CHAOTIC TIME-SERIES PREDICTION AND THE RELOCATING-LMS (RLMS) ALGORITHM FOR RADIAL BASIS FUNCTION NETWORKS

Afsar SARANLI[†], *Buyurman BAYKAL*[‡]
[†] Middle East Technical University, Ankara, TURKIYE.
Tel: +90 312 2104419; fax: +90 312 2101261
e-mail: saranli@rorqual.cc.metu.edu.tr
[‡] Imperial College of Science, Technology and Medicine, London, UK.
e-mail: b.baykal@ic.ac.uk

ABSTRACT

In this study, the problem of real-time chaotic time-series prediction using Radial Basis Function Networks is addressed. The performance of a number of training methods based either on supervised error correction or on adaptive clustering techniques are investigated. Some performance drawbacks due to their exclusive usage are pointed out and a new algorithm combining their desirable properties is presented. The proposed *Relocating-LMS* algorithm is compared with the existing methods on a chaotic time-series produced by the Mackey-Glass Equation and is further tested on a series generated by the Logistic Map function, leading to encouraging results.

1 INTRODUCTION

Nonlinear dynamic systems exhibiting chaotic behavior arise in many real world problems including among others the onset of turbulence in fluids, lasers and plasma physics. The investigation of such systems becomes further attractive since they are able to imitate the behavior of much simpler systems. Identifying deterministic dynamic systems which exhibit such complex, sometimes seemingly *random* behavior[1] is an important and challenging problem in signal processing and have been addressed by many researchers. In the last decade, nonlinear adaptive methods based on various neural network architectures have led to promising results[1][2]. One of these network architectures is the *Radial Basis Function Network* which interprets the identification problem as a function approximation task in a high dimensional space[3][4][5]. This identification is generally performed by a prediction operation in time while the network parameters are optimized according to some criteria.

Although the RBF networks are among the most general nonlinear models for multivariate function approximation, there exist a number of difficulties which become especially important in real-time applications such as the one considered here. Existing training algorithms have severe drawbacks but also many desirable features which should be combined for acceptable performance with reasonable computational cost.

In Section 2, we begin by giving a brief discussion of the two chaotic systems studied in this work. This is followed in Section 3, by a description of the RBF network and the associated prediction setup. Then the existing training strategies and their drawbacks are discussed. The description of the new training method is presented in Section 4. Finally, Section 5 gives the experimental results on the two time-series considered.

2 THE MACKEY-GLASS EQUATION AND THE LOGISTIC MAP FUNCTION

In this study, time-series generated by two deterministic dynamic systems, namely the *Mackey-Glass Differential Delay Equation* and the *Logistic Map Function* are considered. The Mackey-Glass Equation is given by the expression

$$\frac{dy(t)}{dt} = -by(t) + a\frac{y(t-\tau)}{1+y(t-\tau)^{10}}. \qquad (1)$$

The time-series considered is obtained by integrating the equation on fixed time steps. The three parameters of the equation determines the series behavior. When the parameters a and b are taken as $a = 0.2$ and $b = 0.1$, the value $\tau = 17$ leads to a chaotic time-series concentrated around a strange attractor of fractal dimension 2.1[1]. The Logistic Map Function, which is a simpler system is described by the formula

$$y[n+1] = 4by[n](1-y[n]). \qquad (2)$$

This is an iterative map where $y[n]$ gives the resulting time-series. The value of the parameter b determines the series behavior and $b = 1$ leads to an ergodic chaotic time-series[1]. These two time-series can be observed in Fig.4 and in Fig.6 (b) respectively.

3 TIME-SERIES PREDICTION USING THE RBF NETWORK

3.1 The RBF Network

The Radial Basis Function Network is a nonlinear model implementing a multivariate nonlinear function between

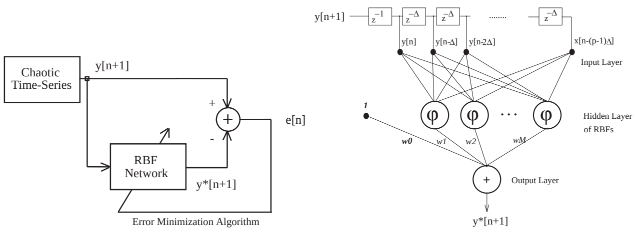

Figure 1: Prediction using the RBF Network **(a)** Prediction setup, **(b)** RBF Network for prediction.

its p-dimensional input space and its output. This model consists of three layers. The *input layer* is composed of p nodes with unity transfer function, each connected to every node in the *hidden layer*. The hidden layer nodes implement M nonlinear *bell-shaped* functions called the *Radial Basis Functions*. These have an effective nonzero value only for a localized region of the input space. An extra node with unit output is present in this layer to model the bias in the series. Finally, the output layer is a single node which computes the weighted sum of the outputs of the M hidden layer nodes. The overall nonlinear mapping implemented by the model is therefore given by

$$F = \sum_{i=1}^{M} w_i \phi_i(||\mathbf{x} - \mathbf{x}_i||). \tag{3}$$

In this expression $\mathbf{x}$ is the input vector, w_i are the linear weights of the outputs of the hidden units and $\mathbf{x}_i$ are the center locations of the basis functions $\phi_i(.)$. The norm in the expression is generally the Euclidean norm while the basis functions used in the expansion are usually taken to be the *Gaussian functions*. The variance σ is the shape parameter and determines their effective width. However, for this study, the *Radially Cubic B-Spline Functions* are used. These lead to very similar performance as the Gaussian functions but have compact support and are shown to be computationally much more efficient[6]. The knot-length parameter d determines the radial width of the basis function in the input space and the function is identically zero outside $2d$ radial distance from the center.

3.2 Prediction Setup

To be able to use the RBF model for the prediction of a time-series, we have to define the prediction setup and the optimization criteria. This setup is illustrated in Fig.1. Here, p past samples of the time-series separated by Δ samples are used to predict the next sample. When considered as a function approximation problem, the time-series defines a nonlinear multivariate function in its p-dimensional input space and the RBF model tries to approximate this function. For supervised training algorithms, the squared prediction error is used to optimize the linear and nonlinear parameters of the model which are updated as each new sample becomes available. When clustering methods are used for training, only the location of each new data point in the prediction input space is used for the adaptation of the nonlinear parameters.

3.3 Existing Training Algorithms and Their Drawbacks

The RBF networks are among the most general nonlinear models for multivariate function approximation. However, when trying to use this class of models in the predictive identification of a chaotic system without any apriori information, one is faced with many problems especially in the determination of the nonlinear parameters of the model. The difficulties are more severe when the identification has to be in real-time. The high dimensionality of the prediction problem necessitates a prohibitively large number of basis function centers to perform identification on the whole input space and hence renders the problem intractable. This problem may be alleviated by exploiting the fact that the phase trajectories of most of the chaotic systems are constrained in a certain region of the entire phase-space, usually called a *strange attractor*. This region can also be time-varying. One has to try to collect the available network resources, namely the available basis functions, to this feasible area. Hence, an efficient clustering technique is necessary to perform identification. The technique should also be adaptive to handle time-varying cases. The algorithms proposed in the literature for the optimization of the RBF network parameters have ignored this fact by either considering constant RBF centers chosen randomly together with quadratic optimization of the linear coefficients or by using combined quadratic and nonlinear optimization of the whole set of network parameters without making use of any clustering technique[4][7]. The former technique wastes the basis functions by ignoring the presence of a finite region spanned by the series and the alternative of choosing the centers from the data set is clearly an off-line method. The latter technique on the other hand is bound to get trapped in local minimum points since it is a difficult task to determine reasonable initial conditions for the nonlinear parameters. Hence a large number of basis functions would still remain in unfeasible regions of the input space. Other researchers considered clustering techniques alone, without using any supervised technique for the nonlinear parameters. One of these is the use of the *Adaptive k-means Clustering Algorithm*[8] where each new data point from the series is used to update the nearest basis function center according to the update formula $\Delta\mathbf{x}_i = \mu(\mathbf{x} - \mathbf{x}_i)$. Here, the parameter μ is the adaptation rate. Associated with this procedure is an adaptive version of the *Nearest Neighbor Heuristics*[8] which determines the function widths leading to a smooth approximating surface.

Although the clustering methods achieved sensible distributions over this region of interest, they ignored the fact that the final optimization criteria was the minimization of the prediction error and hence remained suboptimal in this sense. Only recently, some algorithms combining the desired properties of both class of algorithms were reported in the literature[9]. However, these methods considered off-line formulations where the data set was available apriori.

4 The Relocating-LMS(RLMS) Algorithm

Experimental evidence shows that neither supervised adaptation methods, nor the clustering methods does yield to satisfactory results with acceptable number of basis functions when they are used exclusively. On the other hand, both class of algorithms have features that may be mandatory for optimal performance.

The proposed training algorithm combines the supervised training of the parameters, which is mandatory for optimal prediction error performance, with a clustering effect for the basis function centers. The algorithm operates on two logical layers. The first layer is composed of an optimization method, namely the LMS algorithm to train all the parameters of the network in a supervised manner while the second layer is a higher level mechanism which *relocates* the basis function centers trapped in unfeasable regions in the input space. The relocation of the centers is done by keeping track of the data points falling in the *receptive fields* of each basis function. When a basis function does not receive a data point for a sample of the time-series, its *probability counter* is incremented by an amount μ towards unity. The basis function with highest probability exceeding a threshold value determined by a random experiment, is relocated inside the series trajectory. The new position for the center is the last data point from the time-series. During relocation, the weight of the function is set to zero while its width is initialized using the adaptive Nearest-Neighbor Heuristics. The probability counter of the basis function is set to zero and the counters of all others are decremented by an amount given by ν. All the parameters of the basis function are then left to be trained by the LMS adaptation layer. The described method combines a clustering effect with the supervised training of the parameters and leads to considerably better results which will now be presented.

5 SIMULATION RESULTS

The two chaotic time-series are computed for 10000 sample points. The performance of three existing training methods are compared with the performance of the proposed RLMS algorithm on the task of predicting the Mackey-Glass Equation. The existing methods tested are the use of full LMS adaptation starting from a uniform distribution over the series data, the use of Adaptive k-means Clustering and the use of Adaptive k-means Clustering together with the Adaptive Nearest-

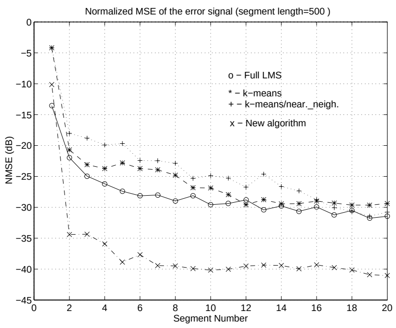

Figure 2: Comparison of the performance of three existing training methods with the RLMS algorithm.

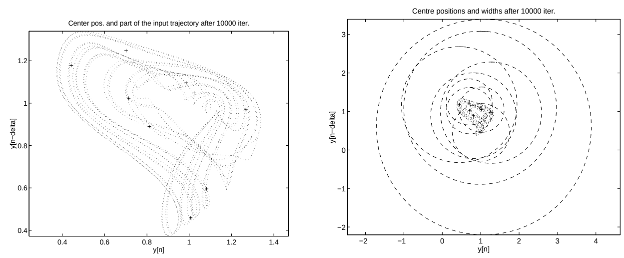

Figure 3: **(a)** The basis function center places, **(b)** Basis function widhts after training with the RLMS algorithm.

Neighbor Heuristics. It should be noted that the initial center distribution used for full LMS adaptation uses apriori information about the series and much worse performance should be expected without this initial condition. The performance of the algorithms are illustrated in Fig.2 by means of the *Normalized Mean Squared Error*(NMSE) value. This is the mean square value of the error signal normalized to the mean square value of the time-series over segments of length $L = 500$. The new method achieved considerably better performance in predicting the time-series in real-time with the same amount ($M = 9$) of basis functions. The curves are averaged over 20 Monte-Carlo runs. The results of training with the RLMS algorithm are presented in Fig.3 and Fig.4. It can be observed that a reasonable distribution of the centers is achieved. Moreover, these locations are also optimal for prediction error performance due to the concurrent supervised optimization.

The method is also tested on the time-series generated by the Logistic Map Function. The prediction performance is illustrated in Fig.5 for different number of basis function centers. The results of prediction for $M = 20$ case are illustrated in Fig.6. Note that the fast changing nature of this series necessitates $M = 30$ centers to achieve the same performance with the previous case and also leads to narrower basis function widths.

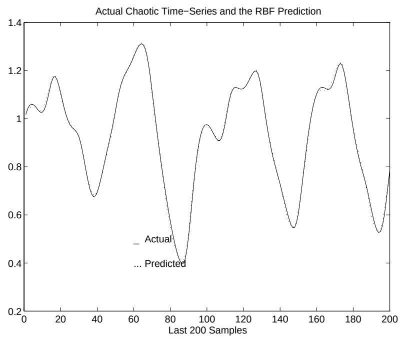

Figure 4: Comparison of actual and the predicted time-series after RLMS adaptation.

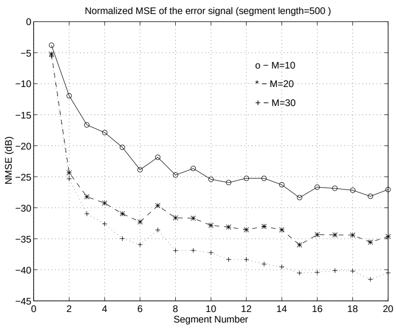

Figure 5: Performance of the RLMS algorithm on the time-series generated by the Logistic Map Function for different number of basis functions.

6 CONCLUSIONS

In this study, the use of the RBF Networks in real-time chaotic time-series prediction is investigated. Existing training algorithms are investigated and their problems which become especially important in real-time applications are addressed. The Relocating-LMS algorithm is proposed in an attempt to combine the desired properties of the existing training methods in an adaptive scheme. The performance of the new algorithm is compared with the existing methods on the chaotic time-series generated by the Mackey-Glass Equation and is further tested on the time-series generated by the Logistic Map Function. Experimental results shows that the algorithm leads to better prediction error performance than both class of existing algorithms.

References

[1] Lapedes, A., Farber, R., *Nonlinear Signal Processing Using Neural Networks: Prediction and System Modelling*, Technical Report, Los Alamos National Laboratory, Los Alamos, New Mexico, 1987.

[2] Casdagli, M., "Nonlinear prediction of chaotic time series," *Physica*, Elsevier Science Publishers, D 35, 1989, pp. 335-356.

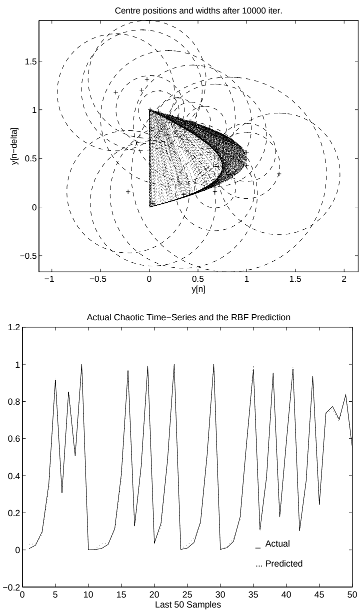

Figure 6: The results of training with the RLMS algorithm ($M = 20$): **(a)** Basis function center places and widhts, **(b)** Comparison of actual and the predicted time-series.

[3] Powell, M.J.D., *Radial Basis Functions for Multivariate Interpolation: A Review,* Preprint, Univ. of Cambridge, 1985.

[4] Broomhead, D.S., Lowe, D., "Multi-variable function interpolation and adaptive networks," *Complex Systems*, 2, (3), pp. 269-303.

[5] Poggio, T., Girosi, F., "Networks for approximation and learning," *Proceedings of IEEE,* 78, pp. 1481-1497.

[6] Saranli, A., *Investigation of an Alternative B-Spline Basis in Adaptive RBF Networks, with Applications to System Identification and Time-Series Prediction*, M.Sc. Thesis, Imperial College, London, 1994.

[7] Lowe, D., "Adaptive radial basis function nonlinearities and the problem of generalization," 1^{st} *IEE International Conference on Artificial Neural Networks*, London, 1989, pp. 171-175.

[8] Moody, J., Darken, C.J., "Fast learning in networks of locally tuned processing units," *Neural Computation*, Vol 1, No. 2, 1989, pp. 281-294.

[9] Chen, S., Cowan, C.F.N., Grant, P.M., "Orthogonal Least Squares Learning Algorithm for Radial Basis Function Networks," *IEEE Transactions on Neural Networks*, Vol. 2, No 2, 1991, pp. 302-309.