SSA, SVD, QR-cp, and RBF Model Reduction

Moisés Salmerón, Julio Ortega, Carlos García Puntonet,
Alberto Prieto, and Ignacio Rojas

Department of Computer Architecture and Computer Technology.
University of Granada — E.T.S. Ingeniería Informática.
Campus Aynadamar s/n. E-18071 Granada (Spain). E-mail: moises@ugr.es

Abstract. We propose an application of SVD model reduction to the class of RBF neural models for improving performance in contexts such as on-line prediction of time series. The SVD is coupled with QR-cp factorization. It has been found that such a coupling leads to more precise extraction of the relevant information, even when using it in an heuristic way. Singular Spectrum Analysis (SSA) and its relation to our method is also mentioned. We analize performance of the proposed on-line algorithm using a "benchmark" chaotic time series and a difficult-to-predict, dynamically changing series.

1 Introduction

In this work we suggest a method for improving the prediction performance of RBF (*Radial Basis Function*) models. For this purpose, the SVD and QR-cp matrix decomposition operations are used in a way similar to how usually SVD linear model reduction is performed. What distinguishes our method from SVD reduction in linear model theory, is the fact that the QR-cp step reorganizes the results of the SVD computation. This serves to identify the relevant information for prediction in the input series and also the relevant nodes in the network, thus yielding parsimonious RBF models. An important characteristic of the proposed method is also its capacity for on-line operation, although this could depend to a given degree on parallelization of the matrix routines.

1.1 SVD Model Reduction for Linear Models

SVD-based model reduction can refer to a variety of algorithms that make use of such a matrix decomposition for simplifying and reducing dimensionality in

⋆ Research partially supported by the Spanish Ministry of Science and Technology Projects TIC2001-2845, TIC2000-1348 and DPI2001-3219.
Submission Notice: This paper has been also submitted to the International Conference on Artificial Neural Networks (ICANN 2002), to celebrate in Madrid, Spain, during August 2002. Note that, as stated in the Instructions for Authors, "*IBERAMIA 2002 allows simultaneous submission of papers to other conferences, provided this fact is clearly indicated when submitting*".

model-driven forms of data adjustment. The SVD (*Singular Value Decomposition*) of a rectangular matrix $\boldsymbol{A}$ of $\mathbb{K}^{m\times n}$ (where $\mathbb{K}$ usually denotes the real or complex number field) dates from the 1960 and 1970 decades. Its computer implementation has been thoroughly analyzed both for serial [GL96,Ste01] and parallel [BCC$^+$97,BL85] computing architectures, although efforts in parallelization (specially for specific applications) are still being carried out (e.g., [SOP$^+$02]).

For a linear system the SVD computation allows to determine whether the system has full or deficient rank. Normally, one looks for the *singular values* σ_i ($i = 1, 2, \ldots, n$), and values near to (or exactly) zero account for linear dependence within the column set. A reduced basis for the subspace of $\mathbb{K}^m$ spanned by the columns in matrix $\boldsymbol{A}$ is sought of dimension equal to r, the numerical rank of the argument matrix. In this case we express $\boldsymbol{A}$ as

$$\boldsymbol{A} = \sum_{i=1}^{r} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T \quad , \tag{1}$$

where r denotes the numerical rank, and $(\boldsymbol{u}_i, \boldsymbol{v}_i)$ are the pairs of left- and right-singular vectors from the SVD computation. Solving the reduced linear system with the $\boldsymbol{A}$ in eq. (1) leads to a reduced linear model with $r < n$ parameters.

1.2 Basic Operation of Singular Spectrum Analysis

The *Singular-Spectrum Analysis* (SSA for short) methodology has been systematized recently in the 1996 monograph of Elsner and Tsonis [ET96] and the 2001 book by Golyandina et al. [GNZ01]. This is a technique that makes use of SVD for extraction of the trend, oscillatory and "structureless" noise components in time series data. We point out that the idea had been implicitly suggested in several papers before the 1990's, e.g., by Broomhead and King [BK86]. The first step in the SSA method is to set up the *trajectory matrix* $\boldsymbol{A}$ by using "moving windows" of width W along the time series values $\{f_t : t = 0, 1, \ldots, N\}$; that is:

$$\boldsymbol{A} \equiv [\boldsymbol{A}_1^T \boldsymbol{A}_2^T \cdots \boldsymbol{A}_K^T]^T \quad , \tag{2}$$

where $\boldsymbol{A}_j = (f_j, f_{j+1}, \ldots, f_{j+W-1})$. The structure of the resulting $\boldsymbol{A}$ is that of a *Hankel matrix* [Bjö96]. A total of $K = N - W + 1$ windows are needed to "cover" all of the N time series values. The second step is the SVD of this trajectory matrix. The series is then approximately reconstructed by an expansion

$$f_t = \sum_{k=1}^{m} f_t^{(k)}, \;\; t = 1, 2, \ldots, N \quad , \tag{3}$$

where the column index set $\{1, 2, \ldots, r\}$ is partitioned as a set of disjoint classes $\{G_k, k = 1, 2, \ldots, m\}$ such that

$$\boldsymbol{A} = \sum_{i=1}^{r} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T = \sum_{k=1}^{m} \sum_{s\in G_k} \sigma_s \boldsymbol{u}_s \boldsymbol{v}_s^T \quad ; \tag{4}$$

and $f_t^{(k)}$ is a smoothed time series obtained from diagonal averaging within the outer product matrices belonging to the G_k group [GNZ01]. We notice that "traditional" SVD corresponds to the special case $G_k = \{k\}, \forall\, k = 1, 2, \ldots, r$. It remains a difficult problem that of determining the "optimal" partition of indexes, such that (a) excessive smoothing is avoided, and (b) the additive-efect implicit assumption in eq. (3) makes sense; that is, it leads to interpretable or more-or-less identifiable effects for every $f_t^{(k)}$ series. These are contradictory requirements. For, if a smaller number of groups are used for the partition (for the sake of identifiability), smoothing has to operate on a higher number of outer products, which in turn can render the method useless.

2 On-line RBF Model that Uses SVD and QR-cp

SSA is a essentially model-free technique [GNZ01]. It does not tell us how to perform adjustment of the series within each index group of $\boldsymbol{u}_s \boldsymbol{v}_s^T$ matrices, it just smooths out each component series $f_t^{(k)}$. Another drawback of the SSA technique is its lack of consideration of possible on-line adjustment, this being partially due to its birth within the statistics community. It has been claimed that SSA offers results for moderate N that resemble very well its asymptotically predicted behavior, but no formal proof has been given to date assuring such an hypothesis. Other "more classical" time series decomposition methods (such as those arising from the Wold Decomposition Theorem [Pol99]) are therefore most likely to be used in practice by theoretically inclined researchers. For on-line prediction, however, neither SSA nor traditional decomposition are useful. But it can be suspected that the SVD role within SSA might be incorporated in a similar way into powerful on-line models; that is exactly what we propose in the following.

In the method we describe here (thoroughly described in [SOPP01]), a RBF neural network performs on-line modelling of an input time series. A reduced model is constructed both for the input delays and for the hidden layer of locally receptive Gaussian activation units [MD89]. This is accomplished by setting up successive trajectory matrices in the way shown in eq. (2), but coupling the SVD with a QR-cp (QR with *column pivoting* [Bjö96]) phase in which we compute

$$\bar{\boldsymbol{V}} \boldsymbol{P} = \boldsymbol{Q} [\boldsymbol{R}_1 \boldsymbol{R}_2] \ , \tag{5}$$

where $\bar{\boldsymbol{V}} = [\boldsymbol{v}_1 \boldsymbol{v}_2 \cdots \boldsymbol{v}_r]$ (r is still the numerical rank of $\boldsymbol{A}$) and the $\boldsymbol{v}_i$'s are the r leftmost right singular vectors of $\boldsymbol{A}$. Matrix $\boldsymbol{Q}$ is $r \times r$ and orthogonal, while $\boldsymbol{R}_1$ and $\boldsymbol{R}_2$ are upper-triangular $r \times r$ and $(n - r) \times r$ matrices, respectively. The point here is the consideration of the $\boldsymbol{P}$ permutation matrix as indicating the relevant columns in matrix $\boldsymbol{A}$. A justification for doing so was heuristically given in [GL96]. This has been theoretically proven correct in more detail in one of the authors' PhD thesis [Sal01].

We suggest that successive matrices $\boldsymbol{A}$ be set up from the input stream of time series values, on the one hand, and from the RBF hidden nodes activation values, on the other. In the case of input matrices, SVD and QR-cp would determine the relevant input lags, which are precisely those corresponding to the selected columns. In the case of RBF activations, a *pruning* routine might use SVD and QR-cp to select the relevant nodes, which are those associated with the selected columns in successive activation trajectory matrices. All this can be done on-line, provided that enough computing resources are available. Unfortunately, the QR-cp does not allow to exploit the Hankel structure in trajectory matrices [Bjö96]. Having this in mind, procedures have been devised recently that concentrate effort on parallelization of the QR-cp over relatively inexpensive concurrent computing platforms, such as "clusters" of computers [SOP$^+$02].

3 Experimental Results

We evaluated the performance of the proposed on-line prediction algorithm. A window length $W = 18$ and $K = 2W$ were used in the case of input matrices, and $W = M$ and $K = 2M$ were used for successive neural activation trajectory matrices, where M denotes the actual number of RBF nodes in the neural network. The network starts empty and a preliminary input lag model is deter-

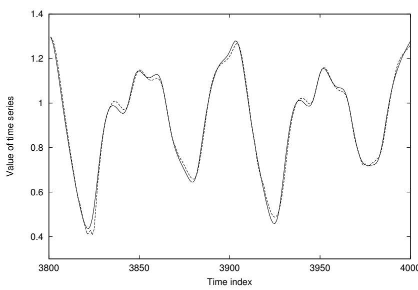

Fig. 1. Prediction of the Mackey-Glass chaotic time series, using our method based on RBF model reduction by SVD and QR-cp matrix decompositions.

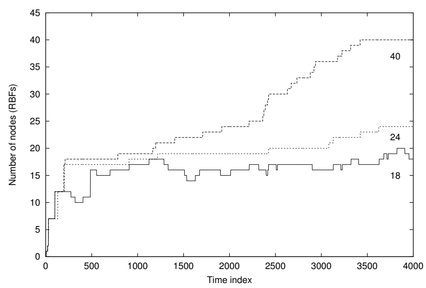

Fig. 2. Comparison of the number of neurons. Our SVD and QR-cp based model attains only 18 neurons; other versions of Platt's algorithm (that doesn't take into account model reduction) need up to 40 neurons.

mined from the first $3W - 1$ input values. Some preliminary RBF nodes are then allocated according to the found lags. When given conditions on the error and position of the RBF centers hold (concretely, those given by the widely-known Platt's algorithm [Pla91]), a new trajectory matrix is determined from past values (with respect to the actual instant) to yield an updated input structure. Five randomly positioned RBFs were added with each increase of the input model dimension. The pruning routine based on the neural activation trajectory matrices is continuously looking for non-relevant nodes. A RBF node is pruned if it is not included in the selected subset of activation columns, for more than 20 consecutive iterations of the algorithm. So, parsimony in the model compensates a bit for the extra computation needed by the matrix decompositions.

Results for prediction of the "benchmark" Mackey-Glass chaotic time series are depicted in Figures 1 and 2. The converging input model determined quickly after the first 500 on-line iterations was the same (lags $0, 6, 12, 18$) as the one proposed by recent chaos-theoretic studies (e.g., [WG93]). In Figure 2 it is observed that our model attains only 18 neurons, compared to up to 40 for the Platt method alone (without SVD and QR-cp). For the dynamically changing

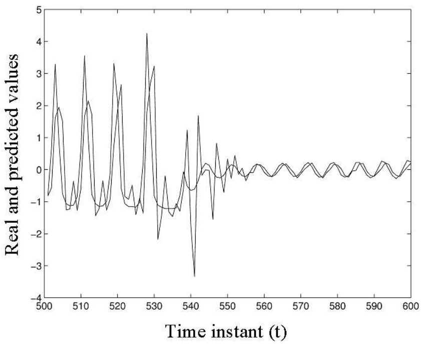

Fig. 3. Results of prediction for the laser time series in [WG93]. Notice the quick adaptation to the abrupt change in the series dynamics.

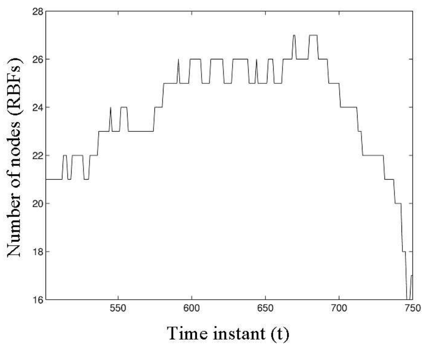

Fig. 4. Evolution of the number of nodes (RBFs) for prediction of the laser series in [WG93]. The effect of the pruning routine based on the SVD and QR-cp decompositions is evident when a change in the series dynamics occurs.

laser series from reference [WG93], very good results are also obtained. In this case the input model $(0, 2)$ is very simple, and our method identified it in as few as six iterations. The interesting point in this case is to see how the number of RBF nodes adapts to a unexpected change in the series dynamics. Figure 3 illustrates the effect of the allocation and pruning strategies, while Figure 4 shows that the prediction, although maybe not as exact as in the Mackey-Glass case, quickly adapts to the overall behavior of the series.

4 Conclusions

We have suggested in this paper a way of incorporating the SVD operation over trajectory matrices (drawn from the SSA research community), coupled with QR-cp decomposition, as an aid to constructing better predictive RBF models. The on-line algorithm suggested is conceptually simple, although some future work on relating the theory of RBF approximations with the matrix decompositions could be useful for better insight into the potential of this and related methods for on-line time series prediction. The experiments shown give an indication of the kind of applications that could benefit from such a kind of unifying theory.

References

[BCC+97] L.S. Blackford, J. Choi, A. Cleary, E. D'Azevedo, J. Demmel, I. Dhillon, J. Dongarra, S. Hammarling, G. Henry, A. Petitet, K. Stanley, D. Walker, and R.C. Whaley. "ScaLAPACK User's Guide". SIAM Publications. Philadelphia, U.S.A. (1997).

[Bjö96] Å. Björck. "Numerical Methods for Least Squares Problems". SIAM Publications. Philadelphia, U.S.A. (1996).

[BK86] D.S. Broomhead and G.P. King. Extracting qualitative dynamics from experimental data. *Physica D* **20**, 217–236 (1986).

[BL85] R.P. Brent and F.T. Luk. The solution of singular value and symmetric eigenvalue problems on multiprocessor arrays. *SIAM Journal on Scientific and Statistical Computing* **6**, 69–84 (1985).

[ET96] J.B. Elsner and A.A. Tsonis. "Singular Spectrum Analysis: A New Tool in Time Series Analysis". Plenum Press. New York, U.S.A. (1996).

[GL96] G.H. Golub and C.F. Van Loan. "Matrix Computations". The Johns Hopkins University Press. Baltimore, Maryland, U.S.A., third edition (1996).

[GNZ01] N. Golyandina, V. Nekrutkin, and A. Zhigljavsky. "Analysis of Time Series Structure: SSA and Related Techniques". Chapman & Hall/CRC Press. Boca Raton, Florida, U.S.A. (2001).

[MD89] J. Moody and C.J. Darken. Fast learning in networks of locally-tuned processing units. *Neural Computation* **1**, 281–294 (1989).

[Pla91] J. Platt. A resource allocating network for function interpolation. *Neural Computation* **3**, 213–225 (1991).

[Pol99] D.S.G. Pollock. "A Handbook of Time Series Analysis, Signal Processing and Dynamics". Academic Press. London, U.K. (1999).

[Sal01] M. Salmerón. "Time Series Prediction with Radial Basis Neural Network and Matrix Decomposition Techniques". PhD thesis (in spanish). Department of Computer Architecture and Computer Technology. University of Granada, Spain (2001).

[SOP+02] M. Salmerón, J. Ortega, A. Prieto, C.G. Puntonet, M. Damas, and I. Rojas. High-performance time series prediction in a cluster of computers. *Concurrency and Computation: Practice & Experience (submitted)* (2002).

[SOPP01] M. Salmerón, J. Ortega, C.G. Puntonet, and A. Prieto. Improved RAN sequential prediction using orthogonal techniques. *Neurocomputing* **41**(1–4), 153–172 (2001).

[Ste01] G.W. Stewart. "Matrix Algorithms—Volume II: Eigensystems". SIAM Publications. Philadelphia, U.S.A. (2001).

[WG93] A.S. Weigend and N.A. Gershenfeld. "Time Series Prediction: Forecasting the Future and Understanding the Past". Addison-Wesley. Reading, Massachusetts (1993).