central interest. How could rules like "pick the longer row" or "pick the more numerous row", for example, simulate the problem size effect? Or, why should children show length bias, rather than density bias? Many researchers view perception as inherently different from cognition, and this leads to the use of auxiliary notions such as salience and attention to account for perceptual effects. Probably this practice has been encouraged by the distinction between competence and performance that has become common in psycholinguistics.

However, even if one separates perception and cognition in this fashion, it is still unclear how these distinct systems might interact to produce coherent behavior. One suggestion emerging from neural network simulations is that both processes exist together within homogenous networks. Perceptual effects like those simulated here fall naturally out of the continuous nature of network computations, in which unit activations vary continuously and inputs to units are summed across sending connections. Larger differences on the inputs create clearer signals and decisions downstream in the network. Because of the way that downstream units integrate input from a variety of sources, perceptual and cognitive information can interact in various ways. In many cases, they might support each other; in others cases, they might compete.

Conservation phenomena are such that perception sometimes conflicts with cognition. The child may know that elongating a row does not change its number, but notice that the longer row now appears more numerous. It is difficult to conceive how this emerging tension between perception and cognition could be modeled in rule-based computation, except perhaps in an ad hoc fashion. Conversely, the competition between, and integration of, perceptual and cognitive solutions occurs quite naturally in neural networks.

Experience and Conservation

Perhaps the most controversial feature of this conservation model is a training procedure that provides networks with target activation patterns concerning row comparisons. The conservation literature is quite silent about the necessary and sufficient experiences for conservation acquisition in the child's natural world. Indeed, some researchers have become quite pessimistic about the possibility of psychological research uncovering the conditions for natural conservation acquisition. "It is not likely that the . . . question can ever receive a direct and precise answer because of the intrinsic difficulty of sorting out the myriad uncontrolled environmental presses to which the child is exposed (Flavell, 1963, p. 377)."

In the absence of constraints supplied by the literature, the networks were trained with the fewest possible environmental constraints. Consequently, the training patterns included both

---

[7]Although not diagnosed in the present simulations, rules have been diagnosed in other network simulations (Buckingham & Shultz, 1994; Mareschal & Shultz, 1993; McClelland, 1995; Shultz et al., 1994a; Shultz et al., 1994b).

small and large number problems, transformations that change number and those that preserve number, and equal and unequal initial rows. All these possibilities occurred and with no particular bias.[8] Hints supplied by the conservation training literature about relevant experience suggest that target feedback on row comparisons is indeed helpful to children (Brainerd & Allen, 1971; Curcio et al., 1972; Hamel & Riksen, 1973; Sheppard, 1974). In this context, it is interesting that networks trained in this fashion exhibit many of the regularities found with children. This is further evidence that such target feedback can yield the requisite phenomena. Such feedback could come from other people who tell children correct comparison answers, or it could derive from children's own estimation techniques. Whether other experiences could also produce these phenomena in networks or other computational systems remains an open question. The current psychological literature does not appear to provide evidence for critical conservation experiences other than feedback on conservation problems.

It is noteworthy that almost all of the phenomena covered here were simulated on test problems, that is, problems that the network had not been trained on. This ensures that the coverage is not entirely dependent on the precise training problems used.

Final Word

The present simulations are not an implementation of Piagetian theory or any other previous model of conservation acquisition. Rather the results show that many conservation phenomena can be simulated from quite a different theoretical perspective, that of the brain-style computation of artificial neural networks. Simulated conservation phenomena arise from the domain-general properties of the cascade-correlation algorithm and the nonlinear nature of a suitably wide range of conservation problems. These domain-general network properties include graded, distributed representations, activation passing, connection weight adjustment, and hidden unit recruitment.

References

Anderson, J. A. (1995). *An introduction to neural networks.* Cambridge, MA: MIT Press.

Anderson, J. R. (1993). *Rules of the mind*. Hillsdale, NJ: Erlbaum.

Ashcraft, M. H. (1992). Cognitive arithmetic: A review of data and theory. *Cognition, 44*, 75-106.

Bates, E. A., & Elman, J. L. (1993). Connectionism and the study of change. In M. H. Johnson (Ed.), *Brain development and cognition* (pp. 623-642). Oxford: Blackwell.

Bochereau, L., & Bourgine, P. (1990). Extraction of semantic features and logical rules from a multilayer neural network. *Proceedings of the IJCNN, 2*, 579-582.

Brainerd, C. J. (1973). Judgments and explanations as criteria for the presence of cognitive structures. *Psychological Bulletin, 79*, 172-179.

Brainerd, C. J., & Allen, T. W. (1971). Experimental inductions of the conservation of "first order" quantitative invariants. *Psychological Bulletin, 75*, 128-144.

[8] The one environmental bias that was important favored constant density, rather than constant length, during number altering transformations. This is critical in simulating length, as opposed to density, bias in conservation responses, and conforms to the practice of most conservation experiments using addition and subtraction transformations. Density is typically left alone when items are added or subtracted.

Bruner, J. S., Olver, R. R., & Greenfield, P. M. (1966). *Studies in cognitive growth*. New York: Wiley.

Bryant, P. E. (1972). The understanding of invariance by very young children. *Canadian Journal of Psychology, 26,* 78-96.

Buckingham, D., & Shultz, T. R. (1994). A connectionist model of the development of velocity, time, and distance concepts. *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society* (pp. 72-77). Hillsdale, NJ: Erlbaum.

Buckingham, D., & Shultz, T. R. (1995). *Computational power and realistic cognitive development*. Technical Report No. 1995, McGill Papers in Cognitive Science, McGill University, Montreal.

Carey, S. (1991). Knowledge acquisition: Enrichment or conceptual change. In S. Carey & R. Gelman (Eds.), *The epigenesis of mind: Essays on biology and cognition* (pp. 257-291). Hillsdale, NJ: Erlbaum.

Cattell, R. B. (1966). The scree test for the number of factors. *Multivariate Behavioral Research, 1*, 245-276.

Chi, M. T. H., & Klahr, D. (1975). Span and rate of apprehension in children and adults. *Journal of Experimental Child Psychology, 19*, 434-439.

Cowan, R. (1979a). Performance in number conservation tasks as a function of the number of items. *British Journal of Psychology, 70*, 77-81.

Cowan, R. (1979b). A reappraisal of the relation between performances of quantitative identity and quantitative equivalence conservation tasks. *Journal of Experimental Child Psychology, 28*, 68-80.

Crick, F. H. C. & Asanuma, C. (1986). Certain aspects of the anatomy and physiology of the cerebral cortex. In J. L. McClelland & D. E. Rumelhart (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition, Volume 2: Psychological and Biological Models* (pp. 333-371). Cambridge, MA: MIT Press.

Cuneo, D. O. (1982). Children's judgments of numerical quantity: A new view of early quantification. *Cognitive Psychology, 14*, 13-44.

Curcio, F., Kattef, E., Levine, D., & Robbins, O. (1972). Compensation and susceptibility to conservation training. *Developmental Psychology, 7*, 259-265.

Elkind, D., & Schoenfeld, E. (1972). Identity and equivalence conservation at two age levels. *Developmental Psychology, 6*, 529-533.

Elman, J. L. (1989). *Representation and structure in connectionist models*. CRL Technical Report 8903, Center for Research in Language, University of California at San Diego.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science, 14,* 179-211.

Elman, J. L., Bates, E. A., Johnson, M. H., Karmiloff-Smith, A., Parisi, D., & Plunkett, K. (1996). *Rethinking innateness: A connectionist perspective on development*. Cambridge, MA: MIT Press.

Fahlman, S. E. (1991). *Common Lisp implementation of cascade-correlation learning algorithm* [Computer program]. Pittsburgh, PA: Carnegie Mellon University, School of Computer Science.

Fahlman, S. E., & Lebiere, C. (1990). The cascade-correlation learning architecture. In D. S. Touretzky (Ed.), *Advances in Neural Information Processing Systems 2* (pp. 524-532). Los Altos, CA: Morgan Kaufmann.

Flavell, J. H. (1963). *The developmental psychology of Jean Piaget*. New York: van Nostrand.

Fuson, K. C., Secada, W. G., & Hall, J. W. (1983). Matching, counting, and conservation of numerical equivalence. *Child Development, 54*, 91-97.

Gelman, R., & Gallistel, R. C. (1978). *The child's understanding of number*. Cambridge, MA: Harvard University Press.

Greenough, W. T., & Bailey, C. H. (1988). The anatomy of memory: Convergence of results across a diversity of tests. *Trends in Neuroscience, 11*, 142-147.

Greenough, W. T., Withers, G. S., & Anderson, B. J. (1992). *Experience-dependent synaptogenesis as a plausible memory mechanism*. Hillsdale, NJ: Erlbaum.

Halford, G. S., & Boyle, F. M. (1985). Do young children understand conservation of number? *Child Development, 56*, 165-176.

Hamel, B. R., & Riksen, B. O. M. (1973). Identity, reversibility, verbal rule instruction, and conservation. *Developmental Psychology, 9*, 66-72.

Hashmi, Y., & Shultz, T. R. (1997). *A connectionist model of number comparison*. Manuscript submitted for publication.

Hayashi, Y. (1991). A neural expert system with automated extraction of fuzzy if-then rules and its application to medical diagnosis. *Advances in Neural Information Processing Systems 3* (pp. 578-584). San Francisco, CA: Morgan Kaufmann.

Hertz, J., Krogh, A., & Palmer, R. G. (1991). *Introduction to the theory of neural computation*. Reading, MA: Addison Wesley.

Hinton, G. E., & Sejnowski, T. J. (1986). Learning and relearning in Boltzmann machines. In D. E. Rumelhart & J. L. McClelland (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition, Volume 1: Foundations* (pp. 282-317). Cambridge, MA: MIT Press.

Karmiloff-Smith, A. (1992). *Beyond modularity.* Cambridge, MA: MIT Press.

Klahr, D. (1984). Transition processes in quantitative development. In R. J. Sternberg (Ed.), *Mechanisms of cognitive development* (pp. 101-139). New York: Freeman.

Klahr, D., & Wallace, J. G. (1976). *Cognitive development: An information processing view*. Hillsdale, NJ: Erlbaum.

Mareschal, D., & Shultz, T R. (1993). A connectionist model of the development of seriation. *Proceedings of the Fifteenth Annual Conference of the Cognitive Science Society* (pp. 676-681). Hillsdale, NJ: Erlbaum.

Mareschal, D., & Shultz, T. R. (1996). Generative connectionist networks and constructivist cognitive development. *Cognitive Development, 11*, 571-603.

McClelland, J. L. (1995). A connectionist perspective on knowledge and development. In T. J. Simon & G. S. Halford (Eds.), *Developing cognitive competence: New approaches to process modeling* (pp. 157-204). Hillsdale, NJ: Erlbaum.

McGarrigle, J., & Donaldson, M. (1974). Conservation accidents. *Cognition, 3,* 341-344.

Miller, K. F. (1989). Measurement as a tool for thought: The role of measuring procedures in children's understanding of quantitative invariance. *Developmental Psychology, 25*, 589-600.

Miller, P. H. (1973). Attention to stimulus dimensions in the conservation of liquid quantities. *Child Development*, *44*, 129-136.

Miller, P. H. (1978). Stimulus variables in conservation. *Merrill-Palmer Quarterly, 24,* 141-159.

Miller, P. H., Grabowski, T. L., & Heldmeyer, K. H. (1973). The role of stimulus dimensions in the conservation of substance. *Child Development, 44*, 646-650.

Miller, P. H., & Heldmeyer, K. H. (1975). Perceptual information in conservation: Effects of screening. *Child Development, 46*, 588-592.

Miller, P. H., & Heller, K. A. (1976). Facilitation of attention to number and conservation of number. *Journal of Experimental Child Psychology, 22*, 454-467.

Moore, C., & Frye, D. (1986). The effect of the experimenter's intention on the child's understanding of conservation. *Cognition*, *22*, 283-298.

Moyer, R. S., & Landauer, T. K. (1967). Time required for judgements of numerical inequality. *Nature, 215*, 1519-1520.

Neilson, I., Dockrell, J., & McKechnie, J. (1983). Does repetition of the question influence children's performance in conservation tasks? *British Journal of Developmental Psychology, 1*, 163-174.

Newell, A. (1990). *Unified theories of cognition*. Cambridge, MA: Harvard University Press.

Piaget, J. (1965). *The child's conception of number*. New York: Norton.

Piaget, J. (1966). *Psychology of intelligence*. Totowa, NJ: Littlefield, Adams.

Piaget, J., & Inhelder, B. (1971). *Mental imagery in the child*. New York: Basic Books.

Plunkett, K., & Sinha, C. (1992). Connectionism and developmental theory. *British Journal of Developmental Psychology, 10,* 209-254.

Quartz, S. R. (1993). Neural networks, nativism, and the plausibility of constructivism. *Cognition, 48*, 223-242.

Quartz , S. R., & Sejnowski, T. J. (In press). The neural basis of cognitive development: A constructivist manifesto. *Behavioral and Brain Sciences*.

Raijmakers, M. E. J., van Koten, S., & Molenaar, P. C. M. (1996). On the validity of simulating stagewise cognitive development by means of PDP networks: Application of catastrophe analysis and an experimental test of rule-like network performance. *Cognitive Science, 20,* 101-136.

Rose, S. A., & Blank, M. (1974). The potency of context in children's cognition: An illustration through conservation. *Child Development, 45*, 499-502.

Sekuler, R., & Mierkiewicz, D. (1977). Children's judgment of numerical inequality. *Child Development*, *48*, 630-633.

Sheppard, J. L. (1974). Compensation and combinatorial systems in the acquisition and generalization of conservation. *Child Development, 45*, 717-730.

Shultz, T. R., Buckingham, D., & Oshima-Takane, Y. (1994a). A connectionist model of the learning of personal pronouns in English. In S. J. Hanson, T. Petsche, M. Kearns, & R. L. Rivest (Eds.), *Computational learning theory and natural learning systems, Vol. 2: Intersection between theory and experiment* (pp. 347-362). Cambridge, MA: MIT Press.

Shultz, T. R., Dover, A., & Amsel, E. (1979). The logical and empirical bases of conservation judgments. *Cognition, 7*, 99-123.

Shultz, T. R., Mareschal, D., & Schmidt, W. C. (1994b). Modeling cognitive development on balance scale phenomena. *Machine Learning, 16*, 57-86.

Shultz, T. R., Oshima-Takane, Y., & Takane, Y. (1995a). Analysis of unstandardized contributions in cross connected networks. In D. Touretzky, G. Tesauro, & T. K. Leen, (Eds). *Advances in Neural Information Processing Systems 7* (pp. 601-608). Cambridge, MA: MIT Press.

Shultz, T. R., Schmidt, W. C., Buckingham, D., & Mareschal, D. (1995b). Modeling cognitive development with a generative connectionist algorithm. In T. J. Simon & G. S. Halford (Eds.), *Developing cognitive competence: New approaches to process modeling* (pp. 205-261). Hillsdale, NJ: Erlbaum.

Siegler, R. S. (1976). Three aspects of cognitive development. *Cognitive Psychology, 8*, 481-520.

Siegler, R. S. (1981). Developmental sequences within and between concepts. *Monographs of the Society for Research in Child Development, 46*, Serial No. 189.

Siegler, R. S. (1995). How does change occur: A microgenetic study of number conservation. *Cognitive Psychology, 28*, 225-273.

Siegler, R. S., & Robinson, M. (1982). The development of numerical understandings. *Advances in Child Development and Behavior, 16*, 241-312.

Silverman, I. W., & Briga, J. (1981). By what process do young children solve small number conservation problems? *Journal of Experimental Child Psychology, 32*, 115-126.

Simon, T. J., & Klahr, D. (1995). A computational theory of children's learning about number conservation. In T. J. Simon & G. S. Halford (Eds.), *Developing cognitive competence: New approaches to process modeling* (pp. 315-353). Hillsdale, NJ: Erlbaum.

Smolensky, P. (1988). On the proper treatment of connectionism. *Behavioral and Brain Sciences*, *11*, 1-74.

Sophian, C. (1995). Representation and reasoning in early numerical development: Counting, conservation, and comparison between sets. *Child Development*, *66*, 559-577.

Tabachnick, B. G., & Fidell, L. S. (1983). *Using multivariate statistics*. New York: Harper & Row.

Takane, Y., Oshima-Takane, Y., & Shultz, T. R. (1994). Approximations of nonlinear functions by feed-forward networks. *Proceedings of the 11th Annual Meeting of the Japan Classification Society* (pp. 26-33). Tokyo: Japan Classification Society.

Thagard, P. (1996). *Mind: Introduction to cognitive science*. Cambridge, MA: MIT Press.

Thompson, R. F. (1986). The neurobiology of learning and memory. *Science, 223*, 941-947.

Wilkening, F. (1981). Integrating velocity, time, and distance information: A developmental study. *Cognitive Psychology, 13*, 231-247.

van der Maas, H. (1993). *Catastrophe analysis of stagewise cognitive development*. Doctoral dissertation, University of Amsterdam, Amsterdam.

Winer, B. J. (1962). *Statistical principles in experimental design*. New York: McGraw-Hill.

Winer, G. A. (1974). Conservation of different quantities among preschool children. *Child Development, 45*, 839-842.

Yeckel, M. F., & Berger, T. W. (1990). Feedforward excitation of the hippocampus by afferents from the entorhinal cortex: Redefinition of the role of the trisynaptic pathway. *Proceedings of the National Academy of Sciences, U. S. A. 87*, 5832-5836.

Author Note

Please address correspondence to Thomas R. Shultz, Department of Psychology, McGill University, 1205 Penfield Ave., Montreal, Quebec, Canada H3A 1B1. E-mail: shultz@psych.mcgill.ca

This research was supported by a grant from the Natural Sciences and Engineering Research Council of Canada. Thanks to Yasser Hashmi, Yuriko Oshima-Takane, Sylvain Sirois, Sheldon Tetewsky, Adam Waese, and two anonymous reviewers for helpful comments on earlier drafts.

This paper was published as Shultz, T. R. (1998). A computational analysis of conservation. *Developmental Science, 1*, 103-126.