# We are IntechOpen,
## the world's leading publisher of Open Access books
## Built by scientists, for scientists

**6,900**
Open access books available

**186,000**
International authors and editors

**200M**
Downloads

Our authors are among the

**154**
Countries delivered to

**TOP 1%**
most cited scientists

**12.2%**
Contributors from top 500 universities

CLARIVATE ANALYTICS
**BOOK CITATION INDEX**
INDEXED

**WEB OF SCIENCE**™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

## Interested in publishing with us?
## Contact book.department@intechopen.com

# On the Recursive Minimal Residual Method with Application in Adaptive Filtering

Noor Atinah Ahmad
*Universiti Sains Malaysia*
*Malaysia*

## 1. Introduction

Adaptive filters have become an integral part of adaptive signal processing and applied in diverse fields such as digital communications, speech recognition, control systems, radar and biomedical engineering. Since the adaptive filtering problem can be conveniently formulated as a stochastic quadratic minimization problem, a wide variety of adaptive filtering algorithms is derived based on numerical methods in unconstrained optimization theory. The class of iterative solvers for the adaptive filtering problem may be classified into the following categories:

1. Stochastic gradient algorithm
2. Stochastic conjugate gradient based algorithm
3. Direction set based algorithm

While stochastic gradient method offers the easiest and reliable method, its performance is proven to be poor when the eigenvalue spread of the autocorrelation matrix is large. Conjugate gradient based method (Boray & Srinath, 1992, Chang & Wilson, 2000, Dien & Bhaya, 2006) has been shown to give much better performance, however other issues arise. Finding the best estimate for gradient and search directions in the absence of the full knowledge of the autocorrelation matrix, and, in the presence of noise in the system is still an open question. Noisy estimates of gradient can easily lead to loss of conjugacy among search directions which may cause instability of the algorithm. A more recent advances can be found in (Chen, 1998) which involves a class of direction set based algorithm. The method generates a set of conjugate search directions using the method due to (Powell, 1964) and (Zangwill, 1967). This class of algorithm is very promising in that it allows for lower complexity variants which is of much improved performance compared to the celebrated Least Mean Square algorithm.

This chapter describes the current development in a class of search algorithm for adaptive filtering which is based on the minimal residual (MR) method. The algorithms are related to the 3 classifications above in one way or another which will be described in detailed in this chapter.

The development of MR based adaptive filtering algorithms involves modification of the standard MR method for iterative solution of a system of linear equation $\Phi\mathbf{x} = \mathbf{p}$. In the standard method, the approximate solution is updated along the current residual vector $\mathbf{r}^{(k)} = \mathbf{p} - \Phi\mathbf{x}^{(k)}$, and, the stepsize is chosen so that the residual norm squared $\left\|\mathbf{p} - \Phi\mathbf{x}^{(k+1)}\right\|_2^2$

is minimized. By doing so, an orthogonality condition $\tilde{\mathbf{r}}^{(k)T}\left(\Phi\mathbf{r}^{(k)}\right)=0$, where $\tilde{\mathbf{r}}^{(k)}=\mathbf{p}-\Phi\mathbf{x}^{(k+1)}$, is imposed, giving rise to conjugation of successive gradients. When the matrix $\Phi$ is positive definite, the method can be shown to converge to the solution of the system.

The standard MR method may also be performed using direction of search other than the residual vector. Suppose $\mathbf{d}^{(k)}$ is the $k$ th direction of search which is not necessarily equal to $\mathbf{r}^{(k)}$. The MR update equations take the form

$$\alpha_k = \frac{\mathbf{r}^{(k)T}\mathbf{b}^{(k)}}{\mathbf{b}^{(k)T}\mathbf{b}^{(k)}}$$
$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k\mathbf{d}^{(k)}$$

(1)

where $\mathbf{b}^{(k)}=\Phi\mathbf{d}^{(k)}$. The iteration in (1) guarantees $\mathbf{r}^{(k+1)T}\mathbf{b}^{(k)}=\left(\mathbf{p}-\Phi\mathbf{x}^{(k+1)}\right)^{T}\mathbf{b}^{(k)}=0$, i.e., (1) represents an orthogonal projection onto the subspace $span\left\{\mathbf{b}^{(k)}\right\}\approx span\left\{\Phi\mathbf{d}^{(k)}\right\}$. For a direction of search $\mathbf{d}^{(k)}$, we no longer insist pairwise conjugation of successive residuals. Instead, the conjugacy condition $\mathbf{r}^{(k+1)T}\left(\Phi\mathbf{d}^{(k)}\right)=0$ is imposed, i.e., we insist the new residual to be conjugated with the search direction $\mathbf{d}^{(k)}$.

For adaptive filtering problem, the MR method is applied on the recursive normal equation $\Phi_n\mathbf{x}=\mathbf{p}_n$, where $\Phi_n$ and $\mathbf{p}_n$ are the autocorrelation matrix and cross correlation vector of the problem repectively, at the $n$ th state. As a direct consequence, the resulting MR based algorithm will involve recursively updated residual and search direction which are updated at every state $n$. Two forms of MR based algorithms are presented in this chapter:

1.    MR based algorithm with recursive residuals as the search direction
2.    MR based algorithms with search directions fixed along the Euclidean unit vectors.

Choosing the current residual as the search direction leads to a stochastic gradient method where successive gradients are forced to be conjugated with respect to the current autocorrelation matrix of the adaptive least squares problem. As a result, we obtain a method whose convergence properties is very much similar and comparable to the conjugate gradient based methods. In other words, we achieve superior convergence rate compared to the standard stochastic gradient method. The superior convergence is achieved with a slightly reduced complexity compared to conjugate gradient based algorithms because the MR based method does not require the computation of conjugate directions.

However, using residuals as search direction has its drawbacks. One of which is that it tends to produce higher misadjustment when the problem is ill conditioned. This problem is greatly improved when search directions are fixed along the Euclidean directions. Two different forms of MR based methods are considered,

**Method 1**: MR iterations are performed along $N$ Euclidean directions independently, where $N$ denotes the filter order as well as the dimension of the problem.

**Method 2:** MR projections are performed along the Euclidean directions cyclically.

Method 1 and Method 2 are in fact closely related. While Method 1 solves the auxilliary system using a direct method, Method 2 implements Gauss-Seidel type iterations. Both methods are shown theoretically to be globally convergent.

Other specific issues pertaining to computational complexity, misadjustment and sensitivity to eigenvalue spread is also discussed. Detailed simulation using several different configuration of the adaptive filtering problem will be provided to highlight these issues.

## 2. Recursive implementation of the MR method in adaptive filtering

Before we discuss the algorithm, let us review the mathematical formulation for the adaptive filtering problem. For simplicity, we confine our discussion to the linear adaptive filtering problem. In almost all adaptive filtering problem, three fundamental signals are required,

i.   the input signal,
ii.  the desired signal
iii. the output signal from an adaptive filter model.

For linear adaptive filtering, we assume the adaptive filter model in the form of a linear model. The objective of an adaptive filtering process is to adapt the model coefficients so as to minimize a certain error criteria which is called the cost function. The most common form for the cost function is a least squares cost function. Its popularity is due to a simple reason; the function has a single global minimum.

A linear adaptive filtering problem is the following minimization problem,

$$\min_{\mathbf{x} \in R^N} J_n(\mathbf{x}) = \sum_{i=m}^{n} \lambda^{n-i} \left( \mathbf{a}_i^T \mathbf{x} - s(i) \right)^2 , \tag{2}$$

where $s(i) \in R$ is the desired signal, and, $y(i) = \mathbf{a}_i^T \mathbf{x}$ is the filter output at the $i$ th sample instant. An example of an adaptive filtering system is depicted in Fig. 1.

For a transversal finite impulse response (FIR) adaptive filter, vectors $\mathbf{a}_i \in R^N$ are formed by the input $u(i)$, such that $\mathbf{a}_i = \begin{bmatrix} u(i) & u(i-1) & \cdots & u(i-N+1) \end{bmatrix}^T$, and vector $\mathbf{x} \in R^N$ is an estimate of the filter coefficient vector. The quantity $e_i = \mathbf{a}_i^T \mathbf{x} - s(i)$ is the error signal and $\mathbf{x}$
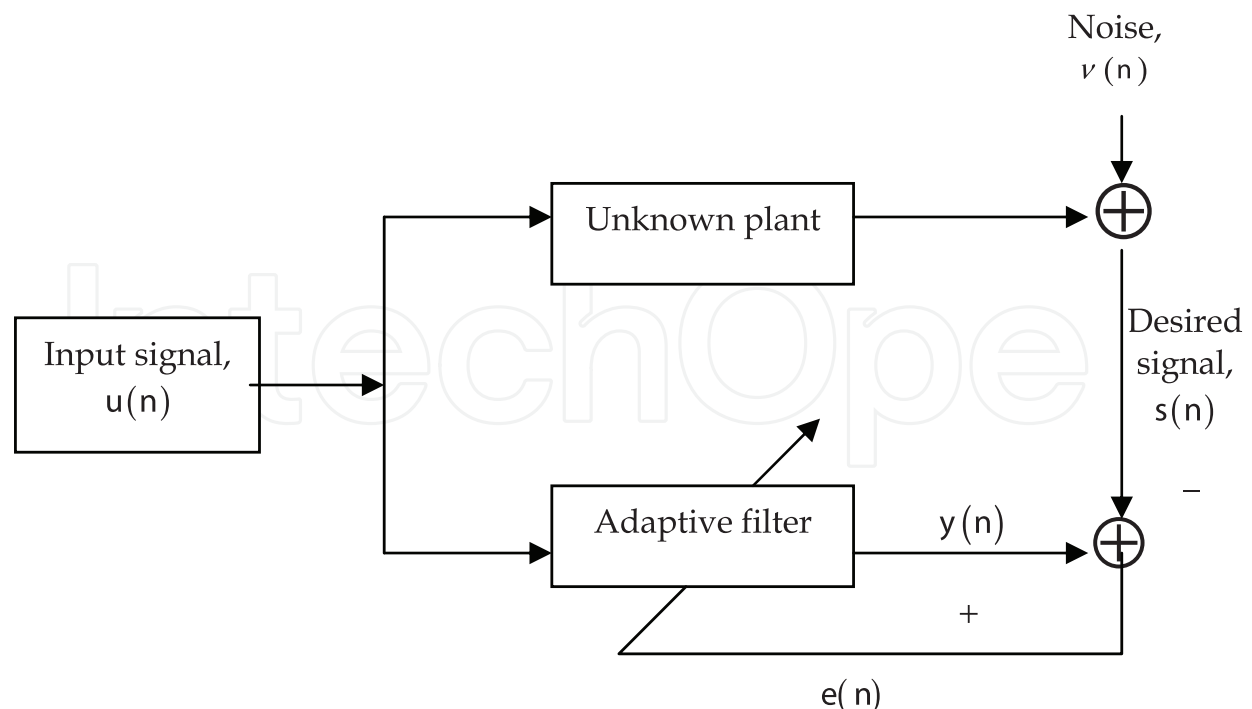


Fig. 1. Basic configuration for adaptive system identification

is updated by minimizing the sum of squared error cost function $J_n(\mathbf{x})$. The constant weight $\lambda \in [0,1]$ is known as the forgetting factor and its role is to give more emphasis/weight on the more recent signals.

For every system update, the adaptive least squares cost function is mathematically equivalent to the standard least squares problem. Consequently, the mathematical treatment of the problem is also similar. However, the time variation of $J_n(\mathbf{x})$ requires certain modification to take place. An advantage of the adaptive filtering problem is that system updating involves a rank-one update of the data matrix. This allows for the required modifications to be represented by simple recursive formulas to update the cost function. Similar to the standard least squares problem, the adaptive least squares problem (2) can also be recast in the form of adaptive normal equations. With definitions

$$\mathbf{A}_n = \left[ \sqrt{\lambda^{n-m}}\mathbf{a}_m \quad \mathbf{a}_{m+1} \quad \cdots \quad \mathbf{a}_n \right]^T, \; 0 \le m < n$$

and,

$$\mathbf{p}_n = \mathbf{A}_n^T \left[ \sqrt{\lambda^{n-m}}s(m), s(m+1), \ldots, s(n) \right]^T,$$

the minimization problem in (2) can be shown to be equivalent to solving the normal equation

$$\Phi_n \mathbf{x} = \mathbf{p}_n \tag{3}$$

where $\Phi_n = \mathbf{A}_n^T \mathbf{A}_n$. The data matrix $\mathbf{A}_n$ is rank-one updated during each system update and this leads to the following recursive formulas for $\Phi_n$ and $\mathbf{p}_n$ (Haykin, 1991),

$$\Phi_n = \lambda \Phi_{n-1} + \mathbf{a}_n \mathbf{a}_n^T, \tag{4}$$

$$\mathbf{p}_n = \lambda \mathbf{p}_{n-1} + \mathbf{a}_n s(n). \tag{5}$$

The minimal residual method may be applied to the normal equation (2), where the approximations to the coefficient vector $\mathbf{x}$ after the $n$th sample update is updated according to

$$\alpha_k = \frac{\mathbf{r}^{(k)T}\mathbf{b}^{(k)}}{\mathbf{b}^{(k)T}\mathbf{b}^{(k)}} \tag{6}$$

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)}$$

where $\mathbf{d}^{(k)}$ is the $k$th direction of search and $\mathbf{b}^{(k)} = \Phi_k \mathbf{d}^{(k)}$. By (6), the orthogonality condition $\mathbf{r}^{(k+1)T}\left(\Phi \mathbf{d}^{(k)}\right) = 0$ no longer holds since $\mathbf{r}^{(k+1)}$ is the residual vector at the next sample update. Instead, the orthogonal condition $\tilde{\mathbf{r}}^T\left(\Phi \mathbf{d}^{(k)}\right) = 0$ is imposed, where $\tilde{\mathbf{r}}^{(k)} = \mathbf{p}_k - \Phi_k \mathbf{x}^{(k+1)}$. A relationship between $\mathbf{r}^{(k+1)}$ and $\tilde{\mathbf{r}}^{(k)}$ can be observed below,

$$
\begin{aligned}
\mathbf{r}^{(k+1)} &= \mathbf{p}_{k+1} - \Phi_{k+1}\mathbf{x}^{(k+1)} \\
&= \lambda\left(\mathbf{r}^{(k)} - \alpha_k \Phi_k \mathbf{r}^{(k)}\right) + \mathbf{a}_{k+1}e_{k+1} \\
&= \lambda\tilde{\mathbf{r}}^{(k)} + \mathbf{a}_{k+1}e_{k+1},
\end{aligned}
\tag{7}
$$

where recursive formulas in (4) and (5) are used to simplify (7).
By construction of the MR method, $\alpha_k$ is chosen so that $\left(\tilde{\mathbf{r}}^{(k+1)}, \Phi_k \mathbf{d}^{(k)}\right) = 0$. Therefore, we have,

$$
\left(\mathbf{r}^{(k+1)}, \Phi_k \mathbf{d}^{(k)}\right) = \lambda\left(\tilde{\mathbf{r}}^{(k+1)}, \Phi_k \mathbf{d}^{(k)}\right) + e_{k+1}\left(\mathbf{a}_{k+1}, \Phi_k \mathbf{d}^{(k)}\right) = e_{k+1}\left(\mathbf{a}_{k+1}, \Phi_k \mathbf{d}^{(k)}\right).
\tag{8}
$$

Assuming, $\mathbf{d}_k$ is a descent direction, (8) implies that, as $e_{k+1} \to 0$, the behaviour of the stochastic MR based adaptive filtering algorithm approaches that of its deterministic counterpart.

## 3. Recursive minimal residual method as a form of stochastic gradient method

When the search direction is set to be the residual vector, the MR method reduces to a gradient descent method. For adaptive filtering application, since the gradient is only available as a stochastic gradient estimation, the method becomes a form of a stochastic gradient method. What is unique about the MR based stochastic gradient method is in the computation of step size $\alpha_k$, chosen so that the orthogonality condition

$$
\left(\tilde{\mathbf{r}}^{(k+1)}, \Phi_k \mathbf{r}^{(k)}\right) = 0,
\tag{9}
$$

is satisfied . In other stochastic gradient method, the step size is a predetermined value which is chosen to be a number inside the open interval $\left(0, 2/\lambda_{\max}\right)$, where $\lambda_{\max}$ denotes the maximum eigenvalue of the autocorrelation matrix $\Phi_k$. This choice guarantees global convergence of stochastic gradient method

### 3.1 The Stochastic Pairwise Conjugate Gradient based algorithm (SPCG)

The Stochastic Pairwise Conjugate Gradient based algorithm (SPCG) (Ahmad, 2008) is the first implementation of stochastic gradient method with MR step size. It was developed based on the idea in (Schraudolph & Graepel, 2002), where it was noted that the orthogonality condition $\left(\tilde{\mathbf{r}}^{(k+1)}, \Phi_k \mathbf{r}^{(k)}\right) = 0$ , also implies pairwise conjugation of succesive gradients. It is well known that conjugate gradient method out performs gradient based method in most applications. Henceforth, by insisting pairwise conjugation of successive gradients, we will achieve a stochastic gradient method which performs comparable to the conjugate gradient based methods.

The SPCG algorithm is summarized in Table 1. As shown in Table 1, the computational complexity of the SPCG algorithm is $O\left(N^2\right)$ which is still rather high for most practical application. The lower complexity versions are described in the next section

| | $\times / \div$ | $+/-$ |
|---|---|---|
| Initialization: | (if $m=1$) | |
| (1a) $\mathbf{X}_m = \mathbf{a}_m \mathbf{a}_m^T$ | 1 | |
| (1b) $\Phi_m, \mathbf{p}_m, \mathbf{x}^{(0)} = \mathbf{0} \in R^N$ | - | |
| (1c) $\mathbf{r}^{(m)} = \mathbf{p}_m - \Phi_m \mathbf{x}^{(0)}$, | - | |
| (1d) $\mathbf{b}^{(m)} = \Phi_m \mathbf{r}^{(m)}$ | 1 | |
| For $k = m+1, \ldots, N$ | | |
| (2) $\alpha^{(k)} = \dfrac{\mathbf{r}^{(k)T} \mathbf{b}^{(k)}}{\mathbf{b}^{(k)T} \mathbf{b}^{(k)}}$ | $2N+1$ | $2(N-1)$ |
| (3) $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha^{(k)} \mathbf{r}^{(k)}$ | $N$ | $N$ |
| New sample update: $\mathbf{a}_k \to \mathbf{a}_{k+1}$, | | |
| (4a) $\begin{aligned}\mathbf{X}_{k+1}(2:N, 2:N) \\ = \mathbf{X}_k(1:N-1, 1:N-1)\end{aligned}$ | - | |
| (4b) $\mathbf{X}_{k+1}(1,:) = u(n)\mathbf{a}_{k+1}^T$ | $N$ | |
| (4c) $\mathbf{X}_{k+1}(:,1) = \mathbf{X}_{k+1}(1,:)^T$ | - | |
| (5) $\Phi_{k+1} = \lambda \Phi_k + \mathbf{X}_{k+1}$ | $\dfrac{1}{2}N(N+1)$ | $\dfrac{1}{2}N(N+1)$ (considering symmetry) |
| (6) $e_{k+1} = \mathbf{a}_{k+1}^T \mathbf{x}^{(k+1)} - s(k+1)$ | $N$ | $2N-1$ |
| (7) $\mathbf{r}^{(k+1)} = \lambda \mathbf{r}^{(k)} - \alpha \lambda \mathbf{b}^{(k)} + \mathbf{a}_{k+1} e_{k+1}$ | $3N+1$ | $2N$ |
| (8) $\mathbf{b}^{(k+1)} = \Phi_{k+1} \mathbf{r}^{(k+1)}$ EndFor | $N^2$ | $N(N-1)$ |
| Total (if $\lambda = 1$) | $\dfrac{3}{2}N^2 + \dfrac{17}{2}N + 4$ $N^2 + 7N + 3$ | $\dfrac{1}{2}(3N^2 + 13N - 3)$ |

Table 1. The SPCG algorithm and calculation of the computational complexity

### 3.2 Low complexity SPCG

The computational complexity of the SPCG algorithm is of $O\left(N^2\right)$, mainly due to the matrix-vector multiplication to compute $\mathbf{b}^{(k)} = \Phi_k \mathbf{r}^{(k)}$. Lower complexity versions of SPCG algorithm introduced here are formed by making lower complexity approximation for $\mathbf{b}^{(k)} = \Phi_k \mathbf{r}^{(k)}$ as follows,

$$
\begin{aligned}
\varepsilon_k &= \mathbf{a}_k^T \mathbf{r}^{(k)}, \\
\hat{\mathbf{b}}^{(k)} &= \lambda \hat{\mathbf{b}}^{(k-1)} + \varepsilon_k \mathbf{r}^{(k)}.
\end{aligned}
\tag{10}
$$

The approximation above requires $3N$ multiplications instead of $N^2$ multiplication in the original version.

Naturally, with the approximations introduced by (10), the conjugacy condition (9) is no longer fulfilled exactly. Instead, the stepsize $\alpha_k$ is calculated based on a slightly modified conjugacy (orthogonality) condition, which is $\tilde{\mathbf{r}}^{(k)T}\left(\hat{\mathbf{b}}^{(k)}\right) = 0$. From geometrical point of view (see Fig. 2), this new orthogonality condition will still result in an improvement to $\left\|\mathbf{r}^{(k)}\right\|_2^2$. However the rate of convergence may be a bit compromised. This observation is verified in Theorem 1.

*Theorem 1*: Convergence of Low complexity SPCG

Let $v_k = \left\|\mathbf{r}^{(k)}\right\|_2^2 - \left\|\tilde{\mathbf{r}}^{(k)}\right\|_2^2$ be the improvement factor in the low complexity SPCG iteration after the $k$ th sample update. Then,

i.    $v_k \geq 0$ for $k = 1, 2, \ldots$

ii.   The iterations result in slow convergence when the angle between $\mathbf{r}^{(k)}$ and $\hat{\mathbf{b}}^{(k)}$ is close to $\pi/2$.
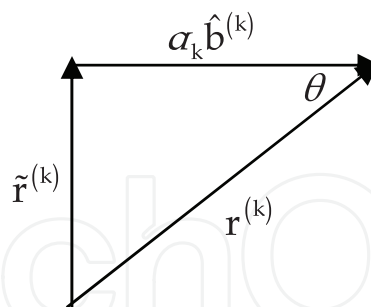


Fig. 2. Geometrical interpretation on Low Complexity SPCG iteration

*Proof*:

i.    $\left\|\tilde{\mathbf{r}}^{(k)}\right\|_2^2 = \left(\mathbf{r}^{(k)} - \alpha_k \hat{\mathbf{b}}^{(k)}\right)^T \left(\mathbf{r}^{(k)} - \alpha_k \hat{\mathbf{b}}^{(k)}\right) = \left\|\mathbf{r}^{(k)}\right\|_2^2 - 2\alpha_k \mathbf{r}^{(k)T}\hat{\mathbf{b}}^{(k)} + \alpha_k^2 \left\|\hat{\mathbf{b}}^{(k)}\right\|_2^2$.

Substituting for $\alpha_k$ gives

$$
v_k = \left\|\mathbf{r}^{(k)}\right\|_2^2 - \left\|\tilde{\mathbf{r}}^{(k)}\right\|_2^2 = \frac{\left(\mathbf{r}^{(k)T}\hat{\mathbf{b}}^{(k)}\right)^2}{\left\|\hat{\mathbf{b}}^{(k)}\right\|_2^2} \geq 0 .
$$

ii. $\quad \dfrac{\left\| \tilde{\mathbf{r}}^{(k)} \right\|_2^2}{\left\| \mathbf{r}^{(k)} \right\|_2^2} = 1 - \dfrac{\left( \mathbf{r}^{(k)T} \hat{\mathbf{b}}^{(k)} \right)^2}{\left\| \mathbf{r}^{(k)} \right\|_2^2 \left\| \hat{\mathbf{b}}^{(k)} \right\|_2^2} = 1 - \cos^2 \theta_k$ ,

where $\quad \theta_k = \dfrac{\mathbf{r}^{(k)T} \hat{\mathbf{b}}^{(k)}}{\left\| \mathbf{r}^{(k)} \right\|_2 \left\| \hat{\mathbf{b}}^{(k)} \right\|}$ , is the angle between $\mathbf{r}^{(k)}$ and $\hat{\mathbf{b}}^{(k)}$. When $\theta_k \approx \pi/2$, $v_k$ is

small, hence resulting in slow convergence.

Slow convergence observed in the case of Theorem 1 part (ii) can be remedied by restarting the algorithm with exact values of $\hat{\mathbf{b}}^{(k)}$. Note that, when exact values of $\hat{\mathbf{b}}^{(k)}$ is used, we will obtain the improvement factor corresponds to the SPCG algorithm, i.e.,

$$v_k = \frac{\left( \mathbf{r}^{(k)T} \mathbf{b}^{(k)} \right)^2}{\left\| \mathbf{b}^{(k)} \right\|_2^2} = \frac{\mathbf{r}^{(k)T} \Phi_k \mathbf{r}^{(k)}}{\left\| \Phi_k \mathbf{r}^{(k)} \right\|_2^2} .$$

Thus, restarting will result in a convergence rate comparable to the original SPCG algorithm.

Introducing restart will automatically increase the computational complexity of the algorithm because the autocorrelation matrix and the cross-correlation vector need to be updated during each sample update. However, if restarting is performed after every $N$ sample updates at least, this will keep the number of multiplications at $O(N)$ per sample update. The autocorrelation matrix and the cross-correlation vector can be updated for every block of $M$ according to

$$\Phi_{(k+1)M} = \lambda \Phi_{kM} + \tilde{\Phi}_{(k+1)M} , \qquad \mathbf{p}_{(k+1)M} = \lambda \mathbf{p}_{kM} + \tilde{\mathbf{p}}_{(k+1)M} \qquad (11)$$

where $\qquad \tilde{\Phi}_{(k+1)M} = \displaystyle\sum_{j=1}^{M} \mathbf{a}_{kM+j} \mathbf{a}_{kM+j}^T , \qquad \tilde{\mathbf{p}}_{(k+1)N} = \displaystyle\sum_{j=1}^{M} \mathbf{a}_{kM+j} s_{kM+j} .$

(Choose $M \geq N$ to keep the number of multiplications $O(N)$ per sample update, at most).

### 3.3 Simulation results

Simulations are based on an adaptive system identification configuration shown in Fig. 1. The unknown plant is a finite impulse response filter of order 10. A white Gaussian input signal of variance $\sigma^2 = 1$ is passed through a colouring filter with frequency response $H(z) = \dfrac{\sqrt{1-\alpha^2}}{1-\alpha z^{-1}}$ (Farhang-Bouroujeny, 1999), where $|\alpha| < 1$. The parameter $\alpha$ controls the eigenvalue spread, or the spectral condition number of the input autocorrelation matrix, where $\alpha = 0$ gives uncorrelated sequence (white) with eigenvalue spread $\approx 1$. The performance of the algorithm is studied based on the propogation of the ensemble average of the mean error norm $\left\| \mathbf{x}^{(k)} - \mathbf{x}^* \right\|_2^2$, where $\mathbf{x}^*$ is the Wiener solution. The ensemble average is formed by 200 independent runs.

In the first part of our simulation (see Fig. 3), we demonstrate the superior performance of the MR based stochastic gradient method, i.e., the SPCG algorihm compared to the well-known stochastic gradient algorithm, that is, the Least Mean Square (LMS) algorithm. In addition, we also show that the SPCG algorithm is comparable in performance compared to a conjugate gradient based adaptive filtering algorithm (referred to as the CG-CLF algorithm (Dien & Bhaya, 2006). The SPCG algorithm is also shown to be more superior when the eigenvalue spread is high (with $\alpha = 0.5$) (see Fig. 4). Increased eigenvalue spread also results in a slight increase in the magnitude of the weight error norm, which implies an increase in misadjustment. Infact our simulation shows that this problem is evident in LMS and CG-CLF as well.

In the second part of this simulation (Fig. 5 and Fig. 6), we investigate the effect of. In the simulation with lower complexity approximations (10). As expected, periodic restarting of the algorithm is required in order to achieve convergence rate comparable to the original version. It is interesting to see that the low complexity approximation (with restart) results in a slightly better misadjustment compared to the original SPCG.

## 4. Recursive MR method with euclidean search direction

Consider minimizing the squared norm residual $\|\mathbf{r}\|_2^2 = \|\mathbf{p} - \Phi\mathbf{x}\|_2^2$ by applying the MR iterations along $N$ Euclidean directions simultaneously and independently, thus

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \sum_{i=1}^{N} \Psi_k^{(i)} \mathbf{e}_i = \mathbf{x}^{(k)} + \Psi_k , \tag{12}$$
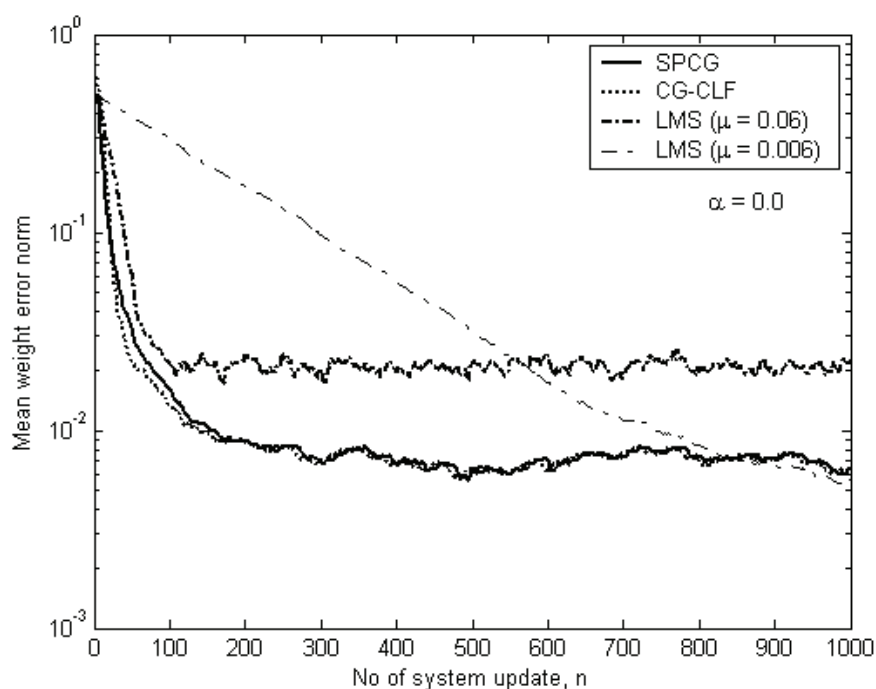


Fig. 3. Comparative performance between SPCG, LMS and CG-CLF for high eigenvalue spread. The LMS weight error norm is produced for two different stepsizes, $\mu = 0.06$ and $\mu = 0.006$.
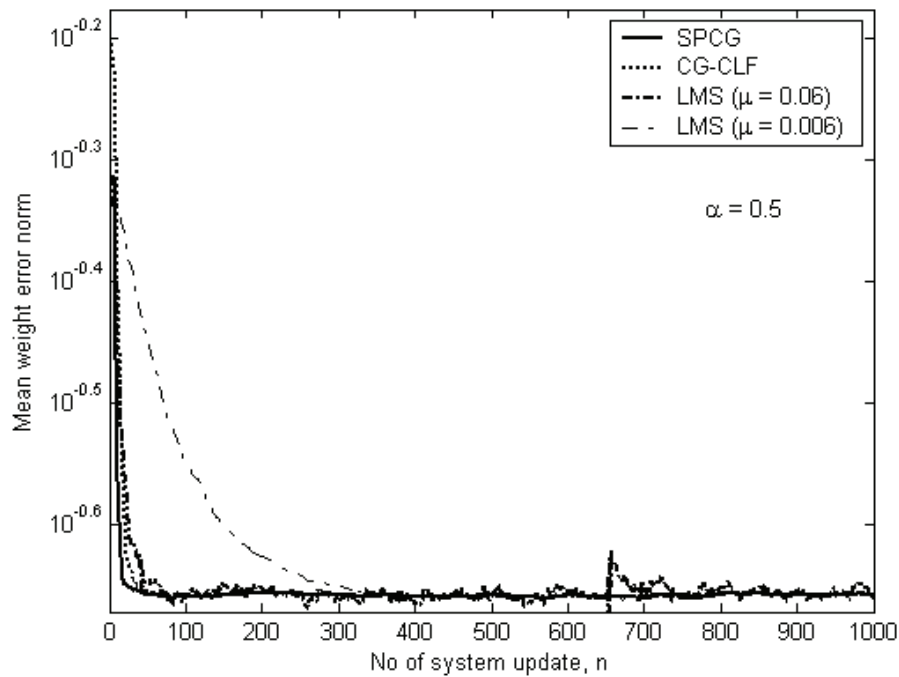
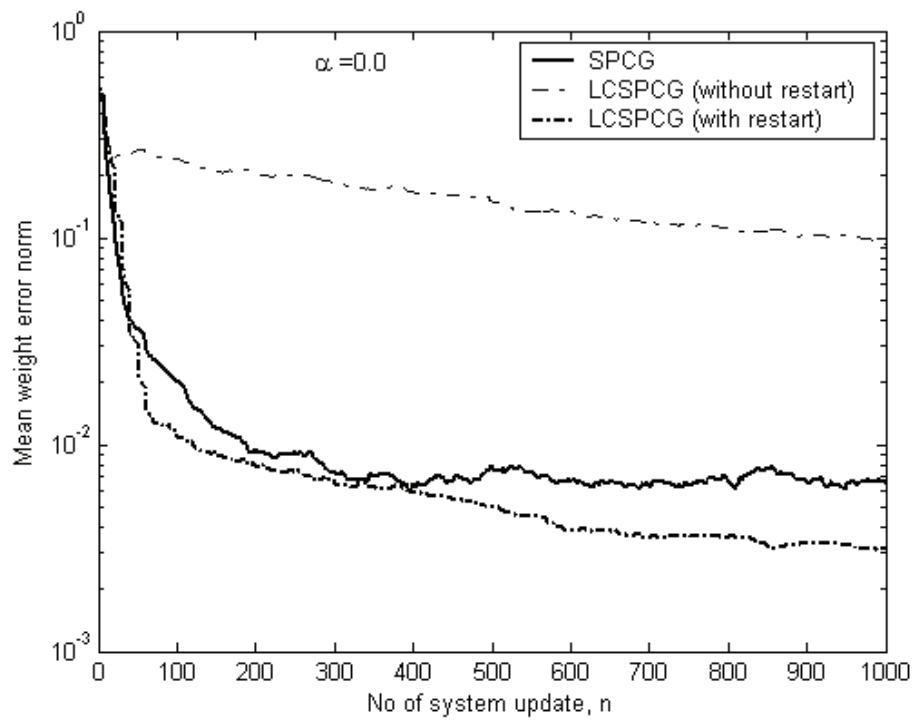Fig. 4. Comparative performance between SPCG, LMS and CG-CLF for high eigenvalue spread.



Fig. 5. Low complexity approximation requiring restart in order to achieve performance comparable to SPCG (small eigenvalue spread)
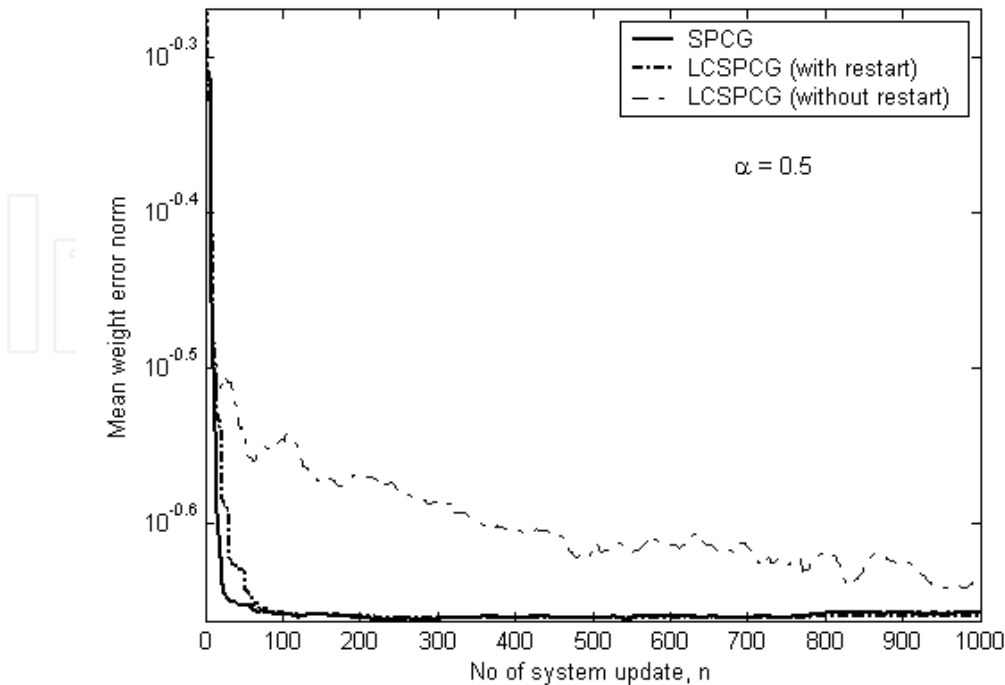
Fig. 6. Low complexity approximation compared to SPCG with large eigenvalue spread.

where $\Psi_k^{(i)}$ is the $i$th entry of the $N \times 1$ vector $\Psi_k$, and, $\mathbf{e}_i = \begin{bmatrix} 0 & 0 & \cdots & 1 & \cdots & 0 \end{bmatrix}^T$, with 1 appearing in the $i$th place. The stepsizes $\Psi_k^{(i)}$ are calculated by insisting that $\mathbf{r}^{(k+1)T} \Phi \mathbf{e}_i = 0$, for all $i = 1, \ldots, n$. For a symmetric $n \times n$ matrix $\Phi$, the procedure above gives rise to a system of linear equations in $\Psi_k^{(i)}$ of the form

$$\Phi^2 \Psi_k = \Phi \mathbf{r}_k , \tag{13}$$

which solves as

$$\Psi_k = \Phi^{-1} \mathbf{r}_k \tag{14}$$

In other words,

$$\mathbf{r}_{k+1} = \mathbf{b} - \Phi \left( \mathbf{x}_k + \Phi^{-1} \mathbf{r}_k \right) = \mathbf{r}_k - \mathbf{r}_k = \mathbf{0} . \tag{15}$$

Eqn (15) implies that the procedure described above results in finite termination to the exact solution after a single step.

### 4.1 Simultaneous update: The auxiliary equation
Applying the procedure above on the adaptive least squares normal equation (3), leads to the auxiliary normal equation

$$\Phi_k^2 \Psi_k = \Phi_k \mathbf{r}^{(k)} . \tag{16}$$

Rather than solving it directly, we adopt an alternative procedure as follows. First consider the equivalent equation,

$$\Psi_k^T \Phi_k^2 \Psi_k = \Psi_k^T \Phi_k \mathbf{r}^{(k)} . \tag{17}$$

Now, we split $\mathbf{Q}_k = \Phi_k^2$ as,

$$\mathbf{Q}_k = \Phi_k^2 = \mathbf{L}_k + \mathbf{D}_k + \mathbf{U}_k , \tag{18}$$

where $\mathbf{D}_k$ is a diagonal matrix consisting of the main diagonal of $\mathbf{Q}_k$, and, matrices $\mathbf{L}_k$ and $\mathbf{U}_k$ are the strict lower and upper triangular parts of $\mathbf{Q}_k$ respectively. This procedure transforms the alternative auxilliary equation to

$$\Psi_k^T \mathbf{L}_k \Psi_k + \Psi_k^T \mathbf{D}_k \Psi_k + \Psi_k^T \mathbf{U}_k \Psi_k = \Phi_k \mathbf{r}^{(k)} .$$

Consider the value of $\Psi_k^T \mathbf{L}_k \Psi_k$:
It is straightforward to see that the entries of $\mathbf{L}_k \Psi_k$ are given as

$$\left[ \mathbf{L}_k \Psi_k \right]_1 = 0 , \tag{19}$$

$$\left[ \mathbf{L}_k \Psi_k \right]_i = \sum_{j=2}^{i-1} \Psi_k^{(j)} \left( \Phi_k^{(i)T} \Phi_k^{(j)} \right) = \sum_{j=2}^{i-1} \Psi_k^{(j)} \mathbf{Q}_k^{(i,j)} , i = 2,\ldots,N \tag{20}$$

where $\mathbf{Q}_k^{(i,j)}$ is the entry of $\mathbf{Q}_k$ in the $i$ th row and $j$ th column. Thus

$$\Psi_k^T \mathbf{L}_k \Psi_k = \sum_{i=2}^{N} \Psi_k^{(i)} \sum_{j=1}^{i-1} \Psi_k^{(j)} \mathbf{Q}_k^{(i,j)} .$$

Switching the order of summation leads to,

$$\begin{aligned}
\Psi_k^T \mathbf{L}_k \Psi_k &= \sum_{i=2}^{N} \Psi_k^{(i)} \sum_{j=1}^{i-1} \Psi_k^{(j)} \mathbf{Q}_k^{(i,j)} \\
&= \sum_{j=1}^{N-1} \Psi_k^{(j)} \sum_{i=j}^{N} \Psi_k^{(i)} \mathbf{Q}_k^{(i,j)} \\
&= \sum_{j=1}^{N-1} \Psi_k^{(j)} \sum_{i=j}^{N} \Psi_k^{(i)} \mathbf{Q}_k^{(j,i)} = \Psi_k^T \mathbf{U}_k \Psi_k.
\end{aligned} \tag{21}$$

The last line of (21) is obtained by using the fact that $\mathbf{Q}_k = \Phi_k^2$ is symmetric. Using (18), we are now able to write (17) as

$$\Psi_k^T \left( \mathbf{D}_k + 2\mathbf{L}_k \right) \Psi_k = \Psi_k^T \Phi_k \mathbf{r}^{(k)} , \tag{22}$$

Hence, the solution to the auxilliary equation is also the solution to

$$\left( \mathbf{D}_k + 2\mathbf{L}_k \right) \Psi_k = \Phi_k \mathbf{r}^{(k)} \tag{23}$$

or equivalently,

$$\left( \mathbf{D}_k + 2\mathbf{U}_k \right) \Psi_k = \Phi_k \mathbf{r}^{(k)} \tag{24}$$

By reducing (16) to (23) or (24), we have in fact reduced the problem to a lower/upper triangular system which can be solved by forward/back substitution.

### 4.2 Cyclic updates: Gauss-seidel iterations
An alternative MR based method is derived by performing MR iterations along the Euclidean directions cyclically. In other words, the current weight vector is updated as follows,

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \Psi_k ,$$

where the $i$ th components of the stepsize vector $\Sigma_k$, namely $\Sigma_k^{(i)}$, is the minimizer of $\left\| \mathbf{p}_k - \Phi_k \left( \mathbf{x}_i^{(k)} + \Psi^{(i)} \mathbf{e}_i \right) \right\|_2^2$, i.e.,

$$\Psi_k^{(i)} = \frac{\mathbf{r}_{i-1}^{(k)T} \Phi_k^{(i)}}{\left( \Phi_k^{(i)} \right)^T \left( \Phi_k^{(i)} \right)} , \qquad\qquad i = 1,\ldots,N$$

with

$$\mathbf{r}_i^{(k)} = \mathbf{r}_{i-1}^{(k)} - \Psi_k^{(i)} \Phi_k^{(i)} ,$$

$$\Phi_k^{(i)} \text{ - the } i \text{ th row of } \Phi_k .$$

The choice of $\Psi_k^{(i)}$ guarantees

$$\mathbf{r}_i^{(k)T} \left( \Phi_k \mathbf{e}_i \right) = \mathbf{r}_i^{(k)T} \Phi_k^{(i)} = 0 . \qquad\qquad (25)$$

Note that, by this construction,

$$\mathbf{r}_i^{(k)} = \mathbf{r}^{(k)} - \sum_{j=1}^{i} \Psi_k^{(j)} \Phi_k^{(j)} ,$$

and, applying (25) gives,

$$\mathbf{r}_i^{(k)T} \Phi_k^{(i)} = 0 = \mathbf{r}^{(k)T} \Phi_k^{(i)} - \left( \sum_{j=1}^{i} \Psi_k^{(j)} \Phi_k^{(j)} \right)^T \Phi_k^{(i)}$$

$$= \mathbf{r}^{(k)T} \Phi_k^{(i)} - \sum_{j=1}^{i} \Psi_k^{(j)} \left( \Phi_k^{(j)T} \Phi_k^{(i)} \right).$$

Thus,

$$\Psi_k^{(1)} = \frac{\mathbf{r}^{(k)T} \Phi_k^{(1)}}{\left\| \Phi_k^{(1)} \right\|_2^2} . \qquad\qquad (26)$$

For $i = 2,\ldots,N$ ,

$$\Psi_k^{(i)} = \left\{ \Phi_k^{(i)T} \left( \mathbf{r}^{(k)} - \sum_{j=1}^{i-1} \Psi_k^{(j)} \Phi_k^{(j)} \right) \right\} \Big/ \left\| \Phi_k^{(i)} \right\|_2^2 . \tag{27}$$

Eqns. (26) and (27) can be written in the compact matrix form as

$$\mathbf{D}_k \Psi_k = \Phi_k \mathbf{r}^{(k)} - \mathbf{L}_k \Psi_k , \tag{28}$$

where $\mathbf{D}_k$ and $\mathbf{L}_k$ are defined as in (18). The form of (28) resembles the iteration procedure of the Gauss-Seidel method applied on the transformed auxilliary equation (17), but with $\Psi_{k-1} = \mathbf{0}$ . The equivalent equation for the Gauss-Seidel method would be

$$\mathbf{D}_k \Psi_k = \Phi_k \mathbf{r}^{(k)} - \mathbf{L}_k \Psi_k - \mathbf{U}_k \Psi_{k-1} . \tag{30}$$

### 4.3 Simulation results
We use the same adaptive filtering configuration to study the performance of MR based algorithm with euclidean direction of search. We compare the two algorithms with the SPCG algorithm to study their performance with respect to eigenvalue spread. When the eigenvalue spread is small (Fig. 7), the SPCG results in lower misadjustment compared to the MR methods with Euclidean direction search. However, for large eigenvalue spread (Fig. 8), SPCG shows a significant increased in misadjustment while the other MR methods appear unaffected by the increased in eigenvalue spread.

## 5. Future developments

In this chapter we have described two different types of recursive MR method for adaptive filtering. The methods defer by the choice of search directions. The first method is a stochastic gradient method with a stepsize determined using the MR stepsize formula that guarantees conjugacy of successive gradients. In the second method, search directions are chosen to be $N$ Euclidean unit vectors in $R^N$ , with two different approaches in performing projections onto the row space of $\Phi$ .
Here's a summary of the performance of these methods:
i.    The SPCG algorithm is proven to be superior compared to the traditional least mean square stochastic gradient method. We have also described a procedure to modify the SPCG algorithm so as to achieve complexity comparable to LMS algorithm.
ii.   A common problem in most gradient based algorithm which is also evident in the SPCG algorithm is the poor performance when eigenvalue spread of the autocorrelation matrix is high. This problem is removed when the search direction is switched towards the Euclidean directions. Our results clearly show that the performance of recursive MR methods are unaffected by the increase in eigenvalue spread.
The two key issues that will be the focus of our research in the efforts to bring further improvements to the recursive MR methods are i) providing low complexity variants, ii) improving misadjustments especially when the eigenvalue spread of the problem is high. Research and investigations on block implementations of these algorithms are currently under way. We are also looking at a class of potential new preconditioners for the adaptive least squares problem in general. These preconditioners are in the form of recursive incomplete QR factorization preconditioners based on different orthogonalization techniques such as Gram-Schmidt, Givens rotations and Householder method.
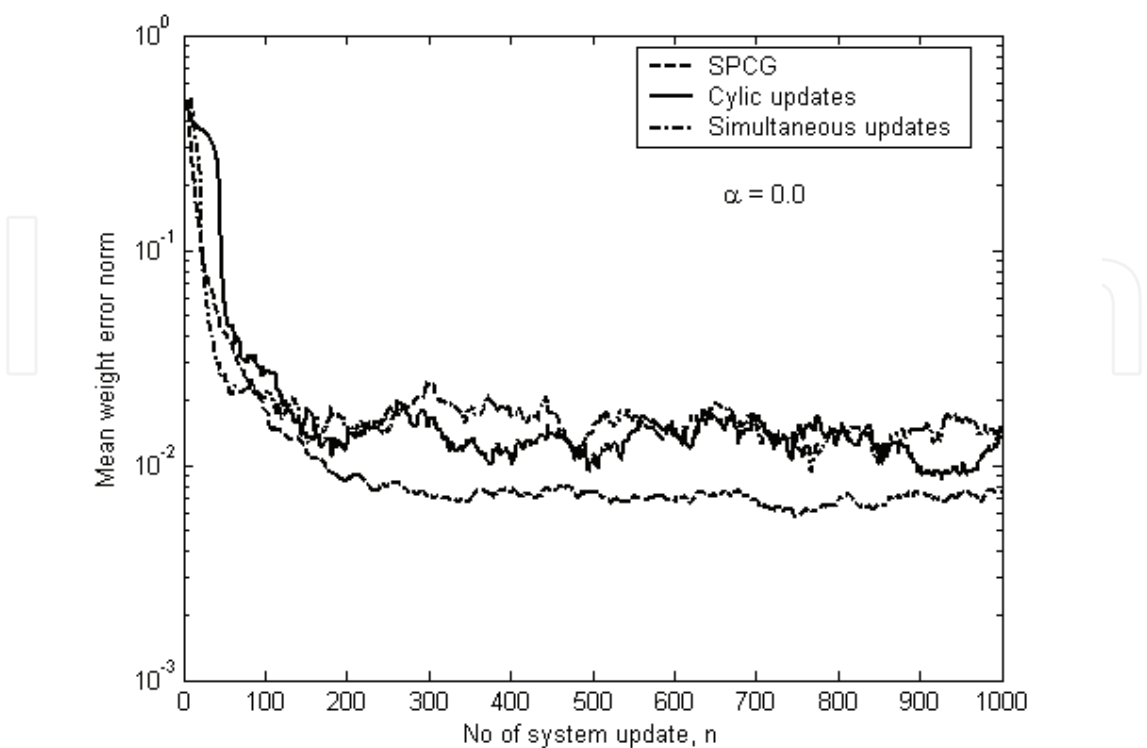
Fig. 7. SPCG compares with MR based algorithms with Euclidean direction of search (small eigenvalue spread)
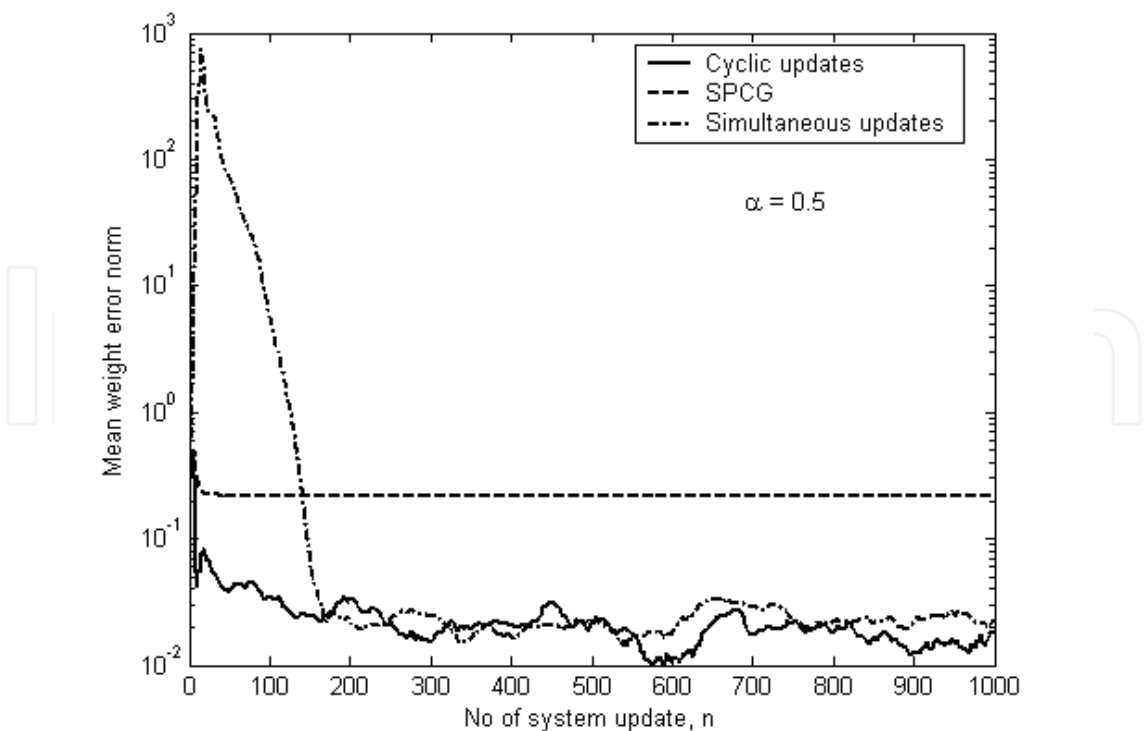


Fig. 8. SPCG compares with MR based algorithms with Euclidean direction of search (large eigenvalue spread)

## 6. References

Ahmad, N.A. (2008) A globally convergent stochastic pairwise conjugate gradient based adaptive filtering algorithm, *IEEE Signal Process. Lett.*, vol. 15, pp. 914-917.

Boray, G.K. & Srinath, M.D. (1992), Conjugate gradient techniques for adaptive filtering, *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol 39, no. 1, pp. 1-10.

Chang, P.S and Wilson, A.N. (2000), Analysis of conjugate gradient algorithms for adaptive filtering, *IEEE Trans. Signal Process.*, vol 48, no.2, pp. 409-418.

Chen, M. Q. (1998). A Direction set based algorithm for least squares problems in adaptive signal processing. *Linear Algebra and its Applications*, 284: 73-94.

Diene, O. and Bhaya, A. (2006), Adaptive filtering algorithms designed using control Liapunov functions, *IEEE Signal Process. Lett.*, vol 13, no. 4, pp. 224-227.

Farhang-Boroujeny, B. (1999), *Adaptive Filters: Theory and Applications*, John Wiley &Sons.

Haykin, S. (1991), *Adaptive Filter Theory*, Englewood Cliffs, NJ: Prentice-Hall.

Powell, M. J. D. (1964), An efficient method for finding the minimum of a function of several variables without calculating derivatives, Comput. J. 7 155-162.

Schraudolph, N. & Graepel, T. (2002), Towards stochastic conjugate gradient method, *Proc. 9th Int. Conf. Neural Processing, Singapore 2002.*

Zangwill, W. (1967). Minimizing a function without calculating derivatives, Comput. J. 10 293-296.
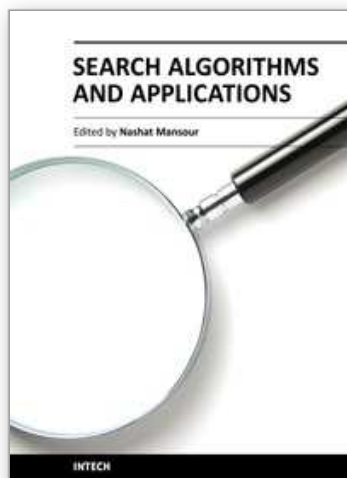
**Search Algorithms and Applications**

Edited by Prof. Nashat Mansour

Search algorithms aim to find solutions or objects with specified properties and constraints in a large solution search space or among a collection of objects. A solution can be a set of value assignments to variables that will satisfy the constraints or a sub-structure of a given discrete structure. In addition, there are search algorithms, mostly probabilistic, that are designed for the prospective quantum computer. This book demonstrates the wide applicability of search algorithms for the purpose of developing useful and practical solutions to problems that arise in a variety of problem domains. Although it is targeted to a wide group of readers: researchers, graduate students, and practitioners, it does not offer an exhaustive coverage of search algorithms and applications. The chapters are organized into three parts: Population-based and quantum search algorithms, Search algorithms for image and video processing, and Search algorithms for engineering applications.

# INTECH
open science | open minds