



Adaptive parallel interference cancellation for CDMA systems—A weight selection and filtering scheme

Y.-T. Hsieh^{a,*}, W.-R. Wu^b

^a Information and Communications Research Laboratories, Industrial Technology Research Institute, Rm. 214, Bldg. 11, 195, Sec. 4, Chung Hsing Rd., Chutung, Hsinchu 310, Taiwan, ROC

^b Department of Electrical Engineering, National Chiao Tung University, Hsinchu, Taiwan, ROC

ARTICLE INFO

Article history:

Received 1 July 2009

Received in revised form

24 February 2010

Accepted 1 June 2010

Available online 8 June 2010

Keywords:

Multiuser detection

Parallel interference cancellation

LMS algorithm

Performance analysis

ABSTRACT

Parallel interference cancellation (PIC) is a well-known multiuser detection algorithm in direct-sequence code-division multiple-access (DS-CDMA) systems. It is typically implemented with a multi-stage architecture. One problem associated with the PIC is that unreliable interference cancellation may occur in the early stages and the system performance may be degraded. Thus, the partial PIC detector was developed to control the cancellation level by use of interference cancellation factors. Partial PIC can be implemented with an adaptive form, in which optimal weights are derived using the least mean square (LMS) algorithm. In this paper, we propose an algorithm improving the conventional adaptive partial PIC. The main idea is to reduce the number of active weights in the LMS algorithm, and to perform weight post-filtering such that the resultant excess mean square error can be reduced. We also analyze the performance of the proposed algorithm and derive the bit error rate of the second stage output. Simulation results verify that the proposed algorithm outperforms the conventional partial PIC, and derived analytical results are accurate.

© 2010 Elsevier B.V. All rights reserved.

1. Introduction

In direct-sequence code division multiple access (CDMA) systems, multiple access interference (MAI) is regarded as the main source limiting the system capacity. Multiuser detection (MUD) is a well-known technique dealing with MAI. Different from the architecture of conventional single-user receivers, MUD conducts detections for all users simultaneously and can achieve much better performance. In [1] a maximum-likelihood multiuser receiver was first proposed. Although significant performance enhancement can be obtained, the required computational complexity is very high, growing exponentially with the user number. This adversely affects its real-

world applications. As a consequence, many suboptimum alternatives were then proposed [2–4].

The subtractive-type interference cancellation is a well-known MUD algorithm. As far as the desired user is concerned, the interference is estimated from the received signal, regenerated, and cancelled with the interference canceller. The canceller is usually implemented with multiple stages to achieve its optimum performance. This type of MUD can be classified into two categories, i.e., successive interference cancellation (SIC) and parallel interference cancellation (PIC). SIC cancels interference from other users sequentially [5,6], while PIC does it all at one time [7,8]. SIC usually conducts signal power ranking to determine the cancellation order. A stronger user often has lower probability of decision errors, and cancellation of this signal gives more reliable result than that of a weaker user. Thus, we can expect that SIC works better when users have unbalanced powers. However, SIC requires extra computation for power ranking and introduces larger delay. By contrast, PIC is more effective

* Corresponding author. Tel.: +886 3 5914854; fax: +886 3 5829733.
E-mail addresses: ythsieh@itri.org.tw (Y.-T. Hsieh),
wrwu@faculty.nctu.edu.tw (W.-R. Wu).

when user powers are similar and does not need to perform the power ranking procedure.

Although the PIC approach is able to cancel interference from other users simultaneously, its interference estimation may not be reliable in early stages. Unreliable interference cancellation will increase interference and degrade the detector's performance. Partial PICs were developed to remedy this problem [9]. In partial PIC, cancellation factors (ranging from zero to unity) are introduced to control the cancellation level. Since the reliability of cancellation increases stage by stage, larger factors can be used in later stages. Optimum cancellation factors can be derived in adaptive or non-adaptive ways. The result of the non-adaptive approach was reported in [10–15], while that of the adaptive approach in [16–23]. The advantage of adaptive detectors is that it can dynamically adjust cancellation factors to accommodate channel variation [16]. Two well known adaptive algorithms have been used in PIC detectors, i.e., the least mean square (LMS) [17–19] and the recursive least squares (RLS) [19,20]. The LMS algorithm enjoys its lower complexity at the expense of slower convergence. The adaptive filters employed in [19,20] require training sequences, while those in [17,18] do not. Besides, the adaptive weights can be updated with the chip or bit rates. In [17–19], the weights are updated with the chip rate and in [20], they are updated with the bit rate. The structure of the proposed adaptive partial PIC detector is similar to that in [17], which does not require training sequences and the weights are updated with the chip rate. This distinct feature enables the adaptive partial PIC detector to converge fast and work well under fast-fading environments. Conventional adaptive receivers that do not use training sequences only require the spreading code of the desired user [21–23]. In contrast, the adaptive partial PIC receiver we consider requires the spreading codes of all users, and it is more suitable for uplink scenarios. Application of this adaptive partial PIC in multirate systems was reported in [24].

It is found that the performance of the adaptive partial PIC in [17] can be further improved. In [25], hard-decision devices are added at the outputs of adapted weights, resembling the full PIC operation. However, errors due to incorrectly decided weights may limit the cancellation performance. In this paper, we propose a new algorithm for the performance enhancement of a multi-stage adaptive partial PIC. The proposed algorithm is composed of two procedures. The first procedure is called weight pre-selection, and the second is weight post-filtering. With these procedures, the resultant excess mean square error (MSE) due to the adaptive algorithm can be reduced. As a result, the performance of the canceller in [17] can be improved. Note that the optimal results for the two procedures in the proposed algorithm are closely related. They have to be jointly considered to attain the best performance. We also conduct performance analysis for the proposed adaptive partial PIC detector and derive its bit error rate (BER). The analysis is based on the method in our previous work [26] in which the performance of the original adaptive partial PIC [17] is analyzed.

The remainder of the paper is organized as follows. Section 2 first describes the conventional non-adaptive and adaptive partial PIC receivers. In Section 3, we then detail the proposed algorithm. In Section 4, we analyze the weight behavior and the output BER of a two-stage adaptive partial PIC with the proposed algorithm. Finally, we report the simulation results in Section 5. Conclusions are given in Section 6.

2. System model

Consider a synchronous CDMA system operated in an AWGN channel. The received signal in a certain bit interval can be expressed as

$$r(n) = \sum_{k=1}^K a_k b_k x_k(n) + v(n), \quad 0 \leq n < N, \quad (1)$$

where a_k , b_k and $x_k(n)$ are the k th user's amplitude, data bit, and signature sequence, respectively, and $v(n)$ is AWGN with variance σ^2 . Let the processing gain be N and the signature sequence be formed by binary chips with amplitude $1/\sqrt{N}$. The matched filter output, which is the first stage output, can then be expressed as

$$y_k^{(1)} = \sum_{n=0}^{N-1} r(n)x_k(n) = a_k b_k + \sum_{j \neq k} a_j b_j \rho_{jk} + \gamma_k, \quad (2)$$

where $\rho_{jk} = \sum_{n=0}^{N-1} x_j(n)x_k(n)$ denotes the signature correlation between user j and k , and $\gamma_k = \sum_{n=0}^{N-1} v(n)x_k(n)$ the noise term after despreading. From (2), we can see that the output signal contains MAI. The operation of a non-adaptive partial PIC can be described as [9]

$$y_k^{(i)} = c_k^{(i)} \left(y_k^{(1)} - \sum_{j \neq k} a_j \hat{b}_j^{(i-1)} \rho_{jk} \right) + (1 - c_k^{(i)}) y_k^{(i-1)}, \quad (3)$$

where $y_k^{(i)}$ and $c_k^{(i)}$ are the soft-output and the cancellation factor for the k th user in the i th stage, respectively. The hard-decision output for the i th stage is then $\hat{b}_k^{(i)} = \text{sgn}\{y_k^{(i)}\}$. The soft-output in (3) can be regarded as a weighted sum of two estimates; one is the full PIC output in the current stage multiplied by the cancellation factor $c_k^{(i)}$, while the other is the weighted soft output estimate from the previous stage.

The partial PIC can be also obtained with an adaptive structure as depicted in Fig. 1. Define an error signal as

$$e^{(i)}(n) = r(n) - \tilde{r}^{(i)}(n), \quad i > 1, \quad (4)$$

where $\tilde{r}^{(i)}(n)$ is the regenerated received signal as expressed by

$$\tilde{r}^{(i)}(n) = \sum_{k=1}^K w_k^{(i)}(n) \hat{b}_k^{(i-1)} x_k(n). \quad (5)$$

Here, $w_k^{(i)}(n)$ is the adapted weight for the k th user in the i th stage. After convergence, $w_k^{(i)}(N)$ is seen as the desired cancellation factor. Define a mean square error (MSE) as

$$J^{(i)}(n) = E[(r(n) - \tilde{r}^{(i)}(n))^2]. \quad (6)$$

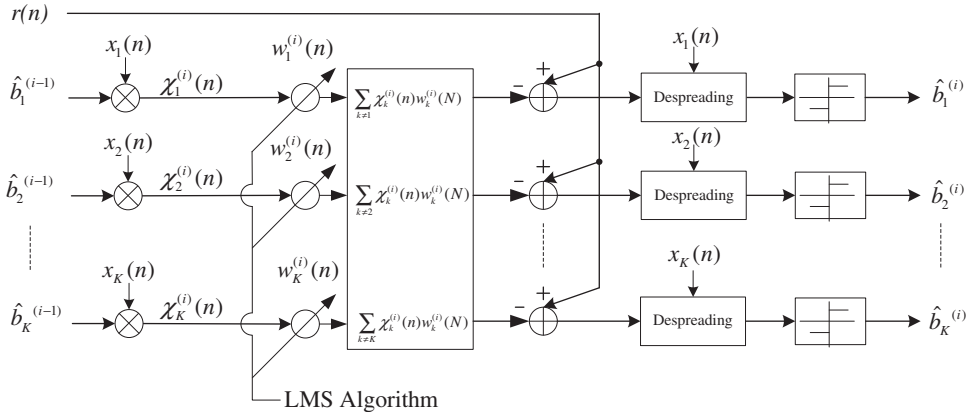


Fig. 1. Block diagram of adaptive partial PIC receivers.

Using the steepest decent algorithm, we can obtain the weight update equation as

$$w_k^{(i)}(n+1) = w_k^{(i)}(n) + \mu_k^{(i)} \chi_k^{(i)}(n) e^{(i)}(n), \quad 0 \leq n < N, \quad (7)$$

where $\mu_k^{(i)}$ is the step size for the k th user in the i th stage. Here, the input signal is

$$\chi_k^{(i)}(n) = \hat{b}_k^{(i-1)} x_k(n). \quad (8)$$

The algorithm in (7) is called the LMS algorithm. The interference-subtracted signal for the k th user is then

$$\hat{r}_k^{(i)}(n) = r(n) - \sum_{j \neq k} \chi_j^{(i)}(n) w_j^{(i)}(N). \quad (9)$$

We then have the matched filter output as

$$y_k^{(i)} = \sum_{n=0}^{N-1} \hat{r}_k^{(i)}(n) x_k(n). \quad (10)$$

Note that the optimization criteria for these two types of partial PICs expressed in (3) and (10) are different. In the non-adaptive type partial PIC, the optimal factor, $c_k^{(i)}$, is determined based on the minimization of the ensemble error averaged over all transmission bits. In other words, optimal weights apply to all received bit signals. On the other hand, the optimal weight for the adaptive partial PIC, $w_k^{(i)}(n)$, is obtained by minimizing the ensemble error averaged over a certain bit interval (given the bit decision in the previous stage). The LMS algorithm is re-initiated at the beginning of each bit period. The input signals in (8) take on different bit decision values for different stages. The signature sequence is also changed bit-by-bit when the long code is used. As a result, the optimal weights change for each bit duration.

We then extend the signal model to multipath channels. Denote the transfer function of the channel impulse response for the k th user as

$$\mathcal{W}_k(z) = \sum_{l=0}^{L-1} h_{k,l} z^{-\tau_{k,l}}, \quad (11)$$

where $h_{k,l}$ and $\tau_{k,l}$ are the gain and delay values for the l th path, respectively, and L is the number of paths. In the receiving end, we can use the maximal ratio combining (MRC) to demodulate the signal. Let the equivalent

baseband received signal be expressed by

$$r(n) = \sum_{l=0}^{L-1} \sum_{k=1}^K b_k a_k h_{k,l} x_k(n - \tau_{k,l}). \quad (12)$$

The first stage output signal is given by

$$y_k^{(1)} = \sum_{l=0}^{L-1} y_{k,l}^{(1)} h_{k,l}, \quad (13)$$

where the branch output from the MRC can be formed as

$$y_{k,l}^{(1)} = \sum_{n=0}^{N-1} r(n) x_k(n - \tau_{k,l}). \quad (14)$$

Following the signal model for the AWGN channel, we can obtain the regenerated received signal as

$$\hat{r}^{(i)}(n) = \sum_{l=0}^{L-1} \sum_{k=1}^K \chi_k^{(i)}(n - \tau_{k,l}) w_{k,l}^{(i)}(n), \quad (15)$$

where $w_{k,l}^{(i)}(n)$ denotes the weight for the l th path of the k th user in the i th stage. We can then formulate the error signal as that in (4), and have a counterpart of $\hat{r}_k^{(i)}(n)$ in (5) as

$$\hat{r}_k^{(i)}(n) = r(n) - \sum_{l=0}^{L-1} \sum_{j \neq k} \chi_j^{(i)}(n - \tau_{j,l}) w_{j,l}^{(i)}(N). \quad (16)$$

The i th-stage matched output using the MRC is then

$$y_k^{(i)} = \sum_{l=0}^{L-1} \sum_{n=0}^{N-1} \hat{r}_k^{(i)}(n) x_k(n - \tau_{k,l}) h_{k,l}. \quad (17)$$

3. Proposed algorithm

It can be seen from (6) that in the ideal condition (without noise), $\hat{r}^{(i)}(n) = r(n)$. In this case, the weights are obtained from (1) and (5) as

$$w_k^{(i)}(n) = \begin{cases} a_k, & \hat{b}_k^{(i)} = b_k, \\ -a_k, & \hat{b}_k^{(i)} \neq b_k. \end{cases} \quad (18)$$

It is found that the ideal weights are determined by the bit decision results. Note that the adaptation period is constrained in one bit period since the ideal weight may

be $+a_k$ or $-a_k$ for each bit. Thus the weight of each user tends to attain the desired value bit by bit. This is also the reason why the adaptive approach performs better than non-adaptive methods. However, although the LMS algorithm has the complexity advantage, its slow convergence may not lead the weights to the desired values in such a short period. In addition, the adapted weights are closely related to the parameters used in the LMS algorithm. Thus, when considering to improve the performance of the LMS algorithm, we have to take several factors into account such as the number of weights, the step size, the number of training data, noise variance, and the weight initials, etc. These factors may interact one another and complicates the adaptation procedure. In this paper, we will mainly focus on the first two factors, i.e., the weight numbers and the step size to obtain improved performance. We propose an algorithm that can reduce the number of adapted weight, leading to a smaller excess MSE (induced by the LMS algorithm). The algorithm also allows a larger step size, accelerating the convergence.

3.1. Weight pre-selection procedure

As mentioned, the MSE of the adaptive partial PIC is proportional to the number of weights adapted in the LMS algorithm. One way to improve the system performance is to reduce the number of weights updated in the LMS algorithm. Here, we propose an algorithm to do the job. The idea of the algorithm is described as follows. If the magnitude of the matched output for a user exceeds a predefined threshold, the corresponding decided bit is deemed reliable, and the weight corresponding to this bit is deactivated. In other words, this weight will not be included in the training process and it is set as the channel gain immediately. This algorithm can be easily expressed using a two step-size scenario described below:

$$\mu_k^{(i)} = \begin{cases} 0 & \text{if } |y_k^{(i-1)}| > a_k \zeta_s^{(i)}, \\ \mu^{(i)} & \text{if } |y_k^{(i-1)}| \leq a_k \zeta_s^{(i)}, \end{cases} \quad (19)$$

where $\zeta_s^{(i)}$ denotes the normalized decision threshold. The step-size decision function, denoted as $\Lambda_S(\cdot)$, is shown in Fig. 2(a). Note that it is possible that some weights are erroneously decided. If this does happen, it will increase the noise variance in the LMS algorithm. Thus, the threshold $\zeta_s^{(i)}$ has to be determined carefully.

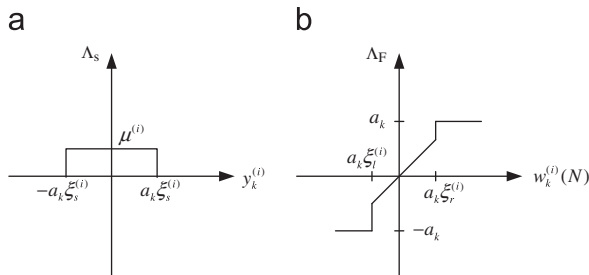


Fig. 2. Functions used in the proposed algorithm: (a) weight pre-selection function; (b) weight post-filtering function.

3.2. Weight post-filtering procedure

It is well known that the convergent weights in the LMS algorithm are random. We can model the convergent weights as optimum weights plus noise. Thus, if we know the weight distribution, we can perform weight post-filtering (estimation). This will enhance the partial PIC performance furthermore. Fig. 3 shows a typical probability density function for the adapted weights. As we can see, the magnitudes of some weights are greater than the corresponding channel gains. However, these weights are not reasonable since a normal weight magnitude always falls between $\pm a_k$ to reflect the cancellation reliability. Thus, the performance can then be enhanced if the mis-adapted weights can be further filtered. Note that given a binary random variable embedded in AWGN, the MMSE estimate corresponds to a transformation with a hyperbolic tangent function. We can then apply the result here, and filter the convergent weights with the hyperbolic tangent function. Since the function is highly nonlinear, the performance analysis is difficult. We then use a piecewise linear function, denoted as $\Lambda_F(\cdot)$, instead. The function is shown in Fig. 2(b). Note that this function has two thresholds, denoted as $\{\zeta_l^{(i)}, \zeta_r^{(i)}\}$. If a weight is greater than the right-hand side threshold $a_k \zeta_r^{(i)}$, it is mapped to a_k . Similarly, if a trained weight is less than the left-hand side threshold $a_k \zeta_l^{(i)}$, it is mapped to $-a_k$. The intermediate values between the thresholds would be kept unchanged.

As seen from Fig. 3, the weight distribution has different mean values for correct/erroneous decision outputs (in the previous stage). Normally, the weight initials for both correct and erroneous decisions are set as the channel gain a_k , and it takes more adaptation steps for weights with erroneous decisions to attain the ideal values around $-a_k$. In other words, the mean value of the adapted weights for erroneous decision bits will be closer to $-a_k$ if N is larger. However, in a practical system, N is usually not large enough. Thus, we have to use a large step size $\mu^{(i)}$ to speed up the convergence for users with

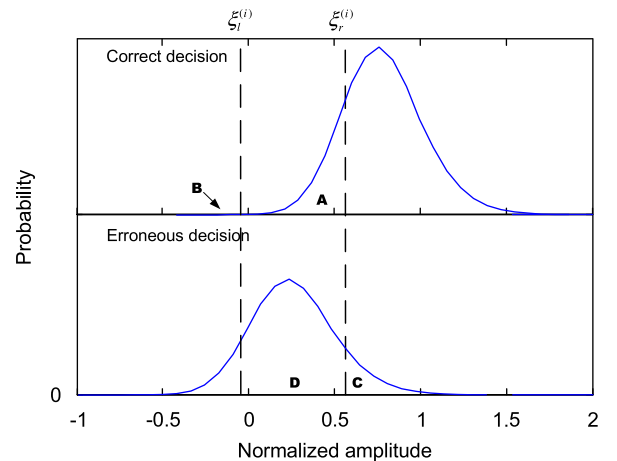


Fig. 3. Probability density function of adapted weights from the LMS algorithm with processing gain $N=31$.

erroneous decisions. However, a larger step size will enlarge the weight variance which adversely affect the final performance. Thus, the choice of the step size is critical. The two procedures proposed above can reduce the number of active weights and further filter the adapted weights. As a result, it is possible to use a larger step size without significantly increasing the weight variance. Apparently, the parameters used in the proposed algorithm are coupled one another, and their optimal values cannot be obtained individually. With some trial-and-errors, we can find a good compromise among parameters $\{\mu^{(i)}, \zeta_s^{(i)}, \zeta_l^{(i)}, \zeta_r^{(i)}\}$ such that near optimum performance can be achieved.

4. Performance analysis for a two-stage detector

The LMS algorithm has been analyzed and developed for over four decades. However, most results cannot be used here. This is because the step size used in this application is large and many assumptions required by the conventional analysis will be violated. The other reason is that we are most concerned about the transient behavior (due to small training period within one bit) while most works are only concerned about the steady-state behavior. In [26], we have derived optimum weights, weight error means, and weight error variances in the second stage for a two-stage adaptive partial PIC receiver shown in Fig. 1. Here, we extend the results to derive the bit error rate (BER) of the proposed algorithm, i.e., the receiver in Fig. 1 with the additional operations described in the preceding section. The first part of this section serves as an excerpt of the derivation in [26] where only important steps of the derivation will be highlighted. Interested readers can refer to [26] for more details.

4.1. Analysis of conventional algorithm

In [26], the single-user case was considered first. The exact solution of optimal weights, weight error means, and weight error variances for correct and erroneous decisions (of the first stage) were derived. For the two-user scenario, the optimal weights and weight error means were derived exactly, while the weight error variance were approximated from that in the single-user scenario. When the analysis is generalized to the multiple-user case (i.e., $K > 2$), all the analytical results are approximated from a simplified two-user model. Assume that the first user is the desired user. We can then rewrite the K -user model in (2) as

$$y_1^{(1)} \simeq a_1 b_1 + a_l b_l \rho + \gamma_1, \quad (20)$$

where we assume that $a_l = \sqrt{\sum_{j \neq 1} a_j^2}$ and $b_l \in \{\pm 1\}$. Here, a_l represents the equivalent amplitude of a virtual user, and ρ the equivalent correlation of the desired and the virtual user. Also note that b_l is virtual and we do not need its actual value for multiple-user cases in derivation. For simplicity the superscript ($i=2$) on the adapted weights are omitted throughout this section.

4.1.1. Optimal weight analysis

The Wiener solution for optimal weights can be represented by $\mathbf{w}_{opt} = \mathbf{Q}^{-1} \mathbf{p}$ where the correlation matrix of input signals is expressed by $\mathbf{Q} \triangleq E\{\boldsymbol{\chi}^{(2)}(n) \boldsymbol{\chi}^{(2)}(n)^T\}$ with $\boldsymbol{\chi}^{(2)}(n) = [\chi_1^{(2)}(n), \chi_l^{(2)}(n)]^T$. The crosscorrelation vector is given by $\mathbf{p} \triangleq E\{\boldsymbol{\chi}^{(2)}(n) r(n)\}$. The joint probability density function for the random vector $\boldsymbol{\gamma} = [\gamma_1, \gamma_l]$ is jointly Gaussian and

$$f(\boldsymbol{\gamma}) = \frac{1}{2\pi |\mathbf{C}_\gamma|^{1/2}} \exp\left\{-\frac{1}{2} \boldsymbol{\gamma}^T \mathbf{C}_\gamma^{-1} \boldsymbol{\gamma}\right\}, \quad (21)$$

where the covariance matrix is given as

$$\mathbf{C}_\gamma \triangleq E\{\boldsymbol{\gamma} \boldsymbol{\gamma}^T\} = \begin{bmatrix} \sigma^2 & \rho \sigma^2 \\ \rho \sigma^2 & \sigma^2 \end{bmatrix}. \quad (22)$$

Note that \mathbf{Q} and \mathbf{p} are functions of $\hat{b}_1^{(1)}$ and $\hat{b}_l^{(1)}$, while the both bit decision outputs are functions of $\boldsymbol{\gamma}$ and ρ . Further, we can see that $\boldsymbol{\gamma}$ is also dependent on ρ . Our objective is to derive the optimal weights with closed-form expressions for first stage correct and erroneous decision outputs under AWGN, and the final result may appear differently from the noise-free case in (18).

The first step of the analysis work is to obtain the conditional optimal weights given fixed $\boldsymbol{\gamma}$ and ρ . Then we remove the conditions by nested expectation operations. We denote the optimal weight vectors for the first stage correct and erroneous decisions by $\mathbf{w}_{opt}^c = [w_{opt,1}^c, w_{opt,l}^c]^T$ and $\mathbf{w}_{opt}^e = [w_{opt,1}^e, w_{opt,l}^e]^T$, respectively. In the following, we give the derivations for the first stage correct decision as an example. The derivation for erroneous first-stage decision can be also conducted in a similar way.

- (a) Express the optimal weight given specific ρ and $\boldsymbol{\gamma}$. Denote a decision pattern from a specific ρ and $\boldsymbol{\gamma}$ (represented by $\tilde{\boldsymbol{\gamma}}$) to be $\hat{\mathbf{B}} = \text{diag}\{\hat{b}_1, \hat{b}_l\}$. The Wiener solution given ρ and $\tilde{\boldsymbol{\gamma}}$ is represented by

$$\tilde{\mathbf{w}}_{opt} = \mathbf{A} \hat{\mathbf{B}} \mathbf{b} + \hat{\mathbf{B}} \mathbf{R}^{-1} \tilde{\boldsymbol{\gamma}}, \quad (23)$$

where $\mathbf{A} = \text{diag}\{a_1, a_l\}$, $\mathbf{b} = [b_1, b_l]^T$, and the correlation matrix is given by

$$\mathbf{R} = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}. \quad (24)$$

- (b) Express the optimal weight given a specific ρ and averaged $\boldsymbol{\gamma}$ for different first stage bit decisions.

As we can see from (23), the second-stage optimal weights depend on $\hat{\mathbf{B}}$. There are four decision patterns, i.e., $\{\hat{b}_1^{(1)}, \hat{b}_l^{(1)}\} = \{\pm b_1, \pm b_l\}$. Note that for each decision pattern, we have two bit patterns that $b_1 = b_l$ and $b_1 \neq b_l$. Let \cup^{ij} denote the set of $\tilde{\boldsymbol{\gamma}}$ yielding the i th decision for the j th bit pattern. Then,

$$\tilde{\mathbf{w}}_{opt}^{ij} = \mathbf{A} \hat{\mathbf{B}}^i \mathbf{b}^j + \hat{\mathbf{B}}^i \mathbf{R}^{-1} E_{\boldsymbol{\gamma}}\{\tilde{\boldsymbol{\gamma}}^{ij}\}, \quad (25)$$

where $\hat{\mathbf{B}}^i$ denotes the i th decision pattern, \mathbf{b}^j denotes the j th bit pattern and the noise integration is given by

$$E_{\boldsymbol{\gamma}}\{\tilde{\boldsymbol{\gamma}}^{ij}\} = \frac{\int_{\cup^{ij}} \boldsymbol{\gamma} f(\boldsymbol{\gamma}) d\boldsymbol{\gamma}}{\int_{\cup^{ij}} f(\boldsymbol{\gamma}) d\boldsymbol{\gamma}}. \quad (26)$$

The complete set of \cup^{ij} for all decision and bit patterns is shown in Table 1. The complete set for the conditional optimal weights in (25) is presented in Table 2 (the results for $b_1=1$ are identical to that for $b_1=-1$).

- (c) Express the optimal weight given a specific ρ and averaged γ for first stage correct decision.

We can see in (25) that the optimal weight of one user is coupled to the other user due to \mathbf{R} . Our objective is to derive \mathbf{w}_{opt}^c and \mathbf{w}_{opt}^e for individual users. Thus we have to determine the components of $\check{\mathbf{w}}_{opt}^c = [\check{w}_{opt,1}^c, \check{w}_{opt,2}^c]^T$ user by user. For example, the optimal weight for the first user with correct decision and a given ρ is

$$\check{w}_{opt,1}^c = \frac{1}{P_{C_1}} \sum_{C_1} \check{w}_{opt,1}^{ij} P_{ij}, \quad (27)$$

where $P_{ij} = \int_{\cup^{ij}} f(\gamma) d\gamma$, $C_1 = \cup^{11} \cup \cup^{12} \cup \cup^{21} \cup \cup^{22}$, and $P_{C_1} = P_{11} + P_{12} + P_{21} + P_{22}$.

- (d) Express the optimal weight given averaged ρ and averaged γ for first stage correct decision.

Taking the first user as example, we have

$$w_{opt,1}^c = E_{\rho} \{ \check{w}_{opt,1}^c \} = \frac{\sum_{\rho} \check{w}_{opt,1}^c P_{\rho} P_{C_1}}{\sum_{\rho} P_{\rho} P_{C_1}}, \quad (28)$$

where the distribution for the correlation coefficient is given by a two-user model as

$$P_{\rho} = \frac{1}{2^N} \binom{N}{N(1+\rho)/2}. \quad (29)$$

Table 1

Sets of γ for all decision and bit patterns.

\cup^{ij}	$\hat{\mathbf{B}}^i \mathbf{b}^j$	Range for γ_1	Range for γ_l
\cup^{11}	$\begin{bmatrix} 1 \\ 1 \end{bmatrix}$	$\gamma_1 > -(a_1 + a_l \rho)$	$\gamma_l > -(a_1 \rho + a_l)$
\cup^{12}		$\gamma_1 > -(a_1 - a_l \rho)$	$\gamma_l < -(a_1 \rho - a_l)$
\cup^{21}	$\begin{bmatrix} 1 \\ -1 \end{bmatrix}$	$\gamma_1 > -(a_1 + a_l \rho)$	$\gamma_l < -(a_1 + a_l \rho)$
\cup^{22}		$\gamma_1 > -(a_1 - a_l \rho)$	$\gamma_l > -(a_1 \rho - a_l)$
\cup^{31}	$\begin{bmatrix} -1 \\ -1 \end{bmatrix}$	$\gamma_1 < -(a_1 + a_l \rho)$	$\gamma_l < -(a_1 + a_l \rho)$
\cup^{32}		$\gamma_1 < -(a_1 - a_l \rho)$	$\gamma_l > -(a_1 \rho - a_l)$
\cup^{41}	$\begin{bmatrix} -1 \\ 1 \end{bmatrix}$	$\gamma_1 < -(a_1 + a_l \rho)$	$\gamma_l > -(a_1 + a_l \rho)$
\cup^{42}		$\gamma_1 < -(a_1 - a_l \rho)$	$\gamma_l < -(a_1 \rho - a_l)$

Table 2

Complete list of conditional optimal weights ($a=[1,-1]^T$ and $\mathbf{J}=\text{diag}\{1,-1\}$).

$\check{\mathbf{w}}_{opt}^{11} = \mathbf{a} + \mathbf{R}^{-1} E_{\gamma} \{\tilde{\gamma}^{11}\}$	$\check{\mathbf{w}}_{opt}^{12} = \mathbf{a} + \mathbf{J} \mathbf{R}^{-1} E_{\gamma} \{\tilde{\gamma}^{12}\}$
$\check{\mathbf{w}}_{opt}^{21} = -\mathbf{a} + \mathbf{J} \mathbf{R}^{-1} E_{\gamma} \{\tilde{\gamma}^{21}\}$	$\check{\mathbf{w}}_{opt}^{22} = -\mathbf{a} + \mathbf{R}^{-1} E_{\gamma} \{\tilde{\gamma}^{22}\}$
$\check{\mathbf{w}}_{opt}^{31} = \mathbf{J} \mathbf{a} - \mathbf{J} \mathbf{R}^{-1} E_{\gamma} \{\tilde{\gamma}^{31}\}$	$\check{\mathbf{w}}_{opt}^{32} = \mathbf{J} \mathbf{a} - \mathbf{R}^{-1} E_{\gamma} \{\tilde{\gamma}^{32}\}$
$\check{\mathbf{w}}_{opt}^{41} = -\mathbf{J} \mathbf{a} - \mathbf{J} \mathbf{R}^{-1} E_{\gamma} \{\tilde{\gamma}^{41}\}$	$\check{\mathbf{w}}_{opt}^{42} = -\mathbf{J} \mathbf{a} - \mathbf{R}^{-1} E_{\gamma} \{\tilde{\gamma}^{42}\}$

4.1.2. Weight error mean analysis

Let the adapted weight of the k th user given $\tilde{\gamma}$ and ρ and correct fist-stage decision as $\tilde{w}_k^c(n)$, $k=1, l$. Then the weight error vector for correct decision is expressed by $\tilde{\mathbf{e}}^c(n) = [\tilde{e}_1^c(n), \tilde{e}_l^c(n)]^T$ with $\tilde{e}_k^c(n) = \tilde{w}_k^c(n) - w_{opt,k}^c$. From (27) we see that $\check{\mathbf{w}}_{opt}^c$ is derived from $\check{\mathbf{w}}_{opt}^{ij}$'s. Thus, we also consider the conditional weight errors as

$$\tilde{\mathbf{e}}^{ij}(n) = \check{\mathbf{w}}^{ij}(n) - \check{\mathbf{w}}_{opt}^{ij}, \quad (30)$$

where the conditional weights are defined as

$$\check{\mathbf{w}}^{ij}(n) = \{\mathbf{w}(n) | \gamma \in \cup^{ij}\}. \quad (31)$$

After some algebraic manipulations, we can have the weight error mean vector as

$$\begin{aligned} \check{\mathbf{e}}_M^{ij}(n) &= [\check{e}_{M,1}^{ij}(n), \check{e}_{M,l}^{ij}(n)]^T \doteq E_{\gamma} \{ \tilde{\mathbf{e}}^{ij}(n) \} \\ &= \left(\mathbf{I} - \frac{\mu^{(2)}}{N} \hat{\mathbf{B}} \mathbf{R} \hat{\mathbf{B}}^i \right)^n \check{\mathbf{e}}_M^{ij}(0), \end{aligned} \quad (32)$$

where

$$\check{\mathbf{e}}_M^{ij}(0) = \mathbf{w}(0) - \check{\mathbf{w}}_{opt}^{ij}. \quad (33)$$

The weight error mean for the first user conditioned on only the first stage correct decision and ρ is represented by

$$\check{e}_{M,1}^c(n) \doteq E \{ \check{e}_1^c(n) \} = \frac{1}{P_{C_1}} \sum_{C_1} \check{e}_{M,1}^{ij}(n) P_{ij}. \quad (34)$$

Then, the averaged weight error mean for correct decision for the first user can be obtained by

$$e_{M,1}^c(n) \doteq E \{ \check{e}_1^c(n) \} = E_{\rho} \{ \check{e}_{M,1}^c(n) \} = \frac{\sum_{\rho} \check{e}_{M,1}^c(n) P_{\rho} P_{C_1}}{\sum_{\rho} P_{\rho} P_{C_1}}. \quad (35)$$

4.1.3. Weight error variance analysis

The weight error variance for correct decision is defined as $e_{V,1}^c(n) = E \{ [\check{e}_1^c(n) - e_{M,1}^c(n)]^2 \}$. The exact analysis for the weight error variance is difficult for multiple users. Thus the analytical result in the single-user case is used to approximate that in the multiple-user scenario, which is

$$e_{V,1}^c(n) = e_{V,1}^c(n) + \beta_1^c(n), \quad (36)$$

where $e_{V,1}^c(n)$ is expressed as

$$e_{V,1}^c(n) = \frac{[\mu^{(2)}]^2}{N^2} \left\{ N \sigma^2 \left(\frac{1 - \alpha^{2n}}{1 - \alpha^2} \right) - \sigma^2 \left(\frac{1 - \alpha^n}{1 - \alpha} \right)^2 \right\}, \quad (37)$$

where $\alpha = 1 - \mu^{(2)}/N$. We also have $\check{\beta}_1^c(n) = E_{\rho} \{ \check{\beta}_1^c(n) \}$ where

$$\check{\beta}_1^c(n) = \frac{\check{\beta}_1^{51}(n) P_{51} + \check{\beta}_1^{52}(n) P_{52}}{P_{\mathbb{B}}}. \quad (38)$$

In the above equation, we have $\cup^{51} = \cup^{11} \cup \cup^{21}$, $\cup^{52} = \cup^{12} \cup \cup^{22}$, $\mathbb{B} = \cup^{51} \cup \cup^{52}$, and $P_{\mathbb{B}} = P_{51} + P_{52}$. The term $\check{\beta}_1^{5j}(n)$ is expressed as

$$\check{\beta}_1^{5j}(n) = \begin{cases} (1 - \alpha^n)^2 E_{\gamma_1} \{ (a_1 + a_l \rho + \gamma_1 - w_{opt,1}^c)^2 | \gamma_1 > -(a_1 + a_l \rho) \}, & j=1, \\ (1 - \alpha^n)^2 E_{\gamma_l} \{ (a_1 - a_l \rho + \gamma_l - w_{opt,1}^c)^2 | \gamma_l > -(a_1 - a_l \rho) \}, & j=2. \end{cases} \quad (39)$$

4.2. Analysis of proposed algorithm

We assume that each user has the same power such that $a_k^2 = a_1^2, \forall k$ for the analysis hereafter. The generalization to the power-imbalanced scenario is straightforward. Substituting (9) into (10), we can have the despread output of the second stage for the first user as

$$y_1^{(2)} = a_1 b_1 + \sum_{j \neq 1} (a_j b_j - w_j(N) \hat{b}_j^{(1)}) \rho_{j1} + \gamma_1. \quad (40)$$

Note that the stage number on the superscript of $w_j(N)$ is omitted. Assuming that $y_1^{(2)}$ is a Gaussian random variable, we can estimate the BER in the second stage output. Note that we have the mean of $y_1^{(2)}$ as $a_1 b_1$. If interference cancellation is perfect with the ideal weights obtained in (18), the variance of $y_1^{(2)}$ is just σ^2 . However, since the interference cancellation is not perfect even for the proposed algorithm, the variance will be increased. There are two major sources of imperfect interference cancellation as demonstrated in the interference term of (40). The first residual interference results from erroneously selected weights (set as the channel gain) out of the pre-selection procedure; the increased variance is denoted by V_S . The other one is due to imperfect interference cancellation using adapted and post-filtered weights; the increased variance is denoted by V_F . Thus, the overall interference and noise variance is

$$\sigma_{out}^2 = \sigma^2 + V_S + V_F. \quad (41)$$

Without loss of generality, we let $b_1 = 1$. Assuming that the interference in (2) is Gaussian distributed, we can have the first stage BER as

$$P_e^{(1)} = \mathcal{Q} \left(\frac{a_1}{\sqrt{\sigma^2 + \frac{1}{N} \sum_{j \neq 1} a_j^2}} \right), \quad (42)$$

where $\mathcal{Q}(\cdot)$ is the Q-function. Then the probability that the user has correct decision in the first stage and its output is greater than the weight-selection threshold ($b_1 = 1$ and $y_1^{(1)} > \zeta_s^{(2)} a_1$) is

$$P_{SC} = \mathcal{Q} \left(\frac{a_1 (\zeta_s^{(2)} - 1)}{\sqrt{\sigma^2 + \frac{1}{N} \sum_{j \neq 1} a_j^2}} \right). \quad (43)$$

In other words, P_{SC} is the probability of correct weight pre-selection. In a similar way, the probability of erroneous weight pre-selection ($b_1 = 1$ and $y_1^{(1)} < -\zeta_s^{(2)} a_1$) is

$$P_{SE} = \mathcal{Q} \left(\frac{a_1 (\zeta_s^{(2)} + 1)}{\sqrt{\sigma^2 + \frac{1}{N} \sum_{j \neq 1} a_j^2}} \right). \quad (44)$$

For the users whose first-stage outputs are greater than $\zeta_s^{(2)} a_k$, the corresponding weights will not be adapted during the LMS algorithm. The interference power due to erroneous cancellation of the j th interference in (40) is calculated by

$$V_I = E\{(a_j b_j \rho_{j1} - \tilde{w}_j^c(N) \hat{b}_j \rho_{j1})^2\} = 4a_j^2/N, \quad (45)$$

where $E\{\rho_{ij}^2\} = 1/N$ is used, and we have $\tilde{w}_j^c(N) = a_j$ and $\hat{b}_j = -b_j$ in this case. The effective weight number in the LMS algorithm is reduced from K to K_{eff} where K_{eff} is approximated by $K_{eff} = K(1 - P_{SC} - P_{SE})$. Note that K_{eff} may not be an integer since it represents an estimate of the averaged weight number. Then we have

$$V_S = KP_{SE}V_I. \quad (46)$$

The enlarged effective noise variance can be obtained as $\sigma_{eff}^2 = \sigma^2 + V_S$. The mean and variance values of $\tilde{w}_j^c(N)$ and $\tilde{w}_j^e(N)$ can be approximated by the analytic results in (35) and (36), respectively. Note that when applying the analytic results in the last subsection to the weight outputs of the pre-selection module, we have to change the weight number from K to K_{eff} , the noise variance from σ^2 to σ_{eff}^2 , and let $a_l = a_1 \sqrt{K_{eff} - 1}$.

Now we calculate V_F . We treat V_F as the sum of two contributors from the interference cancellation: one is V_{FC} contributed from $\tilde{w}_j^c(N)$ and the other one, V_{FE} , from $\tilde{w}_j^e(N)$. We denote their corresponding probability density functions as $f(\tilde{w}_j^c(N))$ and $f(\tilde{w}_j^e(N))$, respectively. Fig. 3 gives an example of the simulated distribution outputs. When $\tilde{w}_1^c(N) > \zeta_r^{(2)} a_1$, no cancellation error will be introduced. Rather, the weights falling on regions 'A' and 'B' in the figure will introduce residual errors and the corresponding interference is represented by

$$V_{FCj} = E\{\rho_{jk}^2\} \int_{\zeta_r^{(2)} a_j}^{\zeta_r^{(2)} a_j} (\tilde{w}_j^c(N) - a_j)^2 f(\tilde{w}_j^c(N)) d\tilde{w}_j^c(N) + V_I \int_{-\infty}^{\zeta_r^{(2)} a_j} f(\tilde{w}_j^c(N)) d\tilde{w}_j^c(N), \quad (47)$$

where V_I is obtained in (45). Under the assumption of the Gaussian distribution, the cumulative density function of regions 'A' and 'B' can be found to be

$$P_{FCj} = 1 - \mathcal{Q} \left(\frac{a_j (\zeta_r^{(2)} - \eta_{Cj})}{\sigma_{Cj}} \right), \quad (48)$$

where η_{Cj} and σ_{Cj} are the analytical weight mean and standard deviation for correct decision output of the j th interference as given in (35) and (36), respectively. As to $\tilde{w}_j^e(N)$, erroneous weight decision occurs when $\tilde{w}_j^e(N) > \zeta_r^{(2)} a_j$, i.e., $A_F(\tilde{w}_j^e(N)) = a_j$ and $E\{(a_j b_j - \tilde{w}_j^e(N) \hat{b}_j)^2\} = E\{(a_j b_j - a_j(-b_j))^2\} = 4a_j^2$, as denoted by the region 'C' in the figure. Thus, we have

$$V_{FEj} = V_I \int_{\zeta_r^{(2)} a_j}^{\infty} f(\tilde{w}_j^e(N)) d\tilde{w}_j^e(N) + E\{\rho_{jk}^2\} \int_{\zeta_r^{(2)} a_j}^{\zeta_r^{(2)} a_j} (\tilde{w}_j^e(N) + a_j)^2 f(\tilde{w}_j^e(N)) d\tilde{w}_j^e(N), \quad (49)$$

where the second term in (49) corresponds to the interference level resulting from weights in region 'D' of the figure. The cumulative density function of regions 'C' and 'D' is expressed by

$$P_{FEj} = \mathcal{Q} \left(\frac{a_j (\zeta_r^{(2)} - \eta_{Ej})}{\sigma_{Ej}} \right), \quad (50)$$

where η_{Ej} and σ_{Ej} are the counterparts of η_{Cj} and σ_{Cj} , respectively, for erroneous decision outputs. Then, V_{FCj}

and $V_{FE,j}$ can be combined as

$$V_F = \sum_{j \neq 1} \frac{V_{FC,j} P_{FC,j} + V_{FE,j} P_{FE,j}}{P_{FC,j} + P_{FE,j}}. \quad (51)$$

Finally, we can express the BER in the second stage output from (41), (46) and (51) as

$$P_e^{(2)} = Q\left(\frac{a_1}{\sigma_{out}}\right). \quad (52)$$

5. Simulation results

In this section, we report simulation results to demonstrate the effectiveness of the proposed algorithm. We use random codes with $N=31$ as spreading sequences, and first consider parameter optimization in the proposed algorithm. As described, there are two new operations in the proposed algorithm, i.e., weight pre-selection and post-filtering. In the first set of simulations, we only consider the operation of weight post-filtering. We let $\zeta_l^{(2)} \ll -a_k$, and do not conduct weight pre-selection. The user number is 20 and $E_b/N_0=7$ dB ($E_b=a_k^2$, and $N_0=2\sigma^2$). Fig. 4 shows the performance comparison for different $\mu^{(2)}$ and $\zeta_r^{(2)}$ values. In the figure the optimal step size is normalized such that $\bar{\mu}^{(2)} = \mu^{(2)}/N$. It can be noted that when $\zeta_r^{(2)}$ is set higher (e.g., 0.7), the enlarged step size from 0.036 to 0.048 does not provide significant performance gain. However, when the post-filtering is reinforced by setting that $\zeta_r^{(2)} < 0.3$, the performance improvement for larger step sizes can be observed. This is because the over-adapted weights due to faster adaptation from a larger step size can be effectively corrected by the post-filtering procedure and thus the error rate decreases. We then incorporate the weight pre-

selection step and the result is shown in Fig. 5. In the figure we can observe that the optimal parameter set given $\bar{\mu}^{(2)}=0.048$ can be determined as $\zeta_s^{(2)}=1.2$. Comparing these figures we also find that the performance becomes less sensitive to the variation of post-filtering setting for $\zeta_r^{(2)} > 0.4$ when the weight pre-selection is utilized. This may be attributed to the fact that, for most users with high reliability, their adapted weights are usually close to the channel gain if no pre-selection is applied. When weight pre-selection is incorporated, most of the weights with large magnitudes will be deactivated. Therefore the influence of $\zeta_r^{(2)}$ on the performance is reduced. The optimization procedure for $\zeta_l^{(2)}$ is similar to that of $\zeta_r^{(2)}$ and is set as $\zeta_l^{(2)} = -0.2$ in the remaining simulations.

Now we report the performance comparison for various multiuser receivers. We consider partial PIC receivers which include the conventional matched filter, the non-adaptive partial PIC (referred to as PPIC) described in (3), the conventional adaptive partial PIC (referred to as the APPIC) described in (4)–(10), and the proposed algorithm. Optimum parameters in each algorithm are obtained empirically (such as $c_k^{(i)}$ for PPIC, $\bar{\mu}^{(i)}$ for APPIC, as well as the $\bar{\mu}^{(i)}$ and thresholds for $A_S(\cdot)$ and $A_F(\cdot)$ in the proposed algorithm). We first compare the performance of the proposed algorithm and other methods for different user numbers with $E_b/N_0=7$ dB. We let the maximum stage number be five. The optimal $c_k^{(i)}$ (same for all users) from Stage two to five are determined as $\{0.6, 0.65, 0.7, 0.75\}$. The optimal $\bar{\mu}^{(i)}$'s for APPIC are $\{0.02, 0.009, 0.004, 0.002\}$. The optimal $\bar{\mu}^{(i)}$ for the proposed algorithm are set as $\{0.055, 0.05, 0.045, 0.04\}$, and the thresholds as $\{\zeta_s^{(i)}, \zeta_l^{(i)}, \zeta_r^{(i)}\} = \{1.2, -0.2, 0.4\}$ for all stages. Fig. 6 shows the BER performance of the second stage

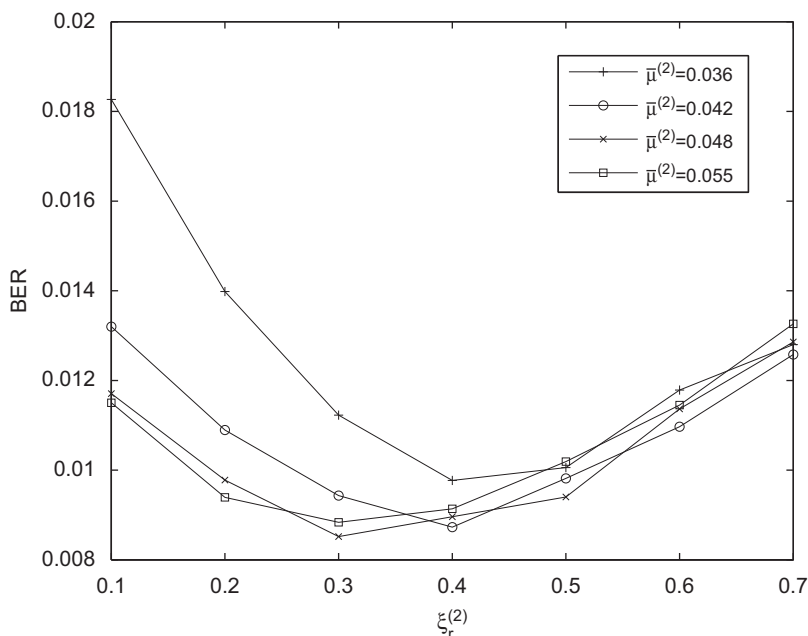


Fig. 4. Second-stage performance of the proposed algorithm (only weight post-filtering is activated).

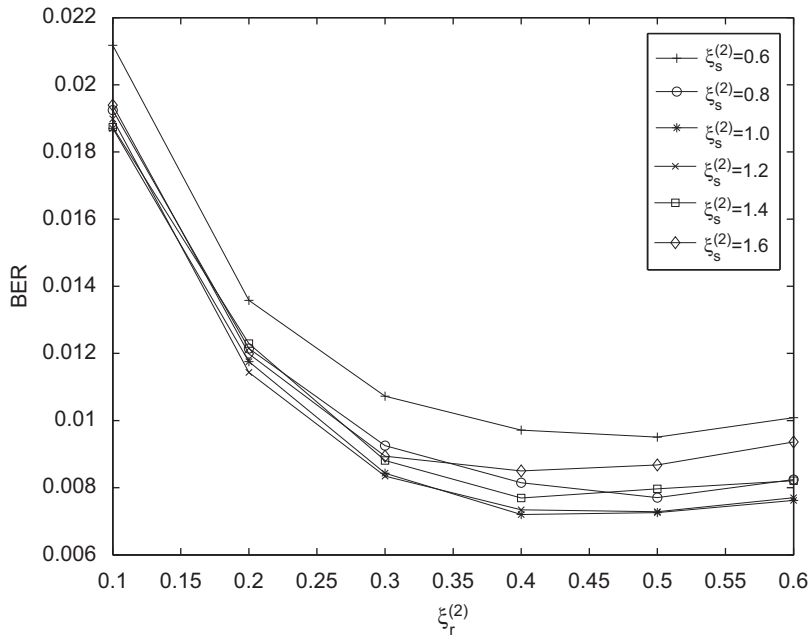


Fig. 5. Second-stage performance of the proposed algorithm (both weight pre-selection and post-filtering are activated).

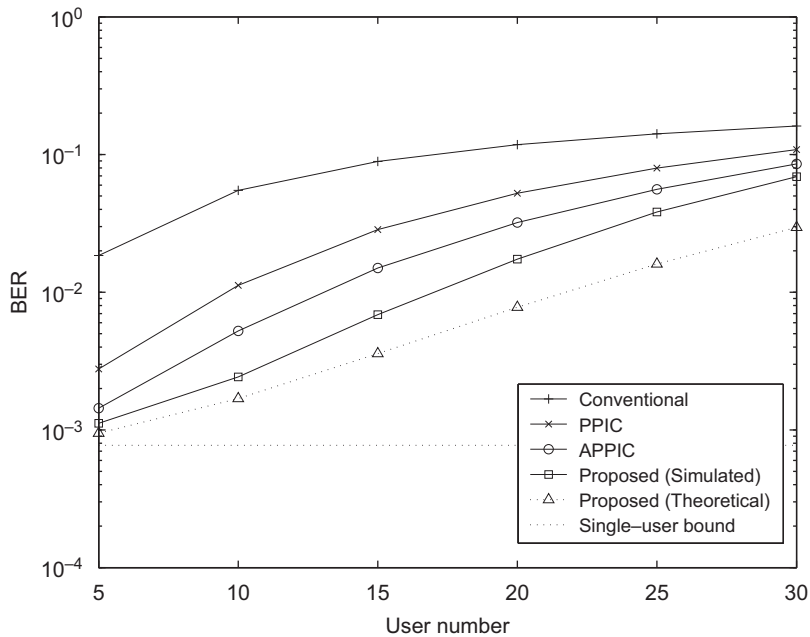


Fig. 6. Second-stage performance comparison for different user numbers ($E_b/N_0=7$ dB).

output. We can find that the conventional matched filter receiver gives the worst result due to MAI. The proposed algorithm performs better than APPIC in all cases. In the figure, the theoretical result in (52) is also shown for comparison. It can be seen that the analysis is accurate when the number of users is small while deviates from the simulated result gradually as the user number grows.

This is reasonable since the weight behavior analysis of the LMS algorithm is approximated from that of the single-user and two-user cases under the assumption of power balance. We also show the performance for the outputs of the fifth stage in Fig. 7. As we can see, the performance of all adaptive partial PIC receivers are close to the single-user bound when the number of users is

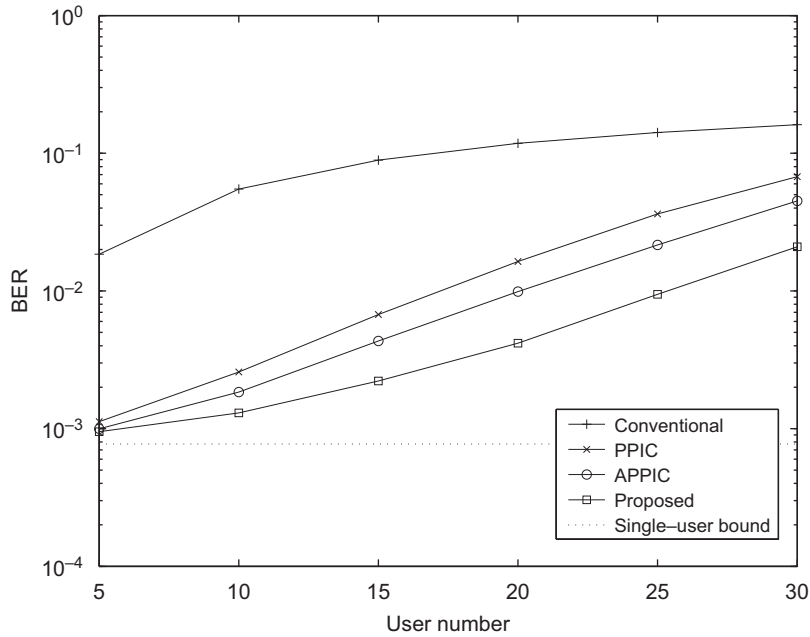


Fig. 7. Fifth-stage performance comparison for different user numbers ($E_b/N_0=7$ dB).

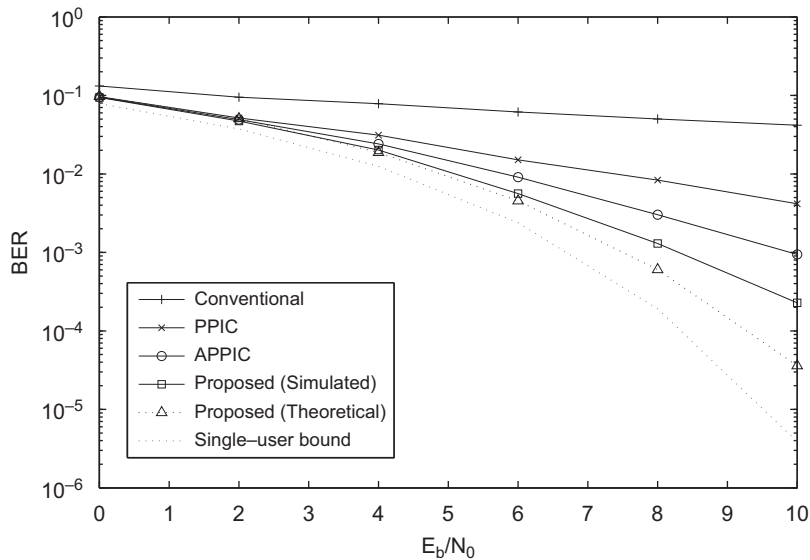


Fig. 8. Second stage performance comparison for different E_b/N_0 ratios ($K=10$).

small. Also note that, from the empirical parameter setting stated above, the optimal step sizes in the proposed algorithm are larger than those in the conventional APPIC approach. Thus, the convergence can be accelerated, and then the performance can be improved accordingly. We then conduct the performance comparison under different E_b/N_0 's (10 users). Figs. 8 and 9 show the performance comparison for the second and fifth stage outputs, respectively. The parameters used here are the same as those in Fig. 6. We can observe that the

performance of the proposed algorithm is close to the single-user bound for low to median E_b/N_0 values. The analytic result for the proposed algorithm at the second stage output is also shown in Fig. 8. From the figure, we can see that the behavior of the analytic result is quite similar to that of simulations for low to moderate E_b/N_0 values. We also compare the system performance under a power-imbalanced scenario. The user powers are equally distributed in linear scale and the power ratio between the strongest and weakest users is set as 15 dB. The

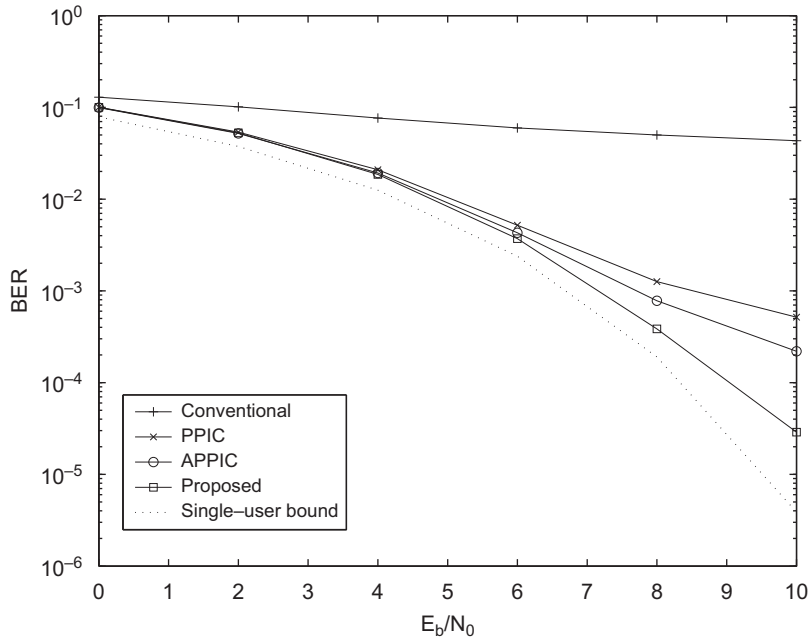


Fig. 9. Fifth-stage performance comparison for different E_b/N_0 ratios ($K=10$).

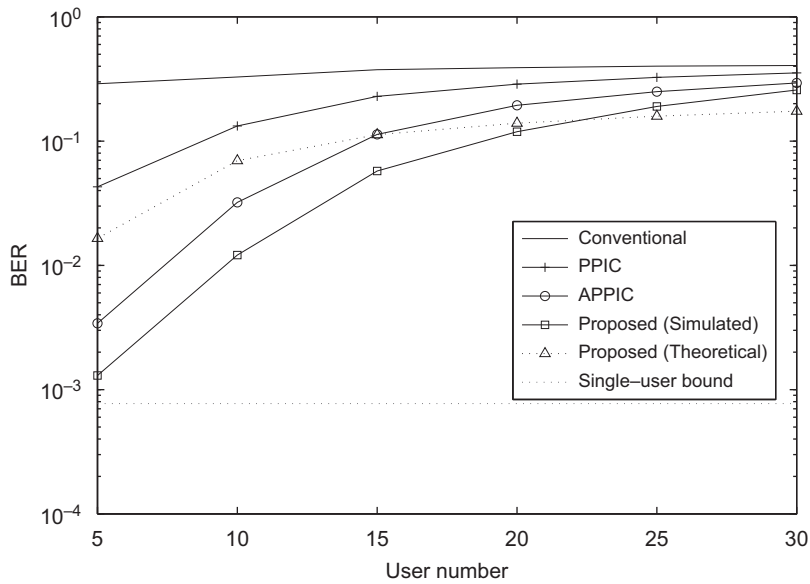


Fig. 10. Second stage performance comparison for the weakest user under power imbalance ($E_b/N_0=7$ dB for the weakest user).

parameters are kept unchanged except that the optimal $\bar{\mu}^{(i)}$ values for APPIC are set as $\{0.034, 0.01, 0.005, 0.002\}$. In Figs. 10 and 11, we show the BER performance for the weakest user in the second and fifth stage outputs, respectively. It can be seen that the proposed algorithm provides a significant performance gain, especially when the user number is large. Note that the proposed algorithm can make the performance of the weakest user indistinguishable from the single-user bound when the user number is smaller than 20. The reason for this superior performance is due to the fact that stronger users

have lower probability of errors. As a result, the weight pre-selection function tends to set the cancellation weights of stronger users as their channel gains, and the effective user number in the LMS algorithm is then decreased stage-by-stage. This behavior is very similar to that in the SIC approach. In addition, the interference is further reduced with the filtered weights. Thus the proposed algorithm can approach the single user bound, just like what SIC performs, but with fewer stages. The performance comparison for the second stage output as depicted in Fig. 10 shows that the gap between the

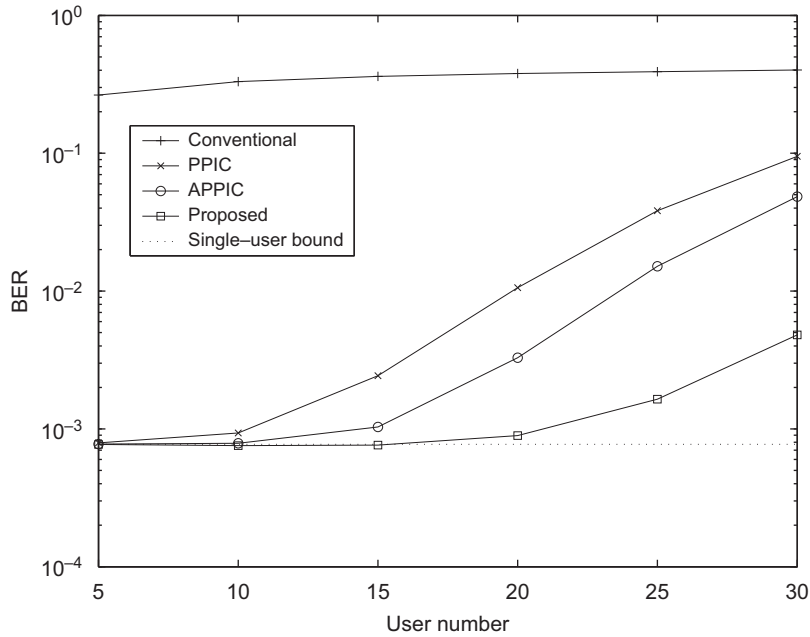


Fig. 11. Fifth-stage performance comparison of the weakest user under power imbalance ($E_b/N_0=7$ dB for the weakest user).

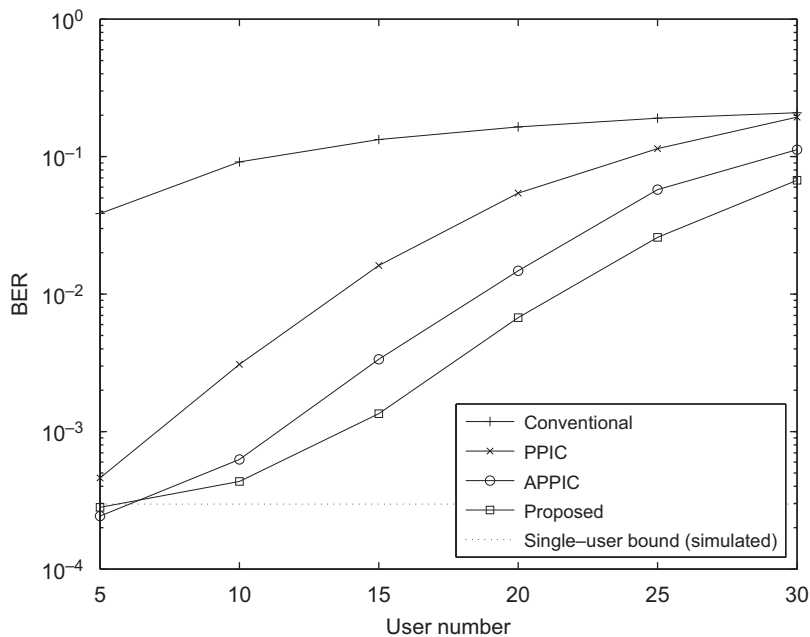


Fig. 12. Fifth-stage performance comparison under a two-ray multipath fading channel ($E_b/N_0=17$ dB).

analytic and simulated results is larger when the number of users is smaller. This may be due to the fact that the theoretical results are derived with the approximation of the equal-power two-user scenario. When the number of users is smaller and the power is imbalanced, the approximation is less valid and analytic results are less accurate.

In the following, we consider the performance of the proposed algorithm under the multipath fading channel.

We use a two-path fading channel where the second path is one chip delay with respect to the main path, and each path gain is Gaussian distributed with zero mean and equal variance. The optimal weights for PPIC are determined as $\{0.7, 0.8, 0.85, 0.9\}$. The optimal $\bar{\mu}^{(i)}$ are set as $\{0.012, 0.007, 0.003, 0.001\}$ for APPIC and $\{0.025, 0.023, 0.021, 0.02\}$ for the proposed algorithm. The thresholds are given as $\zeta_s^{(i)} = 2.4$, $\zeta_l^{(i)} = -0.5$, and $\zeta_r^{(i)} = 0.5$ for all stages. The result is shown in Fig. 12, and we can

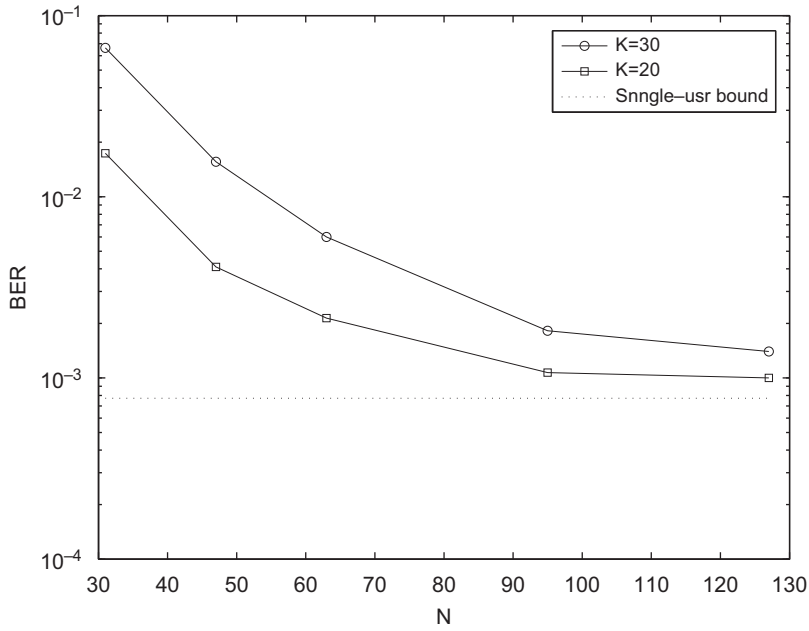


Fig. 13. Second-stage performance comparison for different spreading factors ($E_b/N_0=7$ dB).

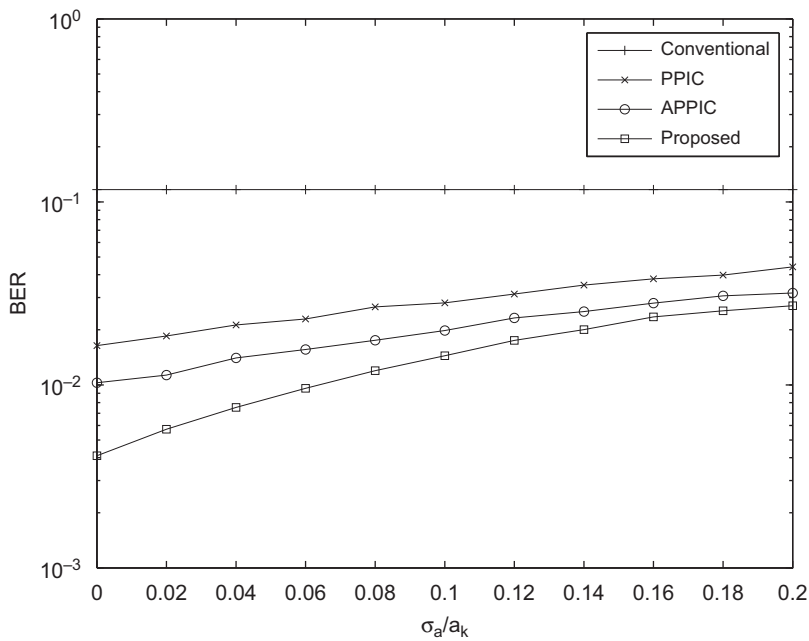


Fig. 14. Fifth-stage performance comparison for different channel estimation errors ($K=20, E_b/N_0=7$ dB).

see that the proposed algorithm still performs better than other approaches.

An attractive feature for the proposed algorithm is that the adaptation is conducted chip-by-chip and convergence is fast. Each time when a new bit is received, the weight values are reset and then adjusted. The larger the value of N , the more data we can have and the better performance we can expect. To conform this assertion, we then conduct simulations with the scenario of varied N .

Fig. 13 shows the performance comparison for the second stage output. The simulation configuration is the same as that of Fig. 6 except that the stepsize is kept constant ($\mu^{(2)} = 0.048 * 31$ for all N values). From the figure, we can observe that the system performance will approach the single-user bound when N is large.

In the adaptive PPIC receiver scenario, the channel information is required for the determination of initial values. When the proposed algorithm is utilized, the

channel information is further used to determine the optimal parameters. All of the simulations conducted above have assumed perfect channel estimation. However, in a practical system, the channel estimation error always exists, and its effect has to be taken into account. To have an idea how multiuser detection algorithms are affected by the error, we model the error as a Gaussian random variable with a standard deviation of σ_a and conduct simulations for different σ_a 's. Fig. 14 shows the simulation result. It can be seen in the figure that the proposed algorithm always performs better than PPIC under different levels of channel estimation error.

6. Conclusions

Multiuser detection is one of the key techniques for enhancing the capacity of DS-CDMA systems. Due to its simplicity and effectiveness, the adaptive partial PIC receivers has been considered as a promising approach in multiuser detection. In this paper, we propose an enhanced algorithm for the adaptive partial PIC. The main idea is to use a weight pre-selection procedure and a post-filtering scheme to reduce the weight error variance. Simulation results show that the proposed algorithm outperforms the conventional adaptive approach in all scenarios. In power-imbalanced systems, the proposed algorithm can even approach the single-user bound. We also conduct performance analysis and derive the output BER in the second stage. Simulations confirm that the analytic results are accurate. In addition to dealing with MAI in single-carrier CDMA systems, the proposed algorithm can also be extended to inter-code interference (ICI) problem in multicarrier CDMA (MC-CDMA) systems [27–29]. Note that the MC-CDMA system has been considered as a candidate for advanced wireless communication. Research on this subject is now underway.

References

- [1] S. Verdu, Minimum probability of error for asynchronous Gaussian multiple-access channels, *IEEE Trans. Inf. Theory* 32 (1) (1986) 85–96.
- [2] S. Moshavi, Multi-user detection for DS-CDMA communications, *IEEE Commun. Mag.* 34 (10) (1996) 124–136.
- [3] A. Duel-Hallen, J. Holtzman, Z. Zvonar, Multiuser detection for CDMA systems, *IEEE Personal Commun.* 2 (2) (1995) 46–58.
- [4] S. Verdu, *Multiuser Detection*, Cambridge University Press, Cambridge, UK, 1998.
- [5] J.M. Holtzman, Successive interference cancellation for direct sequence code division multiple access, in: *Proceedings of the IEEE Military Communication Conference (MILCOM '94)*, Fort Monmouth, NJ, vol. 3, October 1994, pp. 997–1001.
- [6] P. Patel, J.M. Holtzman, Analysis of a simple successive interference cancellation scheme in a DS/CDMA system, *IEEE J. Sel. Areas Commun.* 12 (5) (1994) 796–807.
- [7] M.K. Varanasi, B. Aazhang, Multistage detection in asynchronous code-division multiple-access communications, *IEEE Trans. Commun.* 38 (4) (1990) 509–519.
- [8] R. Kohno, et al., An adaptive canceller of cochannel interference for spread-spectrum multiple-access communication networks in a power line, *IEEE J. Sel. Areas Commun.* 8 (4) (1990) 691–699.
- [9] D. Divsalar, M.K. Simon, D. Raphaeli, Improved parallel interference cancellation for CDMA, *IEEE Trans. Commun.* 46 (2) (1998) 258–268.
- [10] D. Guo, et al., A matrix-algebraic approach to linear parallel interference cancellation in CDMA, *IEEE Trans. Commun.* 48 (1) (2000) 152–161.
- [11] Y. Li, M. Chen, S. Cheng, Determination of cancellation factors for soft-decision partial PIC detector in DS/CDMA systems, *Electron. Lett.* 36 (3) (2000) 239–241.
- [12] B.S. Abrams, A.E. Zeger, T.E. Jones, Efficiently structured CDMA receiver with near-far immunity, *IEEE Trans. Veh. Technol.* 44 (1) (1995) 1–13.
- [13] P.G. Renucci, B.D. Woerner, Optimization of soft interference cancellation for DS-CDMA, *Electron. Lett.* 34 (8) (1998) 731–733.
- [14] W. Zha, S.D. Blostein, Soft-decision multistage multiuser interference cancellation, *IEEE Trans. Veh. Technol.* 52 (2) (2003) 380–389.
- [15] P. Haghghat, A. Ghraryeb, Trickle-based interference cancellation schemes for CDMA systems, *IEEE Trans. Wireless Commun.* 8 (1) (2009) 13–17.
- [16] Y.B. Guo, J.R. Cavallaro, A low complexity and low power SoC design architecture for adaptive MAI suppression in CDMA systems, *J. VLSI Signal Process.* 44 (3) (2006) 195–217.
- [17] G. Xue, et al., Adaptive multistage parallel interference cancellation for CDMA, *IEEE J. Sel. Areas Commun.* 17 (10) (1999) 1815–1827.
- [18] S.R. Kim, et al., Adaptive weighted parallel interference cancellation for CDMA systems, *Electron. Lett.* 34 (22) (1998) 2085–2086.
- [19] W.A. Hamouda, P.J. McLane, A fast adaptive algorithm for MMSE receivers in DS-CDMA systems, *IEEE Signal Process. Lett.* 11 (2) (2004) 86–89.
- [20] A.I. Hassan, D.K. Mehra, Interference suppression in multiuser DS-CDMA communication systems using RMGS algorithm, *Signal Process.* 82 (7) (2002) 961–969.
- [21] M. Honig, U. Madhow, S. Verdu, Blind adaptive multiuser detection, *IEEE Trans. Inf. Theory* 41 (4) (1995) 944–960.
- [22] P.B. Rapajic, B.S. Vucetic, Adaptive receiver structures for asynchronous CDMA systems, *IEEE J. Sel. Areas Commun.* 12 (4) (1994) 685–697.
- [23] U. Madhow, M. Honig, MMSE interference suppression for direct-sequence spread-spectrum CDMA, *IEEE Trans. Commun.* 42 (12) (1994) 3178–3188.
- [24] S.H. Han, J.H. Lee, Adaptive multi-stage parallel interference cancellation receiver for multi-rate DS-CDMA system, *IEICE Trans. Commun.* E 87-B (8) (2004) 2401–2405.
- [25] Y.-F. Huang, H.-L. Hung, J.-H. Wen, Performance of a novel adaptive multistage full parallel interference canceller for CDMA systems, *IEEE Trans. Veh. Technol.* 57 (3) (2008) 1944–1951.
- [26] Y.-T. Hsieh, W.-R. Wu, Performance analysis of an adaptive two-stage PIC CDMA receiver in AWGN channels, *Signal Process.* 88 (6) (2008) 1413–1427.
- [27] C.-L. Wang, C.-C. Chu, C.-C. Wu, Variance-reduced partial parallel interference cancellation for MC-CDMA uplink systems, *IEEE Trans. Wireless Commun.* 8 (10) (2009) 2951–4956.
- [28] M. Frikel, et al., BER analysis of parallel and successive interference cancellation employing MMSE equalizer for MC-CDMA systems, in: *International Conference on Multimedia Computing Systems (ICMCS '09)*, Morocco, April 2009, pp. 158–161.
- [29] M. McGuire, M. Sima, Parallel detection of MC-CDMA in fast fading, *IEEE Trans. Wireless Commun.* 8 (12) (2009) 5916–5927.