Complexity-Reduced Channel Matrix Inversion for MIMO Systems in Time-Varying Channels

Wei Liu*, Kwonhue Choi†, and Huaping Liu*
*School of EECS, Oregon State University, Corvallis, OR 97331 USA. E-mail: {liu, hliu}@eecs.oregonstate.edu
†Dept. of Information and Communication Engineering, Yeungnam University, Gyeongsan, Korea 712-749. Email: gonew@ynu.ac.kr

***Abstract*—Inversion of channel matrix is required for commonly used detection schemes in multiple-input multiple-output (MIMO) systems. In time-varying fading channels, frequent matrix inversion is computationally intensive for mobile terminals operating at high data rates. Several existing papers have addressed this problem for MIMO orthogonal frequency division multiplexing systems by employing interpolation since the channel coefficients are correlated in the frequency domain. The correlation of the channel in the time domain, i.e., the channel matrices at consecutive symbol intervals vary only slightly, could also be exploited. We propose an algorithm that exploits second-order extrapolation in the time domain to lower the computational complexity of matrix inversion. While existing schemes are mainly designed for linear MIMO detection, the proposed algorithm can be applied for both non-linear detection such as ordered successive interference cancelation (OSIC) and linear detection. The proposed scheme can be efficiently implemented with only addition and integer multiplication. Simulation results of the proposed scheme applied in MIMO OSIC detection demonstrate that it can significantly reduce the matrix inversion complexity while maintaining the system performance.**

***Index Terms*—Multiple-input multiple-output system, time-varying fading channel, matrix inversion, extrapolation.**

I. Introduction

In multiple-input and multiple-output (MIMO) communication systems [1], [2], in order to cancel the amplitude and phase distortion introduced by the wireless fading channel, matrix inversions or weight vector calculation in sequential detection such as ordered successive interference cancelation (OSIC) need to be performed at the receiver. For high-data-rate applications, frequent channel matrix inversion could be computationally very intensive.

Iterative methods such as successive over-relaxation (SOR) [3] could be used for matrix inversion. The number of floating point operations that SOR requires is approximately $k \times N^3$ for an $N \times N$ matrix, where k is the iteration count. This is higher than some direct solution methods such as LU decomposition, which requires $\frac{2}{3}N^3$ floating point operations, especially when k is large (desired to ensure convergence). Although convergence of SOR could be accelerated by letting the iteration start from the accurate inverse of previous channel matrix whose elements are slightly different from those of the current one, the complexity of SOR could still be high since k must be large enough to ensure convergence. Therefore, SOR method is, in general, not appropriate for MIMO channel matrix inversion, especially when the number of antennas is large.

A class of interpolation-based algorithm for computationally efficient matrix inversion in MIMO orthogonal frequency division multiplexing (OFDM) zero-forcing (ZF) receivers is proposed in [4]. A linear interpolation-based MIMO detection is analyzed in [5]. Both of these algorithms are interpolation in the frequency domain for linear detection such as ZF and minimum mean-square error (MMSE) detection. It is pointed out in [5] that interpolation operations are not compatible with BLAST detection using OSIC. In [6], a linear channel extrapolation method is proposed to predict the channel for multiuser MIMO E-SDM systems.

In this paper, we propose a novel algorithm to significantly reduce the computational needs to obtain the inverse of the channel matrix for MIMO detection. This algorithm employs second-order extrapolation in the time domain by exploiting the correlation among successive channel matrices. It can be applied for both linear detection and OSIC detection for MIMO systems. We will briefly review the MIMO flat-fading channel model and detection schemes in Sec. II. In Sec. III, we develop the linear extrapolation and second-order extrapolation techniques for matrix inversion. In Sec. IV, we analyze the complexity of the proposed algorithm. Simulation environment and numerical results are given in Sec. V. Sec. VI concludes this paper.

II. MIMO Flat-Fading Channel Model And Detection

A. MIMO Flat-Fading Channel Model

Consider an $M \times N$ MIMO system, where $N \geq M$. Its input-output relationship during one sampling interval can be expressed as [7]

$$\mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{n} \tag{1}$$

where $\mathbf{y} = [y_1 \ y_2 \ \cdots \ y_N]^T$ (superscript ^{T} denotes transpose), $\mathbf{s} = [s_1 \ s_2 \ \cdots \ s_M]^T$, and $\mathbf{n} = [n_1 \ n_2 \ \cdots \ n_N]^T$ represent, respectively, the received signal vector, the transmitted symbol vector, and the received noise vector. y_n and n_n, $n = 1, 2, \cdots, N$, denote the received signal and noise components by the nth receive antenna, respectively; s_m denotes the transmitted symbols with an average energy E_s from the mth transmit antenna. The elements of $\mathbf{n}$ are independent and identically distributed (i.i.d.), zero-mean, complex Gaussian random variables with variance $N_0/2$ per dimension. The channel matrix $\mathbf{H}$ has a dimension of $N \times M$, and its mth column, $\mathbf{h}_m$, represents the channel coefficients from the mth transmit antenna to all receive antennas. Note that since we consider time-varying channels in this paper, the channel matrix should more appropriately be written as a function of time t.

B. MIMO Detection in Flat-Fading Channel

Let us first briefly review linear detection algorithms, i.e., zero-forcing algorithm and MMSE algorithm, for MIMO detection. These two algorithms can be described by

$$\widetilde{\mathbf{s}} = \mathbf{W}\mathbf{y} \tag{2}$$

where $\widetilde{\mathbf{s}}$ denotes the decision statistics of the transmitted symbol vector $\mathbf{s}$ and $\mathbf{W}$ denotes the weight matrix. The transmitted data

This research is supported by the basic science research program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2009-0088286).

symbols can be recovered by passing the elements of $\widetilde{\mathbf{s}}$ through a decision device.

For ZF detection, the weight matrix $\mathbf{W}$ is $\mathbf{H}^{\dagger}$, where superscript † denotes the pseudo-inverse of a matrix. For MMSE algorithm, $\mathbf{W}$ is generated according to the MMSE criterion and is given by [8]

$$\mathbf{W} = \mathbf{H}^H(\mathbf{H}^H\mathbf{H} + \sigma^2\mathbf{I}_N)^{-1} = (\mathbf{H}^H\mathbf{H} + \sigma^2\mathbf{I}_M)^{-1}\mathbf{H}^H \quad (3)$$

where $\sigma^2 = N_0/E_s$, superscript ^{H} stands for the complex conjugate transpose, and $\mathbf{I}_M$ denotes the $M \times M$ identity matrix. Note that for $M \leq N$, the second equality of Eq. (3) leads to lower computational complexity.

The V-BLAST scheme [1] is a nonlinear detection scheme that is based on nulling and OSIC. It includes ZF-OSIC and MMSE-OSIC algorithms. This scheme requires M iterations at each sample interval. The main processing steps at each iteration are as follows [9]:

1) Compute the weight matrix according to zero-forcing or MMSE algorithm;
2) Determine which symbol should be recovered and choose the nulling vector from the columns of the conjugate transpose of the weight matrix;
3) Compute the decision statistic of the desired symbol and obtain the corresponding hard decision result;
4) Delete the interferences of the recovered symbols from the received signal vector.

III. Liner Extrapolation And Second-Order Extrapolation

Let us assume that at some time instants the channel matrices $\mathbf{H}$'s have been accurately estimated. Now, further channel matrix manipulations such as matrix inversion or weight matrix generation are needed for detection. As discussed in the previous section, both ZF and MMSE algorithms require computation of the inverse of matrices, and frequent inversions are computationally too intensive. We now focus on developing a computationally efficient algorithm for computing the weight matrices $\mathbf{W}$'s for detecting the symbol vector, $\mathbf{s}$, at each sample interval assuming that the channel matrices $\mathbf{H}$'s have been estimated.

A straightforward method is to use the brute-force inversion, i.e., directly compute $\mathbf{W}$ with conventional algorithms such as LU decomposition or Gaussian elimination for every MIMO symbol vector. With such a method, matrix inversions should be performed frequently since there will always be a small variation between successive channel matrices in time-varying fading channels. As the channel coherence time decreases, the calculation interval should be shortened accordingly, which results in even higher computational needs at the receiver.

The autocorrelation function of the fading process, $h(t)$, in a rich-scattering environment can be expressed as [10]

$$E\{h^*(t) \times h(t+\delta)\} = \rho^2 \times J_0(2\pi f_{dm}\delta) \quad (4)$$

where $J_0(\cdot)$ is the zeroth order Bessel function of the first kind, ρ^2 is the mean power of the fading channel, and f_{dm} is the maximum Doppler frequency. For small values of $f_{dm}\delta$, $J_0(2\pi f_{dm}\delta)$ can be approximated as [11]

$$J_0(2\pi f_{dm}\delta) \approx 1 - \frac{1}{4}(2\pi f_{dm}\delta)^2 + \frac{1}{64}(2\pi f_{dm}\delta)^4. \quad (5)$$

From this approximation, one can clearly see that when the time interval δ between two adjacent symbol intervals is small (much smaller than the value of f_{dm}), which is the case for receivers operating at high data rates, the channel matrix changes slowly.

Interpolation or extrapolation (I/E) has been used for channel estimation in [6], [11], [12]. This method, as shown in Fig. 1(a), exploits the correlation between consecutive channel matrices. I/E techniques could also be used for channel matrix inversion or other channel matrix manipulations, as shown in Fig. 1(b). Note that for I/E based channel estimation, the processing speed at the matrix manipulation stage is still high but, for I/E based matrix manipulation, since direct calculations are performed only at direct time instants, the processing speed before the I/E block is much lower. From this perspective, I/E at the matrix manipulation stage can lower the processing requirements, and thus lower the burden on DSP hardware.

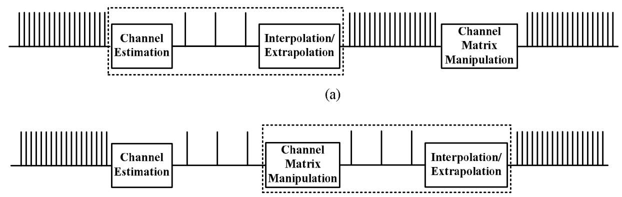

Fig. 1. In/Out sampling intervals of each algorithm block. (a) I/E based channel estimation; (b) I/E based channel matrix manipulation.

Next, we develop a novel algorithm that uses extrapolation to estimate the weight matrix without brute-force inversion of channel matrices for every MIMO symbol interval. Unlike the traditional second-order extrapolation [11], we do not predict the channel matrix inverse with second-order polynomial. Based on the last two or three successive direct (and accurate) inversion results, e.g., obtained by using the brute-force methods, as shown in Fig. 2, we perform linear extrapolation and second-order extrapolation. We assume that the brute-force inversion is performed at every l sample times, i.e., the direct calculation interval equals l; thus there are $l-1$ channel matrices between the two direct calculation instants.

Let $\mathbf{W}(t_{-2})$, $\mathbf{W}(t_{-1})$, $\mathbf{W}(t_0)$, $\mathbf{W}(t_1)$ be four consecutive directly calculated weight matrices. Suppose that the difference between $\mathbf{W}(t_{-1})$ and $\mathbf{W}(t_0)$ is the same as the difference between $\mathbf{W}(t_0)$ and $\mathbf{W}(t_1)$, i.e.,

$$\mathbf{W}(t_0) - \mathbf{W}(t_{-1}) = \mathbf{W}(t_1) - \mathbf{W}(t_0). \quad (6)$$

Then the sample-to-sample increment Δ_1 for this estimated interval is then expressed as

$$\Delta_1 = \frac{1}{l}(\mathbf{W}(t_{-1}) - \mathbf{W}(t_0)). \quad (7)$$

Thus, linear extrapolation can be expressed as

$$\begin{aligned}\hat{\mathbf{W}}^{1st}(t_0 + k\delta) &= \mathbf{W}(t_0) + \frac{k}{l}(\mathbf{W}(t_{-1}) - \mathbf{W}(t_0)) \\ &= \mathbf{W}(t_0) + k\Delta_1 \end{aligned} \quad (8)$$

where δ represents the transmission interval of MIMO symbol vectors. Hence, $\hat{\mathbf{W}}^{1st}(t_0 + k\delta)$ is the linearly extrapolated weight matrix for the kth MIMO symbol vector within time interval $[t_0,\ t_1]$. In practice, after we calculate the sample-to-sample increment Δ_1 for this estimation interval, we obtain the consecutive weight matrices using linear extrapolation within this interval by

$$\begin{aligned}\hat{\mathbf{W}}^{1st}(t_0+\delta) &= \mathbf{W}(t_0)+\Delta_1\\ \hat{\mathbf{W}}^{1st}(t_0+2\delta) &= \hat{\mathbf{W}}^{1st}(t_0+\delta)+\Delta_1\\ &\vdots\\ \hat{\mathbf{W}}^{1st}(t_0+(l-1)\delta) &= \hat{\mathbf{W}}^{1st}(t_0+(l-2)\delta)+\Delta_1. \quad (9)\end{aligned}$$

This algorithm is referred to as direct and linear extrapolation (D&E (linear)).

For second-order extrapolation, we assume the following fact

$$\begin{aligned}&(\mathbf{W}(t_1)-\mathbf{W}(t_0))-(\mathbf{W}(t_0)-\mathbf{W}(t_{-1}))=\\ &(\mathbf{W}(t_0)-\mathbf{W}(t_{-1}))-(\mathbf{W}(t_{-1})-\mathbf{W}(t_{-2})). \quad (10)\end{aligned}$$

Then the total increment for the estimation interval is expressed as

$$\begin{aligned}\mathbf{W}(t_1)-\mathbf{W}(t_0) &= 2(\mathbf{W}(t_0)-\mathbf{W}(t_{-1}))-(\mathbf{W}(t_{-1})-\mathbf{W}(t_{-2}))\\ &= 2\mathbf{W}(t_0)-3\mathbf{W}(t_{-1})+\mathbf{W}(t_{-2}). \quad (11)\end{aligned}$$

The sample-to-sample increment Δ_2 for this estimation interval is calculated as

$$\Delta_2 = \frac{1}{l}(2\mathbf{W}(t_0)-3\mathbf{W}(t_{-1})+\mathbf{W}(t_{-2})). \quad (12)$$

Therefore, the second-order extrapolation is given by

$$\begin{aligned}\hat{\mathbf{W}}^{2nd}(t_0+k\delta) &= \mathbf{W}(t_0)+\frac{k}{l}(2\mathbf{W}(t_0)-3\mathbf{W}(t_{-1})+\mathbf{W}(t_{-2}))\\ &= \mathbf{W}(t_0)+k\Delta_2. \quad (13)\end{aligned}$$

where $\hat{\mathbf{W}}^{2nd}(t_0+k\delta)$ is the kth second-order-extrapolated weight matrix for MIMO symbol vectors within the time interval $[t_0,\ t_1]$.

Similarly, after we calculate the sample-to-sample increment Δ_2 for this estimation interval, we obtain the consecutive weight matrices using second-order extrapolation within this interval by

$$\begin{aligned}\hat{\mathbf{W}}^{2nd}(t_0+\delta) &= \mathbf{W}(t_0)+\Delta_2\\ \hat{\mathbf{W}}^{2nd}(t_0+2\delta) &= \hat{\mathbf{W}}^{2nd}(t_0+\delta)+\Delta_2\\ &\vdots\\ \hat{\mathbf{W}}^{2nd}(t_0+(l-1)\delta) &= \hat{\mathbf{W}}^{2nd}(t_0+(l-2)\delta)+\Delta_2. \quad (14)\end{aligned}$$

This algorithm is referred to as direct and second-order extrapolation (D&E (second-order)).

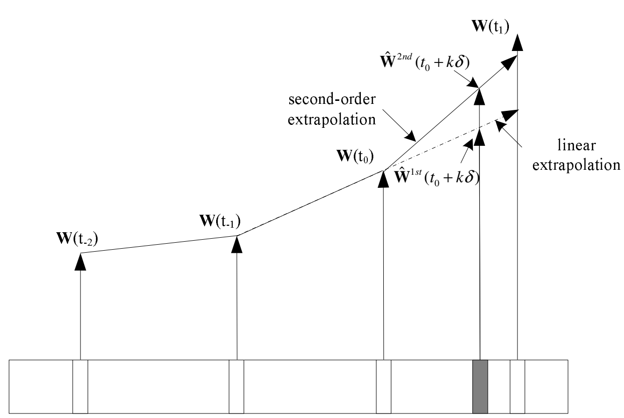

Fig. 2. Linear and second-order extrapolation for weight matrix estimation.

From Eqs. (9) and (14), we observe that only matrix addition, rather than multiplication, is involved when using extrapolation to compute the weight matrices. In other words, since the increment has already been calculated and will remain within one direct calculation interval, the estimates of the weight matrices within this estimation interval can be obtained by simply adding the increment for consecutive matrices.

We can extend the extrapolation techniques to every layer at each iteration in OSIC detection. Since the channel matrix changes slowly, the detection order with OSIC scheme can be maintained for a relatively long period. Therefore, ZF-OSIC and MMSE-OSIC with extrapolation can be realized as follows.

Initialization: Obtain weight matrix $\mathbf{W}$ from $\mathbf{H}$ based on extrapolation
if decoding order does not change
  Recursion: for $m=1,2,\ldots,M-1$
    Use extrapolation to obtain current weight matrix
else
  Recursion: for $m=1,2,\ldots,M-1$
    Directly calculate the weight matrix of this layer for all MIMO symbol vectors within three consecutive direct calculation intervals;
    Buffer the results only at the original direct calculation instant
Solutions: Estimate the transmitted signals;
    Update decoding order.

Finally, for comparison only we also introduce the so-called direct and hold (D&H) algorithm. Similar to the direct-and-extrapolation algorithm mentioned above, for D&H algorithm, brute-force inversion is performed at direct calculation instants and then the inversion result is held as the weight matrices within one direct calculation interval. We will discuss the complexity and performance for these algorithms in the next two sections.

IV. Complexity Analysis

The transmitted and received signals as well as the channel matrices are all of complex valued. Thus, all manipulations including additions, multiplications, and divisions are performed on complex values. In the rest of this paper, when we discuss computational complexity, we refer to multiplications of complex numbers (with two variable operands, i.e., a real part and an imaginary part).

For MMSE detection, from Eq. (3), since $\mathbf{H}^H$ is the Hermitian transpose of $\mathbf{H}$, computing $\mathbf{A}=\mathbf{H}^H\mathbf{H}$ requires $NM^2-\frac{1}{2}M^2+\frac{1}{2}M$ multiplications. Also, $\frac{3}{2}M^3+(N+1)M^2+\frac{1}{2}M$ multiplications are needed to calculate $(\mathbf{A}+\sigma^2\mathbf{I}_M)^{-1}\mathbf{H}^H$, assuming that $\sigma^2\mathbf{I}_M$ has been pre-computed.

Linear extrapolation needs $\frac{1}{2}(N+1)M$ multiplications to compute the increment for one interval; second-order extrapolation requires $3(N+1)M$ multiplications to compute the increment for one interval. Therefore, the average complexity for obtaining one weight matrix of direct and linear extrapolation algorithm is

$$\begin{aligned}&\frac{1}{l}\left[\left(NM^2-\frac{1}{2}M^2+\frac{1}{2}M\right)+\left(\frac{3}{2}M^3+(N+1)M^2\right.\right.\\ &\left.\left.+\frac{1}{2}M\right)+\left(\frac{1}{2}(N+1)M\right)\right]\\ &=\frac{1}{l}\left(2NM^2+\frac{1}{2}NM+\frac{3}{2}M^3+\frac{1}{2}M^2+\frac{3}{2}M\right)\\ &=\frac{1}{l}\left(\frac{7}{2}N^3+N^2+\frac{3}{2}N\right)\ \text{if } M=N\end{aligned}$$

where l is the number of channel matrices within one direct calculation interval.

For direct and second-order extrapolation method, the average complexity for obtaining one weight matrix is

$$\frac{1}{l}\left[\left(NM^2-\frac{1}{2}M^2+\frac{1}{2}M\right)+\left(\frac{3}{2}M^3+(N+1)M^2\right.\right.$$
$$\left.\left.+\frac{1}{2}M\right)+(3(N+1)M)\right]$$
$$=\frac{1}{l}\left(2NM^2+3NM+\frac{3}{2}M^3+\frac{1}{2}M^2+4M\right)$$
$$=\frac{1}{l}\left(\frac{9}{2}N^3+\frac{7}{2}N^2+4N\right)\text{ if } M=N.$$

For direct and hold method, the average complexity for obtaining one weight matrix is

$$\frac{1}{l}\left[\left(NM^2-\frac{1}{2}M^2+\frac{1}{2}M\right)+\left(\frac{3}{2}M^3+(N+1)M^2\right.\right.$$
$$\left.\left.+\frac{1}{2}M\right)\right]$$
$$=\frac{1}{l}\left(2NM^2+\frac{3}{2}M^3+\frac{1}{2}M^2+M\right)$$
$$=\frac{1}{l}\left(\frac{7}{2}N^3+\frac{1}{2}N^2+N\right)\text{ if } M=N.$$

For brute-force inversion at every sample instant, the complexity for calculating one weight matrix is expressed as

$$2NM^2+\frac{3}{2}M^3+\frac{1}{2}M^2+M$$
$$=\frac{7}{2}N^3+\frac{1}{2}N^2+N\text{ if } M=N.$$

For MMSE-OSIC detection, the number of columns of the channel matrix decreases as the interferences from the transmitted symbols is canceled one-by-one at each iteration. Therefore, the channel matrix dimension decreases accordingly, and thus the computational complexities at different layers are different. Let i be the iteration count. Then, the complexity for direct and linear extrapolation is expressed as

$$\frac{1}{l}\sum_{i=0}^{M-1}\left(2N(M-i)^2+\frac{1}{2}N(M-i)+\frac{3}{2}(M-i)^3\right.$$
$$\left.+\frac{1}{2}(M-i)^2+\frac{3}{2}(M-i)\right).$$

The complexity for direct and second-order extrapolation is given by

$$\frac{1}{l}\sum_{i=0}^{M-1}\left(2N(M-i)^2+3N(M-i)+\frac{3}{2}(M-i)^3\right.$$
$$\left.+\frac{1}{2}(M-i)^2+4(M-i)\right).$$

The complexity for direct and hold method is calculated by

$$\frac{1}{l}\sum_{i=0}^{M-1}\left(2N(M-i)^2+\frac{3}{2}(M-i)^3+\frac{1}{2}(M-i)^2\right.$$
$$\left.+(M-i)\right).$$

The complexity for brute-force inversion is given by

$$\sum_{i=0}^{M-1}\left(2N(M-i)^2+\frac{3}{2}(M-i)^3+\frac{1}{2}(M-i)^2\right.$$
$$\left.+(M-i)\right).$$

V. Numerical Results And Discussion

Based on the results in the previous sections, we will compare the complexity versus bit error rate (BER) performance of the above algorithms. The system is assumed to employ 16-QAM without channel coding. Different complexities correspond to different direct calculation intervals, i.e., different values of l.

One important quantity is the normalized Doppler frequency, which is the absolute Doppler frequency divided by MIMO symbol rate in Hz. This parameter shows how fast the channel actually changes relatively to the MIMO transmission rate. In the following simulations, we assume two normalized Doppler frequencies of 1.5×10^{-3} and 2.2×10^{-4}. At a transmission rate 10^6 symbols/sec, these two normalized Doppler frequencies correspond to an absolute Doppler frequency of 1500 Hz and 220 Hz, respectively.

With the algorithms developed in Sec. III, the real part of one estimated weight matrix element with different direct calculation intervals under the normalized Doppler Frequency of 1.5×10^{-3} are shown in Fig. 3 and Fig. 4. As seen in Fig. 3, since the direct calculation interval is 3, D&E (second-order) matches well brute-force inversion, D&E (linear) has a slight offset from the exact result, and D&H has the largest offset among the three. When the direct calculation interval increases to 6, as shown in Fig. 4, D&E (second-order) still matches well brute-force inversion, though with a slightly increased offset compared with the previous case; however, D&E (linear) has a large offset from the exact result; D&H has a much greater offset from the exact values as the direct calculation interval increases.

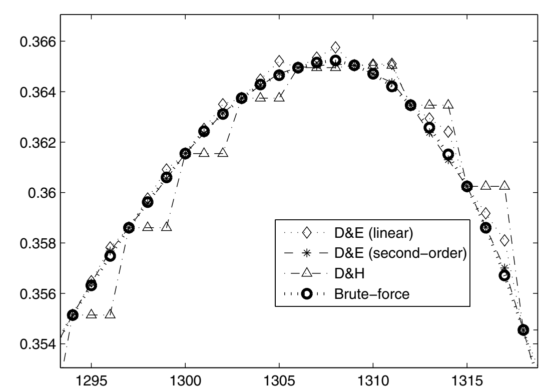

Fig. 3. Weight matrix element estimated by different algorithms ($l=3$). This figure shows a portion of one element of the inverted fading channel matrix versus time. The x-axis shows the sampling instant; the y-axis is the real part of one weight matrix element.

The error performance of the different algorithms for MIMO detection is given in Fig. 5 and Fig. 6. As shown in Fig. 5, for MMSE detection in a 5×6 MIMO system, D&E (second-order) algorithm can significantly reduce the complexity while maintaining the same BER performance as brute-force inversion. At the same bit error rate value, D&E (second-order) algorithm

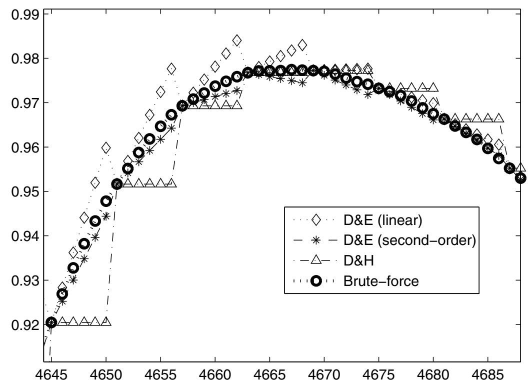

Fig. 4. Weight matrix element estimated by different algorithms ($l = 6$). The x-axis and y-axis are the same as defined in Fig. 3.

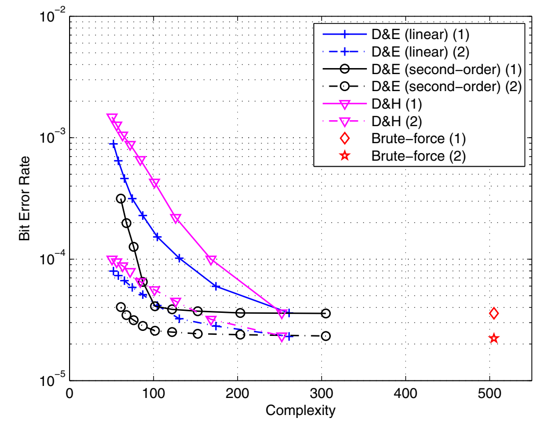

Fig. 5. BER vs complexity of MMSE detector in 5×6 MIMO system with 16-QAM with normalized Doppler frequency: (1) 1.5×10^{-3}; (2) 2.2×10^{-4}.

reduces the complexity of brute-force inversion by 80% when the direct calculation interval is only 6. However, D&E (linear) and D&H algorithm could achieve the same BER as brute-force inversion only at the highest complexity, i.e., when direct calculation interval equals 2.

Fig. 6 demonstates that D&E (second-order) algorithm can maintain the same BER as brute-force inversion with approximately one third of the complexity of the latter for MMSE-OSIC detection in a 4×4 MIMO system. In comparison, D&H algorithm achieves the same BER as brute-force inversion only at the highest complexity. However, D&E (linear) algorithm can hardly reach the same BER even at the highest complexity level. Compared to Fig. 5, for D&E (second-order) algorithm, the direct calculation interval is shortened to be 5 in order to maintain the same BER as brute-force inversion.

We also note from the simulation results that as the direct calculation interval increases (thus the complexity decreases accordingly), the BER performance of all algorithms degrades. In order to maintain the same BER as brute-force inversion, the direct calculation interval should not be set too high. In addition, D&E (second-order) under smaller normalized Doppler frequencies might not possess as much advantage over D&E (linear) or D&H as at higher Doppler frequencies. Furthermore, the BER performance is better under smaller normalized Doppler frequencies.

VI. Conclusion

We have proposed a computationally efficient algorithm for matrix inversion in MIMO linear and nonlinear detection. This algorithm is based on second-order extrapolation. It significantly reduces the complexity for inverting channel matrices while maintaining the same BER performance as the traditional method of brute-force inversion. Simulation shows that for MIMO MMSE detection and MMSE-OSIC detection, the proposed algorithm reduces the complexity of brute-force inversion by approximately 80% and 67%, respectively. The general idea developed in this paper is also applicable for other MIMO detection schemes, such as QR decomposition and singular value decomposition. Therefore, this algorithm can be easily extended to more sophisticated MIMO detection schemes.

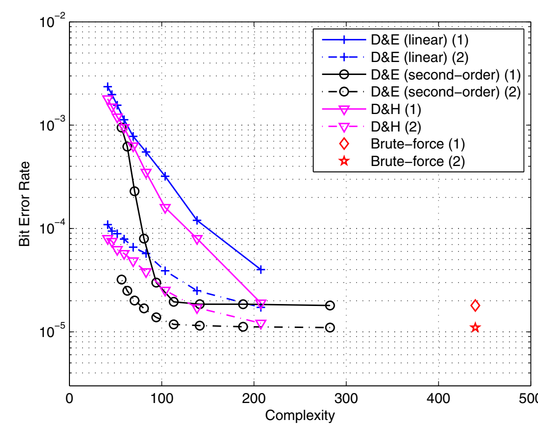

Fig. 6. BER vs complexity of MMSE-OSIC detector in 4×4 MIMO system with 16-QAM with normalized Doppler frequency: (1) 1.5×10^{-3}; (2) 2.2×10^{-4}.

References

[1] 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-98*, Italy, Sep. 1998.

[2] G. J. Foschini and M. J. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Personal Commun.*, vol. 6, pp. 311–335, Mar. 1998.

[3] A. Greenbaum, *Iterative Methods for Solving Linear Systems*, Society for Industrial and Applied Mathematics, 1997.

[4] M. Borgmann and H. Bolcskei, "Interpolation-based efficient matrix inversion for MIMO-OFDM receivers," in *Proc. 38th Asilomar Conf. on Signals, Systems and Computers*, Nov. 2004, pp. 1941–1947.

[5] J. Wang and B. Daneshrad, "Performance of linear interpolation-based MIMO detection for MIMO-OFDM systems," in *Proc. IEEE WCNC'04*, Atlanta, USA, 2004, pp. 981–986.

[6] H. Bui, Y. Ogawa, T. Nishimura, and T. Ohgane, "Multiuser MIMO E-SDM systems: Performance evaluation and improvement in time-varying fading environments," in *Proc. IEEE Globecom*, pp. 1–5, Nov. 2008.

[7] A. Paulraj, R. Nabar, and D. Gore, *Introduction to Space-Time Wireless Communications*. Cambridge University Press, 2003.

[8] 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, Jul. 2003.

[9] Z. Luo, S. Liu, M. Zhao, and Y. Liu, "A novel optimal MMSE-SIC detection algorithm for V-BLAST systems," in *Proc. IEEE ICC'06*, Istanbul, Turkey, Jun. 2006, pp. 3105–3110.

[10] W.C. Jakes, *Microwave Mobile Communications*, John Wiley & Sons., 1974.

[11] B. Phu, Y. Ogawa, T. Ohgane, and T. Nishimura, "Extrapolation of Time-Varying MIMO Channels for an E-SDM System," in *Proc. IEEE VTC'06*, Melbourne, Australia, 2006, pp. 1748–1752.

[12] S. Coleri, M. Ergen, A. Puri, and A. Bahai, "A study of channel estimation in OFDM systems," in *Proc. IEEE VTC'02*, Vancouver, Canada, 2002, pp. 894–898.