INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

ProQuest Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

Nonlinear System Identification and Parameter Estimation

by

Sheng Lu

A dissertation submitted to the Graduate Faculty in Engineering in partial fulfillment of the requirements for the degree of Doctor of Philosophy, The City University of New York

2002

This manuscript has been read and accepted for the Graduate Faculty in Engineering in satisfaction of the dissertation requirement for the degree of Doctor of Philosophy.

(required signature)

7/9/02
Date

Ki H. Chon
Chair of Examining Committee

July 10th. 2002
Date

Mumtaz K. Kamil
Executive Officer

[typed name]
Stephen Cowin

[typed name]
Basu , Mistra

[typed name]
Nguyen . Thao
Supervisory Committee

THE CITY UNIVERSITY OF NEW YORK

Abstract

NONLINEAR SYSTEM IDENTIFICATION AND PARAMETER ESTIMATION

By

Sheng Lu

Advisor: Professor Ki Chon

I. System Identification.

The ARMA (autoregressive moving average) model plays an important role in system identification which is described as:

$$y(n) = -\sum_{i=1}^{p} a(i)y(n-i) + \sum_{j=0}^{q} b(j)x(n-j) + e(n)$$

The ARMA model is broadly used in many diverse fields ranging from signal processing, communications, biomedicine, to economics.

The true ARMA model order (p,q) is unknown; therefore, to circumvent this inherent limitation with the ARMA model predication, we have developed two novel algorithms to obtain model order.

II. Parameter Estimation

Once we have obtained the accurate ARMA structure, the next step is to calculate the parameters. The conventional method are generally biased. We have developed a new algorithm to overcome this shortcoming.

Acknowledgement

To my father and mother

for their love

Contents:

List of Figures

Chapter 1

Introduction

I. System Identification.

The ARMA (autoregressive moving average) model plays an important role in system identification, especially for input-output system. The basic idea of the ARMA model is to predict next states based on previous output and input data. The ARMA model is described as:

$$y(n) = -\sum_{i=1}^{p} a(i)y(n-i) + \sum_{j=0}^{q} b(j)x(n-j) + e(n)$$

where p and q are autoregressive and moving average model orders, respectively. For more complex systems, a nonlinear ARMA model can be utilized. The nonlinear ARMA model may include quadratic, cubic and higher-order terms of the input and output lag data values. The ARMA model is broadly used in many diverse fields ranging from signal processing, communications [20], biomedicine [26], to economics [24]. Specifically, in the area bioengineering, Abdel-Malek et al. studied differences in parameters obtained by the ARMA model in a manual-tracking experiment between patients with Parkinson's disease and a normal control group [27]. Zigel, Y. Cohen, A.,et. al combine ARMA model and coding techniques to compress the ECG signal and the performance is very competitive [35]. In the field of wireless communications, Davis et al. used an ARMA model to predict fast-fading mobile channels [28]. In

financial engineering, some of the stock market predictions are based on the ARMA model process.

The ARMA model is not only useful for time-domain analysis, but it can also be used to obtain the frequency content of the signal. For example, Chon has used FOS (Fast Orthogonal Search) to distinguish between two frequency components that are very close to each other, which was not possible even with some of the most sophisticated approaches such as MUSIC (Multiple Signal Classification) and modified covariance approaches. Furthermore, nonlinear autoregressive (NAR) model parameters can be used to capture the inner dynamic features of chaotic signals. For example, a NAR model based on the FOS approach has been used to model the dynamics of different chaotic systems.

The true ARMA model order is unknown, therefore, the accuracy of the ARMA model is predicated on selecting a correct model order `a priori`. To circumvent this inherent limitation with the ARMA model prediction, many approaches have been developed to obtain accurate model order determination. The following list includes previous work on the attempt to obtain accurate model order determination:

- The Akaike Information Criterion,

$$I_{\text{Akaike}}(p, q) := \ln \hat{\sigma}^2_{p,q} + 2\frac{p+q}{N},$$

- The Akaike's Final Prediction Error Criterion,

$$I_{FPE}\ (p,q) := \hat{\sigma}^2_{p,q}\,\frac{N+(p+q)+1}{N-(p+q)+1}$$

- The Bayesian Information Criterion,

$$I_{\text{Bayesian}}(p, q) := \ln \hat{\sigma}^2_{p,q} + 2\frac{(p+q)\ln N}{N},$$

- The Hannan-Quinn Criterion,

$$I_{\text{Hannan-Quinn}}(p,q) := \ln \hat{\sigma}^2_{p,q} + 2\frac{(p+q)\cdot c \cdot \ln(\ln N)}{N},$$

where N is the signal length in samples and $\hat{\sigma}^2_{p,q}$ denotes the variance of the residuals of the ARMA[p,q] process.

Most of the above ARMA model order criteria, if not all, do not work well in practice, however. Furthermore, these criteria cannot be applied to nonlinear ARMA models. The shortcoming of these model order criteria is demonstrated via simulation examples in the Results section.

To alleviate drawback inherent in many of the previously developed model order criteria, many novel methods have been developed to obtain an accurate estimate of the model order and consequently the parameters of the ARMA model. Some of the approaches include neural network and the maximum likelihood method. For example, Chon et al. [2] established a two step approach to estimate linear and nonlinear stochastic ARMA model parameters using neural network, in which the parameters of the deterministic part of the stochastic ARMA model are first estimated via a three-layer artificial neural network and then reestimated using the prediction error as one of the inputs to the artificial neural networks using an iterative scheme. In the late 1980's, Korenberg [9,10] developed a new algorithm based on an orthogonal vector search called FOS (Fast Orthogonal Search). Details regarding the FOS algorithm will be described in the following chapter. Although FOS is robust, it is suboptimal and in certain cases, it fails to obtain correct parameters even if there is no noise present in the system. In the following chapters, we will introduce two new algorithms that we have

developed. Simulation results are performed and the new algorithms are compared to the FOS to examine the efficacy of the newly-developed algorithms. The performance of these two algorithms in most cases is better than the FOS algorithm.

II. Parameter Estimation.

Once we obtain the accurate ARMA structure, the next step is to calculate the parameters a_i and b_j, i=1...p, j=1...q. The conventional method is to use least squares (LS). The method of least squares assumes that the best-fit curve is the curve that has the minimal sum of the deviations squared (*least square error*) from a given set of data.

$$e^2 = \min[(Ax-b)^2]$$

With LS, the result can be unbiased only if either the independent variable is noise free or if there is no additive noise present in the system output. This is, however, an unlikely scenario in practical situations.

A more sophisticated approach than the LS, known as the total least squares (TLS) provides more robust parameter estimations than LS. With the TLS, both the matrix A and the vector b are assumed to be corrupted by noise sources such that:

$$(A+r)x=b+e, \text{ where } b+r \in \Re(A+r)$$

The standard procedure to solve the TLS is to use the singular value decomposition (SVD) of the matrix C=[A b]. With the smallest singular values obtained from the SVD of the matrix C, denoted as η, the solution of the TLS is estimated by performing the following:

$$x=(A^TA-\eta^2I)^{-1}A^Tb.$$

By removing the error terms with SVD, it increases the accuracy compared to LS especially in nonlinear case even with the independent variables noise contaminated.

To further improve the performance of the TLS, Yeredor introduced Extended Least Squares (XLS) criterion to distinguish measurement errors from modeling errors by properly weighting and balancing the two error sources [11]. While others have suggested the use of different norm criteria instead of the typical norm-2 criterion [12]. While these latest approaches have further improved the accuracy of the TLS, it has limitation when the signal is corrupted with significant amount of noise.

To drastically enhance the performance of parameter estimation even when the signal is heavily corrupted with noise, we introduce a new method based on the principle of minimizing hypersurface distance (MHD). Chapter 4 introduces MHD method and then compares its performance to those of LS and TLS. One major drawback to the MHD method is that it is far more computationally expansive than either LS and TLS. However, the major advantage of the method is that it provides accurate results even when the signal-to-noise ratio is as low as 0dB. Various Simulation results are provided in Chapter 4 to demonstrate the efficacy of the MHD method.

Chapter 2

The GMDH Algorithm

2.1 Abstract:

A new algorithm for autoregressive moving average(ARMA) parameter estimation is introduced. The algorithm is based on the Group Method of Data Handling (GMDH) first introduced by the Russian cyberneticist, A.G. Ivakhnenko, for solving high-order regression polynomials. The GMDH is heuristic in nature and self-organizes into a model of optimal complexity without any a priori knowledge about the system's inner workings. We modified the GMDH algorithm to solve for ARMA model parameters. Computer simulations have been performed to examine the efficacy of the GMDH and comparison of the GMDH is made to one of the most accurate and one of the most widely-used algorithms, the fast orthogonal search(FOS) and the least-squares methods, respectively. The results show that in cases with noise contamination and incorrect model order assumptions, the GMDH performs better than either the FOS or the least-squares methods in providing only the parameters that are associated with the true model terms.

2.2 Introduction

The ubiquity of autoregressive moving average (ARMA) models in physiological system identification is owed to their having been successfully demonstrated in recent

years (2-4,9,10). Their use been greatly aided by advances in overcoming the holy-grail problem of model order determination during ARMA model parameter estimation (3,9,10). One of the most notable and accurate algorithms introduced to determine model order selection efficiently is the novel work by Korenberg (9,10). Korenberg's fast orthogonal search (FOS) algorithm has been shown to be robust in most cases in obtaining correct ARMA model parameters despite incorrect model order selection. This observation remains valid even with significant noise added to the system response (3,9,10). The FOS algorithm which relies on the sequential search procedure to extract only the significant ARMA model terms, however, is suboptimal. It is suboptimal because the FOS does not test for the minimum error across all possible subsets of ARMA model candidate functions within the candidate function space. Thus, there are cases when the FOS is not able to identify correctly the true ARMA model parameters even with no noise present.

To overcome the aforementioned shortcoming of the FOS, we introduce a new algorithm which searches for all possible ARMA model candidate functions and, thus, provides accurate estimation of the system parameters. The algorithm we present is a modification of the algorithm introduced by the Russian mathematician and cybernetic, A.G. Ivakhnenko, in the mid' 60s. The algorithm was use for estimating higher-order regression polynomials and is denoted the Group Method of Data Handling (GMDH). The GMDH was designed to construct successively higher-order regression equations at each iteration from equations of the previous iteration while retaining only those that best approximate the given data set. In this way a high-order regression type model is evolved guided only by survival of the fittest. The main idea

of the GMDH is to have the algorithm construct a model of optimal complexity based only on data; a self-organizing model that can be used to solve prediction, identification, control synthesis, and other system problems (5-8).

The aim of the present study is to show that the GMDH, originally designed for estimating higher-order regression polynomials, can be modified to be one of the better algorithms for ARMA model parameter estimation. To showcase the efficacy of the GMDH, comparisons between the GMDH, the FOS and the least-squares methods are made using simulation examples. The FOS is chosen since it is one of the most accurate algorithms available for ARMA parameter estimation. Least-squares using the Akaike Information Criteria for model order selection is chosen for comparison since it is still one of the most widely utilized methods for ARMA model parameter estimation, especially in the biomedical engineering community.

2.3 Methods

GMDH Algorithm

We describe a modification of the GMDH algorithm to estimate the parameters of the ARMA model, in this section. The GMDH was originally formulated to solve for higher-order regression polynomials and not the difference equation upon which ARMA models are based. Consider an autoregressive moving average process of the form:

$$y(n) = -\sum_{i=1}^{P} a(i)y(n-i) + \sum_{j=0}^{q} b(j)x(n-j) + e(n) \tag{1}$$

where P and Q represent the maximum autoregressive (AR) and moving average(MA) model orders, respectively. The term e(n) in Eq. (1) is considered a noise source or prediction error term. The parameters a(i) and b(j) represent to be-estimated coefficients of the AR and MA terms, respectively. The basic steps of the newly developed algorithm to calculate ARMA model parameters are as follows:

Step(1) Partition the data with the first n rows designated as the training set and the remaining rows as the testing set. From the training set form a matrix of N observations of ARMA model terms like the ones shown below and ignore the prediction error terms, e(n), in Eq. (1).

$$\begin{matrix} y(1) \\ y(2) \\ \vdots \\ y(N) \end{matrix} = \begin{matrix} y(0) & y(-1) & \cdots & y(1-P) & x(1) & x(0) & \cdots & x(1-Q) \\ y(1) & y(0) & \cdots & y(2-P) & x(2) & x(1) & \cdots & x(2-Q) \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ y(N-1) & y(N-2) & \cdots & y(N-P) & x(N) & x(N-1) & \cdots & x(N-Q) \end{matrix}$$

Matrix R

Step (2) Take all the variables in the columns of the matrix R{y(n-1),..,y(n-P), x(n),x(n-1),..,x(n-Q)} two columns at a time and for each of these m(m-1)/2 combinations find the least squares regression that best fits the observation (vector) y's. For each of the combinations evaluate the least squares of the N data points. After evaluating N values, store these N values in the first column of a new array Z. The remaining [m(m-1)/2-1 columns are constructed in a similar manner. The array Z

contains the new variables, which replace the original variables. The objective is to retain those z's that best estimate the output vector y and discard the insignificant variables.

Step (3) To determine which columns of Z (new variables) replace the old variables in the matrix R, compute the least squares error, d_j:

$$d_j^2 = \sum_{i=1}^{N} (y_i - z_{ij})^2 / \sum_{i=1}^{N} y_i^2 \qquad j = 1,2,\ldots,\binom{m}{2} \tag{2}$$

and order the columns of Z according to increasing least square error. One can place a restriction as to some prescribed number of new variables to replace the old variables in the matrix R, i.e., d_j <M, where M is some prescribed number.

Step (4) From Step(3) take the smallest of the d_j. If the value of d_j is smaller than the previous d_j (in the first iteration we assume this to be true) go back and repeat steps (2) and (3). If the value of d_j is greater than the previous d_j we stop the process. It is our observation, however, that the value of d_j becomes smaller than the previous d_j as the number of iterations is increased. It is our experience that performing more iterations results in smaller mean-square error but at the expense of an increase in the number of model terms. Based on our simulations, we have determined that three iterations are sufficient to obtain accurate parameter estimates. Figure illustrates the iterative process involved in the GMDH algorithm. The layers in Fig. 1 correspond to the number of iterations; for three iterations, we have three layers. Note that the first

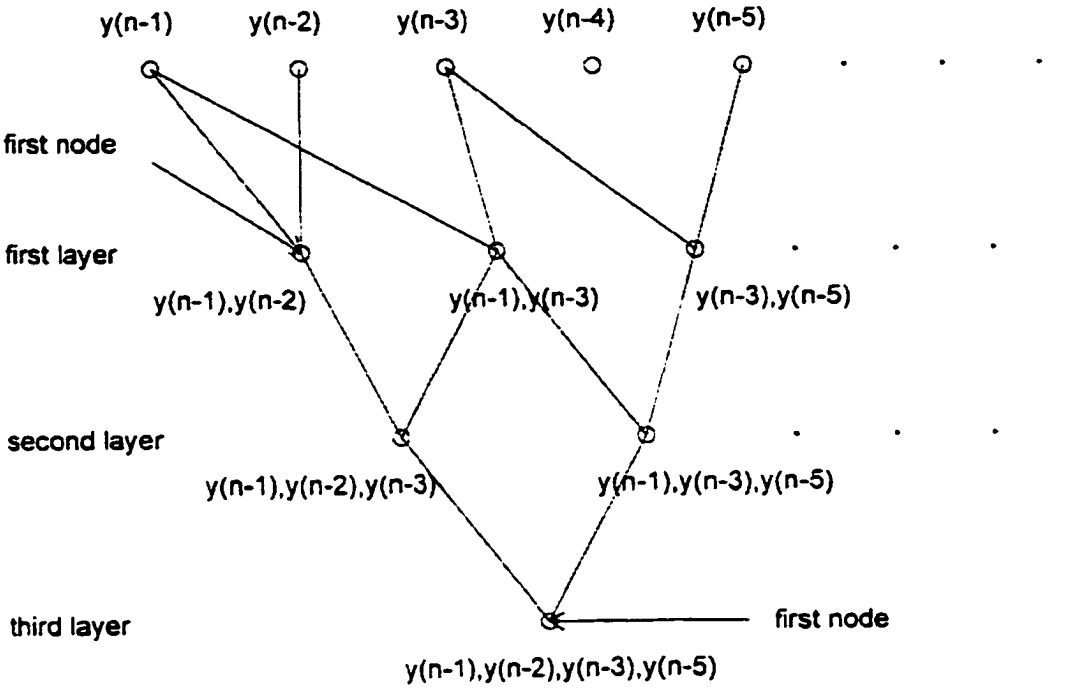


Fig 1. Three layer iterative process of the GMDH algorithm, each layer corresponds to the number of iterations.

and second nodes in the first layer of Fig. 1 may represent terms {y(n-1) and y(n-2), and {y(n-1) and y(n-3)}, respectively, if these terms meet the criteria described in Step (3). With increasing iteration, the number of layers will increase and the number of terms in the nodes at that layer may increase drastically, provided that terms pertaining to those nodes meet the criteria in Step (3). Note that in the third layer, the nodes are ordered according to increasing least square error, i.e., node one is more significant than node two and so on.

Step (5) The GMDH algorithm calls for the use of the least squares to find the coefficients of the retained variables. However, we have made a modification to the original GMDH algorithm at this step. Instead of computing the least squares of all the retained variables (or all of the nodes in layer three), we compute the contribution of each of the nodes to reducing the mean-square error. If we observe either negligible decrease or increase in the mean-square error by adding an additional node, then those nodes are dropped from the model. Only those nodes that reduce the mean-square error by adding an additional node, then those nodes are dropped from the model. Only those nodes that reduce the mean-square error are retained, and the coefficients pertaining to the retained nodes are estimated using the least-square algorithm. The Simulation Results section discusses this step further (Fig.2).

The distinct advantage of the GMDH is more apparent when the number of variables and the degree of the polynomials are large. For example, with least-squares, to find a regression polynomial with 4 variables of degree 4, 70 linear equations with 70 unknowns must be solved. This is not a daunting task with the present computing power, however, the equations are often ill-conditioned and consequently cannot be

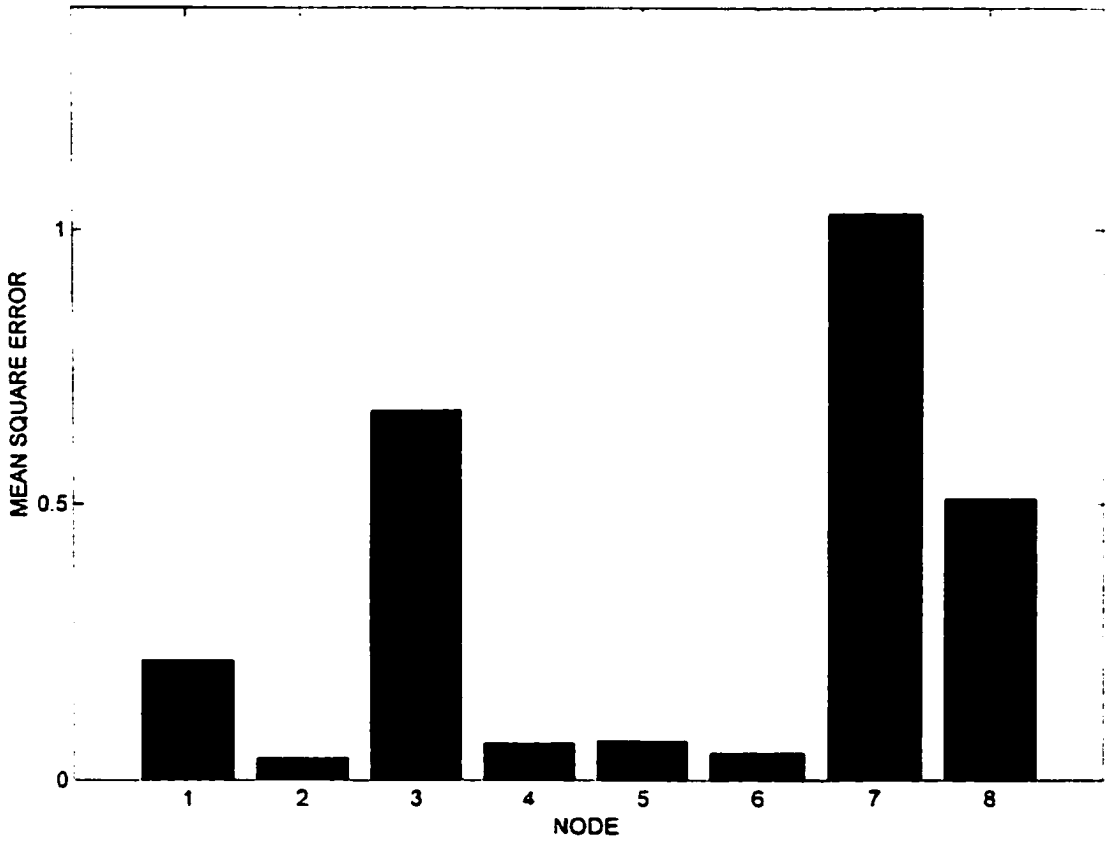


Fig 2. Bar graph of the difference between MSE values for adjacent nodes as a function of the lower nodes numbers.

solved. The GMDH polynomials, however, can be obtained by solving far fewer linear systems of order six. The reason the GMDH requires fewer computations is that it retains only the information that is highly correlated with the checking data set. Consequently, this allows the computation to be more efficient and enables the system of normal equations to be well-conditioned.

Note that the method outlined is also applicable to subclasses of linear and nonlinear ARMA models such as autoregressive (AR) and moving average (MA) models.

2.4 Simulation Results

In this section we demonstrate the effectiveness of the developed algorithm for estimating parameters of the ARMA model. We compare the GMDH results with parameters obtained via the methods of FOS and the least-squares. We chose the FOS since , as far as we are aware, it is one of the most accurate algorithms available. The least-squares is chosen since it is still one of the most widely used methods in practice. Note that the FOS employs its own robust automatic model order selection whereas the least-squares method relies on the use of model order selection schemes such as the Akaike information criteria (AIC) (1). For all simulation examples, we iterate the GMDH algorithm 3 times as discussed in the methods section.

Effects of Incorrect Model Order Selection

For the first simulation example, the following linear ARMA model was simulated with Gausian white noise (GWN) as the input, x(n), so that the output, y(n), contained 1000 data points:

$$y(n)=-0.54y(n-1)+0.34y(n-2)+0.23y(n-3)-0.54x(n)+0.43x(n-1)+0.32x(n-2)+0.21x(n-3) \quad (4)$$

The objective is, based on only the measured input signal, x(n) and the output signal, y(n), estimate the parameters of the above equation as accurately as possible. Although the true ARMA model order for the above process is 3 output lags (y(n-1)...y(n-3)) and 3 input lags (x(n),x(n-1),.. x(n-3)), we purposely selected an incorrect model order of 10 output lags (y(n-1),...,y(n-10)) and 10 input lags (x(n),x(n-1),..,x(n-1)) for all three methods. In real life settings, a true model order is unknown a priori, thus, the real test of an algorithm is to examine its efficacy with an incorrect model order. The model orders for the least-square method were determined according to the AIC and were correctly determined to be 3 output lags and 3 input lags. Figure 2 shows the plot of the difference between mean square error values for adjacent nodes versus the lower node number as a venue for determining model order for the GMDH algorithm. As shown in Fig. 2, we decided to terminate the computation of the GMDH algorithm after six nodes since this corresponded to the minimum error value. It should be pointed out that it is not crucial that we select the minimum error value as the slightly higher values associated with four and five nodes all do result in the same correct ARMA model parameters. The reason we obtain the same model coefficients whether we choose four, five, or six nodes is that, for example, nodes five and six may contain model terms that are already contained in node four. Thus, no new terms from those of node four have been added in nodes five and six, resulting in the same coefficients whether node four or node six is chosen.

Comparison of the results based on the three methods is shown in Table 1. Only the new algorithm (GMDH) and the least-squares are able to identify correctly the true

parameters in the system model without concomitant estimation of the incorrect model terms. The least-squares and GMDH model correctly found coefficients of zero for all the spurious model orders, so they are not shown in Table 1. The FOS, however, performs the worst for this simulation as exemplified by the incorrect estimation of parameters and spurious model terms. It should be pointed out that only in certain cases, such as this example, does the FOS not correctly identify the parameters. In many instances, the FOS does provide accurate parameter estimates despite inaccurate model order selections (3,9,10).

Table 1. Comparison of GMDH, FOS and the least-squares with an incorrect model order selection.

ModelTerms	$y(n-1)$	$y(n-2)$	$y(N-3)$	$y(n-5)$	$x(n)$	$x(n-1)$	$x(n-2)$	$x(n-3)$	$x(n-5)$
Truevalues	−0.54	0.34	0.23	0.00	−0.54	0.43	0.32	0.21	0.00
GMDH	−0.54	0.34	0.23	0.00	−0.54	0.43	0.32	0.21	0.00
FOS	−0.21	−0.29	0.00	0.05	−0.54	0.61	0.06	0.20	0.05
Least−squares	−0.54	0.34	0.23	0.00	−0.54	0.43	0.32	0.21	0.00

ARMA Model with Additive Noise and Incorrect Model Order Selection

To examine the effectiveness of the GMDH in the case of noise contamination, the output signal of Eq. (4) was corrupted with two levels of additive GWN, resulting in signal-to-noise-ratios (SNR) of 10dB and 0dB. The SNR was determined by the ratio of the variance of the signal divided by the variance of the noise signal. The model order was incorrectly selected to be 10 output and 10 input lags for all methods compared. This is a challenging and realistic scenario; not only is the model order purposely incorrectly chosen, but the output signal is heavily corrupted by noise. The results of additive noise for the GMDH, FOS and the least-squares are shown in Table 2. With the SNR of 10dB, the GMDH correctly provides parameters that are

associated with only the true model terms but this is not the case for the FOS; the FOS missed a model term, y(n-3), and incorrectly picked an additional term, x(n-4). The least-squares approach based on AIC resulted in a model order selection of AR=7 and MA=7. Since the least-squares method does not have automatic model order selection, it resulted in estimates of terms and coefficients not included in the actual system in Eq. 4 (due to an overdetermined model order assumption). The normalized mean-square errors(NMSE) for all methods are comparable, however, it should be noted that for the least-squares, the comparable NMSE was obtained with 15 parameters whereas for both the GMDH and FOS, the NMSE values were obtained with only 7 parameters. More qualitative differences between the methods compared emerge when the SNR is 0dB. The GMDH again provided only the correct model terms as was the case when the SNR was 10dB. The FOS missed three model terms, y(n-1), y(n-2) and y(N-3). In addition, the FOS incorrectly picked three additional model terms, y(n-5), x(n-4) and x(N-5). The AIC model order selection resulted in AR=0 and MA=7, and the resulting coefficients via the least-squares are provided in Table 2. With significant noise added, the least-squares based on AIC completely missed the AR terms and then compensated by including more MA terms than necessary. Because the FOS was designed to reduce the NMSE, despite incorrect model terms, the NMSE value obtained for the FOS is comparable to the GMDH and the least-square.

Table 2. Comparison of GMDH and FOS with additive noise and an incorrect model order selection.

ModelTerms	$y(n-1)$	$y(n-2)$	$y(n-3)$	$y(n-5)$	$x(n)$	$x(n-1)$	$x(n-2)$	$x(n-3)$	$x(n-4)$	$x(n-5)$
Truevalues	−0.54	0.34	0.23	0.00	−0.54	0.43	0.32	0.21	0.00	0.00
GMDH(10dB)	−0.13	0.24	−0.04	0.00	−0.54	0.66	−0.03	0.22	0.00	0.00
FOS(10dB)	−0.17	0.19	0.00	0.00	−0.55	0.64	0.00	0.27	−0.06	0.08
Least−squares(10dB)	0.01	0.01	0.00	−0.06	−0.55	0.72	−0.27	0.47	−0.18	0.16
GMDH(0dB)	−0.06	0.06	−0.09	0.00	−0.56	0.65	−0.22	0.38	0.00	0.00
FOS(0dB)	0.00	0.00	0.00	−0.09	−0.57	0.69	−0.30	0.49	−0.16	0.14
Least−squares(0dB)	0.00	0.00	0.00	0.00	−0.56	0.71	−0.28	0.48	−0.19	0.19

ARMA Model with Missing Terms

The next example considers an ARMA process with missing terms and 1000 data points generated for the output y(n):

$$y(n)=0.25y(n-1)+0.1y(n-2)-0.1y(n-6)+0.8x(n)-0.3x(n-1)+0.25x(n-6) \quad (5)$$

Although the true ARMA model order for the process of Eq. 5 is AR=6 and MA=6, we have again purposely selected an incorrect model order of ARMA (10,10) for all three methods. The Akaike Information criteria correctly identified model order of ARMA (6,6). The FOS, due to its automatic model order search criterion, also identified only the correct parameters. The GMDH method provided correct parameter estimates for 4 to 9 nodes. Thus, similar to the previous example, it is not imperative that the exact minimum error value is used as a criterion in deciding when to terminate the GMDH algorithm. Thus, despite an incorrect model order assumption, all three methods provided only the correct parameters. This is clearly and impressive results

by all of the methods compared considering the fact that a grossly incorrect model order was chosen a priori.

ARMA Model with Incorrect Model Order Selection

To further examine the efficacy of the GMDH, consider an ARMA process described by the equation:

$$y(n)=-0.546y(n-1)+0.34y(n-2)+0.243y(n-3)+0.12y(n-4)-0.67x(n)+0.454x(n-1)+0.342x(n-2)+0.123x(n-3)+0.32x(n-4) \quad (6)$$

where the input x(n) is the same input as used in Eq. 4. For all three methods compared, we chose an incorrect model order of AR=10 and MA=10. Table 3 shows the results. We observe that only the least-squares based on AIC model order selection was not able to identify the correct parameters since AIC resulted in model order of AR=3 ,MA=7. The GMDH provided the correct model terms except for the term y(n-4), which in turn resulted in incorrect parameter estimates. The FOS did not fare as well as the GMDH method. It not only missed some of the model terms but it incorrectly identified terms that are not in the true model. The NMSE values for the GMDH and FOS are 1.21×10^{-13} *and* 4.80×10^{-5} respectively. As shown in Table 3, the GMDH performed the best compared to the other two methods. This example is provided to show that the GMDH does not always provide accurate parameters, especially for a simple example such as this.

Table 3. Comparison of GMDH, FOS and the least-squares with an incorrect model order selection.

ModelTerms	*y(n−1)*	*y(n−2)*	*y(n−3)*	*y(n−4)*	*y(n−6)*	*y(n−7)*	*x(n)*	*x(n−1)*	*x(n−2)*	*x(n−3)*	*x(n−4)*	*x(n−6)*	*x(n−7)*
TrueValues	−0.546	0.340	0.243	0.120	0.000	0.000	−0.67	0.454	0.342	0.123	0.32	0.000	0.000
GMDH	−0.460	0.491	0.141	0.000	0.000	0.000	−0.67	0.511	0.374	−0.04	0.34	0.000	0.000
FOS	0.000	0.497	0.000	0.000	0.017	−0.012	−0.67	0.819	0.000	0.013	0.26	0.000	0.015
Least-squares	−0.432	0.517	0.200	0.000	0.000	0.000	−0.67	0.53	0.367	−0.013	0.286	−0.051	0.031

ARMA Model with Dynamic Noise and Incorrect Model Order Selection

The second simulation discussed included additive noise. The noise source was added after the output sequence, y(n), had been generated. However, noise sources can also be of a dynamic nature. Dynamic noise refers to a feedback process where the noise source is recursively added to the output so that the current and future output values are dependent on the past states of the input and noise signals. Thus, to examine the effect of dynamic noise consider an ARMA process described by the difference equation:

$$y(n)=-0.54y(n-1)+0.34y(n-2)+0.23y(n-3)-0.54x(n)+0.43x(n-1)+0.32x(n-2)+0.21x(n-3)+0.25e(n)+0.3e(n-1) \quad (7)$$

Both the input x(n) and the noise component terms [e(n)] were generated using 1000 Gaussian white noise data points that were independent from each other. The noise component e(n) was selected to give a SNR of 10.4 dB. We chose an incorrect model order of AR=10 and MA=10 for all three methods compared. Table 4 shows the results of the coefficients estimated via the three methods. It is clear that only the GMDH provided all of the model terms correctly. The least-squares method completely ignored the AR terms and included a few extra MA model terms that are not in the true model, owing to the AIC model order selection of AR=0, MA=7. The

FOS missed the y(n-3) term and provided two extra terms, x(n-4) and x(n-5). The NMSE values are about 9% for all three methods.

Table 4. Comparison of GMDH and FOS with dynamic noise and an incorrect model order selection.

ModelTerms	$y(n-1)$	$y(n-2)$	$y(n-3)$	$x(n)$	$x(n-1)$	$x(n-2)$	$x(n-3)$	$x(n-4)$	$x(n-5)$	$x(n-6)$	$x(n-7)$
TrueValues	−0.54	0.34	0.23	−0.54	0.43	0.32	0.21	0.00	0.00	0.00	0.00
GMDH	0.04	0.36	−0.07	−0.54	0.77	−0.04	0.18	0.00	0.00	0.00	0.00
FOS	0.11	0.33	0.00	−0.53	0.80	−0.15	0.26	−0.13	0.05	0.00	0.00
Least−squares	0.00	0.00	0.00	−0.55	0.71	−0.25	0.46	−0.16	0.20	−0.06	0.05

We have performed many other simulation examples with the outputs corrupted with various types of noise and incorrect model order assumptions, and it was found that the GMDH generally outperformed the other two methods.

2.4 Conclusions

We presented a new algorithm, the Group Method of Data Handling, for estimating the parameters of ARMA models. The algorithm can be extended to estimate subclasses of linear ARMA models such as autoregressive (AR) and moving average (MA) models, as well as nonlinear ARMA (NARMA) models. In addition, the algorithm can be extended to model the prediction error terms which will significantly improve the model prediction. Various simulation examples have shown the efficacy of the GMDH method. In certain cases of noiseless systems, the least-squares (based on AIC model order selection) is superior to FOS. However, when the system output is confronted with noise (either additive or dynamic), the GMDH method performed

best followed by the FOS and the least-squares in providing only the true model terms. If one is interested in obtaining the smallest mean-square error with the fastest computation time, then simulation examples suggest that the FOS would be the ideal method to be use. However, if obtaining the most accurate representation of the true model is the necessity, then the GMDH method should be the algorithm of choice. As has been reported in the literature, the least-square method based on AIC model selection can work well if the system is noise free. However, the accuracy of the least-squares approach to ARMA model parameter estimation degrades considerably when the system output is corrupted with any type of noise source. The GMDH, unlike the other two methods compared, provides accurate model terms even in cases with significant noise perturbation as well as incorrect model order selection.

Chapter 3

The OPS Method

3.1 Abstract

A linear and nonlinear autoregressive moving average (ARMA) identification algorithm is developed for modeling time series data. The new algorithm is based on the concepts of affine geometry in which the salient feature of the algorithm is to remove the linearly-dependent ARMA vectors from the pool of candidate ARMA vectors. For noiseless time series data with a priori incorrect model order selection, computer simulations show that accurate linear and nonlinear ARMA model parameters can always be obtained with the new algorithm. Many algorithms, including the fast orthogonal search (FOS) algorithm, are not able to obtain correct parameter estimates in every case, even with noiseless time series data, because their model order search criteria are suboptimal. For data contaminated with noise, computer simulations show that the new algorithm performs better than the FOS algorithm for moving average MA) processes, and similarly to the FOS algorithm for ARMA processes. However, the computational time to obtain the parameter estimates with the new algorithm is faster than with FOS. Application of the new algorithm to experimentally-obtained renal blood flow and pressure data show that the new algorithm is reliable in obtaining physiologically understandable transfer function relations between blood pressure and flow signals.

3.2 Method

The key difference between the OPS and the FOS algorithm is that, unlike the FOS, the OPS is based on a non-orthogonal search for model candidate terms. One notable disadvantage of FOS's using an orthogonal search can be seen by the following simple example. A vector P is in the space constructed by vectors X and Y as shown in Fig. 3. If the angle formed by ZOP is smaller than angles YOP and XOP, then the vector Z will be chosen even though the vector Z is not in the space constructed by the vector X and Y. In other words, orthogonal projection finds the closest point to P in the span of the basis vectors, regardless of whether or not the vector belongs to the constructed space. Note that this scenario will produce an erroneous parameter estimate with the FOS when an incorrect model order is chosen a priori. With the OPS, because it is based on a non-orthogonal search, this type of scenario will not occur. Note that if the base vectors are all perpendicular to each other , then there is no difference between orthogonal and non-orthogonal search methods.

The first step in the OPS algorithm is to select only the linearly-independent vectors from the pool of candidate vectors. To examine how linearly-independent vectors are selected, consider an autoregressive moving average process of the form:

$$y(n) = -\sum_{i=1}^{p} a(i)y(n-i) + \sum_{j=0}^{q} b(j)x(n-j) + e(n) \qquad (1)$$

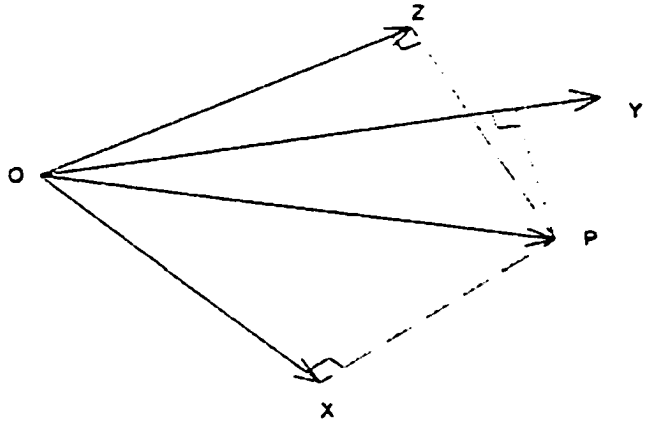


Fig 3. Orthogonal projection finds the closest point to P in the span of the basis vectors.

Where P and Q represent the maximum autoregressive(AR) and moving average(MA) model orders, respectively. The term e(n) in Eq. (1) is considered a noise source or prediction error term. The parameters a(i) and b(j) represent to-be-estimated coefficients of the ARM and MA terms, respectively. The candidate vectors are the following: y(n-1),...,y(n-P) and x(n),..., x(n-Q). These candidate vector can be arranged as the following matrix:

$$\begin{matrix} y(1) \\ y(2) \\ \vdots \\ y(N) \end{matrix} = \begin{matrix} y(0) & y(-1) & \cdots & y(1-P) & x(1) & x(0) & \cdots & x(1-Q) \\ y(1) & y(0) & \cdots & y(2-P) & x(2) & x(1) & \cdots & x(2-Q) \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ y(N-1) & y(N-2) & \cdots & y(N-P) & x(N) & x(N-1) & \cdots & x(N-Q) \end{matrix}$$

For a nonlinear ARMA model, the above matrix can be expanded to include products between the input and output itself as well as cross products between the input and output terms. The first step towards achieving linear independence among candidate vectors is to select y(n-1) as the first candidate vector. The next candidate vector y(n-2), and the first candidate vector y(n-1) are then used to determine linear independence (e.g., use y(n-1) to fit y(n-2) using the least squares method and calculate the error between y(n-2) and the estimated vector). With a perfectly clean signal, linear independence will always be obtained. In the case of correlated noise contamination (error value will not be zero), some preset threshold can be set so that if the error value is smaller than the preset threshold, then the vector y(n-2), for example, can be selected as an independent candidate vector. For simulation examples to be considered in the Results section, we did not use the preset threshold value because it is often difficult to determine a priori what that value should be to obtain correct results, especially with experimental data. Once it has been determined that y(n-2) is a linear

independent candidate vector, the vectors y(n-1)and y(n-2) are used to estimated the candidacy of the linear independence of y(n-3) using the approach just outlined. This procedure is continued until all the linearly-independent vectors are selected to form a new candidate vector pool. Let ϕ=[$w_0, w_1, w_2, ..., w_R$] where R is the number of selected linearly-independent vectors.

With the new candidate pool of linearly-independent vectors, least-squares analysis is performed:

$$y(n) = \theta_g^T \phi + e(n) \tag{3}$$

Where

$$\theta_g = [g_0, g_1, \cdots, g_R]^T \tag{4}$$

The objective is to minimize the equation error, e(n), in the least-squares sense using the criterion function defined as follows:

$$J_N(\theta_g) = \left[y(n) - \theta_g^T \phi\right]^2 \tag{5}$$

The criterion function in Eq. (5) is quadratic in θ_g, and can be minimized analytically with respect to θ_g, yielding the following well-known equation:

With the obtained coefficients, calculate every $\left|g_m^2 w_m^2\right|$, and rearrange the w_m in the descending order.

$$\hat{\theta}_g = \left[\phi\phi^T\right]^{-1} \phi y(n) \tag{6}$$

Note that over bar represents the time average. At this step of the algorithm we need to choose the number of candidate vectors, w_m, necessary for obtaining proper accuracy.

The approach we have taken is to retain only those w_m that reduce the error value significantly. If we observe either negligible decrease or increase in the error value by adding an additional w_m, then those w_m are dropped from the model. Once only those w_m that reduce the error value significantly are obtained, the linear and nonlinear ARMA model terms are estimated using the least-squares method. The simulation results section discusses this step further.

3.3 Simulation Results

In this section we demonstrate the effectiveness of the developed algorithm for estimating parameters of linear and nonlinear MA and ARMA models. We compare the performance of the OPS to that of the FOS method. For all simulations examples involving linear processes (AR and MA) to follow, we have selected an incorrect model order of 10 AR and 10 MA terms (ARMA (10,10)) for both the OPS and FOS. For nonlinear models, 10 AR, 10 MA and 5 second order AR and MA as well as 5 cross AR and MA model orders were chosen for both the OPS and FOS. The Achilles' heel of ARMA models is to determine a priori accurate model orders without knowing the true model order of the system. The model order search for the FOS was obtained by using the automatic model order search criteria as described in [5,6]. Succinctly, the automatic model order search criteria of the FOS retains only those candidate terms that reduce the mean-square error values by a significant amount in conjunction with a statistical 95% interval criterion. The FOS algorithm has been shown to be accurate for various linear and nonlinear ARMA models[5,6,8], and often superior to

the least-squares approach with the Akaike Information criteria for the model order selection process.

I. Moving Average (MA) model with Additive Noise and Incorrect Model Order Selection.

For the first simulation example, consider the following linear moving average (MA) model with Gaussian White noise (GWN) as the input, x (n), so that the output, y (n), contains 1000 data points:

$$y(n)=0.34x(n)-0.23x(n-1)+0.5x(n-2)+0.75x(n-3)-0.15x(n-4)-0.4x(n-5)+0.35x(n-6)-0.2x(n-7) \quad (7)$$

We have purposely selected an incorrect model order of ARMA (10,10) for both methods despite the fact that the above equation does not contain any AR model terms. To subject the algorithm to a more daunting task, we have used additive noise so that signal-to-noise ratios (SNR) of 10 and 0dB were obtained for the above MA process. Comparison of the results based on the two methods for the case of noiseless data with an incorrect model order selection, and the cases with noise added (10 and 0dB) and an incorrect model order selection, are shown in Table 5. With a clean signal, the OPS obtained true model terms and coefficients despite the exaggerated incorrect model order selection. The model order selection process for the OPS is shown in Fig. 4. The ordinate value represents the projection distance of the candidate terms. Note that the projection distance value is zero after the term x (n-4) has been determined. In addition, the projection distance value is the greatest for the term x (n-3) and the lowest for x (n-4) since they have the biggest and smallest coefficient values among the model candidate terms. Thus, without any noise contamination, due to removal of linearly dependent candidate terms, the OPS is able to obtain the correct coefficient

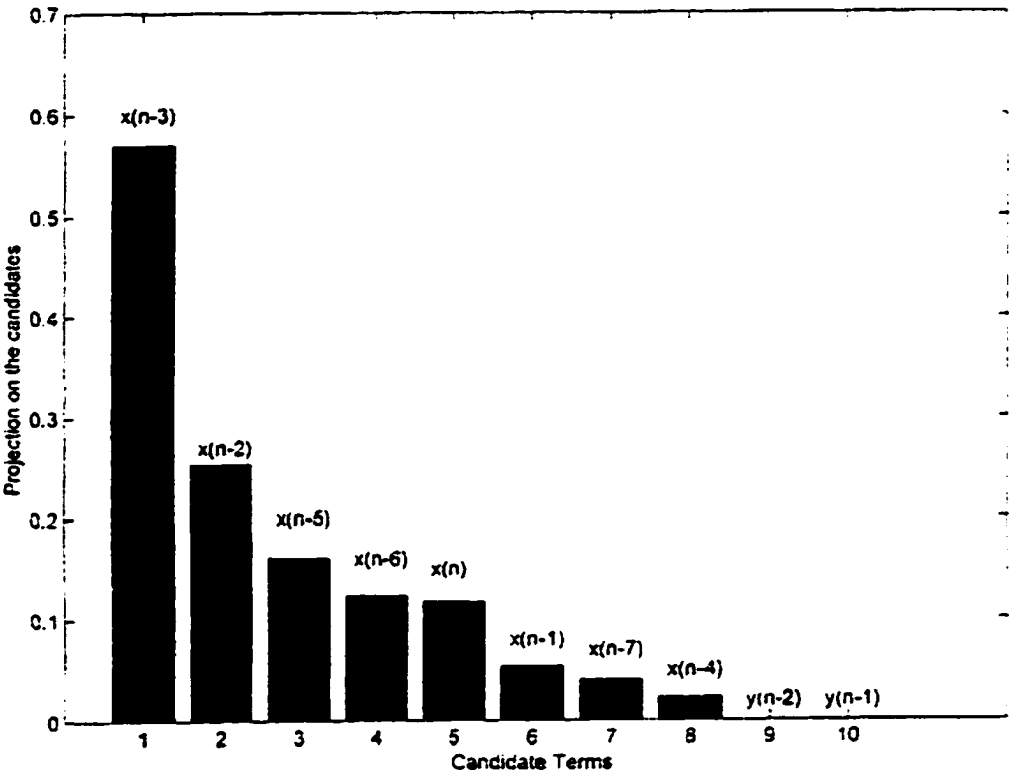


Fig 4. Model-order determination via the projection distance of the candidate terms (noiseless case)

and model candidate terms. The model order search for the FOS was estimated using the automatic model order search criteria as described above and in [5,6]. With this search, however, the FOS completely missed two model terms: x (n-1) and x (n-7). Concomitantly, the FOS incorrectly picked up two additional terms: x (n-8) and y (n-1). With a clean signal, the mean-square-errors (MSE) are 0.0069 for the FOS and 0.00 for the OPS.

When the signal was corrupted with 10dB noise, the OPS correctly identified only the true model terms but with a small deviation of the coefficients from the true coefficients, as expected. The MSE is found to be quite small (9.21e—004). The model order was determined with the aid of Fig. 5. Due to additive noise, the terms y (n-2) and y (n-6) erroneously have small projection distance values. In addition, the projection distance is quite negligible compared to the rest of the candidate terms. Thus, only the first 8 candidate terms were used to estimate the candidate coefficients. The FOS fared poorly compared to the OPS, as it incorrectly identified additional terms that are not part of the true model. In addition, the identified coefficients deviated from the true coefficients, resulting in a slightly higher MSE (0.0060) value than for the OPS. With increased noise level (SNR of 0dB), as expected, the performance of both FOS and OPS deteriorated further. Both methods missed the x (n-4) term. It is interesting to note that in many simulations, including the present example, the FOS appears to be more robust with noise corrupted signals than with clean signals. For example the MSE value is slightly lower with 10dB noise (MSE=0.006) than with the clean signal (MSE=0.0069). In addition, with noise added, the FOS obtained more correct model terms than without noise added.

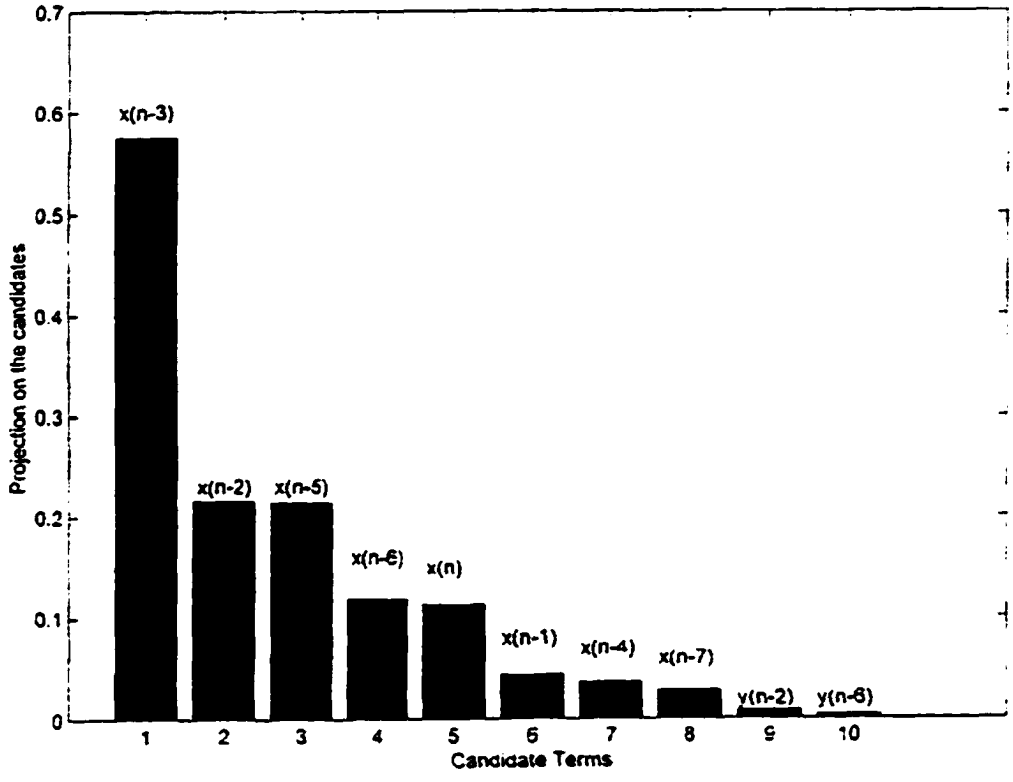


Fig 5. Model-order determination via the projection distance of the candidate terms (SNR=10dB)

Table 5. Comparison of the OPS and FOS with additive noise and an incorrect model order selection.

ModelTerms	$x(n)$	$x(n-1)$	$x(n-2)$	$x(n-3)$	$x(n-4)$	$x(n-5)$	$x(n-6)$	$x(n-7)$	$x(n-8)$	$y(n-1)$
TrueValue	0.34	−0.23	0.5	0.75	−0.15	−0.4	0.35	−0.20	0.00	0.00
OPS(clean)	0.34	−0.23	0.5	0.75	−0.15	−0.4	0.35	−0.20	0.00	0.00
FOS(clean)	0.34	0.0	0.36	1.06	0.32	−0.49	0.10	0.00	−0.12	−0.62
OPS(10dB)	0.34	−0.22	0.51	0.76	−0.13	−0.42	0.35	−0.20	0.00	0.00
FOS(10dB)	0.34	−0.18	0.48	0.83	0.00	−0.44	0.30	−0.15	0.00	−0.13
OPS(0dB)	0.34	−0.21	0.54	0.77	0.00	−0.45	0.36	−0.24	0.00	0.00
FOS(0dB)	0.34	−0.21	0.54	0.77	0.00	−0.45	0.36	−0.24	0.00	0.00

II. Nonlinear MA model with separate additive or dynamic noise and incorrect model order selection

The next simulation example consists of an arbitrarily-chosen nonlinear MA difference equation of the form:

$$y(n)=0.23x(n)+0.4x(n-1)+0.6x(n-2)+0.54x(n-3)+0.28x(n-4)+0.8x(n-5)-0.76x(n-6)+0.35x(n-7)-0.23x(n-8)+0.22x(n-9)-0.65x(n-1)x(n-3) \quad (8)$$

We consider two separate cases of noise corruption, in which the output of Eq.(8) is contaminated by additive Gaussian White noise of the form:

$$z(n)=y(n)+e(n) \quad (9)$$

And the other in which the output of Eq. (8) is disturbed by dynamic noise, which causes Eq. (8) to take the form:

$$y(n)=0.23x(n)+0.4x(n-1)+0.6x(n-2)+0.54x(n-3)+0.28x(n-4)+0.8x(n-5)-0.76x(n-6)+0.35x(n-7)-0.23x(n-8)+0.22x(n-9)-0.65x(n-1)x(n-3)+e(n-1) \quad (10)$$

Note that additive noise is statically added after the clean output signal has been generated. For dynamic noise, the GWN source, e(n), is fed back to the output so that the current and future output values are dependent on the past states of the input and noise signals. Therefore, the outputs described by Eq. (9) and (10) have different values. The SNR for additive noise was obtained for two different levels, 10 and 0dB. For the dynamic noise, the SNR was 3.5 dB. A model order of ARMA (10,10) and nonlinear ARMA (5,5) was incorrectly selected to determine the effectiveness of both approaches(the correct model is MA(10) and NMA(1,3). With incorrect selection of linear and nonlinear ARMA model orders, the numbers of parameters to be determined are 21 linear ARMA terms, and 57 nonlinear ARMA model terms, for a total of 78 terms to be searched. This is a daunting task for any algorithms, as Eq.(8) contained only 11 model terms, but we are subjecting the FOS and OPS to a superfluous model order search concomitant with excessive noise corruption in the data signal. The results of additive and dynamic noise for the OPS and FOS are shown in Table 6. As in the previous examples, with a clean signal, the OPS is again able to obtain accurate parameter estimates associated with only the true model terms. This result is impressive in itself, since we are not aware of any other algorithm that is able consistently to provide accurate parameter estimates despite incorrect model order selection even in the case of a clean signal. The FOS, however, is not able to obtain correct parameter estimates and has obtained incorrect model terms [y (n-1), y (n-2), and y (n-5)] and missed some of the true model terms [x (n-7), x (n-8) and x (n-9)]. With additive noise, either SNR=10 or 0dB, the OPS is accurate in providing only the true model terms but the FOS missed a model term [x(n-8), only for SNR=10dB] and it inaccurately introduced an additional term[y(n-2)]. Moreover, the estimated

coefficients with the OPS are closer to the true model coefficients than are those obtained with the FOS. The MSE of the OPS are 0.0044 and 0.0442 for SNR=10 and 0dB, respectively. The MSE for the FOS are 0.035 and 0.1038 for SNR=10 and 0dB respectively. IT is clear that despite significant noise in the data, and grossly incorrect model order selection, both methods perform rather nicely. However, it is clear that the OPS outperformed the FOS. The MSE values are approximately tree to ten fold less than those obtained via the FOS.

With dynamic noise, the result is the same as with additive noise; the OPS is more accurate than the FOS. The FOS is unable to identify the term x (n-9), and incorrectly introduced a y (n-2) term. The OPS, although the coefficients associated with true model coefficients deviate somewhat, is accurate in only producing coefficients related to the true model terms. The MSE also favors the OPS; the MSE values are 0.013 for the OPS and 0.056 for the FOS.

Table 6:Comparison of the OPS and FOS with additive and dynamic noise with an incorrect model order selection.

ModelTerms	$x(n)$	$x(n-1)$	$x(n-2)$	$x(n-3)$	$x(n-4)$	$x(n-5)$	$x(n-6)$	$x(n-7)$	$x(n-8)$	$x(n-9)$	$x(n-1)x(n-3)$	$y(n-1)$	$y(n-2)$	$y(n-5)$
TureValue	0.23	0.40	0.60	0.54	0.28	0.80	−0.76	0.35	−0.23	0.22	−0.65	0.0	0.0	0.0
CleanData														
OPS	0.23	0.40	0.60	0.54	0.28	0.80	−0.76	0.35	−0.23	0.22	−0.65	0.0	0.0	0.0
FOS	0.22	0.42	0.61	0.54	0.22	0.72	−0.73	0.00	0.00	0.00	−0.64	−0.14	0.22	0.06
AdditiveNoise:														
OPS(10dB)	0.23	0.41	0.62	0.55	0.30	0.78	−0.75	0.35	−0.27	0.25	−0.65	0.0	0.0	0.0
FOS(10dB)	0.23	0.40	0.59	0.49	0.20	0.69	−0.80	0.22	0.00	0.19	−0.65	0.0	0.17	0.0
OPS(0dB)	0.22	0.43	0.66	0.58	0.35	0.72	−0.74	0.35	−0.37	0.30	−0.65	0.0	0.0	0.0
FOS(0dB)	0.0	0.43	0.65	0.57	0.32	0.70	−0.76	0.31	−0.34	0.28	−0.65	0.0	0.05	0.0
DynamicNoise														
OPS(3.5dB)	0.18	0.45	0.66	0.54	0.29	0.79	−0.79	0.35	−0.26	0.17	−0.66	0.0	0.0	0.0
FOS(3.5dB)	0.18	0.45	0.65	0.52	0.29	0.79	−0.81	0.31	−0.22	0.00	−0.66	0.0	0.04	0.0

III. The effect of incorrect model order selection and additive noise on a ARMA model

The next simulation example considers the following linear ARMA model with Gaussian White noise (GWN) as the input, x(n), so that the output, y(n), contains 1000 data points:

$$y(n)=0.35y(n-1)+0.32y(n-2)+0.1y(n-3)-0.54x(n)+0.34x(n-1)+0.23x(n-2)+0.21x(n-3) \quad (11)$$

The objective is, based on only the measured input signal, x(n), and the output signal, y(n), to estimate the parameters of the above equation as accurately as possible. Although the true ARMA model order for the above process is 3 output lags and 3 input lags, we purposely selected an incorrect model order of 10 output lags (y(n-1),...,y(n-10)) and 10 input lags (x(n),x(n-1),..., x(n-10)) for both methods. The efficacy of the ARMA model relies on the algorithm's ability to find the true model order despite an incorrect a priori model order selection. Comparison of the results based on the two methods is shown in Table 3. The OPS algorithm obtained the correct model terms and coefficients but the FOS did not; not only did the FOS miss 3 true model terms [y(n-2),y(n-3) and x(n-2)], but the coefficients obtained for the true model terms deviated from the exact coefficients of the model. In addition, the FOS incorrectly provided three additional model terms, y(n-4),x(n-4) and x(n-5). The MSE of the OPS and the FOS are 0.0 and 4.17e -005, respectively. This is an example that indicates that the goodness of fit according to the MSE value is not always a good measure, since the MSE of 4.17e-005 obtained with the FOS is a very small error. Note that for chaotic systems modeling, even small differences in the obtained coefficients would result in an exponentially divergent change in the output value. As

detailed in the Introduction section, the FOS, because it relies on a sub optimal model order search, is not always able to obtain correct model terms and coefficients when it is subjected to an incorrect model order selection even for a clean signal. This has been demonstrated in this and the previous examples.

When Eq. (11) was corrupted with additive noise levels of 10 and 0dB, the OPS and the FOS provided comparable results. The simulations with additive noise were made further challenging with an a priori incorrect selection of 10 output and 10 input lags. The advantage of the OPS observed with only the incorrect model order selection (no noise added) disappeared with the introduction of additive noise as evidenced by Table 7. However, in terms of MSE values, the OPS performed slightly better than did the FOS. The MSE values for the OPS for 10 and 0dB were 0.08 and 0.74, respectively; for the FOS, the corresponding values were 0.09 and 0.88 respectively.

Table 7. Comparison of the OPS and FOS with an incorrect model order selection and two different levels of noise contamination for an ARMA model.

ModelTerms	$y(n-1)$	$y(n-2)$	$y(n-3)$	$y(n-4)$	$x(n)$	$x(n-1)$	$x(n-2)$	$x(n-3)$	$x(n-4)$	$x(n-5)$
TrueValue	−0.35	0.32	0.10	0.00	−0.54	0.34	0.23	0.21	0.00	0.00
OPS(clean)	−0.35	0.32	0.10	0.00	−0.54	0.34	0.23	0.21	0.00	0.00
FOS(clean)	−0.24	0.00	0.00	0.11	−0.54	0.40	0.00	0.34	0.03	0.05
OPS(10dB)	0.00	0.18	0.00	0.00	−0.54	0.54	−0.04	0.26	−0.10	0.08
FOS(10dB)	−0.13	0.12	0.00	0.09	−0.55	0.47	0.00	0.28	0.00	0.04
OPS(0dB)	0.00	0.09	0.00	0.00	−0.55	0.56	−0.10	0.28	−0.14	0.12
FOS(0dB)	0.00	0.09	0.00	0.00	−0.55	0.56	−0.10	0.28	−0.14	0.12

IV. Nonlinear ARMA (NARMA) model with incorrect model order selection and additive noise.

As an arbitrary example of a NARMA model, the following 1000 data point difference equation was generated:

$$y(n)=0.8x(n)-0.23x(n-1)+0.5x(n-3)+0.32y(n-1)+0.3y(n-2)-0.2x(n-1)^2+0.12y(n-2)^2-0.18x(n-1)y(n-1) \quad (12)$$

The input,x(n) terms, was generated using GWN. To completely test the features of the general NARMA model, Eq. (12) included self-nonlinear input and output terms as well as the cross-nonlinear term. The model order for the two methods compared was selected to be AR=10, MA=10, quadratic MA=5, quadratic AR=5 and cross nonlinear model order between the input and output=5 (total number of parameters searched was 78). Table 8 shows the results of the estimated coefficients obtained by the two methods for clean data and with two levels of additive noise. For the clean signal, as was the case with the three previous simulation examples, the OPS provided the correct terms and coefficients despite cincorrect model order selection; the FOS, however, shows the ill-effects of incorrect model order selection as evidenced by all of the coefficients deviating from the true coefficients. When the output of Eq. (12) was corrupted by either 10 or 0dB, the FOS provided more accurate coefficients than did the OPS. However, in terms of MSE, both methods provided similar error values of 0.4% for 10dB and 2% for 0dB.

Table 4. Comparison of the OPS and FOS with an incorrect model order selection and two diffent levels of noise contamination for a nonlinear ARMA model.

ModelTerms	$x(n)$	$x(n-1)$	$x(n-2)$	$x(n-3)$	$x(n-4)$	$x(n-5)$	$y(n-1)$	$y(n-2)$	$x(n-1)^3$	$y(n-2)^3$	$x(n-1)y(n-1)$	$y(n-2)y(n-3)$	$y(n-2)y(n-6)$
TrueValue:	0.8	-0.23	0.00	0.50	0.00	0.00	0.32	0.30	-0.20	0.12	-0.18	0.00	0.00
CleanData:	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
OPS:	0.8	-0.23	0.00	0.50	0.00	0.00	0.32	0.30	-0.20	0.12	-0.18	0.00	0.00
FOS:	0.79	0.14	0.00	0.51	0.00	0.00	0.25	0.32	-0.31	-0.06	0.14	0.00	0.00
AdditiveNoise	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
OPS(10*dB*)	0.81	0.00	0.00	0.59	0.18	0.00	0.00	0.30	-0.23	0.04	-0.16	0.04	0.00
FOS(10*dB*)	0.82	-0.16	0.00	0.51	0.00	0.00	0.27	0.26	-0.23	0.09	-0.14	0.00	0.00
OPS(0*dB*)	0.84	0.00	0.12	0.60	0.22	0.11	0.00	0.10	-0.36	0.00	0.00	0.03	0.00
FOS(0*dB*)	0.85	0.00	0.00	0.54	0.13	0.00	0.12	0.16	-0.29	0.00	-0.08	0.00	0.05

V. Application of the OPS to experimental data

In this section we demonstrate the use of the OPS in analyzing experimentally-obtained renal blood pressure and flow data. The aim is not be elucidate the physiological mechanisms involved in renal autoregulatory processes, but to examine if the OPS can provide similar impulse response functions(IRF) to those published and if those IRFs are at all physiologically meaningful[11-12].

VI. Data Acquisition and Experimental Procedure

The data analyzed in this investigation were obtained from a previously-published study[11]. Experimental methods are described in detail in [11] and will be briefly summarized. The experimental data were collected from normotensive Sprague-Dawley rats using broadband perturbations of the arterial presure(input) and measuring the resulting renal blood flow(output). Briefly, operating under halothane anesthesia, the aorta inferior to the renal arteries was cannulated with blood-filled tubing connected to a bellows pump which in turn was driven by a computer-controlled motor. Blood pressure was measured in the superior mesenteric artery with a standard pressure amplifier, and renal blood flow was mesured in the left renal artery with an electromagnetic flow probe.The input signal was chosen to be a constant-switching-pace sysmmetric random signal(CSRS) that exhibited the spectral properties of bandlimited white noise [12]. A unique seed was used for the random number generator in each experiment.

Each of the experimental data records used for analysis was 256 s long, with a sampling rate of two samples per second (Nyquist frequency of 1 Hz), after digital low-pass filtering to avoid aliasing. Each data record, containing 512 data points, was

subjected to second-degree polynomial trend removal(which included demeaning) and was normalized to unit variance.

Fig. 6 shows averaged impulse response functions (based on four recordings) computed from the ARMA coefficients obtained from analysis of the OPS[Fig. 6(a)] and from the FOS method[Fig. 6(b)]. For both the OPS and FOS, the model order of ARMA (10,10) was used, which was selected based on our previous work[13]. The dotted lines in the figure represent the standard deviation bounds of the sample mean. We observe that the estimated impulse response wave forms are rather consistent for both methods and are similar to those published[11,13]. However, the impulse response function obtained via the OPS exhibits smoother waveforms and smaller standard deviation bounds than does its counterpart obtained from the FOS. To compare the performance of the two methods quantitatively, model predictions based on the linear ARMA model were computed for both methods. The average MSE obtained the 4 datasets for the OPS and the FOS are 8.04% and 7.89%, respectively. As in the simulation examples presented in this paper, better model prediction is obtained with the OPS than with the FOS method.

It should be pointed out that the computational time for both methods is quite fast. The FOS uses a modified Cholesky decompositon to achieve orthogonality rather than inverting the matrix to solve for the least-squares estimation, econsequently, the computational speed is enhanced. However, the computational time is faster with the OPS than with the FOS since the orthogonal procedure is not utilized with the OPS.

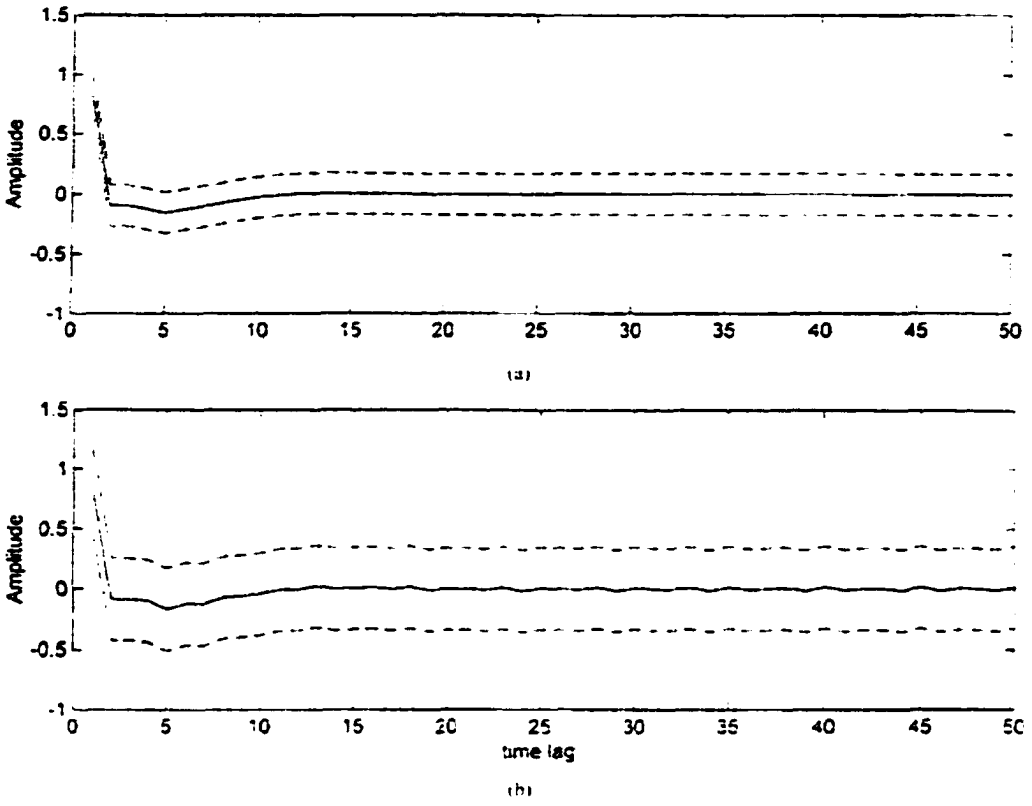


Fig 6. Averaged impulse response functions of renal blood flow and pressure data obtained via ARMA model for (a) the OPS and (b) the FOS.

3.4 Conclusion

Here we introduced a new algorithm, named the OPS. Simulation examples have shown the efficacy of the method and have shown that for clean signals, the OPS is able to extract only the correct parameters despite overdetermined incorrect model order selection. The FOS, one of the most accurate algorithms available, is also able to extract correct parameters under similar circumstances but its ability to obtain correct parameters for all noiseless data is not complete. The OPS, unlike the FOS, does not orthogonalize model terms, resulting in faster computational time. Due to the fact that the OPS achieves linear independence among candidate vectors, it is always able to obtain correct parameters despite a priori incorrect model order selection for noiseless signals. In other words, the OPS is an optimal search method, thus, it is able to obtain correct model parameters for all noiseless signals. To date, we are not aware of any other algorithm that is able to achieve this kind of remarkable results. For the case of noise contaminated linear and nonlinear MA models, the OPS provides performance superior to that of the FOS, as evidenced by the simulation examples considered in the previous content. For noise-contaminated linear and nonlinear ARMA models, both the OPS and FOS provide similar excellent results. In addition, application of the OPS to experimental data indicates the feasibility of the method in obtaining physiologically meaningful transfer function relationships between renal pressure and flow data.

Chapter 4

The MHD Method

4.1 Abstract

The method of least squares (LS) assumes that the best-fit curve is the one that provides the minimal sum of squared deviations from a given set of data. The LS coefficients of an autoregressive (AR) model are generally biased. They are unbiased only if the independent variable is noise free or only has driving noise in the linear system, which is an unrealistic expectation in normal settings. The total least squares (TLS) can be used to increase the accuracy of the LS because it specifically accounts for the fact that the independent variables are noise corrupted. TLS provides better estimates of AR coefficient estimates in the nonlinear models. However, it still cannot obtain the accurate parameters. To overcome the inherent limitations of both LS and TLS methods, we present a new method that is based on minimizing hypersurface distance. Computer simulation examples show that the new method proposed achieves more accurate parameter estimates than either the LS and TLS for all noise levels considered, and retains its accuracy even when the signal-to-noise ratio is as low as 0 dB.

4.2 Introduction

The least squares (LS) method and to a lesser extent, the total least square (TLS) method, are the preferred data fitting techniques for solving an over determined system. The LS estimation can be formulated to find a vector x such that:

$$e^2 = \min[(Ax-b)^2] \qquad (1)$$

the prediction error, e, is minimized. Eq. (1) can be rewritten: Ax=b+e. The solution of LS is $x=(A^TA)^{-1}A^Tb$. Thus, LS assumes that only the vector b and not the matrix A is perturbed by the noise source, e. The LS solution is most closely satisfied when A, the candidate matrix, is clean. If perturbations in A are allowed as well as those in b, then the estimation results will deviate from the true value and will be biased. This is called the error-in-variables problem [31]. The TLS partly solves this error-in-value problem and one can obtain better results than with LS. With the TLS, both the matrix A and the vector b are assumed to be corrupted by noise sources such that:

$$(A+r)x=b+e, \text{ where } b+r \in \Re(A+r) \qquad (2)$$

The standard procedure to solve the TLS is to use the singular value decomposition (SVD) of the matrix C=[A b]. With the smallest singular values obtained from the SVD of the matrix C, denoted as □, the solution of the TLS is estimated by performing the following: $x=(A^TA-\eta^2I)^{-1}A^Tb$.

There exists one major drawback of the TLS, especially for nonlinear models. Consider a simple nonlinear autoregressive model described by the following difference equation: $y(n)=a+by(n-1)+cy(n-1)^2$. The candidate matrix A and the vector b, for this system, are given by:

$$A=\begin{bmatrix} 1 & 0 & 0 \\ 1 & y(1) & y(1)^2 \\ \vdots & \vdots & \vdots \\ 1 & y(n-1) & y(n-1)^2 \end{bmatrix} \qquad b=\begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(n) \end{bmatrix} \qquad (3)$$

The output y is considered to be contaminated with noise. We assume that the signal-to-noise ratio (SNR) is some value designated as S. Thus, for the candidate matrix A, the first column is a constant value, the second column has SNR equal to S, and the third

column has SNR approximately equal to two times S. Despite the fact that the candidate matrix A has a different SNR for each column, TLS assumes that the SNR is the same for all columns of the candidate matrix A. A detailed explanation of the TLS approach is provided in Section II.

To improve the efficacy of the TLS, many approaches have been developed [27-31]. In this paper, we present a new method based on minimization of hypersurface distance (MHD) to overcome the limitations of both the LS and TLS methods. In section II, we introduce a geometrical interpretation of the LS and TLS methods, and in section III we provide several simulation examples demonstrating the efficacy of the proposed MHD method.

4.3 Methodology

This section provides a geometrical interpretation of the LS and TLS methods. Without loss of generality, we consider a 2-dimensional system. For a LS fit, a line is found that minimizes the total vertical distance from the measured point to the line, as shown in Fig. 7. The independent variable is not considered in fitting to the line. Thus, the LS involves minimizing the vertical distance to the line.

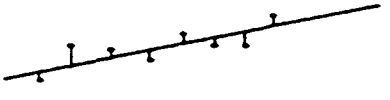


Fig. 7

For TLS, a line is found that minimizes the total distance from the points to the line, as shown in Fig. 8. The TLS, thus, involves minimizing the distance with regard to both the independent and dependent variables.

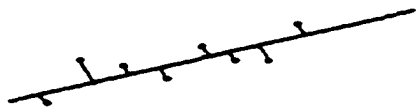


Fig. 8

Eq. (2) can be re-formulated as follows:

$$\min_{\Delta A,\Delta b}\|\Delta A \quad \Delta b\|_F \text{ such that } b+\Delta b \in \text{Range}(A+\Delta A) \tag{4}$$

where $\Delta A, \Delta b$ are perturbations of A and b respectively. To find a solution to the TLS, Eq. (2) is expanded in the homogeneous form:

$$\begin{bmatrix} A+\Delta A | b+\Delta b \end{bmatrix} \begin{bmatrix} X \\ -1 \end{bmatrix} = 0$$

$$\left(\begin{bmatrix} A|b \end{bmatrix} + \begin{bmatrix} \Delta A|\Delta b \end{bmatrix} \right) \begin{bmatrix} X \\ -1 \end{bmatrix} = 0 \tag{5}$$

For simplicity, we denote $[A|b]$ as C and $[\Delta A|\Delta b]$ as Δ. The TLS solutions finds the Δ with smallest norm that makes $C+\Delta$ rank deficient. The standard procedure to solve this TLS problem is to use the singular value decomposition (SVD) of the extended data matrix C such that:

$$C = U\,\text{diag}(\sigma_1, \sigma_2, \ldots, \sigma_{n+1}) V^H = \sum_{k=1}^{n+1} \sigma_k u_k v_k^H \tag{6}$$

Where $U = [u_1, u_2, \cdots, u_m]$, and $V = [v_1, v_2, \cdots, v_{n+1}]$.

It is assumed that σ_{n+1} is the smallest singular value. From this smallest singular value, σ_{n+1}, the TLS amounts to finding a vector X such that:

$$\frac{\left\| [A|b] \begin{bmatrix} X \\ -1 \end{bmatrix} \right\|}{\left\| \begin{bmatrix} X \\ -1 \end{bmatrix} \right\|} = \sigma_{n+1} \tag{7}$$

By squaring Eq. (7), we obtain the following:

$$\min_{x} \frac{\left\| [A|b] \begin{bmatrix} X \\ -1 \end{bmatrix} \right\|^2}{\left\| \begin{bmatrix} X \\ -1 \end{bmatrix} \right\|^2} = \min_{x} \sum_{i=1}^{m} \frac{\left| a_i^H x - b_i \right|^2}{x^H x + 1} \tag{8}$$

The right-hand side is the square of the distance from the point $[a_i,b_i]$ to the nearest point on the hyperplane. For further details refer to [32].

We consider a new cost function based on minimization of hypersurface distance (MHD). For linear systems, the methods of TLS and MHD have the identical geometrical interpretation. However, differences between the TLS and MHD methods occur in the case of nonlinear systems. In general, the MHD method provides higher accuracy than the TLS especially for nonlinear systems at the expense of higher computational complexity. An example delineating differences between the TLS and MHD can be examined using the following nonlinear difference equation:

$$y(n) = ay(n-1) + by(n-2)^2$$

where y(n) is the output signal that is assumed to be corrupted with Gaussian white noise (GWN). The TLS determines a plane Z=aX+bY (here Z, X and Y correspond to the y(n), y(n-1), and $y(n-1)^2$, respectively with n=1,2...,N) such that the summation of the distance for N data points of those three coordinates to the plane will be minimized. Different from the TLS method, MHD method's criterion is to find a surface $Z = aX + bY^2$ such that the summation of the distance for N data points of the three coordinates to the surface is minimized. The geometrical interpretation of the cost function of the MHD

Fig. 9

is provided in Fig. 9.

Similarity of the MHD to the TLS lies in determining the minimal total distance to the surface rather than a plane, as is done with the TLS (Fig. 8). The structure of the surface is case-dependent, and thus, there is no general solution to find the minimal surface. Furthermore, for highly complex nonlinear structure, an analytical solution may not be obtained.

We define details regarding the implementation of the MHD method. There are two separate procedures that are performed prior to the MHD. These two procedures are mainly used to minimize the computational time involved with the MHD. The first involves utilizing our previously developed model order search algorithm so that the number of difference model terms to be searched is minimized. This method is called the optimal parameter search (OPS) algorithm and has been shown to be one of the most accurate methods available to date [33]. The second involves using the TLS to estimate parameter estimation of the model terms determined using the OPS algorithm. The purpose of using the TLS is to obtain an initial better parameter estimate such that the search space for the optimal parameter estimate can be minimized.

Once the significant model terms and their associated initial parameter estimates are found, the MHD method is then applied. To provide an example of the MHD method, consider the following nonlinear autoregressive difference equation:

$$y(n) = ay(n-1) + by(n-2)^2$$

With z=y(n), x=y(n-1), y=y(n-2), the above equation can be rewritten such that:

$$z = ax + by^2 \tag{9}$$

For optimal values of a and b, the minimum total distance to the surface can be found. For simplicity, we use the norm 2 distance to minimize the following cost function:

$$\min[(z-z_1)^2+(x-x_1)^2+(y-y_1)^2] \tag{10}$$

By substituting z with Eq. (9) to Eq. (10), we find the following:

$$f(x,y)=(ax+by^2-z_1)^2+(x-x_1)^2+(y-y_1)^2\frac{\partial f(x,y)}{\partial x}=0$$

$$(ax+by^2-z_1)a+(x-x_1)=0 \tag{11}$$

$$\frac{\partial f(x,y)}{\partial y}=0$$

$$(ax+by^2-z_1)2by+(y-y_1)=0 \tag{12}$$

Initially, we set a range of values for the parameters a and b such that Eqs. (11-12) are minimized. For every values of a and b Eqs. (11-12) are evaluated, and the combination of a and b that minimizes the total distance to the surface is determined.

If the dimension of the system is less than or equal to three, the analytical solution that characterizes the minimum distance can be obtained. If the dimension of the system is higher than three, an analytical solution of the minimum distance may not be obtainable. The MHD method is case-dependent and a unique solution is difficult to obtain especially for systems with dimension higher than three. However, it has been suggested that the spherical polar coordinates method can be used to obtain an analytical solution for systems with dimension higher than three [34].

4.4 Simulation Results

In this section we demonstrate the effectiveness of the MHD method for estimating the parameters of nonlinear AR models and compare its' performance to those of LS and TLS methods. For the first simulation example, we consider a well-known self-

oscillating chaotic dynamic known as Henon map, and it is described by the following nonlinear difference equation: $y(n+1)=1-1.4y(n)^2+0.3y(n-1)$. For the next two simulation examples to follow we assume that the structure of the model is known. The task is then to estimate the coefficients of the known model when it is corrupted by 10 dB GWN. The phase plot of the Henon map is shown in Fig. 10. Note that 10 dB of noise completely obliterate the dynamics of the structure (not shown).

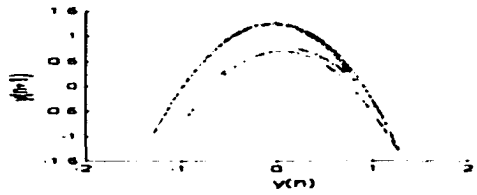
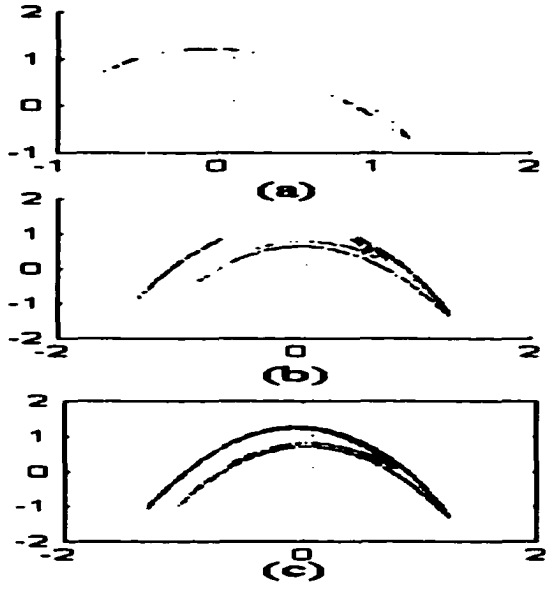


Fig. 10 True attractor

Table 9

	parameter a	parameter b
True Value	-1.4	0.3
LS	-1.1125	0.2404
TLS	-1.2980	0.3465
MHD	-1.42	0.30

Comparison of the coefficient estimates based on the three methods is shown in Table 9. It is clearly evident that the MHD method provides the most accurate parameter estimates of the three. Despite 10 dB noise, the MHD method is able to obtain the correct model parameters. With the other two methods, the coefficient estimates deviated significantly from the true model coefficients. The TLS provides better coefficient estimates than does the LS, and the TLS results themselves are impressive considering



(a). Fig. 11-1 LS, (b). Fig. 11-2 TLS, (c). Fig. 11-3 MHD

the significant noise contamination. Figs. 5-1 to 5-3 show attractors reproduced from the estimated coefficients obtained by the LS, TLS and MHD methods, respectively, as provided in Table 1. We observe that the MHD is the only method that is able to accurately reproduce the true phase dynamics of Henon map.

The second simulation considers another deterministically chaotic attractor that has the following nonlinear difference equation:

$$y(n)=0.7y(n-1)-0.2y(n-1)^2-1.5y(n-2)+0.4y(n-2)^2 \quad (10)$$

Fig. 12 shows the phase plot of the attractor described by Eq. (10).

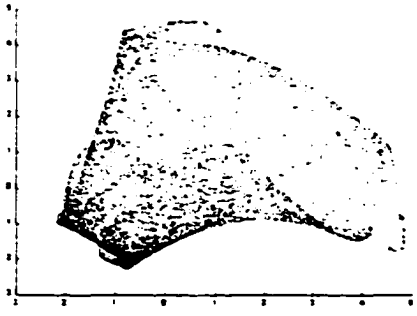
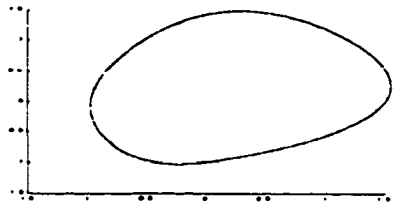
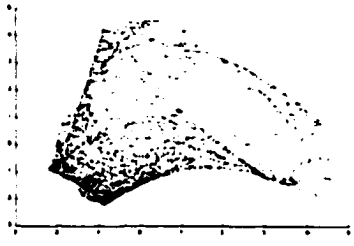


Fig. 12 True attractor



(a)

(b)



(c)

(a). Fig. 13- 1 LS (b). Fig. 13-2 TLS (c). Fig. 13-3 MHD

Table 10

	y(n-1)	$y(n-1)^2$	y(n-2)	$y(n-2)^2$
True Value	0.7	-0.2	-1.5	0.4
LS	0.403	-0.101	-1.054	0.245
TLS	0.614	-0.157	-1.394	0.328
MHD	0.694	-0.196	-1.510	0.396

As in the first simulation example, 10 dB GWN was added to the data. Table 10 shows the comparison of the parameters estimates of the LS, TLS and MHD methods. Figs. 13-1, 13-2 and 13-3 show reconstructed attractors of the LS, TLS and MHD methods, respectively. As shown in Table 2 and the reproduced attractors, unequivocally the MHD method produced the best result. Furthermore, the result obtained by the MHD method is impressive in that it is able to reproduce the attractor despite the fact that significant noise was added to the system. The attractor obtained from the LS is completely different from

the true attractor shown in Fig. 11. The TLS estimated attractor shows overall characteristics of the true attractor but several folding patterns that are not representative in the true attractor are revealed.

Besides white noise, we also consider the 10dB color noise.

$$y(n)=0.7y(n-1)-0.2y(n-1)^2-1.5y(n-2)+0.4y(n-2)^2+e(n)-0.21e(n-1) \quad (11)$$

Table 11.

	y(n-1)	$y(n-1)^2$	y(n-2)	$y(n-2)^2$
True Value	0.7	-0.2	-1.5	0.4
LS	0.344	-0.089	-1.030	0.239
TLS	0.557	-0.147	-1.406	0.331
MHD	0.702	-0.206	-1.524	0.398

From the table 11, again we can prove the robustness of our cost function.

The third simulation considers another chaotic attractor generated by the following difference equation:

$$y(n)=-0.5y(n-1)-y(n-1)*y(n-1)-0.65y(n-2)+0.5y(n-3) \quad (12)$$

Contrary to the two previous examples, in this example, we assume that the model structure is not known *a priori*. As detailed in the Methods section, to determine the correct model structure, we use the method of OPS. Once the model structure is determined, TLS is then used to estimate an initial estimate of the parameters that are associated with the model terms. To reiterate, we use the TLS method since it provides more accurate estimate of the parameters than the LS approach, as detailed in the Methods section. Note that if we used some of the conventional model order criteria (e.g., Akaike Information criteria (AIC) or minimum description length (MDL)), the accuracy

and computational efficiency is significantly compromised. This is because many of the model order criteria do not work well in practice.

Table 12

10dB	A:(-0.5)	B: (-1)	C: (-0.65)	D: (0.5)
LS	-0.2702±0.0155	-0.6365±0.0125	-0.5245±0.0034	0.5558±0.0076
TLS	-0.6438±0.0523	-1.0529±0.0403	-0.7020±0.0144	0.4173±0.0218
MHD	-0.5628±0.0134	-1.0400±0.0200	-0.6750±0.0071	0.4650±0.0112

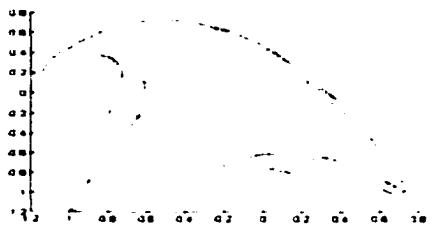


Fig. 14

To challenge the robustness of the algorithm, we add 10 independent GWN to the system output. Table 12 shows the mean and the standard deviation values for LS, TLS and MHD methods. As was the case with the previous two examples, the MHD method provides more accurate parameter estimates than either the LS and TLS methods. Figs. 15-1 to 15-3 show reproduced attractor based on the parameter estimates obtained by the LS, TLS and the MHD methods. Only the MHD method is able to provide attractor that closely resembles the true attractor as shown in Fig. 14. Note that the LS method's attractor significantly deviates from the true attractor.

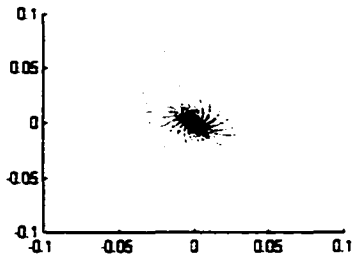


Fig.15-1

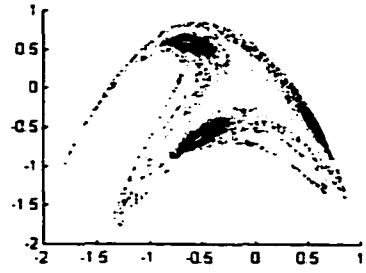


Fig. 15-2

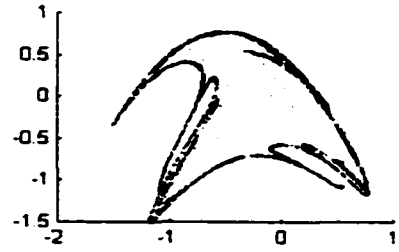


Fig. 15-3

4.5. Conclusion

We presented a new cost function based on minimization of hypersurface distance for accurate nonlinear AR parameter estimation. The technique developed is more computationally intensive than either the LS or TLS, but provided that the correct structure of the model is available, the MHD method can provide accurate nonlinear AR parameter estimates even in the case of significant noise contamination. The efficacy of the method has been shown in nonlinear AR models but the method can be extended to any subclasses of linear and nonlinear autoregressive moving average models. It should be noted that accurate parameter estimates could be obtained by the MHD method because it combines both robust model order search criterion as well as accurate initial estimate of the parameter estimates using the TLS. Combination of these two procedures subsequently leads to an accurate determination of the parameter estimates concomitant with significant reduction in the computational time.

Chapter 5

Proposed Work

We propose to investigate the following problems in the next phase of our work.

I. Chaotic analysis has been shown to be a powerful tool in understanding and characterizing the dynamics of nonlinear systems. We will apply newly- developed NAR model algorithm(OPS) and parameter estimation algorithm(MHD) to some of the well known chaotic models such as Henon, logistic, Rossler and Lorenz attractors. The Lyapunov exponent is a very useful indicator that determines the system's sensitivity dependence on the initial conditions. Estimation of the Lyapunov exponent requires a large amount of data. However, this can be avoided if a nonlinear ARMA model is used (Lyapunov exponent can be directly estimated based on the Jacobian matrix). In addition, since the NAR model also depicts the internel features of the chaotic system, and if the fact that our newly developed NAR algoirthm is accurate, we may be able to pinpoint a few parameters that maybe responsible for a system to undergo a bifurcation to chaos.

II. Data compression scheme is a very critical component in data storage and transmission. An ARMA model can be used as an effective tool to compress significant amount of data. We will examine if combination of our newly developed ARMA model

and other efficient algorithms such as wavelet transform can be used to further enhance the compressibility of the data.

III. It has been noted that growth hormone (GH) secretion is different between sexes. For example, GH secretion in the conscious adult made rat is highly organized into a series of episodic bursts whereas female rats secrete GH continuously. We will examine using our newly developed methods to differentiate underlying dynamics that may be responsible for differences in GH secretion process between the tow sexes.

REFERENCES:

[1] Akaike, H. Power spectrum estimation through autoregressive model fitting. Ann. Instrum. Stat. Math., 21:407-419, 1969.

[2] Chon, K.H., and R.J. Cohen. Linear and nonlinear ARMA model parameter estimation using artificial neural networks. IEEE Trans. Biomed. Eng. 44:168-174, 1997

[3] Chon, K.H., M.J. Korenberg, and N.H. Holstein-Rathlou. Application of fast orthogonal search to linear and nonlinear stochastic systems. Ann. Biomed. Eng. 25:793-801, 1997

[4] Chon, K.H., R.J. Cohen, and N.H. Holstein-Rathlou. Compact and accurate linear and nonlinear autoregressive moving average model parameter estimation using Laguerre functions. Ann. Biomed. Eng. 25:731-738, 1997

[5] Farlow, S.J. The GMDH algorithm of Ivakhnenko. The Am. Statistician, 35:210-215, 1981

[6] Ivakhnenko, A.G., Polynomial theory of complex systems. IEEE Tran. System, Man, Cybernetics, 1:364-378, 1971.

[7] Ivakhnenko, A.G., Group method of data handling-A revival of the method of stochastic approximation. Soviet Automatic Control, 13:43-71, 1966

[8] Ivakhnenko, A.G., G.A. Ivakhnenko, J.A. Muller, Self-organization of neuronets with active neurons. Pattern Recongnition and Image Analysis, 4:177-188, 1994

[9] Korenberg, M.J., S.A. Billings, Y.P. Li and P.J. McIlroy. Orthogonal parameter estimation algorithm for nonlinear stochastic systems. Int. J. Control, 48:193-210, 1988.

[10] Korenberg, M.J. A robust orthogonal algorithm for system identification and time series analysis. Biol. Cyb. 60:267-276, 1989.

[11] A.A. Abdel-Malek, C.H. Markham, P.Z. Marmarelis and V.Z. Marmarelis, Quantifying deficiencies associated with Parkinson's disease by use of time series analysis J. EEG Clinical Neurophys., vol. 69, pp. 24-33, 1988.

[12] H. Akaike, Power spectrum estimation through autoregression model fitting Ann. Inst. Stat. Math, vol. 21, pp. 407-419, 1969.

[13] J. Rissanen, A universal prior for the integers and estimation by minimum description length. Ann. Stat., vol. 11, pp. 417-431, 1983.

[14] T. Soderstrom and P.Stoica, System Identification. Englewood Cliffs, NJ: Prentice Hall, 1989

[15] K.H. Chon and S. Lu, A new algorithm for ARMA model parameter estimation using group method of data handling, In Press Ann. Biomed. Eng., 2000

[16] N.H. Holstein-Rathlou, A.J. Wagner, and D.J. Marsh, Tubuloglomerular feedback dynamics and renal blood flow autoregulation in rats, Am. J. Physiol., vol. 260, pp. F53-F68, 1991.

[17] V.Z. Marmarelis, A family of quasiwhite random signals and its optimum use in biological system identification. Part I: Theory, Biol. Cybern. , vol. 27, pp. 49-56, 1977

[18] K.H. Chon, Y.M. Chen, N.H. Holstein-Rathlou, D.J. Marsh, and V.Z. Marmarelis, On the efficacy of linear system analysis of renal autoregulation in rats, IEEE Trans. Biomed. Eng. Vol. 40, pp.8-20, 1993.

[19] GB Giannakis and E. Serpedin, Blind Identification of ARMA Models with Periodically
Modulated Inputs, IEEE Transactions on Signal Processing, pp. 3099-3104

[20] Linda M. Davis, lain B. Collings, and Robin J. Evans. Coupled estimators for equalization of mobile fast-fading channels. *IEEE Trans. on Communications*, 46(10):1262-1265, 1998.

[21] T. Soderstrom, J. Jezek, and V. Kucera. An efficient and versatile algorithm for computing the covariance function of an ARMA process. IEEE Transactions on Signal Processing, 46:1591-1600, 1998.

[22] W.B. Mikhael, A.S. Spanias, and F.H. Wu, ARMA Modeling Using the Linear Predictor, International Conference on Acoustics Speech and Signal Processing (ICASSP-85), Conf. Proc. ICASSP-85, Vol. 4, pp. 1505-1508, Tampa Bay, Florida, April 1985.

[23] Kan Deng, Andrew W. Moore, and Michael C. Nechyba. Learning to Recognize Time Series: Combining ARMA models with memory-based learning. 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA '97)

[24] Jiann-Fuh Chen, w. M. Wang and c. M. Huang, Analysis of Adaptive Time-series ARMA Model for Short-term Load Forecasting, EPSR, Vol.34, 1995, pp.187-196.

[25] Hodgson, Douglas J. Tse, Yee-man. Adaptive Estimation of Cointegrating Regressions with ARMA Errors, Exact maximum likelihood estimation of vector ARMA processes.

MSocSci-Dept of Economics & Statistics, NUS, 1985.

[26] A.P. Accardo, S. Vlahinic, S. Pensiero. Evaluation of the Characteristics of Smooth Pursuit Eye Movements to Sinusoidal Stimulation by Means of an ARMA Model VIII Mediterranean Conferenceon Medical and Biological Engineering and Computing.

[27] V. Krishnamurthy and G. Yin, "Recursive Algorithms for Estimation of Hidden Markov Models and Autoregressive Models with Markov Regime", IEEE Transactions on Information Theory, vol. 48, pp. 458-476.

[28] B.S. Chen, J.M Chen and S.C Shern, "An ARMA Robust System Identification Using a Generalized l/sub p/ norm Estimation Algorithm," IEEE Trans. Signal processing. vol. 42, pp. 1063-1073, 1994.

[29] A. Yeredor, "The Extended Least Squares Criterion: Minimization Algorithms and Applications, " IEEE Trans. Signal processing, vol. 49, pp. 74-86, 2001

[30] S. Chandrasekaran, M.Gu, A.H. Sayed, K.E.Schubert , "The Degenerate Bounded Errors-In-Variables Model," SIAM J. Matrix Anal. APPL. vol. 23, pp. 138-166, 2001.

[31] L. Jaeger and H. Kantz, "Unbiased reconstruction of the dynamics underlying a noisy chaotic time series," *CHAOS* vol. 6, pp. 440-450, 1996.

[32] T.K. Moon, and W.C. Stirling, Mathematical Methods and Algorithms for Signal Processing, Upper Saddle River, NJ, Prentice Hall, 2000.

[33] S. Lu, K.H. Ju, K.H. Chon "A new algorithm for linear and nonlinear ARMA model parameter estimation using affine geometry," IEEE Trans. Biomd.l Eng., vol 48, pp. 1116-1124, 2001.

[34]. G. Arfken, Spherical Polar Coordinates, Mathematical Methods for Physicists, 3rd ed. Orlando, FL: Academic Press, pp. 102-111, 1985.

[35]. Zigel, Y., Cohen, A. and Kats, A."ECG Signal Compression Using Analysis by Synthesis Coding", *IEEE Trans. on Biomedical Eng.*, vol. 47, pp. 1308-1316, Oct. 2000.