Factorization Approaches in Lattice-Reduction-Aided and Integer-Forcing Equalization

Robert F.H. Fischer[1], Michael Cyran[2], Sebastian Stern[1]

[1]Institut für Nachrichtentechnik, Universität Ulm, Ulm, Germany, Email: {robert.fischer,sebastian.stern}@uni-ulm.de
[2]Lehrstuhl für Informationsübertragung, Universität Erlangen-Nürnberg, Erlangen, Germany, Email: michael.cyran@fau.de

***Abstract*—In this paper, lattice-reduction-aided and integer-forcing equalization are contrasted. In both approaches, the determination of an integer matrix is essential. The different criteria for this calculation available in the literature are summarized in a unified way. A new factorization algorithm for obtaining the integer matrix is proposed. Via extensive numerical simulations the gains of the respective optimization criterion and the gain of the new algorithm over the classical Lenstra-Lenstra-Lovász algorithm are assessed. In particular, the gains achieved by dropping the constraint that the integer matrix has to be unimodular are identified.**

I. Introduction

The design of joint receivers for signals transmitted in parallel, e.g., in multi-user uplink scenarios, is still an important topic in research. The simplest approach for handling the interference in the underlying *multiple-input/multiple-output (MIMO)* channels is to use *linear equalization* (either optimized according to the zero-forcing (ZF) or the minimum-mean squared error (MMSE) criterion). Via a (pseudo left) inverse of the channel matrix, the interference is eliminated at the cost of noise enhancement. However, as the users are perfectly decoupled, individual channel decoding can be performed and individual codes can be used. Some improvement can be gained by utilizing *decision-feedback equalization* (DFE), also known as *successive interference cancellation* (SIC) and as *Bell Laboratories space-time* (BLAST). The optimum receive strategy is *maximum-likelihood decoding*, which, however, for coded transmission has infeasible complexity.

Since more than one decade, low-complexity but well-performing approaches are of particular interest. *Lattice-reduction-aided* (LRA) techniques, e.g., [19], [16], [18], were proven to achieve the optimum diversity behavior [15]. Recently, the concept of *integer-forcing* (IF) receivers [20] was proposed. Both strategies are tightly related; the term "LRA" can be interpreted as a channel-oriented view—it puts emphasis on the mathematical tool applied to the channel matrix. In contrast, the denomination "IF" is signal-oriented—it highlights the main operation on the signals.

In this paper, a brief comparison of both approaches is given and the advantages and disadvantages of the respective procedures are enlightened. Moreover, the different criteria for selecting the integer matrix—which is central in both fields—available in the literature are summarized in a unified way. A new factorization algorithm for determining this matrix in an optimum way is presented. Using this algorithm, via extensive numerical simulations, the gains of the respective optimization criterion and the gain of the new algorithm over the classical Lenstra-Lenstra-Lovász algorithm are assessed.

The paper is organized as follows: In Sec. II the system model is introduced and in Sec. III LRA and IF strategies are contrasted and the different factorization criteria are summarized. A new factorization algorithm is presented in Sec. IV followed by numerical examples in Sec. V. Sec. VI briefly summarizes the paper.

This work was supported by the Deutsche Forschungsgemeinschaft (DFG) within the framework COIN under grant FI 982/4-3.

II. System Model

We assume a classical MIMO channel model with K non-cooperating transmitters (single-antenna users) and a joint receiver with N antennas. Fig. 1 shows the block diagram of the system model.

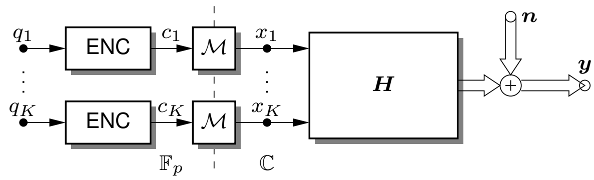

Fig. 1. System model of the MIMO communication scheme.

Each user k, $k = 1, \ldots, K$, wants to communicate its source symbols q_k drawn from a finite field $\mathbb{F}_p$. Blocks of source symbols are encoded via some channel code; the coded symbols c_k are then mapped to complex-valued transmit symbols x_k, drawn from some signal constellation $\mathcal{A}$. Via a suited choice of the code (including interleaving where required) and the mapping this generic model includes all types of coded modulation schemes, lattice-coding approaches, as well as uncoded transmission.

The symbols x_k are then radiated over the users' antennas. Denoting the transmit vector (dimension K) as $\boldsymbol{x}$, the $N \times K$ channel matrix as $\boldsymbol{H}$, and the N-dimensional noise vector as $\boldsymbol{n}$, the receive vector $\boldsymbol{y}$ is given by

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{n} \,. \tag{1}$$

The transmit symbols (per user) have variance σ_x^2 and the zero-mean Gaussian noise has variance σ_n^2 per dimension. Noteworthy, all signals and channel coefficients are complex-valued in the equivalent complex baseband domain.

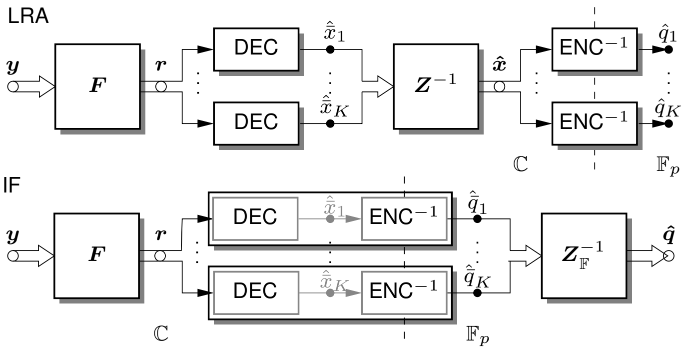

Fig. 2. System model of the receiver. Top: lattice-reduction-aided equalization; Bottom: integer-forcing receiver.

At the receiver side, the N components of the receive vector $\boldsymbol{y}$ can be processed jointly in order to produce estimates of the source symbols q_k. To this end, some form of equalization has to take place and suited channel decoding has to be performed. In the next section, we will have a closer look at the low-complexity, well-performing LRA and IF strategies.

III. Lattice-Reduction-Aided Equalization and Integer-Forcing Equalization

Lattice-reduction-aided (LRA) and integer-forcing (IF) receivers share the same fundamental principle. The main idea is to factorize the channel matrix as

$$\boldsymbol{H} = \boldsymbol{C}\,\boldsymbol{Z}\,. \tag{2}$$

The receive vector can then be written as

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{n} = \boldsymbol{C}\,\boldsymbol{Z}\boldsymbol{x} + \boldsymbol{n} = \boldsymbol{C}\,\bar{\boldsymbol{x}} + \boldsymbol{n}\,. \tag{3}$$

Then, not the transmit vector $\boldsymbol{x}$ itself (blocks of vectors in the coded case) is recovered but the vector $\bar{\boldsymbol{x}} \overset{\text{def}}{=} \boldsymbol{Z}\boldsymbol{x}$. If $\boldsymbol{Z}$ is chosen suitably, this may be done with much less noise enhancement. Taking into account that the symbols of the vector $\bar{\boldsymbol{x}}$ are correlated (due to $\boldsymbol{Z}$), the respective MMSE linear equalizer calculates to [5], [20] (inverse SNR $\zeta \overset{\text{def}}{=} \sigma_n^2/\sigma_x^2$)

$$\boldsymbol{F} = \left(\boldsymbol{C}^{\mathsf{H}}\boldsymbol{C} + \zeta\boldsymbol{Z}^{-\mathsf{H}}\boldsymbol{Z}^{-1}\right)^{-1}\boldsymbol{C}^{\mathsf{H}} \tag{4}$$

$$= \boldsymbol{Z}\left(\boldsymbol{H}^{\mathsf{H}}\boldsymbol{H} + \zeta\boldsymbol{I}\right)^{-1}\boldsymbol{H}^{\mathsf{H}}\,. \tag{5}$$

Hence, detection/decoding is done w.r.t. some changed basis. Having the decoding results, this basis change (the matrix $\boldsymbol{Z}$) is reversed. The LRA and IF strategies differ in the way this final step is done and how the matrix $\boldsymbol{Z}$ is chosen. The block diagrams of the respective receivers are depicted in Fig. 2.

A. Lattice-Reduction-Aided Equalization

LRA equalization has its roots in the field of lattice reduction, i.e., the question of finding a suited basis for a given lattice; here the lattice spanned by the columns of the channel matrix $\boldsymbol{H}$. Consequently, $\boldsymbol{Z}$ is chosen as an integer unimodular matrix. In the complex case, the coefficients of $\boldsymbol{Z}$ are drawn from the *Gaussian integers* $\mathbb{G} \overset{\text{def}}{=} \mathbb{Z} + \mathrm{j}\mathbb{Z}$ and $|\det(\boldsymbol{Z})| = 1$, such that $\boldsymbol{Z}^{-1}$ is also an (complex) integer unimodular matrix. Using lattice-reduction algorithms, most prominently the LLL algorithm [10] or its complex-valued generalization [6], a solution can readily be found.

Decoding and resolution of the interference via $\boldsymbol{Z}^{-1}$ is done over the complex numbers; the linear combinations $\bar{\boldsymbol{x}}$ of the transmit symbols have to be estimated by the decoders. Then, an estimate of the transmit symbols is obtained via $\hat{\boldsymbol{x}} = \boldsymbol{Z}^{-1}\hat{\bar{\boldsymbol{x}}}$. Finally, via the encoder inverses, estimates $\hat{q}_k$ of the source symbols are obtained.

LRA equalization only[1] works if the signal constellation $\mathcal{A}$ is a subset[2] of $\mathbb{G}$, i.e., $\boldsymbol{x} \in \mathbb{G}^K$, such that any (complex) integer linear combination of the points is again drawn from $\mathbb{G}$. Moreover, in the coded case, the codes have to be linear, such that any (complex) integer linear combination of codewords is again a valid codeword. It is true that in the vast majority of the literature, LRA equalization is treated uncoded. This, however, is justified as equalization and decoding can simply be cascaded; coding can straightforwardly be put on top of the uncoded LRA scheme. No further specific restrictions have to be obeyed.

B. Integer-Forcing Equalization

Recently, originating from *compute-and-forward* relaying schemes [11], an integer-forcing linear equalization scheme was proposed in [20]. The main difference, see Fig. 2, is that the integer interference is resolved over the finite field rather than over the complex numbers. To this end, linear combinations $\bar{q}_k$ of the source symbols are delivered by the decoders and the integer matrix is inverted over $\mathbb{F}_p$. Put simply, the order of encoder inverse and inverse of $\boldsymbol{Z}$ is reversed.

However, this imposes much stronger constraints on the codes and the mapping as in the LRA case. Basically, arithmetic over the complex numbers (modulo p) has to be isomorphic to the arithmetic of the finite field $\mathbb{F}_p$. In the simplest version this is achieved by restricting to real-valued signaling and $\mathcal{A}$ is a one-dimensional p-ary constellation where p is a prime. Generalization to complex-valued Gaussian prime constellations [9] or other algebraic structures [4] is possible.

Since the integer interference is resolved over the finite field, the matrix $\boldsymbol{Z}$ has to be invertible over $\mathbb{F}_p$. Since p is a prime this is possible as long as $\boldsymbol{Z}$ has full rank; no restriction on the determinant is required. This gives rise to a new factorization problem: not a *shortest basis problem* as in LRA has to be solved but a *shortest independent vector problem* [20].

C. Comparison

Even though LRA and IF are tightly related, the constraints and restrictions are different. IF imposes strong constraints on the signal constellation and its cardinality and in turn on the applicable codes. In LRA only linearity in signal space is required. Contrary, here unimodularity of $\boldsymbol{Z}$ is forced.

[1]Generalization to other lattices, e.g., the Eisenstein integers [2], are possible. In each case, the signal constellation and the entries of $\boldsymbol{Z}$ have to be taken from the same lattice/algebraic structure, cf. [4].

[2]If an offset is present as in usual QAM constellations, LRA equalization still works if this offset is adequately taken into account, e.g., [17].

The presentation of the IF schemes has sparked a rethinking of the LRA approach—indeed, unimodularity is not required. If[3] $|\det(\boldsymbol{Z})| > 1$ the vector $\bar{\boldsymbol{x}} = \boldsymbol{Z}\boldsymbol{x}$, with $\boldsymbol{x} \in \mathbb{G}^K$, is not taken from $\mathbb{G}^K$ but a sublattice thereof.[4] Given the points from this sublattice, $\boldsymbol{Z}^{-1}$—which has a determinant smaller than one—will recover the original transmit vector $\boldsymbol{x}$. Hence, the LRA equalizer structure can be used with any full-rank integer matrix $\boldsymbol{Z}$, enabling the same gains as in IF but without the restrictions on the signal constellation and the codes.

IF schemes have their main justification not in central but in decentralized receivers. In a distributed antenna system, the partial equalization via $\boldsymbol{F}$ cannot be applied; the residual interference is taken as it is and the decoders produce estimates on linear combinations. In IF schemes, only symbols from $\mathbb{F}_p$ have to be communicated over the backhaul. The integer interference is resolved in some central processing unit. Conversely, using the LRA structure, complex numbers would have to be sent. In a central receiver the LRA structure is preferable.

In summary, LRA and IF have its individual advantages and constraints. However, the calculation of the integer matrix can be done in the same way for both approaches. For that we have to distinguish between the different *criteria* the optimization is based on and between different *factorization algorithms*.

D. Factorization Criteria

We now give an overview on the different criteria the factorization task (2) is usually based on.

C-I *Based on $\boldsymbol{H}$:* In the initial publications [19], [16], lattice reduction is directly applied to the channel matrix $\boldsymbol{H}$

$$\boldsymbol{H} = \boldsymbol{C}_{\mathrm{I}}\, \boldsymbol{Z}_{\mathrm{I}} \,. \tag{6}$$

Any lattice reduction algorithm may be used, e.g., minimizing the orthogonality defect of $\boldsymbol{C}$.

C-II *Based on $\boldsymbol{H}^{-\mathsf{H}}$:* In [15], the factorization

$$\boldsymbol{H}^{-\mathsf{H}} = \boldsymbol{F}_{\mathrm{II}}^{\mathsf{H}}\, \boldsymbol{Z}_{\mathrm{II}}^{-\mathsf{H}} \tag{7}$$

has been proposed. As for square matrices $\boldsymbol{F}^{\mathsf{H}} = \boldsymbol{C}^{-\mathsf{H}} = \boldsymbol{H}^{-\mathsf{H}}\, \boldsymbol{Z}^{\mathsf{H}}$ follows from (6), $\boldsymbol{F}$ is immediately the (ZF) equalization matrix and $\boldsymbol{Z}$ is the required integer matrix. Here, lattice reduction is applied to $\boldsymbol{H}^{-\mathsf{H}}$ instead of $\boldsymbol{H}$ (for non-square channel matrices the Hermitian of the left pseudoinverse has to be used). Since the squared lengths of the columns of $\boldsymbol{F}^{\mathsf{H}}$ give the noise enhancement (in case of ZF linear equalization), this criterion directly optimizes the performance of the scheme instead of a substitute measure as above.

C-III *Based on $\boldsymbol{\mathcal{H}}$:* In [18], an MMSE version to LRA equalization has been given. The main idea is to calculate the ZF solution for the augmented[5] channel matrix; the result is exactly the MMSE solution. The factorization here reads

$$\begin{bmatrix} \boldsymbol{H} \\ \sqrt{\zeta}\boldsymbol{I} \end{bmatrix} \overset{\text{def}}{=} \boldsymbol{\mathcal{H}} = \boldsymbol{\mathcal{C}}_{\mathrm{III}}\, \boldsymbol{Z}_{\mathrm{III}} = \begin{bmatrix} \boldsymbol{C}_{\mathrm{III}} \\ \sqrt{\zeta}\boldsymbol{Z}_{\mathrm{III}}^{-1} \end{bmatrix} \boldsymbol{Z}_{\mathrm{III}} \,. \tag{8}$$

[3]As $\boldsymbol{Z} \in \mathbb{G}^{K\times K}$, $|\det(\boldsymbol{Z})| < 1$ is not possible for full-rank matrices.

[4]The individual decoding/detection of the components of $\bar{\boldsymbol{x}}$ is suboptimal, as non-valid points can be delivered. This is anyway the case as the actual boundary region of the constellation cannot be taken into account in separate decoding, cf. [17]. For sufficiently large SNR this fact is irrelevant.

[5]Augmented matrices are typeset in calligraphic font.

TABLE I
OVERVIEW ON FACTORIZATION STRATEGIES.

based on	channel matrix $\boldsymbol{H}$ ("ZF solution")	augmented matrix $\boldsymbol{\mathcal{H}}$ ("MMSE solution")
$\boldsymbol{H}$	$\boldsymbol{H} = \boldsymbol{C}\,\boldsymbol{Z}$	$\boldsymbol{\mathcal{H}} = \boldsymbol{\mathcal{C}}\,\boldsymbol{Z}$
$(\boldsymbol{H}^{+_\mathrm{l}})^{\mathsf{H}}$	$(\boldsymbol{H}^{+_\mathrm{l}})^{\mathsf{H}} = \boldsymbol{F}^{\mathsf{H}}\,\boldsymbol{Z}^{-\mathsf{H}}$	$(\boldsymbol{\mathcal{H}}^{+_\mathrm{l}})^{\mathsf{H}} = \boldsymbol{\mathcal{F}}^{\mathsf{H}}\,\boldsymbol{Z}^{-\mathsf{H}}$

Interestingly, the left pseudoinverse[6] $\boldsymbol{\mathcal{C}}^{+_\mathrm{l}}$ of $\boldsymbol{\mathcal{C}}$, immediately gives the augmented receive matrix, as a comparison with (4) shows [5].

C-IV *Based on $(\boldsymbol{\mathcal{H}}^{+_\mathrm{l}})^{\mathsf{H}}$:* In [20] a criterion for directly minimizing the noise variance after MMSE linear equalization of the part $\boldsymbol{C}$ has been given. With $\boldsymbol{Z}^{\mathsf{H}} = [\boldsymbol{z}_1, \ldots, \boldsymbol{z}_K]$ it reads

$$\boldsymbol{Z}_{\mathrm{IV}}^{\mathsf{H}} = \underset{\substack{\boldsymbol{Z}\in\mathbb{G}^{K\times K},\\ \mathrm{rank}(\boldsymbol{Z})=K}}{\operatorname{argmin}} \max_{m} ||\boldsymbol{L}^{\mathsf{H}}\boldsymbol{z}_m||^2 \,, \tag{9}$$

where $$\boldsymbol{L}\boldsymbol{L}^{\mathsf{H}} = \left(\boldsymbol{H}^{\mathsf{H}}\boldsymbol{H} + \zeta\boldsymbol{I}\right)^{-1} = \left(\boldsymbol{\mathcal{H}}^{\mathsf{H}}\boldsymbol{\mathcal{H}}\right)^{-1} . \tag{10}$$

Since $\boldsymbol{L}$ can be any "square root" of the right-hand-side matrix, as straightforward calculations show, we can set

$$\boldsymbol{L}^{\mathsf{H}} = \left(\boldsymbol{\mathcal{H}}^{+_\mathrm{l}}\right)^{\mathsf{H}} = \left(\boldsymbol{\mathcal{H}}^{\mathsf{H}}\right)^{+_\mathrm{r}} \tag{11}$$

and the respective factorization task can thus be written as

$$\left(\boldsymbol{\mathcal{H}}^{+_\mathrm{l}}\right)^{\mathsf{H}} = \boldsymbol{\mathcal{F}}_{\mathrm{IV}}^{\mathsf{H}}\, \boldsymbol{Z}_{\mathrm{IV}}^{-\mathsf{H}} \,. \tag{12}$$

Noteworthy, for all optimization criteria a respective factorization task[7] can be stated in which $\boldsymbol{Z}$ has to be chosen such that the squared lengths of the column of the matrix $\boldsymbol{C}_{\mathrm{I}}$, $\boldsymbol{F}_{\mathrm{II}}^{\mathsf{H}}$, $\boldsymbol{\mathcal{C}}_{\mathrm{III}}$, or $\boldsymbol{\mathcal{F}}_{\mathrm{IV}}^{\mathsf{H}}$, respectively, are as short as possible.

Table I gives an overview on the different criteria for the factorization problem.

IV. Factorization Algorithm

The above overview has shown that regardless which optimization criterion is used, a factorization problem has to be solved in order to obtain the required integer matrix $\boldsymbol{Z}$. If we follow the original LRA approach and restrict $\boldsymbol{Z}$ to be unimodular, any *lattice reduction algorithm*, in particular the LLL algorithm [10], can be used. For the complex-valued setting at hand, the CLLL [6] may be applied.

If the unimodularity is dropped, an algorithm for solving the *shortest independent vector problem (SIVP)* has to be applied. Unfortunately, in the literature, only a few approaches are available. In [20], the optimization problem (9) or (12) is solved via a brute-force search with some restrictions to the search space. In [13], [14], low-complexity factorization approaches, all directly based on the CLLL and hence resulting in a unimodular matrix, are given. An algorithm to solve the successive minima problem has been published in [3]. For the distributed antenna setting, in [8] a suited factorization is given, taking into account that no joint feedforward equalization via $\boldsymbol{F}$ is possible.

[6]$\boldsymbol{A}^{+_\mathrm{l}} = (\boldsymbol{A}^{\mathsf{H}}\boldsymbol{A})^{-1}\boldsymbol{A}^{\mathsf{H}}$ denotes the left pseudoinverse of $\boldsymbol{A}$ and $\boldsymbol{A}^{+_\mathrm{r}} = \boldsymbol{A}^{\mathsf{H}}(\boldsymbol{A}\boldsymbol{A}^{\mathsf{H}})^{-1}$ the right pseudoinverse. $\boldsymbol{A}^{-\mathsf{H}} = (\boldsymbol{A}^{\mathsf{H}})^{-1} = (\boldsymbol{A}^{-1})^{\mathsf{H}}$.

[7]We hence denote the corresponding procedure *factorization algorithm*.

Alg. 1 Pseudocode of the factorization algorithm.

function $\boldsymbol{Z} = \texttt{factorize}(\boldsymbol{H}, \zeta)$

$\boldsymbol{G} = \left(\boldsymbol{H}^{\mathsf{H}}\boldsymbol{H} + \zeta\boldsymbol{I}\right)^{-1/2}$	// generator matrix
$[\boldsymbol{G}_{\mathrm{LLL}}, \boldsymbol{Z}_{\mathrm{LLL}}] = \texttt{LLL}(\boldsymbol{G})$	// reduced basis
$R^2_{\max} = \max_i \lvert\lvert \boldsymbol{g}_{i,\mathrm{LLL}} \rvert\rvert^2$	// search radius
$\boldsymbol{U} = \texttt{getlist}(\boldsymbol{G}_{\mathrm{LLL}}, R^2_{\max})$	// get list of short vectors
$\{i_1, \ldots, i_K\} = \texttt{getindices}(\boldsymbol{U})$	// indices of lin. indep. vectors
$\boldsymbol{Z} = \boldsymbol{Z}_{\mathrm{LLL}} \cdot \boldsymbol{U}(:, [i_1, \ldots, i_K])$	// integer matrix

We now present an algorithm which is feasible for MIMO scenarios typically of interest; a pseudocode description is given in Alg. 1. Via numerical simulations we can then study the gain possible by the respective criteria and the loss when restricting $\boldsymbol{Z}$ to be unimodular. To have a compact notation, we rewrite (10), (12) as

$$\boldsymbol{G}_{\mathrm{opt}} = \boldsymbol{G}\,\boldsymbol{Z}^{\mathsf{H}}\,, \tag{13}$$

with $\boldsymbol{G} = \boldsymbol{L}^{\mathsf{H}}$, $\boldsymbol{Z}^{\mathsf{H}} = [\boldsymbol{z}_1, \ldots, \boldsymbol{z}_K] \in \mathbb{G}^{K\times K}$, $\mathrm{rank}(\boldsymbol{Z}) = K$, and the columns of $\boldsymbol{G}_{\mathrm{opt}}$ as short as possible. This means that given the basis $\boldsymbol{G}$ of a lattice, find K linearly independent vectors (lattice points) $\boldsymbol{G}\boldsymbol{u}_i$, $\boldsymbol{u}_i \in \mathbb{G}^K$, which are as short as possible. We do this via performing the following steps:

LLL Reduction:

First, the LLL reduced basis $\boldsymbol{G}_{\mathrm{LLL}} = \left[\boldsymbol{g}_{1,\mathrm{LLL}}, \cdots, \boldsymbol{g}_{K,\mathrm{LLL}}\right]$ is calculated. Since the SIVP is more relaxed than the shortest basis problem, the LLL basis gives an upper bound $R^2_{\max} = \max_i \lvert\lvert \boldsymbol{g}_{i,\mathrm{LLL}} \rvert\rvert^2$ on the norms of the vectors possible in the SIVP. We denote this step as $[\boldsymbol{G}_{\mathrm{LLL}}, \boldsymbol{Z}_{\mathrm{LLL}}] = \texttt{LLL}(\boldsymbol{G})$.

List of Lattice Points:

Then, a (sorted) list of vectors (lattice points) with squared norms bounded by $R^2_{\max}$ is calculated. Let the list be written as matrix $\boldsymbol{U} = \left[\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_\ell\right]$, with $\lvert\lvert \boldsymbol{G}_{\mathrm{LLL}}\boldsymbol{u}_i \rvert\rvert^2 \leq \lvert\lvert \boldsymbol{G}_{\mathrm{LLL}}\boldsymbol{u}_{i+1} \rvert\rvert^2$, $\forall i$. Since for complex lattices the volume is given by the squared magnitude of the determinant of the generator matrix [2, Eq. (87)]), the list size can be approximated by $\ell = (\pi R^2_{\max})^K / (K! \lvert\det(\boldsymbol{G})\rvert^2)$. We denote this step as $\boldsymbol{U} = \texttt{getlist}(\boldsymbol{G}_{\mathrm{LLL}}, R^2_{\max})$.

This step can be implemented efficiently using the idea of the list sphere decoder, Alg. AllClosestPoints in [1]. In principle, this calculation has exponential complexity, however, if the LLL basis is used and $R^2_{\max}$ is small and, thus, the number ℓ of points within the search sphere is small, still an efficient search is obtained.

Select Points:

Among the vectors in the list (matrix $\boldsymbol{U}$) the best combination of vectors, i.e., indices $i_1, \ldots, i_K$, has to be found such that $\boldsymbol{Z} = \left[\boldsymbol{u}_{i_1}, \ldots, \boldsymbol{u}_{i_K}\right]$ has full rank and $\boldsymbol{G}_{\mathrm{LLL}}\boldsymbol{u}_{i_K}$ is as small as possible.

The last step can be solved by performing Gaussian elimination on the matrix $\boldsymbol{U}$, i.e., transforming it to *row echelon form*. Since the list is sorted according to increasing (squared) norms $\lvert\lvert \boldsymbol{G}_{\mathrm{LLL}}\boldsymbol{u}_i \rvert\rvert^2$, the best choice is to select the vectors $\boldsymbol{u}_i$, which first define a new dimension; in row echelon form these are the vectors at the steps. We denote this step as $\{i_1, \ldots, i_K\} = \texttt{getindices}(\boldsymbol{U})$.

Please note, if some restrictions (e.g., on the determinant) of $\boldsymbol{Z}$ have to be obeyed, a search over combinations of candidates can be performed instead of the simple Gaussian elimination. This step can efficiently be implemented by the sphere decoder and offers degrees of freedom not present in other algorithms.

V. Numerical Results

The factorization algorithm has been implemented and extensive numerical simulations have been performed. Thereby, $\boldsymbol{H}$ is an $N \times K$ i.i.d. random zero-mean unit-variance complex Gaussian matrix. The aim is to assess which gains can be attributed to which factorization criterion or algorithm. The proposed straight-forward algorithm gives the same results as the recent one in [3]. For K up to 8 our strategy is faster for most of the realizations; however for a few matrices it requires significant higher complexity. A detailed complexity evaluation is beyond the scope of the present paper.

First, in Fig. 3 the cumulative distribution function of the list size (number of columns in $\boldsymbol{U}$) is plotted. Please note, all apparently linearly dependent vectors (those multiplied by -1, j, and $-$j) are not added to the list in `getlist`. It can be seen that for practical values of K the list size is small to moderate and can easily be handled. The chosen SNR is almost the worst case; for large SNR $\boldsymbol{L}\boldsymbol{L}^{\mathsf{H}} \approx \left(\boldsymbol{H}^{\mathsf{H}}\boldsymbol{H}\right)^{-1}$. For small SNRs smaller list sizes are obtained as $\boldsymbol{L}\boldsymbol{L}^{\mathsf{H}} \approx \frac{\sigma_x^2}{\sigma_n^2}\boldsymbol{I}$ and thus $\boldsymbol{Z} = \boldsymbol{I}$ is optimum.

Second, Tab. II summarizes the distribution of the determinant of $\boldsymbol{Z}$ over 10^6 channel realizations. As the dimension of the channel matrix increases, the number of channels where

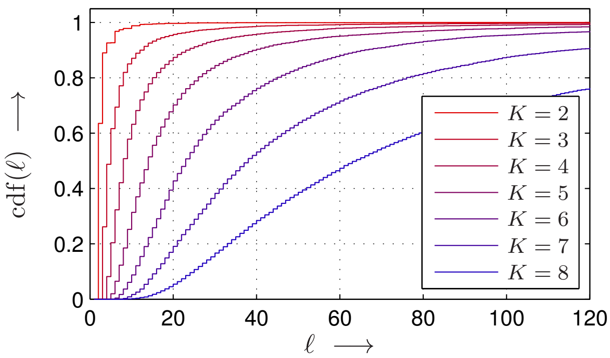

Fig. 3. Cumulative distribution function (cdf) of the list size. I.i.d. channel matrix $\boldsymbol{H}$ with $K = N$. $\sigma_x^2/\sigma_n^2 \,\widehat{=}\, 20$ dB.

TABLE II
Distribution of $\lvert\det(\boldsymbol{Z})\rvert$. $\sigma_x^2/\sigma_n^2 \,\widehat{=}\, 20$ dB.

$\lvert\det(\boldsymbol{Z})\rvert =$	1	$\sqrt{2}$	2	$\sqrt{5}$
$K = N = 2$	100 %	—	—	—
$K = N = 3$	99.8 %	0.2 %	—	—
$K = N = 4$	99.0 %	1.0 %	—	—
$K = N = 5$	97.5 %	2.4 %	0.005 %	—
$K = N = 6$	95.4 %	4.5 %	0.03 %	0.003 %
$K = N = 7$	92.7 %	7.1 %	0.15 %	0.02 %
$K = N = 8$	89.3 %	10.2 %	0.39 %	0.06 %

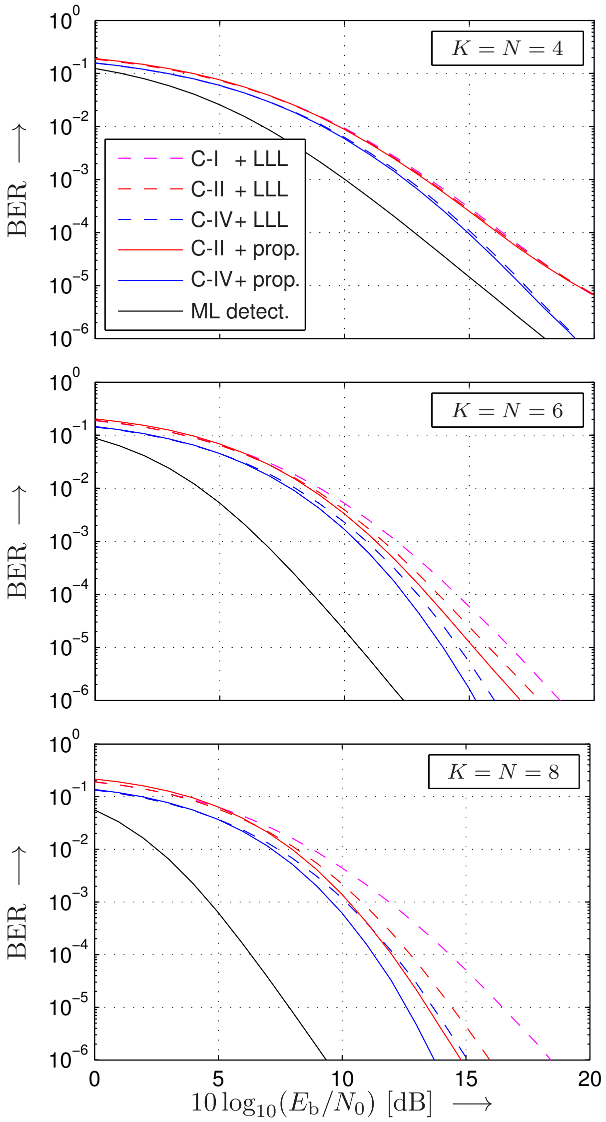

Fig. 4. Bit error rate over the SNR for uncoded transmission. 16QAM transmission. Variation of the optimization criterion (colors) and the factorization algorithm (dashed vs. solid).

$|\det(\boldsymbol{Z})| > 1$ increases. However, only for $K > 6$ non-unimodular matrices are optimum for a significant portion of channels.

Finally, bit-error-rate curves for uncoded transmission are depicted in Fig. 4. 16QAM signaling is used and the SNR is normalized to $\frac{E_b}{N_0} = \frac{\sigma_a^2}{\sigma_n^2 \log_2(16)}$. The factorization criterion and the factorization algorithm are varied; in each case the linear receiver frontend $\boldsymbol{F}$ is adjusted according to the MMSE criterion. For reference, ML detection is included.

Obviously, C-I together with the LLL algorithm has the worst performance. Using C-II gives better results (cf. [15]), best performance is obtained when applying C-IV; still the LLL is used, thus $\boldsymbol{Z}$ is unimodular. Using the proposed algorithm which relaxes the constraint on the determinant of $\boldsymbol{Z}$ some additional gain is possible. This gain, as already can be deduced from Tab. II, increases when K gets larger. Compared to classical LRA equalization using C-I and the LLL, gains in the order of 5 dB are possible for $K = N = 8$ by replacing the criterion and the factorization algorithm. Thereby, however, the LRA receiver structure can be utilized as it is—the constraints on the constellation and the code design in IF can be avoided.

VI. Summary and Conclusions

The tight relation between LRA and IF schemes has been highlighted and a new, optimum factorization algorithm has been proposed. We have restricted ourselves to linear equalization of the residual part. The extension to successive equalization and decoding (DFE/SIC, cf. [12]) is immediately possible. Moreover, the transformation to transmitter-side precoding, dual to receiver-side equalization, is also directly possible, cf. [8], [7].

References

[1] E. Agrell, T. Eriksson, A. Vardy, K. Zeger. Closest Point Search in Lattices. *IEEE Tr. Information Theory*, pp. 2201–2214, Aug. 2002.

[2] J.H. Conway, N.J.A. Sloane. *Sphere Packings, Lattices and Groups*. Springer Verlag, New York, Berlin, 3rd edition, 1999.

[3] L. Ding, K. Kansanen, Y. Wang, J. Zhang. Exact SMP Algorithms for Integer Forcing Linear MIMO Receivers. *IEEE Tr. Wireless Communications*, pp. 6955–6966, Dec. 2015.

[4] C. Feng. D. Silva, F.R. Kschischang. An Algebraic Approach to Physical-Layer Network Coding. *IEEE Tr. Information Theory*, pp. 7576–7596, Nov. 2013.

[5] R.F.H. Fischer, C. Windpassinger, C. Stierstorfer, C. Siegl, A. Schenk, Ü. Abay. Lattice-Reduction-Aided MMSE Equalization and the Successive Estimation of Correlated Data. *AEÜ—Int. Journal of Electronics and Communications*, pp. 688–693, Aug. 2011.

[6] Y.H. Gan, C. Ling, W.H. Mow. Complex Lattice Reduction Algorithm for Low-Complexity Full-Diversity MIMO Detection. *IEEE Tr. Signal Processing*, July 2009.

[7] W. He, B. Nazer, S. Shamai. Uplink-Downlink Duality for Integer-Forcing. *IEEE Int. Symp. Information Theory*, pp. 2544–2548, 2014.

[8] S.-N. Hong, G. Caire. Compute-and-Forward Strategies for Cooperative Distributed Antenna Systems. *IEEE Tr. Information Theory*, pp. 5227–5243, Sept. 2013.

[9] K. Huber. Codes over Gaussian Integers. *IEEE Tr. Information Theory*, pp. 207–216, Jan. 1994.

[10] A.K. Lenstra, H.W. Lenstra, L. Lovász. Factoring Polynomials with Rational Coefficients, *Math. Annalen*, pp. 515–534, 1982.

[11] B. Nazer, M. Gastpar. Compute-and-Forward: Harnessing Interference Through Structured Codes. *IEEE Tr. Information Theory*, pp. 6463–6486, Oct. 2011.

[12] O. Ordentlich, U. Erez, B. Nazer. Successive Integer-Forcing and its Sum-Rate Optimality. *Allerton Conference*, pp. 282–292, 2013.

[13] A. Sakzad, J. Harshan, E. Viterbo. Integer-Forcing MIMO Linear Receivers Based on Lattice Reduction. *IEEE Tr. Wireless Communications*, pp. 4905–4915, Oct. 2013.

[14] A. Sakzad, J. Harshan, E. Viterbo. On Complex LLL Algorithm for Integer Forcing Linear Receivers. *Australian Comm. Theory Workshop*, pp. 13–17, 2013.

[15] M. Taherzadeh, A. Mobasher, A.K. Khandani. LLL Reduction Achieves the Receive Diversity in MIMO Decoding. *IEEE Tr. Information Theory*, pp. 4801–4805, Dec. 2007.

[16] C. Windpassinger, R.F.H. Fischer. Low-Complexity Near-Maximum-Likelihood Detection and Precoding for MIMO Systems using Lattice Reduction. *IEEE Information Theory Workshop*, pp. 345–348, 2003.

[17] C. Windpassinger. *Detection and Precoding for Multiple Input Multiple Output Channels*. Dissertation, Erlangen, June 2004.

[18] D. Wübben, R. Böhnke, V. Kühn, K.D. Kammeyer. Near-Maximum-Likelihood Detection of MIMO Systems using MMSE-Based Lattice Reduction. *IEEE Int. Conf. Communications*, pp. 798–802, 2004.

[19] H. Yao, G. Wornell. Lattice-Reduction-Aided Detectors for MIMO Communication Systems. *IEEE Glob. Telec. Conf.*, pp. 424–428, 2002.

[20] J. Zhan, B. Nazer, U. Erez, M. Gastpar. Integer-Forcing Linear Receivers. *IEEE Tr. Information Theory*, pp. 7661–7685, Dec. 2014.