Neural Processing Letters, Vol. 2, No. 3, 12-16, 1995

Adaptative time constants improve the prediction capability of recurrent neural networks

Jean-Philippe Draye[1,*], Davor Pavisic[1,3], Guy Cheron[2], Gaëtan Libert[1]

[1] *Parallel Information Processing Laboratory, Faculté Polytechnique de Mons rue de Houdain 9, B-7000 Mons, Belgium*
[2] *Laboratory of Biomechanics, University of Brussels Av. Paul Héger 28, B-1050 Brussels, Belgium*
[3] *Universidad Del Valle P.O. Box 4742, Cochabamba, Bolivia*

Abstract. Classical statistical techniques for prediction reach their limitations in applications with nonlinearities in the data set; nevertheless, neural models can counteract these limitations. In this paper, we present a recurrent neural model where we associate an adaptative time constant to each neuron-like unit and a learning algorithm to train these dynamic recurrent networks. We test the network by training it to predict the Mackey-Glass chaotic signal. To evaluate the quality of the prediction, we computed the power spectra of the two signals and computed the associated fractional error. Results show that the introduction of adaptative time constants associated to each neuron of a recurrent network improves the quality of the prediction and the dynamical features of a neural model. The performance of such dynamic recurrent neural networks outperform time-delay neural networks.

1. Introduction

In many areas of scientific research, the problem of predicting the future of dynamical systems arises. Unfortunately, when the observed dynamics are nonlinear with a complex dependence on time, the formulation of reliable predictions becomes extremely difficult. The prediction problem has been studied as a problem of multidimensional function approximation. This approach has produced new methodologies for the analysis of nonlinear time series, including local predictive procedures (see [1]). Neural network architectures have drawn considerable attention in recent years because of their interesting learning abilities. Moreover, they are capable of dealing with the problem of structural instability. Several researchers have reported exceptional results using neural networks [2, 3].

To tackle the problem of optimal prediction, the class of feedforward neural networks defined in a suitable probabilistic environment has often been used. Following the development of recurrent neural networks, these ones have been introduced in the field of time series prediction. Recurrent neural models present new features (not found in feedforward ones), such the learning of attractor dynamics, the storage of information and above all, their ability to deal with time-varying signals.

We introduce here a variation of the traditional neural network model which presents two types of adaptative parameters : the classical weights between the units and the time constants associated with each artificial neuron (these time constants represent the membrane time constants of the biological neurons [4]).

In this paper, we investigate the impact of adaptative time constants on the performance of recurrent neural networks during chaotic signal prediction and production. Signal prediction is the classical task where the input to the network is the time-varying signal and the desired output is a prediction of the signal at a fixed time increment in the future. Nevertheless, once a neural network is trained, an interesting application is to replace the external input by the output of the device. The resulting dynamical system should spontaneously generate its own version of the signal.

* Jean-Philippe Draye is a Research Assistant of the Belgian National Fund for Scientific Research

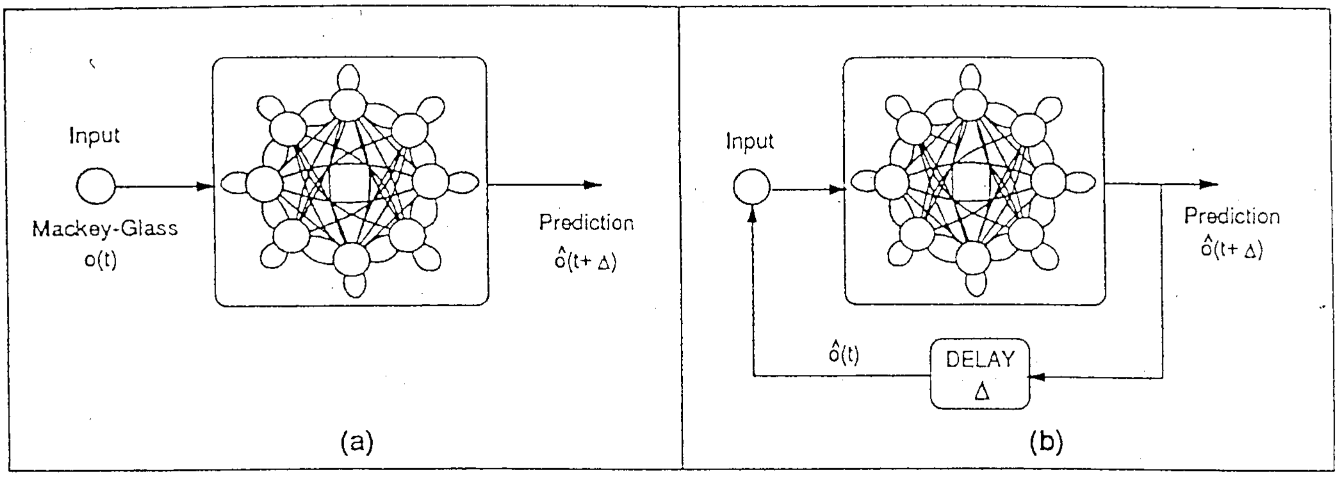

Fig. 1. Configuration of the network for : (a) prediction of the Mackey-Glass signal and (b) for production.

2. Test signal

The chaotic signal produced by integrating the Mackey-Glass delay-differential equation [5, 6]:

$$\frac{dx(t)}{dt} - bx(t) + a\frac{x(t-\tau)}{1+[x(t-\tau)]^{10}} \qquad (1)$$

provides a useful benchmark for testing predictive techniques [1, 7, 8]. For comparison with previous results, we chose $\tau = 17$, $a = 0.2$, $b = 0.1$, and trained the network to predict six time units into the future. We integrated (1) using a four-point Runge-Kutta method with a step of 0.05. The initial conditions were $x(t) = 0.8$ for $t < 0$, and the equation was integrated up to $t = 1000$ to allow transients to die out. The resulting signal is quasi-periodic with a characteristic time of $T_c = 50$, lying on a strange attractor with a fractal dimension of approximately 2.1 [5] (see figure 2.). The signal has been divided by two and shifted to fit in the interval corresponding to the range of values of the output of the sigmoid function.

The network was trained with 1000 points, sampled on the Mackey-Glass signal with time increments of 0.5. It was next evaluated for the next 1000 points.

3. Network setup

We give below a list of the different parameters of the network :

- *Network architecture.* The network is recurrent and consists of a series of fully-connected neurons. Therefore each neuron in a N-neuron network has N connections (including a self-connection)

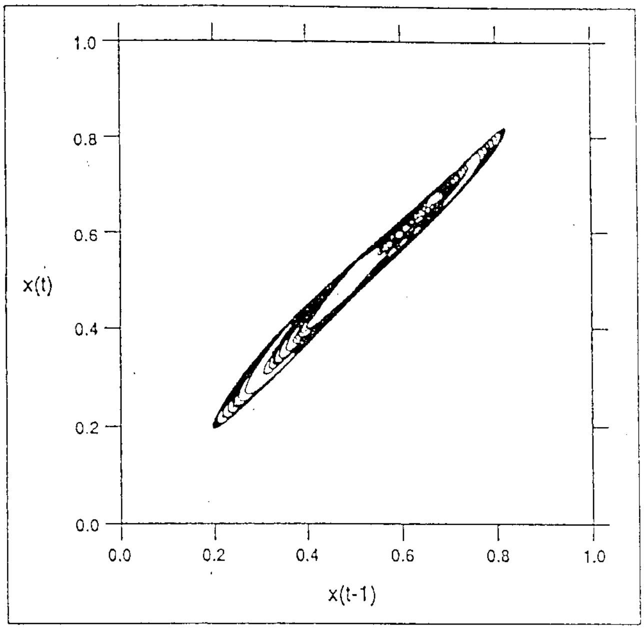

Fig. 2. Mackey-Glass attractor.

- *Training time.* With the term training time, we mean the number of presentation of the entire training set to the network (iterations). All the simulations run on SUN 670MP; convergence is reached within 5000 iterations typically.
- *Learning algorithm.* We consider neural networks governed by the following equations [9]:

$$T_i\frac{dy_i}{dt} = -y_i + F(x_i) + I_i \qquad (2)$$

where y_i is the state or activation level of unit i, $F(\alpha)$ is the squashing function $F(\alpha) = (1+e^{-\alpha})^{-1}$ and x_i is given by

$$x_i = \sum_j w_{ji} y_j \qquad (3)$$

Equation (2) is *the propagation equation of the network.* The time constants T_i will act like a relaxation process. The correction of the time constants will be included in the learning process in order to increase the dynamics of the model. The correction of the weights and the time constants is done by the algorithm of "*backpropagation through time*". In order to avoid a misleading confusion, we have to differentiate the algorithm of backpropagation through time proposed by Rumelhart et al. [10] and the algorithm, also named backpropagation through time, presented in this paper. In the algorithm of Rumelhart, the behaviour of a recurrent network is achieved in a feedforward network at cost of duplicating the structure many times (the recurrent network is unfolded into a multilayer feedforward network that grows by one layer on each time step). Unfortunately, this simple solution is suffering from its growing memory requirement in considerably long training sequences. We use an algorithm that does not unfold the recurrent network but computes the learning equations using a forward and a backward step through time (time appears explicitly in the equations). Since we want our network to exhibit some particular temporal behaviour, the error function will be a functional defined as

$$E = \int_{t_0}^{t_1} q(y(t), t)\,dt \tag{4}$$

where t_0 and t_1 give the time interval during which the correction process occurs. The function $q(y(t),t)$ is the cost function at time t which depends on the vector of the neuron activations y and on time. We then introduce the new variables p_i (called the *adjoint variables*) that will be determined by the system of differential equations :

$$\frac{dp_i}{dt} = \frac{1}{T_i} p_i - e_i - \sum_j \frac{1}{T_j} w_{ij} F'(x_j) p_j \tag{5}$$

with boundary conditions $p_i(t_1) = 0$. After the introduction of these new variables, we can derive the learning equations :

$$\frac{\partial E}{\partial w_{ij}} = \frac{1}{T_i} \int_{t_0}^{t_1} y_i F'(x_j) p_j\,dt \tag{6}$$

$$\frac{\partial E}{\partial T_i} = \frac{1}{T_i} \int_{t_0}^{t_1} p_i \frac{dy_i}{dt}\,dt \tag{7}$$

It is important to notice that these equations can be derived either using a finite difference approximation, the calculus of variation, the Lagrange multiplier, or even from the theory of optimal control in dynamic programming using the Pontryaguin Maximum Principle. A thorough presentation of the learning algorithm and a comparison of some acceleration techniques can be found in [11].

4. Configuration as an adaptative prediction filter

The network consists of a N fully-connected neuron system. One neuron gives the predicted value $\hat{o}(t + \Delta)$ (where Δ is a fixed delay, and in our case, equals 6); all the other neurons receive the input signal $o(t)$. The process of learning adapts all the network parameters forcing the prediction output to produce a signal that approximates $\hat{o}(t + \Delta)$.

Since the objective of this work is to show the performance of dynamic recurrent neural networks, instead of showing a comparison between the real Mackey-Glass signal and the predicted one, we will rather present a scattergram depicting the desired versus the predicted output. The ideal shape in scattergram of figure 3 would be a straight line with a slope 45°. The reason is clear, if the desired output is, let us say 0.45, the ideal predicted output would be 0.45. We see, on figure 3, that our set of predicted values (1000 points) clearly match the ideal straight line (dashed line).

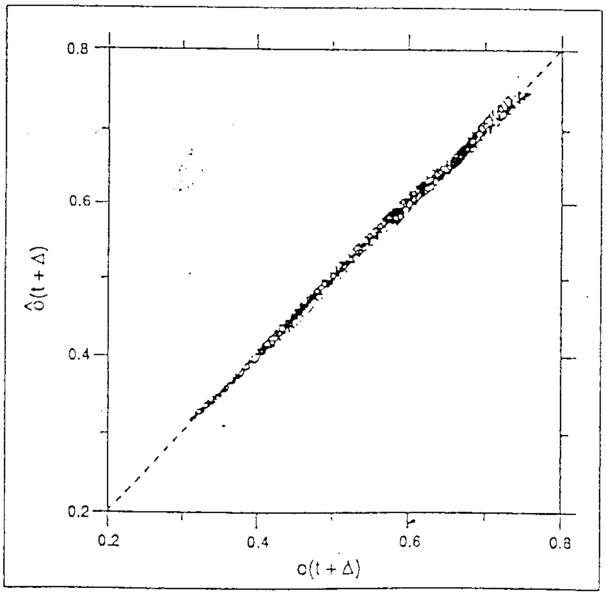

Fig. 3. Scattergrams of the actual predicted values $\hat{o}(t + \Delta)$ versus the desired predicted ourput $o(t + \Delta)$. The ideal curve is a straight line with a slope of 45° (shown in dashed line).

5. Signal production results

As mentionned above, if we freeze the weights and time constants of the network, remove the input signal, and connect the output labeled $\hat{o}(t+\Delta)$ via a delay Δ to the input node, we create a dynamical system that spontaneously generates its own version of the Mackey-Glass signal. In order to evaluate the network performance, figure 4 shows a comparison between the power spectrum of the signal generated by the network and the spectrum of the Mackey-Glass signal, integrated during a four-point Runge-Kutta method. Figure 5 presents the fractional error between these spectra. This figure also shows the fractional error of between the real spectrum and the one produced by a network trained with continuous-time temporal backpropagation with adaptable time delays [12]. This network presents a structure of feedforward network with two hidden layers of 10 neurons, one output and a total of 150 adaptable connections. We clearly see that our dynamic recurrent neural network outperforms the time-delay neural model : the RMS value of the fractional error falls from 0.252 to 0.064.

6. Stability analysis

We conduct a stability analysis of our network according to different parameters : the network architecture, the gradient descent control terms.

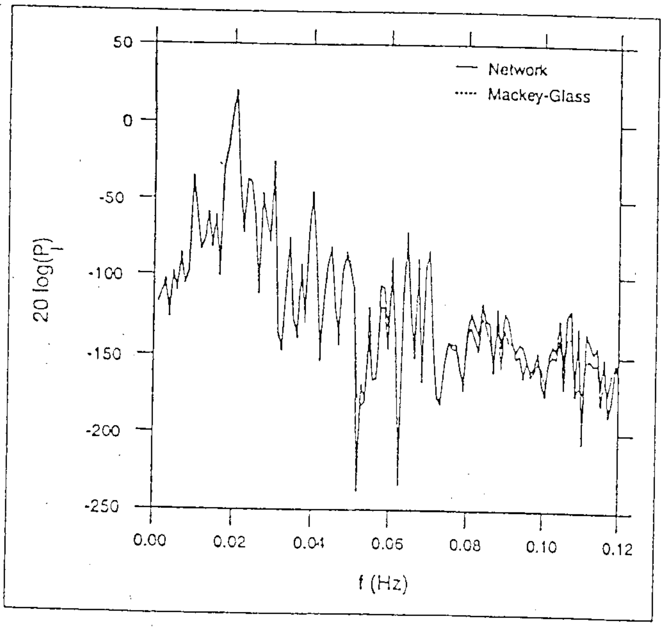

Fig. 4. Comparison between the spectrum of the Mackey-Glass signal and the spectrum of the predicted signal produced by the network. The spectra were computed using Fast Fourier Transform (FFT).

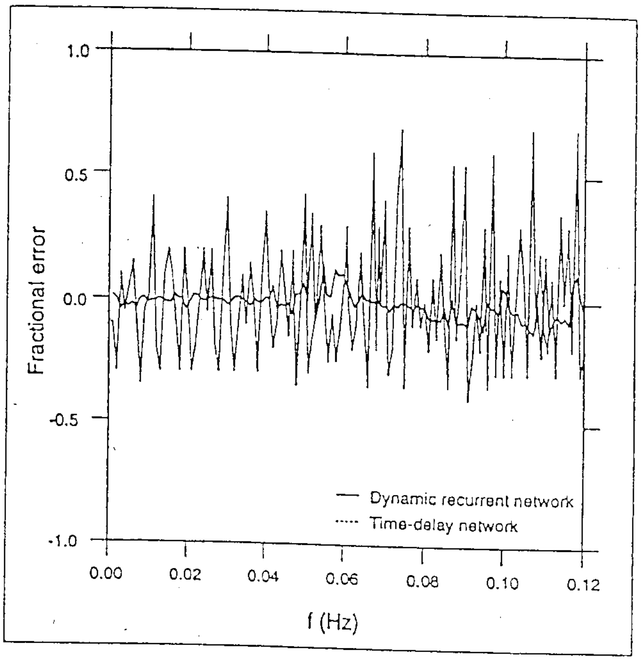

Fig. 5. Fractional error between the real spectrum of the Mackey-Glass signal and the spectrum generated by the network. The fractional error is defined as the difference between the two spectra, divided by the magnitude of the Mackey-glass spectrum. Solid line shows the fractional error for our dynamic recurrent neural network. The dotted line is associated to the fractional error of a neural network with adaptative time-delay connections.

We experimented different architectures of the network, varying the number N of neurons. For all these simulations, we conclude that 20 neurons is a good compromise between speed of convergence and quality of prediction.

Several accelerations techniques were evaluated, such as a momentum term, learning rate adjustment using line search. We found that the method introduced by Silva and Almeida gives the best results. Their method proposes to associate an individual learning rate to each weight. These learning rates are adapted by observing the signs of the last two gradients [13].

7. Conclusion

Classical statistical techniques for prediction reach their limitations in applications with nonlinearities in the data set. It is well known that neural learning procedures can significantly outperform current best practice in typical prediction applications. The performance presented in this paper shows that the introduction of adaptative time constants associated to each neuron of a recurrent network improve the quality of the prediction. Such dynamic recurrent neural networks outperform time-delay neural networks. Indeed, the action of time constants (obtained by a leaky integrator that ends each neuron-like unit) is much more powerful than the information storage of time-delay connections. Time constants simultaneously improve the nonlinearity effect of the sigmoid function and the memory effect of time delays The weights and time constants of the neural model are adapted by the "backpropagation through time" algorithm that has been briefly described in section 3. We tested the model by training it to predict the Mackey-Glass chaotic signal at time $t+6$ using observations of time t. Our simulations suggest that adaptative time constants improve considerably the predictive performance. Moreover, if the output of the network is connected directly to its input, the associated predictor network spontaneously generates the training signal. The spectrum of the synthesized signal appears to approximate the training signal spectrum. The fractional errors between spectra of real and simulated signals show that dynamic recurrent networks is more effective than feedforward networks with adaptative time-delay connections.

References

[1] J.D. Farmer, J.J. Sidorowich. Predicting chaotic time series, *Physical Review Lettres*, vol. 59, pp. 845-848, 1987.

[2] E. Schoenenburg. Stock price prediction using neural networks, a project report, *Neurocomputing* vol. 2, pp. 17-27, 1990.

[3] A. Refenes, M. Azema-Barac, L. Chen, S.A. Karoussos. Currency exchange rate prediction and neural networks design strategies, *Neural Computing & Applications Journal*, vol. 1, no. 2, 1991.

[4] S.W. Kuffler, J.G. Nicholis, A.R. Martin. *From Neuron to Brain*, 2nd ed., Sinauer Associates Inc. Sunderlands, Massachussetts, 1984.

[5] J.D. Farmer. Chaotic attractors of an infinite-dimensional dynamical system, *Physica*, vol. D 4, pp. 366-393, 1982.

[6] M.C. Mackey, L. Glass. Oscillations and chaos in physiological control systems, *Science*, vol. 197,, pp. 287-289, 1977.

[7] A. Lapedes, R. Farber. Nonlinear signal processing using neural networks : prediction and system modelling, *Technical Report Los Alamos National Laboratory - LA-UR-87*-2662, 1987.

[8] M. Casdagli. Nonlinear prediction of chaotic time series, *Physica*, vol. D 35, pp. 335-336, 1989.

[9] B. A. Pearlmutter. Learning state space trajectories in recurrent neural networks, *Neural Computation*, vol. 1, pp. 263-269, 1989.

[10] D. E. Rumelhart, G. E. Hinton, R. J. Williams. Learning internal representations by error propagation, in: *Parallel Distributed Processing : Explorations of the Microstructure of Cognition*, vol. I, Cambridge MA, Bradford Books, 1986.

[11] J.P. Draye, G. Libert. Dynamic Recurrent Neural Networks : Theoretical Aspects and Optimization, *Neural Network World*, vol. 3, no. 6, pp. 705-714, 1993.

[12] S.P. Day, M.R. Davenport. Continuous-time temporal backpropagation with adaptable time delays, *Technical Report University of British Columbia (Vancouver - Canada)*, 1991.

[13] F.M. Silva, L.B. Almeida. Speeding up backpropagation, in : R. Eckmiller ed., *Advanced Neural Computers*, pp. 151-158, 1990.