A FAST RECURSIVE ALGORITHM FOR OPTIMUM SEQUENTIAL SIGNAL DETECTION IN A BLAST SYSTEM

Jacob Benesty, Yiteng (Arden) Huang, and Jingdong Chen

Bell Laboratories, Lucent Technologies
700 Mountain Avenue
Murray Hill, NJ 07974, USA
E-mail: {jb, arden, jingdong}@research.bell-labs.com

ABSTRACT

BLAST (Bell Laboratories layered space-time) wireless systems are multiple-antenna communication schemes which can achieve very high spectral efficiencies in scattering environments, with no increase in bandwidth or transmitted power. The most popular and, by far, the most practical architecture is the so-called vertical BLAST (V-BLAST). The signal detection algorithm of a V-BLAST system is computationally very intensive. If the number of transmitters is M and is equal to the number of receivers, this complexity is proportional to M^4 at each sample time. In this paper, we propose a simple and very efficient algorithm that reduces the complexity by a factor of M.

1. INTRODUCTION

Telatar [1] and Foschini [2] showed that the multipath wireless channel is capable of huge capacities, provided that the multipath scattering is sufficiently rich and is properly exploited through the use of an appropriate processing architecture and multiple antennas (both at transmission and reception). The original architecture proposed in [2] and called D-BLAST (diagonally-Bell Laboratories layered space-time) is theoretically capable of approaching the Shannon capacity for multiple transmitters and receivers but it's very complex to implement. A simplified version known as vertical BLAST (V-BLAST) was proposed in [3], [4] that can still achieve a substantial portion of that capacity. For example, the authors in [3] have demonstrated, using a laboratory prototype and in an indoor environment, spectral efficiencies of 20–40 bps/Hz at average signal-to-noise ratios ranging from 24 to 34 dB. In the rest, we will not make any distinctions between the terms BLAST and V-BLAST.

In a V-BLAST system, a data stream is split into M uncorrelated sub-streams, each of which is transmitted by one of the M transmitting antennas. The V-BLAST algorithm detects the M symbols, at the receiver, in M iterations and it is proven in [3] that the decoding order of this algorithm is optimal from a performance point of view. However, as it will be shown later, the complexity required to achieve this performance is very high.

This paper is organized as follows. Section 2 defines the signal model. In Section 3, we explain in detail the V-BLAST algorithm. In Section 4, we show how to derive a fast algorithm for BLAST. Finally, Section 5 evaluates the complexity of different algorithms.

2. SIGNAL MODEL

The BLAST architecture is a multiple-input multiple-output (MIMO) system where a single user uses a communication link comprising M transmitting antennas and N receiving antennas in a flat-fading environment (meaning that the signals are narrowband). At the receivers, at the sample time k, we have:

$$\begin{aligned} \mathbf{x}(k) &= \sum_{m=1}^{M} \mathbf{h}_{:m} s_m(k) + \mathbf{w}(k) \\ &= \mathbf{H}\mathbf{s}(k) + \mathbf{w}(k), \quad k = 1, 2, ..., K, \end{aligned} \tag{1}$$

where

$$\begin{aligned} \mathbf{x}(k) &= \begin{bmatrix} x_1(k) & x_2(k) & \cdots & x_N(k) \end{bmatrix}^T \\ &= \begin{bmatrix} \mathbf{h}_{1:}^H \mathbf{s}(k) + w_1(k) & \cdots & \mathbf{h}_{N:}^H \mathbf{s}(k) + w_N(k) \end{bmatrix}^T \end{aligned}$$

is the N-dimensional received vector,

$$\begin{aligned} \mathbf{H} &= \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1M} \\ h_{21} & h_{22} & \cdots & h_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ h_{N1} & h_{N2} & \cdots & h_{NM} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{h}_{:1} & \mathbf{h}_{:2} & \cdots & \mathbf{h}_{:M} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{h}_{1:}^H \\ \mathbf{h}_{2:}^H \\ \vdots \\ \mathbf{h}_{N:}^H \end{bmatrix} \end{aligned}$$

is an $N \times M$ complex matrix assumed to be constant for K symbol periods, vectors $\mathbf{h}_{n:}$ and $\mathbf{h}_{:m}$ are respectively of length M and N,

$$\mathbf{s}(k) = \begin{bmatrix} s_1(k) & s_2(k) & \cdots & s_M(k) \end{bmatrix}^T$$

is the M-dimensional transmitted vector,

$$\mathbf{w}(k) = \begin{bmatrix} w_1(k) & w_2(k) & \cdots & w_N(k) \end{bmatrix}^T$$

is a zero-mean complex additive white Gaussian noise (AWGN) vector with covariance:

$$\mathbf{R}_{ww} = E\{\mathbf{w}(k)\mathbf{w}^H(k)\} = \sigma_w^2 \mathbf{I}_{N\times N}, \tag{2}$$

and ^{T} and ^{H} denote respectively transpose and conjugate transpose of a matrix or a vector.

The transmitted vector $\mathbf{s}(k)$ has a total power P_T. This power is held constant regardless of the number of transmitting antennas M and corresponds to the trace of the covariance matrix of the transmitted vector:

$$P_\mathrm{T} = \mathrm{tr}[\mathbf{R}_{ss}] = \text{Constant} = \sum_{m=1}^{M} \sigma_{s_m}^2. \quad (3)$$

In the rest, we suppose that all the antennas transmit with the same power:

$$\sigma_{s_1}^2 = \sigma_{s_2}^2 = \cdots = \sigma_{s_M}^2 = \sigma_s^2,$$

so that:

$$P_\mathrm{T} = M\sigma_s^2. \quad (4)$$

An original information sequence for wireless transmission is demultiplexed into M data sequences $s_m(k),\ m = 1, \cdots, M$ (called substreams) and each one of them is sent through a transmitting antenna. These M substreams are assumed to be uncorrelated, this implies that the covariance matrix of the transmitted vector $\mathbf{s}(k)$ is diagonal:

$$\mathbf{R}_{ss} = E\{\mathbf{s}(k)\mathbf{s}^H(k)\} = \sigma_s^2 \mathbf{I}_{M\times M}. \quad (5)$$

We also suppose the following:

- $N \geq M$.
- $\mathbf{H}$ has full column rank, i.e. $\mathrm{rank}[\mathbf{H}] = M$.

3. THE V-BLAST ALGORITHM

In order to detect the transmitted symbols at the receivers, the complex channel matrix $\mathbf{H}$ needs to be known. In practice, $\mathbf{H}$ is identified by sending a training sequence (known at the reception) at the beginning of each burst. The length of this burst is equal to $K = K_1 + K_2$ symbols where the K_1 symbols are used for training and the K_2 symbols are the data information. The propagation coefficients are assumed to be constant during a whole burst, after which they change to new independent random values which they maintain for another K symbols, and so on. In the remainder of this paper, we will not make the distinction between $\mathbf{H}$ and its estimate.

The first step of the V-BLAST algorithm [3] makes use of the pseudo-inverse of the channel matrix $\mathbf{H}$ or the minimum mean-square error (MMSE) filter $\mathbf{G}$.

Define the error vector signal at time k between the input $\mathbf{s}(k)$ and its estimate:

$$\mathbf{e}(k) = \mathbf{s}(k) - \mathbf{y}(k) = \mathbf{s}(k) - \mathbf{G}^H\mathbf{x}(k). \quad (6)$$

Now, let us define the error criterion:

$$J = E\{\mathbf{e}^H(k)\mathbf{e}(k)\} = \mathrm{tr}\left[E\{\mathbf{e}(k)\mathbf{e}^H(k)\}\right]. \quad (7)$$

The minimization of (7) leads to the Wiener-Hopf equation:

$$\mathbf{G}^H\mathbf{R}_{xx} = \mathbf{R}_{sx}, \quad (8)$$

where

$$\mathbf{R}_{xx} = E\{\mathbf{x}(k)\mathbf{x}^H(k)\} \quad (9)$$

is the output signal covariance matrix, and

$$\mathbf{R}_{sx} = E\{\mathbf{s}(k)\mathbf{x}^H(k)\} \quad (10)$$

is the cross-correlation matrix between the input and output signals.

From expression (8), we find that the MMSE filter is:

$$\mathbf{G} = \left[\mathbf{H}\mathbf{H}^H + \alpha\mathbf{I}_{N\times N}\right]^{-1}\mathbf{H}, \quad (11)$$

where

$$\alpha = \frac{\sigma_w^2}{\sigma_s^2}. \quad (12)$$

It can easily be seen that (11) is equivalent to:

$$\mathbf{G} = \mathbf{H}\left[\mathbf{H}^H\mathbf{H} + \alpha\mathbf{I}_{M\times M}\right]^{-1} = \mathbf{H}\mathbf{Q}. \quad (13)$$

The second form [eq. (13)] is more useful and more efficient in practice since $M \leq N$ and the size of the matrix to invert in (13) is smaller or equal than the size of the matrix to invert in (11).

Instead of the MMSE filter, we can use directly the pseudo-inverse of $\mathbf{H}$ which is:

$$\mathbf{G}_\mathrm{PI}^H = \left[\mathbf{H}^H\mathbf{H}\right]^{-1}\mathbf{H}^H. \quad (14)$$

The only difference between the expressions $\mathbf{G}$ and $\mathbf{G}_\mathrm{PI}$ is that the first one is "regularized" by a diagonal matrix $\alpha\mathbf{I}_{M\times M}$ while the second one is not. This regularization introduces a bias but (13) gives a much more reliable result than (14) when the matrix $\mathbf{H}^H\mathbf{H}$ is ill-conditioned and the estimation of the channel is noisy. In practice, depending on the condition number of the matrix $\mathbf{H}^H\mathbf{H}$, we can take a different value for α than the one given in (12). For example, if this condition number is very high and the SNR is also high, it will be better to take a higher value for α. Thus, the MMSE filter can be seen as a biased pseudo-inverse of $\mathbf{H}$.

In the V-BLAST algorithm, the detection of the symbols $s_m(k)$ is done in M iterations. The order in which the components of $\mathbf{s}(k)$ are detected is important to the overall performance of the system. Let the ordered set

$$\mathcal{S} = \{p_1, p_2, \cdots, p_M\} \quad (15)$$

be a permutation of the integers $1, 2, \cdots, M$ specifying the order in which components of the transmitted symbol vector $\mathbf{s}(k)$ are extracted. The first iteration, which is also the initialization, is performed in three steps (as well as the other iterations):

Step 1: Using the MMSE filter or the pseudo-inverse, we compute:

$$\mathbf{y}(k) = \mathbf{G}^H\mathbf{x}(k). \quad (16)$$

Step 2: The element of $\mathbf{y}(k)$ with the highest SNR is detected. This element is associated with the smallest diagonal entry of $\mathbf{Q}$ for the MMSE filter (as explained in the next section) or the column of $\mathbf{G}$ having the smallest norm for the pseudo-inverse (zero-forcing) [3]. If such a column is p_1, we get:

$$\hat{s}_{p_1}(k) = \mathcal{Q}\left[y_{p_1}(k)\right], \quad (17)$$

with $\mathcal{Q}[\cdot]$ indicating the slicing or quantization procedure according to the constellation in use.

Step 3: Assuming that $\hat{s}_{p_1}(k) = s_{p_1}(k)$, we cancel $s_{p_1}(k)$ from the received vector $\mathbf{x}(k)$, resulting in a modified received vector:

$$\begin{aligned}\mathbf{x}_2(k) &= \mathbf{x}(k) - \hat{s}_{p_1}(k)\mathbf{h}_{:p_1} = \sum_{m \neq p_1} \mathbf{h}_{:m} s_m(k) + \mathbf{w}(k) \\ &= \mathbf{H}_{M-1}\mathbf{s}_{M-1}(k) + \mathbf{w}(k), \quad (18)\end{aligned}$$

where $\mathbf{H}_{M-1}$ is an $N \times (M-1)$ matrix derived from $\mathbf{H}$ by removing its p_1-th column and $\mathbf{s}_{M-1}(k)$ is a vector of length $M-1$ obtained from $\mathbf{s}(k)$ by removing its p_1-th component.

Steps 1–3 are then performed for components $p_2, \cdots, p_M$ by operating in turn on the progression of modified received vectors $\mathbf{x}_2(k), \cdots, \mathbf{x}_M(k)$. Note that at the m-th iteration, we will obtain the $N \times (M-m)$ matrix $\mathbf{H}_{M-m}$ which can be derived from $\mathbf{H}$ by removing m of its columns: $p_1, \cdots, p_m$. As shown in [3], this ordering (choosing the best SNR at each iteration in the detection process) is optimal among all possible orderings.

The arithmetic complexity of the V-BLAST algorithm is very high. This complexity is in $\mathcal{O}(NM^3)$ for each sample time k.

4. A FAST V-BLAST ALGORITHM

Here, the matrix $\mathbf{G}$ is not computed directly. Since this matrix is the product of a rectangular matrix of size $N \times M$ and a square matrix of size $M \times M$, the complexity of such a product is proportional to NM^2 at each iteration. The algorithm requires M iterations, therefore the complexity is in $\mathcal{O}(NM^3)$ even if the matrices are deflated by 1 at each iteration.

Recall that:

$$\mathbf{G} = \mathbf{H}\mathbf{R}^{-1}, \quad (19)$$

where

$$\mathbf{R} = \mathbf{H}^H\mathbf{H} + \alpha\mathbf{I}_{M\times M}. \quad (20)$$

The covariance matrix of the error signal, $\mathbf{e}(k) = \mathbf{s}(k) - \mathbf{y}(k)$, is:

$$\mathbf{R}_{ee} = E\{\mathbf{e}(k)\mathbf{e}^H(k)\} = \sigma_w^2\mathbf{R}^{-1} = \sigma_w^2\mathbf{Q}. \quad (21)$$

Clearly, the element of $\mathbf{y}(k)$ with the highest SNR is the one with the smallest error variance, so that:

$$p_1 = \arg\min_m q_{mm}, \quad (22)$$

where q_{mm} are the diagonal elements of the matrix $\mathbf{Q} = \mathbf{R}^{-1}$.

The matrix $\mathbf{R}$ can be rewritten as follows:

$$\mathbf{R} = \sum_{n=1}^{N} \mathbf{h}_{n:}\mathbf{h}_{n:}^H + \alpha\mathbf{I}_{M\times M}, \quad (23)$$

which means that $\mathbf{R}$ can be computed recursively in N iterations:

$$\begin{aligned}\mathbf{R}_{[l]} &= \sum_{n=1}^{l} \mathbf{h}_{n:}\mathbf{h}_{n:}^H + \alpha\mathbf{I}_{M\times M} = \mathbf{R}_{[l-1]} + \mathbf{h}_{l:}\mathbf{h}_{l:}^H, \quad (24) \\ \mathbf{R}_{[N]} &= \mathbf{R}, \ \mathbf{R}_{[0]} = \alpha\mathbf{I}_{M\times M}. \quad (25)\end{aligned}$$

Using the Sherman-Morrison formula, $\mathbf{Q}$ can also be computed recursively:

$$\mathbf{Q}_{[l]} = \mathbf{Q}_{[l-1]} - \frac{\mathbf{Q}_{[l-1]}\mathbf{h}_{l:}\mathbf{h}_{l:}^H\mathbf{Q}_{[l-1]}}{1 + \mathbf{h}_{l:}^H\mathbf{Q}_{[l-1]}\mathbf{h}_{l:}}. \quad (26)$$

With the initialization $\mathbf{Q}_{[0]} = \frac{1}{\alpha}\mathbf{I}_{M\times M}$, we obtain $\mathbf{Q}_{[N]} = \left[\mathbf{H}^H\mathbf{H} + \alpha\mathbf{I}_{M\times M}\right]^{-1}$. Note that if we start the process at iteration $M+1$ with the initialization $\mathbf{Q}_{[M]} = \sum_{n=1}^{M}\mathbf{h}_{n:}\mathbf{h}_{n:}^H$, we obtain $\mathbf{Q}_{[N]} = \left[\mathbf{H}^H\mathbf{H}\right]^{-1}$. Before going further, it is important to comment on expression (26). Indeed, it is well known that the computation of any recursion introduces numerical instabilities because of the finite precision of the processor units. This instability occurs only after a very large number of iterations. Fortunately in this application, the number of iterations to compute $\mathbf{Q}$ is limited by the number of receiving antennas (N), which is rather small; so in principle we should not expect any particular problem here. In any case, the numerical stability can be improved by increasing α at the initialization. Furthermore, as it will become clearer in the following, we can use any method to compute $\mathbf{Q}$ and still have a very efficient algorithm.

In the proposed algorithm, (26) is computed only one time at the first iteration. The complexity to compute $\mathbf{Q}_{[N]}$ is in $\mathcal{O}(NM^2)$. Once $\mathbf{Q}_{[N]}$ is computed, it's easy to determine p_1 from (22). Continuing the process for this first iteration, the input estimate is computed as follows:

$$\begin{aligned}y_{p_1}(k) &= \sum_{m=1}^{M} q_{p_1 m}\mathbf{h}_{:m}^H\mathbf{x}(k), \quad (27) \\ \hat{s}_{p_1}(k) &= \mathcal{Q}\left[y_{p_1}(k)\right]. \quad (28)\end{aligned}$$

The last step (Step 3) is the same as the one for the V-BLAST algorithm.

For the following iterations, the process is different. We show that the matrix $\mathbf{Q}$ can be deflated recursively. We have:

$$\mathbf{Q}_{[N]} = \mathbf{Q} = \left[\mathbf{H}^H\mathbf{H} + \alpha\mathbf{I}_{M\times M}\right]^{-1} = \mathbf{R}^{-1} \quad (29)$$

$$= \begin{bmatrix} \mathbf{h}_{:1}^H\mathbf{h}_{:1} + \alpha & \mathbf{h}_{:1}^H\mathbf{h}_{:2} & \cdots & \mathbf{h}_{:1}^H\mathbf{h}_{:M} \\ \mathbf{h}_{:2}^H\mathbf{h}_{:1} & \mathbf{h}_{:2}^H\mathbf{h}_{:2} + \alpha & \cdots & \mathbf{h}_{:2}^H\mathbf{h}_{:M} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{h}_{:M}^H\mathbf{h}_{:1} & \mathbf{h}_{:M}^H\mathbf{h}_{:2} & \cdots & \mathbf{h}_{:M}^H\mathbf{h}_{:M} + \alpha \end{bmatrix}^{-1}.$$

After p_1 corresponding to the element $y_{p_1}(k)$ with the smallest variance is determined, we can interchange the p_1-th and M-th entries of the transmitted signal $\mathbf{s}(k)$ such that the M-th signal is currently the best estimate. Of course, the indices of the transmitted signals will be tracked after the reordering. Accordingly, the p_1-th and M-th columns of the channel matrix $\mathbf{H}$ should be interchanged which can be easily done by post-multiplying $\mathbf{H}$ with a permutation matrix $\mathbf{P}_{p_1 M}$. Since

$$\begin{aligned}&\left(\mathbf{H}\mathbf{P}_{p_1 M}\right)^H\left(\mathbf{H}\mathbf{P}_{p_1 M}\right) + \alpha\mathbf{I}_{M\times M} \\ &= \mathbf{P}_{p_1 M}\left(\mathbf{H}^H\mathbf{H} + \alpha\mathbf{I}_{M\times M}\right)\mathbf{P}_{p_1 M}, \quad (30)\end{aligned}$$

it follows that the rows and columns p_1 and M of the matrix $\mathbf{R}$ should be permuted. Equivalently, we can permute the rows and columns p_1 and M of the matrix $\mathbf{Q}$ which can easily be seen from

$$\left(\mathbf{P}_{p_1 M}\mathbf{R}\mathbf{P}_{p_1 M}\right)^{-1} = \mathbf{P}_{p_1 M}\mathbf{R}^{-1}\mathbf{P}_{p_1 M} = \mathbf{P}_{p_1 M}\mathbf{Q}\mathbf{P}_{p_1 M}. \quad (31)$$

These permutations will allow us to remove the effect of the channel $\mathbf{h}_{:p_1}$ easily. In this case, we have:

$$\mathbf{Q}_M = \begin{bmatrix} \mathbf{R}_{M-1} & \mathbf{v}_{M-1} \\ \mathbf{v}_{M-1}^H & \beta_{p_1} \end{bmatrix}^{-1}, \quad (32)$$

where

$$\beta_{p_1} = \mathbf{h}_{:p_1}^H \mathbf{h}_{:p_1} + \alpha,$$
$$\mathbf{v}_{M-1} = \left[\begin{array}{cccc} \mathbf{h}_{:1}^H \mathbf{h}_{:p_1} & \mathbf{h}_{:2}^H \mathbf{h}_{:p_1} & \cdots & \mathbf{h}_{:M-1}^H \mathbf{h}_{:p_1} \end{array} \right]^T,$$
$$\mathbf{R}_{M-1} = \mathbf{H}_{M-1}^H \mathbf{H}_{M-1} + \alpha \mathbf{I}_{(M-1)\times(M-1)}.$$

It can easily be shown that:

$$\mathbf{Q}_M = \left[\begin{array}{cc} \mathbf{T}_{M-1}^{-1} & -\mathbf{T}_{M-1}^{-1}\mathbf{v}_{M-1}/\beta_{p_1} \\ -\mathbf{v}_{M-1}^H \mathbf{T}_{M-1}^{-1}/\beta_{p_1} & \lambda_{p_1} \end{array} \right], \quad (33)$$

where

$$\mathbf{T}_{M-1} = \mathbf{R}_{M-1} - \mathbf{v}_{M-1}\mathbf{v}_{M-1}^H/\beta_{p_1} \quad (34)$$

is the Schur complement of β_{p_1} in $\mathbf{Q}_M^{-1}$ and $\lambda_{p_1} = 1/\beta_{p_1} + \mathbf{v}_{M-1}^H \mathbf{T}_{M-1}^{-1} \mathbf{v}_{M-1}/\beta_{p_1}^2$. Furthermore, from (34) we deduce that:

$$\mathbf{R}_{M-1}^{-1} = \mathbf{Q}_{M-1} = \left[\mathbf{T}_{M-1} + \mathbf{v}_{M-1}\mathbf{v}_{M-1}^H/\beta_{p_1}\right]^{-1} \quad (35)$$

and using the Sherman-Morrison formula, we obtain:

$$\mathbf{Q}_{M-1} = \mathbf{T}_{M-1}^{-1} - \frac{\mathbf{T}_{M-1}^{-1}\mathbf{v}_{M-1}\mathbf{v}_{M-1}^H\mathbf{T}_{M-1}^{-1}}{\beta_{p_1} + \mathbf{v}_{M-1}^H\mathbf{T}_{M-1}^{-1}\mathbf{v}_{M-1}}. \quad (36)$$

Clearly, expression (36) shows that the matrix $\mathbf{Q}$ can be deflated recursively in $\mathcal{O}(M^2)$ at each iteration. In the general case, we have:

$$\mathbf{Q}_{M-m} = \mathbf{T}_{M-m}^{-1} - \frac{\mathbf{T}_{M-m}^{-1}\mathbf{v}_{M-m}\mathbf{v}_{M-m}^H\mathbf{T}_{M-m}^{-1}}{\beta_{p_m} + \mathbf{v}_{M-m}^H\mathbf{T}_{M-m}^{-1}\mathbf{v}_{M-m}}, \quad (37)$$
$$\mathbf{R}_{M-m} = \mathbf{H}_{M-m}^H \mathbf{H}_{M-m} + \alpha \mathbf{I}_{(M-m)\times(M-m)}. \quad (38)$$

Note that $\mathbf{R}_{M-m}$ is not computed but rather easily determined from $\mathbf{R}_{M+1-m}$ by removing its last line and column. Only $\mathbf{R}_M = \mathbf{R}$ is calculated at the first iteration. The complexity of the proposed fast V-BLAST algorithm is in $\mathcal{O}(NM^2 + M^3)$. For $N = M$, the complexity is reduced by a factor of M compared to the V-BLAST algorithm.

5. COMPLEXITY EVALUATION

We now look at the computational complexity of the proposed fast V-BLAST algorithm and compare it to the traditional V-BLAST and the square-root algorithms [5]. Since the transmitted and received signals as well as the channel matrix are complex, all processings are conducted upon complex values. Therefore, unless otherwise specified, multiplications, divisions, and additions refer to complex operations throughout this section.

For the traditional V-BLAST algorithm, the total number of multiplications is

$$\frac{9}{4}M^4 + \frac{4}{3}M^3N + \frac{29}{6}M^3 + \frac{7}{2}M^2N + \mathcal{O}\left(M^2 + MN\right),$$

and the total number of additions is

$$\frac{9}{4}M^4 + \frac{4}{3}M^3N + \frac{25}{6}M^3 + \frac{7}{2}M^2N + \mathcal{O}\left(M^2 + MN\right).$$

If the numbers of transmitting and receiving antennas are the same, i.e. $M = N$, then the total numbers of multiplications and additions are $\frac{43}{12}M^4 + \frac{25}{3}M^3 + \mathcal{O}\left(M^2\right)$ and $\frac{43}{12}M^4 + \frac{23}{3}M^3 + \mathcal{O}\left(M^2\right)$, respectively.

In the square-root algorithm for V-BLAST decoding, the square-root matrix $\mathbf{Q}_{M-m}^{1/2}$ of $\mathbf{Q}_{M-m}$ is recursively computed by using Householder transformations. Applying a Householder transformation to a given matrix with respect to one of its column/row vector requires equal numbers of multiplications and additions. As given in [5], the square-root algorithm requires $\frac{2}{3}M^3 + 7M^2N + 2MN^2 + \mathcal{O}\left(M^2 + MN\right)$ multiplications and additions. If $M = N$, then these numbers turn to $\frac{29}{3}M^3 + \mathcal{O}\left(M^2\right)$. Note that square-root operations were omitted in the evaluation.

For the proposed fast V-BLAST algorithm, the total number of multiplications is

$$\frac{2}{3}M^3 + 3M^2N + \mathcal{O}\left(M^2 + MN\right),$$

and the total number of additions is

$$\frac{1}{2}M^3 + \frac{5}{2}M^2N + \mathcal{O}\left(M^2 + MN\right).$$

If $M = N$, then the proposed fast V-BLAST algorithm requires $\frac{11}{3}M^3 + \mathcal{O}\left(M^2\right)$ multiplications and $3M^3 + \mathcal{O}\left(M^2\right)$ additions. Therefore, the speedups of the proposed algorithm over the traditional V-BLAST in the number of multiplications and additions are $43M/44 + 25/11 \approx M + 2.3$ and $43M/36 + 23/9 \approx 1.2M + 2.6$, respectively. Compared to the square-root algorithm, the proposed algorithm is also more efficient and the speedups in the number of multiplications and additions are $29/11 \approx 2.6$ and $29/6 \approx 4.8$, respectively.

Note that one complex multiplication/division takes 6 floating-point operations (flops) and one complex addition/subtraction needs 2 flops. Therefore, the flop counts of the traditional V-BLAST and the square-root algorithms are approximately $\frac{43}{42}M + \frac{49}{21} \approx M + 2.3$ times and $58/21 \approx 2.76$ times, respectively, more than that of the proposed algorithm in the case of $M = N$.

6. REFERENCES

[1] I. E. Telatar, "Capacity of multi-antenna Gaussian channels," Tech. Rep., Bell Labs, 1995.

[2] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment using multi-element antennas," *Bell Labs Tech. J.*, vol. 1, no. 2, pp. 41–59, 1996.

[3] P. W. Wolniansky, G. J. Foschini, G. D. Golden, and R. A. Valenzuela, "V-BLAST: an architecture for realizing very high data rates over the rich-scattering wireless channel," in *Proc. ISSSE*, 1998, pp. 295–300.

[4] G. D. Golden, C. J. Foschini, R. A. Valenzuela, and P. W. Wolniansky, "Detection algorithm and initial laboratory results using V-BLAST space-time communication architecture," *Electronics Letters*, vol. 35, no. 1, pp. 14–16, Jan. 1999.

[5] B. Hassibi, "A fast square-root implementation for BLAST," in *Conference Record of the Thirty-Fourth Asilomar Conference on Signals, Systems and Computers*, 2000, pp. 1255–1259.