A Square-Root Adaptive V-BLAST Algorithm for Fast Time-Varying MIMO Channels

Athanasios A. Rontogiannis
Institute of Space Applications
and Remote Sensing
National Observatory of Athens
15236, P. Penteli, Athens, Greece
Email: tronto@space.noa.gr

Vassilis Kekatos
Dept. of Computer Eng. & Informatics
and C.T.I.-R&D
University of Patras
26500 Rio-Patras, Greece
Email: kekatos@ceid.upatras.gr

Kostas Berberidis
Dept. of Computer Eng. & Informatics
and C.T.I.-R&D
University of Patras
26500 Rio-Patras, Greece
Email: berberid@ceid.upatras.gr

Abstract— **Among the methods that have been proposed for a Multiple-Input Multiple-Output (MIMO) receiver, the V-BLAST algorithm provides a good compromise between transmission rate, achievable diversity, and decoding complexity. In this paper, we derive a new adaptive V-BLAST type equalization scheme for fast time varying, flat fading MIMO channels. The proposed equalizer stems from the Cholesky factorization of MIMO system's output data autocorrelation matrix and the equalizer's filters are updated in time using numerically robust unitary Givens rotations. The new square-root algorithm exhibits identical performance to a recently proposed V-BLAST adaptive algorithm, offering at the same time substantially reduced computational complexity. Moreover, as expected due to its square-root form and verified by simulations, the algorithm exhibits particularly favourable numerical behaviour.**

I. Introduction

Multiple-Input Multiple-Output (MIMO) wireless communications technology has gained considerable attention in recent years due to its potential for significant increase in effective data rate compared to traditional Single-Input Single-Output (SISO) technology. A widely used architecture for reliable symbol detection in flat fading MIMO systems is the V-BLAST architecture [1]. In a V-BLAST receiver the symbol corresponding to the stream with the highest Signal-to-Noise Ratio (SNR), is detected first. Then the contribution of the detected symbol is subtracted from the output vector. This procedure is repeated for the remaining symbols by always selecting the strongest signal among the undetected ones. Assuming a known MIMO channel, computationally efficient implementations of the V-BLAST technique have been proposed in [2], [3], [4], and [5]. In a fast time varying environment, however, the MIMO channel must be estimated quite frequently before the V-BLAST method is applied, resulting in undesirable increase of the computational complexity.

By taking advantage of the equivalence between the V-BLAST receiver and the Generalized Decision Feedback Equalizer (GDFE) [6], an architecture appropriate for *adaptive* equalization of flat fading MIMO channels has been presented in [7]. To the best of our knowledge, the algorithm proposed in [7] is the only V-BLAST type scheme in which not only the equalizer taps, but also the symbol detection order are updated recursively in time. As a result a significant reduction in computational complexity is achieved compared to previously derived V-BLAST schemes.

Motivated by the work in [7], we have developed a new adaptive V-BLAST algorithm for flat, time varying MIMO channels. In the proposed algorithm, a suitable transformation of the equalizer filters used in [7] is employed, which stems from the Cholesky factorization of the equalizer's input autocorrelation matrix. It turns out that the proposed algorithm offers considerable computational savings compared to the method described in [7] with no loss in performance. Moreover, the numerical behaviour of the algorithm is expected to be quite robust, since it is based on a square root of the system's autocorrelation matrix.

The outline of this paper is as follows. In Section II, the problem is formulated and the V-BLAST equalization architecture of [7] is revisited. In Section III, the new algorithm is derived and computational complexity issues are addressed. The performance of the proposed algorithm is evaluated through simulations in Section IV, while Section V concludes this work.

II. Problem formulation and preliminaries

Let us consider a MIMO flat fading system with M transmit and N receive antennas, and $M \leq N$. The $N \times 1$ received vector $\mathbf{y}(n)$ at time n is expressed as follows:

$$\mathbf{y}(n) = \mathbf{H}(n)\mathbf{d}(n) + \mathbf{v}(n) \qquad (1)$$

where $\mathbf{H}(n)$ is the $N \times M$ channel matrix with entries the complex coefficients of the corresponding $N \times M$ subchannels, $\mathbf{d}(n)$ stands for the $M \times 1$ vector of symbols that are simultaneously transmitted by the M antennas, and $\mathbf{v}(n)$ denotes white Gaussian noise. Adaptive V-BLAST type equalization of the system described in (1) can be performed using the architecture shown in Fig. 1 [7]. In Fig. 1, it is assumed that the symbol detection order is known a priori and is denoted as $\{k_1, k_2, \ldots, k_M\}$, where $k_i \in \{1, 2, \ldots, M\}$. The equalizer comprises M serially connected conventional Decision Feedback Equalizer (DFE) structures. As far as the

This work was partially supported by the General Secretariat for Research and Technology, under grant ΠΕΝΕΔ-No.03ΕΔ838.

1-4244-0355-3/06/$20.00 (c) 2006 IEEE

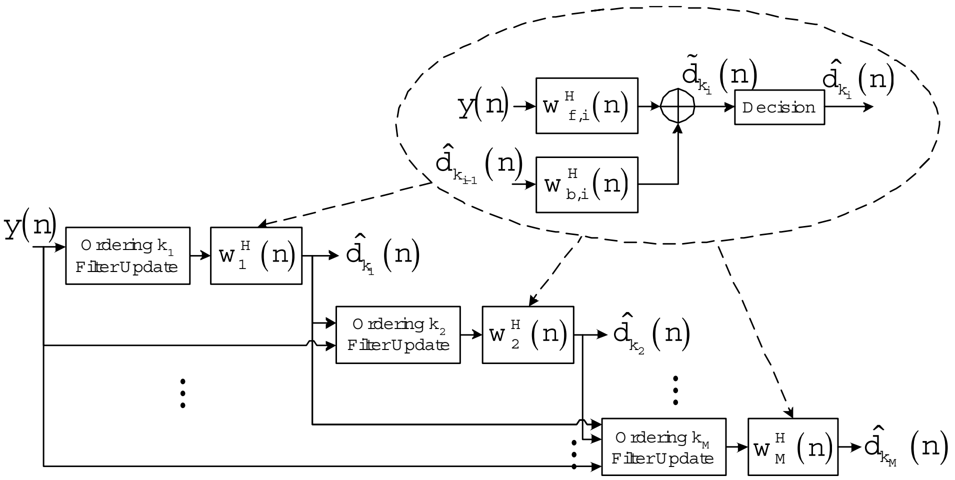

Fig. 1. Adaptive V-BLAST type equalization architecture.

i-th DFE is concerned, the feedforward filter input coincides with the received vector $\mathbf{y}(n)$, while the feedback filter input is the vector of already detected symbols from the previous DFEs. Thus, the output of the i-th DFE is given as follows:

$$\tilde{d}_{k_i}(n) = \mathbf{w}_i^H(n)\mathbf{y}_i(n) \tag{2}$$

where

$$\mathbf{w}_i(n) = \begin{cases} \mathbf{w}_{f,i}(n)\ , & i = 1 \\ \left[\mathbf{w}_{f,i}^T(n), \mathbf{w}_{b,i}^T(n)\right]^T\ , & i = 2, \ldots, M \end{cases} \tag{3}$$

$$\mathbf{y}_i(n) = \begin{cases} \mathbf{y}(n)\ , & i = 1 \\ \left[\mathbf{y}^T(n), \hat{\mathbf{d}}_{k_{i-1}}^T(n)\right]^T\ , & i = 2, \ldots, M \end{cases} \tag{4}$$

$\hat{\mathbf{d}}_{k_{i-1}}(n) = [\hat{d}_{k_1}(n), \hat{d}_{k_2}(n), \ldots, \hat{d}_{k_{i-1}}(n)]^T$ is the vector of already detected symbols, i.e., $\hat{d}_{k_i}(n) = f\{\tilde{d}_{k_i}(n)\}$, where $f\{\cdot\}$ is the decision device function, and $(\cdot)^T$ stands for simple transposition, while $(\cdot)^H$ for complex conjugate transposition.

Under fast fading conditions, not only the equalizer filters but also the detection order need to be adapted at each time instant. An efficient Recursive Least Squares (RLS) approach, which takes into consideration both requirements, has been introduced in [7]. More specifically, let us assume that $i-1$ symbols have been detected, i.e., symbols $\hat{d}_{k_1}(n), \hat{d}_{k_2}(n), \ldots, \hat{d}_{k_{i-1}}(n)$. In order to determine index k_i of the next symbol to be detected, as well as to update the corresponding DFE filters, a double minimization procedure is followed. First, each of the i-th order[1] LS cost functions:

$$\mathcal{E}_{i,j}(n) = \sum_{l=1}^{n} \lambda^{n-l} |\hat{d}_j(l) - \mathbf{w}_{i,j}^H(n)\mathbf{y}_i(l)|^2 \tag{5}$$

is minimized with respect to $\mathbf{w}_{i,j}(n)$ for $j \in S_i = \{1, 2, \ldots, M\} \setminus \{k_1, k_2, \ldots, k_{i-1}\}$, where λ is the usual forgetting factor. Then, the lowest of the resulting LS energies is obtained, and is denoted as $\mathcal{E}_i(n)$. The corresponding index j determines the order of the next symbol to be detected, i.e. k_i. Finally, the filter that gives the lowest LS energy is selected as the i-th DFE at time n, and is denoted as $\mathbf{w}_i(n)$. It can be easily shown that the order selection criterion described above is equivalent to the criterion used in V-BLAST with the expectation operation replaced by exponentially weighted time averaging. By employing an alternative but equivalent parameterization of the equalizer filters, we describe in the next section a numerically robust, computationally efficient method for both detection order, and DFE filters adaptation.

III. Derivation of the new algorithm

It is well known that minimization of $\mathcal{E}_{i,j}(n)$ in (5) with respect to $\mathbf{w}_{i,j}(n)$ leads to the following solution:

$$\mathbf{w}_{i,j}(n) = \mathbf{\Phi}_i^{-1}(n)\mathbf{z}_{i,j}(n) \tag{6}$$

where $\mathbf{\Phi}_i(n)$ stands for the $(N+i-1) \times (N+i-1)$ time-averaged input data autocorrelation matrix, and $\mathbf{z}_{i,j}(n)$ is the $(N+i-1) \times 1$ time-averaged input-output crosscorrelation vector given by [8]

$$\mathbf{z}_{i,j}(n) = \sum_{l=1}^{n} \lambda^{n-l} \mathbf{y}_i(l) \hat{d}_j^*(l) \tag{7}$$

where $(\cdot)^*$ stands for complex conjugation. Let $\mathbf{R}_i(n)$ denote the upper triangular Cholesky factor of $\mathbf{\Phi}_i(n)$, i.e., $\mathbf{\Phi}_i(n) = \mathbf{R}_i^H(n)\mathbf{R}_i(n)$. Then the LS solution given in (6) can alternatively be expressed as follows:

$$\mathbf{w}_{i,j}(n) = \mathbf{R}_i^{-1}(n)\mathbf{p}_{i,j}(n) \tag{8}$$

and $\mathbf{p}_{i,j}(n)$ is defined as

$$\mathbf{p}_{i,j}(n) = \mathbf{R}_i^{-H}(n)\mathbf{z}_{i,j}(n). \tag{9}$$

By substituting (6) in (5), we obtain the following expression of the minimum LS error energy for symbol stream j and order i:

$$\mathcal{E}_{i,j}(n) = \alpha_j(n) - \mathbf{w}_{i,j}^H(n)\mathbf{z}_{i,j}(n) \tag{10}$$

[1] The i-th order problem is defined as the one, which corresponds to the $(N+i-1) \times 1$ input data vector $\mathbf{y}_i(n)$.

where $\alpha_j(n) = \sum_{l=1}^{n} \lambda^{n-l}|\hat{d}_j(l)|^2$. Moreover, from (8) and (9), equation (10) is rewritten as

$$\mathcal{E}_{i,j}(n) = \alpha_j(n) - |\mathbf{p}_{i,j}(n)|^2. \tag{11}$$

Let us now define the matrix

$$\mathbf{Q}(n) = \sum_{l=1}^{n} \lambda^{n-l}\hat{\mathbf{d}}(l)\hat{\mathbf{d}}^H(l) = \lambda\mathbf{Q}(n-1)+\hat{\mathbf{d}}(n)\hat{\mathbf{d}}^H(n) \tag{12}$$

where $\hat{\mathbf{d}}(n) = [\hat{d}_1(n), \hat{d}_2(n), \ldots, \hat{d}_M(n)]^T$. It is straightforward to show that

$$\mathcal{E}_{i,j}(n) = q_{j,j}(n) - |\mathbf{p}_{i,j}(n)|^2 \tag{13}$$

where $q_{j,j}(n)$ stands for the (j,j)-th entry of $\mathbf{Q}(n)$. Furthermore, from (7) and (12) it is easily verified that

$$\mathbf{z}_{i,j}^T(n) = [\mathbf{z}_{i-1,j}^T(n), q_{k_{i-1},j}(n)] \tag{14}$$

In order to compute the minimum LS error energies from (13), vector $\mathbf{p}_{i,j}(n)$ must be obtained first. In the following, we show that $\mathbf{p}_{i,j}(n)$ can be order updated very efficiently resulting in significant computational savings of the proposed scheme.

A. Order update of $\mathbf{p}_{i,j}(n)$

Let $\mathbf{p}_{i-1}(n)$ be the "optimum" vector of order $i-1$, which is related to $\mathbf{w}_{i-1}(n)$ via an expression similar to (8). Then, it can be shown that the upper triangular factor $\mathbf{R}_i(n)$ is given by the following order update expression [9]:

$$\mathbf{R}_i(n) = \begin{bmatrix} \mathbf{R}_{i-1}(n) & \mathbf{p}_{i-1}(n) \\ \mathbf{0}^T & \sqrt{\mathcal{E}_{i-1}(n)} \end{bmatrix} \tag{15}$$

where $\mathcal{E}_{i-1}(n)$ is the minimum LS error energy of order $i-1$. The last expression is the result of the particular characteristics of the LS problems defined in Section II. More specifically, it is easily verified from (4) and (5), that the first $(N+i-2)$ elements of the input data vector of the i-th order problem are identical to the input data vector of the $(i-1)$-th order problem, while its last element coincides with $\hat{d}_{k_{i-1}}(n)$. Since $\mathbf{R}_i(n)$ is the upper triangular factor in a QR decomposition of the i-th order input data matrix, the expression given in (15) is easily derived. From (9), (14), and (15), we get

$$\mathbf{p}_{i,j}(n) = \begin{bmatrix} \mathbf{R}_{i-1}^{-H}(n) & \mathbf{0} \\ -\frac{1}{\sqrt{\mathcal{E}_{i-1}(n)}}\mathbf{p}_{i-1}^H(n)\mathbf{R}_{i-1}^{-H}(n) & \frac{1}{\sqrt{\mathcal{E}_{i-1}(n)}} \end{bmatrix} \begin{bmatrix} \mathbf{z}_{i-1,j}(n) \\ q_{k_{i-1},j}(n) \end{bmatrix}$$

or

$$\mathbf{p}_{i,j}(n) = \begin{bmatrix} \mathbf{p}_{i-1,j}(n) \\ \frac{q_{k_{i-1},j}(n)-\mathbf{p}_{i-1}^H(n)\mathbf{p}_{i-1,j}(n)}{\sqrt{\mathcal{E}_{i-1}(n)}} \end{bmatrix}. \tag{16}$$

Having computed matrix $\mathbf{Q}(n)$ from (12), and $\mathbf{p}_{i,j}(n)$ from (16) for all $j \in S_i$, the LS error energies $\mathcal{E}_{i,j}(n)$, $j \in S_i$ given in (13) can be very efficiently obtained from

$$\mathcal{E}_{i,j}(n) = \mathcal{E}_{i-1,j}(n) - |[\mathbf{p}_{i,j}(n)]_{N+i-1}|^2 \tag{17}$$

where $[\mathbf{p}_{i,j}(n)]_{N+i-1}$ is the last element of $\mathbf{p}_{i,j}(n)$. The minimum of these energies is denoted as $\mathcal{E}_i(n)$ and the corresponding vector as $\mathbf{p}_i(n)$.

B. Computation of $\hat{d}_{k_i}(n)$

So far in our analysis, we have assumed that the detected symbols at time n are available. However, we see from (2) that this requires knowledge of the optimum filter at time n. To overcome this problem, we assume, as in [7], that the detected symbols at time n are computed using the optimum vector and detection order at time $n-1$, i.e.,

$$\bar{d}_{k_i}(n) = \mathbf{w}_i^H(n-1)\mathbf{y}_i(n), \ \ \hat{d}_{k_i}(n) = f\{\bar{d}_{k_i}(n)\} \tag{18}$$

for $i = 1, \ldots, M$, where k_i refers to the detection order at time $n-1$. Using the proposed alternative parameterization and (2), (8), the detected symbol is expressed as

$$\hat{d}_{k_i}(n) = f\{\bar{d}_{k_i}(n)\} = f\{\mathbf{p}_i^H(n-1)\mathbf{g}_i(n)\} \tag{19}$$

where $\mathbf{g}_i(n)$ is defined as

$$\mathbf{g}_i(n) = \mathbf{R}_i^{-H}(n-1)\mathbf{y}_i(n). \tag{20}$$

Substituting the inverse Cholesky factor in the last equation as in (16), and using the relation $\mathbf{y}_i(n) = [\mathbf{y}_{i-1}^T(n), \hat{d}_{k_{i-1}}(n)]^T$, it is easily shown that

$$\mathbf{g}_i(n) = \begin{bmatrix} \mathbf{g}_{i-1}(n) \\ \frac{\hat{d}_{k_{i-1}}(n)-\mathbf{p}_{i-1}^H(n-1)\mathbf{g}_{i-1}(n)}{\sqrt{\mathcal{E}_{i-1}(n-1)}} \end{bmatrix} \tag{21}$$

or

$$\mathbf{g}_i(n) = \begin{bmatrix} \mathbf{g}_{i-1}(n) \\ \frac{\hat{d}_{k_{i-1}}(n)-\bar{d}_{k_{i-1}}(n)}{\sqrt{\mathcal{E}_{i-1}(n-1)}} \end{bmatrix}. \tag{22}$$

Thus, $\mathbf{g}_i(n)$ can be very efficiently order updated, provided that $\mathbf{g}_1(n)$ is available.

C. Initial time-update recursions

To complete the proposed algorithm, first order quantities must be computed, and more specifically vectors $\mathbf{p}_{1,j}(n)$ for $j = 1, 2, \ldots, M$. This is accomplished through the time update recursions described below. Let us assume that $\mathbf{R}_1^{-1}(n-1)$ has been calculated in the previous time instant. Then

$$\mathbf{g}_1(n) = \mathbf{R}_1^{-H}(n-1)\mathbf{y}(n). \tag{23}$$

Next, we produce a sequence of N elementary complex Givens rotation matrices, whose product is denoted by $\mathbf{T}(n)$, according to the following expression:

$$\mathbf{T}(n)\begin{bmatrix} -\frac{\mathbf{g}_1(n)}{\sqrt{\lambda}} \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \star \end{bmatrix} \tag{24}$$

where $\star$ denotes a 'don't care' element. The k-th elementary matrix, $k = 1, 2, \ldots, N$, annihilates the k-th element of $-\frac{\mathbf{g}_1(n)}{\sqrt{\lambda}}$ with respect to the last element of the whole vector,

TABLE I

SQUARE-ROOT ADAPTIVE V-BLAST ALGORITHM

```
1) Compute $\mathbf{g}_1(n)$, $\mathbf{T}(n)$ and $\mathbf{R}_1^{-1}(n)$ from (23), (24) and (25) respectively. $(\underline{3M^2,\ 3M^2/2})$
2) Compute decisions $\hat{d}_{k_i}(n)$, for $i = 1, 2, \ldots, M$ from (19), and (22). $(\underline{3M^2/2,\ 3M^2/2})$
3) Time update $\mathbf{Q}(n)$ from (12). $(\underline{2M^2,\ 0})$
4) For $i = 1, 2, \ldots, M$
   a) Compute $\mathbf{p}_{i,j}(n)$, $\forall j \in S_i$ from (26) for $i = 1$ and from (16) for $i > 1$. $(\underline{2M^3/3 + 9M^2/2,\ 2M^3/3 + M^2})$
   b) Evaluate $\mathcal{E}_{i,j}(n)$, $\forall j \in S_i$ from (13) for $i$=1 and from (17) for $i$>1. Set $\mathcal{E}_i(n)$ the minimum and $\mathbf{p}_i(n)$ the corresponding $\mathbf{p}_{i,j}(n)$ vector. $(\underline{3M^2/2,\ 3M^2/2})$
5) Go to step 1.

Initialization: $k_i$=$i$ $\forall i$, $\mathbf{R}^{-1}(0)$=$\delta^{-1}\mathbf{I}$, $\mathbf{Q}(0)$=$\mathbf{0}$, $\mathbf{p}_{1,j}(0)$=$\mathbf{1}$, for $j$=$1, 2, \ldots, M$, $\mathbf{p}_i(0) = \delta\mathbf{1}$, $\mathcal{E}_i(0)$=0, for $i$=$1, 2, \ldots, M$. $\delta$ is a small positive constant.
```

which initially equals 1.[2] It can be shown [9] that the same rotation matrices can be used for time updating the inverse Cholesky factor as follows:[3]

$$\mathbf{T}(n) \begin{bmatrix} \lambda^{-1/2}\mathbf{R}_1^{-H}(n-1) \\ \\ \mathbf{0}^T \end{bmatrix} = \begin{bmatrix} \mathbf{R}_1^{-H}(n) \\ \\ \star \end{bmatrix}. \qquad (25)$$

Moreover, and most importantly, matrix $\mathbf{T}(n)$ also updates in time $\mathbf{p}_{1,j}(n)$ for $j = 1, 2, \ldots, M$, i.e., [9]

$$\mathbf{T}(n) \begin{bmatrix} \lambda^{1/2}\mathbf{p}_{1,j}(n-1) \\ \\ \hat{d}_j^*(n) \end{bmatrix} = \begin{bmatrix} \mathbf{p}_{1,j}(n) \\ \\ \star \end{bmatrix} \qquad (26)$$

Note that it is not necessary to compute matrix $\mathbf{T}(n)$ explicitly. Only the rotation parameters are calculated from (24), and are then used to update the respective quantities in (25) and (26).

D. Algorithm

The basic steps of the proposed equalization algorithm are summarized in Table I. In the initial training phase, step 2 of the algorithm is not executed and the respective decisions are replaced by a known training sequence. After convergence, the algorithm switches to the decision-directed mode, in which the decisions are computed as described in step 2.

The new algorithm is mathematically equivalent to the algorithm proposed in [7], offering at the same time significant computational savings. Note that, to the best of our knowledge, the algorithm described in [7] was the fastest V-BLAST type equalization scheme. The required operations for each step of the new algorithm are provided in brackets in Table I. The overall computational complexity of the two equalization schemes in terms of the number of complex multiplications and additions[4] is shown in Table II for the case $M = N$. The reduction in computational complexity in favor of the proposed scheme is increased as the number of transmit-receive antennas increases. This reduction is clearly shown in Table III for different values of M. Notice that the new algorithm also requires the computation of M square roots in each iteration.

TABLE II

COMPARISON OF COMPLEXITIES

	Complex Multiplications	Complex Additions
Algorithm of [7]	$2M^3+\frac{19}{2}M^2+O(M)$	$\frac{4}{3}M^3+4M^2+O(M)$
Proposed Algorithm	$\frac{2}{3}M^3+\frac{25}{2}M^2+O(M)$	$\frac{2}{3}M^3+\frac{11}{2}M^2+O(M)$

TABLE III

PERCENTAGE REDUCTION IN COMPUTATIONAL COMPLEXITY

	Complex Multiplications	Complex Additions
$M = 4$	13.3%	12.5%
$M = 8$	30%	26.1%
$M = 12$	39%	32.5%

Moreover, the proposed algorithm is expected to have enhanced numerical robustness, since it stems from the Cholesky factorization of the input data correlation matrix. An explanation of this fact, which is also verified in the simulations, is as follows. Assuming a time-invariant MIMO channel $\mathbf{H}$ and symbol sequences $\{d_j(n)\}$, $j = 1, 2, \ldots, M$, independent between each other and with the noise sequence, it is easily shown from (1) and (4) that the input autocorrelation matrix of the i-th DFE is given as follows:

$$\mathbf{\Phi}_i = E[\mathbf{y}_i(n)\mathbf{y}_i^H(n)] = \sigma_d^2 \begin{bmatrix} \mathbf{H}\mathbf{H}^H + \frac{\sigma_v^2}{\sigma_d^2}\mathbf{I}_N & \mathbf{H}_{i-1} \\ \mathbf{H}_{i-1}^H & \mathbf{I}_{i-1} \end{bmatrix} \qquad (27)$$

where σ_d^2, σ_v^2 stand for the symbol and noise variances respectively, $\mathbf{I}_k$ is the $k \times k$ identity matrix, and $\mathbf{H}_{i-1}$ is a $N \times (i-1)$ matrix, whose columns are the columns of $\mathbf{H}$ that correspond to the $i-1$ previously detected symbols. It can be proven that in the absence of noise (i.e., $\sigma_v^2 = 0$), the $(N+i-1) \times (N+i-1)$ autocorrelation matrix $\mathbf{\Phi}_i$ is rank deficient, with a rank equal to M for every i, $i = 1, 2, \ldots, M$. As a result, for medium to high SNRs, $\mathbf{\Phi}_i$ possesses in general a high condition number, a fact which explains why numerical problems appear in adaptive algorithms dealing directly with $\mathbf{\Phi}_i$, as the one in [7]. When the Cholesky factor $\boldsymbol{R}_i$ is used instead of $\mathbf{\Phi}_i$, the condition number of the problem equals to the square-root of that of the original autocorrelation matrix. Thus, the resulting square-root adaptive algorithms are expected to be much more numerically robust.

IV. SIMULATIONS

In this section, the performance of the proposed equalizer is evaluated through computer simulations. Initially, the numerical robustness of the new algorithm is verified by the

[2] In a vector rotation $\begin{bmatrix} c & -s^* \\ s & c \end{bmatrix} \begin{bmatrix} b \\ a \end{bmatrix} = \begin{bmatrix} 0 \\ d \end{bmatrix}$, the rotation parameters are evaluated as c=$\frac{|a|}{\sqrt{|a|^2+|b|^2}}$ and s=$\frac{b^*}{\sqrt{|a|^2+|b|^2}}\frac{a}{|a|}$.

[3] Even though the analysis in [9] is done for real signals, it can be extended to complex signals in a rather straightforward manner.

[4] In Table I(a) of [7] the number of complex multiplications and additions are calculated as $\frac{4}{3}M^3 + 7M^2$ and $\frac{4}{3}M^3 + 5M^2$ respectively. We believe, however, that careful counting leads to the figures of Table II.

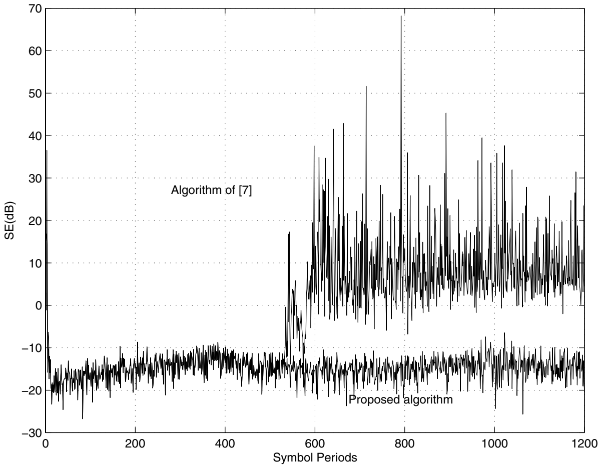

Fig. 2. Squared Error Curves

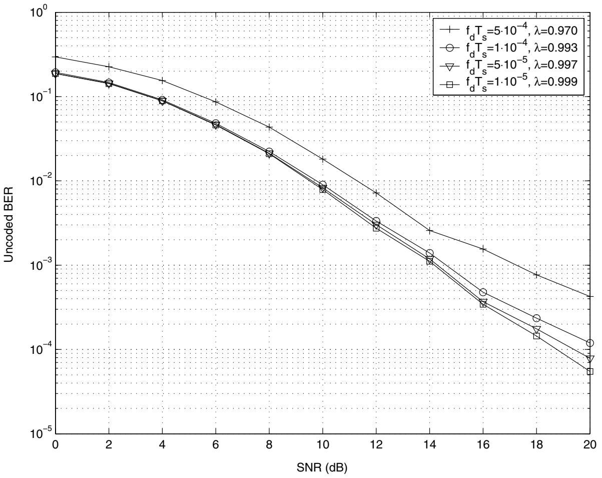

Fig. 3. BER curves for a 4×4 system.

squared error curves shown in Fig. 2, which correspond to the instantaneous squared error at the output of the equalizers, averaged over all DFEs. A MIMO system with $M = N = 4$ was used in our simulations, whose input independent streams were taken from a QPSK sequence. A number of 30 symbol periods was used for training, and the SNR was set to 16dB. A single Rayleigh MIMO channel realization was generated according to the Jakes model, having a normalized Doppler frequency of $f_dT_s = 5\cdot10^{-4}$, where f_d stands for the Doppler frequency, and T_s was the symbol period. To track system variations, a forgetting factor $\lambda = 0.97$ was employed.

We observe from Fig. 2 that both algorithms converge very fast, and have identical performance for a number of iterations. However, after about 500 iterations, the algorithm of [7] starts diverging due to accumulation of numerical errors. On the contrary, the proposed algorithm keeps tracking the underlying system, and retains a numerically robust performance for the whole simulation period. Notice that lower steady state squared error could be achieved after convergence of both algorithms by reducing parameter λ. In such a case, however, divergence of the algorithm of [7] would occur earlier.

To study the effect of channel variations, the performance of the new algorithm was evaluated in terms of uncoded bit error rate (BER) for different fading rates. Again, a Rayleigh 4×4 MIMO channel was simulated for different values of f_dT_s. Assuming a system operating at the 2.4GHz frequency band and having a symbol period of 0.1 to $0.5\mu sec$, then for a mobile velocity of 60 to 100Km/h, a normalized Doppler frequency in the range of $1\cdot10^{-5}$ to $5\cdot10^{-4}$ arises. Thus, four distinct values of f_dT_s were tested, and BER measurements were conducted over 15000 channel realizations. For each channel realization, a packet of duration $512T_s$ was transmitted, while the first 30 symbol periods were used for training. As indicated by the BER curves of Fig. 3, the equalizer tracks effectively channel variations. However, increase of the fading rate inevitably affects system's performance, and a degradation of around 1dB can is observed for fast time varying channels.

V. Conclusions

A new square-root adaptive V-BLAST receiver for flat and fast fading MIMO channels has been proposed. Under fast fading conditions, channel estimation should be performed quite frequently to ensure V-BLAST receiver's optimum performance. An adaptive receiver, as the one described in this paper, can avoid this computational load, reduce training overhead, and keep tracking channel variations. The new adaptive algorithm stems from the Cholesky factorization of the equalizer's input autocorrelation matrix, and is mathematically equivalent to a recently proposed adaptive V-BLAST scheme [7]. To the best of our knowledge, these two algorithms are the only V-BLAST schemes that update efficiently in time both the equalizer taps and symbol detection order. However, compared to [7], the proposed algorithm offers not only substantially reduced computational complexity, but also enhanced numerical robustness.

References

[1] G. J. Foschini, G. D. Golden, R. A. Valenzuela, and P. W. Wolniansky, "Simplified processing for high spectral efficiency wireless communications employing multi-element arrays," *IEEE J. Select. Areas Commun.*, vol. 17, pp. 1841–1852, Nov. 1999.

[2] B. Hassibi, "An efficient square-root algorithm for BLAST," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, Istanbul, June 2000, pp. 737–740.

[3] W. Zha and S. Blostein, "Modified decorrelating decision-feedback detection of BLAST space-time system," in *Proc. IEEE Int. Conf. Communications*, Helsinki, June 2002, pp. 737–740.

[4] J. Benesty, Y. Huang, and J. Chen, "A fast recursive algorithm for optimum sequential signal detection in a BLAST system," *IEEE Trans. Signal Processing*, vol. 51, no. 7, pp. 1722–1730, July 2003.

[5] H. Zhu, Z. Lei, and F. Chin, "An improved square-root algorithm for BLAST," *IEEE Signal Processing Lett.*, vol. 11, no. 9, pp. 772–775, Sept. 2004.

[6] G. Ginis and J. M. Cioffi, "On the relation between V-BLAST and the GDFE," *IEEE Commun. Lett.*, vol. 5, pp. 364–366, 2001.

[7] J. Choi, H. Yu, and Y. H. Lee, "Adaptive MIMO decision feedback equalization for receivers in time-varying channels," in *Proc. IEEE Vehicular Technology Conference,*, Jesu, Korea, Spring 2003, pp. 1851–1856.

[8] S. Haykin, *Adaptive Filter Theory*. New Jersey: Prentice-Hall Inc., 2002.

[9] A. A. Rontogiannis and S. Theodoridis, "New fast QR decomposition least squares adaptive algorithms," *IEEE Trans. Signal Processing*, vol. 46, no. 8, pp. 2113–2121, Aug. 1998.