Low Complexity Turbo-Equalization: A Clustering Approach

Kyeongyeon Kim, *Member, IEEE*, Jun Won Choi, *Member, IEEE*,
Suleyman S. Kozat, *Senior Member, IEEE*, and Andrew C. Singer, *Fellow, IEEE*

***Abstract*—We introduce a low complexity approach to iterative equalization and decoding, or "turbo equalization", which uses clustered models to better match the nonlinear relationship that exists between likelihood information from a channel decoder and the symbol estimates that arise in soft-input channel equalization. The introduced clustered turbo equalizer uses piecewise linear models to capture the nonlinear dependency of the linear minimum mean square error (MMSE) symbol estimate on the symbol likelihoods produced by the channel decoder and maintains a computational complexity that is only linear in the channel memory. By partitioning the space of likelihood information from the decoder based on either hard or soft clustering and using locally-linear adaptive equalizers within each clustered region, the performance gap between the linear MMSE turbo equalizers and low-complexity least mean square (LMS)-based linear turbo equalizers can be narrowed.**

***Index Terms*—Turbo equalization, piecewise linear modelling, hard clustering, soft clustering.**

I. Introduction

Digital communication receivers typically employ a symbol detector to estimate the transmitted channel symbols and a channel decoder to decode the error correcting code that was used to protect the information bits before transmission. There has been great interest in enabling interaction between the symbol estimation task and the channel decoding task, which is often termed "turbo equalization" for digital communication over channels with inter-symbol-interference (ISI). This interest is due to the dramatic performance gains that can be obtained with modest complexity [1] over performing these tasks separately. Turbo equalization methods employing maximum-a-posteriori probability (MAP) detectors demonstrate excellent bit-error-rate (BER) performance. However, their computational complexity often renders their application impractical [1]. As an alternative, linear MMSE-based methods offer comparable performance to MAP-based approaches, with significantly reduced complexity [1] compared with the exponential complexity of the MAP-based approach. However, MMSE-based approaches still require quadratic computational complexity in the channel length per output symbol and require adequate channel knowledge or estimation. Although MMSE-based equalizers can be implemented in the frequency domain with logarithmic complexity, this is only when constant prior information is used over all samples [1]. To further reduce computational complexity and improve efficacy over unknown or time-varying channels, "direct" adaptive least mean square (LMS) linear equalizers are often used, employing only linear complexity [2] in the regressor vector length, which is often on the order of the channel delay spread. In addition, such direct adaptive LMS equalizers are actually preferred in terms of implementation perspective since matrix inversion is not required in calculating their coefficients.

While these direct-adaptive methods may reduce computational complexity and can be shown to converge to their Wiener (MMSE) solution under stationary environments, they usually deliver inferior performance compared to linear MMSE-based methods (except on isolated examples [1]). A primary reason for this performance loss is that the Wiener solution is not time-adaptive, but rather corresponds to the solution of the "stationarized problem" where the likelihood information from the decoder (which is by definition a sample-by-sample probability distribution over the transmitted data sequence and hence non-stationary) is replaced by a suitable time-averaged quantity [2]. On the other hand, both the linear MMSE and MAP-based turbo equalizer (TEQ) consider the log-likelihood ratio (LLR) sequence as time-varying *a priori* statistics over the transmitted symbols. This LLR information is used to construct the linear MMSE equalizer, which depends nonlinearly and in a time dependent manner on the LLR sequence.

To reduce the performance gap between direct adaptive LMS linear TEQ (DA-LMS-TEQ) and linear MMSE-TEQ, we introduce an adaptive approach that can readily follow the time variation of the soft decision data and respect the nonlinear dependence of the MMSE symbol estimates on this LLR sequence while maintaining the low computational complexity of the LMS-adaptive approach. Specifically, we introduce an adaptive, piecewise linear equalizer that partitions the space of LLR vectors from the channel decoder into sets, within which, low complexity LMS-adaptive TEQs can be used. We use a deterministic annealing (DA) algorithm [3] for soft clustering the symbol-by-symbol variances of the transmitted symbols calculated from the soft information. These variances are partitioned into K regions with a partial membership according to their assigned association probabilities [3]. For hard clustering, the association probabilities are either 1 or 0. In each cluster, a local linear filter is updated where the contribution to the local update is weighted by the association probabilities [3]. In addition, we also quantify the mean square error (MSE) of the

Manuscript received January 9, 2014; revised March 28, 2014; accepted March 28, 2014. Date of publication April 8, 2014; date of current version June 6, 2014. This research was funded by the research fund of Hanyang Univeristy (HY-2013). The associate editor coordinating the review of this paper and approving it for publication was K. K. Kuchi.

K. Kim is with Samsung Electronics, Suwon 443-743, Korea.

J. W. Choi is with Hanyang University, Seoul 133-791, Korea.

S. S. Kozat is with the Koc University, Istanbul 34450, Turkey.

A. C. Singer is with the University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA.

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/LCOMM.2014.2316172

approach employing hard clustering and show that it converges to the MSE of the linear MMSE equalizer as the number of regions and the data length increase. While there have been a number of techniques on low complexity MMSE-TEQ [4], [5], the approach for improving direct adaptive TEQs has not been reported yet. In our simulations, we observe that the clustered TEQ significantly improves performance over traditional LMS-adaptive linear equalizers without any significant computational complexity increase.

II. System Description

We consider the framework for linear turbo equalization system described in [1]. Information bits at the transmitter are encoded using forward error correction, interleaved in time, mapped to channel symbols. Any error correction code supporting soft-input and soft-output decoding can be used, including convolutional code, turbo code, or low density parity code (LDPC) code [6]. The channel symbols are transmitted through an ISI channel with impulse response h_l, of length L, $l = 0, \ldots, L-1$. The received signal $y[n]$ is given by $y[n] = \sum_{l=0}^{L-1} h_l x[n-l] + w[n]$, where h_l is assumed to be time invariant for notational ease and $w[n]$ is the additive noise. In the receiver, the decoder and equalizer pass extrinsic log-likelihood ratio information on the information bits to iteratively improve detection and decoding. The decoder computes the extrinsic information L_e^D which are fed back to the equalizer and the equalizer utilizes the *a priori* information L_a^E, which is the interleaved data of L_e^D [1]. A linear equalizer with a feedforward filter $\boldsymbol{f}$ and a feedback filter $\boldsymbol{b}$ is applied to the $(N_1 + N_2 + 1)$ size of observation vector $\boldsymbol{y}[n] \triangleq [y[n-N_2], \ldots, y[n+N_1]]^T$, producing an estimate of the transmitted signal at time n

$$\hat{x}[n] = \boldsymbol{f}^H[n]\boldsymbol{y}[n] - \boldsymbol{b}^H[n]\bar{\boldsymbol{x}}_{-n}[n] \tag{1}$$

where $\bar{\boldsymbol{x}}_{-n}[n] \triangleq [\bar{x}[n-N_2-L+1], \ldots, \bar{x}[n-1], \bar{x}[n+1], \ldots, \bar{x}[n+N_1]]^T$, and N_1 and N_2 are the size of noncausal and causal parts of the equalizer. Note that the length of the feedforward part and that of the feedback part are $N+1$ and $N+L$ with $N \triangleq N_1 + N_2$. The mean symbol values are calculated using the *a priori* information L_a^E provided by the SISO decoder, i.e., $\bar{x}[n] = E[x[n] : \{L_a^E\}]$. If we assumed BPSK signaling for notational simplicity, $\bar{x}[n] = \tanh(L_a^E(x[n])/2)$, where $L_a^E(x[n])$ is the *a priori* LLR associated with $x[n]$. Note that $E[|x[n]|^2 : \{L_a^E\}] = 1$. If a linear MMSE equalizer is used in (1), we get [1]

$$\boldsymbol{f}[n] = \left(\boldsymbol{H}_{-0}\boldsymbol{V}[n]\boldsymbol{H}_{-0}^H + \boldsymbol{s}\boldsymbol{s}^H + \sigma_w^2\boldsymbol{I}\right)^{-1}\boldsymbol{s},\ \boldsymbol{b}[n] = \boldsymbol{H}_{-0}^H\boldsymbol{f}[n] \tag{2}$$

where $\boldsymbol{H}$ is the channel convolution matrix of size $N \times (N+L-1)$ constructed using $\{h_l\}_{l=0}^{L-1}$, $\boldsymbol{s}$ is the (N_2+L)th column of $\boldsymbol{H}$, $\boldsymbol{H}_{-0}$ is the matrix where the (N_2+L)th column of $\boldsymbol{H}$ is eliminated, $\boldsymbol{V}[n] = \mathrm{diag}(\boldsymbol{v}[n])$, where $\boldsymbol{v}[n] \triangleq [v[n-N_2-L+1], \ldots, v[n-1], v[n+1], \ldots, v[n+N_1]])$, $v[n] = 1 - \|\bar{x}[n]\|^2$, and σ_w^2 is the additive noise variance assuming fixed transmit signal power of one.

Remark 1: The linear MMSE equalizer in (2) is time varying due to the symbol-by-symbol variation of the soft input variance, $\boldsymbol{V}[n]$, even if h_l is time invariant. The linear MMSE equalizer is a nonlinear function of $\boldsymbol{V}[n]$. If h_l is also time varying, then (2) could be readily updated by including this time variation.

Unlike the linear MMSE equalizer, "direct" adaptive linear TEQs use adaptive updates (e.g., using LMS or recursive least square (RLS)), for direct estimation of the transmitted symbols by processing the received signal and LLR information without the need for channel estimation [2]. In general, these approaches use only the mean vector $\bar{\boldsymbol{x}}_{-n}[n]$ as feedback, i.e., soft decision data are not considered as *a priori* probabilities, where each component of $\bar{\boldsymbol{x}}_{-n}[n]$ is taken as a random variable with zero mean and variance $\sigma_{\bar{x}}^2$. As an example, if one uses the normalized least mean square (NLMS) direct adaptive linear equalizer, we have the update [7]

$$e[n] = x_d[n] - \boldsymbol{w}^H[n]\boldsymbol{u}[n],$$
$$\boldsymbol{w}[n+1] = \boldsymbol{w}[n] + \mu e^*[n]\boldsymbol{u}[n]/\|\boldsymbol{u}[n]\|^2$$

where $x_d[n]$ is the desired symbol for equalization, $\boldsymbol{w}[n+1] = [\boldsymbol{f}^H[n+1], -\boldsymbol{b}^H[n+1]]^H$, $\boldsymbol{u}[n] = [\boldsymbol{y}^H[n], \bar{\boldsymbol{x}}_{-n}^H]^H$ and μ is the step size. When training the coefficients of the equalizer, we use a known reference symbol (called a pilot symbol) for $x_d[n]$. When detecting a transmitted symbol in decision-directed mode, we use a tentative decision on $x[n]$ such as hard symbol decision or a soft symbol estimate. Under this stationarity assumption on $\bar{x}$ and LLRs, the direct adaptive filter using $\bar{\boldsymbol{x}}_{-n}$ converges to the MSE optimal Wiener (stationary MMSE) solution [7]

$$\boldsymbol{f} = \left(\left(1 - \sigma_{\bar{x}}^2\right)\boldsymbol{H}_{-0}\boldsymbol{H}_{-0}^H + \boldsymbol{s}\boldsymbol{s}^H + \sigma_w^2\boldsymbol{I}\right)^{-1}\boldsymbol{s} \tag{3}$$

and $\boldsymbol{b} = \boldsymbol{H}_{-0}^H\boldsymbol{f}$, assuming zero variance at convergence [1]. Recall that $\sigma_{\bar{x}}^2$ represents the variance of $\bar{x}[n]$. The resulting filter in (3) at convergence is time invariant and is not identical to (2) with time averaged soft information. The linear MMSE in (2) requires $O((N+L)^2)$ computations per output, however, (3) requires only $O(N+L)$ [1]. In addition, since (3) is not time varying and implicitly assumes that the soft information is stationary, there is a large performance gap between the equalizers formulated in (2) and (3) for some cases [1]. We seek to reduce this performance gap between the direct adaptive methods with respect to the linear MMSE approach by capturing the nonlinear dependence of the MMSE solution on the soft-information without requiring the associated computational complexity of (2).

III. Adaptive Turbo Equalization Using Hard or Soft Clustered Linear Models

We propose to use adaptive local linear filters to model the nonlinear dependency of the linear MMSE equalizer on the variance computed from the soft information generated by the SISO decoder in (2). When a vector of the variances (or equivalently LLR values) of the direct adaptive TEQ are localized, the time-averaged statistics used in (3) approaches the values computed from the LLR values in (2). To reduce the gap, we partition the space of variances, $\boldsymbol{V}[n]$, in (2) into a set of regions, where in each region, a single direct adaptive linear equalizer is used. Note that if the partitioned regions are dense enough such that $\boldsymbol{V}[n]$ is assumed to be constant in each region, then the linear MMSE-TEQ in (2) is a time invariant linear equalizer (assuming h_l is time invariant) in each region. Hence,

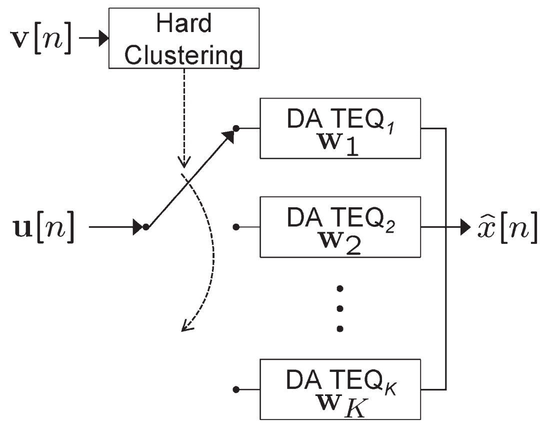

Fig. 1. Block diagram of the hard clustering-based TEQ.

assuming the regions are dense enough and each adaptive linear equalizer assigned to these regions succesfully converge, then we achieve the performance of the linear MMSE-TEQ by using a direct "piecewise-linear" TEQ. As a result, we can retain the computational efficiency of the direct adaptive methods, while capturing the nonlinear dependence (and hence sample-by-sample variation) of the linear MMSE optimal TEQ without requiring *a priori* channel knowledge or estimation.

A. Adaptive Turbo Equalization Based on Hard Clustering

Fig. 1 depicts the block diagram of the hard clustering-based TEQ. First, the hard clustering algorithm such as K-means algorithm (LBG VQ) [3] partitions the space of $\boldsymbol{v}[n]$ into K regions $\{\mathcal{R}_k\}$ with the corresponding centroids $\{\tilde{\boldsymbol{v}}_k\}$ where $k = 1, \ldots, K$. In the LBG VQ algorithm, the centroids and the corresponding regions are iteratively calculated from $\tilde{\boldsymbol{v}}_k \triangleq \sum_{n,\boldsymbol{v}[n]\in\mathcal{R}_k} \boldsymbol{v}[n]/(\sum_{n,\boldsymbol{v}[n]\in\mathcal{R}_k} 1)$ and $\mathcal{R}_k \triangleq \{\boldsymbol{v} : \|\boldsymbol{v} - \tilde{\boldsymbol{v}}_k\| \leq \|\boldsymbol{v} - \tilde{\boldsymbol{v}}_i\|, i = 1, \ldots, K, i \neq k\}$ [3]. In hard clustering-based TEQ, K direct adaptive TEQs are employed and each locally adaptive TEQ is associated with each of K clusters. For each output sample, one of the K local filters is updated by

$$e_k[n] = x_d[n] - \boldsymbol{w}_k^H[n-1]\boldsymbol{u}[n] \quad (4)$$

$$\boldsymbol{w}_k[n] = \boldsymbol{w}_k[n-1] + \mu e_k^*[n]\boldsymbol{u}[n]/\left(\|\boldsymbol{u}[n]\|^2\right) \quad (5)$$

$$\hat{x}[n] = \boldsymbol{w}_k^H[n]\boldsymbol{u}[n], \quad (6)$$

where $k = \arg\min_i \|\boldsymbol{v}[n] - \tilde{\boldsymbol{v}}_i\|$, $\boldsymbol{w}_k[n] = [\boldsymbol{f}_k^H[n], -\boldsymbol{b}_k^H[n]]^H$, and $\boldsymbol{u}[n] = [\boldsymbol{y}^H[n], \bar{\boldsymbol{x}}_{-n}^H]^H$.

Note that the coefficients of the other $K-1$ local filters are frozen, i.e.,. $\boldsymbol{w}_i[n] = \boldsymbol{w}_i[n-1]$, $i = 1, \ldots, K$, $i \neq k$. For the adaptive algorithms to converge in each of these regions, we put a constraint on cluster-size such that each cluster contains at least $N_{\min}$ symbols (the minimum required data length for suitable convergence). Hence, a cluster with less than $N_{\min}$ symbols is merged into clusters of larger size. TEQ operations are conducted on a packet by packet basis where each packet consists of L_T training symbols and L_D data symbols. During the training period, it is necessary to feed specially designed soft symbols into the feedback part for separate training of each local filter. To make sure that each DA-LMS-TEQ converges to the form associated with each cluster during the training period, those "soft training symbols" are computed from the centroid of each cluster, i.e., we replace $\bar{x}[n]$ by $\sqrt{1-\tilde{v}[n]}x_T[n]$ in (4)–(6), where $x_T[n]$ and $\tilde{v}[n]$ are the training symbol and the corresponding centroid at time n. We use $x_T[n]$ in (4) for the desired symbol $x_d[n]$. Note that when the equalizer operates in decision directed mode, the hard quantized value of $\hat{x}[n]$ is used for $x_d[n]$ in (4).

As partitioning of the variance space is denser such that $\boldsymbol{v}[n] \approx \tilde{\boldsymbol{v}}_k$ for all regions, then the adaptive filter in the kth region converges to $\mathbf{g}_k = (\boldsymbol{H}_{-0}\tilde{\boldsymbol{V}}_k\boldsymbol{H}_{-0}^H + \boldsymbol{s}\boldsymbol{s}^H + \sigma_w^2\boldsymbol{I})^{-1}\boldsymbol{s}$, $\tilde{\boldsymbol{V}}_k = \mathrm{diag}(\tilde{\boldsymbol{v}}_k)$. The difference between the MSE of the converged filter, $\mathcal{M}_c$ and the MSE of the linear MMSE equalizer, $\mathcal{M}_m$ is given by [1]

$$\mathcal{M}_c - \mathcal{M}_m = \boldsymbol{g}_k^H\boldsymbol{H}_{-0}\left(\boldsymbol{V}[n] - \tilde{\boldsymbol{V}}_k\right)\boldsymbol{H}_{-0}^H\boldsymbol{g}_k + \left(1 - \boldsymbol{g}_k^H\boldsymbol{s}\right) - \left(1 - \boldsymbol{f}[n]^H\boldsymbol{s}\right). \quad (7)$$

By defining $\boldsymbol{A} = (\boldsymbol{H}_{-0}\tilde{\boldsymbol{V}}\boldsymbol{H}_{-0}^H + \boldsymbol{s}\boldsymbol{s}^H + \sigma_w^2\boldsymbol{I})$, $\boldsymbol{B} = \boldsymbol{A} + \boldsymbol{H}_{-0}\boldsymbol{E}\boldsymbol{H}_{-0}^H$ and $\boldsymbol{E} = \boldsymbol{V}[n] - \tilde{\boldsymbol{V}}$, the difference (7) yields

$$\begin{aligned}
&\mathcal{M}_c - \mathcal{M}_m \\
&= \boldsymbol{s}^H\boldsymbol{A}^{-1}\boldsymbol{H}_{-0}\boldsymbol{E}\boldsymbol{H}_{-0}^H\boldsymbol{A}^{-1}\boldsymbol{s} + \boldsymbol{s}^H(\boldsymbol{B}^{-1} - \boldsymbol{A}^{-1})\boldsymbol{s} \\
&= \boldsymbol{s}^H\boldsymbol{A}^{-1}\boldsymbol{H}_{-0}\boldsymbol{E}\boldsymbol{H}_{-0}^H\boldsymbol{B}^{-1}\boldsymbol{H}_{-0}\boldsymbol{E}\boldsymbol{H}_{-0}^H\boldsymbol{A}^{-1}\boldsymbol{s} \quad (8) \\
&\leq \lambda_{\max}\left(\boldsymbol{H}_{-0}\boldsymbol{E}\boldsymbol{H}_{-0}^H\boldsymbol{B}^{-1}\boldsymbol{H}_{-0}\boldsymbol{E}\boldsymbol{H}_{-0}^H\right)\boldsymbol{s}^H\boldsymbol{A}^{-2}\boldsymbol{s} \quad (9) \\
&\leq e_{\max}^2\lambda_{\max}^2\left(\boldsymbol{H}_{-0}\boldsymbol{H}_{-0}^H\right)\lambda_{\min}(\boldsymbol{B})\boldsymbol{s}^H\boldsymbol{A}^{-2}\boldsymbol{s}, \quad (10)
\end{aligned}$$

where $e_{\max}$ is the maximum element of the error diagonal matrix $\boldsymbol{E}$. Here, (8) follows from $(\boldsymbol{B}^{-1} - \boldsymbol{C}^{-1}) = \boldsymbol{B}^{-1}(\boldsymbol{C} - \boldsymbol{B})\boldsymbol{C}^{-1}$, (9) follows from $\mathrm{tr}(\boldsymbol{C}\boldsymbol{D}) = \mathrm{tr}(\boldsymbol{D}\boldsymbol{C})$ and $\mathrm{tr}(\boldsymbol{C}\boldsymbol{D}) \leq \lambda_{\max}(\boldsymbol{C})\mathrm{tr}(\boldsymbol{D})$, and the last line follows from $\lambda_{\max}(\boldsymbol{C}\boldsymbol{D}) \leq \lambda_{\max}(\boldsymbol{C})\lambda_{\max}(\boldsymbol{D})$. Since $\lambda_{\min}(\boldsymbol{B}) \geq \sigma_w^2$ and $\lambda_{\max}(\boldsymbol{H}_{-0}\boldsymbol{H}_{-0}^H) \leq \lambda_{\max}(\boldsymbol{H}\boldsymbol{H}^H) \leq (\sum_m |h_m|)$ for the Toeplitz matrix $\boldsymbol{H}$, the MSE difference in (7) is bounded by $Ce_{\max}^2$ for some $C < \infty$. Hence, the MSE of the hard clustered linear equalizer converges to the MSE of the linear MMSE equalizer as the number of the regions increase provided there is enough data for training.

B. Adaptive Turbo Equalization Based on Soft Clustering

In addition to partitioning the space of $\boldsymbol{v}[n]$, the soft clustering algorithms produce the probabilistic weights of each cluster region [3]. The deterministic annealing (DA) algorithm [3] can be used to find the centroid $\tilde{\boldsymbol{v}}_k$ and association probability $P(\cdot|\tilde{\boldsymbol{v}}_k)$ for each cluster. For each output sample, we update all of K adaptive filters as

$$\begin{aligned}
e_i[n] &= \tilde{x}[n] - \boldsymbol{w}_i^H[n-1]\boldsymbol{u}[n], \\
\boldsymbol{w}_i[n] &= \boldsymbol{w}_i[n-1] + \mu P\left(\boldsymbol{v}[n]|\tilde{\boldsymbol{v}}_i\right)e_i^*[n]\boldsymbol{u}[n]/\left(\|\boldsymbol{u}[n]\|^2\right), \quad (11) \\
\hat{x}[n] &= \boldsymbol{w}_i^H[n]\boldsymbol{u}[n], \quad (12)
\end{aligned}$$

where $i = 1, \ldots, K$. Unlike the hard clustering-based TEQ, each adaptive filter is updated with different step size proportional to the association probability. To generate the final output, the output of the local filter with the largest association probability is selected. The rest of procedure is the same as that of the hard clustering-based TEQ.

TABLE I
COMPLEXITY OF THE TEQs PER OUTPUT SYMBOL

Alg.	Complexity (# of multiplication)
NLMS TEQ	$O(2(N+L))$
Hard clustering-based TEQ	$O(\frac{IK(N+L)}{B(L_T+L_D)}+2(N+L))$
Soft clustering-based TEQ	$O(\frac{IK(N+L)}{B(L_T+L_D)}+2K(N+L))$
MMSE-TEQ with NLMS channel est.	$O((N+L)^2)+3(N+L)$

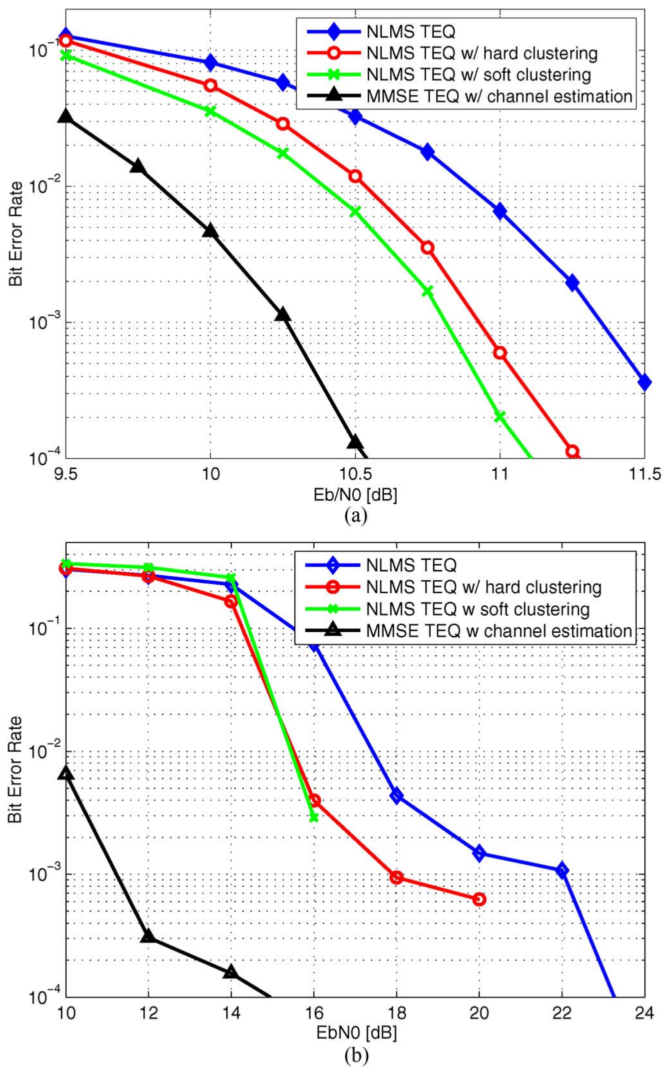

Fig. 2. Plot of BER versus E_b/N_0 for (a) $h_l = [0.227, 0.46, 0.688, 0.46, 0.227]$ with QPSK modulation and and (b) $h_l = [0.407, 0.815, 0.407]$ with 16QAM modulation.

C. Complexity of Clustering-Based TEQs

The clustering-based TEQs retain the complexity of $O(N + L)$ with the additional steps for clustering. One can show that the complexity of the VQ algorithm used for both hard and soft clusterings follows $O(IK(N + L))$ [3], where I is the number of iteration for VQ. Hence, the clustering-based TEQs achieve the linear complexity in channel memory. Furthermore, when multiple packets are sent over the same channel, the VQ algorithm is only needed on the first few packets. Once the whole space is known, a look up table can be used for future packets. For example, by running the VQ algorithm every B packet, the complexity overhead due to the clustering step can be reduced by the factor of B. The computational complexity of the TEQs of interest is summarized in the Table I.

IV. SIMULATION RESULTS

In this subsection, we evaluate the performance of the proposed TEQs through computer simulations. We used the rate 1/2-rate convolutional code with feedback polynomial $1 + D + D^2$ and feedforward polynomial $1 + D^2$. We choose $L_T = 1024$, $L_D = 4096$, $N_{\min} = 1200$ and $K_{\max} = 8$. The parameters of the adaptive TEQ are set by $N_1 = 5$, $N_2 = 5$ and $\mu = 0.03$. A total of $2 \cdot 10^6$ information bits were generated to measure BER. BER was measured after the performance converges. The proposed TEQ would be more suitable for time-invariant channels since for time-varying channels the clustering step should be performed more often. So we restricted our focus to the time-invariant channels. In simulations, we compare performance of the conventional NLMS-TEQ [2], the hard clustering-based NLMS-TEQ and the soft clustering-based NLMS-TEQ. As a reference, we include the MMSE-TEQ with NLMS-based channel estimation as well. Fig. 2(a) shows the BER performance of the TEQs for a linear time-invariant channel $h_l = [0.227, 0.46, 0.688, 0.46, 0.227]$ with QPSK modulation. Though the MMSE-TEQ shows the best performance, we observe substantially longer simulation time due to the quadratic complexity $O(N + L)^2$. The performance gap from the MMSE-TEQ is narrowed by 0.3 dB and 0.5 dB via the hard clustering and the soft clustering, respectively at 10^{-3} BER. In Fig. 2(b), the BER performance of the TEQs is shown for Proakis B channel [2] $h_l = [0.407, 0.815, 0.407]$ with 16-QAM modulation. Note that Proakis B channel exhibits strong ISI so that the large gap between the MMSE-TEQ and NLMS-TEQ is expected. We observe that the proposed TEQs achieves up to 4 dB and 6 dB performance gain over the NLMS-TEQ when the hard clustering and the soft clustering are used, respectively. Note that the clustering-based TEQs retain linear complexity in channel memory, not requiring significant complexity increase with the clustering approach.

V. CONCLUSION

We introduced adaptive locally linear filters based on hard and soft clustering to model the nonlinear dependency of the linear MMSE turbo equalizer on soft information from the decoder. The performance gain of the proposed TEQs over the existing LMS TEQ is demonstrated via computer simulations.

REFERENCES

[1] M. Tüchler, R. Koetter, and A. Singer, "Turbo equalization: Principles and new results," *IEEE Trans. Commun.*, vol. 50, no. 5, pp. 754–767, May 2002.

[2] C. Laot, A. Glavieux, and J. Labat, "Turbo equalization: Adaptive equalization and channel decoding jointly optimized," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 9, pp. 1744–1752, Sep. 2001.

[3] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Boston, MA, USA: Kluwer, 1992.

[4] R. Otnes and M. Tüchler, "Iterative channel estimation for turbo equalization of time-varying frequency-selective channels," *IEEE Trans. Wireless Commun.*, vol. 3, no. 6, pp. 1918–1923, Nov. 2004.

[5] Y. Sun, V. Tripathi, and M. Honig, "Adaptive turbo reduced-rank equalization for MIMO channels," *IEEE Trans. Wireless Commun.*, vol. 4, no. 6, pp. 2789–2800, Nov. 2005.

[6] J. W. Choi, T. J. Riedl, K. Kim, A. C. Singer, and J. C. Preisig, "Adaptive linear turbo equalization over doubly selective channels," *IEEE J. Ocean. Eng.*, vol. 36, no. 4, pp. 473–489, Oct. 2011.

[7] S. Haykin, *Adaptive Filter Theory*, 4th ed., 4th ed. Englewood Cliffs, NJ, USA: Prentice-Hall, 2002.