Low-Complexity Equalization by Iterative Interference Cancellation for UWB Communications

Yafei Tian, *Member, IEEE*, and Chenyang Yang, *Senior Member, IEEE*

***Abstract*—Efficiently reducing complexity and capturing energy are two critical issues for implementing equalizers in a high-rate ultra-wideband communication link with severe intersymbol interference. In this letter, we propose a low-complexity equalization algorithm using iterative interference cancellation. Its complexity is comparable with the Rake receiver, but its performance can approach that of the matched filter bound. The proposed algorithm can accommodate various signaling schemes, including pulse amplitude/position modulation and M-ary bi-orthogonal keying.**

***Index Terms*—Equalization, interference cancellation, iterative algorithm, ultra-wideband (UWB).**

I. Introduction

THE large amount of resolvable paths in ultra-wideband (UWB) channel will lead to very high-complexity equalization when severe intersymbol interference (ISI) exists in high-rate communication links. When the channel delay spread is much longer than the UWB pulse or chip duration, chip-level linear minimum-mean-square error (LMMSE) or decision feedback equalization (DFE), which are often used in wideband systems, will require a large number of equalizer taps. Moreover, UWB systems often employ M-ary pulse position modulation (PPM), and then M different equalizers are required [1].

One way to cope with this problem is to reduce the number of equalizer taps. References [2] and [3] proposed symbol-level algorithms, but they need a chip-level Rake combiner before down-sampling the signal to the symbol-rate. This will aggravate the computational burden of the data detection. References [4] and [5] designed tap selection methods by exploiting the sparsity feature of UWB channels, but the selection algorithms are not simple.

Some other detection algorithms, such as interference cancellation, can use the channel estimation directly to detect the signal. When the training sequence is appropriately designed, channel coefficients can be estimated much more easily than the equalization taps. Nevertheless, the interference cancellation algorithms are originally designed for multiuser detection. They will waste the multipath energy when directly applied for equalization.

Manuscript received July 18, 2008; revised September 12, 2008. This work was supported by the National Natural Science Foundation of China (NSFC) under Grant 60802015. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Kainam Thomas Wong.

The authors are with the School of Electronics and Information Engineering, Beihang University, Beijing 100191, China (e-mail: ytian@buaa.edu.cn; cyyang@buaa.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LSP.2008.2007621

Considering that the multipath channel response covers multiple symbols, we can regard ISI as a sort of mutual interference from multiple "virtual users," while the channel response is the characteristic signature of these "users." Inspired by the principle of iterative multiuser joint decoding [6], we propose an iterative interference cancellation (IIC) algorithm for channel equalization, where the ISIs are removed but their energies are reserved. This is critical to improve the performance of the strictly power-constraint UWB systems.

While various iterative equalization algorithms have been developed, for example, [7]–[9], the highlight of the IIC algorithm is exploiting the equivalence of ISI and virtual multiuser interference, and achieving good performance with low complexity. The IIC algorithm is an approximate maximum *a posteriori* (MAP) detector. Its performance can approach that of the matched filter bound (MFB), while its complexity is as low as that of Rake receivers. The IIC algorithm can work with various modulation schemes, including pulse amplitude modulation (PAM), PPM, and M-ary bi-orthogonal keying (MBOK), etc. In sparse channels, its complexity can be further reduced by selecting significant multipath components.

The rest of this letter is organized as follows. Section II introduces the signal models. Section III derives the IIC algorithm and presents the practical implementation issues. Simulation results and performance comparison with other algorithms are shown in Section IV, and conclusions are provided in the last section.

II. Signal Description

Consider the general waveform modulated signal

$$s(t) = \sqrt{E_s}\sum_{n=0}^{N-1}\sum_{m=0}^{M-1} a_{n,m}\phi_m(t-nT_s) \tag{1}$$

where $\phi_m(t),\ 0<t<T_s$ represents the mth modulation waveform, N is the number of transmitted symbols, E_s is the symbol energy, T_s is the symbol duration, and $a_{n,m}\in\{\pm 1, 0\}$ is the amplitude modulated on the waveform. In each symbol, only one $a_{n,m}$ can pick the nonzero value, i.e., only one waveform is used for each symbol transmission, and the number of possible waveforms is $2M$ considering the bipolar modulation.

The waveform can be a low-duty cycle pulse or a sort of spreading sequences. For example, when $M=1$, the expression of (1) can represent PAM or direct-sequence spreading (DSS) signals, and when $M>1$, it can represent pulse position amplitude modulation (PPAM) or MBOK signals.

Define the transmitted amplitude vector as

$$\boldsymbol{a}_n = [a_{n,0}, a_{n,1}, \ldots, a_{n,M-1}]^T$$

where $\boldsymbol{a}_n \in \mathcal{A}$, which is a space including $2M$ amplitude vectors. By writing the transmitted waveforms as a vector

$$\boldsymbol{\phi}(t) = [\phi_0(t), \phi_1(t), \ldots, \phi_{M-1}(t)]^T$$

we can obtain a brief expression of the transmitted signal as

$$s(t) = \sqrt{E_s} \sum_{n=0}^{N-1} \boldsymbol{a}_n^T \boldsymbol{\phi}(t - nT_s). \tag{2}$$

Considering the impulse response of the UWB multipath channel $h(t)$, the local reference pulse waveform $p(t)$, the composite channel responses for the mth waveform can be defined as

$$h_m(t) = \phi_m(t) * h(t) * p(t),\ m = 0, \ldots, M-1 \tag{3}$$

where "$*$ " denotes convolution operation.

Sampling $h_m(t)$ with interval T_c, we obtain the discrete equivalent channel response

$$h_m[i] = h_m(iT_c),\ i = 0, 1, \ldots, KN_c - 1 \tag{4}$$

where K is the number of symbols that the composite channel response covers, $N_c = T_s/T_c$ is the number of samples in each symbol, and T_c can be the pulse duration in IR signals or chip duration in DSS signals. If the channel is sparse, many values of $h_m[i]$ will be close to zero.

Divide the channel response $h_m[i]$ into K segments, each of which has a duration of one symbol. The N_c components in the kth segment are formed as a vector

$$\begin{aligned} \boldsymbol{h}_{k,m} &= [h_m[kN_c],\ h_m[kN_c+1], \ldots, h_m[(k+1)N_c - 1]]^T \\ k &= 0, \ldots, K-1,\ m = 0, \ldots, M-1. \end{aligned} \tag{5}$$

To represent the discrete channel response of the M possible waveforms at the kth segment, we further write these vectors as a matrix $\boldsymbol{H}_k = [\boldsymbol{h}_{k,0}, \boldsymbol{h}_{k,1}, \ldots, \boldsymbol{h}_{k,M-1}]$, which is analogous to a multiple-input multiple-output (MIMO) channel matrix with M transmit and N_c receive antennas [1].

Further consider that the received signal in each symbol duration will be corrupted by $K-1$ previous symbols, and we can express the samples in the nth received symbol as follows:

$$\boldsymbol{y}_n = \sum_{k=0}^{K-1} \boldsymbol{H}_k \boldsymbol{a}_{n-k} + \boldsymbol{z}_n,\ n = 0, \ldots, N-1 \tag{6}$$

where $\boldsymbol{y}_n = [y_{n,0}, \ldots, y_{n,N_c-1}]^T$, $\boldsymbol{z}_n$ is the Gaussian noise vector, which has zero mean and covariance matrix $\sigma^2 \boldsymbol{I}_{N_c}$. The noise variance $\sigma^2 = N_0/(2E_s)$, $N_0/2$ is the two-sided power spectrum density, and $\boldsymbol{I}_{N_c}$ is the identity matrix of size N_c. In (6), we have dropped the multiplication of the signal energy $\sqrt{E_s}$ for brevity, and we count the effect of the signal-to-noise ratio (SNR) to the noise variance.

It is shown from (6) that the received symbol is a sum of the signals from K transmitted symbols. Therefore, we can regard the K transmitted symbols as K "virtual users" in each received symbol, where the signals from different "virtual users" will induce ISI. Considering the fact that UWB signals will lead to abundant resolvable paths, the channel response vector in (5) can be viewed as the pseudo-noise (PN) sequence in multiuser systems. Since same transmitted information is carried on different "PN sequences," it is not appropriate to simply discard these multipath signals after interference cancellation.

III. Iterative Interference Cancellation Algorithm

A. Algorithm Derivations

The idea of our IIC algorithm is to iteratively cancel the "virtual user" interference in each symbol and combine the residual signals to enhance the estimation of the transmitted information. We start from the MAP detection and use the sum-product algorithm as a tool to develop the IIC algorithm in a systematic way by introducing some approximations.

Conditioned on all the received signal vectors $\boldsymbol{y}_0, \ldots, \boldsymbol{y}_{N-1}$, $\boldsymbol{a}_n$ can be detected using the MAP algorithm, which makes the symbol error rate minimum, i.e.,

$$\hat{\boldsymbol{a}}_n = \arg\max_{\boldsymbol{a} \in \mathcal{A}} \mathrm{APP}_n(\boldsymbol{a}) \tag{7}$$

where $\mathrm{APP}_n(\boldsymbol{a})$ is the marginal *a posteriori* probability (APP) of symbol $\boldsymbol{a}_n$

$$\begin{aligned} \mathrm{APP}_n(\boldsymbol{a}) &= p(\boldsymbol{a}_n = \boldsymbol{a} | \boldsymbol{y}_0, \ldots, \boldsymbol{y}_{N-1}) \\ &= \sum_{\sim\{\boldsymbol{a}_n\}} p(\boldsymbol{a}_0, \ldots, \boldsymbol{a}_{N-1} | \boldsymbol{y}_0, \ldots, \boldsymbol{y}_{N-1}) \end{aligned} \tag{8}$$

where $\sum_{\sim\{\boldsymbol{a}_n\}}$ represents the summation over all possible values of $\boldsymbol{a}_0, \ldots, \boldsymbol{a}_{N-1}$ excluding $\boldsymbol{a}_n$.

According to (6), the joint APP of the transmitted symbols $p(\boldsymbol{a}_0, \ldots, \boldsymbol{a}_{N-1} | \boldsymbol{y}_0, \ldots, \boldsymbol{y}_{N-1})$ can be factorized as the product of local functions

$$\begin{aligned} &p(\boldsymbol{a}_0, \ldots, \boldsymbol{a}_{N-1} | \boldsymbol{y}_0, \ldots, \boldsymbol{y}_{N-1}) \\ &\quad \propto p(\boldsymbol{y}_0, \ldots, \boldsymbol{y}_{N-1} | \boldsymbol{a}_0, \ldots, \boldsymbol{a}_{N-1}) \\ &\quad \propto \prod_n p(\boldsymbol{y}_n | \boldsymbol{a}_{n-K+1}, \ldots, \boldsymbol{a}_n) \\ &\quad \propto \prod_n \exp\left(-\frac{1}{2\sigma^2} \left\| \boldsymbol{y}_n - \sum_{k=0}^{K-1} \boldsymbol{H}_k \boldsymbol{a}_{n-k} \right\|^2 \right) \end{aligned} \tag{9}$$

where $\propto$ denotes proportionality. The last row of (9) comes from the vector Gaussian distribution of $\boldsymbol{y}_n$ conditioned on the transmitted symbols and the given channel realization.

Brute force computation of the marginal APP shown in (8) has complexity of an order of $(2M)^N$, which is intractable for practical applications. A general approximation method is to apply the sum-product algorithm on a factor graph [10]. According to the factorization in (9), a bipartite graph can be drawn as in Fig. 1, where K is assumed to be 3 in this example, $\boldsymbol{a}_n$ is the nth variable node, and $f_n = p(\boldsymbol{y}_n | \boldsymbol{a}_{n-K+1}, \ldots, \boldsymbol{a}_n)$ is the nth factor node.

In the sum-product algorithm, adjacent nodes in the factor graph exchange messages in either directions along the edges. Let $\mu_{\boldsymbol{a}_{n'} \to f_n}(\boldsymbol{a})$ denote the message sent from node $\boldsymbol{a}_{n'}$ to node f_n and $\mu_{f_n \to \boldsymbol{a}_{n'}}(\boldsymbol{a})$ denote the message sent from node f_n to node $\boldsymbol{a}_{n'}$, where $\boldsymbol{a}_{n'}$ is the neighbor node of f_n. Both $\mu_{\boldsymbol{a}_{n'} \to f_n}(\boldsymbol{a})$ and $\mu_{f_n \to \boldsymbol{a}_{n'}}(\boldsymbol{a})$ are probability mass functions (pmf) defined over $\mathcal{A}$, and hence have $2M$ assumptions. Following the rules of sum-product algorithm, the message computations performed at the variable and factor nodes are[10]

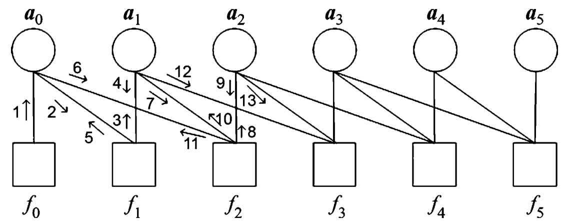

Fig. 1. Factor graph for an ISI channel with $K = 3$.

$$\mu_{\boldsymbol{a}_{n'} \to f_n}(\boldsymbol{a}) = \prod_{h \in \mathcal{N}(\boldsymbol{a}_{n'}) \setminus \{f_n\}} \mu_{h \to \boldsymbol{a}_{n'}}(\boldsymbol{a}) \tag{10}$$

$$\mu_{f_n \to \boldsymbol{a}_{n'}}(\boldsymbol{a}) = \sum_{\sim \{\boldsymbol{a}_{n'}\}} \left[f_n \prod_{\boldsymbol{c} \in \mathcal{N}(f_n) \setminus \{\boldsymbol{a}_{n'}\}} \mu_{\boldsymbol{c} \to f_n}(\boldsymbol{a}) \right] \tag{11}$$

where $\mathcal{N}(\boldsymbol{a}_{n'}) \setminus \{f_n\}$ means all the neighbors of $\boldsymbol{a}_{n'}$ excluding f_n, and $\sim \{\boldsymbol{a}_{n'}\}$ means all the variables of f_n excluding $\boldsymbol{a}_{n'}$.

The variables of f_n comprises K symbols; therefore, the computational complexity of (11) is an order of $(2M)^K$, which is still unacceptable when K is large. Since the received symbol is the summation of virtual multiuser signals, interference cancellation can be used to simplify the expression of (11). Replacing $\mu_{\boldsymbol{a}_m \to f_n}(\boldsymbol{a})$ with its single mass point approximation [6], i.e., using the hard decision of the transmitted symbol instead of its probability, we can obtain

$$\tilde{\mu}_{\boldsymbol{a}_{n'} \to f_n}(\boldsymbol{a}) = \begin{cases} 1, & \boldsymbol{a} = \arg\max_{\boldsymbol{a}' \in \mathcal{A}} \mu_{\boldsymbol{a}_{n'} \to f_n}(\boldsymbol{a}') \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

Equation (11) reduces to

$$\mu_{f_n \to \boldsymbol{a}_{n'}}(\boldsymbol{a}) \propto \exp\left(-\frac{\|\boldsymbol{y}_n - \tilde{\boldsymbol{y}}_{n',n} - \boldsymbol{H}_{n-n'}\boldsymbol{a}\|^2}{2\sigma^2} \right) \tag{13}$$

where

$$\tilde{\boldsymbol{y}}_{n',n} = \sum_{\substack{k=0 \\ k \neq n-n'}}^{K-1} \boldsymbol{H}_k \tilde{\boldsymbol{a}}_{n-k} \tag{14}$$

denoting the interference from other symbols when detecting $\boldsymbol{a}_{n'}$, with

$$\tilde{\boldsymbol{a}}_{n-k} = \arg\max_{\boldsymbol{a}' \in \mathcal{A}} \mu_{\boldsymbol{a}_{n-k} \to f_n}(\boldsymbol{a}') \tag{15}$$

which is the tentative hard decision from its pmf obtained from (10).

Actually, the pmf obtained from (10) is an extrinsic information (EXT). While it is necessary to use EXTs as passing messages in the standard sum-product algorithm, using APPs here can obtain more accurate tentative hard decisions and thus can achieve better performance. Moreover, for reducing the complexity, the messages can be computed in the logarithm domain, i.e.,

$$\bar{\mu}_{f_n \to \boldsymbol{a}_{n'}}(\boldsymbol{a}) \propto -\|\boldsymbol{y}_n - \tilde{\boldsymbol{y}}_{n',n} - \boldsymbol{H}_{n-n'}\boldsymbol{a}\|^2 \tag{16}$$

$$\bar{\mu}_{\boldsymbol{a}_{n'} \to f_n}(\boldsymbol{a}) = \sum_{h \in \mathcal{N}(\boldsymbol{a}_{n'})} \bar{\mu}_{h \to \boldsymbol{a}_{n'}}(\boldsymbol{a}) \tag{17}$$

where (17) is the APP instead of the EXT. After several iterations, (17) can be used to obtain the final decision of $\boldsymbol{a}_{n'}$.

Now the sum-product algorithm degenerates to an iterative interference cancellation (IIC) algorithm. Equation (16) embodies the interference cancellation operation, while (17) represents the combining of information.

Denoting $\bar{\boldsymbol{y}}_{n',n} = \boldsymbol{y}_n - \tilde{\boldsymbol{y}}_{n',n}$ as the received signal of $\boldsymbol{a}_n'$ in the nth symbol after interference cancellation, we can further simplify the expression of (16) by assuming that $\|\boldsymbol{H}_{n-n'}\boldsymbol{a}\|^2$ is approximately identical for different values of $\boldsymbol{a}$. Then, we have

$$\bar{\mu}_{f_n \to \boldsymbol{a}_{n'}}(\boldsymbol{a}) \propto C + \bar{\boldsymbol{y}}_{n',n}^T \boldsymbol{H}_{n-n'}\boldsymbol{a} \tag{18}$$

where C is a constant which is of no use in the iterative process, and the second term of the right-hand side is a correlation between the estimated interference-free signal $\bar{\boldsymbol{y}}_{n',n}$ and the desired noise-free received signal $\boldsymbol{H}_{n-n'}\boldsymbol{a}$.

B. Implementation Issues

In practical implementations of the IIC algorithm, we need also consider the scheduling in the iterative process of message passing. Usually, there are serial and parallel scheduling that can be used. Since the parallel scheduling requires large memory to store the information of each node, we use serial scheduling and the order of message passing is labeled at Fig. 1. After the first round of iterations, we can obtain quite accurate decisions of all the transmitted symbols. When we use these decisions instead of (15) for other rounds of iterations, the detection performance can be further improved.

We summarize the procedure of this algorithm as follows.

1) Initialize a register for correlation values $c_{reg}(n, m) = 0$, $n = 0, \ldots, N + K - 2$, $m = 0, \ldots, M - 1$ and a register for tentative hard decisions

$$\tilde{\boldsymbol{a}}_n = \begin{cases} \boldsymbol{a}_n, & n = -K+1, \ldots, -1 \\ \boldsymbol{0}, & n = 0, \ldots, N+K-2 \end{cases}$$

where $\tilde{\boldsymbol{a}}_n$ is a vector of size M, $\tilde{a}_{n,m}$ is its mth element, and $\boldsymbol{a}_{-K+1}, \ldots, \boldsymbol{a}_{-1}$ are the known training symbols.

2) Cancel the interference iteratively according to the sequence of received symbols, from $\boldsymbol{y}_0$ to $\boldsymbol{y}_{N+K-2}$, and obtain hard decisions of transmitted symbols $\boldsymbol{a}_0, \ldots, \boldsymbol{a}_{N-1}$. Specifically, when receiving $\boldsymbol{y}_n$, the following three steps are implemented for each node $\boldsymbol{a}_{n'}$, $n' = n, \ldots, n-K+1, (0 \leq n' \leq N-1)$, respectively.

a) Applying the tentative hard decisions $\tilde{\boldsymbol{a}}_n, \ldots, \tilde{\boldsymbol{a}}_{n-K+1}$, cancel the interference signals from the received signal

$$\bar{\boldsymbol{y}}_{n',n} = \boldsymbol{y}_n - \sum_{\substack{k=0 \\ k \neq n-n'}}^{K-1} \boldsymbol{H}_k \tilde{\boldsymbol{a}}_{n-k}.$$

b) For each of the M possible transmitted waveforms, update the correlation values

$$c_{reg}(n', m) = \begin{cases} \bar{\boldsymbol{y}}_{n',n}^T \boldsymbol{h}_{n-n',m}, & n' = n \\ c_{reg}(n', m) + \bar{\boldsymbol{y}}_{n',n}^T \boldsymbol{h}_{n-n',m}, & \text{otherwise} \end{cases}$$

where $m = 0, \ldots, M-1$.

c) Search the maximum from the M correlation values

$$m' = \arg\max_{m}\{|c_{reg}(n', m)|\}.$$

For $n' \neq n - K + 1$, if it is the first round of iterations, update the tentative hard decisions

$$\tilde{a}_{n',m} = \begin{cases} \text{sign}\{c_{reg}(n', m')\}, & m = m' \\ 0, & \text{otherwise.} \end{cases}$$

For $n' = n - K + 1$, update the tentative hard decision $\tilde{\boldsymbol{a}}_{n-K+1}$, and obtain the decision of the transmitted symbol

$$\boldsymbol{a}_{n-K+1} = \tilde{\boldsymbol{a}}_{n-K+1}.$$

3) Repeat step 2 until the predetermined rounds of iterations are completed.

It is shown from the procedure that the proposed IIC algorithm only requires MK correlations to detect one symbol in one round of iterations. This implies that the multiplication operations are the same as that of Rake receivers. In sparse channels, we can use only significant multipath components in the IIC algorithm just as in selective Rake receivers. This is implemented by forcing the values of $h_m[i]$ in (4) to zeros except for the selected taps, which is much simpler than existing tap selection schemes [4], [5].

IV. Simulation Results

In this section, we will evaluate the performance of the IIC algorithm through computer simulations and compare it with the maximal ratio combining (MRC)-Rake receiver, LMMSE equalizer, and the MFB. The serial scheduling of the IIC algorithm with one to three rounds of iterations is considered.

IEEE802.15.4a channel model is used in our simulations [11], and the "CM2" environment, i.e., non-line-of-sight residential, is considered. The root-mean-square (RMS) delay spread is 19 ns, and we restrict the maximal multipath delay within 100 ns. In this model, the total energy of the channel in each realization is normalized.

In the simulation system, 4-PPM plus 2-PAM are employed. The symbol duration $T_s = 20$ ns, so that the ISI covers five symbols. There is only one pulse in each symbol, and the pulse width $T_p = 1$ ns and the sampling interval in the receiver $T_c = T_p$. Since 3 bits are modulated in each symbol, 150 Mbps date rate can be achieved. Perfect channel information are used in simulations.

Fig. 2 shows the BERs of the MFB, MRC-Rake, LMMSE equalizer, and IIC algorithm with $1 \sim 3$ rounds of iterations. LMMSE equalizer is implemented at chip-level with known channel statistics, and four groups of tap weights are computed for the 4-ary position modulation. It is shown from the figure that MRC-Rake exhibits apparent error floor because of the severe ISI, and the LMMSE equalizer has good performance at the cost of large complexity. The performance of the IIC algorithm with one round of iterations can achieve that of the LMMSE equalizer. With two rounds of iterations, the IIC algorithm outperforms the LMMSE equalizer. With more rounds of iterations, its performance does not significantly improve any more. That

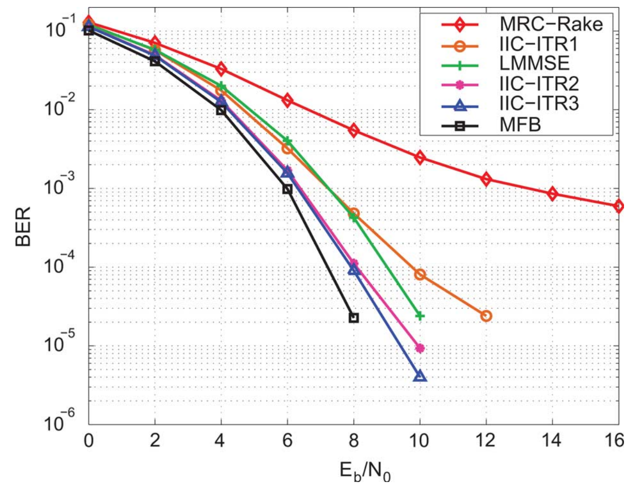

Fig. 2. BER $\sim E_b/N_0$, compare the performance of the MFB, MRC-Rake, LMMSE equalizer, and IIC algorithm with $1 \sim 3$ rounds of iterations.

means, in the given scenario, two rounds of iterations are enough to make the algorithm converge.

V. Conclusion

We propose a low-complexity equalization algorithm for UWB signals with various modulation and spreading schemes. By combining the multipath energy after iterative interference cancellation, it can outperform the optimal LMMSE equalizer with comparable complexity to Rake receivers. Its complexity can be further reduced by simply forcing insignificant channel coefficients to zeros in sparse channels.

References

[1] J. R. Barry, "Sequence detection and equalization for pulse-position modulation," in *Proc. IEEE Int. Conf. Communications (ICC'94)*, May 1994, vol. 3, pp. 1561–1565.

[2] M. D. Audeh, J. M. Kahn, and J. R. Barry, "Decision-feedback equalization of pulse-position modulation on measured nondirected indoor infrared channels," *IEEE Trans. Commun.*, vol. 47, no. 4, pp. 500–503, Apr. 1999.

[3] M. Eslami and X. Dong, "Rake-MMSE-equalizer performance for UWB," *IEEE Commun. Lett.*, vol. 9, pt. 6, pp. 502–504, Jun. 2005.

[4] S. Gezici, M. Chiang, H. V. Poor, and H. Kobayashi, "A genetic algorithm based finger selection scheme for UWB MMSE rake receivers," in *Proc. IEEE Int. Conf. Ultra-Wideband (ICU 2005)*, Zurich, Switzerland, Sep. 2005, pp. 164–169.

[5] Z. Lin, A. B. Premkumar, and A. S. Madhukumar, "Tap selection based multipath channel equalization for UWB systems," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP 2006)*, 2006, vol. 4.

[6] J. Boutros and G. Caire, "Iterative multiuser joint decoding: Unified framework and asymptotic analysis," *IEEE Trans. Inf. Theory*, vol. 48, no. 7, pp. 1772–1793, Jul. 2002.

[7] A. M. Chan and G. W. Wornell, "A class of block-iterative equalizers for intersymbol interference channels: Fixed channel results," *IEEE Trans. Commun.*, vol. 49, no. 11, pp. 1966–1976, Nov. 2001.

[8] B. Lu, G. Yue, X. Wang, and M. Madihian, "Factor-graph-based soft self-iterative equalizer for multipath channels," *EURASIP J. Wireless Commun. Netw.*, vol. 2005, no. 2, pp. 187–196, 2005.

[9] Q. Guo, L. Ping, and J. S. A. Commun, "LMMSE turbo equalization based on factor graphs," *IEEE J. Select. Areas Commun.*, vol. 26, no. 2, pp. 311–319, Feb. 2008.

[10] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[11] A. F. M. Molish *et al.*, IEEE 802.15.4a Channel Model—Final Report, 2005, Tech. Rep., Document IEEE 802.1504-0062-02-004a.