Analysis and Optimization of CDMA Systems With Chip-Level Interleavers

Lihai Liu, Student Member, IEEE, Jun Tong, and Li Ping, Member, IEEE

Abstract—In this paper, we present an unequal power allocation technique to increase the throughput of code-division multiple-access (CDMA) systems with chip-level interleavers. Performance is optimized, respectively, based on received and transmitted power allocation. Linear programming and power matching techniques are developed to provide solutions to systems with a very large number of users. Various numerical results are provided to demonstrate the efficiency of the proposed techniques and to examine the impact of system parameters, such as iteration number and interleaver length. We also show that with some very simple forward error correction codes, such as repetition codes or convolutional codes, the proposed scheme can achieve throughput reasonably close to that predicted by theoretical limit in multiple access channels.

Index Terms—Code-division multiple-access (CDMA), iterative detection, linear programming, multiuser detection, power allocation.

I. INTRODUCTION

ITERATIVE multiuser detection has been widely investigated as a potential approach to enhance the performance of code-division multiple-access (CDMA) systems and significant progress has been made recently [1]–[5]. The complexity of CDMA multiuser detection has always been a serious concern for practical systems, which increases rapidly with the number of users (e.g., per-user complexity increases quadratically for the well-known minimum mean-squared error (MMSE)-based approach in [3]). The analysis of multiuser detection for random waveform CDMA systems is also a difficult issue. This has been studied in detail in [4], [6]–[8], and [18] for various CDMA multiuser detectors. The key problem is to find the distribution of the eigenvalues of the correlation matrix characterizing the correlation among the signature sequences. The modeling involved is not trivial, as shown in [7]. Large random matrix theory [9] has been employed to tackle the problem [6]–[8] but the mathematics involved is quite demanding. The large system assumption is also not so intuitive for small systems.

In [10]–[12], chip-level interleaving is proposed for random waveform CDMA systems. The associated chip-by-chip (CBC) estimation algorithm [10]–[12] is essentially a low-cost iterative soft-cancellation technique to treat both multiple-access interference (MAI) and intersymbol interference (ISI) [3]. It has been shown [12] that the CBC algorithm can achieve performance close to the theoretical limits at low-to-medium throughput for systems with equal power control. The computational cost of the CBC algorithm is very low, being independent (when normalized to each user) of the total number of users. Simulation work involving hundreds of simultaneous users can be easily accomplished.

In this paper, we will develop a fast technique to characterize the CBC detection process by tracking the signal-to-noise ratio (SNR) evolution [4], [13], [14]. With chip-level interleavers, the performance of the CBC algorithm is independent of the correlation among spreading sequences. (In fact, the same spreading sequence can be used by all the users.) As a result, the analysis of the CBC algorithm is a straightforward issue and the derivation is very concise.

It has been shown in [15] that unequal power control together with iterative multiuser detection can enhance the performance of CDMA systems. This principle has been further studied in [16]–[18]. In this paper, we will show that CDMA with chip-level interleavers provides a very simple and efficient framework to realize the principle. With the help of the SNR evolution technique, the optimized power profile can be found by exhaustive searching when user number is small. Linear programming and power matching techniques are developed to provide solutions to systems with a very large number of users. Various numerical results are provided to demonstrate the efficiency of the proposed techniques and to examine the impact of system parameters such as iteration number and interleaver length. Both analysis and simulation show that very high throughput and good power efficiency can be achieved simultaneously, which (together with other well-known features of conventional CDMA systems such as diversity against fading, easy treatment of the multipath effect and mitigation against cross-cell interference) are attractive for future wireless communication systems.

II. TRANSMITTER AND RECEIVER STRUCTURES

A. Transmitter

Consider the $K$-user uplink CDMA system shown in Fig. 1. At the transmitter side (the upper part of Fig. 1), the information bit stream $d_k$ for user-$k$ is first encoded by a generalized encoder (ENC) (assumed to be binary in this paper), interleaved and then transmitted over a Gaussian multiple-access channel (MAC). Here, the generalized encoder can include a conventional forward error correction (FEC) encoder or a spreader or both. Unlike conventional CDMA in which interleaving is applied between FEC coding and spreading, interleaving is applied at the final stage of the transmitter, i.e., at the chip level, in the
system shown in Fig. 1. Several frames of chip sequences at the output of a spreader are collected and interleaved together. Such a scheme is referred to as chip-interleaved CDMA (cil-CDMA) in [10]. A key principle here is that the interleavers \(\{π_k\}\) should be different for different users. In this paper, we assume that \(\{π_k\}\) are generated randomly and independently. The use of independent interleavers has an interesting consequence: The signals from different users are separable even when the same spreading sequence is used for all the users. In this case, spreading can be replaced by repetition coding or by other low-rate coding. Then, interleaving remains the only means to distinguish users, and such a scheme is referred to as interleaved-division multiple access (IDMA) [11].

We use \(c_k = \{c_k(j)\}\) and \(x_k = \{x_k(j)\}\) with \(c_k(j), x_k(j) ∈ \{+1, −1\}\) to denote the chip streams before and after interleaving, respectively. The received signal can be written as

\[
r(j) = \sum_{k=1}^{K} h_k x_k(j) + n(j), \quad j = 1, 2, \ldots, J
\]

where \(h_k\) is the coefficient for user-\(k\) representing the combined effect of power control and channel loss. \(\{n(j)\}\) are samples of an additive white Gaussian noise (AWGN) process with zero-mean and variance \(σ^2 = N_0/2\), and \(J\) is the frame length. For simplicity, we only consider real \(\{h_k\}\) but the results can be easily extended to quadrature channels [19].

**B. Elementary Signal Estimator (ESE) Function**

The receiver (the lower part of Fig. 1) consists of an ESE and a bank of \(K\) single-user a posteriori probability (APP) decoders (DECs), operating iteratively. Let us concentrate on the detection for user-\(k\). We rewrite (1) as

\[
r(j) = h_k x_k(j) + ζ_k(j)
\]

where \(ζ_k(j) = \sum_{k' ≠ k} h_k x_k(j) + n(j)\) \(3\) represents the distortion component with respect to \(x_k(j)\). We treat each \(x_k(j)\) as an independent random variable with mean \(E(x_k(j))\) and variance \(Var(x_k(j))\). According to the central limit theorem, we can approximate (3) by a Gaussian random variable with

\[
E(ζ_k(j)) = \sum_{k' ≠ k} h_k E(x_k(j))\]

\[
Var(ζ_k(j)) = \sum_{k' ≠ k} |h_k|^{2} Var(x_k(j)) + σ^2
\]

where \(E(x_k(j))\) and \(Var(x_k(j))\) [see (8) and (9)] are initialized to 0 and 1, respectively. Under the hypothesis that \(x_k(j)\) is either +1 or −1, the output of the ESE can be computed as [19]

\[
e_{\text{ESE}}(x_k(j)) = \frac{2h_k}{\sqrt{Var(ζ_k(j))}} (r(j) - E(ζ_k(j))).
\]

**C. DEC Function**

The DECs in Fig. 1 perform soft-in–soft-out decoding after the operations in the ESE are completed. Using \(e_{\text{ESE}}(x_k(j))\) as the a priori information, DEC-\(k\) generates extrinsic log-likelihood ratios (LLRs) \(e_{\text{DEC}}(x_k(j)), ∀j\) for user-\(k\). As an example, we consider a trivial length-\(S\) repetition code. Let \(d_k = \{d_k(i)\}\) be the input data sequence of user-\(k\) (see Fig. 1) and denote by \(I_k(i)\) the set of location indexes for the \(S\) replicas related to \(d_k(i)\) in the interleaved sequence \(\{x_k(j)\}\). The extrinsic LLR with respect to \(x_k(j)\) is computed as

\[
\tilde{e}_{\text{ESE}}(x_k(j)) = \log \Pr(d_k(i) = +1 | d_k(i) = -1) - e_{\text{ESE}}(x_k(j))
\]

\[
= \sum_{j' ∈ I_k(i) \cap j' ≠ j} e_{\text{ESE}}(x_k(j')).
\]

The APP decoding techniques for other codes can be found in [20]–[22]. Furthermore, if an outer code is used together with the repetition code as the inner code (see [19]), then the DEC function consists of the operations in (7) cascading with the APP decoding for the outer code. Since the cost involved in (7) is very low, the overall cost is usually dominated by the outer code.

**D. Overall Iterative Detection Procedure**

The extrinsic LLRs produced by the DECs contain refined information for the transmitted chips and so they can be used to improve the ESE outputs. In this case, we treat the extrinsic LLRs from the DECs as a priori LLRs and update the a priori means and variances required in (4) and (5) as

\[
E(x_k(j)) = \tanh \left(\frac{\tilde{e}_{\text{ESE}}(x_k(j))}{2}\right), \quad ∀k, j
\]

\[
Var(x_k(j)) = 1 - (E(x_k(j)))^2, \quad ∀k, j.
\]
This is a standard treatment in a turbo-type detection process [3]. We can then apply the outcome of (8) and (9) in the CBC algorithm and start the next iteration. This process is repeated for a preset number of times before a hard decision decoding is applied to produce the final outputs.

The above CBC detection process is summarized as follows.

1) Compute the interference means and variances using (4) and (5).
2) Compute \( \{e_{ESE}(x_k(j))\} \) using (6).
3) Perform the DEC function to generate \( \{f_{ESE}(x_k(j))\} \).
4) Update the means and variances using (8) and (9).

The computational complexity of the CBC detection process is very low, being independent (when normalized to each user) of the total number of users. Refer to [11] and [12] for the details.

III. PERFORMANCE EVALUATION

Next, we outline a SNR evolution technique [4], [13] to evaluate the performance of the CBC algorithm. Again, we will concentrate on real channels in this section, but the principle is applicable to quadrature channels [19].

By substituting (2) into (6), we have

\[
e_{ESE}(x_k(j)) = \frac{2h_k}{\text{Var} \left( \tilde{z}_k(j) \right)} \left( h_k x_k(j) + \zeta_k(j) - E \left( \zeta_k(j) \right) \right),
\]

(10)

The SNR in \( e_{ESE}(x_k(j)) \) with respect to \( x_k(j) \) after observing \( r \) is calculated as

\[
\text{SNR}_k(j) = \frac{E \left( \frac{2h_k h_k^* x_k(j) + \zeta_k(j) - E \left( \zeta_k(j) \right) }{\text{Var} \left( \tilde{z}_k(j) \right)} \left| r \right. \right)^2 }{E \left( \frac{2h_k h_k^* \zeta_k(j)}{\text{Var} \left( \tilde{z}_k(j) \right)} \right)^2 } = \frac{|h_k|^2}{\text{Var} \left( \zeta_k(j) \right)}.
\]

(11)

The above SNR is computed for a particular received \( r \). We define the average SNR as

\[
\text{SNR}_k \equiv E(\text{SNR}_k(j)) = E \left( \frac{|h_k|^2}{\text{Var} \left( \zeta_k(j) \right)} \right) \geq \frac{|h_k|^2}{E(\text{Var} \left( \zeta_k(j) \right))}
\]

(12)

where the expectation is taken over all possible \( r \). (Note: The expectation in (11) is conditional on receiving a particular \( r = \{r(j)\} \), so \( \text{SNR}_k(j) \) is a function of \( j \). The expectation in (12) is unconditional, so \( \text{SNR}_k \) is not a function of \( j \). It is still a function of \( k \) since \( h_k \) (assumed to be a constant) can be different for different \( k \). The last inequality in (12) follows from Jensen’s inequality [23] since \( \text{SNR}_k(j) \) is a convex function of \( \text{Var} \left( \zeta_k(j) \right) \). From (5), we have

\[
E(\text{Var} \left( \zeta_k(j) \right)) = \sum_{k' \neq k} |h_k|^2 |v_{k'}| + \sigma^2
\]

(13)

where

\[
v_k \equiv E \left( \text{Var} \left( x_k(j) \right) \right).
\]

(14)

Substituting (13) and (14) into (12), we obtain a lower bound for \( \text{SNR}_k \)

\[
\text{SNR}_k \geq \gamma_k \equiv \sum_{k' \neq k} |h_k h_k^* v_k| + \sigma^2, \quad \forall k.
\]

(15)

Recall that \( \text{Var} \left( x_k(j) \right) \) in (9) is calculated based on \( \tilde{E}_{ESE}(x_k(j)) \), the deinterleaved form of \( e_{ESE}(x_k(j)) \). The latter is produced by the DEC using \( \{e_{ESE}(x_k(j))\} \) as its inputs. Therefore, \( v_k \) in (14) is a function of the input SNR \( \text{SNR}_k \). We express this function as

\[
v_k \equiv f(\text{SNR}_k).
\]

(16)

We also define the bit-error rate (BER) performance for the DEC \( k \) as a function of \( \text{SNR}_k \) as

\[
\text{BER} \equiv g(\text{SNR}_k).
\]

(17)

Both \( f(\cdot) \) and \( g(\cdot) \) can be obtained by the Monte Carlo method similar to [4], [13], [14], and [18].

Fig. 2 shows two examples of \( f(\cdot) \) and \( g(\cdot) \). One is for a length-16 \( S = 16 \) repetition code, and the other is for a concatenation of a rate-1/2 convolutional code with generator polynomials \( [23, 35]_8 \) followed by a length-8 repetition code. In general, both \( f(\text{SNR}_k) \) and \( g(\text{SNR}_k) \) are decreasing functions of \( \text{SNR}_k \) over \([0, \infty)\) with \( f(0) = 1 \), \( g(0) = 0.5 \), \( f(\infty) = 0 \), and \( g(\infty) = 0 \).

Based on the lower bound in (15), we can apply the following recursion to obtain a lower bound of \( \text{SNR}_k \) at the \( q \)th iteration:

\[
\gamma_k^{(q)} = \sum_{k' \neq k} |h_k h_k^* f \left( \gamma_k^{(q-1)} \right) + \sigma^2, \quad \forall k.
\]

(18)

where \( \gamma_k^{(q)} \) is the output SNR of the ESE at the \( q \)th iteration of the CBC algorithm. Initially, we start with \( f(\gamma_k^{(0)}) = 1 \), for all \( k \), implying no feedback from the DECs. (Notice that the maximum variance of a binary variable over \( \{+1, -1\} \) is 1.) Repeating (18), we can track the SNR evolution for the iterative process in the CBC algorithm. During the final iteration, we can estimate the BER performance by substituting the final SNR.
values \(\{\gamma_k^{(Q)}\}\) into (17), where \(Q\) is the maximum number of iterations. We summarize the SNR evolution process as follows, which is illustrated graphically in Fig. 3.

**The SNR Evolution Process for the CBC Algorithm**

1) Initialization: \(f(\gamma_k^{(0)}) = 1, \forall k\).

2) SNR updating

\[
\gamma_k = \frac{|h_k|^2}{\sum_{k' \neq k} |h_k|^2 f(\gamma_{k'}^{(q-1)}) + \sigma^2},
\]

for \(k = 1 : K\) and \(q = 1 : Q\). (19)

3) Termination: BER for user-\(k = g(\gamma_k^{(Q)}), \forall k\).

Strictly speaking, the proposed performance evaluation method is only valid for infinite interleaver length, since only in this case is the independent identically distributed (i.i.d.) assumption mentioned above correct. Also, since \(\gamma_k\) is a lower bound on the actual SNR [see (15)], the performance obtained by SNR evolution may be pessimistic. Overall, we have observed that even for finite interleaver lengths, SNR evolution and simulation are in good agreement as shown by simulation results later.

The above derivation can be compared with the asymptotic performance analyses of various iterative multiuser detectors for random CDMA systems, such as the parallel interference cancellation (PIC), serial interference cancellation (SIC) detectors [4], [18], and the decision feedback detector [6]. The performance of conventional CDMA systems is a function of spreading sequences. The characterization of the correlation among the spreading sequences is a difficult issue, and the solution based on large random matrices is very mathematically demanding [6]–[8], [18]. In addition, the results based on the large system assumption may not be applicable for small systems. However, it can be verified that the selection of spreading sequences no longer affects system performance after chip-level interleavers are introduced. This greatly simplifies the problem. Consequently, the SNR analysis presented above is very concise, as can be seen by comparing the discussion in this section with the large system method detailed in [7].

**IV. PERFORMANCE OPTIMIZATION BASED ON RECEIVED POWER ALLOCATION**

As demonstrated in [15]–[18], a proper unequal power control can be used to enhance CDMA system performance. The intuition is that the strong power users can converge first and benefit the weak power users. For ideal codes or “good” codes, perfect or near perfect cancellation can be made with a successive stripping technique [16], [17]. However, for “poor” codes, the stripping technique may result in an excessive amount of energy use. In the following, we show that this difficulty can be overcome for the proposed system using the iterative detection approach together with power allocation. This method is very efficient even for “very poor” codes (such as repetition codes). In fact, we will see that for very high rate applications, repetition codes are sufficient to yield good performance.

We will discuss the received power allocation problem in this section, and leave the transmitted power allocation problem to the next section. As shown below, the first problem can be approximately resolved by a linear programming technique.

**A. Problem Formulation**

Return to the SNR evolution process for the CBC algorithm as discussed in Section III. Assume that we can adjust the received power \(|h_k|^2\) for user-\(k\) through a certain power control mechanism. Our objective is to minimize the sum received power \(P_k = \sum_k |h_k|^2\), while achieving a specified minimum SNR value \(\Gamma\) for all users after \(Q\) iterations (i.e., \(\gamma_k^{(Q)} \geq \Gamma, \forall k\)). Here, \(\Gamma\) can be determined by the specified BER_{max}, i.e., the maximum allowed BER, by \(\Gamma = g^{-1}(\text{BER}_{max})\) with \(g(\cdot)\) given in (17). The problem can be formulated as follows.

**The received power optimization problem (Form I):**

Find the distribution \(\{h_k|^2\}\) that minimizes \(P_k\), subject to \(\gamma_k^{(Q)} \geq \Gamma, \forall k\), where \(\{\gamma_k^{(Q)}\}\) are obtained through the following SNR evolution process (with initialization \(f(\gamma_k^{(0)}) = 1, \forall k\))

\[
\gamma_k = \frac{|h_k|^2}{\sum_{k' \neq k} |h_k|^2 f(\gamma_{k'}^{(q-1)}) + \sigma^2},
\]

for \(k = 1 : K\) and \(q = 1 : Q\). (20)

**B. Linear Programming Approach**

When the number of users \(K\) is small, the above problem can be solved by searching all possible values of \(|h_k|^2\). When \(K\) is large, however, the exhaustive search method becomes impractical. The problem in this case can be nevertheless approximately solved by linear programming as follows. To facilitate the discussion, we quantize the received power \(P\) into \(M + 1\) discrete values: \(\{P(m), m = 0, 1, \ldots, M\}\) with \(P(m-1) < P(m), m = 1, \ldots, M\). This means that \(P(m) = |h_k|^2\) for some \(k\). Again, we assume that \(h_k\) includes both power control factor and channel loss. We partition \(K\) users into \(M + 1\) groups according to their power levels. Let \(\lambda(m)\) denote the number of
users assigned with power level $P(m)$ and let $y(m)$ denote the total power of these $\lambda(m)$ users. As such

$$\sum_m \lambda(m) = K \quad (21a)$$

$$y(m) = \lambda(m)P(m) \quad (21b)$$

and the sum received power is

$$P_{\text{total}} = \sum_m \lambda(m)P(m) = \sum_m y(m),$$

Denote by $\gamma(m)$ the SNR for the users in the $m$th group with power $P(m)$. Define

$$I = \sum_m y(m)f(\gamma(m)) + \sigma^2$$

which is the total interference power (including noise) after soft cancellation in the ESE output. When $K$ is large, (20) can be approximated as

$$\gamma(m)(q) = \frac{P(m)}{I(q) - P(m)f(\gamma(m)(q-1))} \approx \frac{P(m)}{I(q)}$$

where $I(q)$ denotes the value of $I$ at the $q$th iteration. Using (22) and (23), we have the update rule

$$I(q) = \sum_m y(m)f\left(\frac{P(m)}{I(q-1)}\right) + \sigma^2.$$  

Equation (24) characterizes the evolution of the total interference variance at each iteration. If iterative detection converges, $I(q)$ should be lower than $I(\infty)$. Equivalently, we can write the convergence condition as

$$\sum_m y(m)f\left(\frac{P(m)}{I}\right) + \sigma^2 \leq (1 - \delta)I, \quad I_{\text{min}} \leq I \leq I_{\text{max}}$$

where $0 < \delta < 1$ is a decay factor that controls the convergence speed. $I_{\text{max}}$ and $I_{\text{min}}$ specify the total interference at the beginning and end of the iterative detection. In summary, we re-formulate the optimization problem as

**The received power optimization problem (Form II):** Find $\{y(m)\}$ to minimize $P_{\text{total}} = \sum_m y(m)$ subject to

$$\sum_m y(m)f\left(\frac{P(m)}{I}\right) + \sigma^2 \leq (1 - \delta)I, \quad I_{\text{min}} \leq I \leq I_{\text{max}} \quad (26a)$$

$$y(m) \geq 0, \quad m = 0, 1, \ldots, M. \quad (26b)$$

We now examine the impact of the approximation in (23). We use a four-user system ($K = 4$), in which each user employs a rate-1/2 convolutional code with generator polynomials $(23,35)_8$, as an example. We search for the optimal power distribution using (20) and (23), respectively, with $Q = 30$ iterations. Fig. 4 shows the simulated lowest power-user’s performance using the power levels obtained by exhaustive searching for both methods. It is seen that the result using the power profile optimized based on (20) is better by about 5 dB than that based on (23). Clearly, for such a system with a small $K$, the approximation in (23) results in considerable performance degradation. However, we expect that the problem will become less serious when $K$ increases. The approximation in (23) should become more accurate when $K$ is large (since then $P(m)f(\gamma(m)(\infty))$ should become negligible compared with $I(q)$). For a large $K$, exhaustive searching is not practical. Then, the linear programming technique becomes a good compromise, as shown by the following examples.

**C. Linear Programming Examples**

Although the linear programming method is suboptimal, the results are reasonably good when $K$ is large. We will show this using several examples. In particular, we will show that the linear programming results are quite close to the capacity limits (see Fig. 10).

We use two system models to examine the linear programming method. One is based on a rate-1/16 repetition code. The other is based on a rate-1/2 convolutional code with generator polynomials $(23,35)_8$ followed by a rate-1/8 repetition code. The corresponding $f(\cdot)$ and $g(\cdot)$ functions have been shown in Fig. 2. With quadrature phase-shift keying (QPSK) modulation, the overall single-user rate for both systems is 1/8 bits per chip. Note that the repetition coded system is equivalent to an uncoded CDMA, except for the chip-level interleavers and the all-one spreading sequences for all users.

The optimization target is to minimize the required $E_b/N_0$ to achieve $\text{BER} \leq 10^{-4}$. Two cases of 16 and 48 users are
TABLE I
POWER PROFILES FOR FIG. 5

<table>
<thead>
<tr>
<th>Power Level (dB)</th>
<th>User Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Repetition code, K=16</td>
<td>5 0×16</td>
</tr>
<tr>
<td>Repetition code, K=48</td>
<td>30 0×26, 7.4509×8, 10.3484×8, 10.7623×6</td>
</tr>
<tr>
<td>Convolutional code, K=16</td>
<td>30 0×16</td>
</tr>
<tr>
<td>Convolutional code, K=48</td>
<td>80 0×15, 3.3112×3, 4.1391×3, 4.5530×3, 6.6227×2, 7.0366×2, 7.4505×3, 9.5201×1, 9.9341×3, 10.3480×3, 10.7619×1, 13.2455×4, 13.6594×5</td>
</tr>
</tbody>
</table>

Fig. 5. Performance comparison between simulation and evolution. The iteration numbers $Q$ are listed in Table I. Data lengths are 512 for the repetition coded system and 2048 for the convolutionally coded one. Tested. The power profiles optimized by linear programming are listed in Table I. We show in Fig. 5 the performance of the users with the lowest power obtained by both simulation and evolution. (The performance of other users is always better, because they have higher power.) From Fig. 5, we can see that the performance curves obtained by SNR evolution and simulation are in good agreement. Note that the performance difference between the evolution and simulation results is caused by two factors: first the bound in (15) and second the finite block length used. These two factors have opposite effects and so the evolution can only be used as an approximate prediction for simulation. It is also interesting to see that the relative performance difference between the repetition coded and convolutionally coded systems reduces when the number of users (K) increases.

D. Impact of Interleaver Length

The interleaver length mainly affects the independence assumption for the SNR evolution in Section III. With shorter interleaver lengths, the ESE and DEC outputs become more correlated as the iterative decoding proceeds. Fig. 6 shows the BER performance obtained by simulation with different interleaver lengths and K values for the repetition coded IDMA. The same power profiles shown in Table II are used for different data lengths. We can see that the performance improves with increased data lengths, as expected. We can also see that a data length of 512 is sufficient to achieve relative good performance. Performance enhancement is marginal with information block longer than 512.

E. Convergence Speed

As discussed in Section IV-B and [24], parameter $\delta$ can be used to control the tradeoff between convergence speed and the sum received power. For illustration, we select two different $\delta$ values (0.2 and 0.02) to optimize a 48-user repetition coded system. The power profiles for these two cases are listed in Table III and the BER performance obtained by SNR evolution and simulation with 10 and 30 iterations are shown in Fig. 7. It can be seen from this figure that a larger $\delta$ (= 0.2) leads to faster convergence speed during the first few iterations. However, if more iterations are used, a smaller $\delta$ (= 0.02) leads to better performance.

In Fig. 8, we illustrate the impact of the number of iterations required for different loading situations for the repetition coded system. The power profiles are listed in Table IV with optimized $\delta$ for each case (i.e., for each user number K and iteration number Q, we select a $\delta$ value leading to the optimized performance). We observe that if K ≤ 32, fewer than 20 iterations are sufficient (measured at 10⁻⁴) and more iterations do not bring about significant performance improvement. However, when K ≥ 48, more iterations can be beneficial.

F. Impact of FEC Codes

We have also examined the system in Fig. 1 with some more sophisticated FEC codes listed below.

Fig. 8. The impact of the number of iterations (Q). The data lengths for all simulation are 512.

TABLE IV
POWER PROFILES FOR FIG. 8

<table>
<thead>
<tr>
<th>Q</th>
<th>Power level (dB) × (user number)</th>
</tr>
</thead>
<tbody>
<tr>
<td>K=16</td>
<td>5 × 16</td>
</tr>
<tr>
<td>K=32</td>
<td>15, 20, 30 × 25, 5.3811×7</td>
</tr>
<tr>
<td>K=48</td>
<td>20 × 26, 7.4509×8, 10.3484×8, 10.7623×6</td>
</tr>
<tr>
<td>K=64</td>
<td>50 × 27, 7.8645×8, 9.9341×4, 10.3480×9</td>
</tr>
<tr>
<td>K=64</td>
<td>20 × 24, 7.8646×6, 8.2785×8, 13.6596×8, 14.0735×4, 19.4545×14</td>
</tr>
<tr>
<td>K=64</td>
<td>30 × 25, 7.8645×7, 8.2784×7, 13.2455×5, 13.6594×7, 18.6266×13</td>
</tr>
<tr>
<td>K=64</td>
<td>50 × 26, 7.8648×6, 8.2787×9, 12.4179×7, 16.9711×13, 17.3851×3</td>
</tr>
</tbody>
</table>

- A standard rate-1/3 turbo code [22] with generator polynomials (23, 35) in serial concatenation with a length-6 repetition code (overall rate = 1/18).

The rate-1/16 repetition and convolutionally/repetition coded systems considered earlier are also included for comparison. The performance of the repetition and convolutionally coded systems is less sensitive to data length, as shown in Fig. 6. However, the performance of turbo and turbo-Hadamard coded systems is quite sensitive to data length. This is because the coding gains of turbo-type codes are closely related to the data length involved. Therefore, the information lengths are selected as 4095 and 4096, respectively, for turbo and turbo-Hadamard coded systems, and 2048 for both repetition and convolutionally coded systems. The f(·) functions for these codes are shown in Fig. 9. The corresponding performance curves based on SNR evolution in terms of the achieved rate versus average E_b/N_0 (measured at BER ≤ 10^-3) are plotted in Fig. 10. We have also carried out simulation to confirm the SNR prediction in Fig. 10. Some of the simulation results can be found in [11] and [12].

We can see that in the low E_b/N_0 regime, FEC coding is essential. In particular, the use of the turbo-Hadamard code can lead to performance close to the channel capacity [12]. However, in the high E_b/N_0 regime, the benefit of sophisticated FEC coding becomes less impressive (similar observation is also found in [25] and [26]). For example, at E_b/N_0 = 30 dB, even repetition coded system can achieves a rate of 11.5 bits per chip, as compared with the capacity of 13.7 bits per chip. The relative difference is (13.7 − 11.5)/13.7 = 16.1%. Therefore, in this regime, repetition codes may be sufficient.

(Note: The performance difference between capacity and the repetition coded IDMA is significant in terms of E_b/N_0 (e.g., 9 dB at rate = 1 bit per chip). However, for a fixed E_b/N_0 > 20 dB, the relative difference is not significant in terms of achievable rate (≤20%), and it diminishes with increased rate according to our numerical results.)

When normalized to each user, the complexity of the receiver in Fig. 1 is independent of the number of users K (see [19]) and the overall receiver complexity is O(K). In particular, if repetition codes are employed, the complexity is very low as
shown in (7). Based on the discussion above, we can adopt the following strategy.

- When $K$ is small, use a powerful code (such as a turbo or turbo-Hadamard code).
- When $K$ is large, switch to a simple repetition code.

In this way, we can achieve a good tradeoff between performance and the overall receiver complexity, since at high rate regimes, the relative rate loss due to the use of a repetition code is less crucial.

V. PERFORMANCE OPTIMIZATION BASED ON TRANSMITTED POWER ALLOCATION

A. Transmitted Power Allocation

Let $g_k$ denote the channel coefficient and $a_k$ the amplitude of the transmitted signal for user-$k$. The received signal power is given by $|h_k|^2 = |g_k|^2 |a_k|^2$. In practice, minimizing $\sum_k a_k^2 = \sum_k |h_k|^2 / |g_k|^2$ may be of more interest since it is directly related to power consumption, as well as interference to other systems (or cells). The problem can be formulated as follows.

The transmitted power optimization problem: Find the distribution $\{a_k^2\}$ that minimizes $\sum_k a_k^2$, subject to $\gamma_k^{(2)} \geq \Gamma, \forall k$, where $\{\gamma_k^{(2)}\}$ are obtained through the following SNR evolution process (with initialization $f(\gamma_k^{(0)}) = 1, \forall k$):

$$\gamma_k^{(q)} = \frac{|g_k|^2 |a_k^2 |}{\sum_{k', k} |g_{k'}|^2 |a_{k'}|^2 f(\gamma_{k'}^{(q-1)}) + \sigma^2},$$

for $k = 1 : K$ and $q = 1 : Q_{\text{max}}$. (27)

For a small $K$, the above problem can again be solved by exhaustive searching. However, when $K$ is large, exhaustive searching becomes impractical. We now outline an alternative solution. First, a definition. For a set of indexed values $\eta = \{|h_k|^2\}$, we will say that $\eta^* = \{|h_k^*|^2\}$ is a permutation of $\eta$ if there is a one-to-one identity mapping between the elements of $\eta$ and $\eta^*$. For example, let $\eta = \{|h_1|^2, |h_2|^2\}$ and $\eta^* = \{|h_2|^2, |h_1|^2\}$. Then, $\eta^*$ is a permutation of $\eta$ if $|h_1^2|^2 = |h_2|^2$ and $|h_2^2|^2 = |h_1|^2$. The following two-step method provides a simple, suboptimal approach to the transmitted power allocation problem.

Step 1) Find $\eta = \{|h_k|^2\}$ using the received power allocation technique discussed in Section IV.

Step 2) Find a permutation $\eta^*$ of $\eta$ that minimizes $\sum_k |h_k^2|^2 / |g_k|^2$ and assign $|h_k^2|^2$ to user-$k$ (i.e., the transmitted power of user-$k$ is set to $a_k^2 = |h_k^2|^2 / |g_k|^2$).

Without loss of generality, we assume that $0 < |g_k|^2 \leq |g_2|^2 \leq \ldots \leq |g_K|^2$. Then, from the rearrangement inequality [27], it can be proved that the optimal solution to Step 2) is the ordered set $\eta^*$ with $|h_1^2|^2 \leq |h_2^2|^2 \leq \ldots \leq |h_K^2|^2$, i.e.,

$$\sum_k |h_k^2|^2 / |g_k|^2 \leq \sum_k |h_k^2|^2 / |g_k|^2,$$

for any given $\{|h_k|^2\}$ and $\{|g_k|^2\}$. In other words, the above strategy assigns a lower received power level to a user with a lower channel gain. We refer to this strategy as ordered matching.

Comments.

- The above two-step approach is optimal if the FEC coding involved is ideal (i.e., capacity achieving) and $\{|h_k|^2\}$ are obtained based on successive stripping decoding. See [28] and [29] for detailed discussion.
- For practical codes, the above two-step approach is sub-optimal. (It is easy to construct an example with two users to verify the suboptimality.) However, as shown below in Fig. 11, the outcome of the two-step approach should be quite close to the optimal solution of (27).

B. Simulation Examples

We consider a $K$-user IDMA system in quasi-static fading environment. The channel coefficients are modeled as $|g_k|^2 = A \xi_k^{\nu} \gamma_k^{10/\nu} / \gamma_k$, where $A$ is a constant, $\xi_k (0 \leq \rho_k \leq 1)$ the normalized distance between user-$k$ and the receiver, $\nu$ the path-loss exponent, $\xi_k$ a zero-mean Gaussian random variable with variance $\sigma^2_{\xi}$ that characterizes shadow fading, and $\gamma_k$ the Rayleigh-fading gain with unit mean. For simplicity, we will set $\nu = 1$, so the power levels discussed below are relative values. We set $\nu = 4$, $\sigma_{\xi} = 8$, $K = 64$ and assume that all users are uniformly located in a circular cell area.

Since the required transmitted powers may be very large for the users with deep fading, we allow transmission outage for the users with channel gains below a given fading threshold $G_0$. That is, if $|g_k|^2 < G_0$, then an outage is declared for user-$k$ and $a_k^2$ is set to zero (a similar strategy has been adopted in [30]). The above two-step strategy is then applied to the active users. We define the outage probability $P_{\text{out}}(G_0) = \Pr(|g_k|^2 < G_0), \forall k$. The target BER for the active users is set to $10^{-4}$.

Fig. 11 shows the required average transmitted power $\sum_k a_k^2$ versus $P_{\text{out}}(G_0)$ with the two-step power allocation strategy for the repetition and convolutional codes considered in Section IV-C and an ideal code (all with rate 1/16). We also compare the results with a random matching strategy in which $\eta^*$ is a randomly generated permutation of $\eta$. It can be observed
that ordered matching requires significantly less average transmitted power than random matching. For $P_{out}(G_0) = 10^{-2}$, ordered matching can provide more than 10-dB power gain over random matching. Furthermore, with ordered matching, the gap between the convolutional coding scheme and the ideal coding scheme is only about 4 dB. Recall that there is an inherent performance gap between the convolutional code and the ideal code (about 4 dB at the BER $= 10^{-4}$ in an AWGN channel), we may expect that the performance loss due to the two-step approach should be marginal.

VI. CONCLUSION

In this paper, we have shown that the SNR evolution approach provides a fast and relatively accurate method to assess the performance of the iterative CBC detection algorithm for CDMA systems with chip-level interleavers. Using this approach as a searching tool, we have developed a linear programming technique to optimize the system performance. Both analytical and simulation results show that the system throughput is power limited, not interference limited as in conventional CDMA systems with single-user detection and equal power control. We have shown that for high rate systems, even the repetition code is sufficient to achieve throughput reasonably close to the theoretical limit. Besides multiple access systems, the proposed method can also be applied to other systems, such as adaptive transmission systems [31].

ACKNOWLEDGMENT

The authors are grateful to Dr. H. D. Pfister for enlightening discussions and for help in developing the linear programming software used in this work.

REFERENCES

Jun Tong received the B.E. degree in information engineering and the M.E. degree in signal and information processing from the University of Electronic Science and Technology of China, Chengdu, in 2001 and 2004, respectively. He is currently working towards the Ph.D. degree at the City University of Hong Kong. His research interests are antenna arrays and signal detection in communication systems.

Li Ping (S’87–M’91) received the Ph.D. degree at Glasgow University, Glasgow, U.K., in 1990. He lectured at the Department of Electronic Engineering, Melbourne University, from 1990 to 1992, and worked as a Member of Research Staff at the Telecom Australia Research Laboratories from 1993 to 1995. He has been with the Department of Electronic Engineering, City University of Hong Kong since January 1996, where he is now a Professor. His research interests are mixed analog/digital circuits, communications systems, and coding theory.

Dr. Li Ping was awarded a British Telecom-Royal Society Fellowship in 1986, the IEE J. J. Thomson Premium in 1993, and a Croucher Senior Research Fellowship in 2005.