4

FINAL PROJECT REPORT

Contract N00014-87-k-0284

Principal Investigators: Professor Yih-Fang Huang
Professor Ruey-wen Liu
Department of Electrical & Computer Engineering
University of Notre Dame
Notre Dame, IN 46556

Submission Date: March 8, 1989

Department of
ELECTRICAL AND COMPUTER ENGINEERING

UNIVERSITY OF NOTRE DAME, NOTRE DAME, INDIANA

4

FINAL PROJECT REPORT
Contract N00014-87-k-0284

Principal Investigators: Professor Yih-Fang Huang
Professor Ruey-wen Liu
Department of Electrical & Computer Engineering
University of Notre Dame
Notre Dame, IN 46556

Submission Date: March 8, 1989

Accession For		
NTIS GRA&I	X	
DTIC TAB	☐	
Unannounced	☐	
Justification		
By		
Distribution/		
Availability Codes		
Dist	Avail and/or Special	
A-1		

Unclassified, distribution unlimited

Table of Contents

I. EXECUTIVE SUMMARY

This report summarizes accomplishments and overall progress made during the period of May 1, 1987 to December 31, 1988. It is the final project report for the research project sponsored by the Office of Naval Research under Contract N00014-87-k-0284. It enlists publications and professional activities of the Principal Investigators during this period. A selected subset of the Principal Investigators' publications is included in the Technical Appendices.

Recent advances in integrated circuits and signal processing technology have prompted the need of algorithms with higher degrees of *modularity* and *pipelineability*, resulting in higher degrees of *concurrency* and *machine perception*. It is generally believed in the research community that higher degrees of concurrency and machine perception will bring forth many breakthroughs in many signal processing areas, particularly in adaptive systems, speech and image processing and recognition.

The objective of this research project is to develop an adaptive signal processing architecture with important features such as *modularity, pipelineability*, and *intelligent use of information*. The ground work upon which this research project rests is a recursive parameter estimation algorithm, i.e., the so-called OBE algorithm, which features a *discerning update* strategy. This discerning update is in sharp contrast to the *continual update* used by most existing algorithms.

The estimation algorithm has been developed with a set-theoretic framework. In particular, starting with the assumption that the underlying noise (of the system being studied) is bounded in magnitude, a recursive least-sqaures type of estimation algorithm was obtained with a discerning update strategy. An important outcome of such discerning updates is that the resulting algorithm can be implemented with two modules: an *information processor* followed by an *updating processor*. The former decides whether an update is needed, and the decision is based on the evaluation of the *"information quality" of the input data*, the *prediction error*, and the *noise bound*. It is essential that the information evaluation involves very little computational effort, which is the case here. The latter then updates the parameter estimates when the information processor decides that such is needed.

Simulation results have shown, in general, that only less than 20% of the input data are used to update the parameter estimates. This is true for most practical systems that can be modeled by autoregressive processes with exogeneous inputs

(ARX) or autoregressive moving average (ARMA) processes whose order is less than ten.

Conceptually, thanks to the modularity and to the fact that only less than 20% of input data are used to update the parameter estimates, an adaptive signal processing network may be constructed. The network will consist of a number of such modular recursive estimators, each of which is comprised of two modules, namely, the information evaluator and the updating processor. As such, the idling time of both the information evaluator and the updating processor can be reduced, thus the data throughput rate will be increased. In addition, the reliability of signal processing can be improved greatly. In essence, this type of adaptive networks will be able to process *multi-channel adaptation and filtering*, improving *reliability* and *data throughput rates.* One of the important applications for this is *adaptive array processing* in sonar systems.

In this project, several fundamental issues associated with the development of this type of networks are investigated. These issues include modelling, statistical analysis of the underlying estimation algorithm, theoretical studies on convergence issues, and finite word-length effects.

Modelling of the pipelined concurrent networking structure was investigated with the aid of discrete event models. In particular, we have devised a time-sharing type of network structure and employed Petri nets to model the data flow. Timing, synchronization, and routing of the underlying arithmetic operations and propagation of data are shown to be easily modeled by *timed Petri nets.* It was seen that the virtues of the adaptive network structure include improvements of data throughput rates, cost reduction of signal processing hardware, and improvements of reliability.

Statistical analysis was also performed for the estimation algorithm in its application to ARX parameter estimation. This study is important in the development of routing strategy of data flow and performance analysis of the network. *Asymptotic unbiasedness* of the estimates and *boundedness of estimation error covariance* have been established under very general conditions, such as whiteness and zero mean for the underlying noise.

An extension of the OBE algorithm (the EOBE algorithm) has been developed to deal with ARMA parameter estimation. The results are particularly useful for applications such as spectral estimation. The algorithm retains the feature of discerning update strategy and is shown to yield bounded *a posteriori* prediction errors without the premise of the *strictly positive real* (SPR) condition. This is in sharp contrast to the conventional algorithms such as the extended least-squares (ELS) algorithm.

It is well known that the SPR condition plays a crucial role in the analysis of the ELS algorithm. Specifically, without the SPR condition, the ELS can only be shown to have bounded *mean-square a posteriori* prediction error. In addition, simulation results have shown that performance of the EOBE algorithm is comparable to that of the ELS, in terms of the asymptotic bias of the estimates, *a posteriori* prediction error, and *a priori* prediction error; even though the EOBE used only less than 20% of the input data to update the estimates. An important point here is that implementation of the EOBE requires only that the underlying noise is bounded in magnitude. The development of this EOBE algorithm will help us in the application of the adaptive network to spectral estimation.

Implementation on finite word-length processors has been studied via simulations. In particular, the effects of roundoff error accumulation and numerical stability were studied with fixed point simulations. Based on these preliminary results, it has been seen that the OBE and the EOBE appear to be superior to the RLS and the ELS, respectively. One of the possible reasons for such encouraging results is the discerning update strategy which updates parameter estimates less frequently, thereby accumulates less roundoff errors. Another reason is imbedded in the update equations which may require more detailed analysis. Nevertheless, these results further verify our conjecture that eliminating redundant use of information, contained in the received data, would reduce the effects of roundoff errors.

In summary, our studies showed that the proposed adaptive network structure is viable. The concept of discerning update, on which the OBE algorithm and its extension (the EOBE algorithm) are based, is appealing in practice as well as in theory. The progress made in the project not only demonstrated the OBE algorithm as a viable scheme for parameter estimation, but also opened some new avenues in the subject area. The algorithm is applicable to a large class of systems for which no *a priori* statistical information is available and the resulting parameter estimates are assured of good quality. The only assumption made in the implementation and analysis of the algorithms is boundedness on the magnitude of the underlying noise. Due to the discerning update strategy of recursive estimation, the resulting adaptive system features a higher level of *machine perception*, higher degrees of modularity, and is less susceptible to finite word-length processing.

In addition to the above accomplishments, our experience in adaptive signal processing using algorithms with discerning updates have facilitated studies on artificial neural networks. In particular, a learning algorithm in neural networks with selective updates has been developed as an additional accomplishment of the project. The resulting algorithm is the so-called *selective update back propagation* (SUBP).

Recent resurgence in the field of artificial neural networks has called for much attention to *learning algorithms.* Learning capability of artificial neural networks may essentially be responsible for distinguishing those networks from conventional computing and data processing methodologies.

A commonly studied learning algorithm is the so-called *back propagation* (BP) algorithm for perceptron-like networks. The back propagation algorithm is believed to have the most potential to date for generalization. It bears a great deal of similarity to the least-mean-squares (LMS) algorithm in adaptive systems. It updates continually regardless of the benefit of such updates.

As shown by many simulation studies, learning with BP appears to be rather inefficient as the algorithm often fails to converge. As a matter of fact, in some practical pattern classification problems, it can be shown that using BP to train the network will take it farther and farther away from the desired results. We suspect that the continual updating used by the BP algorithm, which results in redundant use of data that may not be informative, is a handicap of the learning procedure.

The SUBP algorithm, employing the concept of discerning updates, appears to have good potential for circumventing these difficulties. Further investigations on SUBP will be needed.

II. INVITED LECTURES/SEMINARS

II.A. Invited Seminars at Academic Institutions:

- Y. F. Huang:

  1. "Intelligent Learning of Artificial Neural Networks," June 27, 1988. Institute of Information Science, Sinica Academia, Taiwan.
  2. "Discerning Update in Recursive Parameter Estimation," Nov. 3, 1988. Department of Electrical Engineering, Princeton University, Princeton, N.J.

- R. Liu:

  1. "Neural Networks – A New Breed of Computers", June 27, 1988. Institute of Information Science, Sinica Academia, Taiwan.
  2. "Neural Computing", October 10, 1988. Fu-dan University, Shanghai, China.
  3. "Neural Networks: Architecture and Learning", October 13-14, 1988. Chinese Academy of Science, Beijing, China.
  4. "Neural Networks: Architecture and Learning, October 21, and 24, 1988. National Taiwan University, Taipei, Taiwan.

II.B. Invited Conference Presentations

- Y. F. Huang:
  1. Y. F. Huang and R. Liu,
     "A High Level Parallel-Pipelined Network Architecture for Adaptive Signal Processing", *Twenty-Sixth IEEE Conf. on Decision and Control*, Los Angeles, CA, December 9-11, 1987.
  2. A. K. Rao and Y. F. Huang,
     "Recursive ARMA Parameter Estimation with a Discerning Update Strategy - Finite Precision Effects," *ComCon-88, Advances in Communication and Control Systems*, Baton Rouge, LA, October 19-21, 1988.
  3. V. C. Soon and Y. F. Huang,
     "Applications and Analysis of Second Order Artificial Neural Networks," *Twenty-Seventh IEEE Conf. on Decision and Control*, Austin, TX, December 7-9, 1988.
- R. Liu:
  1. Qiu Huang and R. Liu,
     "Diagnosis of Piecewise-Linear Circuits", *IEEE International Symposium on Circuits and Systems*, Philadelphia, PA, May 4-6, 1987.

2. Vijay Raman and R. Liu,
   "Stabilization of Chaos – An Algebraic Theory", *IEEE International Symposium on Circuits and Systems*, Helsinki, Finland, June 6-8, 1988.

3. D. Graupe and R. Liu,
   "Neural Networks for Medical Signal Processing", *27th IEEE Conf. on Decision and Control*, Los Angeles, CA, December 9-11, 1988.

III. PROFESSIONAL ACTIVITIES

- Y. F. Huang:
  1. Chaired a Session entitled "Estimation," at the 22nd *Annual Conference on Information Sciences and Systems*, Princeton University, Princeton, N.J., March, 1988.
  2. Organized a Special Session entitled, "Adaptive Signal Processing," at the ComCon-88, Advances in Communications and Control Systems, Baton Rouge, LA, October 1988.

- R. Liu:
  1. Organized a special session entitled, "Advanced Topics in Nonlinear Theory, Testing and Robotics", *IEEE International Symposium on Circuits and Systems*, Philadelphia, PA, May 1987.
  2. Organized two special sessions entitled, "Dynamics of Discrete Events Systems I & II", *IEEE Conf. on Decision and Control*, Los Angeles, CA, Dec. 1987.
  3. Organized a special session entitled, "Nonlinear Behavior of Neural Networks", *IEEE International Symposium on Circuits and Systems*, Helsinki, Finland, June 7-9, 1988.
  4. Organized a special session entitled, "Advanced Nonlinear Theory", *International symposium on Circuits and Systems*, Helsinki, Finland, June 7-9, 1988.
  5. Associate Editor, *IEEE Transactions on Circuits and Systems*, June 1, 1985 - May 31, 1987.
  6. Member of Program Committee, *1988 International Symposium on Communication and Control*, Baton Rouge, LA, Oct. 1988.
  7. Member of Program Committee, *IEEE International Symposium on Circuits and Systems*, Portland, OR, May 9-11, 1989.
  8. Member of Administrative Committee of the IEEE Circuits and Systems Society, 1987-1989.

IV. THESES AND DISSERTATIONS DIRECTED

The following is a list of masters theses and doctoral dissertations that are sponsored, in part, by the grant.

- Y. F. Huang:

  1. Ashok K. Rao (M.S.)
     Thesis Title: Applications and Extensions of a Novel Recursive Estimation Algorithm with Selective Updating. (Graduation Date: August,1987)

  2. James C. Francis (M.S.)
     Thesis Title: Distributed Detection: A Totally Neyman-Pearson Optimal Design Scheme. (Graduation Date: January, 1988)

  3. Shih-Chi Huang (M.S.)
     Thesis Title: Learning with Selective Updates for Artificial Neural Networks. (Graduation Date: August, 1988)

  4. Victor C. Soon (M.S.)
     Thesis Title: On the Capacity of Perceptron-Like Neural Networks. (Graduation Date: January, 1989)

  5. Ashok K. Rao (Ph.D.) Dissertation Title: Recursive Estimation for ARMA Parameters with Selective Updates. (Expected Graduation Date: August 1989).

- R. Liu:

  1. Qiu Huang (Ph.D.)
     Thesis Title: A Novel Approach to CAD for Large-Scale Piecewise Linear Systems, (Graduation Date: May, 1988).

  2. Zhi-hong Chai (M.S.)
     Thesis Title: Complexity of Inversion of Large Matrices. (Expected Graduation Date: December, 1989).

V. PUBLICATIONS

V.A. Invited Conference Publications

*1. Y. F. Huang,
"A Modular Recursive Estimator for Adaptive Signal Processing", *Proc. of the Fifth International Conf. on Systems Engineering*, Dayton, Ohio, pp. 481-484, 1987.

*2. Y. F. Huang and R. Liu
"A High Level Parallel-Pipelined Network Architecture for Adaptive Signal Processing", *Proc. of the 26th IEEE Conf. on Decision and Control*, Los Angeles, CA, pp. 662-667, 1987.

*3. A. K. Rao and Y. F. Huang
"Recursive ARMA Parameter Estimation with a Discerning Update Strategy - Finite Precision Effects," *Proc. of ComCon-88, Advances in Communication and Control Systems*, Baton Rouge, LA, 1988.

*4. V. C. Soon and Y. F. Huang
"Applications and Analysis of Second Order Artificial Neural Networks," *Proc. of the 27th IEEE Conf. on Decision and Control*, pp. 348-349, Austin, TX, Dec., 1988.

5. Qiu Huang and R. Liu
"Fault Diagnosis of Piecewise-Linear Systems", *Proc. IEEE ISCAS*, pp. 418-421, May 4-7, 1987.

6. V. Raman and R. Liu
"Stabilization of Chaos - An Algebraic Theory", *Proc. ISCAS*, pp. 1981-82, June 7-9, 1988.

7. D. Graupe and R. Liu
"Applications of Neural Networks to Medical Signal Processing", *Proc. CDC*, pp. 343-347, 1988

*8. R. Liu, L. Tong and J. Deng
"An On-line Learning Neural Network for Discrimination of Walk Functions in Paraplegics", *1989 IEEE ISCAS*, (to appear).

9. R. Liu, L. Tong and L. Zhang
"A Necessary and Sufficient Condition for the Testability of Hybrid Circuits", *1989 IEEE ISCAS*, (to appear).

V.B. Refereed Conference Publications

*1. V. C. Soon and Y. F. Huang
"Artificial Neural Networks with Second Order Discriminant Functions", *Proc. of the 22nd Annual Conf. on Inform. Sciences and Systems*, pp. 308-313, Princeton, N.J., March 1988.

*2. A. K. Rao and Y. F. Huang
"Statistical Properties of a Novel Recursive Estimation Algorithm with Information-Dependent Updating", *Proc. of 1988 International Conf. on Acoustics, Speech and Signal Proc.*, pp. 2436-2439, New York, N.Y. 1988.

*3. A. K. Rao and Y. F. Huang
"ARMA Parameter Estimation Using a Novel Recursive Estimation Algorithm with Selective Updating," abstract appeared in *Abstracts of Papers, 1988 IEEE International Symp. on Inform. Theory*, pp. 74-75, Kobe, Japan, 1988.

4. M. Ammar and Y. F. Huang
"Qualitative Analysis of Quantizers for Signal Detection," pp. 173-182, C. I. Byrnes, C. F. Martin, and R. E. Sacks, eds., Elsevier Science Pub. B. V., 1988.

*5. A. K. Rao, Y. F. Huang, and S. Dasgupta
"An Extended OBE Algorithm for ARMA Parameter Estimation," *Proc. 26th Annual Allerton Conf. on Commun., Control, and Computing*, pp. 229-238, University of Illinois, Urbana-Champaign, September 28-30, 1988.

*6. Qiu Huang and R. Liu
"A New Efficient Algorithm for Analysis of Piecewise-Linear Circuits", *1989 IEEE ISCAS*, (to appear).

V.C. Journal Publications

*1. A. K. Rao, Y. F. Huang and S. Dasgupta
"ARMA Parameter Estimation Using a Novel Recursive Estimation Algorithm with Selective Updating," submitted for publication, revised.

2. M. Ammar and Y. F. Huang
"An M-Interval Quantizer-Detector Based on Statistical Moments," submitted for publication, revised.

3. Y. F. Huang and E. Y. Zhang
"Echo Cancellation in Full-Duplex Systems Using a Modified Stochastic Gradient Estimation Method", submitted for publication.

*4. S. C. Huang and Y. F. Huang
"Learning with a Selective Update Strategy for Neural Networks," submitted for publication.

*5. Qiu Huang and R. Liu
"A Simple Algorithm for Finding All Solutions of Piecewise-Linear Networks", *IEEE Trans. on Circuits and Systems*, (to appear).

6. H. H. Ammar, Y. F. Huang and R. Liu
"Hierarchical Models for Systems Reliability, Maintainability, and Availability", *IEEE Trans. on Circuits and Systems*, pp. 629-639, June 1987.

* Supported, in part, by the contract N00014-87-k-0284.

Technical Appendices

Publication V.A.1.

Fifth International Conference on Systems Engineering

September 9-11, 1987

Sponsored by

Department of Electrical Systems Engineering
Wright State University

Cosponsored by

IEEE Aerospace and Electronic Systems Society

Dayton Section IEEE

Holiday Inn Conference Center/I-675
Fairborn, Ohio

IEEE Catalog No.: 87CH2480-2

A MODULAR RECURSIVE ESTIMATOR FOR ADAPTIVE SIGNAL PROCESSING

Y.F. Huang
Department of Electrical and Computer Engineering
University of Notre Dame
Notre Dame, IN 46556

Abstract - This paper addresses a newly developed recursive estimation algorithm which features a discerning update strategy for parameter estimates. This is in contrast to conventional algorithms' continual update of parameter estimates. Of particular interests here is its higher degree of modularity, as a result of the discerning update strategy, and its potential for pipelined adaptive signal processing architecture.

INTRODUCTION

Effectiveness of recursive estimation is known to be extremely critical in the design and implementation of adaptive systems. Issues such as accuracy of parameter estimates, speed of convergence, and numerical stability of recursive estimation algorithms have received a great deal of attention throughout the years with numerous books and journal articles on these subjects, see, e.g., [1-7]. Recently, due to the advent of the very large scale integrated (VLSI) circuits technology, there have been intense interests on constructing more efficient signal processors to be compatible with such technology [8-11]. Algorithms which can be implemented with higher degrees of modularity and pipelineability (hence higher degrees of concurrency) will be very much in demand. In addition, to further improve the effectiveness of data processing, estimation algorithms which can extract information intelligently will prove to be practically appealing for adaptive signal processing systems. There exist immediate applications of such algorithms to many modern communication systems, such as ADPCM for speech processing and adaptive echo cancellation in telecommunications.

A common feature of most, if not all, existing recursive estimation algorithms is the continual update of parameter estimates without regard to the benefits provided. Thus even if a new measurement contains no fresh information and even if its use fails to result in any improvement in the quality of estimation, the update does not cease. A classical argument for that is to assume that the underlying signal sequence is an independent one. It thus implies that some innovative information can be acquired from each new measurement. In practice, this assumption often fails to hold as data from many natural sources, such as speech processes and underwater signal processes, have exhibited dependent characteristics. Another issue of much concern is that of numerical stability. Experience in digital filter theory over the last two decades has shown that special attention must be paid to problems arising from finite word-length implementations [9]. Typical such problems are propagation of round-off errors, overflow oscillations, and limit cycles. The continual update strategy usually involves more computational complexity, thus is more likely to result in numerical instability. In addition, the time delay due to updating may result in deviations from the presumed model on which the estimation process is based. This not only destroys the purpose of estimation, but could also cause instability of the entire system.

In short, the continual update strategy of existing recursive algorithms is not only redundant, but is often detrimental. Recently, a recursive estimation algorithm with a discerning update strategy has been developed [12-15]. The fundamental concept of the algorithm is based on the realization that, in reality, not every received datum contains sufficient innovative

CH2380-? 87/0000-0481 $1.00 © 1987 IEEE

information to yield improvements on the parameter estimates. Thus at every data point, a decision is made on whether the received data contains enough fresh information to improve the parameter estimates. As shown in Fig. 1, the resulting estimator consists of two modules, an information processor followed by an updating processor. The former evaluates the information content of the received data and decides whether an update is necessary. The updating processor thus computes a new parameter estimate only when the information processor decides that an update is needed.

PROBLEM FORMULATION

Many processes commonly encountered in the area of digital signal processing can be formulated by the following autoregressive exogeneous input (ARX) model:

$$y_k = \sum_{i=1}^{n} a_i\, y_{k-i} + \sum_{j=0}^{m} b_j u_{k-j} + v_k \tag{1}$$

The above equation can be simplified as

$$y_k = \theta^{*T} x_k + v_k \tag{2}$$

where $\theta^{*T} \triangleq [a_1, \ldots, a_n, b_0, b_1, \ldots, b_m]$ and $x_k^T \triangleq [y_{k-1}, \ldots y_{k-n}, u_k, \ldots u_{k-m}]$. The problem of recursive estimation is essentially concerned with the evaluation of an estimate of θ^*, at every time instant k, with the given measurement (y_k, x_k) and the estimate at time $k-1$.

The algorithm proposed in [12] and its extensions [14,15] are derived on the basis of a boundedness assumption on v_k. In particular, assuming that

$$v_k^2 \le \gamma^2 \quad \text{for all } k. \tag{3}$$

then (2) and (3) together yield

$$(y_k - \theta^{*T} x_k)^2 \le \gamma^2 \tag{4}$$

Let S_k be a subset of R^{n+m+1} defined by

$$S_k = \{\theta: (y_k - \theta^T x_k)^2 \le \gamma^2, \gamma \in R^{n+m+1}\} \tag{5}$$

Note that S_k is a convex polytope in R^{n+m+1}. At any instant k, consider the intersection of the sequence of the polytopes $S_1, \ldots, S_k$. It must contain the modeled parameter θ^*. However, formulation of this intersection set is analytically complex. A better alternative is to consider ellipsoids which bound it. Clearly, such ellipsoids must also contain θ^*.

Another type of processes also commonly encountered can be modeled by the following autoregressive-moving-average exogeneous input (ARMAX) model:

$$y_k = a_1 y_{k-1} + \cdots + a_n y_{k-n} + b_1 u_{k-1} + \cdots + b_m u_{k-m} + v_k + c_1 v_{k-1} + \cdots + c_r v_{k-r} \tag{6}$$

where $\{v_k\}$ is a white noise sequence. Similarly to (2), (6) can be rewritten as

$$v_k = y_k - \theta^{*T} x'_k \tag{7}$$

where $\theta^{*T} \triangleq [a_1, \ldots, a_n b_1, \ldots, b_m, c_1, \ldots, c_r]$, is the vector of true parameters. At time k if an estimate of θ^* is available, v_k could be estimated by ε_k according to:

$$\varepsilon_k = y_k - \theta'^T_k x'_k \tag{8}$$

with

$$x'_k = [\, y_{k-1}, \ldots, y_{k-n}, u_{k-1}, \ldots, u_{k-m}, \varepsilon_{k-1}, \ldots, \varepsilon_{k-r}]^T$$

and θ'_k being the estimate of the true parameter θ^*. As such, one may obtain analogous formulations for the convex polytope as (5), as well as for the bounding ellipsoids.

THE ALGORITHM

Recursive algorithms have been derived based on both the ARX and the ARMAX models. The recursive algorithm is initialized with a sufficiently large ellipsoid which covers all possible values of θ^*. After (y_1, x_1) is acquired, one is to find an ellipsoid which bounds the intersection of the initial ellipsoid and S_1, and which is in some sense "optimal". Such an ellipsoid is denoted by E_1. By the same token, one can proceed to obtain a sequence of optimal bounding ellipsoids (OBE) $\{E_k\}$. The estimate for θ^*, at the k^{th} instant, is defined to be the center. Thus the problem of recursive estimation is converted to one of finding a sequence of OBE's. The striking feature of this procedure is that, in general, the OBE needs to be updated only when a new observation set intersects it in such a way that the resulting intersection set is "significantly smaller" than the previous one. As shown in [12], the decision to update is based on the result of the optimization procedure. This essentially results in an "information evaluation procedure", as shown in Fig. 1.

The OBE may be chosen to be the one which has the minimum volume [12,13]. On the other hand, one may also choose the OBE as the one for which a normalized bound on the estimation error is minimized [14]. The information evaluation procedure involves much less computation, compared to updating of the parameter estimates. In fact, the information evaluation of [14] involves even less computation than its counterpart of [12].

DISCUSSIONS

According to our simulation experience, only 20% of received data are needed for updating estimates and the results are as accurate as those of the recursive least-squares. This is true for almost all cases in which the number of parameters to be estimated is less than ten. This result demonstrates tremendous redundancy involved in conventional recursive estimation algorithms. Other features of the algorithm include a better tracking capability [14] for slowly time-varying parameters, as well as cessation of updating thus improves numerical stability and the accuracy of the final estimates.

As a result of the modular structure, there exists a great potential for constructing a pipelined architecture. Furthermore, due to the simplicity of computation needed by the information evaluation procedure, one may be able to increase significantly the data-throughput rate by constructing a time-sharing type of processor networks. This is a subject of some on-going research projects.

References

[1] G.C. Goodwin and K.S. Sin, Adaptive Filtering, Prediction and Control. Englewood Cliffs, NJ: Prentice-Hall, 1982.

[2] R.M. Johnston and B.D.O. Anderson, "Exponential convergence for recursive least-squares with exponential forgetting factor-adaptive control," *Systems and Control Letters*, Vol. 2, pp. 69-76, 1982.

[3] B.D.O. Anderson and C.R. Johnson, "Exponential convergence of adaptive identification and control algorithms," *Automatica*, Vol. 18, pp. 1-13, 1982.

[4] L. Ljung and T. Soderstrom, Theory and Practice of Recursive Identification. Cambridge, Mass.: MIT Press, 1983.

[5] B. Widrow and E. Walach, "On the statistical efficiency of the LMS algorithm with nonstationary inputs," *IEEE Trans. Inform. Theory*, Vol. IT-30, pp. 211-221, 1984.

[6] H. Lev-Ari, T. Kailath, and J.M. Cioffi, "Least-squares adaptive lattice and transversal filters: A unified geometric theory," *IEEE Trans. Inform. Theory*, Vol. IT-30, pp. 222-236, 1984.

[7] Simon Haykin, Adaptive Filter Theory. Englewood-Cliffs, New Jersey: Prentice-Hall, 1986.

[8] H.T. Kung, "Why systolic architectures?," *IEEE Computer*, Vol. 25, pp. 37-46, 1982.

[9] S.Y. Kung, H.J. Whitehouse, and T. Kailath, Eds., VLSI and Modern Signal Processing. Englewood-Cliffs, New Jersey: Prentice-Hall, 1985.

[10] S.K. Rao and T. Kailath, "VLSI arrays for digital signal processing: Part I - A Model identification approach to digital filter realizations," *IEEE Trans. Circuits and Syst.*, Vol. CAS-32, pp. 1105-1118, 1985.

[11] H.H. Lu, E.A. Lee, and D.G. Messerschmitt, "Fast recursive filtering with multiple slow processing elements," *IEEE Trans. Circuits and Systs.*, Vol. CAS-32, pp. 1119-1129, 1985.

[12] Y. F. Huang, "A recursive estimation algorithm using selective- updating for spectral analysis and adaptive signal processing," *IEEE Trans, Acoust., Speech, Signal Processing.*, Vol. ASSP-34, pp. 1331-1334, 1986.

[13] Y.F. Huang and M. Yeh, "Adaptive hybrid balancing based on a novel recursive estimation algorithm and spectral analysis," *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. ASSP-35, No. 3, pp. 397-399, 1987.

[14] S. Dasgupta and Y.F. Huang, "Asymptotically convergent modified recursive least-squares with data-dependent updating and forgetting factor for systems with bounded noise," *IEEE Trans. Inform. Theory*, Vol. IT-33, No. 3, pp. 383-392, 1987.

[15] Y.F. Huang and A.K. Rao, "Application of a recursive estimation algorithm with information-dependent updating to ARMAX models and ARMA models with unknown inputs," *Proc. 1987 IEEE International Conf. Acoust., Speech, Signal Processing*, pp. 1007-1010, April, 1987.

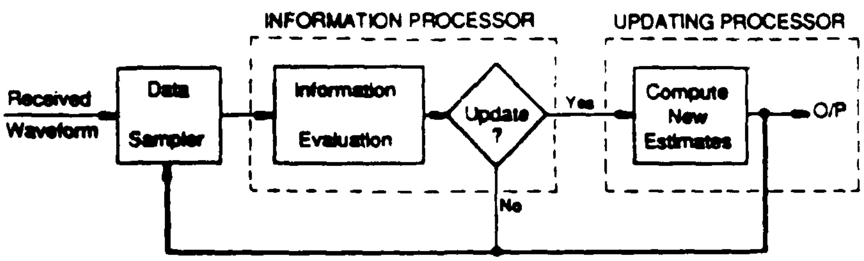


Figure 1. A Modular Recursive Estimator

ACKNOWLEDGEMENTS

This work has been supported in part by the National Science Foundation under Grant ECS-8505218.

Publication V.A.2.

PROCEEDINGS OF THE

26th IEEE CONFERENCE ON DECISION AND CONTROL

DECEMBER 9-11, 1987
WESTIN CENTURY-PLAZA HOTEL
LOS ANGELES, CALIFORNIA

IEEE
Control
Systems
Society

VOLUME 1 OF 3

87CH2505-6

A HIGH LEVEL PARALLEL-PIPELINED NETWORK ARCHITECTURE FOR ADAPTIVE SIGNAL PROCESSING

Y.F. Huang and R.W. Liu
Department of Electrical and Computer Engineering
University of Notre Dame
Notre Dame, IN 46556

Abstract

This paper investigates a recently developed *modular recursive estimation* (MRE) [1,2] algorithm using discrete event models (DEM). It also proposes a concurrent, pipelined adaptive signal processing architecture based on parallel networking of these MRE's. The main feature of the MRE is a discerning update strategy for parameter estimates, in contrast to the continual update strategy of conventional algorithms. Not only does this discerning update strategy result in a higher degree of modularity, but also does it facilitate more effective use of data information.

I. INTRODUCTION

The growing need of high speed digital processing of large volumes of data has resulted in a great deal of interest in signal processing algorithms which possess important features of modularity, pipelineability, regularity, and flexibility. The advent of the very large scale integrated (VLSI) circuit technology has made available low cost signal processors with higher degrees of concurrency in computation. Design and implementation of adaptive signal processing algorithms will undoubtedly benefit from such progress.

Much work has been done to improve modularity and concurrency in the computational aspects of signal processing algorithms, see, e.g., [3-7]. Little was done to devise modular algorithms at the "higher" level. Recently, Huang [1,2] proposed a recursive estimation algorithm which features a *modular* structure, as depicted in Fig.1. In particular, the process of recursive estimation is carried out in two steps: information evaluation of the received data first, and then updating of parameter estimates. The latter proceeds only if the former decides that an update of parameter estimates is necessary. In addition to higher degrees of concurrency, this sort of schemes also have the benefit of better numerical stability.

A common feature of most, if not all, existing recursive estimation algorithms is the continual update of parameter estimates without regard to the benefits provided. Thus, even if a new measurement contains no fresh information and even if its use fails to result in any improvement in the quality of estimation, the update does not cease. A classical argument for that is to assume that the underlying signal sequence is an independent one. It thus implies that some innovative information can be acquired from each new measurement. In practice, this assumption often fails to hold as data from many natural sources, such as speech processes and underwater signal processes, have exhibited dependent characteristics.

In general, the continual update strategy results in algorithms that are numerically intensive. It is thus more likely to result in numerical instability. Furthermore, the time delay due to updating may result in deviations from the presumed model on which the estimation process is based. This not only destroys the purpose of estimation, but could also cause instability of the entire system.

In addition to the virtues discussed above, an outgrowth of this *modular* recursive estimation (MRE) is a parallel-pipelined networking structure. Note that many data samples will be rejected by the information evaluation (IE) procedure. Our simulation experience has shown that, for lower order systems (say, eighth order), only 20% of the received data are used for updating parameter estimates and the results are as accurate as those of the recursive least-squares. In fact, this number could even be smaller in some applications [8].

As shown in [1,2], it can be designed so that the computational complexity of IE is much less than that of the updating (UPD) procedure. As a result, both the IE and the UPD could involve a good amount of idle time. A viable idea is thus to consider a parallel, pipelined network configuration of such modular estimators to (multiplex) process signals from multiple channels. In this case, issues of data throughput and those of routing (or queuing) of parameter estimates become complicated and have be addressed.

The purpose of this paper is to investigate these aforementioned issues, and effectiveness of the parallel, pipelined network architecture. Realizing that data throughput and routing in signal processing are sequences of discrete events, we investigate the potential of DEM to facilitate our study.

II. A MODULAR RECURSIVE ESTIMATION ALGORITHM

This section summarizes the estimation algorithm proposed in [1,2]. Consider the following auto-regressive exogeneous input (ARX) model:

$$y_k = \sum_{i=1}^{p} a_i y_{k-i} + \sum_{j=0}^{q} b_j u_{k-j} + v_k \qquad (1)$$

$$= \theta^{*T} x_k + v_k \qquad (2)$$

where $\theta^{*T} \triangleq [a_1 \cdots a_p \; b_0 \; b_1 \cdots b_q]$ and $x_k^T \triangleq [y_{k-1} \cdots y_{k-p} \; u_k \cdots u_{k-q}]$. The problem of recursive estimation is essentially concerned with the evaluation of an estimate of θ^*, at every time instant k, with the given measurement (y_k, x_k) and the estimate at time $k-1$.

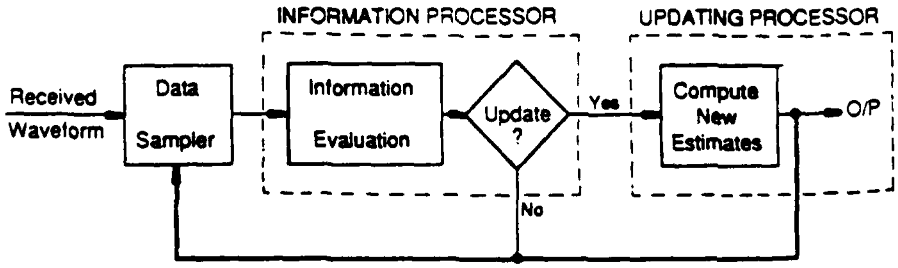


Figure 1: A modular recursive estimator.

The algorithm proposed in [1,2] is derived on the basis of a boundedness assumption on v_k, namely,

$$v_k^2 \le \gamma^2, \text{ for all } k. \tag{3}$$

Parameter estimates resulting from this algorithm is formulated as follows:

$$\theta_k = \theta_{k-1} + \lambda_k P_k x_k \delta_k \tag{4a}$$

$$P_k = \frac{1}{1-\lambda_k}[P_{k-1} - \lambda_k \frac{P_{k-1} x_k x_k^T P_{k-1}}{1-\lambda_k + \lambda_k G_k}] \tag{4b}$$

$$\sigma_k^2 = (1-\lambda_k)\sigma_{k-1}^2 + \lambda_k \gamma^2 - \frac{\lambda_k(1-\lambda_k)\delta_k^2}{1-\lambda_k + \lambda_k G_k} \tag{4c}$$

where θ_k is the parameter estimate at time k, $\delta_k \triangleq y_k - x_k^T \theta_{k-1}$ is the prediction error, and $G_k \triangleq x_k^T P_{k-1} x_k$. The variable λ_k is defined as follows:

(*i*) If $\gamma^2 \ge \sigma_{k-1}^2 + \delta_k^2$ then $\lambda_k = 0$. (5a)

(*ii*) Otherwise, $\lambda_k = min(\alpha, \upsilon_k)$; where (5b)

$$\upsilon_k = \begin{cases} \alpha & \text{if } \delta_k^2 = 0 \quad (6a) \\ \frac{1-\beta_k}{2} & \text{if } G_k = 1 \quad (6b) \\ \frac{1}{1-G_k}\left[1 - \sqrt{\frac{G_k}{1+\beta_k(G_k-1)}}\right] & \text{if } \beta_k(G_k-1) + 1 > 0 \quad (6c) \\ \alpha & \text{if } \beta_k(G_k-1) + 1 \le 0 \quad (6d) \end{cases}$$

with $\beta_k \triangleq (\gamma^2 - \sigma_{k-1}^2)/\delta_k^2$.

Note that the decision of update or not depends on the value of λ_k evaluated by Eqs. (5). In general, the algorithm can be initialized with $\theta_0=\underline{0}$ and $P_0=I/\varepsilon$, where $\underline{0}$ and I are null vector and identity matrix of appropriate dimensions, respectively; and ε is a very small positive real number.

III. A DEM FOR THE MRE

In this section, it is shown that DEM are employed to investigate the data flow and timing of event sequences associated with computation involved in the MRE. In particular, Petri nets are used to model events and states corresponding to computation at three different levels. These models provide a clairvoyant picture of blocks of computation that can be carried out concurrently. They also show propagations of data computation in information evaluator and updating processor.

Note that the models used here are the so-called (extended) timed Petri nets [9-11]. In these Petri net models, there are essentially two types of transitions: immediate transitions and timed transitions. In our study, immediate transitions signify events such as *shift*, *compare*, or simply *synchronization*. On the other hand, timed transitions stand for *computations*, such as additions or multiplications, which take non-negligible amounts of time. Depending on the level of modeling, this transition time may be deterministic or random. These models are shown in Figs. 2-5, in which bars stand for immediate transitions whereas rectangles are timed transitions.

Figure 2 depicts propagation of data computation in IE. Let τ_a and τ_m denote the times needed for each addition and for each multiplication of two real numbers, respectively. It is assumed here that an unlimited number of multipliers and adders are available concurrently and that memory access and data transfer are events which consume no time. Thus, as shown in Fig. 2, with concurrent multiplications IE can be accomplished in $2\tau_m + \tau_a[1+\kappa]$, where κ is the integer satisfying $\kappa-1 \le \log_2 n \le \kappa$ with $n=p+q+1$. A building block of matrix-vector multiplication is shown in Fig. 3, which also shows that such an operation elapses $\tau_m+\kappa\tau_a$ time units.

An ETPN model for a complete cycle of the MRE process is depicted in Fig. 4. Note that conflicting transitions, such as t_{14} and t_{22}, t_{23} and t_{25}, etc., are used to model decision making formulated by Eqs. (5) and (6). Thus, given a set of data samples, the sequence of events in IE and UPD and the corresponding timing can be obtained easily from Figs. (4a) and (4b). Note that the time consumed by each cycle of IE and UPD is determined provided that the routing of data flow is given.

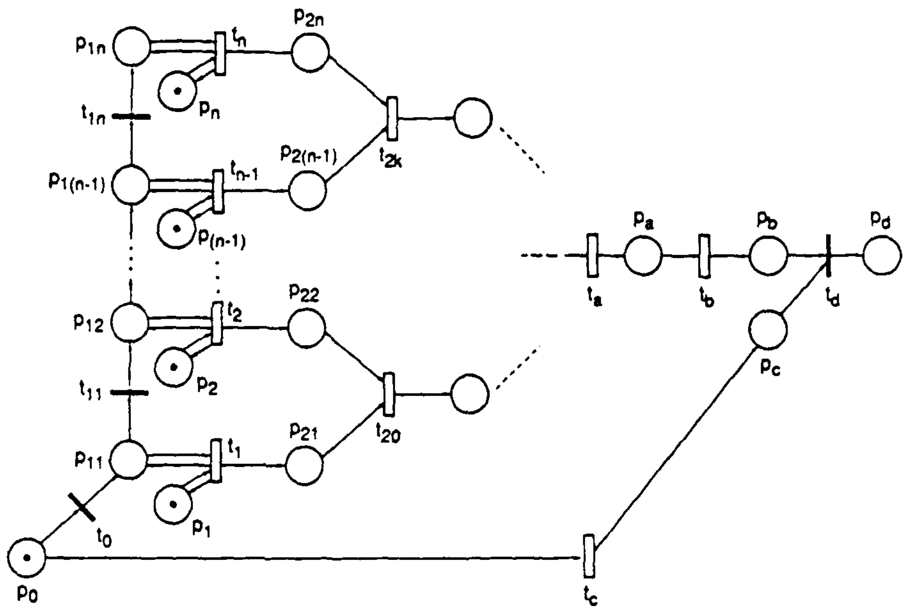
In Fig. 5, an ETPN is used to depict the MRE at the higher level. It is seen that transitions t_2 and t_3 are conflicting, and that t_4 is a randomly timed transition whereas t_5 is deterministically timed. Routing of data flow depends on probabilities of occurrences of event t_2 and t_3, which may be determined from the sampled data and the underlying system characteristics. Randomness of t_4 is a result of routing as seen in Fig. 4(a). Its firing time is a discrete random variable which assumes one of only three values, depending on the decisions formulated by Eqs.(6).

Examining Figs.(4) and (5) reveals that completion of one cycle of IE needs $2\tau_m+\tau_a(\kappa+1)$, and completion of UPD requires $5\tau_m+\tau_a(\kappa+1) + T$, where T is the random transition time of t_4 in Fig.5. In general, one can expect that the average value of T is in the order of $3\tau_m+\kappa\tau_a$. Thus if all sampled data proceed through both IE and UPD, the latter will become a *bottle-neck* of the estimation process while the former will have a significant amount of idle time. On the other hand, if a large percentage of data stops at the completion of IE, the UPD may frequently be idle. Considering either scenario, one sees the inefficient use of the signal processors. Furthermore, in some practical cases, the sampling rate may be greater than the data throughput rate of IE and UPD. In such a situation, real-time signal processing may be seriously handicapped. One of the possible methods to resolve such inefficiency and handicap is a parallel-pipelined architecture to be discussed in the next section.

IV. DESIGNING A PARALLEL-PIPELINED ARCHITECTURE FOR MULTI-CHANNEL ADAPTIVE SIGNAL PROCESSING

The objective of this section is to investigate design methodologies which may improve data throughput rate and which may result in more efficient use of signal processors. We also hope to examine the power of DEM as an aid to the design procedure. The architecture proposed here is shown in Fig.6. It is assumed here that signals are received from N_1 different channels, and that N_2 IE's and N_3 UPD's are available. In general, it may be desirable that both N_2 and N_3 are less than N_1 for cost-effectiveness purposes. Nevertheless, such may not be the case if higher throughput rates and reliability are more critical.

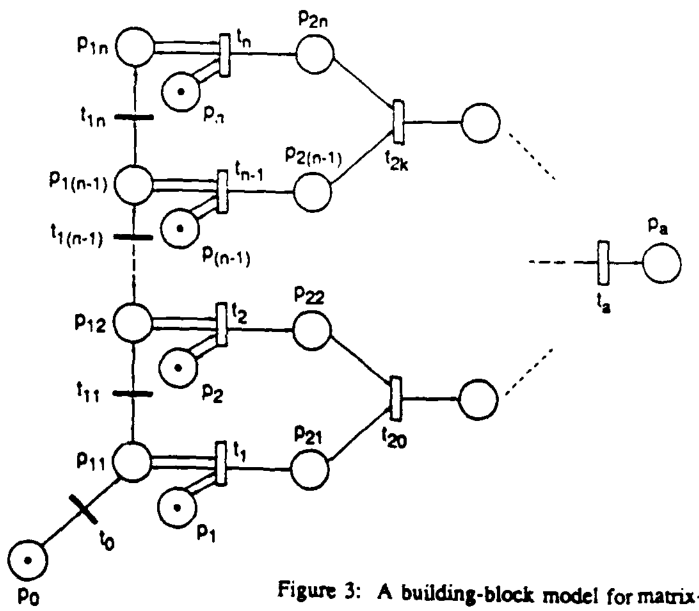
Consider the following scenario: All signals are received at the same sampling rate which is much faster than data throughput rates of both IE and UPD, and only a small percentage of samples proceeds through the UPD. Assume that sampled data and parameter estimates for all channels can be stored in buffers A and B. Thus all IE and UPD will be transparent to different channels. In essence, both buffer A and buffer B are queues of customers which are sharing and competing for a limited number of resources. An important issue here will be the design of the numbers N_2 and N_3 as functions of N_1, sampling rates, data throughput rates, channel statistics, and other application constraints. In practice, this is often an issue of tradeoffs, as opposed to optimality. Also, the queueing discipline of buffer A and buffer B need to be designed as well. In particular, there



Places	Description
p_0 :	Input data ready
$p_1,\ldots,p_n$:	components of θ_{k-1}
$p_{11},\ldots,p_{1n}$:	components of x_k
$p_{21},\ldots,p_{2n}$:	products of components of θ_{k-1} and x_k
p_a :	$\hat{y}_k = \theta_{k-1}^T x_k$
p_b :	$s_k^2 = (y_k - \hat{y}_k)^2$
p_c :	$\delta^2 - \sigma_{k-1}^2$
p_d :	Information evaluation complete

Transitions	Description
t_0 :	shift
$t_1,\ldots,t_n$:	concurrent scalar multiplications
$t_{11}\cdots t_{1n}$:	shift
$t_{20}\cdots t_{2k}$:	concurrent additions
t_a :	addition
t_b :	scalar addition and multiplication
t_c :	scalar addition and multiplication
t_d :	compare

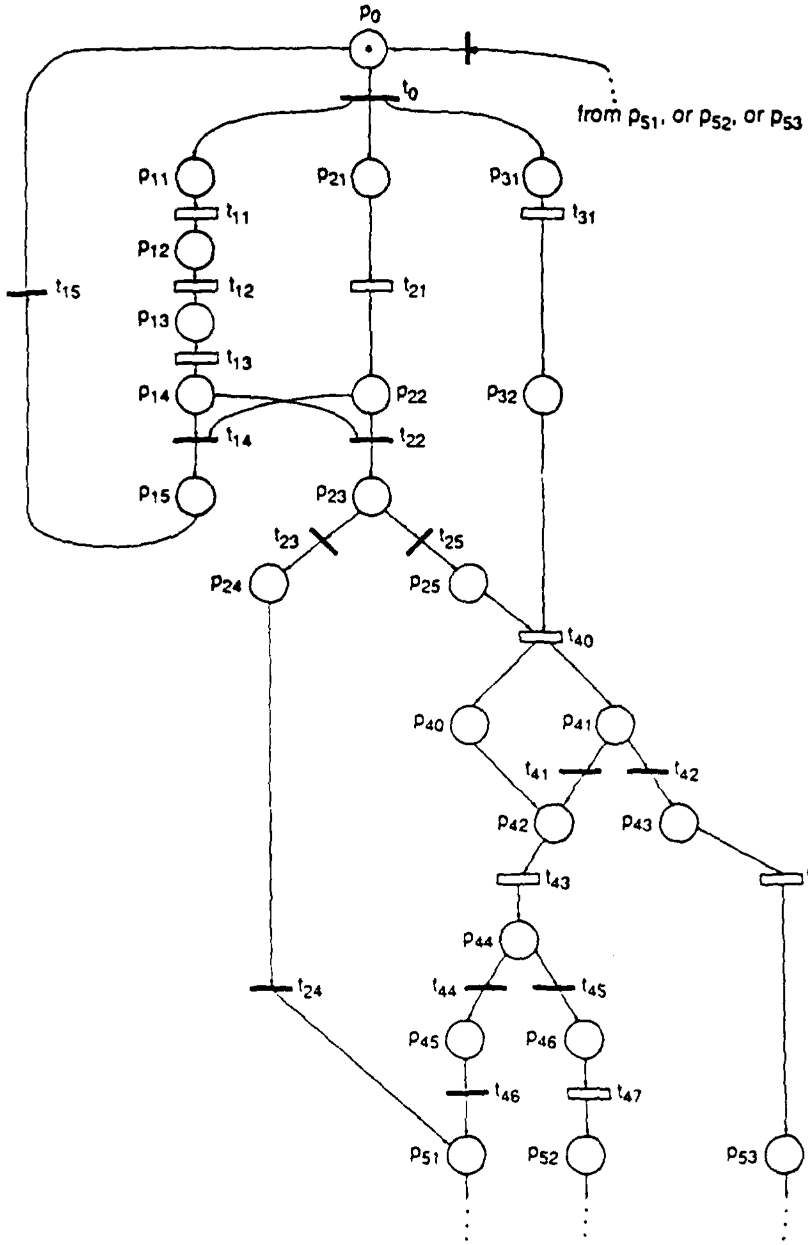
Figure 2: A Petri net model for information evaluation.



Places	Description
p_0	input data ready
$p_1,\cdots,p_n$	i^{th} column of P_{k-1}, $i = 1,2,\cdots,n$.
$p_{11},\cdots,p_{1n}$	components of x_k
$p_{21},\cdots,p_{2n}$	products of components x_k and i^{th} column of P_k
p_a	computation of $P_{k_1} x_k$ complete

Transitions	Descriptions
t_0	shift
$t_1,\cdots,t_n$	concurrent scalar-vector multiplications
$t_{11},\cdots,t_{1n}$	shift
$t_{20},\cdots,t_{2k}$	concurrent additions
t_a	addition

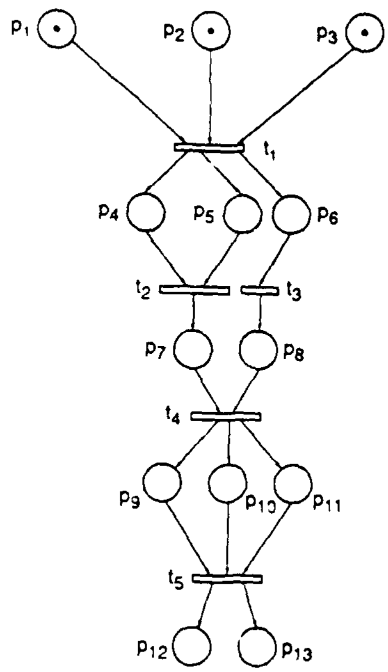
Figure 3: A building-block model for matrix-vector multiplications.



Places	Descriptions
p_0	ready to fetch sample again
p_{11}, p_{21}, p_{31}	input data ready $(y_k, x_k, \theta_{k-1}, P_{k-1}, \sigma^2_{k-1}, \gamma^2)$
p_{12}	$y_k = \theta_{k-1} x_k$
p_{13}	$\delta_k = y_k - \hat{y}_k$
p_{14}	δ_k^2
p_{15}	$\delta_k^2 \leq \gamma^2 - \sigma^2_{k-1}$
p_{22}	$\gamma^2 - \sigma^2_{k-1}$
p_{23}	$\delta_k^2 > \gamma^2 - \sigma^2_{k-1}$
p_{32}	$P_{k-1} x_k$
p_{24}	$\delta_k^2 = 0$
p_{25}	$\delta_k^2 \neq 0$
p_{40}	β_k
p_{41}	$G_k = x_k P_{k-1} x_k$
p_{42}	$G_k \neq 1$
p_{43}	$G_k = 1$
p_{44}	$\beta_k(G_k - 1)$
p_{45}	$\beta_k(G_k - 1) \leq -1$
p_{46}	$\beta_k(G_k - 1) > -1$
p_{51}	$\lambda_k = \alpha$
p_{52}	$\lambda_k = \frac{1}{1-\beta_k}\left[1 - \sqrt{\frac{G_k}{1+\beta_k(G_k-1)}}\right]$
p_{53}	$\lambda_k = \frac{1-\beta_k}{2}$

Transitions	Descriptions
t_{11}, t_{31}	concurrent vector multiplications
t_{12}, t_{21}	concurrent additions
t_{13}	mulltiplication
$t_{14}, t_{22}, t_{23}, t_{24}, t_{25}$	compare
t_{40}	concurrent vector
t_{41}, t_{42}	compare
t_{43}, t_{48}	multiplications and additions
t_{44}, t_{45}	compare
t_{46}	immediate transition
t_{47}	multiplication, division, square root, etc.

Figure 4(a): A Petri net model for MRE at the lower level (cont'd to 4(b)).



Places	Descriptions
p_1, p_2, p_3	data ready $(\lambda_k, G_k, p_{k-1}x_k)$
p_4	$1 - \lambda_k$
p_5	$\lambda_k G_k$
p_6	$P_{k-1}x_k x_k^T P_{k-1}$
p_7	$D_k \triangleq 1 - \lambda_k + \lambda_k G_k$
p_8	$\lambda_k P_{k-1} x_k x_k^T P_{k-1}$
p_9	P_k
p_{10}	$\lambda_k \delta_k$
p_{11}	$\lambda_k (1 - \lambda_k)\delta_k / D_k$
p_{12}	θ_k
p_{13}	σ_k^2

Transitions	Descriptions
t_1	multiplication (scalar to matrix)
t_2	addition
t_3	multiplication
t_4	concurrently compute $\frac{1}{1-\lambda_k}$ and $\lambda_k \frac{P_{k-1}x_k x_k^T P_{k-1}}{D_k}$
t_5	vector multiplication

Figure 4(b): A Petri net model for MRE at the lower level (cont'd from 4(a)).

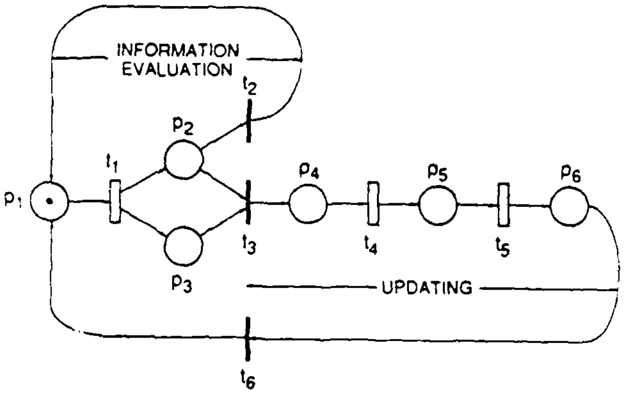


Figure 5: A Petri net model for MRE at the higher level.

Places	Descriptions
p_1 :	new input data ready (register memory available)
p_2 :	information evaluation completed
p_3 :	$P_{k-1}x_k$ computation finished
p_4 :	update process starts here
p_5 :	λ_k available
p_6 :	new set of parameter available

Transitions	Descriptions
t_1 :	information evaluation concurrent with computation of $P_{k-1}x_k$
t_2 :	no update, sample new data
t_3 :	update, start updating processor
t_4 :	computing λ_k
t_5 :	updating parameters
t_6 :	sample new data

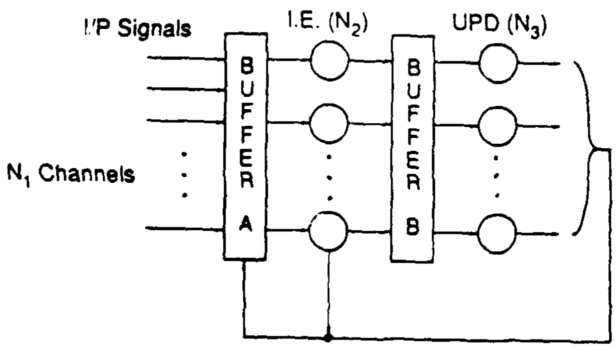


Figure 6: A pipelined concurrent adaptive signal processing network.

may be different priorities, in terms of the desperateness for parameter updates, for different channels. Tracking of variations of parameters and system characteristics should also be of much concern, especially in the content of overall system stability. These issues will be further complicated if sampling rates for different channels are different.

The virtues of this type of signal processing architecture are improvements of data throughput rates, cost reduction of signal processing hardware, and improvements of reliability. If some of the IE's or UPD's fail, the network will still be able to function properly. It would also benefit from efficient use of information contained in the received data, a potential for interplay of adaptive signal processing and artificial intelligence.

Accomplishments of these design problems will certainly rely on effectiveness of the modeling tool available. State-of-the-art models, such as finite state machine (FSM), Petri nets (PN), and finitely recursive processes (FRS), will be investigated as to their suitability to this signal processing design problem.

V. CONCLUSIONS

The propagation of data computation and timing of a modular recursive estimation algorithm are examined using a discrete event model. The key to the MRE algorithm is a decision-making regarding the information content of the received data. As a result, it features modularity at the higher level, as opposed to the computational level. This feature enabled us to consider a parallel-pipelined architecture for adaptive signal processing network which makes more efficient use of signal processors and will facilitate achievement of real time signal processing.

References

[1] Y. F. Huang, "A recursive estimation algorithm using selective-updating for spectral analysis and adaptive signal processing," *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. ASSP-34, pp. 1331-1334, 1986.

[2] S. Dasgupta and Y.F. Huang, "Asymptotically convergent modified recursive least-squares with data-dependent updating and forgetting factor for systems with bounded noise," *IEEE Trans. Inform. Theory*, Vol. IT-33, No. 3, pp. 383-392, 1987.

[3] H.T. Kung, "Why systolic architectures?," *IEEE Computer*, Vol. 25, pp. 37-46, 1982.

[4] S.Y. Kung, H.J. Whitehouse, and T. Kailath, Eds., VLSI and Modern Signal Processing. Englewood-Cliffs, New Jersey: Prentice-Hall, 1985.

[5] S.K. Rao and T. Kailath, "VLSI arrays for digital signal processing: Part I - A Model identification approach to digital filter realizations," *IEEE Trans. Circuits and Syst.*, Vol. CAS-32, pp. 1105-1118, 1985.

[6] H.H. Lu, E.A. Lee, and D.G. Messerschmitt, "Fast recursive filtering with multiple slow processing elements," *IEEE Trans. Circuits and Systs.*, Vol. CAS-32, pp. 1119-1129, 1985.

[7] E.A. Lee and D.G. Messerschmitt, "Pipeline interleaved programmable DSP's: Architecture," *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. ASSP-35, No. 9, pp. 1320-1333, 1987.

[8] Y.F. Huang and M. Yeh, "Adaptive hybrid balancing based on a novel recursive estimation algorithm and spectral analysis," *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. ASSP-35, No. 3, pp. 397-399, 1987.

[9] C.V. Ramamoorthy and G.S. Ho, "Performance evaluation for asynchronous concurrent systems using Petri nets," *IEEE Trans. Software Engrg.*, Vol. SE-6, No. 5, pp. 440-449, 1980.

[10] M.K. Molloy, "Performance analysis using stochastic Petri nets," *IEEE Trans. Computer*, Vol. C-31, No. 9, pp. 913-917, 1982.

[11] G. Balbo, S.C. Bruell, and S. Ghanta, "Combining queueing network and generalized stochastic Petri net models for the analysis of some software blocking phenomena," *IEEE Trans. Software Engrg.*, Vol. SE-12, No. 4, pp. 561-576, 1986.

[12] C.A.R. Hoare, *Communicating Sequential Processes*. Prentice-Hall International, U.K., Ltd., 1985.

[13] Y.C. Ho, "Performance evaluation and perturbation analysis of discrete event dynamic systems," *IEEE Trans. Automat. Contr.*, Vol. AC-32, No. 7, pp. 563-572, 1987.

[14] K. Inan and P. Varaiya, "Finitely recursive process models for discrete event systems," Technical Report, University of California, Berkeley, July, 1987.

[15] P.J. Ramadge and W.M. Wonham, "Supervisory control of a class of discrete event processes," *SIAM J. Contr. Optim.*, Vol. 25, No. 1, pp. 206-230, 1987.

ACKNOWLEDGEMENTS

This work has been supported in part by the Office of Naval Research under Contract N00014-87-k-0284, and in part by the National Science Foundation under Grant ECS-8505218.

Publication V.A.3.

The Proceedings of The 1988 International Conference on

ADVANCES IN COMMUNICATION & CONTROL SYSTEMS

Volume I

October 19-21, 1988
Baton Rouge, LA

Edited by
William A. Porter
and
Subhash C. Kak

Department of Electrical and Computer Engineering
Louisiana State University
Baton Rouge, LA 70803

RECURSIVE ARMA PARAMETER ESTIMATION WITH A DISCERNING UPDATE STRATEGY- FINITE PRECISION EFFECTS

Ashok K. Rao
University of Notre Dame

Yih-Fang Huang
University of Notre Dame

Introduction. The performance of adaptive filter algorithms in finite precision environments has received a lot of attention in the past few years. The problem is important because a practical implementation of these algorithms will impose constraints on the word-length, which may cause significant degradation in the performance. For example, some of the fast least-squares algorithms, though appealing in theory, have been found to be unstable in finite word-length implementations [1]. Round-off and quantization errors affect different adaptive algorithms in different ways. The accumulation of round-off errors in the recursive least-squares (RLS) algorithm can cause the inverse of the associated estimated covariance matrix to become indefinite and the algorithm to diverge fairly early, especially if the order of the filter is large[2,3]. This effect is pronounced if the data is ill conditioned, i.e., the data autocorrelation matrix has a large eigenvalue spread. On the other hand, it can take millions of iterations before the effect of quantization errors becomes noticeable in the widely used LMS algorithm[4].

In this paper we first study the effects of roundoff errors in a fixed point implementation of the so-called Optimal Bounding Ellipsoid (OBE) algorithm[5]. This algorithm estimates recursively the coefficients of autoregressive with exogenous inputs (ARX) processes. One of the main features of this algorithm is a discerning update strategy. This feature, obtained by the introduction of an *information dependent updating/forgetting factor*, yields a modular structure thereby increasing the potential for concurrent and pipelined processing of signals. The presence of such a forgetting factor also gives the algorithm the ability to track time varying parameters.

The OBE algorithm belongs to a broad family of algorithms known as membership set parameter estimation algorithms [6],[7],[8]. These algorithms are particularly useful when the statistical properties of the noise sequence $\{v(t)\}$ are unknown, but instantaneous bounds on its magnitude are available. In the past few years, there has been a resurgence of interest in these algorithms. However, the key issue of finite precision effects has not received much attention. We have found that in small word-length situations, the performance of the OBE algorithm is superior to that of the RLS algorithm (with and without forgetting factor). The EOBE algorithm, which is an extension of the OBE algorithm to ARMA models[9] is studied next and simulation results also indicate that the EOBE algorithm has better numerical properties than the extended least-squares (ELS) algorithm(see[3]for details of the RLS and ELS algorithms).

This paper is organised as follows: The first section introduces the concept of membership-set parameter estimation and describes in fuller detail the OBE algorithm. The next section presents the extension of the OBE algorithm for parameter estimation of ARMAX processes.The simulation procedure and the simulation results are presented in the following two sections.The paper concludes with discussions of the simulation results.

The OBE Algorithm. Membership-set parameter estimation is concerned with the determination of sets of parameters which are consistent with the measurements, model structure, and noise constraints. The model and noise representations commonly used are

$$y(t) = \theta^{*T}\Phi(t) + v(t), \quad |v(t)| \leq r^{1/2}(t) \qquad (1)$$

where $\{y\}$ is a sequence of scalar observations, θ^* is the parameter vector to be identified, $\Phi(t)$ is a n-vector of variables known at time t, $\{v\}$ is the noise sequence and $\pm r^{1/2}(t)$ are the time varying noise bounds. Given a sequence $\{y(i),\Phi(i)\}$, i=1..k, the optimal membership set

$$\psi^0_k = \cap^k_{i=1} S_i$$

where

$$S_i = \{\theta : (y(i) - \theta^T\Phi(i))^2 \leq r(i),\ \theta \in R^n\}$$

From a geometrical viewpoint, S_i is a convex polytope in R^n and contains the true parameter vector. Finding ψ^0_k is often computationally intractable and it is therefore necessary to approximate ψ^0_k by some set which approximates it closely and which can be described and updated economically. The different membership-set algorithms differ in the way the optimal membership set is approximated and in the method used to obtain an optimum(in some sense) set.

The OBE algorithm estimates the coefficients of ARX processes described by

$$y(t) = a_1\, y(t-1)+..+ a_n\, y(t-n) + b_0\, u(t) + b_1\, u(t-1) +..+ b_m\, u(t-m) + v(t)$$

where y(t) is the output, u(t) is the input and v(t) is the noise contaminating the observations.
The above equation can be recast as :

$$y(t) = \theta^{*T}\Phi(t) + v(t)$$

where

$$\theta^* = [a_1, a_2 \ldots a_n, b_0, b_1, \ldots, b_m]^T$$

is the vector of true parameters, and

$$\Phi(t) = [\, y(t-1), y(t-2), .. y(t-n),\ u(t), u(t-1), .. u(t-m)\,]^T$$

is the regressor vector. It is assumed that the noise is uniformly bounded in magnitude, i.e., there exists $\gamma_0 \geq 0$, such that

$$v^2(t) \leq \gamma_0^2 \qquad \text{for all t, hence}$$

$$(y(t) - \theta^{*T}\Phi(t))^2 < \gamma_0^2$$

Let S_t be a subset of the euclidean space R^{n+m+1}, defined by

$$S_t = \{\theta : (y(t) - \theta^T\Phi(t))^2 \leq \gamma_0^2,\ \theta \in R^{n+m+1}\}$$

The OBE algorithm starts off with a large ellipsoid, E_0, in R^{n+m+1} which contains all possible values of the modelled parameter θ^*. After the first observation $y(1)$ is acquired, an ellipsoid is found which bounds the intersection of E_0 and the convex polytope S_1. To hasten convergence, this ellipsoid must be optimized in some sense, say minimum volume[7] or by any other criterion[5,10]. Denoting the optimal ellipsoid by E_1 one can proceed exactly as before with the future observations and obtain a sequence of optimal bounding ellipsoids(OBE) $\{E_t\}$.

The center of the ellipsoid E_t can be taken as the parameter estimate at the t-th instant and is denoted by $\theta(t)$. If at a particular time instant i, the resulting optimal bounding ellipsoid would be of a " smaller size ", thereby implying that the data point y(i) contains some " information " regarding the parameter estimates, then the parameter estimates are updated. Otherwise E_i is set equal to E_{i-1}, and the estimates are not updated. In essence, the recursive estimator consists of two modules, an information evaluator followed by an updating processor. At each data point, the received data proceed to the updating processor only if the information evaluator indicates that some fresh information is contained in the data.

Specifically, let the ellipsoid E_{t-1} at the (t-1)-th instant be formulated by

$$E_{t-1} = \{ \theta : (\theta - \theta(t-1))^T P^{-1}(t-1)\,(\theta - \theta(t-1)) \le \sigma^2(t-1) \}$$

for some positive definite matrix P (t-1) and a non-negative scalar $\sigma^2(t-1)$. Then, given y(t), an ellipsoid which bounds $E_{t-1} \cap S_t$ "tightly" is

$$\{ \theta : (1-\lambda_t)(\theta - \theta(t-1))^T P^{-1}(t-1)(\theta - \theta(t-1)) + \lambda_t\,(y(t) - \theta^T\Phi(t))^2 \le (1-\lambda_t)\,\sigma^2(t-1) + \lambda_t \gamma_0^2 \}$$

where the forgetting factor λ_t satisfies $0 \le \lambda_t \le \alpha < 1$, with α being a user chosen upper bound on the forgetting factor. The size of the bounding ellipsoid is related to the scalar $\sigma^2(t-1)$ and the eigenvalues of P(t-1). The update equations for $\theta(t)$, P(t) and $\sigma^2(t)$, derived in [5], are as follows

$$\theta(t) = \theta(t-1) + K(t)\delta(t) \tag{2a}$$

$$\delta(t) = y(t) - \theta^T(t-1)\,\Phi(t) \tag{2b}$$

$$K(t) = \frac{\lambda_t P(t-1)\Phi(t)}{1-\lambda_t + \lambda_t G(t)} \tag{2c}$$

$$G(t) = \Phi^T(t)\,P(t-1)\,\Phi(t) \tag{2d}$$

$$P(t) = \frac{1}{1-\lambda_t}\left[\, I - K(t)\Phi^T(t) \,\right] P(t-1) \tag{2e}$$

where $\Phi(t)$ is the regressor vector which contains present and previous input and output samples.

The optimal ellipsoid which bounds the intersection of E_{t-1} and S_t is defined in terms of an optimal value of λ_t. For the OBE algorithm of [5], the optimum value λ^*_t is determined by minimization of $\sigma^2(t)$ with respect to λ_t at every time instant. The minimization procedure results in a discerning update procedure. In particular, λ^*_t is set equal to zero (no update) if

$$\sigma^2(t) + \delta^2(t) \leq \gamma_0^2(t) \tag{3}$$

On the other hand, if (3) is not satisfied, then the optimal value of λ_t is computed as follows:

$$\lambda^*_t = \min(\alpha, v_t) \tag{4}$$

where

$$v_t = \begin{cases} \alpha & \text{if } \delta^2(t) = 0 \\ \dfrac{1-\beta(t)}{2} & \text{if } G(t) = 1 \\ \dfrac{1}{1-G(t)}\left[1 - \sqrt{\dfrac{G(t)}{1+\beta(t)(G(t)-1)}}\right] & \text{if } \beta(t)(G(t)-1)+1 > 0 \\ \alpha & \text{if } \beta(t)(G(t)-1)+1 \leq 0 \end{cases}$$

and

$$\beta(t) \triangleq \frac{\gamma^2 - \sigma^2(t-1)}{\delta^2(t)}$$

The above recursions(2), and the selective update criterion (3,4), along with the initial values

$$P^{-1}(0) = I, \quad \theta(0) = 0 \text{ and } \sigma^2(0) = 1/\varepsilon \text{ with } \varepsilon \ll 1$$

form the basis of the OBE estimation algorithm. Note that the OBE algorithm is similar in form to the weighted recursive least-squares (WRLS) algorithm, with the information dependent updating factor acting as a weighting factor on the observations. Note also that the complexity of the information evaluation procedure (3) is much less than that of the updating procedure (2).

The Extended OBE algorithm. An ARMA (n,r) process is of the form

$$y(t) = a_1 y(t-1) + .. + a_n y(t-n) + w(t) + c_1 w(t-1) + .. + c_r w(t-r) \tag{5}$$

where y(t) is the output and w(t) is the input noise which is assumed to be uncorrelated and unknown. If w(t) is assumed to be bounded in magnitude by γ_0, then the OBE algorithm can be extended to this ARMA parameter estimation problem, if estimates of w(t-1), w(t-2)..,w(t-r) are available[9]. The algorithm is essentially the same except for the following changes:

(i) The regressor vector is now given by

$$\Phi(t) = [\, y(t-1), \;..\; y(t-n), \varepsilon(t-1), \;\ldots\; \varepsilon(t-r) \,]^T$$

where

$$\varepsilon(t) = y(t) - \theta^T(t)\Phi(t)$$

is the *a posteriori* prediction error and the parameter estimate $\theta(t)$ is the estimate of the a_i, $i = 1,2,\ldots,n$ and c_j, $j = 1,2,\ldots,r$.

(ii) The γ_0 in (3), which is the upper bound on the noise, is replaced by γ, an upper bound on the magnitude of the output y(t).

It is easily shown that, for the EOBE algorithm, minimizing $\sigma^2(t)$ with respect to λ_t at every time instant yields the same updating criterion (3) and the same algorithm for determining the optimum value of the forgetting / updating factor λ^*_t, as in [5]. The algorithm thus retains the discerning update strategy and the modular adaptive filter structure.

Simulation Setup. A fixed point implementation of the OBE algorithm was simulated by performing the operations in integer arithmetic. The input and output observations, which are generated as floating point numbers, are converted to integers by the formula

$$x_{quant} = \begin{cases} \mathrm{INT}(\, x.\, 2^{ibit} + 0.5)\,,\ x > 0 \\ \mathrm{INT}(\, x.\, 2^{ibit} - 0.5)\,,\ x \le 0\,. \end{cases}$$

where *ibit* is the number of bits assigned for the integer representation of the fractional part of the real number x. In the simulations, since an integer is stored in 32 bits, all registers and word sizes are 32 bits. Multiplication is performed by forming the product in a 48-bit word, scaling down by 2^{-ibit}, and then rounding off to the nearest integer. Inner products are formed similarly by accumulating the products in a 48-bit word, scaling down and then rounding off.

The upper bound α on the forgetting factor, has to be chosen with care in the fixed point implementation of the OBE and EOBE algorithms. If α is chosen greater than 0.1, then the elements of the matrix P often increase rapidly in magnitude and overflows can occur. The reason for this is that in the initial stages, the optimum value of the forgetting factor λ equals α fairly often. Consequently, since $1-\lambda$ appears in the denominator of (2e), the magnitude of the elements of P can increase and cause overflows. On the other hand, if α is chosen too small then the algorithm takes more iterations to converge and the number of updates increases. A value of α =0.1 was found to yield a satisfactory convergence rate and inhibit overflows in the update equation for P(t).

In addition to α, the initial value $\sigma^2(0)$ has to be chosen small enough to prevent overflows in the subsequent calculations of λ^*. This is because if, at any time t, $\sigma^2(t-1)$ is large and $\delta^2(t)$ is small then $\beta = (\gamma^2-\sigma^2(t-1))/\,\delta^2(t)$ can become a very large negative number and the product $\beta(G-1)$ can overflow. However, if overflows can be detected and a saturation value is used for β, then the calculation of λ^* will not be affected. Since β is negative and large in magnitude, $1+\beta(G-1)$ is a large positive or negative number, depending on whether G is greater than or less than unity. In case $1+\beta(G-1)$ is positive, then it can be seen from (4) that v_t is greater than unity, and consequently $\lambda^* = \alpha$. On the other hand if $1+\beta(G-1)$ is negative then $\lambda^* = \alpha$ from (4). Thus large values of $\sigma^2(0)$ can be used if care is taken to account for overflows in the algorithm for calculating λ^*. In our simulations, the initial (unquantized) value taken is $\sigma^2(0) = 100$.

For the RLS algorithm, the initial value P(0) is also important. Since the bias in the estimates is inversely proportional to P(0), P(0) should be large. However large values can cause the Kalman gain vector K to overflow, and the parameter estimates to grow exponentially in the initial stage [11]. Therefore a compromise value P(0) = 10 I, where I is the identity matrix, was chosen.

Simulation Results. To compare the performance of the OBE algorithm *vis a vis* the RLS and EWLS algorithms, simulations were performed with an AR(4) and an ARX(4,4) process

Example 1. AR(4) process

$$y(t) = -0.6\, y(t-1) - 1.58\, y(t-2) - 0.464\, y(t-3) - 0.5576\, y(t-4) + v(t)$$

The noise sequence $\{v(t)\}$ is generated by a pseudo-random number generator with a uniform probability distribution in $[-1.0, 1.0]$. The upper bound γ^2 was set equal to 1.0. The parameter estimates were obtained by applying the OBE and the RLS algorithms to 500 point data sequences. Twenty five runs of the algorithm were performed on the same model but with different noise sequences. The number of bits used for the fractional part, *ibit*, was varied from 16 down to 8 bits. The average squared parameter error L is computed for each value of ibit according to the formula

$$L = \frac{1}{25} \sum_{j=1}^{25} (\theta_j - \theta^*)^T (\theta_j - \theta^*)$$

where θ_j is the final parameter estimate in the j-th run and θ^* is the true parameter. The average tap error for the OBE, RLS and exponentially weighted RLS(EWLS)with forgetting factor λ =0.99, is plotted against *ibit* in Fig.1. It can be seen that the performance of the OBE algorithm appears to be constant as the number of bits varies from 16 to 9. In contrast, the performance of the RLS algorithm degrades substantially as the word-length decreases. The performance of the EWLS algorithm is even worse. The RLS and EWLS algorithms overflowed for ibit $\leq$ 8. The OBE algorithm overflowed for ibit $\leq$ 7.

Example 2. ARX(4,4) process

$$y(t) = 0.5y(t-1) - 0.4y(t-2) + 0.6y(t-3) + 0.2y(t-4) + u(t) - 0.29u(t-1) + 0.5u(t-2) - 0.7u(t-3) + v(t)$$

The input and noise sequences are generated by a pseudo-random number generator as before. The average tap error L, for the OBE, RLS and EWLS algorithms is plotted against *ibit* in Fig. 2. As before, the average tap error of the OBE algorithm appears constant as *ibit* varies from 16 to 7 bits. The RLS and EWLS algorithms do not work well for ibit $\leq$ 8.

Simulations were also performed for an ARX(10,10) model. However the large order seems to have caused greater accumulation of round-off errors in both the RLS and OBE algorithms and consequently overflows occurred.

Example 3.

The performance of the EOBE algorithm was evaluated by simulating an ARMA(3,3) process

$$y(t) = -0.4\, y(t-1) + 0.2\, y(t-2) + 0.6\, y(t-3) + w(t) - 0.6\, w(t-1) + 0.2\, w(t-2) + 0.6\, w(t-3)$$

The noise sequence $\{w(t)\}$ is generated by a pseudo-random number generator with a uniform probability distribution in $[-1.0, 1.0]$. The upper bound γ^2 was set equal to 25.0. The average parameter error L is plotted for the EOBE and ELS(with forgetting factor $\lambda = 1$ and λ =0.99) algorithms in Fig. 3. As in the previous case, it can be seen that while the performance of the EOBE algorithm is fairly constant over a range of word-lengths, the ELS algorithm does not perform properly for *ibit* < 9. The performance of the ELS algorithm with a forgetting factor of 0.99 was worse. The algorithm overflowed for *ibit* < 13.

Discussions. The superior performance of the OBE (EOBE) algorithms, as compared to the RLS (ELS) algorithms, is quite encouraging. One of the reasons could be the selective update strategy of the OBE algorithm. Such an update strategy may be responsible for a slower accumulation of roundoff errors on account of the updates being performed infrequently. Hence, if the RLS(ELS) and OBE(EOBE) algorithms operate on large sets of data, then the OBE(EOBE)algorithm could be less prone to divergence, simply because it does not update as often.

The difference in performance could also result from the differences in the update equation for P(t). The update equation for the RLS (ELS)algorithm with a forgetting factor λ is

$$P(t) = \left[I - \frac{P(t-1)\Phi(t)\Phi^T(t)}{\lambda + \Phi^T(t)P(t-1)\Phi(t)} \right] \frac{P(t-1)}{\lambda} \tag{6}$$

The corresponding equation for the OBE(EOBE) algorithm is (2e), which can be rewritten as

$$P(t) = \left[I - \frac{P(t-1)\Phi(t)\Phi^T(t)}{\frac{1-\lambda_t}{\lambda_t} + \Phi^T(t)P(t-1)\Phi(t)} \right] \frac{P(t-1)}{1-\lambda_t} \tag{7}$$

Since $1-\lambda_t$ plays the same role in the OBE algorithm as does λ in the RLS algorithm, the only difference between (6) and (7) is that the factor $(1-\lambda_t)/\lambda_t$ appears in the denominator of the term within braces in (7) as opposed to the corresponding term λ in (6). The degradation of performance occurs primarily because the term within braces becomes indefinite(has positive and negative eigenvalues) on account of round-off errors. Since λ_t is usually much smaller than unity, the term which is being subtracted from the identity matrix in (7) is much smaller than the one in (6). Thus P(t) in the RLS(ELS) algorithm has a greater tendency to become indefinite than the P(t) in the OBE(EOBE) algorithm. This observation has been confirmed by examining the eigenvalues of P(t), for runs in which the RLS algorithm performed poorly.

The failure of the RLS algorithm when the order is large(>10) is well known and there exist several methods like the UDU' [2,3] and QR factorization [4] methods, to make the P update numerically stable. For the OBE algorithm, a recenlty proposed systolic array implementation[12] may have better numerical properties. The derivation of other numerically stable recursions for the OBE algorithm is currently under investigation.

Conclusions. The finite-precision performance of the OBE and the EOBE algorithms has been studied through simulations. The performance of these algorithms in small word-length environments is superior to that of the well known RLS and ELS algorithms . The improvement is attributed to differences in the resursion of the matrix P(t) and less number of updates of the OBE algorithm. For large order processes, both the RLS and the OBE algorithm did not work properly when the word-length was small and hence more numerically robust algorithms may be required for such situations.

Acknowledgement. This work has been supported in part by the National Science Foundation under Grant MIP-87-1174 and in part by the office of Naval Research under Contract N00014-87-k-0284. Inquiries may be directed to Dr. Yih-Fang Huang, Department of Electrical & Computer Engineering, University of Notre Dame, Notre Dame, IN 46556.

REFERENCES

[1] J. M. Cioffi and T. Kailath, " Fast RLS transversal filters for adaptive filtering," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-32, No. 2, pp. 304-337, April 1984.

[2] G. J. Bierman, *Factorization Methods for Discrete Estimation*, Academic Press, New York 1977.

[3] L. Ljung and T. Soderstrom,*Theory and Practice of Recursive Identification*, Cambridge, Mass: MIT Press, 1983.

[4] J. M. Cioffi, "Limited-precision effects in adaptive filtering," *IEEE Trans. Circuits and Systems*, vol. CAS-34, No. 7, pp. 821-833, July 1987.

[5] S. Dasgupta and Y.F. Huang, " Asymptotically convergent modified recursive least-squares," *IEEE Trans. Information Theory*, vol. IT-33, No.3, pp. 383-392, May 1987.

[6] F. C. Schweppe, " Recursive state estimation : unknown but bounded errors and system inputs" *IEEE Trans. Automatic Control*, vol. AC-13, No.1, pp. 22-28, Feb. 1968.

[7] E. Fogel and Y.F. Huang, " On the value of information in system identification - bounded noise case," *Automatica*, vol. 18, No. 2, pp. 229-238, March 1982.

[8] M. Milanese and G. Belaforte, " Estimation theory and uncertainty intervals," *IEEE Trans. Automatic Control*, vol. AC-27, No. 2, pp. 408-414, April 1982.

[9] A. K. Rao, Y. F. Huang and S. Dasgupta, " An extended OBE algorithm for ARMA parameter estimation, " Proc. 24th Annual Allerton Conf., Urbana-Champaign, Sept. 28-30, 1988.

[10] Y. F. Huang, " A recursive estimation algorithm using selective updating for spectral analysis and adaptive signal processing,"*IEEE Trans. Acoust.Speech and Signal Process.*, vol. ASSP-34, No. 5, pp. 1131-34, October 1986.

[11] S. H. Ardalan and S. T. Alexander," Fixed-point roundoff error analysis of the exponentially windowed RLS algorithm for time-varying systems", *IEEE Trans. Acoust.Speech and Signal Process.*, vol. ASSP-35, No. 6, pp. 1131-34, June 1987.

[12] J. R. Deller, " 'Systolic array ' implementations of the optimal bounding ellipsoid algorithm," *IEEE Trans. Acoust.Speech and Signal Process.*, to be published.

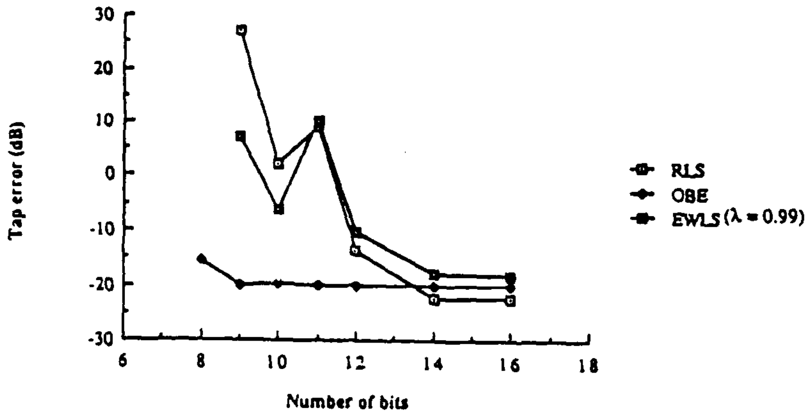


Fig. 1 Average tap error of the OBE and RLS algorithms for an AR(4) process.

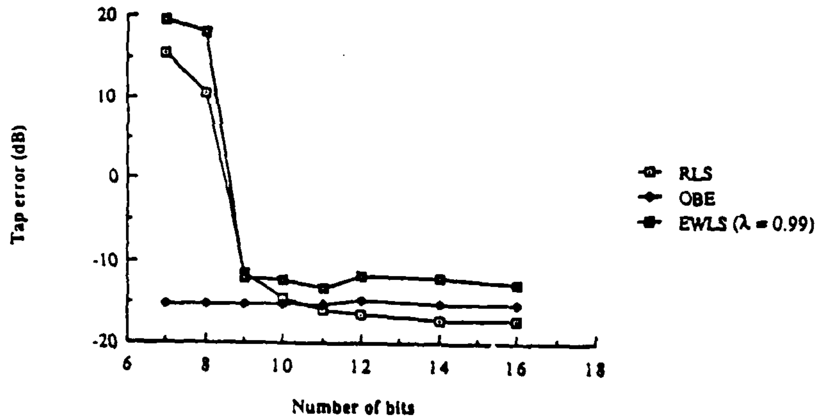


Fig. 2 Average tap error of the OBE and RLS algorithms for an ARX(4,4) process.

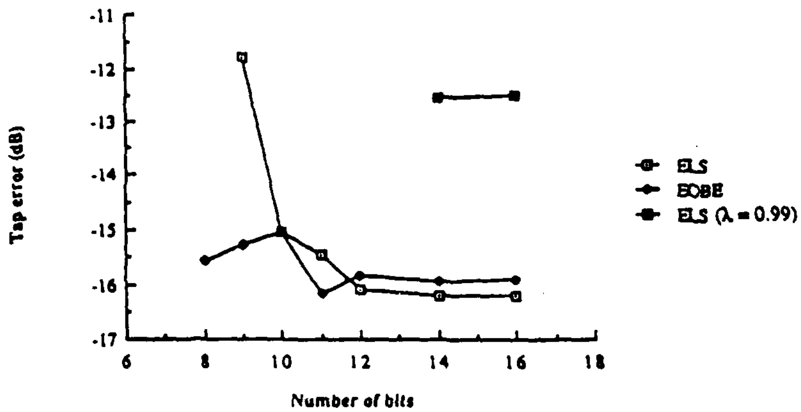


Fig. 3 Average tap error of the EOBE and ELS algorithms for an ARMA(3,3) process.

Publication V.A.4.

PROCEEDINGS OF THE

27th IEEE CONFERENCE ON DECISION AND CONTROL

DECEMBER 7-9, 1988
HYATT REGENCY AUSTIN ON TOWN LAKE
AUSTIN, TEXAS

IEEE
Control
Systems
Society

VOLUME 1 OF 3

88CH2531-2

on Decision and Control
Austin, Texas • December 1988

APPLICATIONS AND ANALYSIS OF SECOND ORDER ARTIFICIAL NEURAL NETWORKS

V.C. Soon and Y.F. Huang
Department of Electrical and Computer Engineering
University of Notre Dame, IN 46556

Abstract

This paper investigates a two layer artificial neural network which consists of one hidden layer of second order neurons and an output layer of first order neurons. The second order neurons yield conic-surface type of decision regions such as ellipses and parabolas. A simulation example is presented. It is shown that such networks yield more flexibility than first order ones. Analysis on the memory capacities of single-layer and two-layer networks is presented.

INTRODUCTION

A feedforward neural network could have one or several hidden layers of neurons and a single layer of output neurons. Information flows upward from the lowest hidden layer, which receives the inputs, to the output layer of neurons from which the outputs of the neural network are retrieved. Typically, the output of each neuron in the network is a semi-linear function of the inputs formulated by

$$O_j = f\left(\sum_i \omega_{ji} O_i' + \theta_j\right) \qquad (1)$$

where O_j denotes the output of the j-th neuron, ω_{ji} is the weight for the connection from the i-th to the j-th neuron, and O_i' is the i-th input to the j-th neuron. The function f(.) in (1) is typified by the sigmoid function

$$f(x) = \frac{1}{(1+e^{-x})} \qquad (2)$$

The sum $\Sigma_i \omega_{ji} O_i + \theta_j$ is often referred to as the discriminant function.

The concept of first and second order discriminant functions arise quite naturally in classification problems of statistical pattern recognition [2]. Assume that it is desired to classify a d-dimensional input pattern $\mathbf{x}$ as either of class A_0 or of class A_1, given that the conditional probability density functions $p(x|A_0)$, $p(x|A_1)$ are Gaussian with mean vectors μ_0 and μ_1 and covariance matrices R_0 and R_1, respectively.

This problem can be formulated as one of a simple hypothesis testing. Using the Bayes or the Neyman-Pearson test, we obtain the following second order discriminant function,

$$T(\mathbf{x}) = (\mu_1^T R_1^{-1} - \mu_0^T R_0^{-1})\mathbf{x} + \frac{1}{2}\mathbf{x}^T (R_0^{-1} - R_1^{-1})\mathbf{x} \underset{A_0}{\overset{A_1}{\gtrless}} \tau \qquad (3)$$

where τ is some appropriate threshold. If the distributions of the classes A_0 and A_1 have the same covariance matrices, (i.e., $R_0=R_1$), then the discriminant function in (3) defines a hyperplane in d-dimensions. When the covariances are not equal, (i.e., $R_0 \neq R_1$), then the discriminant function is of second order.

SECOND ORDER NETWORKS

High order generalizations of (1) can be obtained by considering high order discriminant functions as follows, [3]

$$x = \sum_i \omega_{ji} O_i' + \sum_{i,k} \omega_{jik} O_i' O_k' + .. + \theta_j \qquad (4)$$

where x is the argument in the sigmoid function, (2). We can see that there are more degrees of freedom in a higher order neuron and this will increase the capabilities of the neuron, at the expense of increasing the number of weights needed per neuron.

The second order discriminant function for a second order neuron is obtained from (4) by excluding all terms higher than second order, hence

$$x = \sum_i \omega_{ji} O_i' + \sum_{i,k} \omega_{jik} O_i' O_k' + \theta_j \qquad (5)$$

Assuming that there are d inputs to the neuron, then there are $(d+1)(d+2)/2$ weights per neuron. This is considerably more than the conventional (first order) neuron. However, second order discriminant functions can define conic-surface types of decision regions such as ellipses, hyperbolas and hyperplanes, instead of just the hyperplane decision regions available with the first order neuron. An advantage here is that, with such conic-surface type of decision regions, one can obtain smoother regions with less number of hidden neurons.

Simulation Example

The characters 'A', 'C', 'E', and 'O' are represented by a 3-by-3 matrix of one bit pixels and each character is presented to a single-layer feedforward network as a 9-by-1 vector (or pattern). The single layer network is composed of two output neurons. The network is then trained to recognize a character by giving a two-bit decision using the two neurons. Figure 1 shows the estimated root mean square error of the first output neuron versus the number of iterations (number of presentations of the character-set). Figure 2 shows the same plot for the second output neuron. Both figures demonstrate that second order neurons can be trained faster than first order neurons.

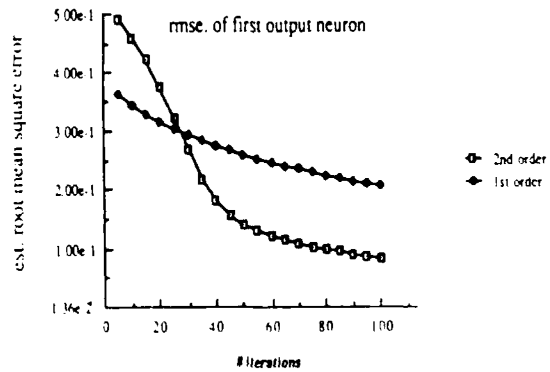


Figure 1: RMSE of first neuron vs. number of iterations.

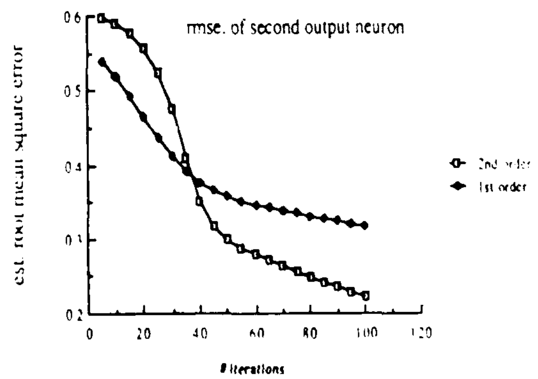


Figure 2: RMSE of second neuron vs. number of iterations.

To achieve a more general decision region, we could approximate a given decision region as a collection of ellipses and take the union of these elliptical decision regions. Each elliptical decision region can be formed with a single second order neuron and the final decision region is found by taking the union (a logical OR operation) of the elliptical decision regions of all the second order neurons.

The above can be achieved with a two-layer feedforward neural network composed of first order output neurons and a single layer of hidden second order neurons, as shown in Figure 3.

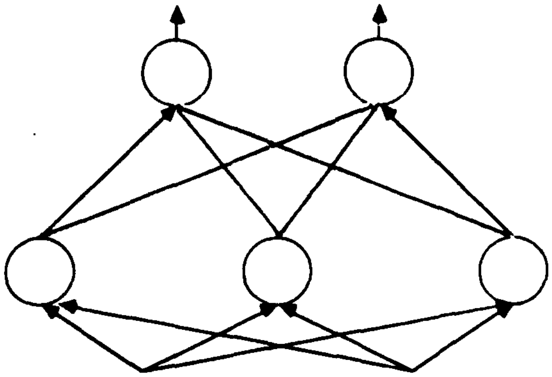


Figure 3: Proposed feedforward network

Simulation examples using the proposed network can be found in [4].

EXTENSIONS OF BACKPROPAGATION ALGORITHM

The backpropagation algorithm [5], is easily extended to the proposed two-layer network. Given the inputs and the desired outputs, $\{X_i^u\}, \{V_i^u\}$ respectively, the objective here is to minimize the following cost function

$$\varepsilon = \frac{1}{2}\sum_i (V_i^u - O_i)^2 \qquad (6)$$

where u denotes the u-th pattern of input/output vectors to be learned, and O_i is the actual output of the i-th output neuron. Using the negative gradient of ε as a descent direction we obtain

(i) For weights associated with the output neurons

$$\delta_j = O_j(1-O_j)(V_j^u - O_j) \qquad (7a)$$

$$\omega_{ji}(k+1) = \omega_{ji}(k) + \eta O_i' \delta_j \qquad (7b)$$

(ii) For weights associated with the hidden neurons

$$\delta_j = O_j(1-O_j)\sum_k \omega_{kj}\delta_k \qquad (8a)$$

Here, the index k refers to the output neurons.

$$\omega_{ji}(k+1) = \omega_{ji}(k) + \eta O_i' \delta_j \qquad (8b)$$

$$\omega_{jim}(k+1) = \omega_{jim}(k) + \eta O_i' O_m' \delta_j \qquad (8c)$$

MEMORY CAPACITY OF NETWORKS

We next examine the problem of how much information can be stored within a feedforward neural network.

To begin with, define *fundamental memories* as the input-patterns to be implemented and *associated memories* as the corresponding output-patterns.

Definition *The memory capacity of a given neural network is the maximum value of K, such that the neural network can map any K-set of fundamental memories to any K-set of associated memories*

We make the additional assumption that the K-set of fundamental memories are in *general position*, i.e., no (d+1) of the fundamental memories on the same (d-1)-dimensional hyperplane.

Then, we examine the memory capacity of a single-layer feedforward neural network with N output neurons and d inputs.

Theorem 1

The memory capacity of the single -layer feedforward neural network is K *such that*

K = d+1.

Remarks

(1) The proof follows from our definition of memory capacity and by looking at a single neuron in the network.

(2) In the case where the K-set of fundamental memories are not in general position, the capacity K is upper bounded by d+1, (i.e., K ≤ d+1).

We next look at the memory capacity of the Madaline network [6], which is a two-layer feedforward neural network composed of a single output neuron, m hidden neurons and d inputs.

Theorem 2

The memory capacity of the Madaline network is K *such that*

K = md.

Remarks

(1) The proof uses Lemma 1 and Theorem 1 in Baum [1]

(2) The above result agrees with the intuitive notion that increasing the number of hidden neurons increases the computational ability of the neural network.

The memory capacity of a two-layer feedforward neural network composed of N output neurons, m hidden neurons and d inputs is stated in the next theorem.

Theorem 3

The memory capacity of the two-layer feedforward neural network is K *such that*

md/N ≤ K ≤ md.

Remarks

(1) The proof uses Theorem 2.

(2) The second order discriminant function can be viewed as a mapping from d-space into (d+1)(d+2)/2 -space, which increases the dimensionality of the inputs. Hence, making the stronger assumption that the 'new' fundamental memories that are mapped from d -space are in general position, we apply the previous theorem to the proposed network and obtained,

$$m(d+1)(d+2)/2N \le K \le m(d+1)(d+2)/2$$

Thus, using second order discriminant functions increases the memory capacity of the two-layer feedforward neural network. This is achieved, however, at the cost of increasing the number of synaptic weights needed per neuron.

CONCLUSIONS

A two-layer feedforward neural network is presented which consists of first order output neurons and second order hidden neurons. Analysis on the memory capacities of the single-layer and two-layer feedforward neural networks was presented and applied to the proposed network.

Acknowledgement: This work has been supported in part by the Office of Naval Research under Contract N00014-87-k-0284, and in part by the National Science Foundation under Grant MIP-87-11174.

References

[1] E. B. Baum, "On the capabilities of multilayer perceptrons," Proc. IEEE Conf. Neural Info. Proc. Sys., Denver, 1987.

[2] S.T. Bow, Pattern recognition, Marcel Dekker Inc., New York, 1984.

[3] T. Maxwell, C. L. Giles, Y.C. Lee and H.H. Chen, "Non-linear dynamics of artificial neural systems," AIP Conf., Snowbird, 1986.

[4] V C. Soon and Y F. Huang, "Artificial neural networks with second order discriminant functions," Proc. Conf. Info. Sci. and Syst., Princeton University, 1988.

[5] D E. Rumelhart, G.E. Hinton and R.J. Williams, "Learning internal representations by error propagation," in PDP vol 1, eds. D.E. Rumelhart and J.L. McClelland, MIT Press, Cambridge, MA, 1986.

[6] B. Widrow, "Generalization and information storage in networks of Adaline neurons," in Self Organization Systems, eds. M.C. Yovits, G.J. Jacobi and G.D Goldstein, Spartan, Washington D.C., 1962.

Publication V.B.2.

ICASSP 88

V & E
VOLUME IV
VLSI
SPECTRAL ESTIMATION

1988
INTERNATIONAL CONFERENCE
ON
ACOUSTICS, SPEECH, AND
SIGNAL PROCESSING

APRIL 11-14, 1988
NEW YORK HILTON
NEW YORK CITY, USA

Sponsored by
THE INSTITUTE OF ELECTRICAL
AND ELECTRONICS ENGINEERS,
ACOUSTICS, SPEECH, AND
SIGNAL PROCESSING SOCIETY

E5.10

STATISTICAL PROPERTIES OF A NOVEL RECURSIVE ESTIMATION ALGORITHM WITH INFORMATION-DEPENDENT UPDATING

Ashok K. Rao and Y. F. Huang
Department of Electrical & Computer Engineering
University of Notre Dame
Notre Dame, IN 46556

ABSTRACT

Statistical analysis of a recursive parameter estimation algorithm is performed. The algorithm has been used for the estimation of parameters of ARX processes with bounded noise. Previous analyses of algorithms used in the bounded noise situation have been essentially deterministic. Unbiasedness of the estimates is shown under the assumption that the noise is white and zero mean. An upper bound on the covariance of parameter estimates is derived. An improved version of the algorithm is proposed which yields smoother estimates and greater resistance to outliers.

INTRODUCTION

The optimal bounding-ellipsoid (OBE) algorithm [1-3], is a recursive parameter estimation algorithm which incorporates information-dependent updating. It has been used for estimating the parameters of autoregressive processes with auxiliary inputs and bounded disturbances (ARX processes). The key feature of the OBE algorithm is that at every time step, a decision is made whether or not to update the parameter estimates. Simulation results have shown that the algorithm, on the average, updates only 20% of the time. This fact makes it eminently suitable for processing multiple channels simultaneously as it makes a time-sharing type of signal processor feasible. Another advantageous feature of the OBE algorithm is that after the algorithm converges the parameter updating stops. This is particularly useful in adaptive control applications where non-cessation of parameter updating may lead to instability. The OBE algorithm has been used for adaptive signal processing applications such as spectral estimation and adaptive hybrid balancing [4]. In this paper, some properties of the OBE estimator will be investigated. The analysis of bounded error algorithms [1,3,6] has been essentially deterministic. The statistical properties presented here provide a better understanding of the algorithm and also facilitate the performance analysis of the algorithm. In addition, a modification of the algorithm which improves its performance will be proposed.

THE OBE ALGORITHM

Consider the ARX model formulated by

$$y(t) = a_1 y(t-1) + a_2 y(t-2) + .. + a_n y(t-n) + b_0 u(t) + b_1 u(t-1) + .. + b_m u(t-m) + v(t) \quad (1)$$

where $y(t)$ is the measurable output, $u(t)$ is the measurable input and $v(t)$ represents the unknown but bounded disturbance at time instant t. The ARX equation can be recast as

$$y(t) = \Phi^T(t)\,\theta^* + v(t) \quad (2)$$

where

$\theta^*(t) = [a_1(t), a_2(t), \ldots, a_n(t), b_0(t), b_1(t), \ldots, b_m(t)]^T$

is the vector of system parameters and

$\Phi(t) = [\,y(t-1), y(t-2), \ldots y(t-n), u(t), u(t-1), \ldots u(t-m)]^T$

is the regressor vector. Given the input and output sequences and an upper bound γ on the magnitude of the noise, the OBE algorithm can be used to estimate the parameters $a_1, \ldots, a_n$, $b_0, \ldots, b_m$. The key equations defining the OBE algorithm of [3] are

$$P^{-1}(t) = (1-\lambda_t)\,P^{-1}(t-1) + \lambda_t \Phi(t)\Phi^T(t) \quad (3a)$$

$$\theta(t) = \theta(t-1) + \lambda_t P(t)\Phi(t)\,\delta(t) \quad (3b)$$

$$\delta(t) = y(t) - \theta^T(t-1)\,\Phi(t) \quad (3c)$$

$$\sigma^2(t) = (1-\lambda_t)\sigma^2(t-1) + \lambda_t\gamma^2 - \frac{\lambda_t(1-\lambda_t)\,\delta^2(t)}{1-\lambda_t+\lambda_t G(t)} \quad (3d)$$

$$P^{-1}(0) = I \ \text{ and } \ \theta(0) = 0,\ \sigma^2(0) = 1/\varepsilon,\ \varepsilon << 1 \quad (3e)$$

where

$\theta(t) = [\hat{a}_1(t), \hat{a}_2(t) \ldots \hat{a}_n(t), \hat{b}_0(t), \hat{b}_1(t), \ldots, \hat{b}_m(t)]^T$

is the vector of parameter estimates and

$G(t) = \Phi^T(t)\,P(t-1)\,\Phi(t)$

In this paper, the information dependent updating factor λ_t is determined by minimization of the non-negative quantity $\sigma^2(t)$ at every time instant. The optimal value of λ_t could be zero, thus implying that the data point $\{y(t), u(t)\}$ contains insufficient information, and consequently the estimates would not be updated. Further details and applications of the algorithm can be found in [3,7].

UNBIASEDNESS OF THE ESTIMATES

The following analysis assumes that the noise sequence $\{v(t)\}$ is white and is zero mean. From (3a) and (3e) it is easily shown that

$$P^{-1}(t) = \left[\prod_{i=1}^{t}(1-\lambda_i)\right] I + \sum_{i=1}^{t} q_i\,\Phi(i)\,\Phi^T(i) \quad (4)$$

where

$$q_i = \lambda_i \prod_{s=i+1}^{t}(1-\lambda_s) \quad (5)$$

CH2561-9/88/0000-2436 $1.00

Substituting (3a) and (3c) in (3b) yields

$$P^{-1}(t)\theta(t) = (1-\lambda_t)\, P^{-1}(t-1)\theta(t-1) + \lambda_t \Phi(t)\, y(t) \qquad (6)$$

Hence

$$P^{-1}(t)\theta(t) = \sum_{i=1}^{t} q_{it}\, \Phi(i)\, y(i) + (1-\lambda_1)..(1-\lambda_t)P^{-1}(0)\theta(0) \qquad (7)$$

Since the initial value $\theta(0) = 0$ the second term in (7) vanishes. Substituting for y(i) from (2) yields

$$\theta(t) = P(t)\sum_{i=1}^{t} q_{it}\Phi(i)\Phi^T(i)\,\theta^* + P(t)\sum_{i=1}^{t} q_{it}\Phi(i)v(i) \qquad (8)$$

Using (4) in (8) yields

$$\theta(t) = \theta^* - P(t)\prod_{i=1}^{t}(1-\lambda_i)\,\theta^* + P(t)\sum_{i=1}^{t} q_{it}\Phi(i)v(i) \qquad (9)$$

Since $0 \le \lambda_i < 1$, it is generally true that for large t,

$\prod^{t}_{i=1}(1-\lambda_i)$ is very small. Thus the second term in the right hand side of (9) can be neglected. Consider the subsequences of q_{it}, $\Phi(i)$ and v(i) obtained by considering only those instants for which λ_{it} and hence q_{it} is non-zero The third term in (9) can be replaced by a summation over these subsequences of length $t' \le t$. Taking expectations on both sides yields

$$\lim_{t\to\infty} E\{\theta(t)\} = \theta^* + E\{P(t)\sum_{\substack{i=1\\ \lambda_i\neq 0}}^{t'} q_{it}\Phi(i)v(i)\} \qquad (10)$$

It has been shown [3] that if the input is "persistently exciting" then λ_t tends to 0 and P(t) is bounded asymptotically. It follows then that P(t) tends to a constant, say P asymptotically. To simplify the above equation it is further assumed here that v(i) is uncorrelated with $\Phi(i)$ and $q_{it'}$, for time instants when λ_i is non-zero. Simulation studies confirmed that the correlation coefficient between $q_{it'}$ and v(i) is very close to zero. A possible explanation for this is that $q_{it'}$ which is actually the product of λ_i, $(1-\lambda_{i+1})$, ..., $(1-\lambda_{t'})$ is dependent on the data set $\{y_1 \ldots y_t\}$ as λ_i depends on y(i), y(i-1),..., y(1). A change in v(i) will not really affect the calculated value of $q_{it'}$, as long as t and t' are large enough. In addition, $\Phi(i)$ and v(i) are uncorrelated, as $\{v(i)\}$ is a white noise sequence. Therefore

$$E\{q_{it'}\Phi(i)v(i)\} = E\{q_{it'}\Phi(i)\}\, E\{v(i)\}$$

From (10) it follows that

$$\lim_{t\to\infty} E\{\theta(t)\} = \theta^* + E\{P\sum_{\substack{i=1\\ \lambda_i\neq 0}}^{t'} q_{it'}\Phi(i)\}\; E\{v(i)\}$$

and since $\{v_i\}$ is zero mean

$$\lim_{t\to\infty} E\{\theta(t)\} = \theta^*$$

Thus the estimator is shown to be asymptotically unbiased.

ASYMPTOTIC COVARIANCE

The asymptotic covariance of the estimates is a good measure of the steady state performance of recursive estimation algorithms. Finding an exact expression for the covariance of the OBE estimates is a formidable task because of the presence of the factor q_{it} in (9). This factor is related to the data in a highly non-linear fashion. However, after making a few simplifying assumptions, an upper bound for the covariance can be obtained.

With the same assumptions as before (9) can be reduced to

$$\theta(t) - \theta^* = P(t)\sum_{i=1}^{t} q_{it}\Phi(i)\, v(i) \qquad (11)$$

Let $\tilde\theta(t) = \theta(t) - \theta^*$, then since $P(t) = P^T(t)$

$$\tilde\theta(t)\tilde\theta^T(t) = P(t)\Big[\sum_{i=1}^{t}\sum_{j=1}^{t} q_{it}q_{jt}\Phi(i)\Phi^T(j)\, v(i)v(j)\Big]P(t)$$

$$= P(t)\Big[\sum_{i=1}^{t-1}\sum_{j=i+1}^{t} q_{it}q_{jt}\Phi(i)\Phi^T(j)\, v(i)\, v(j)\Big]P(t)$$

$$+ P(t)\sum_{i=1}^{t} q_{it}^2\Phi(i)\Phi^T(i)\, v^2(i)\, P(t)$$

$$+ P(t)\sum_{j=1}^{t-1}\sum_{i=j+1}^{t} q_{it}q_{jt}\Phi(i)\Phi^T(j)\, v(i)\, v(j)\, P(t) \qquad (12)$$

It is assumed here, as before, that for large t the factor q_{it}, at the updating instants, is uncorrelated with v(j) for all i and j, and that P(t) tends to a constant P asymptotically. Also since $\{v(i)\}$ is white, $\Phi(i)$ is uncorrelated with v(j) for all $j \ge i$ This would imply that in (12), the term $(q_{it}\, q_{jt}\Phi(i)\,\Phi^T(j)\, v(i))$ is uncorrelated with v(j) for all $j > i$ and the term $(q_{it}\, q_{jt}\,\Phi(i)\,\Phi^T(j)\, v(j))$ is uncorrelated with v(i) for all $i \ge j$. Taking expectations on both sides of the above expression yields

$$\lim_{t\to\infty} E\{\tilde\theta(t)\tilde\theta^T(t)\} =$$

$$\lim_{t\to\infty} E\Big\{\sum_{i=1}^{t-1}\sum_{j=i+1}^{t} P\, q_{it}q_{jt}\Phi(i)\,\Phi(j)^T v(i) P\Big\}\; E\{v(j)\}$$

$$+ \sum_{i=1}^{k} E\{P\, q_i^2\,\Phi(i)\,\Phi^T(i)\, P\}\,\sigma_v^2$$

$$+ \sum_{j=1}^{t-1}\sum_{i=j+1}^{t}\{P\, q_{it}\, q_{jt}\,\Phi(i)\,\Phi^T(j)\, v(j)\}\; E\{v(i)\} \qquad (13)$$

where σ_v^2 is the variance of the white noise sequence $\{v(i)\}$. Since $\{v(i)\}$ is zero mean, the first and last terms of (13) vanish and thus

$$\lim_{t\to\infty} E\{\tilde\theta(t)\tilde\theta^T(t)\} = \sigma_v^2\sum_{i=1}^{t} E\{P\, q_{it}\, q_{it}\,\Phi(i)\,\Phi^T(i)\, P\}$$

Since $q_{it}\,\Phi(i)\,\Phi^T(i)$ is positive semi-definite and since

$0 \le q_{it} \le \alpha$, where α is a user chosen upper bound on λ, it follows that

$$\lim_{t\to\infty} E\{\tilde{\theta}(t)\,\tilde{\theta}^T(t)\} \le \sigma_v^2 \;\; \alpha\, E\{P\sum_{i=1}^{t} q_{it}\,\Phi(i)\,\Phi^T(i)\,P\}$$

From (4), for sufficiently large t,

$$\sum_{i=1}^{t} q_{it}\,\Phi(i)\,\Phi^T(i) = P^{-1}(t)$$

and since $P(t) \to P$, this would imply that $P^{-1}(t) \to P^{-1}$. Hence for sufficiently large t

$$E\{\tilde{\theta}(t)\,\tilde{\theta}^T(t)\} \le \alpha\,\sigma_v^2\,E\{P\} \qquad (14)$$

or

$$E\{tr[\tilde{\theta}(t)\,\tilde{\theta}^T(t)]\} \le \alpha\,\sigma_v^2\,E\{tr[P]\} \qquad (15)$$

If the input sequence is persistently exciting, it can be shown [3] that there exists a constant μ such that $P \le \mu I$. Therefore from (14)

$$\lim_{t\to\infty} E\{\tilde{\theta}(t)\,\tilde{\theta}^T(t)\} \le \alpha\,\sigma_v^2\,\mu I$$

It must be emphasized that this upper bound is rather conservative, because the maximum value of q_{it} will actually be much lower than α.

IMPROVED ALGORITHM

In some applications,the parameter estimates obtained by applying the OBE algorithm fluctuate excessively. This naturally causes the algorithm to have a larger steady state error and parameter covariance as compared to the recursive least-squares algorithm. These fluctuations are highly undesirable. A possible explanation for this unwanted property is presented below.

The OBE algorithm of [3] has the property that the quantity $\sigma^2(t)$, which is related to the size of the optimal bounding ellipsoid E_t, is monotone non-increasing. When the size of the ellipsoid E_t does decrease, thus decreasing the size of the region in which the estimates are constrained to lie, the parameter estimates themselves which correspond to the center of E_t may shift considerably. This shifting of the center of the optimal bounding ellipsoid accounts for the fluctuation of parameter estimates.

For the OBE algorithm, the parameter estimate update equation is

$$\theta(t) = \theta(t-1) + \lambda_t\,P(t)\Phi(t)\,\delta(t)$$

Thus the amount of variations of the estimates is directly proportional to λ_t. Hence one way to reduce the fluctuation is to reduce the value of λ_t. However if λ_t is reduced by the same amount for all t, then the tracking ability of the algorithm deteriorates and the bounding ellipsoid at every iteration t, will no longer be optimal in the sense that $\sigma^2(t)$ is minimum.Therefore a compromise value of λ_t is required which does reduce the amount of fluctuation and also retains the tracking property. A possible solution is to multiply the value of λ calculated at each iteration by a scaling factor M(t), which is small for isolated updates but close to unity for a series of successive updates. A possible choice for the scaling factor is

$$M(t) = \exp(-s\,/\,nup(t)) \qquad (17)$$

with

s a user chosen constant in the range $0 < s < 10$, and

$$nup(t) = 0 \quad \text{if } \lambda_t = 0$$

otherwise

$$nup(t) = nup(t-1) + 1$$

If there is a string of updates, as would be the case if the algorithm is trying to track the parameters, then only the first λ in the string would be decreased significantly and the rest would be decreased marginally. Thus isolated updates which may be due to noisy observations are reduced significantly thereby reducing the "random" fluctuation in parameter estimates. The modified algorithm thus has greater resistance to outliers. A similar scaling factor has been used to reduce the steady state mean-square prediction error in the exponentially weighted least-squares algorithm [5].

It must be emphasized that the choice of the scaling factor used here is somewhat arbitrary. A scaling factor is required which will be small for $nup(t) \le 2$ and will be close to unity for larger values of nup(t). The scaling factor used here satisfies these requirements. Simulations have been performed to test the effectiveness of the scaling factor and the results are presented in the next section.

SIMULATION RESULTS

In order to verify the above analysis, Monte Carlo simulations were performed. The OBE algorithm of [3] was applied to 100 different data sets generated by the following AR (2) model

$$y(t) = -0.4\,y(t-1) - 0.85\,y(t-2) + v(t)$$

where $\{v(t)\}$ is an i.i.d. sequence having a uniform distribution in [-1.0, 1.0]. Each data set contains 1000 data points. For each data set, the estimate vector obtained at each iteration was subtracted from the true parameter vector. The sample bias defined by

$$\frac{1}{100}\sum_{i=1}^{100}[\,\theta\,(t) - \theta^*\,]$$

was computed for t = 1 to 1000. In Figure 1, this average is plotted against the iteration index t. As the number of iterations increases, the bias becomes closer to 0, thus experimentally verifying the asymptotic unbiasedness of the estimator.

The sample variance for each component of the estimate vector defined as

$$\frac{1}{100}\sum_{i=1}^{100}[\,\theta_t\,(j,i) - \theta^*(j)\,]^2$$

where $\theta_t(j,\cdot)$ is the j 'th component of the estimate of the i'th data set at time instant t, is calculated for t going from 1 to 1000. The sample variances for each component are added and plotted in Fig. 2 against the iteration index t. The asymptotic value of P(t) is obtained as

$$P = \frac{1}{100}\sum_{i=1}^{100} P_{1000}(i)$$

where $P_{1000}(i)$ is the the matrix $P(1000)$ obtained from the i'th data set. The upper bound is computed using (15) with $\alpha = 0.5$ and $\sigma_v^2 = 0.33$. The computed upper bound is equal to 0.84.

The OBE algorithm of [3], with and without the scaling factor, was applied to data generated by the above AR (2) model. A scaling factor with $s = 2$, (c.f. (17)), was found to yield good results. The normal and smoothed parameter estimates are plotted against the iteration index t in Figure 3 . It is clear from the plot that using a scaling factor on λ reduces the amount of fluctuation and yields smoother parameter estimates.

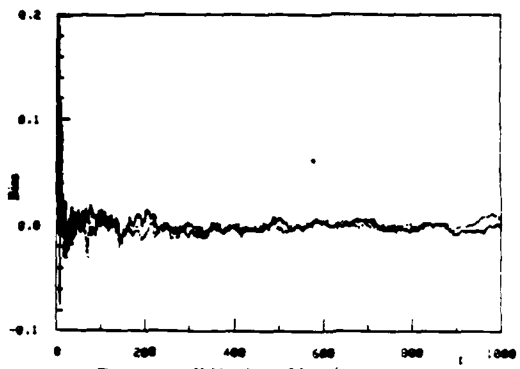


Figure 1 Unbiasedness of the estimator

CONCLUSION

We have performed an analysis of certain statistical properties of a new recursive estimation algorithm. The estimator is shown to be asymptotically unbiased and an upper bound on the asymptotic covariance of parameter estimates has been derived with some reasonable assumptions. The analysis has been verified through Monte Carlo simulations of a particular AR(2) model. Finally an improved algorithm has been proposed and shown to yield smoother estimates.

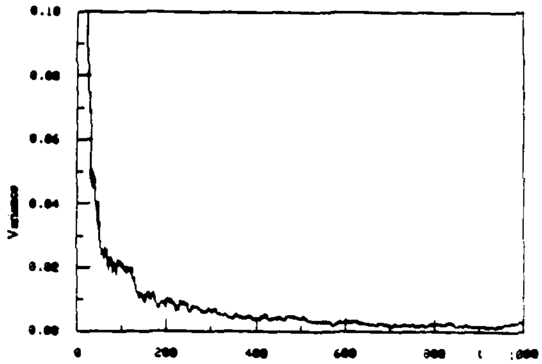


Figure 2 Variance of the estimator

REFERENCES

[1] E. Fogel and Y.F.Huang, " On the value of information in system identification - bounded noise case," *Automatica* , vol. 18, No. 2, pp. 229-238, 1982.

[2] Y. F. Huang, " A recursive estimation algorithm using selective updating for spectral analysis and adaptive signal processing, "*IEEE Trans Acoust.,Speech and Sig. Process.*, vol. 34, pp. 1131-1134, Oct. 1986.

[3] S. Dasgupta and Y. F. Huang, " Asymptotically convergent modified recursive least-squares," *IEEE Trans. Inform. Theory*, vol. IT-33, pp. 383-392, May 1987.

[4] Y. F. Huang and M. Yeh, " Adaptive hybrid balancing based on a novel recursive estimation algorithm and spectral analysis,"*IEEE Trans Acoust.,Speech and Sig. Process.*, vol. 35, pp. 397-399, March 1987.

[5] L. Ljung and T. Soderstrom ,*Theory and Practice of Recursive Identification*. Cambridge, MA: MIT Press 1983.

[6] B.B.Peterson and K.S. Narendra , " Bounded Error Adaptive control," *IEEE Trans. Automatic Control*, Vol. AC-27,No.6, Dec. 1982, pp. 1161-1168.

[7] A. K. Rao, " Applications and extensions of a novel recursive estimation algorithm with selective updating," M. S. Thesis, Dep. of Electrical & Computer Eng., Univ. of Notre Dame, Notre Dame, IN Aug. 1987.

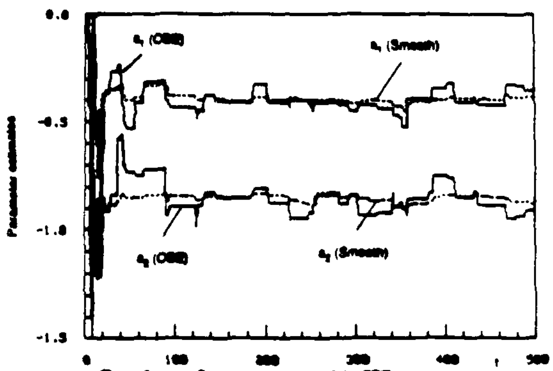


Figure 3 Parameter estimates of the OBE and the improved OBE algorithm

ACKNOWLEDGMENT

This work has been supported in part by the National Science Foundation under Grant ECS- 850528 and in part by the Office of Naval Research under contract N00014-87-K-0284.

Publication V.B.5.

PROCEEDINGS

TWENTY-SIXTH ANNUAL ALLERTON CONFERENCE ON COMMUNICATION, CONTROL, AND COMPUTING

VOLUME I

Mark W. Spong
Bruce Hajek
Conference Co-Chairmen

Conference held
September 28-30, 1988
Allerton House
Monticello, Illinois

Sponsored by
The Coordinated Science Laboratory
and
The Department of Electrical and Computer Engineering
of the
UNIVERSITY OF ILLINOIS
at
Urbana-Champaign

AN EXTENDED OBE ALGORITHM FOR ARMA PARAMETER ESTIMATION

ASHOK K. RAO*, YIH-FANG HUANG* and SOURA DASGUPTA**

*Department of Electrical & Computer Engineering, University of Notre Dame, Notre Dame, IN 46556. ** Department of Electrical & Computer Engineering, University of Iowa, Iowa City, Iowa 52242.

ABSTRACT

Recently, there appears to be a resurgence of interest in the area of membership-set parameter estimation. One such algorithm, the so-called Optimal Bounding Ellipsoid (OBE) algorithm, proves to be appealing in theory and practice. The algorithm, which features a discerning update strategy, was first derived for the recursive estimation of parameters of autoregressive with auxiliary input (ARX) models. This paper investigates an extension of the algorithm to ARMA parameter estimation. The convergence analysis is complicated due to the discerning update strategy which incorporates an information dependent forgetting factor. It is shown that if the input noise is bounded and if the process is stable, then the *a posteriori* prediction error is bounded even without the SPR condition. This is in sharp contrast to the crucial role of the SPR condition in the ELS and output error algorithms.

I. INTRODUCTION

The autoregressive moving average(ARMA) parameter estimation problem arises in many adaptive signal processing applications such as speech processing, seismic data processing and channel equalization. Typically, samples of the measured signal y(t) are modeled as the output of an IIR filter driven by unknown white noise w(t) [1]. The ARMA model is described by the temporal recursion

$$y(t) = a_1\, y(t-1) + .. + a_n\, y(t-n) + w(t) + c_1\, w(t-1) + .. + c_r\, w(t-r) \qquad (1)$$

Fitting this ARMA model to the measured data y(t), t =1..T , requires the estimation of the parameters $a_1 .., a_n, c_1 .., c_r$. Recursive schemes like the extended least-squares (ELS), the recursive maximum likelihood (RML) and multi-stage least-squares algorithms have been used to estimate ARMA parameters [2,3]. The ELS algorithm uses the *a posteriori* prediction error $\varepsilon(t)$, as an estimate of w(t). The regressor vector is formed from y(t-1),..., y(t-n) and $\varepsilon(t-1)$,..., $\varepsilon(t-r)$. The standard recursive least-squares (RLS) algorithm is then employed to update the estimates. The algorithm is conceptually simple but restrictive in the sense that convergence of the algorithm can be assured only if the underlying transfer function $H(q^{-1}) = 1/C(q^{-1}) - 1/2$ is strictly positive real (SPR), with q^{-1} being the delay operator and

$$C(q^{-1}) = 1 + c_1 q^{-1} + c_2 q^{-2} + .. + c_r q^{-r}$$

The RML algorithm, which uses a filtered version of the regressor vector used in the ELS algorithm, does not require $H(q^{-1})$ to be SPR. However the estimates have to be monitored and projected into a stability region to ensure convergence[2].

In addition to the aforementioned least-squares based methods, there exists a different class of estimation algorithms that estimate membership sets of parameters which are consistent with the measurements, model structure and noise constraints [4]-[8]. These algorithms are particularly useful when the noise distribution is unknown but constraints in the form of bounds on the instantaneous values of the noise are available. The model and noise representations commonly used are

$$y(t) = \theta^{*T} \Phi(t) + v(t), \quad |v(t)| \le r^{1/2}(t)$$

where {y} is a sequence of scalar observations, θ^* is the parameter vector to be identified, $\Phi(t)$ is a n-vector of variables known at time t, {v} is the noise sequence and $\pm\, r^{1/2}(t)$ are the time varying noise bounds. Given a sequence $\{y(i), \Phi(i)\}$, i=1..k, the optimal membership set

$$\psi^0_k = \cap^{k}_{i=1} S_i$$

where

$$S_i = \{ \theta : (y(i) - \theta^T \Phi(i))^2 \leq r(i), \ \theta \varepsilon \mathbf{R}^n \}$$

From a geometrical point of view S_i is a convex polytope in $\mathbf{R}^n$ and contains the true parameter vector. Finding ψ^0_k is computationally intractable and it is therefore necessary to approximate ψ^0_k by some set which encloses it tightly and which can be described and updated economically [9]. The different membership-set algorithms differ in the way the optimal membership set is approximated and in the method used to obtain a tight enclosure.

Among these algorithms based on membership sets, a seminal recursive algorithm is the so-called optimal bounding ellipsoid (OBE) algorithm[6-8]. One of the main features of this algorithm is a discerning update strategy. This feature, obtained by the introduction of an information dependent updating/forgetting factor, yields a modular structure thereby increasing the potential for concurrent and pipelined processing of signals. The presence of such a forgetting factor also gives the algorithm the ability to track time varying parameters. The algorithm has the advantageous feature of automatic asymptotic cessation of updates in the fixed parameter case.

In this paper, we extend one of these OBE algorithms[8] to the ARMA case. For the ARMA parameter estimation problem, the OBE algorithm cannot be applied in its present form. However, by assuming that the input white noise is bounded in magnitude, the OBE algorithm can be extended in a manner similar to the ELS algorithm. The convergence analysis of the resulting algorithm, as opposed to that of the ELS algorithm, is deterministic and is performed under the assumptions that the process is stable and that the noise is bounded.The *a posteriori* prediction error is shown to be bounded without imposing any SPR condition. This is in contrast to the convergence analysis of the ELS or output error algorithms in which the SPR condition is used to prove boundedness of the prediction errors and convergence of parameter estimates[10]. By imposing a persistence of excitation condition on the regressor vector, the *a priori* prediction error of the extended OBE algorithm is shown to be bounded and the parameter estimates are shown to converge to a neighborhood of the true parameter vector.

The paper is organized in the following manner. In Section II, a brief review of the OBE algorithm and its properties is presented. In Section III, the algorithm is extended to ARMA parameter estimation. Convergence analysis of the extended algorithm is performed in Section IV. The performance of the algorithm is compared with the ELS algorithm through simulation studies in Section V. Section VI concludes the paper.

II. THE OBE ALGORITHM

Consider the ARX model described by

$$y(t) = a_1 y(t-1) + .. + a_n y(t-n) + b_0 u(t) + b_1 u(t-1) + .. + b_m u(t-m) + v(t)$$

The above equation can be recast as :

$$y(t) = \theta^{*T} \Phi(t) + v(t) \qquad (2)$$

where

$$\theta^* = [a_1, a_2 \dots a_n, b_0, b_1, \dots, b_m]^T$$

is the vector of true parameters and

$$\Phi(t) = [y(t-1), y(t-2), .. y(t-n), \ u(t), u(t-1), .. u(t-m)]^T$$

is the regressor vector. It is assumed that the noise is uniformly bounded in magnitude, i.e., there exists $\gamma_0 \geq 0$, such that

$$v^2(t) \leq \gamma_0^2 \qquad \text{for all t, hence}$$

$$(y(t) - \theta^{*T} \Phi(t))^2 < \gamma_0^2$$

Let S_t be a subset of the euclidean space $\mathbf{R}^{n+m+1}$, defined by

$$S_t = \{ \theta : (y(t) - \theta^T \Phi(t))^2 \leq \gamma_0^2 \ , \theta \varepsilon \mathbf{R}^{n+m+1} \}$$

The OBE algorithm starts off with a large ellipsoid, E_0, in $\mathbf{R}^{n+m+1}$ which contains all possible values of the modelled parameter θ^*. After the first observation $y(1)$ is acquired, an ellipsoid is found which bounds the intersection of E_0 and the convex polytope S_t. This ellipsoid must be optimal in some sense, say minimum volume[6-7] or by any other criterion[6-8], to hasten convergence. Denoting the optimal ellipsoid by E_1 one can proceed exactly as before with the future observations and obtain a sequence of optimal bounding ellipsoids $\{E_t\}$. The center of the ellipsoid E_t can be taken as the parameter estimate at the t-th instant and is denoted by $\theta(t)$. If at a particular time instant i, the resulting optimal bounding ellipsoid would be of a " smaller size ", thereby implying that the data point y(i) conveys some " information " regarding the parameter estimates, then the parameters are updated. Otherwise E_i is set equal to E_{i-1} and the parameters are not updated.

Let the ellipsoid E_{t-1} at the (t-1)-th instant be formulated by

$$E_{t-1} = \{ \theta : (\theta - \theta(t-1))^T P^{-1}(t-1) (\theta - \theta(t-1)) \leq \sigma^2(t-1) \}$$

for some positive definite matrix P (t-1) and a non-negative scalar $\sigma^2(t-1)$. Then, given $y(t)$, an ellipsoid which bounds $E_{t-1} \cap S_t$ "tightly" is

$$\{ \theta : (1-\lambda_t)(\theta - \theta(t-1))^T P^{-1}(t-1)(\theta - \theta(t-1)) + \lambda_t (y(t) - \theta^T \Phi(t))^2$$

$$\leq (1-\lambda_t)\sigma^2(t-1) + \lambda_t \gamma_0^2 \} \tag{3}$$

where the forgetting factor $\lambda(t)$ satisfies $0 \leq \lambda(t) < 1$. The size of the bounding ellipsoid is related to the scalar $\sigma^2(t-1)$ and the eigenvalues of P(t-1). The update equations for $\theta(t)$, P(t) and $\sigma^2(t)$ are derived in [8]. The optimal ellipsoid which bounds the intersection of E_{t-1} and S_t is defined in terms of an optimal value of λ_t. For the OBE algorithm of [8], the optimum value λ^*_t is determined by minimization of $\sigma^2(t)$ with respect to λ_t at every time instant. The minimization procedure results in λ^*_t being set equal to zero (no update) if

$$\sigma^2(t) + \delta^2(t) \leq \gamma^2(t) \tag{4}$$

If (4) is not satisfied, then the optimal value of λ_t is computed. The parameter estimation procedure is depicted in Fig. 1. An outgrowth of the modular recursive estimation procedure is a parallel-pipelined networking structure [11]. The algorithm is such that the computational complexity of the information evaluation (IE) procedure is much less than that of the updating procedure (UPD). Since, in general, a good number of data samples would be rejected by the IE, both the IE and the UPD would involve significant amounts of idle time. A viable scheme then is to configure a parallel-pipelined network comprising of such modular estimators to process signals from multiple channels. Apart from reducing hardware costs, such a scheme would offer increased reliability since the failure of one UPD processor would not cause any of the channels to fail, in contrast to a system with a dedicated UPD processor for each channel.

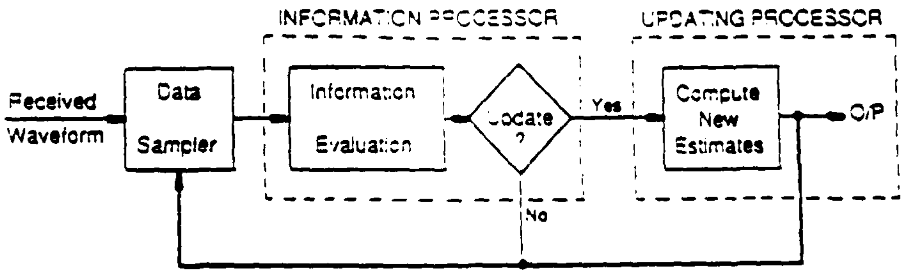


Figure 1 Modular recursive parameter estimator

III. EXTENSION TO ARMA MODELS

The ARMA model described by (1), can be rewritten as

$$w(t) = y(t) - \theta^{*T}\Phi'(t) \tag{5}$$

where θ^* is the vector of true parameters and is now defined by

$$\theta^* = [a_1, a_2 \ldots a_n, c_1, c_2 \ldots, c_r]^T$$

$$\Phi'(t) = [y(t-1), \ldots, y(t-n), w(t-1), \ldots, w(t-r)]^T$$

Here again, $w(t)$ is assumed to be bounded in magnitude by γ_0. Since the values of the noise sequence $\{w(t)\}$ are not available, the regressor vector $\Phi'(t)$ is not known exactly. If, however, at time t, an estimate of θ^*,

$$\theta(t) = [a_1(t), \ldots, a_n(t)\ c_1(t), \ldots, c_r(t)]^T \tag{6}$$

is available, $w(t)$ could be estimated by its estimate $\varepsilon(t)$ according to

$$\varepsilon(t) = y(t) - \theta^T(t)\Phi(t)$$

where

$$\Phi(t) = [y(t-1), \ldots y(t-n), \varepsilon(t-1), \ldots \varepsilon(t-r)]^T \tag{7}$$

It follows from (5) and the boundedness of $|w(t)|$ that

$$(y(t) - \theta^{*T}\Phi'(t))^2 \le \gamma_0^2$$

Hence if $\varepsilon(t) \approx w(t)$, then $\Phi'(t) = \Phi(t)$ and for a suitable $\gamma \ge 0$, such that $\gamma^2 > \gamma_0^2$

$$(y(t) - \theta^{*T}\Phi(t))^2 \le \gamma^2 \tag{8}$$

Thus $\theta^* \in S_t$, where S_t is defined by

$$S_t = \{\theta : (y(t) - \theta^T\Phi(t))^2\} \le \gamma^2, \theta \in \mathbf{R}^{n+r}\}$$

Hence if the difference between $\varepsilon(t)$ and $w(t)$ is small, applying the OBE algorithm will yield a sequence of bounding ellipsoids $\{E_t\}$ in the parameter space. If (8) holds for all t and $\theta^* \in E_0$, then it is easy to see that $\theta^* \in E_t$ for all time instants t. The optimal bounding ellipsoid E_t is described by

$$E_t = \{\theta \in \mathbf{R}^{n+r} : (\theta - \theta(t))^T P^{-1}(t)(\theta - \theta(t)) \le \sigma^2(t)\} \tag{9}$$

and the update equations which follow directly from [8] are

$$P^{-1}(t) = (1-\lambda_t)P^{-1}(t-1) + \lambda_t\Phi(t)\Phi^T(t) \tag{10a}$$

$$\theta(t) = \theta(t-1) + \lambda_t P(t)\Phi(t)\delta(t) \tag{10b}$$

$$\delta(t) = y(t) - \theta^T(t-1)\Phi(t) \tag{10c}$$

$$\sigma^2(t) = (1-\lambda_t)\sigma^2(t-1) + \lambda_t\gamma^2 - \frac{\lambda_t(1-\lambda_t)\delta^2(t)}{1-\lambda_t + \lambda_t G(t)} \tag{10d}$$

where

$$G(t) = \Phi^T(t)P(t-1)\Phi(t) \tag{10e}$$

The above recursive relations along with the initial values

$$P^{-1}(0) = I,\ \theta(0) = 0 \text{ and } \sigma^2(0) = 1/\varepsilon \text{ with } \varepsilon \ll 1$$

form the basis of the extended optimal bounding ellipsoid (EOBE) estimation algorithm[12]. The matrix inversion lemma can be used to obtain the recursion for P(t)

$$P(t) = \frac{1}{1-\lambda_t}\left[P(t-1) - \frac{\lambda_t P(t-1)\Phi(t)\Phi^T(t)P(t-1)}{1-\lambda_t + \lambda_t G(t)}\right] \tag{11}$$

It is easily shown from (10d) that for the EOBE algorithm, minimizing $\sigma^2(t)$ with respect to λ_t at every time instant yields the same updating criterion (4) and the same algorithm for determining the optimum value of the forgetting / updating factor λ^*_t, as in [8]. The algorithm thus retains the discerning update strategy and the modular adaptive filter structure

Choice of γ^2

Finding a value of γ^2 to ensure that (8) holds for all time instants is not easy. Instead, we will show in the next section that choosing γ^2 to be an upper bound on the square of the magnitude of the output y(t) will ensure that the parameter estimates obtained by applying the EOBE algorithm converge to a neighborhood of the true parameters.

If the ARMA process is stable then if $|w(t)|$ is bounded so is $|y(t)|$. The output process y(t) can be expressed as

$$y(t) = h(t) * w(t) ,$$

where * denotes discrete convolution and h(t) is the equivalent impulse response. Stability of the process implies that there exists a finite M such that

$$\sum_{i=0}^{\infty} | h(i) | \quad \leq M < \infty \tag{12}$$

and if $| w(t) | \leq \gamma_0$ then

$$| y(t) | \quad \leq \sum_{i=0}^{\infty} | h(i) | \, | w(t-i) | \leq M \gamma_o = \gamma \tag{13}$$

In order to obtain a value for γ, the user would therefore need the bound on $|w(t)|$ and an estimate of M. Or alternatively, the outputs could be monitored and a bound on $|y(t)|$ could be obtained before starting the actual parameter estimation procedure. A loose upper bound on either M or the magnitude of the output would suffice since simulations have shown that using values for γ that are several times larger or smaller than the actual bound on $|y(t)|$ has no deleterious effects on the quality of estimates or convergence rate.

IV. CONVERGENCE ANALYSIS

We will first show that if the updating factor sequence is that which minimizes σ^2 at every instant (denoted by $\{ \lambda^*_t \}$) then the parameter estimates converge. To show that, the following lemma will be needed.

Lemma 1. If the magnitude of the noise w(t) is upper bounded and the process is stable then $\sigma^2(t)$, defined in (10d), is always non-negative provided that the initial values $\theta(0)$, $P^{-1}(0)$ and $\sigma^2(0)$ are chosen such that $\theta^T(0)P^{-1}(0)\theta(0) \leq \sigma^2(0)$

Proof. From (10a), (10b) and (10c)

$$\theta^T(t) P^{-1}(t) \theta(t) = (1-\lambda_t) \theta^T(t-1) P^{-1}(t-1) \theta(t-1) + \lambda_t [y^2(t) - \delta^2(t) (1- \lambda_t \Phi^T(t) P(t) \Phi(t))]$$

$$= (1- \lambda_1) (1- \lambda_2) \ldots (1-\lambda_t) \theta^T(0) P^{-1}(0) \theta(0)$$

$$+ \sum_{i=1}^{t} q_{it} [y^2(i) - \delta^2(i) (1 - \lambda_i \Phi^T(i) P(i) \Phi(i))]$$

where

$$q_{it} = \lambda_i (1- \lambda_{i+1}) \ldots (1- \lambda_t) \geq 0 \quad \forall \, i, t.$$

Since the process is stable and $\{w(t)\}$ is bounded, there exists a γ^2 such that $y^2(t) \leq \gamma^2$ The positive semi-definiteness of $P^{-1}(t)$ will therefore imply that

$$\prod_{i=1}^{t}(1-\lambda_i)\;\theta^T(0)P^{-1}(0)\theta(0) + \sum_{i=1}^{t} q_{it}[\gamma^2 - \delta^2(i)(1-\lambda_i\Phi^T(i)P(i)\Phi(i))] \geq 0 \tag{14}$$

But from (11)

$$\lambda_i\Phi^T(i)P(i)\Phi(i) = \frac{\lambda_i G(i)}{1-\lambda_i+\lambda_i G(i)} \tag{15}$$

Using (15) in (14) yields

$$\prod_{i=1}^{t}(1-\lambda_i)\;\theta^T(0)P^{-1}(0)\theta(0) + \sum_{i=1}^{t} q_{it}[\gamma^2 - \delta^2(i)\frac{1-\lambda_i}{1-\lambda_i+\lambda_i G(i)}] \geq 0 \tag{16}$$

From (10d), a non-recursive expression for $\sigma^2(t)$ can be obtained as

$$\sigma^2(t) = (1-\lambda_1)(1-\lambda_2)\ldots(1-\lambda_t)\sigma^2(0) + \sum_{i=1}^{t} q_{it}[\gamma^2 - \frac{\delta^2(i)(1-\lambda_i)}{1-\lambda_i+\lambda_i G(i)}] \tag{17}$$

Since $\sigma^2(0) \geq \theta^T(0)P^{-1}(0)\theta(0)$, (17) implies that $\sigma^2(t) \geq 0$ for all t. This result will hold for any sequence of forgetting factors $\{\lambda_t\}$ with $0 \leq \lambda_t < 1$.

Theorem 1. If the assumptions of Lemma 1 are satisfied then

1. $\lim_{t\to\infty} \varepsilon^2(t) \in [0, \gamma^2]$ (18)

2. $\lim_{t\to\infty} \sigma^2(t) \in [0, \gamma^2]$ (19)

Proof. See [13].

Note that throughout this section, expressions like (18) should not be taken to mean that $\lim_{t\to\infty}\varepsilon^2(t)$ exists but rather that $\varepsilon^2(t)$ becomes asymptotically less than or equal to γ^2.

The next lemma relates the positive definiteness of $P^{-1}(t)$ to the richness of the regressor vector $\Phi(t)$.

Lemma 2. If there exist positive α_3 and N such that for all t

$$\sum_{i=t}^{t+N} \Phi(i)\Phi^T(i) \geq \alpha_3 I > 0 \tag{20}$$

then there exists a positive α_4 such that

$$P^{-1}(t) \geq \alpha_4 I > 0$$

Proof of the lemma is the same as that of Theorem 4.1 of [8], it is thus omitted here.

Remark. The positive definiteness of $P^{-1}(t)$ implies that the eigenvalues of P(t) are upper bounded.

Theorem 2. If assumptions of Lemma 1 are satisfied and (20) holds then the EOBE algorithm ensures :

(a) Parameter difference convergence

$$\lim_{t \to \infty} \| \theta(t) - \theta(t-k) \| = 0 \qquad (21)$$

for any finite k.

(b) Bounded *a priori* prediction errors

$$\lim_{t \to \infty} \delta^2(t) \in [0, \gamma^2] \qquad (22)$$

(c) Bounded parameter misadjustment

$$\lim_{t \to \infty} \| \theta(t) - \theta^* \|^2 \leq M\gamma^2 < \infty \qquad (23)$$

for some finite M.

Proof.
(a) From (10b) and (11)

$$\| \theta(t) - \theta(t-1) \|^2 = \frac{\lambda_t^{\,2} \Phi^T(t) P^2(t-1) \Phi(t) \delta^2(t)}{(1 - \lambda_t + \lambda_t G(t))^2}$$
$$\leq e_{max}\{ P(t-1) \} \frac{\lambda_t^{\,2} G(t) \delta^2(t)}{(1 - \lambda_t + \lambda_t G(t))^2} \qquad (24)$$

If (20) holds then by Lemma 2, e_{max} { P(t-1) }, the maximum eigenvalue of P(t-1), is bounded for all t. The other term on the right hand side is shown to tend to zero in the proof of Theorem 1. Thus

$$\| \theta(t) - \theta(t-1) \| \to 0 \qquad (25)$$

Applying the Minkowski inequality to $\| \theta(t) - \theta(t-k) \|$ and using (25) completes the proof of (21).

(b) From (10e) and (13) it follows that

$$G(t) \leq e_{max} \{P(t-1)\} [n\gamma^2 + r \max_{t-r \leq i \leq t-1} \varepsilon^2(i)] \qquad (26)$$

where n is the order of the AR process and r is the order of the MA process. If (20) holds then e_{max} { P(t-1) }, the maximum eigenvalue of P(t-1), is bounded. Since {$\varepsilon(t)$} is bounded by Theorem 1, {G(t)} is bounded. Then it can be shown (see Theorem 3.2 of [8]), that the *a priori* prediction errors satisfy (22).

(c) Since w(t) and y(t) are bounded by γ and $\varepsilon(t)$ is aymptotically bounded by Theorem 1, it can be shown that for large enough t

$$\sum_{l=t}^{t+N} \theta^T(l) \Phi(l) \Phi^T(l) \theta(l) \leq O(\gamma^2) \qquad (27)$$

where $\theta(t) = \theta(t) - \theta^*(t)$, and N is given by Lemma 2. Then the result follows by using (21) and (20) in (27) and performing some algebraic manipulations.

Remark: To summarize, the above analysis has shown that if the process is stable and if the driving noise is bounded then the *a posteriori* prediction errors are bounded. In addition if a richness condition is imposed on the regressor vector, then the *a priori* prediction errors are bounded and the parameter estimates are asymptotically contained within a neighborhood of the true parameters.

V. SIMULATION RESULTS

Simulations have been performed to investigate the performance of the EOBE algorithm vis a vis the ELS algorithm. In this paper, we present simulation results for two examples- a broad band ARMA (3,3) process and a non SPR ARMA(3,3) process where the indices p, q in an ARMA(p,q) process refer to the orders of the $A(q^{-1})$ and $C(q^{-1})$ polynomials, respectively.

Example 1. Broad band ARMA (3,3) process
The output data $\{y(t)\}$ is generated by the following difference equation
$y(t) = -0.4\, y(t-1) + 0.2\, y(t-2) + 0.6\, y(t-3) + w(t) - 0.6\, w(t-1) + 0.2w(t-2) + 0.4\, w(t-3)$
The noise sequence $\{w(t)\}$ is generated by a pseudo-random number generator with a uniform probability distribution in [-1.0, 1.0]. The upper bound γ^2 was set equal to 25.0. The parameter estimates were obtained by applying the EOBE algorithm to 1000 point data sequences. One hundred runs of the algorithm were performed on the same model but with different input noise sequences. The average squared parameter error L(t), is computed for the AR coefficients according to the formula

$$L(t) = \frac{1}{100} \sum_{j=1}^{100} l_j(t)$$

where $l_j(t)$, the squared AR parameter error at time t for the j'th run, is defined by

$$l_j(t) = \sum_{i=1}^{n} (a_i(t) - a_i)^2$$

with a_i and $a_i(t)$ being defined by (1) and (6), respectively. The average squared parameter error for the MA coefficients is defined analogously. Fig. 2 displays the average squared estimation errors for AR and MA parameters using the EOBE and ELS algorithms. The figures show that the performance of the two algorithms is comparable. It may be noted that the AR parameter estimates have markedly lower variance (about the true parameters) than the MA parameters. The average number of updates for the EOBE algorithm was 139 for 1000 point data sequences. Thus less than 15% of the samples are used for updates, as compared to the ELS algorithm which updates at every sampling instant.

The tracking capability of the EOBE algorithm(with α=0.2), was compared with that of the ELS algorithm (with forgetting factor=0.99). The same model was used to generate 400 data points. The parameters were then changed by 150% and the next 400 points were generated. Finally the last 200 points were generated by using the original parameters. The average squared parameter error was evaluated over 25 runs and is shown in Figure 3. Even though the formulation of bounding ellipsoids is based on the assumption that the parameters are constant, the simulation results show that the algorithm is able to accommodate changes in model parameters. Analysis of the tracking ability of the algorithm is currently under investigation.

Example 2. Non SPR ARMA(3,3) process

The output data $\{y(t)\}$ is generated by the following difference equation
$y(t) = -0.6\, y(t-1) - 0.58\, y(t-2) - 0.464\, y(t-3) + w(t) + 0.2\, w(t-1) + 0.6w(t-2) + 0.2\, w(t-3)$
The noise sequence is generated as in the first example. The upper bound γ^2 was set equal to 3.25. The maximum values (taken over 25 runs of the algorithm) of the squared residual errors, at each iteration were computed and were well within the upper bound γ^2 even though the SPR condition is violated.

VI. CONCLUSION

A recursive parameter estimation algorithm has been extended for ARMA parameter estimation. The main features of the algorithm are a membership set theoretic formulation and a discerning update strategy. Convergence analysis of the algorithm has been performed under the assumptions that the process is stable and that the noise is bounded. The results of the analysis are that the algorithm yields bounded *a posteriori* prediction errors without SPR or persistence of excitation type condition. With a persistence of excitation condition on the regressor vector, boundedness of the *a priori* prediction error can then be established and the parameter estimates are shown to converge to a neighborhood of the true parameters. Simulation results show that the performance of the algorithm is comparable to the ELS algorithm while requiring far fewer updates.

VII. ACKNOWLEDGEMENT

This work has been supported in part by the National Science Foundation under Grant MIP-87-11174; in part by the office of Naval Research under Contract N00014-87-k- 0284, and in part by the National Science Foundation under Grant ECS-8618240.

REFERENCES

[1] B. Friedlander, " System identification techniques for adaptive signal processing, "*IEEE Trans. Acoust. Speech and Signal Process.*, vol. ASSP-30, No. 2, pp. 240-246, April 1982.

[2] L. Ljung and T. Soderstrom,*Theory and Practice of Recursive Identification*, Cambridge, Mass: MIT Press, 1983.

[3] D. Q. Mayne *et al*, " Linear identification of ARMA processes," *Automatica* vol. 18, No. 4, pp. 461-466, July 1982.

[4] F. C. Schweppe, " Recursive state estimation : unknown but bounded errors and system inputs, " *IEEE Trans. Automatic Control*, vol. AC-13, No.1, pp. 22-28, Feb. 1968.

[5] M. Milanese and G. Belaforte, " Estimation theory and uncertainty intervals," *IEEE Trans. Automatic Control*, vol. AC-27, No. 2, pp. 408-414, April 1982.

[6] E. Fogel and Y.F. Huang, " On the value of information in system identification - bounded noise case," *Automatica*, vol. 18, No. 2, pp. 229-238, March 1982.

[7] Y. F. Huang, " A recursive estimation algorithm using selective updating for spectral analysis and adaptive signal processing,"*IEEE Trans. Acoust.Speech and Signal Process.*, vol. ASSP-34, No. 5, pp. 1131-34 , October 1986.

[8] S. Dasgupta and Y.F. Huang, " Asymptotically convergent modified recursive least-squares," *IEEE Trans. Information Theory*, vol. IT-33, No.3, pp. 383-392, May 1987.

[9] J. P. Norton, " Identification and application of bounded-parameter models," *Automatica*, vol. 23, No. 4, pp. 497-507, 1987.

[10] G. C. Goodwin and K.S. Sin, *Adaptive Filtering, Prediction and Control*, Englewood Cliffs, NJ: Prentice-Hall, 1982.

[11] Y. F. Huang and R. W. Liu, " A high level parallel-pipelined network architecture for adaptive signal processing, " in *Proc. 26th IEEE Conf. on Decision and Control*, vol.1, pp. 662-666, Dec. 1987.

[12] A. K. Rao, " Applications and extensions of a novel recursive estimation algorithm with selective updating," M.S. Thesis, Dept. of Electrical & Computer Engineering, University of Notre Dame, Notre Dame, IN. Aug. 1987.

[13] A. K. Rao, Y. F. Huang and S. Dagupta, " ARMA parameter estimation using a novel recursive estimation algorithm with selective updating," submitted for publication to *IEEE Trans. Acoust.Speech and Signal Process.*

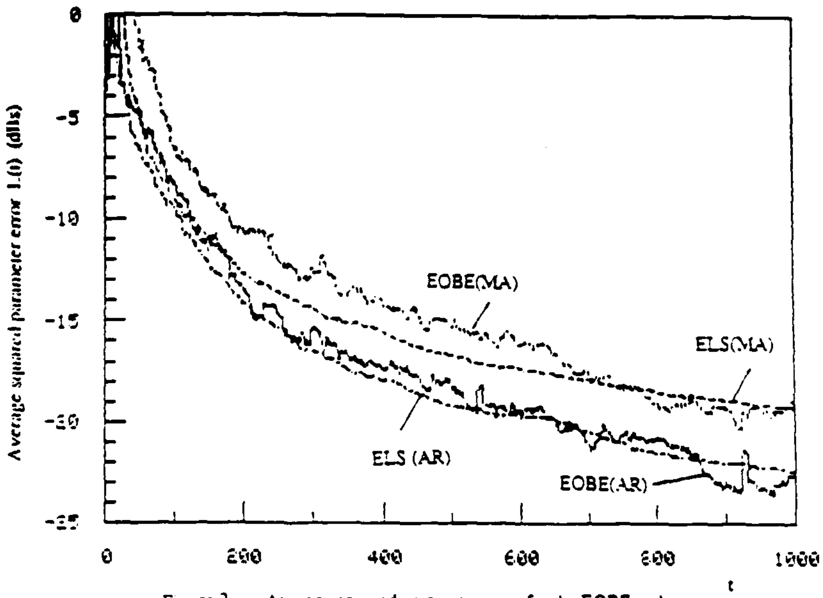


Figure 2. Average squared parameter error for the EOBE and ELS algorithms - Example 1.

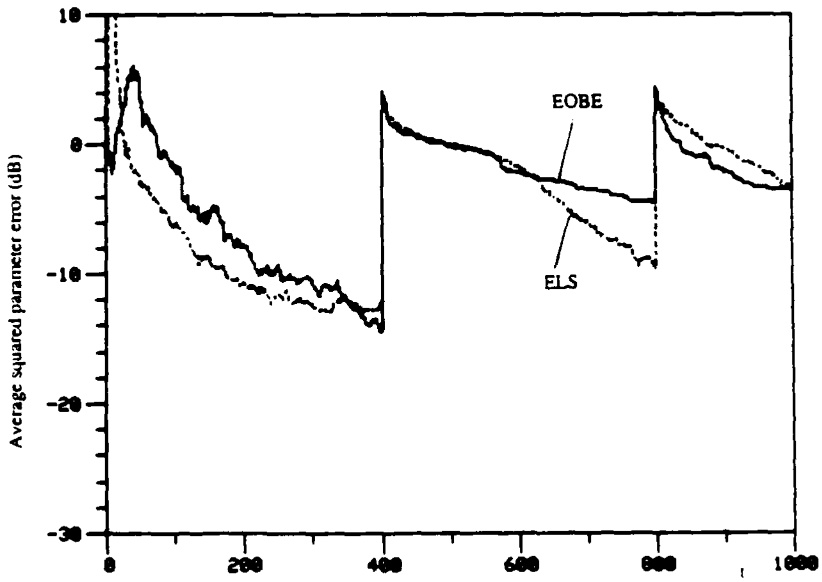


Figure 3. Average squared parameter error for the EOBE and ELS algorithms - Time varying case.

Publication V.C.1.

ARMA PARAMETER ESTIMATION USING A NOVEL RECURSIVE ESTIMATION ALGORITHM WITH SELECTIVE UPDATING ±

Ashok K. Rao*, Yih-Fang Huang* and Soura Dasgupta**

* Department of Electrical & Computer Engineering, University of Notre Dame, Notre Dame, IN 46556. ** Department of Electrical & Computer Engineering, University of Iowa, Iowa City, Iowa 52242.

ABSTRACT

This paper investigates an extension of a recursive estimation algorithm (the so-called OBE algorithm) [9-11], which features a discerning update strategy. In particular, an extension of the algorithm to ARMA parameter estimation is presented here along with convergence analysis. The extension is similar to the extended least-squares algorithm. However, the convergence analysis is complicated due to the discerning update strategy which incorporates an information-dependent updating factor. The virtues of such an update strategy are : (i) More efficient use of the input data in terms of information processing, and (ii) a modular adaptive filter structure which would facilitate the development of a parallel-pipelined signal processing architecture. It is shown in this paper that if the input noise is bounded and if the process is stable, then the a posteriori prediction errors are bounded even without the SPR condition. This is in sharp contrast to the crucial role of the SPR condition in the analysis of the ELS and output error algorithms. With an additional persistence of excitation condition, the parameter estimates are shown to converge to a neighborhood of the true parameters and the *a priori* prediction error and and the parameter misadjustment are shown to be asymptotically bounded. Simulation results show that the parameter estimation error for the EOBE algorithm is comparable to that for the ELS algorithm.

± This work has been supported in part by the National Science Foundation under Grant MIP-87-11174*; in part by the office of Naval Research under Contract N00014-87-k-0284*, and in part by the National Science Foundation under Grant ECS-8618240**.

I INTRODUCTION

In many adaptive signal processing applications such as speech processing, seismic data processing and channel equalization, a signal y(t) is often considered as the output of an IIR filter driven by unknown white noise w(t) [1]. The signal y(t) can therefore be modeled as an autoregressive moving average(ARMA) process of the form

$$y(t) = a_1\, y(t-1)+..+ a_n\, y(t-n) + w(t) + c_1\, w(t-1) +..+ c_r\, w(t-r) \qquad (1)$$

Fitting this ARMA model to the measured data y(t), t =1..T , requires the estimation of the parameters $a_1 ..., a_n, c_1..., c_r$. Many methods for the estimation of ARMA parameters have been proposed in the literature, particularly from the spectral estimation viewpoint. Among the more recent are Cadzow's overdetermined rational equation method [2], the spectral matching technique of Friedlander and Porat [3], and the extended Yule-Walker method of Kaveh [4]. A common feature of these methods is the use of the sample autocorrelation sequence of the output process y(t). In the context of system identification, the extended least-squares (ELS), the recursive maximum likelihood (RML) and multi-stage least-squares algorithms have been used to recursively estimate ARMA parameters [5,6]. The ELS algorithm uses the *a posteriori* prediction error $\varepsilon(t)$, as an estimate of w(t). The regressor vector is formed from y(t-1),..., y(t-n) and $\varepsilon(t-1),..., \varepsilon(t-r)$. The standard recursive least-squares (RLS) algorithm is then employed to update the estimates. The algorithm is conceptually simple but restrictive in the sense that convergence of the algorithm can be assured only if the underlying transfer function $H(q^{-1}) = 1/ C(q^{-1}) - 1/2$ is strictly positive real (SPR), with q^{-1} being the delay operator and

$$C(q^{-1}) = 1 + c_1 q^{-1} + c_2 q^{-2} + .. + c_r q^{-r}$$

The RML algorithm, which uses a filtered version of the regressor vector used in the ELS algorithm, does not require $H(q^{-1})$ to be SPR. However the estimates have to be monitored and projected into a stability region to ensure convergence[5].

In addition to the aforementioned least-squares based methods, there exists a different class of estimation algorithms that estimate membership sets of parameters which are consistent with the measurements and noise constraints [7]-[11]. These algorithms are particularly useful when the noise distribution is unknown but constraints in the form of bounds on the instantaneous values of the noise are available. To the best of our knowledge, none of the algorithms has been applied to the problem of ARMA parameter estimation. Among these algorithms based on membership sets, a group of seminal recursive algorithms are the so-called optimal bounding ellipsoid (OBE) algorithms[9-11]. The OBE algorithms had been developed using a set-theoretic formulation and are applicable to autoregressive with auxiliary input (ARX) models with bounded noise. One of the main features of these temporally recursive algorithms is a discerning update strategy. This feature, obtained by the introduction of an information dependent updating/forgetting factor, yields a modular structure thereby increasing the potential for concurrent and pipelined processing of signals. The presence of such a forgetting factor also gives the algorithms the ability to track slowly time varying parameters. The algorithms have the advantageous feature of automatic asymptotic cessation of updates. If a loose upper bound on the noise magnitude is known, and if the input is persistently exciting and sufficiently uncorrelated with the noise, then it can be shown that the parameter estimates converge asymptotically to a neighborhood of the true parameter vector.

In this paper, we extend one of the OBE algorithms[11] to the ARMA case. For the ARMA parameter estimation problem, the OBE algorithm cannot be applied in its present form. However, by assuming that the input white noise is bounded in magnitude, the OBE algorithm can be extended in a manner similar to the ELS algorithm. Convergence analysis of the resulting algorithm is performed under the assumption that the process is stable and that the noise is bounded. The *a posteriori* prediction error is shown to be bounded without imposing any SPR condition. This is in contrast to the convergence analysis of the ELS or output error algorithms in which the SPR condition is used to prove boundedness of the prediction errors and convergence of parameter estimates[12]. By imposing a persistence of excitation condition on the regressor vector, the *a priori* prediction error of the extended OBE algorithm is shown to be bounded and the parameter estimates are shown to converge to a neighborhood of the true parameter vector.

The paper is organized in the following manner. In Section II, a brief review of the OBE algorithm and its properties is presented. In Section III, the algorithm is extended to ARMA parameter estimation. Convergence analysis of the extended algorithm is performed in Section IV. The performance of the algorithm is compared with the ELS algorithm through simulation studies in Section V. Section VI concludes the paper.

II THE OBE ALGORITHM

Consider the ARX model described by

$$y(t) = a_1\, y(t-1) + .. + a_n\, y(t-n) + b_0 u(t) + b_1\, u(t-1) + .. + b_m\, u(t-m) + v(t)$$

The above equation can be recast as :

$$y(t) = \theta^{*T}\Phi(t) + v(t) \tag{2}$$

where

$$\theta^* = [a_1, a_2 \dots a_n, b_0, b_1, \dots, b_m]^T$$

is the vector of true parameters and

$$\Phi(t) = [y(t-1), y(t-2), .. y(t-n), u(t), u(t-1), .. u(t-m)]^T$$

is the regressor vector. A key assumption here is that the noise is bounded in magnitude, i.e., there exists a $\gamma_0 \geq 0$, such that

$$v^2(t) \leq \gamma_0^2 \qquad \text{for all t, hence}$$

$$(y(t) - \theta^{*T}\Phi(t))^2 < \gamma_0^2$$

Let S_t be a subset of the euclidean space $\Re^{n+m+1}$, defined by

$$S_t = \{ \theta : (y(t) - \theta^T \Phi(t))^2 \leq \gamma_0^2 \ , \theta \in \Re^{n+m+1} \}$$

From a geometric point of view, S_t is a convex polytope in the parameter space and contains the vector of true parameters. The OBE algorithm starts off with a large ellipsoid, E_0 . in $\Re^{n+m+1}$ which contains all possible values of the modelled parameter θ^*. After the first observation $y(1)$ is acquired, an ellipsoid is found which bounds the intersection of E_0 and the convex polytope S_1.This ellipsoid must be optimal in some sense, say minimum volume[9,10] or by any other criterion[9,11], to hasten convergence. Denoting the optimal ellipsoid by E_1, one can proceed exactly as before with the future observations and obtain a sequence of optimal bounding ellipsoids $\{ E_t \}$. The center of the ellipsoid E_t can be taken as the parameter estimate at the t-th instant and is denoted by $\theta(t)$. If at a particular time instant i, the resulting optimal bounding ellipsoid would be of a " smaller size ", thereby implying that the data point $y(i)$ conveys some " information " regarding the parameter estimates, then the parameters are updated. Otherwise E_i is set equal to E_{i-1} and the parameters are not updated. It can also be shown [11] that all the ellipsoids $\{ E_t \}$ contain the true parameter θ^*, provided that E_0 does.

Let the ellipsoid E_{t-1} at the (t-1)-th instant be formulated by

$$E_{t-1} = \{ \theta : (\theta - \theta(t-1))^T P^{-1}(t-1) (\theta - \theta(t-1)) \leq \sigma^2(t-1) \}$$

for some positive definite matrix $P(t-1)$ and a non-negative scalar $\sigma^2(t-1)$. Then, given $y(t)$, an ellipsoid which bounds $E_{t-1} \cap S_t$ "tightly" is

$$\{ \theta : (1-\lambda_t)(\theta - \theta(t-1))^T P^{-1}(t-1)(\theta - \theta(t-1)) + \lambda_t (y(t) - \theta^T\Phi(t))^2 \leq (1-\lambda_t)\sigma^2(t-1) + \lambda_t \gamma_0^2 \} \tag{3}$$

where the forgetting factor $\lambda(t)$ satisfies $0 \leq \lambda(t) < 1$. The size of the bounding ellipsoid is related to the scalar $\sigma^2(t-1)$ and the eigenvalues of $P(t-1)$. The update equations for $\theta(t)$, $P(t)$ and $\sigma^2(t)$ are derived in [11]. The optimal ellipsoid which bounds the intersection of E_{t-1} and S_t is defined in terms of an optimal value of λ_t. For the OBE algorithm of [11], the optimum value λ^*_t is determined by minimization of $\sigma^2(t)$ with respect to λ_t at every time instant. The minimization procedure results in λ^*_t being set equal to zero (no update) if

$$\sigma^2(t) + \delta^2(t) \leq \gamma^2(t) \tag{4}$$

If (4) is not satisfied, then the optimal value of λ_t is computed. The parameter estimation procedure is depicted in Fig. 1. An outgrowth of the modular recursive estimation procedure is a parallel-pipelined networking structure [13]. The algorithm is such that the computational complexity of the information evaluation (IE) procedure is much less than that of the updating procedure (UPD). Since, in general, a good number of data samples would be rejected by the IE, both the IE and the UPD would involve significant amounts of idle time. A viable scheme then is to configure a parallel-pipelined network comprising of such modular estimators to process signals from multiple channels. Apart from reducing hardware costs, such a scheme would offer increased reliability since the failure of one UPD processor would not cause any of the channels to fail, in contrast to a system with a dedicated UPD processor for each channel.

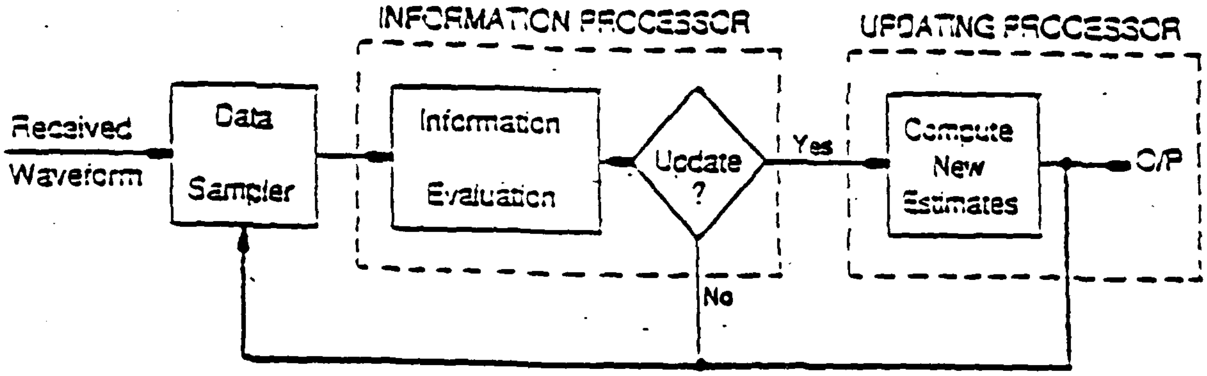


Figure 1 Modular recursive parameter estimator

III EXTENSION TO ARMA MODELS

The ARMA model described by (1), can be rewritten as

$$w(t) = y(t) - \theta^{*T}\Phi'(t) \tag{5}$$

where θ^* is the vector of true parameters and is now defined by

$\theta^* = [a_1 , a_2 \ ... \ a_n , c_1 , c_2 \ ..., \ c_r \ ]^T$

$\Phi'(t) = [\ y(t\text{-}1) , ..., y(t\text{-}n) , w(t\text{-}1) , ..., w(t\text{-}r) \ ]^T$

Here again, $w(t)$ is assumed to be bounded in magnitude by γ_0. Since the values of the noise sequence $\{ w(t) \}$ are not available, the regressor vector $\Phi'(t)$ is not known exactly. If, however, at time t, an estimate of θ^*,

$$\theta(t) = [\ a_1(t), ..., a_n(t) \ \ c_1(t), ..., c_r(t) \ ]^T \tag{6}$$

is available, $w(t)$ could be estimated by

$$\varepsilon(t) = y(t) - \theta^T(t)\Phi(t)$$

where

$$\Phi(t) = [\ y(t\text{-}1) , .. \ y(t\text{-}n) , \varepsilon(t\text{-}1) , \ ... \ \varepsilon(t\text{-}r) \ ]^T \tag{7}$$

It follows from (5) and the boundedness of $| \ w(t) \ |$ that

$$(\ y(t) - \theta^{*T}\Phi'(t) \ )^2 \leq \gamma_0^2$$

Hence if $\varepsilon(t) = w(t)$, then $\Phi'(t) = \Phi(t)$ and for a suitable $\gamma \geq 0$, such that $\gamma^2 > \gamma_0^2$

$$(\ y(t) - \theta^{*T}\Phi(t) \ )^2 \leq \gamma^2 \tag{8}$$

Thus $\theta^* \in S_t$, where S_t is defined by

$$S_t = \{ \ \theta : \ (y(t) - \theta^T \Phi(t) \ )^2 \ \} \leq \gamma^2 , \theta \in R^{n+r} \ \}$$

In essence, if the difference between $\varepsilon(t)$ and $w(t)$ is small, applying the OBE algorithm will yield a sequence of bounding ellipsoids $\{E_t\}$ in the parameter space. If (8) holds for all t and $\theta^* \in E_0$, then it is easy to see that $\theta^* \in E_t$ for all time instants t. The optimal bounding ellipsoid E_t is described by

$$E_t = \{ \ \theta \in R^{n+r} : \ (\ \theta - \theta(t) \ )^T \ P^{-1}(t) \ (\ \theta - \theta(t) \ ) \ \leq \ \sigma^2(t) \ \} \tag{9}$$

and the update equations which follow directly from [11] are

$$P^{-1}(t) = (1-\lambda_t) \ P^{-1}(t\text{-}1) + \lambda_t \Phi(t) \ \Phi^T(t) \tag{10 a}$$

$$\theta(t) = \theta(t\text{-}1) + \lambda_t P(t)\Phi(t) \ \delta(t) \tag{10b}$$

$$\delta(t) = y(t) - \theta^T(t\text{-}1) \ \Phi(t) \tag{10c}$$

$$\sigma^2(t) = (1-\lambda_t)\,\sigma^2(t-1) + \lambda_t\gamma^2 - \frac{\lambda_t(1-\lambda_t)\,\delta^2(t)}{1-\lambda_t+\lambda_t G(t)} \tag{10d}$$

where

$$G(t) = \Phi^T(t)\,P(t-1)\,\Phi(t) \tag{10e}$$

The above recursive relations along with the initial values

$$P^{-1}(0) = I\,,\ \theta(0) = 0 \text{ and } \sigma^2(0) = 1/\varepsilon \text{ .with } \varepsilon << 1$$

form the basis of the extended optimal bounding ellipsoid (EOBE) estimation algorithm[14]. The matrix inversion lemma can be used to obtain the recursion for P(t)

$$P(t) = \frac{1}{1-\lambda_t}\left[P(t-1) - \frac{\lambda_t P(t-1)\Phi(t)\Phi^T(t)\,P(t-1)}{1-\lambda_t+\lambda_t G(t)} \right] \tag{10f}$$

It is easily shown from (10d) that for the EOBE algorithm, minimizing $\sigma^2(t)$ with respect to λ_t at every time instant yields the same updating criterion and the same algorithm for determining the optimum value of the forgetting / updating factor λ^*_t, as in [11]. In particular,

$$\text{If } \sigma^2(t) + \delta^2(t) \le \gamma^2(t) \text{ then } \lambda^*_t = 0, \tag{11}$$

otherwise

$$\lambda^*_t = \min(\alpha, v_t) \tag{12}$$

where

$$v_t = \begin{cases} \alpha & \text{if } \delta^2(t) = 0 \quad (12a) \\ \dfrac{1-\beta(t)}{2} & \text{if } G(t) = 1 \quad (12b) \\ \dfrac{1}{1-G(t)}\left[1-\sqrt{\dfrac{G(t)}{1+\beta(t)\,(G(t)-1)}}\,\right] & \text{if } \beta(t)\,(G(t)-1)+1 > 0 \quad (12c) \\ \alpha & \text{if } \beta(t)\,(G(t)-1)+1 \le 0 \quad (12d) \end{cases}$$

and

$$\beta(t) \triangleq \frac{\gamma^2 - \sigma^2(t-1)}{\delta^2(t)} \tag{12e}$$

The algorithm thus retains the discerning update strategy and the modular adaptive filter structure[11,13].

Choice of γ^2

Finding a value of γ^2 to ensure that (8) holds for all time instants may not be easy. Instead, we will show in the next section that choosing γ^2 to be an upper bound on the square of the magnitude of the output y(t) will ensure that the parameter estimates obtained by applying the EOBE algorithm converge to a neighborhood of the true parameters.

It is easy to see that if $|w(t)|$ is bounded so is $|y(t)|$, as the output y(t) is related to the input according to

$$Y(z) = \frac{C(z^{-1})}{A(z^{-1})} W(z)$$

where Y(z) and W(z) are the z-transforms of y(t) and w(t), respectively, and

$$A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2} + .. + a_n z^{-n},$$
$$C(z^{-1}) = 1 + c_1 z^{-1} + c_2 z^{-2} + .. + c_r z^{-r}$$

The output process y(t) can thus be expressed as

$$y(t) = h(t) * w(t),$$

where * denotes discrete convolution and h(t) is the impulse response of the filter $C(z^{-1}) / A(z^{-1})$. Assume that the process is stable, then there exists a finite K such that

$$\sum_{i=0}^{\infty} |h(i)| \le K < \infty$$

and if $|w(t)| \le \gamma_0$ then

$$|y(t)| \le \sum_{i=0}^{\infty} |h(i)|\,|w(t-i)| \le K\gamma_0 = \gamma \qquad (13)$$

In order to obtain a value for γ, the user would therefore need the bound on $|w(t)|$ and an estimate of K. Aternatively, the outputs could be monitored and a bound on $|y(t)|$ could be obtained before starting the actual parameter estimation procedure. A loose upper bound on either K or the magnitude of the output would suffice since simulations have shown that using values for γ that are several times larger or smaller than the actual bound on $|y(t)|$ has no deleterious effects on the quality of estimates or convergence rate.

IV CONVERGENCE ANALYSIS

We will first show that if the updating factor sequence is that which minimizes σ^2 at every instant (denoted by $\{ \lambda^*_t \}$) then the parameter estimates converge. To show that, the following lemma will be needed.

Lemma 1. If the magnitude of the noise w(t) is upper bounded and the process is stable then $\sigma^2(t)$, defined in (10d), is always non-negative provided that the initial values $\theta(0)$, $P^{-1}(0)$ and $\sigma^2(0)$ are chosen such that $\theta^T(0)P^{-1}(0)\theta(0) \le \sigma^2(0)$

Proof. From (10a), (10b) and (10c)

$$\theta^T(t)\, P^{-1}(t)\, \theta(t) = (1-\lambda_t)\, \theta^T(t-1)\, P^{-1}(t-1)\, \theta(t-1) + \lambda_t [\, y^2(t) - \delta^2(t)\, (1-\lambda_t \Phi^T(t)\, P(t)\, \Phi(t)\,)\,]$$

$$= (1-\lambda_1)(1-\lambda_2) \ldots (1-\lambda_t)\, \theta^T(0)\, P^{-1}(0)\, \theta(0)$$

$$+ \sum_{i=1}^{t} q_{it} [\, y^2(i) - \delta^2(i)\, (1-\lambda_i \Phi^T(i)\, P(i)\, \Phi(i)\,)\,]$$

where

$$q_{it} = \lambda_i (1-\lambda_{i+1}) \ldots (1-\lambda_t) \ \ge\ 0 \quad \forall\, i, t.$$

Since the process is stable and $\{w(t)\}$ is bounded, there exists a γ^2 such that $y^2(t) \le \gamma^2$
The positive semi-definiteness of $P^{-1}(t)$ will therefore imply that

$$\prod_{i=1}^{t}(1-\lambda_i)\ \ \theta^T(0)P^{-1}(0)\theta(0) + \sum_{i=1}^{t} q_{it} [\, \gamma^2 - \delta^2(i)\, (1-\lambda_i \Phi^T(i)\, P(i)\, \Phi(i)\,)\,] \ \ge\ 0 \tag{14}$$

But from (10f)

$$\lambda_i \Phi^T(i)\, P(i)\, \Phi(i) = \frac{\lambda_i G(i)}{1-\lambda_i + \lambda_i G(i)} \tag{15}$$

Using (15) in (14) yields

$$\prod_{i=1}^{t}(1-\lambda_i)\ \ \theta^T(0)P^{-1}(0)\theta(0) + \sum_{i=1}^{t} q_{it} [\, \gamma^2 - \delta^2(i) \frac{1-\lambda_i}{1-\lambda_i + \lambda_i G(i)}\,] \ \ge\ 0 \tag{16}$$

From (10d), a non-recursive expression for $\sigma^2(t)$ can be obtained as

$$\sigma^2(t) = (1-\lambda_1)(1-\lambda_2) \ldots (1-\lambda_t)\, \sigma^2(0) + \sum_{i=1}^{t} q_{it} [\, \gamma^2 - \frac{\delta^2(i)\, (1-\lambda_i)}{1-\lambda_i + \lambda_i G(i)}\,] \tag{17}$$

Since $\sigma^2(0) \geq \theta^T(0)P^{-1}(0)\theta(0)$, (17) implies that $\sigma^2(t) \geq 0$ for all t. This result will hold for any sequence of forgetting factors $\{\lambda_t\}$ with $0 \leq \lambda_t < 1$.

Remark: The fact that $\sigma^2(t)$ is non-negative and non-increasing will play a crucial role in the subsequent convergence analysis. It also guarantees that the ellipsoids are non singular (non empty) sets.

Theorem 1. If the assumptions of Lemma 1 are satisfied then

$$1. \quad \lim_{t\to\infty} \varepsilon^2(t) \in [0, \gamma^2] \tag{18}$$

$$2. \quad \lim_{t\to\infty} \sigma^2(t) \in [0, \gamma^2] \tag{19}$$

Note that throughout this section, expressions like (18) should not be taken to mean that $\lim_{t\to\infty}\varepsilon^2(t)$ exists necessarily, but rather that $\varepsilon^2(t)$ becomes asymptotically less than or equal to γ^2.

Proof. It is easily shown from (10b), (10c) and (15) that the *a posteriori* and *a priori* prediction errors are related by

$$\varepsilon(t) = \frac{1-\lambda_t}{1-\lambda_t+\lambda_t G(t)}\,\delta(t) \tag{20}$$

Note that the non-negativeness of $G(t)$ implies that $\varepsilon^2(t) \leq \delta^2(t)$. Substituting (20) in (10d) and using the fact that $0 \leq \lambda_t \leq \alpha < 1$, yields

$$\sigma^2(t) \leq \sigma^2(t-1) + \lambda_t\gamma^2 - \lambda_t\left(\frac{1-\lambda_t+\lambda_t G(t)}{1-\lambda_t}\right)\varepsilon^2(t) \tag{21}$$

Since $G(t)$ is non-negative

$$\sigma^2(t) \leq \sigma^2(t-1) + \lambda_t[\gamma^2 - \varepsilon^2(t)]$$

or

$$\lambda_t[\gamma^2 - \varepsilon^2(t)] \geq \sigma^2(t) - \sigma^2(t-1)$$

The right hand side tends to zero since $\sigma^2(t)$ is a non negative, non increasing sequence. Thus

$$\lim_{t\to\infty} \lambda_t[\gamma^2 - \varepsilon^2(t)] \in [0, \gamma^2] \tag{22}$$

This the case if and only if either

$$1. \quad \lim_{t\to\infty} \varepsilon^2(t) \in [0, \gamma^2] \quad \text{in which case (18) is established.} \tag{23a}$$

or

2. $\lim_{t\to\infty} \lambda_t = 0$ and $\lim_{t\to\infty} \lambda_t \varepsilon^2(t) = 0$ (23b)

Let the forgetting factor which minimizes $\sigma^2(t)$ at every instant t be denoted by λ^*_t, then

$$\text{if } \lambda^*_t > 0, \quad \text{then} \quad \left.\frac{d\,\sigma^2(t)}{d\,\lambda_t}\right|_{\lambda_t = \lambda^*_t} \leq 0 \tag{24}$$

Taking derivatives with respect to λ_t in (10d), and using (24) yields

$$\sigma^2(t) \leq \sigma^2(t\text{-}1) - \frac{\lambda^{*2}_t \delta^2(t)\, G(t)}{(1 - \lambda^*_t + \lambda^*_t G(t))^2} \tag{25}$$

The non-negativity of $\sigma^2(t)$ therefore implies

$$\sum_{i=1}^{t} \frac{\lambda^{*2}_i \delta^2(i)\, G(i)}{(1 - \lambda^*_i + \lambda^*_i G(i))^2} \leq \sigma^2(0) - \sigma^2(t) < \infty \tag{26}$$

Hence

$$\lim_{t\to\infty} \frac{\lambda^{*2}_t \delta^2(t)\, G(t)}{(1 - \lambda^*_t + \lambda^*_t G(t))^2} = 0 \tag{27}$$

Denote the optimal forgetting factor by λ_t for ease of notation. Substituting (20) in (27) yields

$$\lambda^2_t G(t)\, \varepsilon^2(t) \to 0 \tag{28}$$

Thus if (23b) holds, then for all $\Delta > 0$, there exists $N_1 > 0$ such that for all $t > N_1$,

$$\lambda_t < \Delta \tag{29a}$$

$$\lambda_t \varepsilon^2(t) < \Delta \tag{29b}$$

$$\lambda^2_t G(t)\ \varepsilon^2(t) < \Delta \tag{29c}$$

Consider the four cases of (12) applicable to this situation

Case 1. $\delta^2(t) = 0$. Then $\varepsilon^2(t) \leq \delta^2(t) \leq \gamma^2$.

Case 2. $G(t) = 1$, $\lambda_t = (1-\beta(t))/2 < \Delta$. Hence $\beta(t) > 1 - 2\Delta$. It then follows from the definition of $\beta(t)$ that $\varepsilon^2(t) \leq \gamma^2$.

Case 3. $\beta(t)\,(G(t) - 1) + 1 \leq 0$. If $\Delta < \alpha$, then (11) has to hold and so $\varepsilon^2(t) \leq \gamma^2$.

Case 4. $\beta(t)\,(G(t) - 1) + 1 > 0$. Define

$$\beta'(t) \triangleq \frac{\gamma^2 - \sigma^2(t\text{-}1)}{\varepsilon^2(t)} \tag{30}$$

Then

$$\beta'(t) = \beta(t)\left[\frac{1-\lambda_t+\lambda_t G(t)}{1-\lambda_t}\right]^2 \tag{31}$$

For this case the following relation between λ_t and $\beta(t)$ can be derived from (12c), exactly as in [11].

$$\beta(t) = \frac{1-\lambda_t^2(G(t)-1) - 2\lambda_t}{[1-\lambda_t + \lambda_t G(t)\ ]^2} \tag{32}$$

Substituting (32) into (31) and multiplying by $\varepsilon^2(t)$ yields

$$\gamma^2 - \sigma^2(t-1) = \varepsilon^2(t)\,\beta'(t) = \varepsilon^2(t) - \frac{\lambda_t^2 G(t)\,\varepsilon^2(t)}{(1-\lambda_t)^2} \tag{33}$$

Substituting (29) into (33) gives

$$\gamma^2 - \sigma^2(t-1) \ \geq\ \varepsilon^2(t) - \frac{\Delta}{(1-\lambda_t)^2}$$

$$\geq\ \varepsilon^2(t) - \frac{\Delta}{(1-\Delta)^2}$$

Thus

$$\varepsilon^2(t) + \sigma^2(t-1) \leq \gamma^2 + O(\Delta)$$

and since Δ can be arbitrarily small, (18) is satisfied. Furthermore, from (10d)

$$\sigma^2(t) - \gamma^2 \leq (1-\lambda_t)(\sigma^2(t-1) - \gamma^2)$$

And from (11), $\lambda_t > 0$ if $\sigma^2(t) > \gamma^2$. This, together with Lemma 1, would imply that

$$\lim_{t\to\infty} \sigma^2(t) \in [0, \gamma^2]$$

Hence (19) is also satisfied.

Remark. Boundedness of $\varepsilon^2(t)$, the *a posteriori* prediction error, has been shown without imposing any persistence of excitation conditions on the regressor vector . Furthermore, in contrast to the ELS algorithm, the SPR condition is not required.

Boundedness of $\delta^2(t)$, the *a priori* prediction error, and convergence of the parameter estimates to a neighborhood of the true parameter can be assured, by requiring the regressor vector to be persistently exciting. The next lemma relates the positive definiteness of $P^{-1}(t)$ to the richness of the regressor vector $\Phi(t)$.

Lemma 2. If there exist positive α_3 and N such that for all t

$$\sum_{i=t}^{t+N} \Phi(i)\,\Phi^T(i) \;\geq\; \alpha_3 I \;>\; 0 \tag{34}$$

then there exists a positive α_4 such that

$$P^{-1}(t) \geq \alpha_4 I > 0$$

Proof of the lemma is the same as that of Theorem 4.1 of [11], it is thus omitted here.

Remark. The positive definiteness of $P^{-1}(t)$ implies that the eigenvalues of P(t) are upper bounded.

Theorem 2. If assumptions of Lemma 1 are satisfied and (34) holds then the EOBE algorithm ensures :

(a) Parameter difference convergence

$$\lim_{t\to\infty} \| \theta(t) - \theta(t-k) \| = 0 \tag{35}$$

for any finite k.

(b) Bounded *a priori* prediction errors

$$\lim_{t\to\infty} \delta^2(t) \in [\,0\,,\gamma^2\,] \tag{36}$$

(c) Bounded parameter misadjustment

$$\lim_{t\to\infty} \| \theta(t) - \theta^* \|^2 \;\leq\; M\gamma^2 \;<\; \infty \tag{37}$$

for some finite M.

Proof.

(a) From (10b) and (10f)

$$\| \theta(t) - \theta(t-1) \|^2 = \frac{\lambda_t{}^2 \Phi^T(t)\, P^2(t-1)\, \Phi(t)\, \delta^2(t)}{(\,1-\lambda_t + \lambda_t\, G(t)\,)^2}$$

$$\leq e_{max}\{\, P(t-1)\,\}\, \frac{\lambda_t{}^2 G(t)\, \delta^2(t)}{(1-\lambda_t + \lambda_t\, G(t)\,)^2} \tag{38}$$

If (34) holds then by Lemma 2, e_{max} { P(t-1) }, the maximum eigenvalue of P(t-1), is bounded for all t , and hence by (27)

$$\| \theta(t) - \theta(t-1) \| \;\rightarrow 0 \tag{39}$$

Applying the Minkowski inequality to $\| \theta(t) - \theta(t-k) \|$ and using (39) completes the proof of (35).

(b) From (10e) and (13) it follows that

$$G(t) \leq e_{max}\{P(t-1)\}[\,n\gamma^2 + r \max_{t-r \leq i \leq t-1} \varepsilon^2(i)\,] \tag{40}$$

where n is the order of the AR process and r is the order of the MA process. If (34) holds then $e_{max}\{P(t-1)\}$, the maximum eigenvalue of P(t-1), is bounded. Since $\{\varepsilon(t)\}$ is bounded by Theorem 1, $\{G(t)\}$ is bounded. Then it can be shown (see Theorem 3.2 of [11]), that the *a priori* prediction errors satisfy (36).

(c) From (18) we conclude that there exists N_1 such that for all $t > N_1$, $|\varepsilon(t)| \leq \gamma$.
Define

$$x_1(t) = [\,y(t-1), y(t-2), ..., y(t-n)\,]^T,$$
$$x_2(t) = [\,\varepsilon(t-1), \varepsilon(t-2), ..., \varepsilon(t-r)\,]^T,$$
$$x^0_2(t) = [\,w(t-1), w(t-2), ..., w(t-r)\,]^T,$$

Denote the actual and the estimated AR parameters by a and a(t), respectively, and the actual and the estimated MA parameters by c and c(t), respectively. Thus

$$|\,y(t) - a^T(t)\,x_1(t) - c^T(t)\,x_2(t)\,| \leq \gamma \tag{41}$$

Substituting (1) in (41) yields

$$|\tilde{a}^T(t)x_1(t) + \tilde{c}^T(t)\,x_2(t) + w(t) + c^T x_2^0(t) - c^T x_2(t)| \leq \gamma \tag{42}$$

where

$$\tilde{a}(t) = a(t) - a \text{ and } \tilde{c}(t) = c(t) - c.$$

Hence

$$|\tilde{a}^T(t)x_1(t) + \tilde{c}^T(t)\,x_2(t)| \leq \gamma + |w(t)| + |c^T x_2^0(t)| + |c^T x_2(t)| \tag{43}$$

From (18), and the boundedness of $\{w(t)\}$, it follows that the right hand side of (43) is of $O(\gamma)$. Since $\Phi^T(t) = [\,x_1^T(t), x_2^T(t)\,]^T$, it follows that

$$|\tilde{\theta}^T(t)\,\Phi(t)| \leq O(\gamma)$$

where

$$\tilde{\theta}^T(t) = [\,\tilde{a}^T(t), \tilde{c}^T(t)\,]^T$$

Hence

$$|\tilde{\theta}^T(t)\,\Phi(t)\,\Phi^T(t)\,\tilde{\theta}(t)| \leq O(\gamma^2)$$

So

$$\sum_{l=t}^{t+N} \tilde{\theta}^T(l)\,\Phi(l)\,\Phi^T(l)\,\tilde{\theta}(l) \leq O(\gamma^2) \tag{44}$$

From (35), for all $\varepsilon > 0$, there exists N' such that for all $t > N'$

$$\| \tilde{\theta}(t) - \tilde{\theta}(t+1) \| = \| \theta(t) - \theta(t+1) \| < \varepsilon .$$

Hence for any finite k

$$\| \tilde{\theta}(t) - \tilde{\theta}(t+k) \| = \| \tilde{\theta}(t) - \tilde{\theta}(t+1) + \tilde{\theta}(t+1) - \tilde{\theta}(t+2) + .. + \tilde{\theta}(t+k-1) - \tilde{\theta}(t+k) \|$$

Applying The Minkowski inequality yields

$$\| \tilde{\theta}(t) - \tilde{\theta}(t+k) \| \le [\| \tilde{\theta}(t) - \tilde{\theta}(t+1) \| + \| \tilde{\theta}(t+1) - \tilde{\theta}(t+2) \| + .. + \| \tilde{\theta}(t+k-1) - \tilde{\theta}(t+k) \|]$$
$$\le O(\varepsilon) \qquad (45)$$

Using Lemma A-1, (see appendix), yields for $t > \max (N_1 , N')$

$$\tilde{\theta}^T(t) [\sum_{l=t}^{t+N} \Phi(l) \Phi^T(l)] \tilde{\theta}(t) \le O(\gamma^2) + O(\varepsilon^2)$$

And substituting (34) yields

$$\lim_{t \to \infty} \tilde{\theta}^T(t) \tilde{\theta}(t) \le O(\gamma^2)$$

and hence (37) follows.

Remark: To summarize, the covergence analysis has shown that if the process is stable and if the driving noise is bounded then the *a posteriori* prediction errors are bounded. In addition if a sufficient richness condition is imposed on the regressor vector, then the *a priori* prediction errors are bounded and the parameter estimates are asymptotically contained within a neighborhood of the true parameters.

V SIMULATION RESULTS

Simulations have been performed to investigate the performance of the EOBE algorithm *vis a vis* the ELS algorithm. In this paper, we present simulation results for two examples- a broad band ARMA (3,3) process and a narrow band ARMA(2,2) process where the indices p, q in an ARMA(p,q) process refer to the orders of the $A(q^{-1})$ and $C(q^{-1})$ polynomials, respectively.

Example 1. Broad band ARMA (3,3) process

The output data $\{y(t)\}$ is generated by the following difference equation

$$y(t) = -0.4\, y(t-1) + 0.2\, y(t-2) + 0.6\, y(t-3) + w(t) - 0.6\, w(t-1) + 0.2 w(t-2) + 0.4\, w(t-3)$$

The noise sequence $\{w(t)\}$ is generated by a pseudo-random number generator with a uniform probability distribution in [-1.0, 1.0]. The upper bound γ^2 was set equal to 25.0. The parameter estimates were obtained by applying the EOBE algorithm to 1000 point data sequences. One hundred runs of the algorithm were performed on the same model but with different input noise sequences. The average squared parameter error L(t), is computed for the AR coefficients according to the formula

$$L(t) = \frac{1}{100} \sum_{j=1}^{100} l_j(t)$$

where $l_j(t)$, the squared AR parameter error at time t for the j'th run, is defined by

$$l_j(t) = \sum_{i=1}^{n} (a_i(t) - a_i)^2$$

with a_i and $a_i(t)$ being defined by (1) and (6), respectively. The average squared parameter error for the MA coefficients is defined analogously. Figure 2 displays the average squared estimation errors for AR and MA parameters using both the EOBE and the ELS algorithms. The curves show that the performance of the two algorithms is comparable. It may be noted that the AR parameter estimates have markedly lower variance (about the true parameters) than the MA parameters. The average number of updates for the EOBE algorithm was 139 for 1000 point data sequences. Thus less than 15% of the samples are used for updates, as compared to the ELS algorithm which updates at every sampling instant.

The effect of different choices for the upper bound γ^2 on the performance has also been studied. For each value of γ^2, the asymptotic average squared parameter error T, was computed over 25 runs of the algorithm, according to the formula

$$T = \frac{1}{25} \sum_{j=1}^{25} \| \theta_j(1000) - \theta^* \|^2$$

where $\theta_j(1000)$ is the parameter estimate at the 1000'th iteration, in the j'th run. The second column of Table I lists the different values of T obtained when γ^2 is varied from 0.1 to 200. It is clear that the algorithm is insensitive to the value of γ^2, since both the tap error and the average number of updates are almost constant. It was seen that for low values of γ^2, even though the true parameter may fall outside the bounding ellipsoid, it is captured in subsequent iterations.

The performance of the algorithm, when the noise sequence $\{w(t)\}$ has gaussian distribution, was evaluated in a similar fashion. A constant value of $\gamma^2 = 25$ was used and the standard deviation of the noise was varied. The results for 25 runs of the algorithm are shown in Table II. It is clear that even though the unbounded noise does cause the output to exceed the bound, the effect on the parameter estimates is marginal.

Finally, the tracking capability of the EOBE algorithm(with α=0.2), was compared with that of the ELS algorithm (with forgetting factor=0.99). The same model was used to generate 400 data points. The parameters were then changed by 150% and the next 400 points were generated. Finally the last 200 points were generated by using the original parameters. The average squared parameter error was evaluated over 25 runs and is shown in Figure 3. Even though the formulation of bounding ellipsoids is based on the assumption that the parameters are constant, the simulation results show that the algorithm is able to accomodate changes in model parameters. Analysis of the tracking ability of the algorithm is currently under investigation.

Example 2. Narrow band ARMA (2,2) process

The output data $\{y(t)\}$ is generated by the following difference equation

$$y(t) = 1.4\, y(t-1) - 0.95\, y(t-2) + w(t) - 0.86\, w(t-1) + 0.431\, w(t-2)$$

The noise sequence is uniformly distributed in [-1.0,1.0], as in the first example. The upper bound γ^2 was set equal to 20.0. The average squared parameter errors are calculated over 100 runs and plotted in Fig.4. The average number of updates was 85 for 1000 point data sequences.

For this example too, different values of the upper bound γ^2 were used and no significant difference in the quality of estimates, number of updates or convergence rate was observed. Thus, it is verified once again that a precise knowledge of the upper bound is not a prerequisite for satisfactory performance of the algorithm.

Example 3. Non SPR ARMA(3,3) process

The output data $\{y(t)\}$ is generated by the following difference equation

$$y(t) = -0.6\, y(t-1) - 0.58\, y(t-2) - 0.464\, y(t-3) + w(t) + 0.2\, w(t-1) + 0.6 w(t-2) + 0.2\, w(t-3)$$

The noise sequence is generated as in the first example. The upper bound γ^2 was set equal to 3.25. The maximum values (taken over 25 runs of the algorithm), of the squared residual errors, at each iteration are displayed in Fig. 5. Note that the squared *a posteriori* prediction errors are well within the upper bound γ^2 even though the SPR condition is violated.

VI CONCLUSION

A recursive parameter estimation algorithm has been extended for ARMA parameter estimation. The main features of the algorithm are a membership set theoretic formulation and a discerning update strategy. Convergence analysis of the algorithm has been performed under the assumptions that the process is stable and that the noise is bounded. The main results of the analysis are that the algorithm yields bounded *a posteriori* prediction errors without SPR or persistence of excitation type condition. With a persistence of excitation condition on the regressor vector, boundedness of the *a priori* prediction error can then be established and the parameter estimates are shown to converge to a neighborhood of the true parameters. Simulation results show that the performance of the algorithm is comparable to the ELS algorithm while requiring far fewer updates.

REFERENCES

[1] B. Friedlander, " System identification techniques for adaptive signal processing, "*IEEE Trans. Acoust. Speech and Signal Process.*, vol. ASSP-30, No. 2, pp. 240-246, April 1982.

[2] J. A. Cadzow, " Spectral estimation : an overdetermined rational model equation approach," *Proc. IEEE*, vol. 70, No. 9, pp. 907-939, Sept 1982.

[3] B. Friedlander and B. Porat, "A spectral matching technique for ARMA parameter estimation, " *IEEE Trans. Acoust. Speech and Signal Process.*, vol. ASSP-32, No. 2, pp. 338-343, April 1984.

[4] M. Kaveh, " High resolution spectral estimation for noisy signals," *IEEE Trans. Acoust. Speech and Signal Process.*, vol. ASSP-27, No. 3, pp. 286-287, June 1979.

[5] L. Ljung and T. Soderstrom,*Theory and Practice of Recursive Identification*, Cambridge, Mass: MIT Press, 1983.

[6] D. Q. Mayne *et al*, " Linear identification of ARMA processes," *Automatica* vol. 18, No. 4, pp. 461-466, July 1982.

[7] F. C. Schweppe, " Recursive state estimation : unknown but bounded errors and system inputs, " *IEEE Trans. Automatic Control*, vol. AC-13, No.1, pp. 22-28, Feb. 1968.

[8] M. Milanese and G. Belaforte, " Estimation theory and uncertainty intervals," *IEEE Trans. Automatic Control*, vol. AC-27, No. 2, pp. 408-414, April 1982.

[9] E. Fogel and Y.F. Huang, " On the value of information in system identification - bounded noise case," *Automatica*, vol. 18, No. 2, pp. 229-238, March 1982.

[10] Y. F. Huang, " A recursive estimation algorithm using selective updating for spectral analysis and adaptive signal processing,"*IEEE Trans. Acoust.Speech and Signal Process.*, vol.

ASSP-34, No. 5, pp. 1131-34, October 1986.

[11] S. Dasgupta and Y.F. Huang, " Asymptotically convergent modified recursive least-squares," *IEEE Trans. Information Theory*, vol. IT-33, No.3, pp. 383-392, May 1987.

[12] G. C. Goodwin and K.S. Sin, *Adaptive Filtering, Prediction and Control*, Englewood Cliffs, NJ: Prentice-Hall, 1982.

[13] Y. F. Huang and R. W. Liu, " A high level parallel-pipelined network architecture for adaptive signal processing, " in *Proc. 26th IEEE Conf. on Decision and Control*, vol.1, pp. 662-666, Dec. 1987.

[14] A. K. Rao, " Applications and extensions of a novel recursive estimation algorithm with selective updating," M.S. Thesis, Dept. of Electrical & Computer Engineering, University of Notre Dame, Notre Dame, IN. Aug. 1987.

APPENDIX

Lemma A-1 For the EOBE algorithm of (10), if for any $\varepsilon > 0$, there exist positive N, N_1, k and γ^2 such that for $t > N_1$

(i) $\| \tilde{\theta}(t) - \tilde{\theta}(t+k) \| < O(\varepsilon)$ (A-1)

(ii) $|e(t)| \le \gamma,\ |y(t)| \le \gamma$ (A-2)

(iii) $$\sum_{l=t}^{t+N} \tilde{\theta}^T(l)\,\Phi(l)\,\Phi^T(l)\,\tilde{\theta}(l) \le O(\gamma^2) \qquad \text{(A-3)}$$

Then

$$\tilde{\theta}^T(t)\Big[\sum_{l=t}^{t+N} \Phi(l)\,\Phi^T(l)\Big]\tilde{\theta}(t) \le O(\gamma^2) + O(\varepsilon^2) \qquad \text{(A-4)}$$

Proof. Define

$$[\, d_1(t), d_2(t), \dots, d_{n+r}(t) \,]^T = \tilde{\theta}(t)$$

$$[\, r_1(t), r_2(t), \dots, r_{n+r}(t) \,]^T = \Phi(t)$$

Then

$$\sum_{l=t}^{t+N} \tilde{\theta}^T(l)\,\Phi(l)\,\Phi^T(l)\,\tilde{\theta}(l) = \sum_{l=t}^{t+N} \Big\{ \sum_{i=1}^{n+r} d_i(l)\, r_i(l) \Big\}^2 \qquad \text{(A-5)}$$

For any $l \in [\, t, t+N \,]$, we have

$$\sum_{i=1}^{n+r} d_i(t)\, r_i(l) \le \sum_{i=1}^{n+r} d_i(l)\, r_i(l) + \sum_{i=1}^{n+r} | d_i(t) - d_i(l) |\, | r_i(l) | \qquad \text{(A-6)}$$

By assumption (A-2), $|r_1(l)| \le \gamma$. Hence applying the Schwartz inequality to the last term of (A-6) yields

$$\sum_{i=1}^{n+r} d_i(t)\, r_i(l) \le \sum_{i=1}^{n+r} d_i(l)\, r_i(l) + \gamma (n+r)^{\frac{1}{2}} \Big\{ \sum_{i=1}^{n+r} | d_i(t) - d_i(l) |^2 \Big\}^{\frac{1}{2}} \qquad \text{(A-7)}$$

Hence by assumption (A-3), it follows that

$$\sum_{i=1}^{n+r} d_i(t)\, r_i(l) \le \sum_{i=1}^{n+r} d_i(l)\, r_i(l) + O(\varepsilon) \qquad \text{(A-8)}$$

It can be shown similarly that

$$\sum_{i=1}^{n+r} d_i(l)\, r_i(l) \le \sum_{i=1}^{n+r} d_i(t)\, r_i(l) + O(\varepsilon) \qquad \text{(A-9)}$$

Thus

$$\left|\sum_{i=1}^{n+r} d_i(1)\, r_i(1) - \sum_{i=1}^{n+r} d_i(t)\, r_i(1)\right| \le O(\varepsilon) \tag{A-10}$$

Using the fact that $|b| - |a| \le |a - b|$ for any $a, b \in R$ yields

$$\left|\sum_{i=1}^{n+r} d_i(t)\, r_i(1)\right| \le \left|\sum_{i=1}^{n+r} d_i(1)\, r_i(1)\right| + O(\varepsilon)$$

Consequently

$$\left|\sum_{i=1}^{n+r} d_i(t)\, r_i(1)\right|^2 \le 2\left|\sum_{i=1}^{n+r} d_i(1)\, r_i(1)\right|^2 + 2\, O(\varepsilon^2) \tag{A-11}$$

And finally (A-4) is obtained by using (A-11) in (A-5).

TABLE I

Upper bound γ^2	Average tap error T	Average number of updates	Total number of times $\lvert y(t)\rvert > \gamma$	Number of times θ^* is out of ellipsoid
0.1	1.12×10^{-2}	154	20207	2935
0.5	1.11×10^{-2}	154	14539	2600
1.0	1.09×10^{-2}	154	10814	6
5.0	1.07×10^{-2}	154	1728	0
15.0	1.08×10^{-2}	153	12	0
25.0	1.14×10^{-2}	154	0	0
50.0	1.45×10^{-2}	156	0	0
200.0	1.15×10^{-2}	156	0	0

TABLE II

Standard deviation of noise	Average tap error T	Average number of updates	Total number of times $\lvert y(t)\rvert > \gamma$	Number of times θ^* is out of ellipsoid
0.25	0.27	139	0	0
0.5	0.24	133	0	0
1.0	0.29	128	436	0
1.5	0.24	122	2705	0
2.0	0.22	119	5968	50
3.0	0.27	118	10728	1573

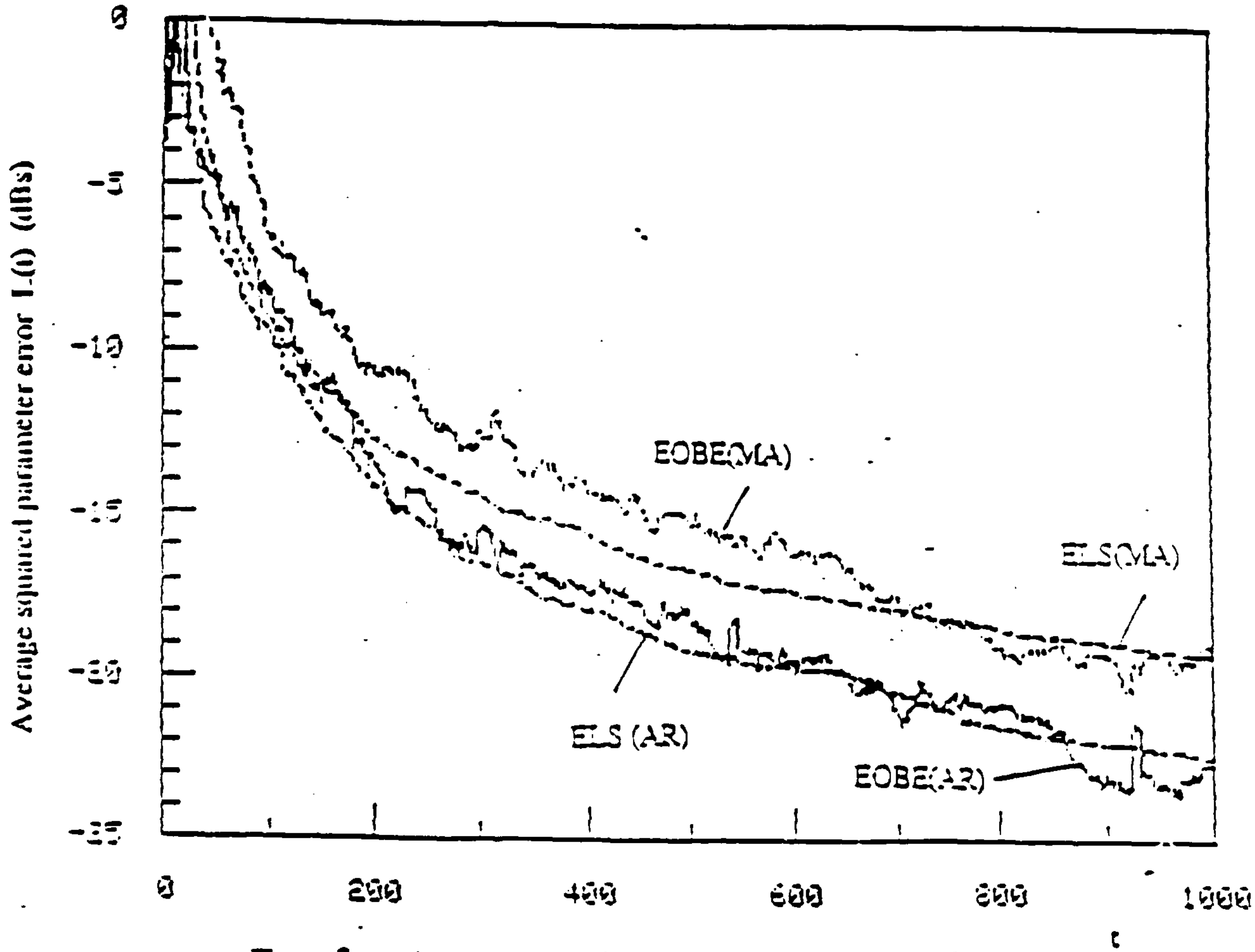


Figure 2. Average squared parameter error for the EOBE and ELS algorithms - Example 1.

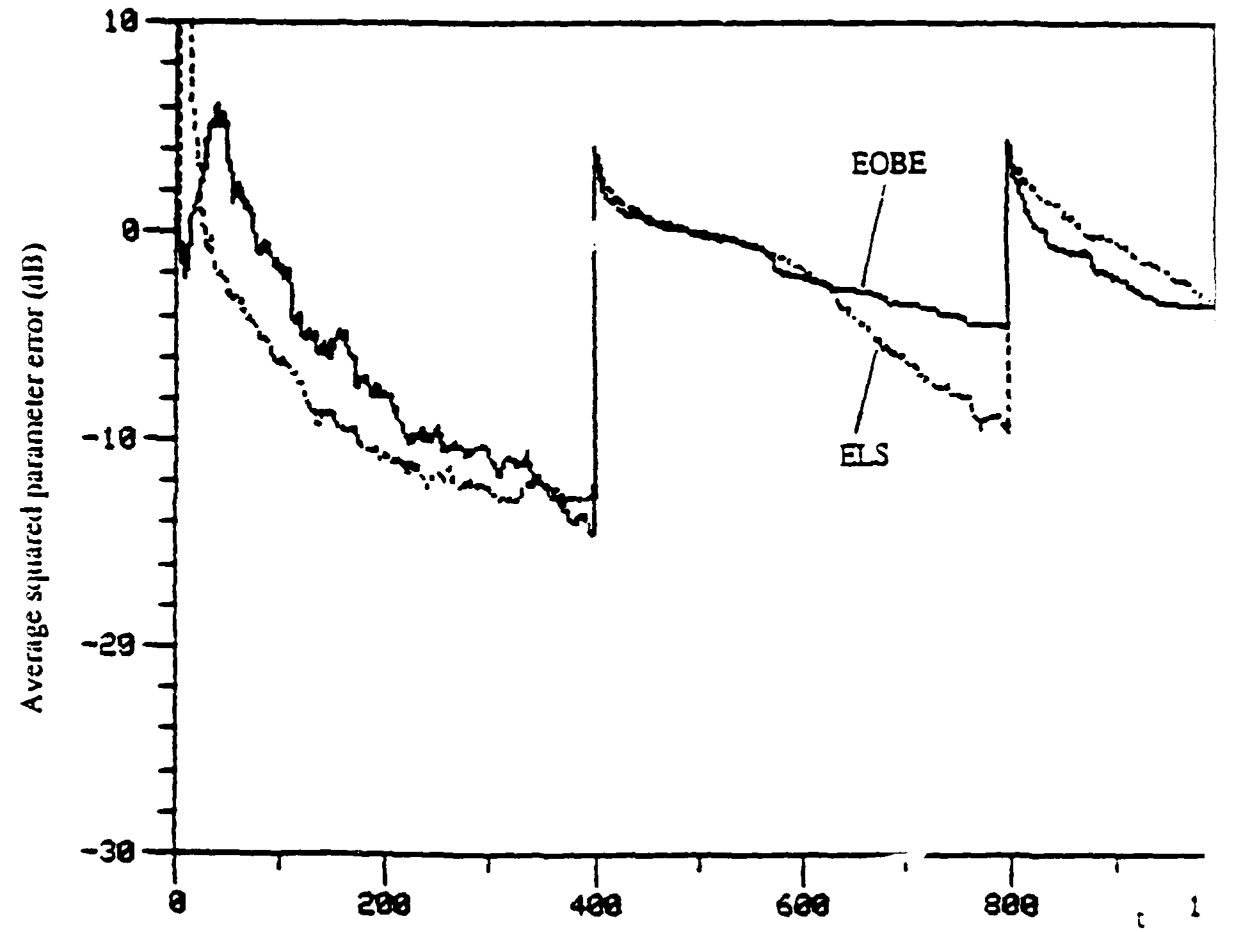


Figure 3. Average squared parameter error for the EOBE and ELS algorithms - Time varying case.

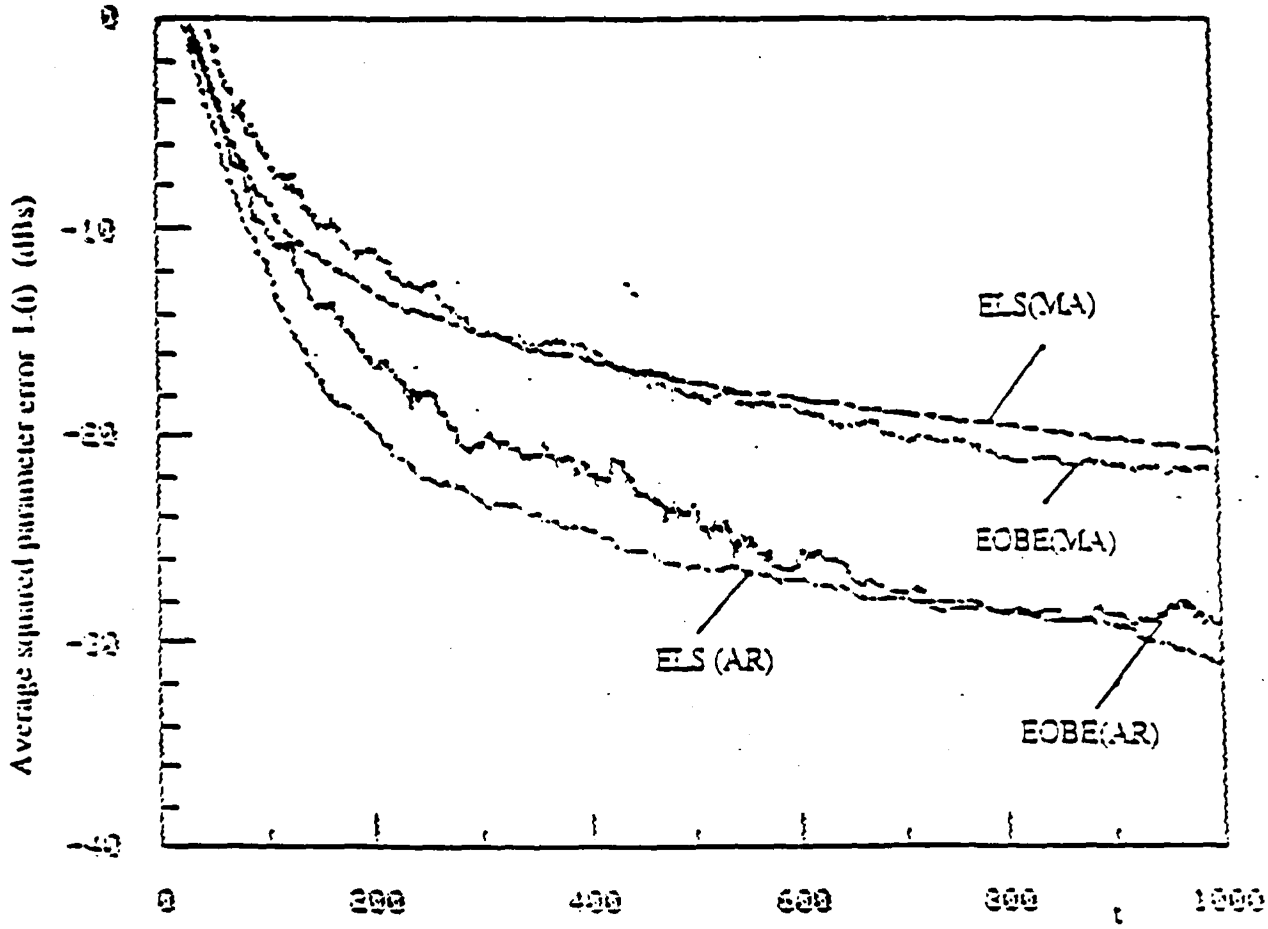


Figure 4. Average squared parameter error for the EOBE and ELS algorithms - Example 2.

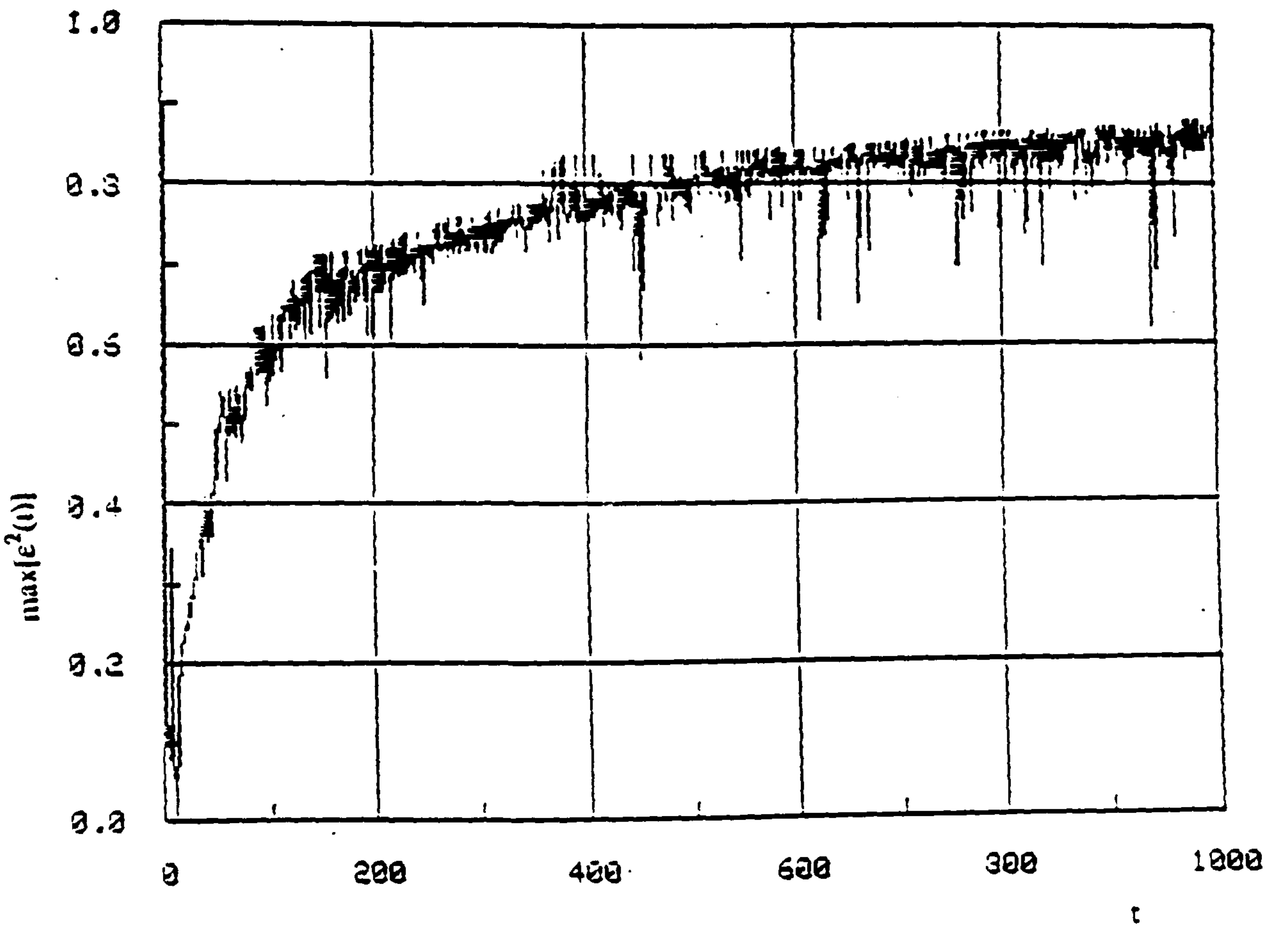


Figure 5. Maximum residual error over 25 runs for
a Non-SPR model - Example 3.