# FAST ERROR WHITENING ALGORITHMS FOR SYSTEM IDENTIFICATION AND CONTROL

Yadunandana N. Rao, Deniz Erdogmus, Geetha Y. Rao, Jose C. Principe
Computational NeuroEngineering Laboratory, University of Florida
Gainesville, FL 32611-6130
E-mail: {yadu, deniz, geethak, principe}@cnel.ufl.edu

**Abstract. Linear system identification with noisy inputs is a critical problem in signal processing and control. Conventional techniques based on the Mean Squared-Error (MSE) criterion can at best provide a biased estimate of the unknown system being modeled. Recently, we proposed a new criterion called the Error Whitening Criterion (EWC) to solve the problem of linear parameter estimation in the presence of additive white noise. In this paper, we present a fixed-point type algorithm with $O(N^2)$ complexity for EWC, called the Recursive Error Whitening (REW) algorithm. We will also show that the EWC solution can be solved by using the computational principles of Total Least-Squares (TLS). A novel EWC-TLS algorithm with $O(N^2)$ complexity is derived. We will then apply the EWC methods for adaptive inverse control and show the superiority over existing methods.**

## INTRODUCTION

Mean-squared Error (MSE) has been around for many years and has been widely applied for a variety of signal processing and control applications [1], [2]. Inverse control and system identification are some of the key applications in automatic control where MSE plays a vital role. System identification is the problem of estimating the parameters of an unknown system using the observed input and output sequences [3]. The objective of inverse control is to design a controller that would work in tandem with the actual system to produce a desired reference output [2]. The existence of cost effective and efficient algorithms like the stochastic Least Mean-Squares (LMS) [4] and the Recursive Least Squares (RLS) [1] has benefited the extensive application of the MSE criterion for system identification and control. However, in the presence of additive disturbances (both correlated and white) on the input and the output signals of interest, MSE can at best provide a biased solution. Noise-free data are seldom available for many real-world applications. Depending on the type of application, several noise enhancement methods exist in literature. Subspace Wiener filtering [1] is a powerful data conditioning technique based on Principal Subspace Analysis (PSA). However, this solution is too expensive and suffers from the curse of dimensionality. Total Least Squares (TLS) on the other hand, can provide us with bias-free solutions in the presence of additive white noise [5]. But, TLS requires the variances of the disturbances on the input and output data to be the same. If this criterion is not met, TLS can give biased solutions [6]. The Instrumental Variables (IV) method proposed as an extension to the Least-Squares (LS) has been previously applied for parameter estimation in white noise [3]. This method requires choosing a set of *instruments* that are uncorrelated with the noise in the input. Recently, we proposed a novel criterion called the *Error Whitening Criterion* (EWC), which can produce unbiased parameter estimates of a linear system in the presence of additive white noise [7,8]. Instead of minimizing the mean-squared error, the EWC formulation enforces zero autocorrelation of the error signal beyond a certain lag, and hence the name *Error Whitening Criterion* (EWC). As a consequence, the IV method mentioned before appears as a special case of EWC. In the next section, we will briefly summarize the criterion using a system identification framework and then propose a recursive adaptive technique for EWC called Recursive Error Whitening (REW) algorithm.

## ERROR WHITENING CRITERION IN A SYSTEM ID FRAMEWORK

Consider the problem of identifying a linear system characterized by the parameter vector $\mathbf{w}_T \in \Re^N$ as shown in fig 1. Let $(\mathbf{x}_k, d_k)$ denote the actual input and output of the system. Further, we will model the measurement errors and system disturbances by uncorrelated additive white noise sequences $u_k$ and $v_k$ (with unknown variances) that appear at the output and input of the system respectively. The problem of system identification can now be stated as follows: Given the noisy data pair $(\hat{\mathbf{x}}_k, \hat{d}_k)$ where $\hat{\mathbf{x}}_k \in \Re^N = \mathbf{x}_k + \mathbf{v}_k$ and $\hat{d}_k \in \Re^1 = d_k + u_k$, we have to determine the parameter vector $\mathbf{w} \in \Re^M$ that best describes the underlying system. Without loss of generality, we will assume that the length of $\mathbf{w}$ is at least $N$, the number of parameters in the actual system or $M \geq N$. Since $d_k = \mathbf{x}_k^T \mathbf{w}_T$, the error is $\hat{e}_k = \mathbf{x}_k^T(\mathbf{w}_T - \mathbf{w}) + u_k - \mathbf{v}_k{}^T \mathbf{w}$. Defining a vector $\boldsymbol{\varepsilon} = \mathbf{w}_T - \mathbf{w}$, the error autocorrelation at some arbitrary lag $L$ is given by,

$$\rho_{\hat{e}}(L) = \boldsymbol{\varepsilon}^T E[\mathbf{x}_k \mathbf{x}_{k-L}^T]\boldsymbol{\varepsilon} + \mathbf{w}^T E[\mathbf{v}_k \mathbf{v}_{k-L}^T]\mathbf{w} \tag{1}$$

If the chosen lag $L \geq M$, it is obvious that $E[\mathbf{v}_k \mathbf{v}_{k-L}^T] = \mathbf{0}$. Also, if the matrix

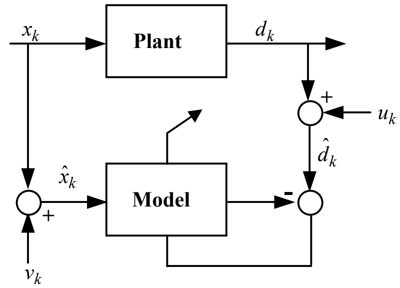


Figure 1. System Identification block diagram

$E[\mathbf{x}_k \mathbf{x}_{k-L}^T]$ is full rank, $\rho_{\hat{e}}(L) = 0$ only when $\mathbf{w} = \mathbf{w}_T$ [7,8]. Therefore, if we make the error autocorrelation at any lag $L \geq M$ zero, then the estimated weight vector will be exactly equal to the true weight vector. In other words, the criterion tries to *whiten* the error signal for lags greater than or equal to the adaptive filter length, i.e., $\rho_{\hat{e}}(L) = 0$ for $L \geq M$ and hence the name *Error Whitening Criterion*. Defining $\dot{\hat{e}}_k = (\hat{e}_k - \hat{e}_{k-L})$, (1) can be rewritten in a convenient form as [7,8],

$$J(\mathbf{w}) = E(\hat{e}_k^2) + \beta E(\dot{\hat{e}}_k^2) \tag{2}$$

where, $\beta$ is a constant. It is easy to see that when $\beta = -0.5$, (2) reduces to the error autocorrelation $\rho_{\hat{e}}(L)$. The goal is to find the weight vector $\mathbf{w}$ that would make $J(\mathbf{w}) = 0$ with $\beta = -0.5$. Note that when $\beta = 0$, (2) reduces to the MSE cost function. We will now derive the Recursive Error Whitening (REW) algorithm that optimally estimates the stationary point of the EWC cost function.

## RECURSIVE ERROR WHITENING (REW) ALGORITHM

We will begin this section by defining some matrices that will be used throughout the rest of the paper. Define input correlation matrices as $\mathbf{R} = E[\mathbf{x}_k \mathbf{x}_k^T]$, $\hat{\mathbf{R}} = E[\hat{\mathbf{x}}_k \hat{\mathbf{x}}_k^T]$, $\mathbf{R}_L = E[\mathbf{x}_{k-L}\mathbf{x}_k^T + \mathbf{x}_k \mathbf{x}_{k-L}^T]$, and $\hat{\mathbf{R}}_L = E[\hat{\mathbf{x}}_{k-L}\hat{\mathbf{x}}_k^T + \hat{\mathbf{x}}_k \hat{\mathbf{x}}_{k-L}^T]$ for noise-free and noisy signals (denoted by capped variables). Further, the input noise vector autocorrelation matrices are $\mathbf{V} = E[\mathbf{v}_k \mathbf{v}_k^T]$ and $\mathbf{V}_L = E[\mathbf{v}_{k-L}\mathbf{v}_k^T + \mathbf{v}_k \mathbf{v}_{k-L}^T]$. Additionally, we will define the matrices $\mathbf{S} = E[\dot{\mathbf{x}}_k \dot{\mathbf{x}}_k^T]$ and $\hat{\mathbf{S}} = E[\dot{\hat{\mathbf{x}}}_k \dot{\hat{\mathbf{x}}}_k^T]$. The dot is used to symbolize difference between the current and $L^{\text{th}}$ previous sample vector/scalar, for example, $\dot{\mathbf{x}}_k = \mathbf{x}_k - \mathbf{x}_{k-L}$. We will also define cross-correlation vectors between the input vector and the desired signal as $\mathbf{P} = E[\mathbf{x}_k d_k]$, $\hat{\mathbf{P}} = E[\hat{\mathbf{x}}_k \hat{d}_k]$, $\mathbf{P}_L = E[\mathbf{x}_{k-L} d_k + \mathbf{x}_k d_{k-L}]$, and $\hat{\mathbf{P}}_L = E[\hat{\mathbf{x}}_{k-L}\hat{d}_k + \hat{\mathbf{x}}_k \hat{d}_{k-L}]$ for both noise-free and noisy data. Also, we will define vectors $\mathbf{Q} = E[\dot{\mathbf{x}}_k \dot{d}_k]$ and $\hat{\mathbf{Q}} = E[\dot{\hat{\mathbf{x}}}_k \dot{\hat{d}}_k]$. Using the above definitions, we can rewrite $J(\mathbf{w})$ in (2) as,

$$J(\mathbf{w}) = E[d_k^2 + \beta \dot{d}_k^2] + \mathbf{w}^T(\mathbf{R} + \beta \mathbf{S})\mathbf{w} - 2(\mathbf{P} + \beta \mathbf{Q})^T \mathbf{w} \tag{3}$$

The above equation can be easily derived by substituting $e_k = \hat{d}_k - \hat{\mathbf{x}}_k^T \mathbf{w}$ and $\dot{\hat{e}}_k = \dot{\hat{d}}_k - \dot{\hat{\mathbf{x}}}_k^T \mathbf{w}$ in (2). Taking the gradient with respect to $\mathbf{w}$ and equating to zero, we get,

$$\frac{\partial J(\mathbf{w})}{\partial \mathbf{w}} = 2(\hat{\mathbf{R}} + \beta \hat{\mathbf{S}})\mathbf{w} - 2(\hat{\mathbf{P}} + \beta \hat{\mathbf{Q}}) = \mathbf{0} \tag{4}$$

Then the optimal weight vector is given by,

$$\mathbf{w}_* = (\hat{\mathbf{R}} + \beta \hat{\mathbf{S}})^{-1}(\hat{\mathbf{P}} + \beta \hat{\mathbf{Q}}) \tag{5}$$

When $\beta = 0$, (5) reduces to the minimum MSE solution. Simple calculations show that $\hat{\mathbf{S}} = 2\hat{\mathbf{R}} + \hat{\mathbf{R}}_L$ and $\hat{\mathbf{Q}} = 2\hat{\mathbf{P}} + \hat{\mathbf{P}}_L$. Also, the noisy correlation matrices are related to the noise-free signal and noise correlation matrices through the following set of equations.

$$
\begin{aligned}
&\hat{\mathbf{R}} = E[\hat{\mathbf{x}}_k \hat{\mathbf{x}}_k^T] = \mathbf{R} + \mathbf{V} \\
&\hat{\mathbf{S}} = E[(\hat{\mathbf{x}}_k - \hat{\mathbf{x}}_{k-L})(\hat{\mathbf{x}}_k - \hat{\mathbf{x}}_{k-L})^T] = 2(\mathbf{R} + \mathbf{V}) - \mathbf{R}_L - \mathbf{V}_L \\
&\hat{\mathbf{P}} = E[\hat{\mathbf{x}}_k \hat{d}_k] = \mathbf{P} \\
&\hat{\mathbf{Q}} = E[(\hat{\mathbf{x}}_k - \hat{\mathbf{x}}_{k-L})(\hat{d}_k - \hat{d}_{k-L})] = 2\mathbf{P} - \mathbf{P}_L
\end{aligned}
\tag{6}
$$

Using (6), the optimal weight vector can be simplified as,

$$\mathbf{w}_* = \left[(1+2\beta)(\mathbf{R}+\mathbf{V}) - \beta(\mathbf{R}_L + \mathbf{V}_L)\right]^{-1}\left[(1+2\beta)\mathbf{P} - \beta\mathbf{P}_L\right] \tag{7}$$

For $L \geq M$ and $\beta = -0.5$, we immediately see that all the noise matrices in the above equation cancel out and the optimal solution reduces to $\mathbf{w}_* = \mathbf{R}_L^{-1}\mathbf{P}_L$ which is nothing but the true weight vector $\mathbf{w}_T$. For the sake of notational simplicity, we will consider the noise-free case to derive the Recursive Error Whitening (REW) algorithm. With $\mathbf{Z}_k = \mathbf{R}_k + \beta\mathbf{S}_k$ and $\boldsymbol{\theta}_k = \mathbf{P}_k + \beta\mathbf{Q}_k$, a recursive relation for $\mathbf{Z}_k$ can be easily derived as,

$$\mathbf{Z}_k = \mathbf{Z}_{k-1} + (2\beta\mathbf{x}_k - \beta\mathbf{x}_{k-L})\mathbf{x}_k^T + \mathbf{x}_k(\mathbf{x}_k - \beta\mathbf{x}_{k-L})^T \tag{8}$$

Recall the Sherman-Morrison-Woodbury identity, also known as the matrix inversion lemma [9].

$$(\mathbf{A} + \mathbf{BCD}^T)^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{B}(\mathbf{C}^{-1} + \mathbf{D}^T\mathbf{A}^{-1}\mathbf{B})^{-1}\mathbf{D}^T\mathbf{A}^{-1} \tag{9}$$

Define $\mathbf{A} = \mathbf{Z}_k$, $\mathbf{B} = [2\beta\mathbf{x}_k - \beta\mathbf{x}_{k-L} \quad \mathbf{x}_k]$, $\mathbf{C} = \mathbf{I}_{2x2}$, a 2x2 identity matrix, and $\mathbf{D} = [\mathbf{x}_k \quad (\mathbf{x}_k - \beta\mathbf{x}_{k-L})]$. Then (8) reduces to,

$$\mathbf{Z}_k^{-1} = \mathbf{Z}_{k-1}^{-1} - \mathbf{Z}_{k-1}^{-1}\mathbf{B}(\mathbf{I}_{2x2} + \mathbf{D}^T\mathbf{Z}_{k-1}^{-1}\mathbf{B})^{-1}\mathbf{D}^T\mathbf{Z}_{k-1}^{-1} \tag{10}$$

Notice that this recursion for the inverse of $\mathbf{Z}_k$ is different than the conventional RLS algorithm. It requires the inversion of a 2x2 matrix $(\mathbf{I}_{2x2} + \mathbf{D}^T\mathbf{Z}_{k-1}^{-1}\mathbf{B})^{-1}$, which is still trivial. With this, we are able to reduce the complexity of inverting a sum of two matrices from $O(N^3)$ to $O(N^2)$. The recursive estimator for $\boldsymbol{\theta}_k$ is much simpler and can be expressed as,

$$\boldsymbol{\theta}_k = \boldsymbol{\theta}_{k-1} + [(1+2\beta)d_k\mathbf{x}_k - \beta d_k\mathbf{x}_{k-L} - \beta d_{k-L}\mathbf{x}_k] \tag{11}$$

From (10) and (11), the optimal solution is given by,

$$\mathbf{w}_k = \mathbf{Z}_k^{-1}\boldsymbol{\theta}_k \tag{12}$$

To convert (12) into a recursive form, define a gain matrix (analogous to the Kalman gain in the RLS algorithm) as,

$$\boldsymbol{\kappa}_k = \mathbf{Z}_{k-1}^{-1}\mathbf{B}\left(\mathbf{I}_{2x2} + \mathbf{D}^T\mathbf{Z}_{k-1}^{-1}\mathbf{B}\right)^{-1} \tag{13}$$

Using (13) in (10), we get,

$$\mathbf{Z}_k^{-1} = \mathbf{Z}_{k-1}^{-1} - \boldsymbol{\kappa}_k\mathbf{D}^T\mathbf{Z}_{k-1}^{-1} \tag{14}$$

Multiplying (13) from the right by $\left(\mathbf{I}_{2x2} + \mathbf{D}^T \mathbf{Z}_{k-1}^{-1}\mathbf{B}\right)$, and using (14), we obtain

$$\boldsymbol{\kappa}_k \left(\mathbf{I}_{2x2} + \mathbf{D}^T \mathbf{Z}_{k-1}^{-1}\mathbf{B}\right) = \mathbf{Z}_{k-1}^{-1}\mathbf{B} \Rightarrow \boldsymbol{\kappa}_k = \mathbf{Z}_k^{-1}\mathbf{B} \tag{15}$$

Substituting (11) in (12),

$$\mathbf{w}_k = \mathbf{Z}_k^{-1}\boldsymbol{\theta}_{k-1} + \mathbf{Z}_k^{-1}[(1+2\beta)d_k\mathbf{x}_k - \beta d_k \mathbf{x}_{k-L} - \beta d_{k-L}\mathbf{x}_k] \tag{16}$$

which can be further simplified as,

$$\mathbf{w}_k = \mathbf{w}_{k-1} - \boldsymbol{\kappa}_k \mathbf{D}^T \mathbf{w}_{k-1} + \mathbf{Z}_k^{-1}[(1+2\beta)d_k\mathbf{x}_k - \beta d_k \mathbf{x}_{k-L} - \beta d_{k-L}\mathbf{x}_k] \tag{17}$$

From the definition of **B**, $(1+2\beta)d_k\mathbf{x}_k - \beta d_k \mathbf{x}_{k-L} - \beta d_{k-L}\mathbf{x}_k = \mathbf{B}[d_k; d_k - \beta d_{k-L}]$, where $[d_k; d_k - \beta d_{k-L}]$ is a column vector with elements $d_k$ and $d_k - \beta d_{k-L}$. Therefore, the update equation can then be written as,

$$\mathbf{w}_k = \mathbf{w}_{k-1} - \boldsymbol{\kappa}_k \mathbf{D}^T \mathbf{w}_{k-1} + \boldsymbol{\kappa}_k [d_k; d_k - \beta d_{k-L}] \tag{18}$$

Note that the product $\mathbf{D}^T \mathbf{w}_{k-1} = [y_k \quad y_k - \beta y_{k-L}]^T$, where $y_k = \mathbf{x}_k^T \mathbf{w}_{k-1}$, $y_{k-L} = \mathbf{x}_{k-L}^T \mathbf{w}_{k-1}$ represent the outputs with the weights at the previous iteration. Defining an *apriori error vector* $\mathbf{e}_k$ as,

$$\mathbf{e}_k = [d_k - y_k; d_k - y_k - \beta(d_{k-L} - y_{k-L})] = [e_k; e_k - \beta e_{k-L}] \tag{19}$$

we can simplify (18) to give us the REW update equation,

$$\mathbf{w}_k = \mathbf{w}_{k-1} + \boldsymbol{\kappa}_k \mathbf{e}_k \tag{20}$$

A summary of the REW algorithm is shown in table 1.

Table 1. Summary of the REW algorithm

Initialize $\mathbf{Z}^{-1}(0) = c\mathbf{I}$, $c$ is a large positive constant

$\mathbf{w}(0) = \mathbf{0}$

At every iteration index *n*, compute

$\mathbf{B} = [(2\beta\mathbf{x}(n) - \beta\mathbf{x}(n-L)) \quad \mathbf{x}(n)]$ , $\mathbf{D} = [\mathbf{x}(n) \quad (\mathbf{x}(n) - \beta\mathbf{x}(n-L))]$

$\boldsymbol{\kappa}(n) = \mathbf{Z}^{-1}(n-1)\mathbf{B}\left(\mathbf{I}_{2x2} + \mathbf{D}^T\mathbf{Z}^{-1}(n-1)\mathbf{B}\right)^{-1}$

$y(n) = \mathbf{x}^T(n)\mathbf{w}(n-1)$ and $y(n-L) = \mathbf{x}^T(n-L)\mathbf{w}(n-1)$

$\mathbf{e}(n) = [d(n) - y(n); d(n) - y(n) - \beta(d(n-L) - y(n-L))] = [e(n); e_k - \beta e(n-L)]$

$\mathbf{w}(n) = \mathbf{w}(n-1) + \boldsymbol{\kappa}(n)\mathbf{e}(n)$

$\mathbf{Z}^{-1}(n) = \mathbf{Z}^{-1}(n-1) - \boldsymbol{\kappa}(n)\mathbf{D}^T\mathbf{Z}^{-1}(n-1)$

The complexity of the REW algorithm is O($N^2$) which is comparable to that of the RLS algorithm. The fixed-point nature of the REW algorithm results in fast convergence. In the next section, we will derive another algorithm for computing the EWC solution based on the Total Least-Squares (TLS) framework.

## EWC-TLS ALGORITHM

TLS is a powerful technique, which is widely used in parameter estimation problems in signal processing. Mathematically speaking, TLS solves an over-determined set of linear equations of the form $\mathbf{Ax} = \mathbf{b}$, where $\mathbf{A} \in \Re^{m\times n}$ is the data matrix, $\mathbf{b} \in \Re^m$ is the desired vector, and $\mathbf{x} \in \Re^n$ is the parameter vector and $m$ denotes the number of different observation vectors each of dimension $n$ [10]. Alternatively, the linear equations can be written in the form $[\mathbf{A};\mathbf{b}][\mathbf{x}^T;-1] = \mathbf{0}$, where $[\mathbf{A};\mathbf{b}]$ denotes an augmented data matrix. Let $\mathbf{S}$ be the SVD of the augmented matrix $[\mathbf{A};\mathbf{b}]$ such that $\mathbf{S} = \mathbf{U\Sigma V}^T$, where $\mathbf{U}^T\mathbf{U} = \mathbf{I}_m$, $\mathbf{V}^T\mathbf{V} = \mathbf{I}_{n+1}$ and $\mathbf{\Sigma} = [diag(\sigma_1, \sigma_2, \sigma_3, \sigma_4, ...., \sigma_{n+1});\ \mathbf{0}_{(m-n-1\times n+1)}]$ with all singular values $\sigma_k > 0$. If $[\mathbf{A};\mathbf{b}][\mathbf{x}^T;-1] = \mathbf{0}$, the smallest singular value must be zero. This is possible only if $[\mathbf{x}^T;-1]$ is a vector of $\mathbf{V}$ (corresponding to the zero singular value) normalized such that its $(n+1)^{\text{th}}$ element value is -1. When $[\mathbf{A};\mathbf{b}]$ is a symmetric square matrix, the solution reduces to the finding the eigenvector corresponding to the smallest eigenvalue of $[\mathbf{A};\mathbf{b}]$. The TLS solution in this special case is then,

$$[\mathbf{x};-1] = -\mathbf{v}_{n+1} / \mathbf{v}_{n+1,n+1} \tag{21}$$

where, $\mathbf{v}_{n+1,n+1}$ is the last element of the minor eigenvector $\mathbf{v}_{n+1}$. The Total Least-Squares technique can be easily applied to estimate the optimal MSE solution using fast minor components estimation algorithms [5,11]. However, in the case of EWC, these algorithms cannot be applied directly to solve the optimal EWC solution given by $\mathbf{w}_* = \mathbf{R}_L^{-1}\mathbf{P}_L$. This is mainly because of the fact that the eigenvalues of the augmented data matrix $\mathbf{G}$ (analogous to $[\mathbf{A};\mathbf{b}]$ mentioned before) given by (22) can take both positive and negative values.

$$\mathbf{G} = \begin{bmatrix} \mathbf{R}_L & \mathbf{P}_L \\ \mathbf{P}_L^T & 2\rho_d(L) \end{bmatrix} \tag{22}$$

The term $\rho_d(L)$ in (22) denotes the autocorrelation of the desired signal at lag $L$. It is important to note that the matrix (22) is square symmetric due to the symmetry of $\mathbf{R}_L$. Hence, the eigenvectors of $\mathbf{G}$ are real which is highly desirable. We would like to stress the fact that (22) still holds even with noisy data. This is because, the entries of $\mathbf{G}$ are unaffected by the noise terms. It is trivial to show that the minimum eigenvalue of (22) is zero. However, as we said before, the other eigenvalues of $\mathbf{G}$ can take both positive and negative values. This can be a problem for iterative gradient or fixed-point type algorithms. In order to rectify this problem, we propose to use the matrix $\mathbf{G}^2$ instead of $\mathbf{G}$. It is obvious that by squaring the matrix $\mathbf{G}$, all the eigenvalues are squared, but the eigenvectors still remain the same. The $O(N^2)$ algorithm proposed in [5] can now be used for fast and accurate estimation of the minor eigenvector of $\mathbf{G}^2$. Nevertheless, the squaring operation brings in additional computational overhead resulting in $O(N^3)$ complexity. We will now present a simple gradient procedure with adaptive step-size for estimating the minor component of the matrix $\mathbf{G}^2$. Consider the cost function in (23).

$$J(\mathbf{w}, \lambda) = \log(\mathbf{w}^T\mathbf{G}^2\mathbf{w}) - \log(\mathbf{w}^T\mathbf{w}) + \alpha\lambda(\mathbf{w}^T\mathbf{w} - 1) - \alpha\lambda^2 \tag{23}$$

If $\alpha$ is zero in (23), the cost reduces to the regular log energy type function that is typically associated with the principal components analysis algorithms. The parameter $\lambda$ can be considered as the lagrangian and together with $(\mathbf{w}^T\mathbf{w}-1)$ forms a penalty function that penalizes any deviation from unit norm weight estimates. The last term $\alpha\lambda^2$ ($\alpha\geq 0$) is required to limit the values of $\lambda$. This procedure of including additional penalty terms has been studied in optimization literature as penalty/barrier methods and augmented lagrangians [12]. Similar ideas have been applied successfully to increase the convergence speed of the LMS algorithm [13]. The goal is to minimize the cost function in (23) with respect to $\mathbf{w}$ and simultaneously maximize over $\lambda$. The gradients of (23) are then given by,

$$\begin{aligned}&\nabla_{\mathbf{w}} J(\mathbf{w},\lambda) = \mathbf{G}^2\mathbf{w}/(\mathbf{w}^T\mathbf{G}^2\mathbf{w}) - \mathbf{w}/(\mathbf{w}^T\mathbf{w}) + \alpha\lambda\mathbf{w}\\ &\nabla_{\lambda} J(\mathbf{w},\lambda) = \alpha(\mathbf{w}^T\mathbf{w}-1) - 2\alpha\lambda\end{aligned} \tag{24}$$

The corresponding update equations for $\mathbf{w}$ and $\lambda$ are,

$$\begin{aligned}&\mathbf{w}_{k+1} = \mathbf{w}_k - \eta_{\mathbf{w}}[\mathbf{G}^2\mathbf{w}_k/(\mathbf{w}_k^T\mathbf{G}^2\mathbf{w}_k) - \mathbf{w}_k/(\mathbf{w}_k^T\mathbf{w}_k) + \alpha\lambda\mathbf{w}_k]\\ &\lambda_{k+1} = \lambda_k + \eta_{\lambda}[\alpha(\mathbf{w}_k^T\mathbf{w}_k - 1) - 2\alpha\lambda]\end{aligned} \tag{25}$$

The terms $\eta_{\mathbf{w}}$ and $\eta_{\lambda}$ are small positive step-sizes. The parameter $\alpha$ must be always a positive quantity significantly higher than the step-sizes. An example parameter set is ($\eta_{\mathbf{w}}$, $\eta_{\lambda}$, $\alpha$) = (0.01, 0.1, 2). Generally, the eigenspread of $\mathbf{G}^2$ is very high due to the squaring and also because of the fact that the minimum eigenvalue of $\mathbf{G}^2$ is usually close to zero. In such cases, choosing very high $\alpha$ (in the range 10~50) can be helpful. It can be shown that $\mathbf{w}_k$ asymptotically converges to the eigenvector associated with the minimum eigenvalue of $\mathbf{G}^2$ and $\lambda$ converges to zero. We will defer the proof of convergence to a later paper due to space constraints.

**Online version of EWC-TLS algorithm:** The terms $\mathbf{G}^2\mathbf{w}_k$ and $\mathbf{w}_k^T\mathbf{G}^2\mathbf{w}_k$ can be directly estimated from the samples without explicitly estimating the matrix $\mathbf{G}^2$. Let $\boldsymbol{\psi}_k = [\mathbf{x}_k; d_k]$ and $\boldsymbol{\psi}_{k-L} = [\mathbf{x}_{k-L}; d_{k-L}]$. Let $\mathbf{G}_k$ denote the instantaneous estimate of $\mathbf{G}$ that can be computed by the recursive expression $\mathbf{G}_k = (1-\beta)\mathbf{G}_{k-1} + \beta(\boldsymbol{\psi}_k\boldsymbol{\psi}_{k-L}^T + \boldsymbol{\psi}_{k-L}\boldsymbol{\psi}_k^T)$. $\beta$ is a scalar forgetting factor in the range (0,1). Let $\mathbf{P}_k$ denote the instantaneous product of the matrix $\mathbf{G}_k$ and the weight estimate, i.e., $\mathbf{P}_k = \mathbf{G}_k\mathbf{w}_k$. Assuming that the weight vector does not change drastically during iterations, we can write the following recursion.

$$\mathbf{P}_k = (1-\beta)\mathbf{P}_{k-1} + \beta(\boldsymbol{\psi}_k y_{k-L} + \boldsymbol{\psi}_{k-L} y_k)\ ;\ y_k = \mathbf{w}_k^T\boldsymbol{\psi}_k\ ;\ y_{k-L} = \mathbf{w}_k^T\boldsymbol{\psi}_{k-L} \tag{26}$$

Note that the terms $y_k$ and $y_{k-L}$ represent the projected outputs corresponding to the composite inputs $\boldsymbol{\psi}_k$ and $\boldsymbol{\psi}_{k-L}$ respectively. The terms $\mathbf{G}^2\mathbf{w}_k$ and $\mathbf{w}_k^T\mathbf{G}^2\mathbf{w}_k$ in (25) can be approximated as $\mathbf{G}_k\mathbf{P}_k$ and $\mathbf{w}_k^T\mathbf{G}_k\mathbf{P}_k$ respectively. Thus, the complexity of the algorithm can be cut to $O(N^2)$ which is comparable to the REW method proposed in the previous section. In the next section, we will present some case studies including the design of an inverse controller using EWC.

## EWC CASE STUDIES

**System Identification:** In this example, we try to identify an unknown linear system (FIR filter of length 4) using the proposed EWC algorithms. The input signal is colored and corrupted with white noise (input SNR was set at 5dB) whereas the desired signal is clean. We performed MonteCarlo runs using different input and output signals and the results of EWC algorithms, IV method and optimal Wiener solution are shown in fig.2. The performance measure is the norm of the error vector (difference between true and estimated) measured in dB. EWC-TLS and REW algorithms outperform the Wiener solution for MSE criterion. The IV method also produces better results than the Wiener solution. For the IV method, we chose the delayed input vector $\mathbf{x}_{k-\Delta}$ as the instrument. The lag $\Delta$ was chosen to be four, which is the length of the true filter. Mathematically speaking, the IV method computes the solution $\mathbf{w} = E[\mathbf{x}_k \mathbf{x}_{k-\Delta}^T]^{-1} E[\mathbf{x}_{k-\Delta} d_k]$. Notice that there is a similarity between the IV solution and the recursive EWC solution $\mathbf{w}_* = \mathbf{R}_L^{-1}\mathbf{P}_L$. However, the EWC formulation is one that is based on the error whereas the IV does not have an associated cost function. One of niceties of the EWC solution is that the matrix $\mathbf{R}_L$ is symmetric and Toeplitz which facilitates robust and faster solutions. (Toeplitz matrix vector multiplications can be done in $O(N\log_2 N)$ compared to the regular $O(N^2)$ complexity). This also helps design robust algorithms based on the TLS framework.

**Inverse Modeling and Control Using REW algorithm:** We will show the application of EWC for designing a model reference inverse controller. Fig. 3 shows a block diagram of model reference inverse control [2]. Clearly, we require the plant parameters (which are typically unknown) to devise the controller. Once we have a model for the plant, the controller can be easily designed using conventional MSE minimization techniques. In this example, we will assume that the plant (AR system) transfer function is $P(z) = 1/(1+0.8z^{-1} - 0.5z^{-2} - 0.3z^{-3})$. The reference model is chosen to be an FIR filter with 5 taps. The block diagram for the plant identification is shown in fig. 4. Notice that the output of the plant is

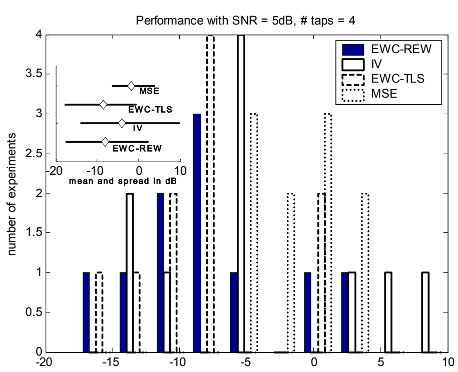


Figure 2. Histogram of the modeling errors

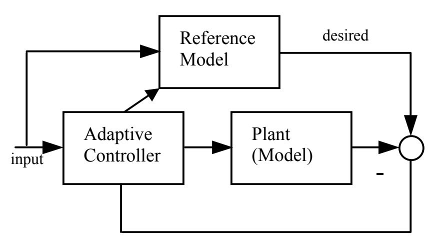


Figure 3. Block diagram for model reference inverse control

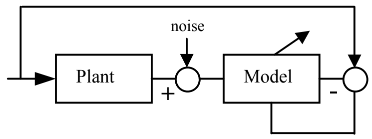


Figure 4. Block diagram for inverse modeling

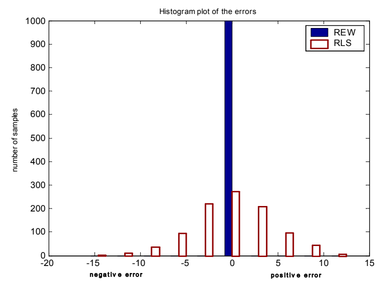


Figure 5. Histogram of tracking errors

corrupted with additive white noise due to measurement errors. The SNR at the plant output was set to 0dB. We then ran the REW and RLS algorithms to estimate the model parameters given the noisy input and desired signals. The model parameters thus obtained are used to derive the controller (see fig. 3) using standard backpropagation of error. We then tested the adaptive controller-plant pair for trajectory tracking by feeding a random time series and observing the responses. Ideally, the controller-plant pair must follow the trajectory generated by the reference model. Fig. 5 shows a histogram of the sample tracking errors (differences between the reference model response and the controller-plant output). Notice that the errors with REW controller are all concentrated around zero, giving an almost perfect controller for the plant. In contrast, the errors produced by the MSE based controller are high and could become worse if the SNR levels drop further.

## CONCLUSIONS

In this paper, we presented two algorithms to solve the recently proposed *Error Whitening Criterion* (EWC). The principal advantage of the EWC criterion is its ability to estimate the parameters of a linear system given noisy input and desired signals. The presence of noise in the input and output signals results in a significant bias if we use the conventional MSE based methods. The recursive algorithm called REW derived in this paper is a truly fixed-point type method with $O(N^2)$ complexity similar to the RLS algorithm. We then derived the EWC-TLS algorithm using minor components analysis similar to the Total Least Squares (TLS) method for MSE. The EWC-TLS method is based on a novel stochastic gradient algorithm that computes the minor eigenvector. We demonstrated the superiority of the EWC algorithms over its MSE counterparts in a system identification problem. We then explored the design of a model based inverse controller for an AR plant with noisy data. Adaptive controllers were designed using the plant models derived by the RLS and REW algorithms. The controller-plant pairs were then tested for trajectory tracking. RLS gave very poor plant representations and hence the controllers designed were inaccurate. On the other hand, REW algorithm deduced a very good approximation to the plant leading to the design of an accurate controller. The *Error Whitening Criterion* coupled with

the fast algorithms presented in this paper form a powerful tool that can be used in several engineering applications requiring accurate parameter estimation. Theoretical analysis of the sensitivity of the REW algorithm and the effect of input correlations are currently under study.
**Acknowledgements-** This work was partially supported by the National Science Foundation under Grant NSF ECS-9900394.

## REFERENCES

1. S. Haykin. "**Adaptive Filter Theory**." Prentice Hall, Upper Saddle River, New Jersey, 1996.
2. B. Widrow, E. Walach. "**Adaptive Inverse Control**." Prentice Hall, New Jersey, 1995.
3. T. Söderström, P. Stoica. "**System Identification**." Prentice-Hall, London, United Kingdom, 1989.
4. B. Widrow, M.E. Hoff Jr. "Adaptive switching circuits." **IRE Western Electric Show and Convention Record**, Part 4, pp. 96-104, August 23, 1960.
5. Y.N. Rao, J.C. Principe. "Efficient Total Least Squares Method for System Modeling Using Minor Component Analysis." Proceedings of **Neural Networks for Signal Processing** XII, pp. 259-268, 2002.
6. S.C. Douglas. "Analysis of an Anti-Hebbian Adaptive FIR Filtering Algorithm." **IEEE Trans. Circuits and Systems-II: Analog and Digital Signal Processing**, vol. 43, no. 11, 1996.
7. Y.N. Rao, D. Erdogmus, J.C. Principe. "Error Whitening Criterion Based Adaptive System Identification." Accepted for publication in **IJCNN** 2003.
8. J.C. Principe, Y.N. Rao, D. Erdogmus. "Error Whitening Wiener Filters: Theory and Algorithms." To appear as chapter 10 in **Advances in LMS Filters**, ed., S. Haykin, B. Widrow.
9. G.H. Golub, C.F. Van Loan. "**Matrix Computations**." Johns Hopkins University Press, Baltimore, MD, 1989.
10. F. Deprettere. Ed. "**SVD And Signal Processing–Algorithms, Applications and Architectures**." North-Holland, 1988.
11. C.E. Davila. "An Efficient Recursive Total Least Squares Algorithm for FIR Adaptive Filtering." **IEEE Trans. Signal Processing**, vol. 42, no. 2, February 1994.
12. D.G. Luenberger. "**Linear and Nonlinear Programming**."$2^{nd}$ ed., Addison-Wesley, Massachusetts, 1984.
13. Y. Wei, S.B. Gelfand, J.V. Krogmeier. "Noise Constrained Least Mean Squares Algorithm," **IEEE Trans. Signal Processing**, vol. 49, no. 9, September 2001.