A Low-Complexity Upgrade of the Linear Detector for MIMO Channels via Partial Decision Feedback

Deric W. Waters, *Member, IEEE*, and John R. Barry, *Senior Member, IEEE*

***Abstract*— The BLAST-ordered decision-feedback (BODF) detector is a nonlinear detection strategy for multiple-input multiple-output (MIMO) channels that can significantly outperform a linear detector. However, for some applications even the BODF detector is too complex. We propose the partial decision-feedback (PDF) detector, a stripped-down version of the BODF detector that only feeds back one decision. The PDF detector performs close to the BODF detector, with complexity comparable to the linear detector. For example, over a 3-input 3-output Rayleigh-fading channel with 64-QAM inputs, the PDF detector is 21% less complex than the BODF detector, yet requires only 0.3 dB more average signal energy to reach a bit-error rate of 10^{-3}.**

***Index Terms*— Decision-feedback, MIMO, linear detection, low-complexity MIMO detector.**

I. Introduction

MULTIPLE-input multiple-output (MIMO) communications systems have attracted a flurry of research recently because of their promise of high spectral efficiency and spatial diversity [1]. The high complexity of MIMO detection is a practical bottleneck to realizing a MIMO communications system. For example, the maximum-likelihood (ML) detector is not practical at high data rates, because its complexity grows exponentially with the spectral efficiency and the number of channel inputs. Even the low-complexity linear detector — which simply inverts the channel — can be a challenge to implement in real-time.

Many MIMO detectors have been proposed to close the performance and complexity gaps between the linear and ML detectors. For example, the BLAST-ordered decision-feedback (BODF) detector [2] can significantly outperform the linear detector. The BODF detector may be implemented in two stages, where the first stage is the linear detection filter [3]–[5], and the second stage is a noise-prediction mechanism that reduces noise variance. Another low-complexity detector is the group detector [6]–[8], which divides symbols into two groups, and then detects the first group using ML detection. After cancelling the interference due to the first group of symbols, the second group of symbols is detected using a suboptimal technique. Finally, in contrast to the decision-feedback detector, the partial feedback multiuser detector of [9] cancels the interference of only a subset of available decisions. It first divides the users into groups according to their signal energies. Then, the detection strategy for each user group is different, but a given user always uses every decision from stronger users for interference cancelation.

In this paper we propose a new method for improving upon the linear detector called the partial decision-feedback (PDF) detector. When computations are at a premium, for example to avoid burdening a small portable receiver, even the BODF detector may be too complex to implement. In such applications, the PDF detector offers a way to improve upon the linear detector with much less additional complexity. Like the BODF detector, the first stage of the PDF detector is also a linear detection filter. The second stage is a simplified noise-prediction mechanism that attains most of the performance improvement achieved by the BODF detector, while adding significantly fewer computations. In fact, in the limit of high signal-to-noise ratio (SNR), the word-error rate of the PDF detector converges to that of the BODF detector. The PDF detector can also be viewed as a variation of the group detector where the first and second groups are both detected using linear detection. In this paper we focus specifically on the case where the first group contains only a single symbol. The PDF detector differs from the partial feedback multiuser detector not only in how it orders the users, but also because it removes the interference from only a subset of the stronger users.

The remainder of this paper is organized as follows. In Section II we describe the PDF detector. In Section III we describe the complexity of the PDF detector. In Section IV we compare the performance and complexity of the PDF, BODF, and linear detectors. Finally, in Section V we make concluding remarks.

II. Partial Decision-Feedback Detection

This paper considers a memoryless flat-fading MIMO channel with N inputs $\boldsymbol{a} = [a_1, \ldots, a_N]^{\mathrm{T}}$ and M outputs $\boldsymbol{r} = [r_1, \ldots, r_M]^{\mathrm{T}}$:

$$\boldsymbol{r} = \mathbf{H}\boldsymbol{a} + \boldsymbol{w}, \tag{1}$$

where $\mathbf{H}$ is a complex $M \times N$ channel matrix, and where $\boldsymbol{w} = [w_1, \ldots, w_M]^{\mathrm{T}}$ is additive white noise. We assume that the columns of $\mathbf{H}$ are linearly independent, which implies that there are at least as many outputs as inputs, $M \geq N$. We assume that the noise components are uncorrelated with complex variance σ^2, so that $E[\boldsymbol{w}\boldsymbol{w}^*] = \sigma^2\mathbf{I}$, where $\boldsymbol{w}^*$ denotes the conjugate transpose of $\boldsymbol{w}$. Further, we assume that

Manuscript received July 16, 2005; revised March 2, 2006; accepted April 26, 2006. The associate editor coordinating the review of this paper and approving it for publication was G. Leus. Portions of this work were presented at IEEE Globecom Conference, vol. 4, pp. 2635-2639, Dallas, TX, U.S.A., Nov. 29 - Dec. 3, 2004. This research was supported by National Science Foundation grants 0431031 and 0121565.

D. W. Waters is with Texas Instruments, Dallas, TX 75243 (e-mail: deric@ti.com).

J. R. Barry is with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 (e-mail: barry@ece.gatech.edu).

Digital Object Identifier 10.1109/TWC.2007.05551.

the inputs are chosen from the same unit-energy alphabet $\mathcal{A}$ and are uncorrelated, so that $E[\boldsymbol{a}\boldsymbol{a}^*] = \mathbf{I}$.

The PDF detector has five steps, as outlined below:

Step 1: Apply a linear filter to the channel output $\boldsymbol{r}$.
Step 2: Identify the index i of the symbol to detect first.
Step 3: Detect the i-th symbol by slicing the i-th output of the linear filter.
Step 4: Cancel the interference due to the i-th symbol.
Step 5: Detect the remaining symbols linearly.

A straightforward implementation of the zero-forcing PDF detector would use the i-th row of the channel pseudoinverse to detect the i-th symbol, and the pseudoinverse of $\mathbf{H}^{(i)}$ to detect the remaining symbols, where $\mathbf{H}^{(i)}$ is the submatrix created by swapping the first and i-th columns of $\mathbf{H}$ then deleting the first column. However, in this section we propose a lower complexity implementation of the PDF detector based on noise prediction. We describe this noise-predictive implementation in a general way that applies to both its zero-forcing (ZF) and minimum-mean-squared error (MMSE) versions.

Step 1: The PDF detector begins by applying the linear filter [3] $\mathbf{C} = (\mathbf{H}^*\mathbf{H} + \hat{\sigma}^2\mathbf{I})^{-1}\mathbf{H}^*$ to the channel output, where the parameter $\hat{\sigma} \in \{0, \sigma\}$ determines whether the ZF ($\hat{\sigma} = 0$) or MMSE ($\hat{\sigma} = \sigma$) version of the filter is implemented. The linear filter can be expressed as $\mathbf{C} = \mathbf{L}^{-1}\mathbf{Q}^*$, where the matrices $\mathbf{L}$ and $\mathbf{Q}$ are defined by the following QR decomposition of the extended channel matrix [10], [11]:

$$\begin{bmatrix}\mathbf{H} \\ \hat{\sigma}\mathbf{I}\end{bmatrix} = \tilde{\mathbf{Q}}\mathbf{L}. \tag{2}$$

The $(M+N) \times N$ matrix $\tilde{\mathbf{Q}}$, which satisfies $\tilde{\mathbf{Q}}^*\tilde{\mathbf{Q}} = \mathbf{I}$, can be further decomposed according to [11]:

$$\tilde{\mathbf{Q}} = \begin{bmatrix}\mathbf{Q} \\ \mathbf{Q}_2\end{bmatrix}, \tag{3}$$

so that $\mathbf{Q}$ is defined as the first M rows of $\tilde{\mathbf{Q}}$. The $N \times N$ matrix $\mathbf{L}$ is lower triangular with positive and real diagonal elements.

Applying the linear filter to the channel output yields $\boldsymbol{y} = \mathbf{C}\boldsymbol{r}$, which reduces to:

$$\boldsymbol{y} = \boldsymbol{a} - \hat{\sigma}^2\mathbf{U}^*\mathbf{U}\boldsymbol{a} + \mathbf{U}^*\mathbf{Q}^*\boldsymbol{w}, \tag{4}$$

where $\mathbf{U} = (\mathbf{L}^{-1})^*$, and where we used the fact that $\mathbf{Q}^*\mathbf{H} = \mathbf{L} - \hat{\sigma}^2\mathbf{U}$. The output of this linear filter is the desired signal plus an effective noise:

$$\boldsymbol{y} = \boldsymbol{a} + \boldsymbol{n}, \tag{5}$$

where the effective noise $\boldsymbol{n} = [n_1, \ldots, n_N]^{\mathrm{T}}$ is no longer white; its autocorrelation matrix is $\mathbf{R}_{\mathbf{nn}} = \sigma^2\mathbf{U}^*\mathbf{U}$, since $\mathbf{Q}^*\mathbf{Q} = \mathbf{I} - \hat{\sigma}^2\mathbf{U}\mathbf{U}^*$, and $\hat{\sigma} \in \{0, \sigma\}$. Although $\boldsymbol{n}$ includes a residual interference term in addition to the noise when $\hat{\sigma} \neq 0$, we continue to refer to it simply as *noise*.

It is well known that applying the MMSE linear filter $\mathbf{C}$ to the channel output results in a biased signal. This can be corrected by left-multiplying $\boldsymbol{y}$ by a bias removal matrix $\mathbf{B}$, whose i-th diagonal element is $1/(1 - \hat{\sigma}^2\|\boldsymbol{u}_i\|^2)$, where $\boldsymbol{u}_i$ is the i-th column of $\mathbf{U}$. For simplicity we describe the PDF detector without removing the bias.

Step 2: We propose choosing i as the symbol with the smallest noise variance, which corresponds to the first index of the BLAST ordering. The noise variance of the j-th symbol is proportional to the j-th diagonal element of the autocorrelation matrix $\mathbf{R}_{\mathbf{nn}}$, so the index of the first symbol is chosen according to:

$$i = \underset{j \in \{1,2,\ldots,N\}}{\operatorname{argmin}} \|\boldsymbol{u}_j\|^2. \tag{6}$$

Step 3: The i-th symbol is detected by quantizing the i-th output of the linear filter to the nearest symbol in the alphabet, $\hat{a}_i = \mathrm{dec}\{y_i\}$, where $\mathrm{dec}\{x\}$ rounds x to the nearest element of $\mathcal{A}$.

Step 4: Noise-predictive decision feedback is used to reduce the noise variance, and is described by the following equation:

$$\boldsymbol{z} = \boldsymbol{y} - \boldsymbol{p}(y_i - \hat{a}_i), \tag{7}$$

where $\boldsymbol{p} = [p_1, \ldots p_N]^{\mathrm{T}}$ is a vector of prediction coefficients. The difference $y_i - \hat{a}_i$ reduces to the noise n_i whenever the decision is correct ($\hat{a}_i = a_i$). The term $p_k(y_i - \hat{a}_i)$ is a prediction of n_k that exploits the correlation between n_k and n_i. Since the i-th symbol is detected first, it cannot benefit from noise prediction, meaning $p_i = 0$. The k-th prediction coefficient is chosen to minimize the mean-squared error (MSE) for the k-th symbol [5], which reduces to the following when $\hat{a}_i$ is correct:

$$E[|n_k - p_k n_i|^2] = E[|(\boldsymbol{u}_k^* - p_k\boldsymbol{u}_i^*)(\mathbf{Q}^*\boldsymbol{w} - \hat{\sigma}^2\mathbf{U}\boldsymbol{a})|^2] \tag{8}$$

$$= \sigma^2\|\boldsymbol{u}_k - p_k^*\boldsymbol{u}_i\|^2, \tag{9}$$

where the second equality relies upon the fact that $\hat{\sigma} \in \{0, \sigma\}$, and we have substituted $n_j = \boldsymbol{u}_j^*\mathbf{Q}^*\boldsymbol{w} - \hat{\sigma}^2\boldsymbol{u}_j^*\mathbf{U}\boldsymbol{a}$ from the definition of $\boldsymbol{n}$. From (9) we see that the noise variance is minimized when the term $p_k^*\boldsymbol{u}_i$ is the projection of $\boldsymbol{u}_k$ onto the subspace spanned by $\boldsymbol{u}_i$, so the k-th prediction coefficient is given by:

$$p_k = \boldsymbol{u}_k^*\boldsymbol{u}_i/\|\boldsymbol{u}_i\|^2, k \neq i. \tag{10}$$

Step 5: The remaining symbols are detected by quantizing the elements of $\boldsymbol{z}$ from (7):

$$\hat{a}_k = \mathrm{dec}\{z_k\}, k \neq i. \tag{11}$$

Finally, the PDF detector's hard decision regarding a_k can be summarized succinctly as:

$$\hat{a}_k = \mathrm{dec}\{y_k - p_k(y_i - \hat{a}_i)\}. \tag{12}$$

Fig. 1 shows the block diagram of the PDF detector after i and $\boldsymbol{p}$ have been calculated. Fig. 2 describes a computationally efficient implementation of the noise-predictive PDF detector.

It can be shown that the performance of the PDF detector converges to that of the BODF detector at high SNR [12]. This is due to the fact that the performance of both detectors is dominated by the probability of error for the first symbol detected. Since the first symbols detected by the PDF and BODF detectors are mathematically equivalent, their performances will converge when the overall probability of error is dominated by the first symbol, which occurs at high SNR.

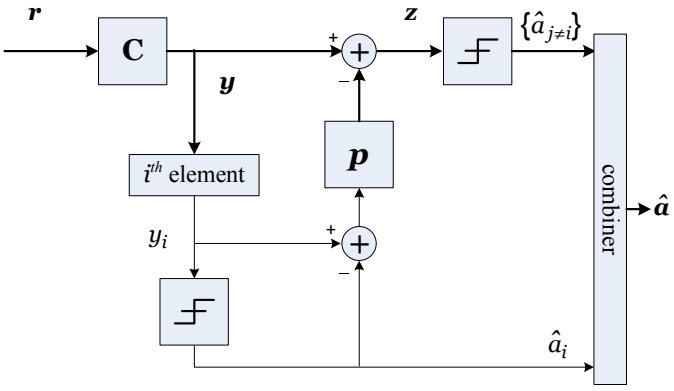

Fig. 1. Noise-predictive partial-DF detector.

PDF Algorithm: Input: $\mathbf{H}$, $\boldsymbol{r}$, $\hat{\sigma}$ Output: $\hat{\boldsymbol{a}}$

1) $[\tilde{\mathbf{Q}}, \mathbf{L}] = QR\left(\begin{bmatrix}\mathbf{H}\\ \hat{\sigma}\mathbf{I}\end{bmatrix}\right)$.
2) $\mathbf{U} = \left(\mathbf{L}^{-1}\right)^* = [\boldsymbol{u}_1, \ldots \boldsymbol{u}_N]$.
3) for $j = 1$ to N, $e_j = \|\boldsymbol{u}_j\|^2$, end.
4) $i = \underset{j \in \{1,2,\ldots,N\}}{\text{argmin}} e_j$.
5) $\mathbf{Q}$ = First M rows of $\tilde{\mathbf{Q}}$.
6) $\boldsymbol{y} = \mathbf{U}^*\mathbf{Q}^*\boldsymbol{r}$.
7) $\hat{a}_i = \text{dec}\{y_i\}$.
8) $n_i = y_i - \hat{a}_i$.
9) for $k \neq i$,
10) $\quad p_k = \boldsymbol{u}_k^* \boldsymbol{u}_i / e_i$.
11) $\quad \hat{a}_i = \text{dec}\{y_k - p_k n_i\}$.
12) end

Fig. 2. Noise-predictive partial-DF detector algorithm.

III. Complexity

We quantify the complexity of the proposed detector by counting the number of real multiplications per symbol period required to implement the algorithm described in Fig. 2. Complex multiplications are counted as three real multiplications, and the squared absolute values of complex numbers are counted as two real multiplications. Although this complexity measure disregards additions, divisions, and square-roots it is still a reasonable measure of complexity since the number of multiplications dominates the overall complexity. Furthermore, the complexity is divided into *preprocessing* and *core-processing* complexities. The preprocessing is the set of computations that are performed only once for each channel estimate, while the core-processing computations are performed every symbol period. We assume that the channel estimate is updated every L symbol periods. Thus, the total complexity per symbol period is the sum of the preprocessing complexity divided by L plus the core-processing complexity. Although our complexity metric accounts only for multiplications, it is worth pointing out that the preprocessing complexity of the PDF detector requires roughly half as many divisions and square-roots as the BODF detector.

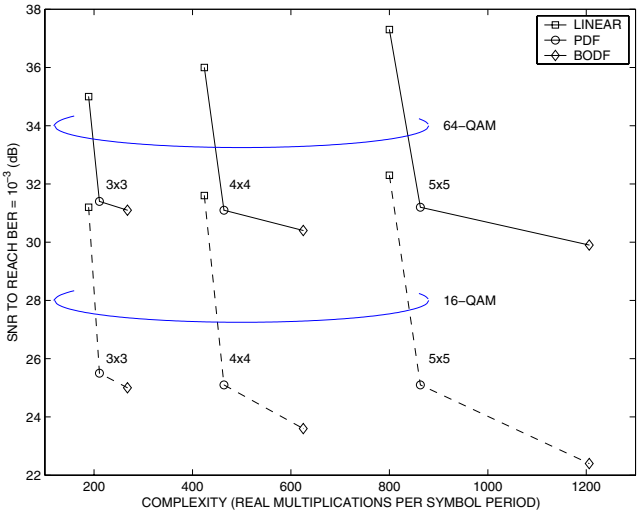

Fig. 3. Performance versus complexity for the MMSE versions of the linear, PDF, and BODF detectors. Results are averaged over 10^5 $N \times N$ Rayleigh-fading channels where $L = 1$.

For the implementation of the PDF detector proposed in Fig. 2, the preprocessing complexity includes lines (1)–(5), and (10). The core-processing complexity includes lines (6)–(8), and (11). Continuing our view of the PDF detector as an add-on to the linear filter, we consider only those computations it requires beyond the linear detector. Specifically, the linear detector requires lines (1)–(2) and (6) of Fig. 2 whether or not the PDF additional processing is used. Therefore, the additional preprocessing complexity required by the PDF detector are N^2 real multiplications in line (3), and a maximum of $3N^2/2 - 5N/2 + 1$ real multiplications in line (10) when $i = N$. The additional core-processing complexity of the PDF detector is only the $3(N-1)$ real multiplications needed at line (11).

IV. Numerical Results

In this section, we compare the performance and complexity of the MMSE versions of the linear, partial DF, and BLAST-ordered DF detectors. We will show that the performance-complexity trade-off depends on the dimensions of the channel, as well as the size of the input alphabet. Although it can be shown that the PDF and BODF detectors converge to the same performance at high SNR, we will see that there can be a significant gap at practical SNR. However, even when the BODF detector significantly outperforms the PDF detector, the PDF detector still offers a way for the receiver to significantly improve upon the performance of the linear detector with a relatively small complexity increase. The SNR is defined as the received signal energy per signaling interval divided by the one-sided noise power spectral density at each receive antenna, divided by the number of bits per symbol, $SNR = E[\|\mathbf{H}\boldsymbol{a}\|^2]/(E[\|\boldsymbol{w}\|^2]\log_2|\mathcal{A}|)$. We assume that the receiver has perfect knowledge of the channel parameters $\mathbf{H}$ and σ^2. We measure performance as the SNR required to reach BER 10^{-3}, and the complexity as the maximum number of real multiplications per symbol period each detector may require.

The complexity reduction of the PDF detector over the BODF detector is greatest for fast-fading channels. For example, Fig. 3 demonstrates the performance versus complexity trade-off between the MMSE versions of the linear, PDF, and BODF detectors for $N \times N$ channels with 16- and 64-QAM inputs, assuming that the channel estimate is updated every symbol period ($L = 1$). Fig. 3 clearly demonstrates that most of the BODF detector's performance improvement over the linear detector is also achieved by the PDF detector, but with only a fraction of the complexity increase. For example, upgrading from the linear to BODF detector when $N = 3$ and the input alphabet is 64-QAM reduces the necessary SNR by 3.9 dB, but requires 79 additional multiplications per symbol period. On the other hand, upgrading from the linear to PDF detector reduces the necessary SNR by 3.6 dB, while requiring only 22 additional multiplications per symbol period. Therefore, in terms of the additional complexity required beyond that of an existing linear detector, the PDF detector is 72% less complex than the BODF detector. In terms of an absolute performance-complexity trade-off, the PDF detector performs 0.3 dB worse than the BODF detector, but is 21% less complex.

We observe in Fig. 3 that the PDF detector performance is closer to that of the BODF detector for larger QAM alphabets and smaller numbers of antennas. On the other hand, the PDF detector decreases complexity more relative to the BODF detector as N increases. The performance gap between the two detectors is smaller for larger alphabets because they require operating at higher SNR, where the performance of the PDF and BODF detectors will converge. For the same reason the performance gap between the zero-forcing PDF and BODF detectors is significantly smaller than for the MMSE versions of these detectors.

V. Conclusion

The partial decision-feedback detector combines the strategies of the BLAST-ordered decision-feedback detector and the linear detector. We have shown that when the goal is to upgrade the performance of the linear detector, while keeping complexity low, the PDF detector offers an attractive performance-complexity trade-off. In particular, the PDF reduces the preprocessing complexity, which is important for fast-fading channels. Specifically, by feeding back only one decision, the PDF detector incurs a small performance loss relative to the BODF detector. In addition, the PDF detector is significantly less complex than the BODF detector. For example, for a 3-input 3-output Rayleigh-fading channel with 64-QAM inputs, the PDF detector is 21% less complex than the BODF detector, yet suffers only 0.3 dB of penalty in SNR.

References

[1] G. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas," *Bell Labs Tech. J.*, pp. 41-59, Autumn 1996.

[2] G. Foschini, G. Golden, R. Valenzuela, P. Wolniansky, "Simplified processing for wireless communication at high spectral efficiency," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 11, pp. 1841-1852, 1999.

[3] S. Verdú, *Multiuser Detection*. Cambridge University Press, 1998.

[4] A. Duel-Hallen, "Decorrelating decision-feedback multiuser detector for synchronous code-division multiple-access channel," *IEEE Trans. Commun.*, vol. 41, no. 2, pp. 285-290, Feb. 1993.

[5] D. W. Waters and J. R. Barry, "Noise-predictive decision-feedback detection for multiple-input multiple-output channels," *IEEE Trans. Signal Processing*, vol. 53, no. 5, pp. 1852-1859, May 2005.

[6] Y. Li and Z. Luo, "Parallel detection for V-BLAST system," in *Proc. IEEE Int. Conf. Commun. 2002*, vol. 1, pp. 340-344.

[7] A. Varanasi, "Near-optimum detection in synchronous code-division multiple-access systems," *IEEE Trans. Commun.*, vol. 39, no. 5, pp. 725-736, May 1991.

[8] W. Choi, R. Negi, J. Cioffi, "Combined ML and DFE decoding for the V-BLAST system," in *Proc. IEEE Int. Conf. Commun. 2000*, vol. 3, pp. 1243-1248, June 2000.

[9] A. Duel-Hallen, "A family of multiuser decision-feedback detectors for asynchronous code-division multiple-access channels," *IEEE Trans. Commun.*, vol. 43, no. 2/3/4, pp. 421-434, Feb./Mar./Apr. 1995.

[10] B. Hassibi, "An efficient square-root algorithm for BLAST," in *Proc. IEEE Conf. on Acoustics, Speech, and Signal Processing 2000*, vol. 2, pp. 737-740.

[11] R. Böhnke, D. Wübben, V. Kühn, and K. Kammeyer, "Reduced complexity MMSE detection for BLAST architectures," in *Proc. IEEE Global Commun. Conf. 2003*, vol. 4, pp. 2258-2262.

[12] D. Waters, "Signal detection strategies and algorithms for multiple-input multiple-output channels," Ph.D. thesis, Georgia Institute of Technology, 2005.