Multilevel Codes in Lattice-Reduction-Aided Decision-Feedback Equalization

Robert F. H. Fischer, 1 Sebastian Stern, 1 Johannes B. Huber2

1Institut für Nachrichtentechnik, Universität Ulm, Ulm, Germany, Email: {robert.fischer,sebastian.stern}@uni-ulm.de
2Lehrstuhl für digitale Übertragung, Universität Erlangen-Nürnberg, Erlangen, Germany, Email: johannes.huber@fau.de

Abstract—The application of multilevel codes in lattice-reduction-aided (LRA) decision-feedback equalization (DFE) is discussed. There, integer linear combinations of the codewords in signal space have to be decoded. Since multilevel codes do not generate lattices in general and non-integer interference of not yet decoded users is present, straightforward decoding is not possible. A generalized version of multistage decoding adapted to LRA DFE is proposed. Thereby, multilevel constructions using state-of-the-art binary channel codes can be used, which makes coded LRA DFE schemes applicable in practice. The performance of the proposed structure is covered via numerical simulations.

I. INTRODUCTION

Lattice-reduction-aided (LRA) schemes [20], [19] and the tightly related integer-forcing (IF) schemes [13], [22] are low-complexity but well-performing approaches for the equalization in multiple-input/multiple-output (MIMO) multiuser uplink scenarios. They share the concept of decoding integer linear combinations of the user’s signals; they differ in the way how the integer interference is handled, cf. [4].

In IF schemes a strong coupling between integer equalization and decoding/code constraints is present. In LRA schemes the code has to be linear in signal space, i.e., lattice codes can be used. In [5], and independently in [1], it has been shown that for LRA linear equalization (LE) this linearity—integer linear combinations of codewords are valid codewords—can be relaxed and multilevel codes (MLC) can be employed together with a generalized version of multistage decoding (MSD) incorporating “carry correction”.

In this paper, 1 we generalize this result to LRA decision-feedback equalization (DFE). Using DFE, the noise prediction gain over linear equalization can be utilized leading to improved performance [2]. However, the successive decoding in DFE and the carry correction procedure in [5] cannot be combined straightforwardly. To solve this problem, we introduce a new version of generalized MSD which employs tentative decisions. Via this approach, which requires only marginal additional complexity compared to independent MSD, multilevel constructions using state-of-the-art binary channel codes can be used in LRA DFE schemes, which simplifies implementation significantly.

The paper is organized as follows: The system model is introduced in Sec. II and LRA DFE is discussed. Sec. III reviews multilevel codes, multistage decoding with carry correction, and introduces the new decoding scheme. Results from numerical simulations are presented in Sec. IV. The paper is briefly summarized in Sec. V.

1A more comprehensive version can be found in [6].
performance over linear equalization. Desiring an equalization according to the minimum mean-squared error (MMSE) criterion, for lattice-reduction-aided decision-feedback equalization and successive integer-forcing equalization, the augmented (stacked) channel matrix $\mathbf{H} = \begin{bmatrix} \mathbf{H}^\text{H} & \mathbf{I} \end{bmatrix}$, where $\mathbf{H}^\text{H}$ includes row vectors of $\mathbf{H}$, is factorized according to \cite{14, 16}

$$\begin{align*}
\left(\mathbf{H}^\text{H}\right)^\text{H} = \mathbf{F} \mathbf{B}^{-1} \mathbf{Z}^{-1},
\end{align*}$$

where $\mathbf{Z} \in \mathbb{G}^{K \times K}$, $\mathbb{G} = \mathbb{Z} + j\mathbb{Z}$, is a full-rank Gaussian-integer matrix, $\mathbf{B}$ is the upper triangular, unit main diagonal feedback matrix, and the left $N_R$ columns of $\mathbf{F}$ (with orthogonal rows) give the feedforward matrix $\mathbf{F}_{\text{DFE}} = \left[\mathbf{F}\right]_{\text{left columns}}$. Thereby, the factorization is performed such that the columns vectors of $\mathbf{F}^\text{H}$ are as short as possible. As shown in \cite{14} (cf. also \cite{16}), one can restrict to unimodular matrices $\mathbf{Z}$ and the Hermite–Korkine–Zolotarev (HKZ) lattice reduction algorithm \cite{8, 11} is optimum.\footnote{Hereby, the size reduction step is irrelevant; hence an \textit{effective HKZ reduction} is sufficient \cite{16}.}

In the LRA DFE structure (Fig. 1 (bottom)), the feedforward matrix $\mathbf{F}_{\text{DFE}}$ guarantees that the noise is (spatially) white and that the cascade $\mathbf{F}_{\text{DFE}} \mathbf{H} \mathbf{Z}^{-1}$ has an (almost) upper triangular form. This establishes a causality of the interference among the parallel data streams \cite{2}.

Incorporating $\mathbf{F}_{\text{DFE}}$ into the channel, the remaining part of the receiver has to deal with

$$\begin{align*}
\mathbf{r} = \mathbf{B} \hat{\mathbf{z}} x + \hat{n} = \mathbf{B} \hat{x} + \hat{n},
\end{align*}$$

where $\hat{n}$ is the effective disturbance after equalization including filtered channel noise and residual user interference.

In LRA linear equalization \cite{3, 22, 5}, not the users’ signals are decoded but at the decoder input (noisy versions of) Gaussian integer linear combinations thereof are present. In contrast to LRA LE, in LRA DFE they are not decoded simultaneously in parallel but \textit{successively}—the depicted feedback loop is processed branch by branch; due to the upper triangular form of $\mathbf{B}$ the processing order is $l = K, \ldots, 1$.

Noteworthy, (LRA) DFE can also be implemented in the noise prediction structure, shown in Fig. 2 (top), which gives the same performance \cite{2}. Here, the feedforward matrix is given by $\mathbf{F}_{\text{DFE}} = \mathbf{B}^{-1} \mathbf{F}_{\text{DFE}}$, which is identical to the feedforward matrix in linear equalization. Basically, the successive IF structure (Fig. 2 (bottom)) is similar to the LRA noise prediction structure but here the decoding results and noise samples are treated modulo $\mathbf{A}_b$, the boundary lattice of the used signal constellation $\mathcal{A}$. Moreover, the integer interference is resolved over the finite field (as in the linear IF receiver).

These different orders of encoder inverse and inverse of $\mathbf{Z}$ is the main difference between the LRA and IF structures leading to different constraints on the codes. In LRA (linear and DFE) schemes, integer linear combinations in signal space have to be decodable; hence lattice codes are suited. In IF schemes, non-binary codes, tight to the prime signal constellation have to be used \cite{22}. Since the LRA (DFE or noise prediction)}
where $\psi(\cdot)$, $\text{mod}(\cdot)$, and the offset $O$ are applied component-wise.

Eliminating the offset $O$ and ignoring the modulo reduction (inherently assuming an infinite number of extra uncoded levels) multilevel codes can be lattices if the component codes are chosen suitably [12]. However, the respective constraints typically cannot be fulfilled in practical schemes (unless only the lowest level is encoded which results in lattice construction $A$ [21])—in turn integer linear combinations of MLC codewords are not valid codewords of the code and cannot be decoded.

B. Carry Correction

To circumvent this problem and to enable the use of MLC in LRA linear schemes, in [5] (and independently for one-dimensional signaling in [11]) a generalized version of multistage decoding which incorporates a “carry correction” has been proposed. Thereby, the main idea is that the parallel decoders (cf. Fig. 1 (bottom)) can exchange decoding results.

Instead of decoding each linear combination (via multistage decoding) separately, the lowest level in each branch is decoded. To this end, we note that the effective codewords at level $\mu = 0$, i.e., the results of Gaussian integer linear combinations in signal space, are given by

$$
\begin{bmatrix}
\ell^{(\mu)}_{\text{eff}, 1} \\
\ell^{(\mu)}_{\text{eff}, 2} \\
\vdots \\
\ell^{(\mu)}_{\text{eff}, K}
\end{bmatrix} = \mathbf{3}_0 \begin{bmatrix}
c^{(\mu)}_1 \\
c^{(\mu)}_2 \\
\vdots \\
c^{(\mu)}_K
\end{bmatrix},
$$

where $\mathbf{3}_0 = [3^{(i,j)}_0]$ and $3^{(i,j)}_0$ is the least significant bit (LSB) of $z_{i,j}$ w.r.t. to the basis $\phi$. As long as $\det(Z) \in 1 + \phi \mathbb{G}$ [5], (6) can be solved and the original codewords of each user in the lowest level can be regenerated.

Having these estimates ($\ell^{(\mu)}_{\text{eff}}$), the “carries” to the higher levels can be calculated and subtracted. The contributions (over $\mathbb{C}$) $\mathbf{3}^{(0)}$ of the superposition of these code words into the higher levels (carries) of user $k$ calculate to ($\mu = 0$)

$$
\begin{bmatrix}
\ell^{(\mu)}_1 \\
\ell^{(\mu)}_2 \\
\vdots \\
\ell^{(\mu)}_K
\end{bmatrix} = \mathbf{Z} \begin{bmatrix}
\psi_{\ell^{(\mu)}}(1) \\
\psi_{\ell^{(\mu)}}(2) \\
\vdots \\
\psi_{\ell^{(\mu)}}(K)
\end{bmatrix},
$$

This procedure is repeated over the levels $\mu$.

C. Carry Correction in DFE

Unfortunately, this procedure cannot be applied straightforwardly in LRA DFE. In linear equalization, integer linear combinations cause “interference” from the lower levels to the upper ones—causality over the levels is present. Such a causality w.r.t. the code levels does not exist in DFE since the channel is equalized only towards an upper triangular matrix. The not yet decoded upper levels cause interference via the non-integer off-diagonal entries of $\mathbf{B}$. The fractional part of $b_{lj,k}$ determines how the upper levels of user $k$ interfere with a particular level of linear combination $l \leq k$.

Hence, to eliminate interference of other users when decoding level $\mu$ of linear combination $l$, all lower levels $\nu = 0, \ldots, \mu - 1$ of all users have to be known and the upper levels $\nu = \mu, \ldots, m - 1$ of users $K, K - 1, \ldots, l + 1$ whose interference has to be subtracted via $\mathbf{B}$ also have to be known.

This, at first glance, prevents the application of multistage decoding with carry correction as developed for the linear equalization case. However, a small modification is sufficient to use the same philosophy in LRA DFE. The main idea is to employ tentative decisions on the higher levels. When having a decoding result $\widehat{c}^{(\mu)}_{\text{eff},l}$ for level $\mu$ of linear combination $l$ (levels $\nu = 0, \ldots, \mu - 1$ are already available from previous decoding stages) symbol-by-symbol decisions $\overline{c}^{(\mu),l}$ on all upper levels jointly of this linear combination are additionally generated via quantization $Q_{\mathbb{G}}(\cdot)$ to the signal point lattice $\mathbb{G}$—thereby, the code constraints in higher levels are simply ignored. As now for all levels (tentative) decoding results are available, a tentative estimate (row vector)

$$
\overline{c}^{\nu}_{\text{eff},l} = \sum_{l = 0}^{\nu} \overline{c}^{(\mu)}_{\text{eff},l} \phi^\nu + \overline{c}^{(\mu),l}\phi^\nu + 1
$$

of linear combination $l$ can be calculated. Thereby, the contributions $\overline{c}^{(\nu)}_{\text{eff},l}$ of the lower levels to the higher ones are calculated as in (7). The tentative estimates of linear combinations $l + 1, \ldots, K$ are used in the feedback loop to eliminate the integer interference (“carry correction”) and the non-integer residual interference. This procedure is repeated over the levels.

Fig. 3 visualizes the dependencies in the decoding process. The effective codewords at the individual levels of the multilevel construction are shown. Having decoded all effective codewords $c^{(\nu)}_{\text{eff},l}$ at one level $\nu$, the original codewords at this level can be calculated using (6) and the “carries” to the higher levels can be calculated using (7) and subtracted.

When decoding $c^{(\mu)}_{\text{eff},l}$ (bold frame), the effective codewords (and hence initial codewords) of levels $0, \ldots, \mu - 1$ are already decoded (dark gray shaded). Due to the successive procedure (going from $l = K$ to 1, i.e., right to left in the figure), the effective codewords $l + 1, \ldots, K$ at levels $\mu$ have been decoded, too. In addition, hard (tentative) decisions on the upper levels $> \mu$ are generated (light gray shaded). Using the tentative decisions (8) of data streams $l + 1, \ldots, K$, the interference is subtracted via the feedback matrix $\mathbf{B}$. Hence, $c^{(\mu)}_{\text{eff},l}$ can be decoded free of carries of lower levels and interference of other users. This is successively done for all
data streams at one coding level. The decoding process then continues with the next level.

In QAM signaling the next level operates at a 3 dB higher SNR; the tentative decisions are hence reliable enough. Thus, there are only a few erroneous tentative decisions at higher levels when compared to the lower SNR at the actual level. These errors are controllable by the codes at the actual level without a serious performance degradation. Moreover, for LRA DFE a unimodular integer matrix $Z$ is optimal (cf. Sec. II). For such matrices it is guaranteed that (6) is solvable [5] and carry correction works.

In Alg. 1, a pseudo-code description of this generalized version of multistage decoding is given. Noteworthy, if $B = I$, this algorithms reduces to that in [5] and if additionally $Z = I$ conventional multistage decoding in parallel for the users results.

The complexity is dominated by the runs of the component decoders; each level of each user is decoded exactly once. Consequently, the same effort as $K$ times conventional multistage decoding (parallel, individual decoding of the users) is required.

IV. NUMERICAL RESULTS

To study the performance of the above proposed decoding algorithm, numerical simulations have been conducted. As a simple example, we assume $K = 3$ users, each employing a 16QAM constellation. The same low-density parity-check (LDPC) codes, in particular irregular repeat-accumulate codes [10], as in [5] with rates $R_0/R_1/R_2/R_3 = 0.282/0.753/0.964/1$ (sum rate 3 bits per QAM symbol) and code length $N = 5000$ are employed as component codes.

To enlighten the effects of decoding integer linear combinations in the LRA DFE structure and to show the gains over LRA linear equalization, first the channel matrix is randomly chosen and kept fixed. The selected channel matrix reads

$$H = \begin{bmatrix}
0.336 + 0.151j & -0.566 - 0.014j & -0.255 + 0.454j \\
-1.101 + 0.581j & 0.247 - 0.185j & -0.373 - 0.465j \\
-1.848 - 1.037j & 0.019 + 0.758j & 1.776 - 1.298j
\end{bmatrix}.$$ (9)

For LRA linear equalization we employ the Minkowski reduction, as for i.i.d. Gaussian channel matrices the restriction to unimodular matrices ($|\det(Z)| = 1$) causes no noticeable loss, cf. [4], [15]. The following integer matrix is obtained

$$Z_{MK} = \begin{bmatrix}
1 & -1+j & -1-j \\
-1+j & 0 & -j \\
-1-j & 0 & 1
\end{bmatrix}.$$ (10)

For LRA DFE (see Sec. II) we employ the HKZ reduction on the factorization problem (2); here the integer matrix and the feedback matrix calculate to

$$Z_{HKZ} = \begin{bmatrix}
-2 & 1 & -2j \\
-1+j & 0 & -j \\
-1-j & 0 & 1
\end{bmatrix},$$ (11)

$$B_{HKZ} = \begin{bmatrix}
1 & 0.422 - 0.433j & -0.609 - 0.395j \\
0 & 1 & -0.288 - 0.108j \\
0 & 0 & 1
\end{bmatrix}.$$ (12)

In all cases, the feedforward equalizers are calculated according to the MMSE criterion.

Fig. 4 shows the error rates of the individual bits of the user over the blow noise power (in dB). For comparison, the performance of uncoded transmission is shown (black, dashed) and that of the multilevel codes (no linear combinations) over the single-input/single-output (SISO) AWGN channel (green). Noteworthy, due to the uncoded ($R_3 = 1$) highest level, the asymptotic (gross) coding gain is limited to 9 dB (dotted).

In all cases, user 2 has the worst performance, which is due to the noise enhancement in the feedforward filter. In LE this effect is much more pronounced (approximately 2.4 dB worse) than in case of DFE (gain by not equalizing the channel to (almost) identity matrix but only to upper triangular form). Users 1 and 3 perform almost the same (the curves lie almost on top of each other) for a given receiver type but better in case of DFE. This positive effect cannot be explained by reduced noise enhancement in the frontend as linear combinations number 2 and 3 almost have the same noise enhancement in the linear and the DFE case. The better performance is due to the fact that in the successive procedure correlated
linear combinations have to be decoded [3]; these correlations are exploited in DFE but ignored in linear equalization. The performance of these two users is very close to that of the original code over the SISO AWGN channel.

Next, the channel matrix is randomly chosen with i.i.d. circular symmetric complex unit-variance Gaussian entries. A block-fading channel is assumed, where the channel matrix is constant over the codeword. Hence, the code cannot exploit temporal diversity. Given the channel matrix, the integer matrices are calculated using the Minkowski reduction (which gives the optimal unimodular matrix for LRA LE) and the HKZ reduction (which gives the optimal matrix for LRA DFE), respectively. \( N_R = 3 \) receive antennas and \( K = 3 \) users are assumed; the codes and signal constellations from above are assumed.

Fig. 5 shows the average error rates of the information bits of the users over the inverse noise power (in dB). 16QAM. Component codes: LDPC codes of length \( N = 5000 \). Rate 3 bit/symbol. \( N_R = 3 \) receive antennas, \( K = 3 \) users. Block-fading channel with i.i.d. circular symmetric complex Gaussian entries. Average over 100000 i.i.d. circular symmetric complex Gaussian channel matrices and 10 codewords per channel realization.

V. SUMMARY AND CONCLUSIONS

In this paper, we have studied the application of multilevel codes in LRA decision-feedback equalization. Employing DFE, the noise prediction gain overlinear equalization can be utilized leading to better performance. A generalized version of multistage decoding incorporating carry correction and tentative decisions has been proposed. Only marginal additional complexity compared to independent decoding is required. Via the multilevel construction, state-of-the-art binary channel codes can be used and no lattice structure of the code is required. This simplifies implementation significantly or even makes coded LRA schemes applicable in practice. 

REFERENCES