Control of Attention in Neural Networks

Eric Mjolsness

Research Report YALEU/DCS/RR-545
June 1987

Control of Attention in Neural Networks

Eric Mjolsness
Computer Science Department
Yale University
New Haven, CT 06520

Abstract

"Attention" – the sequential selection of portions of a large computation to be performed now, later, or not at all – is important in the study of neural networks and in other areas of computing as well. A mechanism for controlling attention which may be applied to any network governed by a minimization principle is proposed here. Experiments with Hopfield and Tank's network for the Traveling Salesman Problem show that one may partially serialize the computation and suffer only a minor degradation of solution quality. As a result, a new kind of interaction between neural networks is obtained in which one network controls the second network's sequencing. The new, controlling network is also governed by a minimization principle. One might expect considerable savings in the cost of computation to be achieved by such an attention control mechanism, but no savings is demonstrated in the Traveling Salesman network. Finally, the method is related to standard optimization methods such as Newton's method and the conjugate gradient method, and some advantages of "windowed" attention are discussed.

1 Introduction

Neural networks are generally thought of as a purely and radically parallel approach to computation, but on occasion moderation may be called for in parallel computation. For example it may be necessary to spend a long time serially scanning input to and output from a network on a silicon chip. To balance costs between input, output and computation the optimal network would take a similarly long time to compute its output while using the chip in a parallel manner. In other words, the optimal network would have an intermediate degree of parallelism.

We are led to consider the partial serialization of neural networks. An advantage of serialization is that one can often use previous results to eliminate the need for computations which would be unavoidable in a maximally parallel scheme; the combination of partial serialization and opportunism in a neural network may be referred to as "attention". For other discussions of attention in neural networks see ([1],[2]). We will propose a mechanism for attention applicable to a large class of neural networks: the class of networks governed by an objective function or energy function which steadily decreases as the network operates. An example is the Traveling Salesman network of Hopfield and Tank [3].

In the course of minimizing an objective function which depends on all the neuron values, many neurons may not significantly participate in the network dynamics for long periods of time. For example, in the traveling salesman network each neuron corresponds to a hypothesized assignment of a city to a tour position, and many such assignments would introduce such large distances into the total tour length that they are completely incompatible with the rest of the current configuration. These neurons will turn off and stay off until the network configuration changes a great deal. Such a slowly varying neuron could be left out of a network simulation for a while, saving on computing costs. Simulating the slow neuron would simply require that its value be stored for future use. Even

this storage requirement can be eliminated if slow neurons have predictable values such as "all off"; then we have a "virtual neuron" which is created when needed and destroyed when no longer needed.

We shall be concerned with controlling the attentional transition between slow and fast neurons, rather than with the equally important circuits or software needed to perform the transition in a simulator. The result is a pair of dynamical systems similar to neural networks, one controlling the sequencing of the other. The mechanism we will propose could equally be applied to neuron-like learning dynamics for synapses such as Lapedes's master/slave network [4] .

The particular form of the TSP network we use is [3]

$$V_{ab} = g(u_{ab}) \qquad \dot{u}_{ab} = -r_{ab} dE/dV_{ab} \tag{1}$$

where V_{ab} is the output of the neuron indexed by a city a and a day (tour position) b; each V_{ab} represents a hypothesised assignment of a city to a day. u_{ab} is a time-integrated input to the same neuron. Also

$$E(V) = A \sum_{abc} D_{ac} V_{ab} (V_{cb+1} + V_{cb-1}) + \frac{B}{2} \sum_a (\sum_b V_{ab} - 1)^2 + \frac{B}{2} \sum_b (\sum_a V_{ab} - 1)^2$$

$$- C \sum_{ab} V_{ab}^2 + \sum_{ab} \int^{V_{ab}} g^{-1}(x) dx. \tag{2}$$

Here D_{ac} is the distance between cities a and c which are placed randomly on the unit square, so that the first term of E measures the tour length to be minimized. The last term insures that each neuron output lies between 0 and 1, and it uses a sigmoidal gain function $g(u)$ whose slope at the origin is the gain g_0. The terms with coefficient B insure a one-to-one match between cities and days in the tour and the term with coefficient C discourages intermediate values of V_{ab}.

This network has n cities, n^2 neurons and n^3 interconnections in an efficient implementation. Ordinarily the rate parameters r_{ab} are all equal and constant in time and may be set to 1; we will selectively disable a neuron by diminishing or zeroing its r_{ab} (thus the neuron moves even more slowly than it would for $r_{ab} = 1$) so the r_{ab} become unequal and dynamic.

2 Theory

If

$$V_i = g(u_i) \quad \text{and} \quad \dot{u}_i = -r_i \frac{\partial E}{\partial V_i} \tag{3}$$

describes the neural network dynamics of N neurons, the energy function E decreases even if r_i is a function i and of time:

$$\frac{dE(V)}{dt} = \sum_i \frac{\partial E}{\partial V_i} \frac{dV_i}{dt} = -\sum_i r_i(t) g'(u_i) (\frac{\partial E}{\partial V_i})^2 \leq 0$$

assuming that the gain function g is monotonically increasing and that $r_i \geq 0$. We would like to control r_i by a similar dynamical system

$$r_i = g(q_i) \quad \text{and} \quad \dot{q}_i = -r_i \rho \frac{\partial E_{\text{focus}}}{\partial r_i} \tag{4}$$

where the new objective function $E_{\text{focus}}(r, V)$ controls the "focus of attention" and is yet to be determined. r_i will be restricted to $(0, 1)$ by g as is V_i, and ρ serves as a relative rate parameter for $\dot{u}_i$ and $\dot{q}_i$. One problem with (3) is evident already: r_i can become zero, and the closer to zero it

gets the longer it will take to increase again if needed. Truly modeling a disabled neuron v_i would require $r_i = 0$. One cannot have

$$\dot{q}_i = -\rho \frac{\partial E_f}{\partial r_i} \tag{5}$$

for cost reasons: a disabled neuron V_i would be about as expensive to simulate as any neuron in the usual TSP network because r_i and q_i would need to be simulated at full speed. A resolution to this problem using "windowed" attention is suggested later.

In the meantime we will use equation 4 and interpret r_i as something like a time-averaged update rate in a discrete ordinary differential equation (ODE) solver. If the solver has independent adaptive time step sizes for the different neurons, it can control the rate at which different neurons are updated. Under this interpretation, small r_i implies less computational cost. But contrary to this interpretation, r_i also influences the direction of movement in the energy landscape.

The benefit of a large r_i ($r_i = 1$) is that the system loses energy quickly. This suggests minimizing

$$E_{f,\text{benefit}} = -\hat{A}\frac{dE}{dt} = -\sum_i r_i h(V_i). \tag{6}$$

The cost of a large r_i depends on the implementation of the attention mechanism, of which there may be many. If a out of n^2 neurons are allowed to be fully attended to ($r_i = 1$) at one time, a simplified model cost is

$$E_{f,\text{cost}} = \frac{\hat{B}}{2}(\sum_i r_i - a)^2. \tag{7}$$

More accurate cost expressions will also be considered, but this one was selected for the experiments. To confine r_i to (0,1) and discourage intermediate values of r_i, we also use

$$E_{f,\text{potential}} = -\sum_i \int^{r_i} \hat{g}^{-1}(x)dx + \hat{C}\sum_i r_i^2. \tag{8}$$

Finally we propose

$$E_f(r,V) = E_{f,\text{benefit}}(r,V) + E_{f,\text{cost}}(r) + E_{f,\text{potential}}(r) \tag{9}$$

which has the form of a neural network energy function. If $\dot{V} \approx 0$, which is *not* usually the case, then

$$\frac{dE_f}{dt} = -\sum_i r_i \hat{g}'(q_i)(\frac{\partial E_f}{\partial r_i})^2 \leq 0.$$

Costs

A conservative way to measure the cost of simulating equations 1 and 2 is to estimate the number of updates required by a first-order ODE solver (the forward Euler method) with different adaptive time steps for different neurons. In fact, direct implementation as an analog op-amp circuit (as in [3]) appears to be similar to a first-order ODE solver with fixed time step. For adaptive time step $\Delta t_i(t)$,

Cost = total number of updates, summed over time and space

$$\text{Cost} = \sum_{i=1}^{N} \int_0^\infty \frac{dt}{\Delta t_i(t)}. \tag{10}$$

Let r_i control the relative update frequencies

$$\frac{\Delta t_i}{\Delta t_j} = \frac{r_j}{r_j} \quad (\text{so} \quad \Delta t_i = \frac{1}{r_i}\frac{\sum r_j}{N}\Delta t)$$

and let the absolute update frequency or time step Δt be determined by the accuracy criterion that each neuron's error per update step (which is second order for a first-order method) be small with respect to the distance from V_i to 0 and to 1:

$$(\Delta t_i)^2|\ddot{V}_i| \leq \epsilon V_i(1-V_i)$$

$$\frac{1}{\Delta t} \geq \frac{1}{\sqrt{\epsilon}}\frac{1}{N}(\sum_j r_j)(\max_i \frac{\sqrt{|\ddot{V}_i|/V_i(1-V_i)}}{r_i})$$

$$\frac{\text{Cost}}{\text{unit time}} = \frac{1}{\sqrt{\epsilon}}(\sum_j r_j)(\max_i \frac{\sqrt{|\ddot{V}_i|/V_i(1-V_i)}}{r_i}). \tag{11}$$

Two special cases of interest are $r_i = 1$ (no attention control) and $r_i \propto \sqrt{|\ddot{V}_i|/V_i(1-V_i)}$ (minimal cost).

Rather than use (11) directly, simulations were performed using the simpler dynamical system (9) and the cost (11) was monitored during the run. Other cost metrics modeling different implementations could be monitored instead. For example, the accuracy criterion could be relaxed so that ODE solution errors can be large as long as they don't affect dE/dt much; after all, we care about the final low-energy configuration and not the trajectory to it. One kind of cost that should be modeled is communication cost. Just counting updates as in equation 10 assumes the equivalent of a shared memory for neuron values.

Comparison with Newton's Method and the Conjugate Gradient Method

There are similarities between the attention control dynamics and well-established optimization methods such as Newton's method and the conjugate gradient method which suggest that large improvements in the effectiveness of the attention scheme are possible. One may express the attention control dynamics (slightly modified) as

$$\Delta V_i = -(\Delta t) r_i \frac{\partial E}{\partial V_i} \quad (r_i \geq 0) \quad \text{i.e.} \tag{12}$$

$$\Delta V_i = -(\Delta t)\sum_j r_{ij}\frac{\partial E}{\partial V_i}, \qquad r_{ij} = \delta_{ij} r_i \tag{13}$$

where

$$\Delta r_i = -(\Delta t) r_i \frac{\partial E_f}{\partial r_i}.$$

Newton's method makes a specific recommendation for the metric r_{ij}:

$$\Delta V_i = -(\Delta t)\sum_j D_{ij}^{(-1)}\frac{\partial E}{\partial V_i} \quad \text{where} \quad D_{ij} = \frac{\partial E}{\partial V_i \partial V_j}$$

In this method, the step size and direction ΔV_i depends on the first and second derivatives of the objective function E. Done incautiously, this procedure could be very expensive since there are $O(N^2)$ matrix elements instead of $O(N)$ rate parameters. The conjugate gradient method uses only $O(N)$ storage to find successive directions $c_i(t), c_i(t+\Delta t), \ldots$ which are orthogonal in the metric D_{ij}:

$$\Delta V_i = -(\Delta t) c_i \qquad \Delta c_i = a c_i + b\frac{\partial E}{\partial V_i} \tag{14}$$

where a and b must be chosen carefully [5] in order for this simple update scheme to successfully extract and use second derivative (D_{ij}) information. The slow neuron attention mechanism apparently fails to use this information, but it attempts to do more than optimize E: it also tries to disable many neurons to save the cost of simulating them. To extend the conjugate gradient method to an attention mechanism, then, it would be necessary to modify the coordinate-independent dynamics of equation 14 by introducing a special coordinate system (one coordinate per neuron, which we have been using anyway) and to alter the successive directions so that each direction vector $\vec{c}$ has as few nonzero components c_i as possible.

The unalloyed conjugate gradient method is optimial in some respects [6] among methods which do not take into account the structure of the second derivative matrix D_{ij}, which is closely related to the neural connection matrix. These matrices have a good deal of useful structure in the TSP network, however, and the structure can be used in selecting the next direction c_i or the corresponding parameters r_i in the attention control scheme. This leads us to the topic of windowed attention.

Windowed Attention

As explained earlier, N rate parameters r_i would be too expensive to compute dynamically if they did not "slow down" when the corresponding neurons slow down; i.e. $\dot{r}_i = r_i \partial E_f / \partial r_i$. This precludes $r_i = 0$ and therefore precludes modeling the switching among simulated neurons at the finest time scales. But if r_i were determined by many fewer parameters w_α describing a "window" of attention, so that $r_i = r_i(w_1, \ldots w_\alpha, \ldots)$, the w_α's could be simulated at full speed:

$$\dot{w_\alpha} = -\rho \frac{\partial E'_f}{\partial w_\alpha} = -\rho \sum_i \frac{\partial E'_f}{\partial r_i} \frac{\partial r_i}{\partial w_\alpha}. \tag{15}$$

Here E'_f differs from E_f by replacing $\sum_i \int^{r_i} \hat{g}^{-1}(x)dx$ with $\sum_\alpha \int^{w_\alpha} \hat{g}^{-1}(x)dx$. This insures that the gain functions $\hat{g}$ are only required for the relatively few quantities w_α so that the neural network for r_i is greatly reduced in size and cost. For matching problems such as TSP or inexact graph matching [7], a natural kind of window parameterization is

$$r_{ab} = r_a^{(1)} r_b^{(2)} \quad \text{so that} \quad w_\alpha = r_a^{1 \text{ or } 2} \tag{16}$$

which uses the structure of the connection matrix. Other, even more succinct parameterizations of r_{ab} may be worth considering as well. Equation 16 has the advantage that the cost constraint (7) is factored:

$$E_{\text{f,cost}} = \frac{1}{2}(\sum_{ab} r_{ab} - a)^2 = \frac{1}{2}[(\sum_a r_a^{(1)})(\sum_b r_b^{(2)}) - a]^2 = \frac{1}{2}[R^{(1)} R^{(2)} - a]^2$$

and one may distinguish three kinds of attention:

$R^{(1)} \ll R^{(2)}$	:	a few days are the focus of attention; they are being matched to many cities.
$R^{(1)} \gg R^{(2)}$	:	a few cities are the focus of attention.
$R^{(1)} \simeq R^{(2)}$	:	attention is restricted in both days and cities.

The choice among these alternatives is made automatically, by the dynamical system of equations (9, 15, 16).

3 Experimental Methods and Results

Our principal result is qualitative: for the 20-city TSP network of equations 2, 2, and 4 about 80 of the 400 neurons need to be attended to at a time. Also, the resulting tours are as good as those

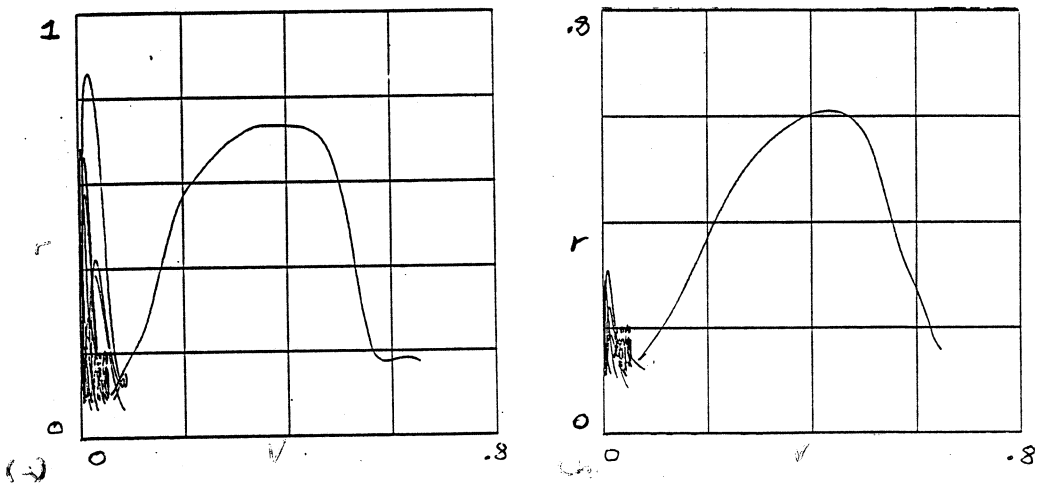

Figure 1: Thrashing can be eliminated by use of ρ. Plotted: trajectories $(V_{ab}(t), r_{ab}(t))$ for $a = 0$, all b and t. However, only fast-moving neurons are plotted: only those whose speed in the (r, v) plane exceeds a threshold. (a): Thrashing, $\rho = 1$; (b): not thrashing, $\rho = .5$.

obtained without an attention mechanism. For a 10-city network without the TSP data term ($A = 0$ in equation 2), a toy network which attains minimal energy at any permuatation matrix, the figure drops to 10 or less out of 100. We do not have scaling results to determine whether 80 is, for example, $4n$ or $n^{3/2}$ or $n^2/5$ for $n = 20$.

The number of neurons which must be in the focus of attention was measured as

$$P = \sum_{i=1}^{N} r_i \tag{17}$$

which is correct if each $r_i(t) = 0$ or 1. As discussed earlier, restriction to $r_i(t) = 0$ or 1 would be preferable and may be possible with windowed attention, using thresholding on the r's if necessary to insure that they are 0 or 1.

In order to measure P, the free coefficients in the dynamical systems for V_i and r_i were adjusted manually to avoid various standard problems. For $E(V)$, the tours had to be syntactically correct (for most random initial conditions) and as short as possible. These conditions involve the coefficients A and B in equation 2. The relative rate ρ was reduced just enough to control thrashing: a condition where most computational effort is spent updating r's rather than neural values, as in figure 1. It is a nontrivial experimental observation that ρ controls thrashing so well.

One may indefinitely depress the value of P by decreasing the coefficient a, but below $a = 80$ (experiments were done with $a = 400$, 200, 120, 80, 40, and 20) various difficulties arise. The r_i's are considered to be time averages of binary-valued r parameters at a finer time scale; by excessive time averaging one can lose all information as to which neurons are attended to and when. To prevent such long time averaging we want several r's to average out to nearly 1. The coefficient $\hat{C}$ of r_i's self-interaction is increased until this condition is met except in the early part of the network evolution; further work will be needed to satisfy the condition for all times. If the $r_i \simeq 1$ condition is not imposed, demanding smaller $P = \sum r_i$ eventually just scales each r_i down by the same factor and no new transition information is obtained despite the very small values of P (very large serialization) achieved.

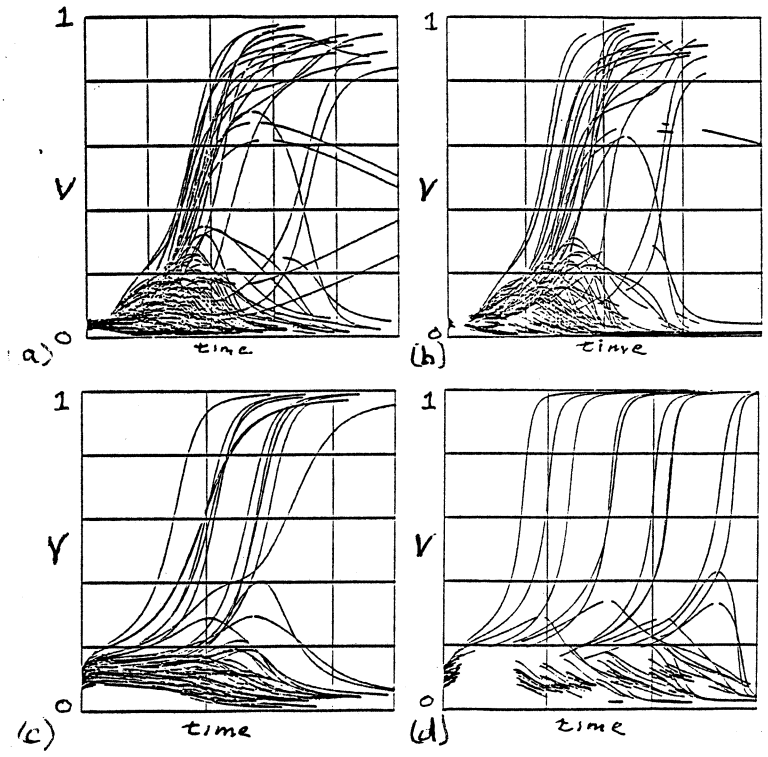

Figure 2: Traces of fast-moving neurons. Plotted: $(t, V_{ab}(t))$ for all a and b. (a): TSP network without attention control. (b): TSP network with attention control, for $\rho = .2$ which produces longer tours than those quoted in the text. (c): Permutation network without attention control. (d): Permutation network with attention control.

One may interpret the limitation on P simply: complete serialization of the TSP network is possible, but if thrashing is prohibited then equations 4 and 9 are only capable of controlling partial serialization. The problems with r_i scaling and with thrashing are what currently limit the size of the focus of attention to 80 or more neurons at a time.

The coefficients used were:

TSP Network				Permutation Network			
$A = .07$	$B = .1$	$C = .04$	$g_0 = 200$	$A = 0$	$B = .1$	$C = .2$	$g_0 = 50$
$\hat{A} = 500$	$\hat{B} = .3$	$\hat{C} = .5$	$\hat{g_0} = 10$	$\hat{A} = 200$	$\hat{B} = .6$	$\hat{C} = .5$	$\hat{g_0} = 10$
$n = 20$	$\rho = .5$	$a = 80$		$n = 10$	$\rho = .1$	$a = 10$	

Superposed traces of $V(t)$ are shown in figure 2 for the network with and without attention. Even in this figure some serialization may be observed, though most of the serialization has to do with $V_i \approx 0$ neurons which are assigned low values of r_i, as in figure 1.

Tour length was $4.13 \pm .199$ for four runs of the TSP network without attention (where different runs used different random positions for the 20 cities in the unit square), and $4.28 \pm .178$ for the same four TSP problems with the attention mechanism as described. The four runs were selected from a set of nine runs of which the other 5 were rejected because the TSP network without attention produced invalid tours. The attention network's tour was equally valid but slightly longer in each of the four cases. Thus there is little degradation of tour quality under the attention mechanism.

More sophisticed measures of cost than P are unaffected or worsened by the attention mechanism of equations (2, 3, 4, and 9). For equation 11 and its two specializations, the results are

Experiment	controlled	no control	minimal cost
TSP network - attention	481 ± 64.5	369 ± 22.1	89.9 ± 5.86
TSP network - no attention	240 ± 26.4	240 ± 26.4	86.9 ± 8.09
Permutation network - attention	51.2 ± 6.23	146 ± 23.6	23.6 ± 1.46
Permutation network - no attention	81.3 ± 6.93	81.3 ± 6.93	23.9 ± 2.06

We conjecture that minimizing such second-derivative cost metrics will require a dynamical system using $\partial^2 E/\partial V_i \partial V_j$ information, such as the conjugate gradient method.

4 Conclusions

A dynamical system was proposed for controlling attention in neural networks governed by energy function minimization. For the 20-city Traveling Salesman network of Hopfield and Tank, 80 or more of the 400 neurons must be attended to at once. The quality of solution was not significantly degraded by adding the attention mechanism. Estimated costs of simulating the network were increased, despite success in serializing it. Similar dynamical systems based on the conjugate gradient method were suggested as the solution to this problem. Difficulties in controlling the focus of attention at the finest time scale also suggest modifying the scheme so that the focus of attention is specified by relatively few parameters.

References

[1] T. Poggio and A. Hurlbert. Spotlight on attention. *Trends in Neurosciences*, August 1985.

[2] Christof Koch and Shimon Ullman. *Selecting One Among the Many: A Simple Network Implementing Shifts in Selective Visual Attention.* Springer, 1985. Human Neurobiology.

[3] J. J. Hopfield and D. W. Tank. 'Neural' computation of decisions in optimization problems. *Biological Cybernetics*, 52, 1985.

[4] Alan Lapedes and Robert Farber. *Programming a Massively Parallel, Computation Universal System: Static Behavior.* Technical Report LAUR 86-1179, Los Alamos National Laboratory, 1986.

[5] L. E. Scales. *Introduction to Non-Linear Optimization.* Springer-Verlag New York, 1985. Page 75.

[6] S. C. Eisenstat, H. C. Elman, and M. H. Schultz. Variational iterative methods for nonsymmetric systems of linear equations. *SIAM Journal for Numerical Analysis*, 20, April 1983.

[7] J. J. Hopfield. Graph matching. Personal communication.