Appl. Math. Lett. Vol. 4, No. 2, pp. 69–73, 1991
Printed in Great Britain

0893-9659/91 $3.00 + 0.00
Pergamon Press plc

Adjoint-Operators and Non-Adiabatic Learning Algorithms in Neural Networks

[1]N. TOOMARIAN AND [1,2]J. BARHEN

[1]Jet Propulsion Laboratory
[2]Division of Engineering and Applied Science

(Received July 1990)

Abstract. Adjoint sensitivity equations are presented, which can be solved *simultaneously* (i.e., forward in time) with the dynamics of a nonlinear neural network. These equations provide the foundations for a new methodology which enables the implementation of temporal learning algorithms in a highly efficient manner.

1. INTRODUCTION

The biggest promise of artificial neural networks as computational tools lies in the hope that they will ultimately enable complex information processing, comparable in sophistication to that carried out by biological systems. It is generally argued that, in order to achieve such an ambitious goal, versatile methodologies for "learning" should be available. Early efforts in that area have largely focused on the study of schemes for encoding nonlinear mappings characterized by time-independent inputs and outputs. The most widely used approach in that context has been the error backpropagation algorithm [13], which involves either static (i.e., "feedforward" [11]), or dynamic (i.e., "recurrent" [9]) networks. In a different vein [1-3], Barhen, Toomarian, Gulati and Zak have exploited the concepts of adjoint operators and terminal attractors to provide a firm mathematical foundation for learning such mappings with dynamical neural networks, while achieving a dramatic reduction in the overall computational costs.

More recently, there has been considerable interest in developing learning algorithms capable of modeling time-dependent phenomena [7]. The most general approach, based upon principles of non-lipschitzian dynamics [4,15], enables neural networks driven by vanishingly small noise to "self-program" in time, i.e., spontaneously to change their structural behavior by changing the location and nature of their attractors and repellers. This mimics, in a phenomenological sense, typical brain activities [5].

In a more restricted application domain, attention has focused on learning temporal sequences. The problem can be formulated as the minimization, over an arbitrary but finite time interval, of an appropriate error functional. Thus, the gradients of the functional with respect to the various parameters of the neural architecture, e.g., synaptic weights, neural gains, etc. must be computed. A number of methods have been proposed for carrying out this task. Williams and Zipser [14] discuss a scheme similar to the well known "forward sensitivity" method [6,12], in which the same set of sensitivity equations has to be solved again and again for each network parameter of interest. Clearly, this is computationally very expensive [2,3,6], and scales poorly to large systems. Pearlmutter [8], on the other hand, describes a variational approach which yields a set of equations which can be interpreted as a

This research was carried out at the Center for Space Microelectronics Technology, Jet Propulsion Laboratory, California Institute of Technology. Support for the work came from Agencies of the U.S. Department of Defense and from the Office of Basic Energy Sciences of the Department of Energy, through an agreement with the National Aeronautics and Space Administration.

simple instance of "adjoint sensitivity" theory [6,12]. These equations must be solved backwards in time and involve storage of the state variables from the forward network dynamics, which is impractical. Pineda [10] suggests combining the existence of disparate time scales with a heuristic gradient computation. However, the underlying adiabatic assumptions and highly "approximate" gradient evaluation technique place severe limits on the applicability of his approach.

In this paper we introduce a rigorous derivation of a novel system of adjoint equations, which can be solved *simultaneously* (i.e., forward in time) with the network dynamics, and thereby enable the implementation of temporal learning algorithms in a computationally efficient manner.

2. NON-ADIABATIC LEARNING

We formalize a neural network as an adaptive dynamical system whose temporal evolution is governed by the following set of coupled nonlinear differential equations:

$$\dot{u}_n + \kappa_n u_n = \sum_m T_{nm}\, g(\gamma_m u_m) + {}^k I_n \qquad t > 0 \tag{1}$$

where u_n represents the mean soma potential of the nth neuron [$u_n(0)$ being the steady state solution], and T_{nm} denotes the synaptic coupling from the $m-$th to the $n-$th neuron. The constant κ_n characterizes the decay of neuron activity. The sigmoidal function $g(\cdot)$ modulates the neural response, with gain given by γ_m; typically, $g(\gamma z) = \tanh(\gamma z)$. The time-dependent "source" term, ${}^k I_n(t)$, encodes component-contribution of the k-th target temporal pattern ${}^k\bar{a}(t)$ via the expression

$${}^k I_n(t) = \begin{cases} {}^k a_n(t) - g[\gamma_n u_n(t)]^{\beta} & \text{if } n \in S_X \\ 0 & \text{if } n \in S_H \cup S_Y \end{cases} . \tag{2}$$

The topographic input, output, and hidden network partitions S_X, S_Y and S_H, respectively, are architectural requirements related to the encoding of mapping-type problems. Details are given in [1]. In previous articles [1-2], we have demonstrated that in general, for $\beta = (2i+1)^{-1}$ and i a positive integer, Eq. (2) induces terminal attractor dynamics for static patterns, and provides opportunity for learning static phenomena in real-time.

To proceed formally with the development of a temporal learning algorithm, we consider an approach based upon the minimization of a "neuromorphic" energy functional E, given by the following expression

$$E(\bar{u},\bar{p}) = \int_t \frac{1}{\alpha} \sum_k \sum_n {}^k\Gamma_n^{\alpha}\, dt = \int_t F dt \tag{3}$$

where

$${}^k\Gamma_n(t) = \begin{cases} {}^k a_n(t) - g[\gamma_n {}^k u_n(t)] & \text{if } n \in S_X \cup S_Y \\ 0 & \text{if } n \in S_H \end{cases} \tag{4}$$

Typically, a positive value such as 2 is used for α. The indices n and k span over all neurons in the network and mapping samples respectively. The proposed objective function along with the specific form of the source term (2) enforce convergence of every neuron in S_X and S_Y to attractor coordinates corresponding to the time-dependent components of the input-output training patterns, thereby prompting the network to learn the underlying invariances. Since in our model the internal dynamical parameters of interest are the synaptic strengths T_{nm} of the interconnection topology, the characteristic decay constants κ_n, and the gain parameters γ_n, a vector of system parameters [3] will be formed as

$$\bar{p} = \{ T_{11}, \cdots T_{NN} \mid \kappa_1, \cdots \kappa_N \mid \gamma_1, \cdots \gamma_N \mid \} \tag{5}$$

We will assume that elements of $\bar{p}$ are, in principle, independent. Furthermore, we will also assume that, for a specific choice of parameters, a unique solution of Eq. (1) exists. Hence, $\bar{u}$ is an implicit function of $\bar{p}$.

Lyapunov stability requires the energy functional to be monotonically decreasing during learning time, τ. This translates into

$$\frac{dE}{d\tau} = \sum_{\mu=1}^{M} \frac{dE}{dp_\mu} \cdot \frac{dp_\mu}{d\tau} < 0 \tag{6}$$

Thus, one can always choose, with $\eta > 0$

$$\frac{dp_\mu}{d\tau} = -\eta \frac{dE}{dp_\mu} \tag{7}$$

Integrating the above dynamical system over the interval $[\tau, \tau + \Delta\tau]$, one obtains,

$$p_\mu(\tau + \Delta\tau) = p_\mu(\tau) - \eta \int_\tau^{\tau+\Delta\tau} \frac{dE}{dp_\mu} dt \tag{8}$$

Equation (8) implies that, in order to update a system parameter p_μ, one must evaluate the gradient of E with respect to p_μ in the interval $[\tau, \tau + \Delta\tau]$. Furthermore, using Eq. (3), one can write;

$$\frac{dE}{dp_\mu} = \int_t \frac{dF}{dp_\mu}\, dt = \int_t \frac{\partial F}{\partial p_\mu}\, dt + \sum_k \int_t \frac{\partial F}{\partial {}^k\bar{u}} \cdot \frac{\partial {}^k\bar{u}}{\partial p_\mu}\, dt \tag{9}$$

Since F is known analytically [viz. Eq. (4)] computation of $\partial F/\partial {}^k u_n$ and $\partial F/\partial p_\mu$ is straightforward. Henceforth, we will use the shorthand notation $F_{,kn}$ and $F_{,\mu}$ for these derivatives. Thus

$$F_{,kn} = -\, {}^k\Gamma_n^{\alpha-1} \gamma_n\, {}^k\hat{g}_n \tag{10}$$

$$F_{,\mu} = -\left[\sum_k {}^k\Gamma_n^{\alpha-1}\, {}^k\hat{g}_n\, {}^k u_n\right] \delta_{\gamma_n p_\mu} \tag{11}$$

where ${}^k\hat{g}_n$ represents the derivative of g_n with respect to ${}^k u_n$, and δ denotes the Kronecker symbol.

The quantity that needs to be determined is the vector ${}^k\bar{u}_{,\mu}$. Differentiating the activation dynamics, Eq. (1), with respect to p_μ, we obtain a set of equations to be referred to as "forward sensitivity equations":

$$\begin{cases} {}^k\dot{u}_{n,\mu} + \sum_m {}^k A_{nm}\, {}^k u_{m,\mu} = {}^k S_{n,\mu} & t > 0 \\ u_{n,\mu} = 0 & t = 0 \end{cases} \tag{12}$$

in which

$$\begin{aligned} {}^k A_{nm} &= -\left[-\kappa_n + \frac{\partial\, {}^k I_n}{\partial\, {}^k u_m}\right] \delta_{nm} - T_{nm}\, {}^k\hat{g}_m\, \gamma_m \\ {}^k A_{nm} &= -\, {}^k\eta_n\, \delta_{nm} - \gamma_m\, {}^k\hat{g}_m\, T_{nm} \end{aligned} \tag{13}$$

$${}^k S_{n,\mu} = [-\, {}^k u_n]\delta_{p_\mu,\kappa_n} + \left[\sum_m g(\gamma_m\, {}^k u_m)\right] \delta_{p_\mu,T_{nm}} + \left[\sum_m T_{nm}\, {}^k\hat{g}_m\, {}^k u_m + \frac{\partial\, {}^k I_n}{\partial \gamma_n}\right] \delta_{p_\mu,\gamma_n} \tag{14}$$

Since the initial conditions of the activation dynamics, Eq.(1), are excluded from the system parameter vector $\bar{p}$, the initial conditions of the forward sensitivity equations will be taken as zero. Computation of the gradients in Eq. (7) using the forward sensitivity scheme would

require solving Eq. (12) repeatedly, since the source term is explicitly depends on p_μ. This is undesirable.

An alternative way, however, exists. It is based upon the concept of adjoint operators, which eliminates the need for explicit appearance of ${}^k\bar{u}_{,\mu}$ in Eq. (9). The vector of adjoint functions, $\bar{v}$, contains all the information required for computing all the "sensitivities", dE/dp_μ. The necessary and sufficient conditions for constructing adjoint equations are discussed elsewhere [3,6,12]. It can be shown that the adjoint system pertaining to the forward system of equations (12) can be formally written as

$$-{}^k\dot{v}_n + \sum_m {}^kA^T_{nm}\,{}^kv_m = {}^kS^*_n \qquad t > 0 \tag{15}$$

In order to specify Eq. (15) in closed mathematical form, we must define the source term ${}^kS^*_n$ and the time-boundary conditions for the system. Both should be independent of p_μ and its derivatives. Multiplying the forward sensitivity equations by $\bar{v}$ and the adjoint system by $\bar{u}_{,\mu}$, subtracting the two equations and integrating over the time interval (t_o, t_f) yields:

$$(\bar{v}\;{}^k\bar{u}_{,\mu})_{t_f} - (\bar{v}\;{}^k\bar{u}_{,\mu})_{t_o} = \int_{t_o}^{t_f} [(\bar{v}\;{}^k\bar{S}_{,\mu}) - ({}^k\bar{u}_{,\mu}\;{}^k\bar{S}^*)]dt \tag{16}$$

By identifying ${}^kS^*_n$ with $\partial F/\partial\,{}^ku_n$, selecting the final time condition $\bar{v}(t = t_f) = 0$, and incorporating the initial condition of Eq. (12) into Eq. (16), we obtain:

$$\int_{t_o}^{t_f} \frac{\partial F}{\partial\,{}^k\bar{u}}\,{}^k\bar{u}_{,\mu}\,dt = \int_{t_o}^{t_f} \bar{v}\;{}^k\bar{S}_{,\mu}\,dt \tag{17}$$

This paradigm requires that the neural activation dynamics, Eq. (1), be solved first [forward in time, i.e., from t_o to t_f], followed by the adjoint system [integrated backwards in time]. The principal advantage of adjoint methods is the dramatic reduction in computational costs (e.g., at least $O(N^2)$ for an N-neuron network in the adiabatic approximation [3]). For temporal learning, however, a major drawback to date has resided with the necessity to store quantities such as ${}^kA$, ${}^k\bar{S}^*$ and ${}^k\bar{S}_{,\mu}$ at each time step.

Is it possible to overcome these rather severe limitations? We notice that the adjoint system [viz. Eq. (15)] is *linear* in the variables $\bar{v}$. Therefore, it is posible to obtain identical contributions to Eq. (9) with an alternative choice for adjoint source and time-boundary conditions. Indeed, let us choose:

$$\begin{cases} {}^k\bar{S}^* = \frac{\partial F}{\partial\,{}^k\bar{u}} - \bar{v}\delta(t - t_f) \\ \bar{v}(t = 0) = 0 \end{cases} \tag{18}$$

Then, in Eq. (9)

$$\int_t \frac{\partial F}{\partial\,{}^k\bar{u}}\,{}^k\bar{u}_{,\mu}dt = \int_t {}^k\bar{S}^*\;\;{}^k\bar{u}_{,\mu}dt + [\bar{v}\;\;{}^k\bar{u}_{,\mu}]_{t=t_f} \tag{19}$$

Taking into consideration Eq. (16) we see that

$$\frac{dE}{dp_\mu} = \int_t \frac{\partial F}{\partial p_\mu}dt + \int_t \bar{v}\;\;{}^k\bar{S}_{,\mu}\;dt \tag{20}$$

where $\bar{v}$ is the solution of the adjoint system [viz. Eqs. (15) and (18)]. In contradistinction to previous approaches, however, this system is now integrated forward in time, concomittantly with the neural activation dynamics.

3. CONCLUSIONS

A new methodology has been developed which enables the implementation of temporal learning algorithms in a highly efficient manner. Specifically, it combines the advantage of dramatic reductions in computational complexity inherent in adjoint methods with the ability to solve the equations forward in time. Not only is a large amount of computation and storage saved, but the handling of real-time applications also becomes possible. Finally, no limiting assumptions such as the adiabatic approximation are involved.

REFERENCES

1. J. Barhen, S. Gulati and M. Zak, Neural learning of constrained nonlinear transformations, *IEEE Computer* **22** (6), 67–76 (1989).
2. J. Barhen, N. Toomarian and S. Gulati, Adjoint operator algorithms for faster learning in dynamical neural networks, *Adv. Neur. Inf. Proc. Sys.* **2**, 498–508 (1990).
3. J. Barhen, N. Toomarian and S. Gulati, Application of adjoint operators to neural learning, *Appl. Math. Lett.* (in press) (1990).
4. J. Barhen, M. Zak and N. Toomarian, *Non-Lipschitzian Neural Dynamics* In *Neural Networks for Sensory Motor Control*, R. Eckmiller ed. (in press), North Holland, (1990).
5. E. Basar, *EEG-Brain Dynamics*, Elsevier, (1980).
6. D.G. Cacuci, Sensitivity theory for nonlinear systems, *J. Math. Phys.* **22** (12), 2794–2802 (1981).
7. M.W. Hirsch, Convergent activation dynamics in continuous time networks, *Neural Networks* **2** (5), 331–349 (1989).
8. B.A. Pearlmutter, Learning state space trajectories in recurrent neural networks, *Neural Computation* **1** (2), 263–269 (1989).
9. F. Pineda, Dynamics and architecture in neural computation, *J. of Complexity* **4**, 216–245 (1988).
10. F. Pineda, Time dependent adaptive neural networks, *Adv. Neur. Inf. Proc. Sys.* **2**, 710–718 (1990).
11. D.E. Rumelhart and J.L. McC., *Parallel and Distributed Processing*, MIT Press, (1986).
12. N. Toomarian, E. Wacholder and S. Kaizerman, Sensitivity analysis of two-phase flow problems, *Nucl. Sci. Eng.* **99** (1), 53–81 (1987).
13. P. Werbos, Beyond regression: new tools for prediction and analysis in the behavioral sciences, *Ph.D. Thesis, Harvard Univ.* (1974).
14. R.J. Williams and D. Zipser, A learning algorithm for continually running fully recurrent neural networks, *Neural Computation* **1** (2), 270–280 (1989).
15. M. Zak, Terminal attractors for addressable memory in neural networks, *Complex Systems* **3**, 471–492 (1989).

[1]Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109
[2]Division of Engineering and Applied Science, California Institute of Technology, Pasadena, CA 91109