Abstract—The filter-based turbo equalization scheme has been proposed in several papers to avoid the prohibitive complexity imposed by the trellis-based turbo equalization. In the existing literature, the filter-based approach has been solely implemented by a linear MMSE filter, the coefficients of which are updated to minimize the mean-square error for every output symbol of the equalizer. A new turbo equalization algorithm is introduced in this paper. It has a lower computational complexity compared to most of the existing MMSE filter-based turbo equalization schemes. The complexity reduction is accomplished by deriving log-likelihood ratios (LLRs) directly from the output of an interference canceler, thus avoiding the MMSE filtering and its inherent matrix inversion for each symbol estimate. Numerical results show that the proposed scheme enables ISI-free transmission for some frequency selective channels.

I. INTRODUCTION AND PROBLEM FORMULATION

In a cellular mobile communications environment, multipath propagation causes dispersion of transmitted signals. The spread of time delay causes intersymbol interference (ISI) and degrades system performance. Therefore, equalization methods which can mitigate the effects of ISI must be employed. Douillard et al. proposed turbo equalization algorithm by extending the idea of iterative decoding that was used to decode turbo codes [1]. It is shown that turbo equalization significantly improve the performance over separate equalization and decoding. In its original form, turbo equalization [2] employed maximum a posteriori probability (MAP) for both equalization and decoding. For channels with large delay spreads and for large constellation sizes, it suffers from prohibitive computational complexity due to increasing number of trellis states. To tackle this problem, the filter-based approach has been proposed, e.g., in [3–7], where the equalizer is typically implemented by a linear transversal (FIR) filter, the coefficients of which are adjusted to minimize the mean-square error. It was shown that the performance of this approach is similar to the MAP-based receiver, while providing a significant reduction in the computational complexity. However, most filter-based turbo equalization algorithms require matrix inversion at a symbol rate in order to compute the MMSE filter coefficients, which is computationally intensive. The solution in [7] is to derive the filter coefficients adaptively using least mean square (LMS) algorithm. Compared to other adaptive algorithms, e.g., the recursive least square (RLS), the square root kalman (SRK), the LMS is relatively simple but suffers from performance loss and slow convergence. In this paper, we present a different approach to reduce the equalizer complexity without incurring a performance penalty in most cases. Its simplicity and effectiveness make it suitable for practical implementations.

The transmission system under study is shown in Fig. 1. The information sequence \( \{ b_n \} \) is convolutionally encoded into code bits \( \{ u_n \} \), which are subsequently interleaved and each block of two coded and interleaved bits \( \{ u_n^*[0], u_n^*[1] \} \) is mapped into one of the four QPSK symbols denoted as \( s_n = x_n + jy_n \), where \( x_n, y_n = \frac{1}{\sqrt{2}} \), and \( |s_n|^2 = 1 \). The symbols are transmitted over the ISI channel, which can be modeled by an equivalent baseband system where the concatenation of the transmit filter, the channel and the receive filter, is represented by a discrete-time transversal filter with finite-length impulse response \( h_n = \sum_{\ell=0}^{L-1} h_{\ell}\delta_{n-\ell} \) where \( L \) is the number of channel taps, and the complex channel coefficients \( h_{\ell} \) are assumed to remain constant during the transmission of one block of data. The channel output sequence \( \{ r_n \} \) can be expressed as

\[
r_n = \sum_{\ell=0}^{L-1} s_{n-\ell} h_{\ell} + w_n
\]

where \( w_n \) is complex additive white Gaussian noise (AWGN) with zero mean and variance \( N_0 \).

The task of the receiver is to detect the transmitted information bits \( \{ b_n \} \) given the received observation \( \{ r_n \} \). To this end, we need first to detect the transmitted QPSK symbols \( \{ s_n \} \) which are corrupted with ISI and AWGN noise. An equalizer is required to remove the detrimental effect of ISI. The estimated symbols are then converted to coded bits, which are subsequently deinterleaved and decoded to obtain an estimate of the information sequence. In this paper, we focus on the turbo equalization algorithm which combines equalizer and channel decoder in an iterative fashion. The existing techniques can be broadly classified into trellis (MAP) based and filter based approaches. For detailed descriptions of these turbo equalization algorithms, refer to [2–7]. Here, we introduce a simplified approach, which will be described next.

II. TURBO EQUALIZATION DESIGN

The proposed turbo equalization algorithm is illustrated in Fig. 2. First, we use some training sequence to derive
channel estimate $\mathbf{h} = [\hat{h}_0 \ \hat{h}_1 \ldots \hat{h}_{L-1}]^T$. In the meantime, some simple decision feedback equalizer (DFE) can be applied to obtain an initial estimate of transmitted symbols $\hat{s}_n = \hat{x}_n + j\hat{y}_n$. The channel estimate $\mathbf{h}$ and symbol estimates $\hat{s}_n$ are passed to the equalizer (the soft-input, soft-output (SISO) inner block in Fig. 2), which computes the log-likelihood ratio (LLR) values of $s_n$ by \( \lambda(s_n) = \lambda(x_n) + j\lambda(y_n) \). The LLR values of symbols are mapped into LLR values of coded bits \( \{\lambda(u'_n; O)\} \), which are deinterleaved to \( \lambda(u'_n; I) \). Based on the soft input \( \lambda(u'_n; I) \), a SISO outer channel decoder computes the LLR of each information bit $\lambda(h_n; O)$ and each coded bit $\lambda(u_n; O)$. The former is used to make decision on the transmitted information bit at the final iteration, and the latter is interleaved and passed through a bit-to-symbol converter (BSC) to derive a soft symbol estimate $\hat{\bar{s}}_n = \bar{x}_n + j\bar{y}_n$, which is used for equalization at the next iteration. We use the notations $\lambda(\cdot; I)$ and $\lambda(\cdot; O)$ at the input and output ports of a SISO device. Several SISO algorithms can be used to compute the LLRs at the channel decoder output. For the purpose of this study, we consider the use of the Log-MAP algorithm [8]. The equalization algorithm in the SISO inner block will be described in detail next.

Based on (1), the received signal can be written as

\[
\begin{align*}
\mathbf{r}_n &= h_0s_n + h_1s_{n-1} + \ldots + h_{L-1}s_{n-L+1} + w_n \\
\mathbf{r}_{n+1} &= h_0s_{n+1} + h_1s_n + \ldots + h_{L-1}s_{n-L+2} + w_{n+1} \\
&\vdots \\
\mathbf{r}_{n+L-1} &= h_0s_{n-L+1} + h_1s_{n-L} + \ldots + h_{L-1}s_n + w_{n+L-1}
\end{align*}
\]

Let us denote $\hat{s}_{n+i}$ as an estimate of $s_{n+i}$ from previous iteration. The ISI canceled version of the received signal can be written as

\[
\begin{align*}
\mathbf{r}'_n &= h_0s_n + (h_1s_{n-1} - \hat{h}_1\hat{s}_{n-1}) + \ldots \\
&\quad + (h_{L-1}s_{n-L+1} - \hat{h}_{L-1}\hat{s}_{n-L+1}) + w_n \\
\mathbf{r}'_{n+1} &= h_1s_n + (h_0s_{n+1} - \hat{h}_0\hat{s}_{n+1}) + \ldots \\
&\quad + (h_{L-1}s_{n-L} - \hat{h}_{L-1}\hat{s}_{n-L}) + w_{n+1} \\
&\vdots
\end{align*}
\]

The above formulas can be written in vector form as

\[
\mathbf{r}'_n = \begin{bmatrix} r'_0 \\ r'_1 \\ \vdots \\ r'_{n+L-1} \end{bmatrix} = \begin{bmatrix} h_0 \\ h_1 \\ \vdots \\ h_{L-1} \end{bmatrix} \mathbf{s}_n + \begin{bmatrix} v_0 \\ v_{n+1} \\ \vdots \\ v_{n+L-1} \end{bmatrix} = \mathbf{hs}_n + \mathbf{v}_n
\]

where $\mathbf{h} = [h_0 \ h_1 \ldots \ h_{L-1}]^T$ is the channel vector, and $\mathbf{v}_n$ stands for the combined noise and interference cancellation residual vector. This interference cancellation procedure is similar to the one proposed in [3, 4] where the LLRs are derived from the output of a MMSE filter applied on the interference canceled signal. Refer to the above references for a detailed description of this MMSE filter based scheme. Next, we shall demonstrate how the LLRs can be computed directly from the interference canceled signal without using a MMSE filter. The main idea is to approximate each element of $\mathbf{v}_n$ as a Gaussian random variable, i.e., $v_n \sim \mathcal{N}(0, \sigma_v)$, the conditional pdf of $r'_n$ is thus derived as

\[
f(r'_n|s_m) = \frac{1}{(\pi \sigma_v)^L} \exp \left( - \frac{\|r'_n - \mathbf{hs}_m\|^2}{\sigma_v} \right)
\]

Based on (5), the LLR of $x_n$ can be computed as

\[
\lambda(x_n) = \ln \frac{f(r'_n|x_n = 0)}{f(r'_n|x_n = 1)} = \ln \frac{f(r'_n|s_0) + f(r'_n|s_3)}{f(r'_n|s_1) + f(r'_n|s_2)} \\
\approx \ln \frac{\exp \left( - \frac{\|r'_n - \mathbf{hs}_+\|^2}{2\sigma_v} \right)}{\exp \left( - \frac{\|r'_n - \mathbf{hs}_-\|^2}{2\sigma_v} \right)}
\]

\[
= \frac{1}{\sigma_v} \left\{ \|r'_n - \mathbf{hs}_+\|^2 - \|r'_n - \mathbf{hs}_-\|^2 \right\}
\]

\[
= \frac{2}{\sigma_v} \text{Re} \{ (\mathbf{hs}_+)^* r'_n - (\mathbf{hs}_-)^* r'_n \}
\]

where $s_+$ denotes the QPSK symbol corresponding to $\max\{f(r'_n|s_0), f(r'_n|s_3)\}$, and $s_-$ denotes the QPSK symbol corresponding to $\max\{f(r'_n|s_1), f(r'_n|s_2)\}$ since the real part of the symbols $s_0, s_3$ corresponds to 0, and the real part
of the symbols $s_1, s_2$ corresponds to 1 as shown in Fig. 3. Dual maxima approximation [9] is used in (6) utilizing the fact that one term usually dominates each sum. Similarly,

$$\lambda(y_n) = \ln \frac{f(r_n^s|s_0) + f(r_n^s|s_1)}{f(r_n^s|s_2) + f(r_n^s|s_3)}$$

$$\approx \frac{2}{N_v} \Re \{((h^s_+)^*r_n^s - (h^s_-)^*r_n^s)\}$$

where $s_+$ denotes the QPSK symbol corresponding to $\max\{f(r_n^s|s_0), f(r_n^s|s_1)\}$, and $s_-$ denotes the QPSK symbol corresponding to $\max\{f(r_n^s|s_2), f(r_n^s|s_3)\}$ since the imaginary part of the symbols $s_0, s_1$ corresponds to 0, and the imaginary part of the symbols $s_2, s_3$ corresponds to 1. From (4), we know that

$$\mathbb{E}[||r_n^s||^2] = \mathbb{E}[(hs_n+v)(hs_n+v)^*]$$

$$= \mathbb{E}[||hs_n||^2] + \mathbb{E}[||v||^2]$$

$$= \mathbb{E}[||h||^2] + LN_v = P + LN_v$$

Therefore, the variance of noise plus residual interference can be derived statistically as

$$N_v = \mathbb{E}[||r_n^s||^2] - P \approx \frac{||r||^2 - P}{L}$$

(8)

where $P = \sum_{i=0}^{L-1} ||h||^2$ denotes the total received power from different paths, and $||r||^2$ is the energy of the vector $r_n^s$ averaged over the whole block of symbols. For QPSK modulated signals, the symbol LLR $\lambda(s_n) = \lambda(x_n) + j\lambda(y_n)$ to bits LLRs $\lambda(u_n^0[0]), \lambda(u_n^1[1])$ mapping rule is simply

$$\lambda(u_n^0[0]; O) = \lambda(x_n); \lambda(u_n^1[1]; O) = \lambda(y_n)$$

To avoid error propagation, we can replace the hard decisions $\{\hat{s}_{n+i}\}$ in (3) with their soft estimates $\{\hat{s}_{n+i} = \hat{x}_{n+i} + j\hat{y}_{n+i}\}$ computed as follows

$$\hat{x}_{n+i} = \mathbb{E}[x_{n+i}|r_{n+i}'] = \frac{1}{\sqrt{2}}P(x_{n+i} = \pm 1\sqrt{2}r_{n+i}')$$

$$+ \frac{-1}{\sqrt{2}}P(x_{n+i} = -1\sqrt{2}r_{n+i}') = \tanh[\lambda(x_{n+i})/2]/\sqrt{2}$$

$$\hat{y}_{n+i} = \mathbb{E}[y_{n+i}|r_{n+i}'] = \frac{-1}{\sqrt{2}}P(y_{n+i} = -1\sqrt{2}r_{n+i}')$$

$$+ \frac{-1}{\sqrt{2}}P(y_{n+i} = +1\sqrt{2}r_{n+i}') = \tanh[\lambda(y_{n+i})/2]/\sqrt{2}$$

The principle is that the interference cancellation using hard decisions tends to propagate errors and increase the interference with incorrect decision feedback; while with soft cancellation, an erroneously estimated symbol usually has a small LLR, and hence the soft estimate of this symbol is small and does not make much contribution to the feedback, therefore error propagation is avoided.

To find out the theoretical performance potential of the proposed approach, we analyze the performance bound that can be achieved by our turbo equalization scheme. It is derived based on the assumption that the interference cancellation and channel estimation are perfect and equalizer output is ISI-free. That would be the ideal situation leading to best achievable performance. In the derivation of the theoretical lower bound, we use the facts that the Max-Log-MAP algorithm provides exactly the same hard decisions as the Viterbi algorithm [8], and the Max-Log-MAP is an approximation of the Log-MAP, i.e., it does not include a correction term. Let us reform the LLR expression in (7) as

$$\lambda(x_n) = \frac{2}{N_v} \Re \{(hs_+^s)^*r_n^s - (hs_-^s)^*r_n^s\}$$

$$= \frac{2}{N_v} \Re \{s_+^s(h^*r_n^s) - s_-^s(h^*r_n^s)\}$$

$$= \frac{2}{N_v} \Re \{s_+^s z_n - s_-^s z_n\}$$

(9)

where

$$z_n = h^*r_n^s = h^*(hs_n + v_n) = ||h||^2 s_n + h^*v_n$$

$$= Ps_n + \eta = Px_n + \eta + j(\eta y_n + \eta Q)$$

In the case of perfect cancellation, cancellation residual vanishes, and $\eta$ only contains the noise, it is a Gaussian random variable with zero mean and variance $N_\eta = ||h||^2 N_0 = PN_0$. Its real and imaginary part $\eta_L, \eta_Q$ are independent zero mean Gaussian random variables with variance $PN_0/2$. Since the modulation/demodulation of a QPSK system is equivalent to two independent (phase-quadrature) BPSK systems [10], the hard decisions produced by this coded QPSK system with $\lambda(x_n)$ (expressed by (9)) at the input of its Log-MAP decoder provides the ones produced by a coded BPSK system with $r_1 = \frac{P}{\sqrt{2}} c_1 + u_1$ at the input of its Viterbi decoder, where $c_1 = \pm 1$, with +1 corresponding to the binary digit 0 and −1 corresponding to 1. The noise $u_1$ has the same distribution and variance as $\eta_L$ and $\eta_Q$, i.e., $u_1 \sim \mathcal{N}(0, N_u)$. $N_u = PN_0/2$. The conditional PDF is thus

$$p(r_i|c_i) = \frac{1}{\pi N_u} \exp \left(\frac{|r_i - \frac{P}{\sqrt{2}} c_i|^2}{N_u}\right)$$

Assuming the maximum likelihood decoding, after neglecting the common terms, the metrics corresponding to the all-zero path and the first error event path can be expressed as

$$CM^{(0)} = \sum_{i=1}^{d} (+1) \left(\frac{P}{\sqrt{2}} + u_i\right)$$

$$CM^{(1)} = \sum_{i=1}^{d} (-1) \left(\frac{P}{\sqrt{2}} + u_i\right)$$
where the index $l$ runs over the set of $d$ bits since the coded bits in the two paths are identical except in $d$ positions, and the common terms due to the identical bits are left out in the above metrics. The pairwise error probability $P_2(d)$, which is defined as the probability of decoding in favor of a codeword with weight $d$ when all-zero codeword is transmitted is computed as the probability that the error path has better metric than the all-zero path, i.e.,

$$P_2(d) = P_r(CM^{(1)} > CM^{(0)})$$

$$= P_r \left[ \sum_{l=1}^{d} (-1) \left( \frac{P}{\sqrt{2}} + u_l \right) > \sum_{l=1}^{d} (+1) \left( \frac{P}{\sqrt{2}} + u_l \right) \right]$$

$$= P_r \left[ 2 \sum_{l=1}^{d} u_l < -2dP/\sqrt{2} \right] = P_r \left[ \eta < -dP/\sqrt{2} \right]$$

$$= Q \left( \frac{dP}{\sqrt{2} \sqrt{dPN_0}} \right) = Q \left( \sqrt{\frac{dP}{N_0}} \right)$$

where $\eta = \sum_{l=1}^{d} u_l$, and $N_0 = dN_u = dP/N_0/2$. The bit error probability $P_b$ is upper bounded by [10]

$$P_b \leq \sum_{d=d_{free}}^{\infty} c_d P_2(d) = \sum_{d=d_{free}}^{\infty} c_d Q \left( \sqrt{\frac{dP}{N_0}} \right)$$

$$\approx \frac{N_t}{d_{free}} c_d Q \left( \sqrt{\frac{dP}{N_0}} \right)$$

(10)

where $d_{free}$ is the free distance of the code, and $c_d$ is the sum of the information weights of all error paths with weight $d$, which can be computed from the transfer function of the code. However, $c_d$ is also tabulated, e.g., in [11] for most good codes of practical interest, including the one we have used in our simulations. In equation (10), $N_t$ denotes the truncation length. The pairwise error probability $P_2(d)$ is rapidly decreasing with $d$. Hence, for a sufficient high value of $d > N_t$, the terms $c_d P_2(d)$ will be negligible, and we can truncate the sum without compromising the bound. The above equation works for static channels. For non-static channels, we have to average the error probability over the distributions of channel gains of different paths.

### III. Numerical Results

The proposed scheme is compared numerically with some existing algorithms in this section to demonstrate its efficiency. In particular, we look at the comparison between our algorithm and the MMSE filter based turbo equalization proposed by Tuchler, et. al. in [5] and the adaptive turbo equalization introduced by Laot, et. al. in [7]. It should be noted that the MMSE scheme presented by Wang and Poor in [3] is identical to Tuchler’s scheme in a single-user case. The original algorithms were derived for the BPSK modulated system, have to be modified to suit the QPSK constellation considered in this paper. In the simulations, we employ a rate 1/3 Maximum Free Distance convolutional code with constraint length 5 and generator polynomials $(25, 33, 37)$ in octal form. During each Monte-Carlo run, the block size is set to 5000 information bits followed by 4 tail bits to terminate the trellis, which corresponds to $5004 \times 3 = 15012$ coded bits. They are interleaved by an $108 \times 139$ block interleaver and transmitted over a ISI channel. The noise variance $N_0$ and path delays are assumed to be known to the receiver. For the initial equalization, we use a decision feedback equalizer (DFE) with 5 feedforward, 3 feedback taps. It uses 200 pilot symbols for training the equalizer coefficients. In the meantime, the modified maximum likelihood algorithm presented in [12] is used for channel estimation during training period. The channel estimates are averaged over many estimated samples to reduce the noise effect.

Both time-varying and static channels are considered for our simulations. In the former case, we select the SUI-3 channel in broadband fixed wireless access (FWA) systems introduced in [13]. To simplify the simulations, we assume the transmitted data rate is 4Mbps so that the channel is modeled as a 3-tap FIR filter with adjacent taps spaced equally at symbol duration. The channel coefficients vary from one data block to another, however, they are assumed to remain constant during the transmission of one block of data. It is therefore a quasi-static channel. Fig. 4 shows the performance of the proposed turbo equalization algorithm for the FWA SUI-3 channel. It takes only 4 stages for the algorithm to converge. Fast convergence shortens the latency of the iterative process and reduces the receiver complexity, it is therefore a desired feature. Compared to the initial stage with one time DFE equalization and Log-MAP decoding, the subsequent turbo equalization stages achieve much better performance. The gain by applying turbo equalization increases as SNR increases.

The performance comparison between the proposed scheme and the MMSE-based schemes for the SUI-3 channel is given in Fig. 5. For Laot’s adaptive algorithm [7], the step size $\mu$ is set to 0.006 during the training period and 0.002 during the tracking period. Results show that the proposed scheme yields almost the identical performance to Tuchler’s MMSE scheme, while achieves a gain of up to 0.7dB compared to Laot’s adaptive scheme after the system reaches convergence. We noticed from the experiments that replacing $N_0$ (derived statistically with equation (8)) by $N_0$ (the noise variance) does not make much difference, which means that the interference is effectively canceled with the proposed turbo equalization scheme so that the cancellation residual can be neglected.

We also tested two static channels with impulse response $h_1[n] = (0.632456 + 0.632456j)\delta[n] + (0.273861 + 0.273861j)\delta[n-1] + (0.158114 + 0.158114j)\delta[n-2]$, and $h_2[n] = 0.407\delta[n] + 0.815\delta[n-1] + 0.407\delta[n-2]$, respectively. The former has a strong line-of-sight component (the first tap is stronger than the other taps), while the latter is a much harsher channel and introduces more frequency-selectivity. The results of different turbo equalizers for first channel are shown in Fig. 6. Just as previous case, the performance of the proposed scheme is comparable to that of the Tuchler’s scheme. Both outperform the adaptive scheme by 0.2–0.4 dB.

In Fig. 7, the proposed turbo equalization scheme is compared with the performance bound for the first static channel. The simulated bound is obtained by transmitting QPSK symbols over an AWGN channel, and applying the same convolutional code to the information bits prior to QPSK modulation. The theoretical bound is given by (10).
TABLE I
Comparison of complexity for one symbol estimate in one iteration for the algorithms considered.

<table>
<thead>
<tr>
<th>operations</th>
<th>multiplication</th>
<th>division</th>
<th>addition/subtraction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposed scheme</td>
<td>$L^2 + 3L + 2$</td>
<td>2</td>
<td>$L^2 - L + 2$</td>
</tr>
<tr>
<td>Tuchler’s MMSE</td>
<td>$8L^3 - 4L^2 + 3L + 8$</td>
<td>$2L^2 + 2$</td>
<td>$8L^3 - 3L^2 + 4L$</td>
</tr>
<tr>
<td>Laot’s scheme</td>
<td>$6L + 4$</td>
<td>4</td>
<td>$6L + 4$</td>
</tr>
</tbody>
</table>

The plot shows fairly close agreement between the theoretical curve and the simulated curve when $E_b/N_0 > 3.5$dB. The small discrepancy is due to the approximation assumptions made in the derivation of the analytical performance. This means that the derived theoretical bound is a tight bound, it provides good insight into the asymptotical behavior of the proposed scheme. Fig. 7 also shows that the performance of the proposed scheme is very close to the lower bound after it reaches convergence. This indicates that the effect of ISI can be effectively removed by the proposed turbo equalization algorithm, and ISI-free transmission is approached for this static channel.

The results for the second channel are shown in Fig. 8. Unlike the previous case, all the schemes are far from the performance bound. ISI-free transmission cannot be fulfilled for this harsh channel. Tuchler’s MMSE scheme actually has better performance, and we observe a gain up to 1dB by performing MMSE filtering in this particular case.

Table I shows the required number of complex multiplications, divisions, and additions/subtractions for each QPSK symbol estimate by different turbo equalization schemes. $L$ is the number of channel taps. The figures for the two MMSE filter based algorithms are based on the modified versions tailored to the QPSK modulation. One can see from the table that Laot’s scheme has the lowest complexity, which is linear with $L$. However, it has the worst performance as shown by the numerical results presented earlier. Compared to Tuchler’s schemes, the proposed scheme reduces the complexity from $O(L^3)$ to $O(L^2)$. Furthermore, its complexity is comparable to Laot’s scheme when the delay spread is short, i.e., for small values of $L$. Note that two approximate implementations of Tuchler’s scheme was given in [5]. They have lower complexity than the original algorithm, but also lead to some degree of performance loss. Here, we only use its original implementation for performance and complexity comparison.

IV. Conclusions

In this paper, we introduced a new approach to jointly equalizing and decoding of coded data over ISI channels. Numerical comparison with the existing MMSE filter-based
turbo equalization schemes indicates that in most cases, the MMSE filtering is not necessary, we can derive the LLR values of the transmitted bits directly from the output of an interference canceler, leading to comparable performance and reduced complexity. For some frequency selective channels, the proposed algorithm almost completely removes the effect of ISI and approaches the coded Gaussian channel performance. Although the algorithm was presented for QPSK modulated system, however, it can be easily extended to higher order PSK and QAM constellations. It provides a feasible solution for practical implementations, especially for the channels with large delay spreads and systems with high constellation sizes.

REFERENCES


