Belief-Directed Sequential Probabilistic Data Association Multiuser Detector

Yufei Huang, Yufang Yin
Department of Electrical Engineering
The University of Texas at San Antonio
San Antonio, TX 78249-0669
E-mail: yhuang@utsa.edu, yyin@lonestar.utsa.edu

Jianqiu (Michelle) Zhang
Department of Electrical and Computer Engineering
University of New Hampshire
Durham, NH 03824.
Email: jianqiu.zhang@unh.edu

***Abstract*—We propose in this paper a novel soft-input-soft-output (SISO) multiuser detector for synchronous CDMA systems. The detector is called the belief-directed sequential probabilistic data association detector (BD-SPDAD). The BD-SPDAD is developed based on a general framework of the probabilistic data association detector (PDAD) proposed in [1]. However, novel extensions to the general framework are proposed for the BD-SPDAD, which results in a low complexity sequential implementation. In specific, the complexity of the BD-SPDAD is reduced from $\mathcal{O}(K^3)$ of the original PDAD to $\mathcal{O}(K^2)$, where K is the number of users in the system. Moreover, we show through simulation that the performance of the BD-SPDAD is comparable and even better than the original PDAD especially at high signal-to-noise regions.**

I. Introduction

Multiuser detection (MUD) for CDMA systems [2] has received a great deal of attention after it was introduced in the eighties. The popularity is largely due to its potential for increasing the capacity of systems. Since optimum MUD is exponential in complexity, numerous suboptimal detectors have been developed with different trade-offs between complexity and performance [3], [4], [5].

Recently, in [6], a very promising iterative soft detector called probabilistic data association detector (PDAD) was proposed based on probabilistic data association (PDA), a very popular approach for multiple target tracking in cluttered environment [7]. In [8], for comparison of many popular suboptimal detectors including the decision feedback detector, coordinate descent, quadratic programming with constraints, semi-definite relaxation, PDAD, branch and bound, and etc., the PDAD stands out to be the best in terms of its performance and complexity. Another appealing advantage of the PDAD is that it provides soft (probabilistic) information about the unknown data bits, which thus makes it naturally applicable to turbo multiuser detection of coded systems.

In one of our recent work [1], we proposed a generalized framework for the PDAD. Based on this framework, some important insights were observed about the original PDAD and connections were able to draw between the popular conditional linear minimum mean squared error (CLMMSE) SISO MUD [9] and the generalized PDAD.

As another extension to the general framework, we propose, in this paper, a reduced complexity algorithm procedure called sequential PDAD (SPDAD). In the SPDAD, the original multiuser system is transformed into several independent multiuser subsystems and the PDAD is applied to each subsystems yet using the information obtained from other subsystems as the priors. An important approach is also proposed to avoid double counting the likelihoods. A specific algorithm called belief-directed SPDAD (BD-SPDAD) is presented, which is designed based on whitened matched filter outputs. The complexity of the BD-SPDAD is reduced from $\mathcal{O}(K^3)$ of the original PDAD to $\mathcal{O}(K^2)$, where K is the number of users in the system. Moreover, we show through simulation that the performance of the BD-SPDAD is comparable at low to mid signal-to-noise (SNR) range yet is much better than the original PDAD at high SNR region.

The remaining of the paper is organized as follows. In section II, system model of multiuser detection is formulated and the concerned problem is presented. The background of generalized PDAD is described in section III. In section IV, the algorithm of the BD-SPDAD is derived. Simulations are included and conclusion is drawn in section V.

II. Problem Formulation

Consider a synchronous CDMA system with a chip rate K users [2]. The matched filter output $\mathbf{y}$ can be expressed according to

$$\mathbf{y} = \mathbf{R}\mathbf{A}\mathbf{b} + \mathbf{n} \tag{1}$$

where $\mathbf{y} = [y_1\ y_2\ \cdots\ y_K]^\mathsf{T}$ ($^\mathsf{T}$ stands for vector or matrix transposition), $\mathbf{A} = \mathrm{diag}\{A_1, \cdots, A_K\}$ is the diagonal matrix of the channel state information, $\mathbf{b} = [b_1\ b_2\ \cdots\ b_K]^\mathsf{T}$ is the user symbol vector, $\mathbf{n}$ is zero mean Gaussian noise with the covariance matrix equal to $\sigma^2\mathbf{R}$, and $\mathbf{R}$ denotes the cross-correlation matrix of the signature waveform.

Let us assume that $\mathbf{R}$, $\mathbf{A}$, and σ^2 are known to the receivers, and the *a priori* probability of each user's data symbol, i.e. $p(b_k)\forall k$, are also available. Our objective is perform multiuser detection given the matched filter output $\mathbf{y}$.

III. The generalized PDAD

We review in this section the generalized PDAD and summarize our previous results. Without loss of generality, let us assume the data signals are antipodally modulated, i.e.,

$\mathbf{b} \in \{-1, 1\}^K$. The goal of PDAD is to obtain the *a posteriori* probability $p(b_k|\mathbf{y}) \ \ \forall k$. Since

$$p(b_k|\mathbf{y}) = \frac{p(\mathbf{y}|b_k)p(b_k)}{\sum_{b_k} p(\mathbf{y}|b_k)p(b_k)}$$

$p(b_k|\mathbf{y})$ can be obtained easily if $p(\mathbf{y}|b_k)$ is known. However, to obtain $p(\mathbf{y}|b_k)$, marginalization must be carried out as

$$p(\mathbf{y}|b_k) = \sum_{\mathbf{b}_{-k}} p(\mathbf{y}|\mathbf{b})p(\mathbf{b}_{-k}) \tag{2}$$

where $\mathbf{b}_{-k}$ represents a $(K-1) \times 1$ signal vector that contains all data signals except b_k. Apparently, (2) is of complexity exponentially increasing with K and thus computationally infeasible for large K. Note the high complexity rises from (2) being a mixture Gaussian. One simplification is however possible by performing data association, i.e., approximating the mixture by a single Gaussian with mean and variance matched to the mixture. But such approximation is still quite rough. To refine it, the PDAD employs an iterative scheme. The canonical algorithm of the generalized PDAD can be described as follows:

Generalized PDA Detector

1) **Initialization:** Set $P_k = p(b_k = 1) \forall k$ and calculate the log-prior ratio (LPrR) as

$$\lambda_{LPrR}(b_k) = \ln \frac{p(b_k = 1)}{p(b_k = -1)} \qquad \forall k; \tag{3}$$

2) For $k = 1$ to K
   - **Interference Cancellation:** Compute $\tilde{\mathbf{y}}_k = \mathbf{W}_k \mathbf{y}$, where $\mathbf{W}_k$s are linear filters to be defined;
   - **Data Association**: Compute $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$ by

$$\boldsymbol{\mu}_k = \mathbf{W}_k \mathbf{m}_k \tag{4}$$

   and

$$\boldsymbol{\Sigma}_k = \mathbf{W}_k \mathbf{Q}_k \mathbf{W}_k^\top \tag{5}$$

   where

$$\mathbf{Q}_k = \sum_{i \neq k} A_i^2 \mathbf{R} \mathbf{e}_i \mathbf{e}_i^\top \mathbf{R} \sigma_{b_i}^2 + \sigma^2 \mathbf{R} \tag{6}$$

$$\mathbf{m}_k = \sum_{i \neq k} A_i \mathbf{R} \mathbf{e}_i \tilde{b}_i \tag{7}$$

   $\sigma_{b_i}^2 = VAR[b_i] = 4P_i(1-P_i)$, $\tilde{b}_i = E[b_i] = 2P_i - 1$, and $\mathbf{e}_i$ is a $K \times 1$ vector whose only nonzero element is the ith element and is 1.
   - **Probability update**: Calculate the log-likelihood ratio (LLR) and log-posterior ratio (LPR) as

$$\lambda_{LLR}(b_k) = 2A_k \mathbf{e}_k^\top \mathbf{R} \mathbf{W}_k^\top \boldsymbol{\Sigma}_k^{-1} (\tilde{\mathbf{y}}_k - \boldsymbol{\mu}_k) \tag{8}$$

   and

$$\lambda_{LPR}(b_k) = \lambda_{LLR}(b_k) + \lambda_{LPrR}(b_k). \tag{9}$$

   Update P_k by

$$P_k = \frac{1}{2}\Big\{1 + \tanh\big[\frac{1}{2}\lambda_{LPR}(b_k)\big]\Big\}. \tag{10}$$

3) **Convergence testing:** If P_ks converge, go to 4. Otherwise, go back to 2.
4) **Detection:** Detect b_ks according to

$$\hat{b}_k = \begin{cases} 1 & \text{if} \quad P_k \geq 0.5 \\ -1 & \text{otherwise} \end{cases} \tag{11}$$

Under the above general framework, we examined in [1] two different cases on the choice of the interference cancellation (IC) filter $\mathbf{W}_k$. In the first case, we examined invertible $\mathbf{W}_k$s, which include $\mathbf{W}_k = \mathbf{I}_K$, $\mathbf{W}_k = \mathbf{A}^{-1}\mathbf{R}^{-1}$, and $\mathbf{W}_k = (\mathbf{R} + \sigma \mathbf{A}^{-2})^{-1}$, and they correspond to applying no interference cancellation, a decorrelating detector, and a LMMSE detector in the interference cancellation step of the general PDAD algorithm, respectively. We showed that in this case no $\mathbf{W}_k$ is needed for the calculation of $\lambda(b_k)$ and thus any choice other than $\mathbf{W}_k = \mathbf{I}_K$ or any interference cancellation step will be computationally redundant. Since the original PDAD in [6] corresponds to the case for $\mathbf{W}_k = \mathbf{A}^{-1}\mathbf{R}^{-1}$. This result indicates that the original PDAD is computational inefficient. In the second case, we examined IC by the CLMMSE, i.e., $\mathbf{W}_k \triangleq \bar{\mathbf{W}}_k = A_k \mathbf{e}_k^\top \mathbf{R}(\mathbf{R}\mathbf{V}_k\mathbf{R} + \sigma^2\mathbf{R})^{-1}$ with $\mathbf{V}_k = \sum_{i \neq k} A_i^2 \sigma_{b_i}^2 \mathbf{e}_i \mathbf{e}_i^\top + A_k^2 \mathbf{e}_k \mathbf{e}_k^\top$. We showed that in this case the LLR is the same as that of the popular SISO MUD [9]. Since the SISO detector performs only one iteration for $k = 1$ to K, we can consider the SISO detector as one iteration of the generalized PDAD. Nonetheless, from a PDAD perspective, the probabilities of b_ks may not convergence after one iteration. As a result, the PDAD paradigm suggests that more iterations until convergence would produce improved soft information and thus better overall bit-error-rate performance for the turbo MUD!

We also examine the connection between the above two choices of $\mathbf{W}_k$. Surprisingly, we show that the two PDAD algorithms from the two different $\mathbf{W}_k$ are equivalent! Since no IC step is needed in the first choice of $\mathbf{W}_k$, we conclude that the use of the conditional LMMSE interference cancellation $\bar{\mathbf{W}}_k$ in the PDAD is also redundant and so is it in the SISO MUD!

IV. Sequential PDA Multiuser Detector

Let us consider an extension to the canonical algorithm of the generalized PDAD. We choose the IC filter as

$$W_k = \mathbf{e}_k^\top \tilde{\mathbf{W}}_k \tag{12}$$

where $\tilde{\mathbf{W}}_k$ is a linear filter to be specified and note that W_k is a scalar. One important restriction on $\tilde{\mathbf{W}}_k$ is that the resulting filter output $\tilde{y}_k$ must be independent of $\tilde{y}_i \ \forall i \neq k$. From a probabilistic perspective, we require that the joint likelihood can be expressed as

$$p(\tilde{y}_1, \cdots, \tilde{y}_K|\mathbf{b}) = \prod_{k=1}^{K} p(\tilde{y}_k|\mathbf{b}). \tag{13}$$

As an example, we see that the LMMSE IC filter cannot be a candidate filter of (12), even though it has the exact structure as (12). This is because that due to $\mathbf{V}_k$, the LMMSE IC filter output is dependent on other IC outputs. As a result of (13), the posterior distribution can be calculated in a sequential fashion in K steps and at the lth step, the APPs can be expressed by

$$p(b_k|\tilde{y}_1, \cdots, \tilde{y}_l) \propto p(\tilde{y}_l|b_k)p(b_k|\tilde{y}_1, \cdots, \tilde{y}_{l-1}) \quad \forall k \quad (14)$$

where $p(b_k|\tilde{y}_1, \cdots, \tilde{y}_{l-1}) = p(b_k)$ when $l = 1$. Notice that the same difficulty as in (2) arises in evaluating $p(\tilde{y}_k|b_k)$ and PDA can be thus applied to approximate it.

An alternative view on the above sequential update can be cast from a system perspective. We can regard the problem as K, rather than one, multiuser systems, with the output of the lth system as $\tilde{y}_l$. Since all the outputs are independent, we can then calculate the APPs of the bits sequentially from the first to the Kth system and, for the lth system, the APPs obtained from the $l-1$th system will be used as the *a priori* probabilities. Since each system is still a multiuser system, same complexity difficulty exits, which, however, can be overcome by PDA.

If there is no approximation in calculating the likelihood $p(\tilde{y}_k|b_k)$ at each step, then after K steps we would obtain the desired APPs exactly from (14). However, since PDA produces approximated APPs at each steps, a 'smoothing' process is thus preferred after an iteration of K steps. In our design, smoothing is performed by repeating another iteration of K similar steps with, however, the most recently estimate of APPs as the priors. When recycling the APPs as the priors, it is important to avoid double counting the likelihoods. To see the problem, suppose we are at the lth step of the smoothing process and we want to update the APP $p(b_k|\tilde{y}_1, \cdots, \tilde{y}_K)$ based on $\tilde{y}_l$. According to the Bayes' theorem we have

$$p(b_k|\tilde{y}_1, \cdots, \tilde{y}_K) \propto p(\tilde{y}_l|b_k)p(b_k|\tilde{\mathbf{y}}_{-l}) \quad (15)$$

where $\tilde{\mathbf{y}}_{-l}$ is a $(K-1) \times 1$ vector including all the filter outputs except $\tilde{y}_l$. Notice to recycle the likelihood $p(\tilde{y}_l|b_k)$ for an update on the APP, the prior probability is $p(b_k|\tilde{\mathbf{y}}_{-l})$, which does not include $\tilde{y}_l$. However, unlike during the first iteration, the most recent APP on b_k is an estimate on $p(b_k|\tilde{y}_1, \cdots, \tilde{y}_K)$, which already incorporates the likelihood information about $\tilde{y}_l$ and thus the likelihood $p(\tilde{y}_l|b_k)$ would be counted twice if the recent APP on b_k were used directly as a prior. The problem can be avoided by removing the likelihood from the APP before being used as the prior. The process is more conveniently performed on the log-ratio scale

$$\ln \frac{p(b_k = 1|\tilde{\mathbf{y}}_{-l})}{p(b_k = -1|\tilde{\mathbf{y}}_{-l})} \quad (16)$$

$$= \ln \frac{p(b_k = 1|\tilde{y}_1, \cdots, \tilde{y}_K)}{p(b_k = -1|\tilde{y}_1, \cdots, \tilde{y}_K)} - \ln \frac{p(\tilde{y}_l|b_k = 1)}{p(\tilde{y}_l|b_k = -1)} \quad (17)$$

and actually implemented as

$$\lambda_{LPrR}^{(m,l)}(b_k) = \lambda_{LPR}^{(m,l-1)}(b_k) - \lambda_{LLR}^{(m-1,l)}(b_k) \quad (18)$$

where the superscript (m, l) denotes the m-th iteration and the l-th step. As in the regular PDA, the above procedure iterates until all the APPs converge. In light of the above discussion, we term the sequential calculation of the APPs by PDA as the sequential PDA detector (SPDAD).

As to the specific choice of the IC filter $\tilde{\mathbf{W}}_k$ in (12), one example is the whitened matched filter, i.e.,

$$\tilde{\mathbf{W}}_k = \mathbf{F}^{-\top}$$

where $\mathbf{F}$ is the uniquely defined $K \times K$ lower triangular matrix obtained from the Cholesky factorization of $\mathbf{R}$. Then $\tilde{y}_l$, the output of the lth subsystem, can be expressed as

$$\tilde{y}_l = \mathbf{e}_l^\top \mathbf{F}^{-\top} \mathbf{y} = \sum_{k=1}^{l} A_k F_{lk} b_k + \bar{n}_l \quad (19)$$

where F_{lk} is the lkth element of $\mathbf{F}$ and $\bar{n}_l$ is white Gaussian noise with variance $\frac{N_0}{2}$. Apparently, $\tilde{y}_l$ is independent of other IC outputs. Based on $\tilde{y}_l$, PDAD can be performed to update P_k for $k = 1, \cdots, l$ and, for the update of P_k, the LLR is calculated according to

$$\lambda_{LLR}(b_k) = 2A_k F_{lk}(\tilde{y}_l - \mu_k)/\Sigma_k \quad (20)$$

where μ_k and Σ_k are obtained from (4) and (5) as

$$\mu_k = \sum_{i=1, i\neq k}^{l} A_i F_{li} \tilde{b}_i \quad (21)$$

and

$$\Sigma_k = \sum_{i=1, i\neq k}^{l} A_i^2 F_{li}^2 \sigma_{b_i}^2 + \frac{N_0}{2}. \quad (22)$$

We call the SPDAD derived from the whitened matched filter as the belief-directed SPDAD (BD-SPDAD). In the implementation of the BD-SPDAD, we want to point out that we allow only one sweep for the PDAD from $k = 1$ to l when calculating the APPs at each subsystem l. It is because the information on the user bits in $\tilde{y}_l$ can be vague and more PDA sweeps might result in suboptimum APPs. Since no matrix inversion is need when performing data association, the complexity of each iteration in BD-SPDAD is of $\mathcal{O}(K^2)$. Compared with the complexity of $\mathcal{O}(K^3)$ of the original PDAD, there is a major reduction especially for systems with large numbers of users. The algorithm of the BD-SPDAD can be summarized in the following chart

Belief-Directed Sequential PDA Detector

1) **Initialization**: Set $m = 1$, $P_k = p(b_k) \forall k$ and calculate $\lambda_{LPrR}(b_k) \forall k$.
2) **WMF output**: Calculate $\tilde{y}_k \forall k$ from (19).
3) For $l = 1$ to K
   - For $k = 1$ to l
     - **Data Association**: Compute μ_k and Σ_k from (21) and (22);
     - **Belief Calculation**: Calculate the LLR $\lambda_{LLR}^{(m,l)}(b_k)$, LPR $\lambda_{LPR}(b_k)$, and P_k according to (20), (9), and (10);

- **Belief passing**: Set, for $k = 1, \cdots, l$

$$\lambda_{LPrR}(b_k) = \begin{cases} \lambda_{LPR}(b_k) & \text{if } m = 1 \\ \lambda_{LPR}(b_k) - \lambda_{LLR}^{(m-1,l)}(b_k) & otherwise \end{cases}$$

4) **Convergence testing**: If the APPs converge, go to 5) . Otherwise, go back to 3) and set $m = m + 1$.
5) **Detection**: Detect b_ks according to (11).

V. Simulation and Conclusion

To evaluate the bit-error-rate (BER) performance of the BD-SPDAD, we considered a 15-user system with random signature sequences of length 17. In addition, we assumed each users have equal power E_b and the SNR is calculated as E_b/N_0. In our simulation, the BER at a specific SNR was computed as the average BER among all users. For all the tested PDAD algorithms, we stoped the iteration when either P_ks converge or a maximum number of iteration has been reached, which is set as 10 in the simulation.

In Figure 1, the BER performance versus E_b/N_0 is depicted, where the BD-SPDAD is compared with the PDAD, the BD-SPDAD after 1 iteration (BD-SPDAD-itr1), the PDAD after 1 iteration (PDAD-itr1), and the decision feedback detector (DFD). Even though the DFD is computationally simple, it has an apparent performance loss as E_b/N_0 increases. We observe that the performance of the BD-SPDAD is very close to that of PDAD in the lower E_b/N_0 area from 0 to 20 dB, yet, in the high E_b/N_0 area above 20 dB, the BD-SPDAD actually outperforms the PDAD. And the improvement is more pronounced as E_b/N_0 increases.

We also find the gap between the BD-SPDAD-itr1 and the BD-SPDAD is significantly smaller than that between the PDAD-itr1 and PDAD. Especially, BD-SPDAD-itr1 is almost superposed with BD-SPDAD as E_b/N_0 is above 18 dB. This means the improvement for BD-SPDAD after one iteration is really very small.

In Figure 2, we present the average number of iterations, another important indication about algorithm convergence. It shows that overall the BD-SPDAD and the PDAD converge at almost same average number of iterations, yet the BD-SPDAD requires less iterations than the PDAD in $E_b/N0$ region from 6dB to 20dB. Considering the complexity reduction of the BD-SPDAD over the PDAD in one iteration, we can see the BD-SPDAD has much less computation.

Based on the simulation results depicted in Figure 1 and 2, it is not hard to draw the conclusion for the BD-SPDAD: The BD-SPDAD provides comparably good and even improved performance than the PDAD with much less computational cost.

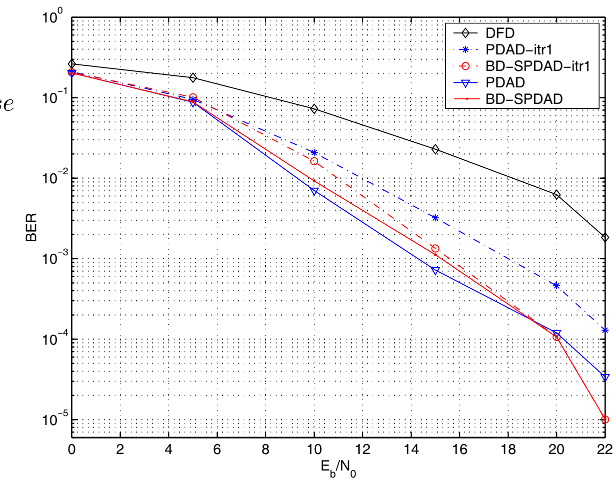

Fig. 1. BERs as functions of E_b/N_0.

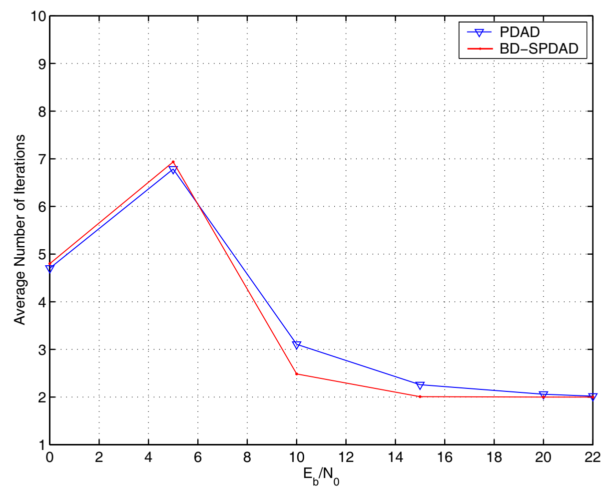

Fig. 2. Average number of iterations as functions of E_b/N_0.

References

[1] Y. Huang and J. Zhang, "A generalized probabilistic data association multiuser detector for cdma systems," in *the IEEE International Symposium on Information Theory*, Chicago, June 2004.
[2] S. Verdú, *Multiuser Detection*. New York: Cambridge University Press, 1998.
[3] M. K. Varanasi and B. Aazhang, "Mear-optimum detection in synchronous Code-Division Multiple-Access systems," *IEEE Transactions on Communications*, vol. 39, no. 5, pp. 725–736, May 1991.
[4] A. Duel-Hallen, "Decorrelating decision-feedback multiuser detector for synchronous Code-Division Multiple-Access channel," *IEEE Transactions on Communications*, vol. 41, no. 2, pp. 285–290, February 1993.
[5] D. Divsalar, M. K. Simon, and D. Raphaeli, "Improved parallel interference cancellation for CDMA," *IEEE Trans. Commun.*, no. 46, pp. 258–268, Feb. 1998.
[6] J. Luo, K. R. Pattipati, P. K. Willett, and F. Hasegawa, "Near-optimal multiuser detection in synchronous CDMA using probabilistic data association," *IEEE COMMUNICATION LETTERS*, vol. 5, no. 9, pp. 361–363, Sept. 2001.
[7] Y. Bar-Shalom and X. R. Li, *Estimation and Tracking: Principles, Techniques, and Software*. Dedham, MA: Artech House, 1993.
[8] F. Hasegawa, J. Luo, K. Pattipati, and P. Willett, "Speed and accuracy comparison of techniques for multi-user detection in synchronous CDMA," *IEEE Trans. on Comm.*, 2004, to appear, available at http://www.isr.umd.edu/˜rockey/Publications.htm.
[9] X. Wang and H. V. Poor, "Iterative (turbo) soft interference cancellation and decoding for coded CDMA," *IEEE Trans. Commun.*, vol. 47, pp. 1046–1061, July 1999.