

Short block length trellis-based codes for interference channels

ISSN 1751-8628
 Received on 11th April 2018
 Revised 27th June 2018
 Accepted on 11th July 2018
 E-First on 11th October 2018
 doi: 10.1049/iet-com.2018.5198
 www.ietdl.org

Mehdi Dabirnia¹ ✉, Shahrouz Sharifi², A Korhan Tanc³, Tolga M. Duman⁴

¹Department of Information and Communication Technologies, Universitat Pompeu Fabra, 08018 Barcelona, Spain

²Maxlinear, Carlsbad, 92008 CA, USA

³Department of Biomedical Engineering, Biruni University, 34020 Istanbul, Turkey

⁴Department of Electrical and Electronics Engineering, Bilkent University, 06800 Ankara, Turkey

✉ E-mail: mehdi.dabirnia@upf.edu

Abstract: In this study, the authors consider Gaussian interference channels and fading interference channels, and design short block length codes based on trellis-based constructions. For both joint maximum likelihood (JML) decoding and single user minimum distance decoding, they obtain error-rate bounds to assess the code performance. Then they employ the obtained bounds for code design and present several design examples. For the case of quasi-static fading, they note that while the simple version of the derived bound is not sufficiently tight for code search purposes, one can obtain a tight performance bound with a higher complexity that can be used for a theoretical performance investigation. For the Gaussian case under JML decoding, they show that the newly designed codes provide significant improvements over point-to-point (P2P) trellis-based codes and off-the-shelf low density parity check codes. They also demonstrate that, for the case of independent and identically distributed fading, the best codes obtained by performing code search are P2P optimal ones, which is also verified by simulation results.

1 Introduction

Interference channel (IC) is a physical model of a shared medium over which several sender–receiver pairs communicate. During the last several decades, the characterisation of such channels has been investigated thoroughly and many information theoretic problems have been addressed, see, e.g. [1, 2]. However, regarding practical code designs, only a limited number of studies exist; see [3–5] for the design of asymptotically long block length low density parity check (LDPC) codes for the two-user Gaussian interference channels (GICs).

It is suggested in [5] that, in the large block length regime, performance close to the capacity or rate-region boundaries can be attained by using optimised irregular LDPC codes. However, the asymptotic design conjectures are not valid in the short block length regime, and short block length codes picked up from the optimised ensembles are rather inferior. This is due to the fact that, it is not possible to match the optimised degree distributions exactly, and also small cycles in the Tanner graph cannot be eliminated completely. To construct short block length LDPC codes, several approaches specifically based on avoiding small stopping sets and girth-conditioning are proposed in the literature [6–9], and performance enhancements over random constructions are reported for point-to-point (P2P) transmissions. However, these P2P constructions do not usually perform well over multi-user channels. Moreover, there do not exist similar (specifically tailored) code designs for these scenarios.

Motivated by the practical applications demanding low hardware complexity and small decoding delay, in this work, we study short block length code design for the two-user ICs. We focus on the utilisation of trellis-based codes for this purpose since such codes can achieve superior performance in P2P communications and space–time coding scenarios [10]. Also, it is possible to evaluate error-rate bounds of such codes efficiently leading to proper code designs. Recently, Ozcelikkale and Duman [11] have designed short block length trellis-based codes for the two-user Gaussian multiple access channels (GMACs) and it has been reported that the designed codes outperform the P2P optimal ones. In this study, we address the short block length code design

problem for the case of two-user GICs. We derive error-rate bounds for trellis-based codes by using the union bound techniques and employ these bounds in the code design. Through several numerical examples, we also study the performance of our newly designed codes.

In addition to GICs with fixed channel coefficients, we also consider two different fading scenarios, namely, independent and identically distributed (i.i.d.) fading and quasi-static fading cases, and develop average union bound expressions on the error rates to obtain a simple performance characterisation. We study the tightness of the derived bounds through numerical examples, and provide several error-rate performance results for the designed codes. We demonstrate that for GICs with joint maximum-likelihood (JML) decoding, the newly designed codes significantly outperform the P2P optimal ones and off-the-shelf LDPC codes with the same block lengths. We also show that for the case of i.i.d. fading IC, the best codes obtained by performing code search are also P2P optimal. For the case of quasi-static fading, we derive a simple version of the union bound, however, it is not sufficiently tight for code search purposes. Hence, we also provide a tight performance bound with a higher complexity that is useful for a theoretical assessment of the code performance.

The rest of the paper is organised as follows. In Section 2, we introduce the system description for the two-user ICs. In Section 3, we utilise the error-rate bounds obtained for GMACs and Gaussian broadcast channels (GBCs) from the existing literature and develop a framework towards designing trellis-based codes for the two-user Gaussian and fading ICs. We also present the specific design procedures for different decoding strategies in this section. In Section 4, we provide various code design examples and comprehensive performance evaluations. Finally, we conclude the paper in Section 5.

2 System model

The block diagram of a two-user IC with Gaussian noise is given in Fig. 1. For receiver i , the received signal vector with length- n can be expressed as

$$\mathbf{y}_i = \boldsymbol{\alpha}_i \mathbf{c} + \mathbf{z}_i, \quad i = 1, 2. \quad (1)$$

Here \mathbf{c} is the binary phase shift keying-modulated codeword matrix which is defined as follows:

$$\mathbf{c} = \begin{bmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \end{bmatrix} = \begin{bmatrix} c_{1,1} & c_{1,2} & \dots & c_{1,n} \\ c_{2,1} & c_{2,2} & \dots & c_{2,n} \end{bmatrix}, \quad (2)$$

where \mathbf{c}_1 and \mathbf{c}_2 are the codewords utilised at transmitter 1 and transmitter 2, respectively. The one-by-two vector $\boldsymbol{\alpha}_i = [\alpha_{i1} \ \alpha_{i2}]$ denotes the channel gains vector for receiver i where α_{ji} is a complex number (with real and imaginary components α_{ji}^R and α_{ji}^I , respectively) denoting the channel gain from the transmitter j to the receiver i . \mathbf{z}_i represents the vector of i.i.d. zero-mean complex Gaussian noise samples having variance of $N_0/2$ at receiver i . For the GIC case, the channel gains are fixed values (and naturally, they are known both at the transmitter and the receiver). The signal-to-noise ratio (SNR) and interference-to-noise ratio (INR) at receiver i are defined as

$$\text{SNR}_i = \frac{|\alpha_{ii}|^2}{N_0}, \quad \text{INR}_i = \frac{|\alpha_{ji}|^2}{N_0}, \quad (3)$$

for $i, j = 1, 2$. For a GIC, based on the signal and interference levels, the interference can be classified as strong (if $\text{INR}_i > \text{SNR}_j$), weak (if $\text{SNR}_i > \text{INR}_j$), or mixed (if $\text{INR}_i > \text{SNR}_j$, $\text{INR}_j < \text{SNR}_i$) with $i \neq j$ [12].

We consider two models for the case of fading ICs. In the first one, the fading coefficients α_{ji} are modelled as samples of independent zero-mean circularly symmetric complex Gaussian random variables with variance $\sigma_{ji}^2/2$ per dimension, i.e. i.i.d. for each transmission instant. This case can be considered as fast fading. In the second model, we assume that the fading is slow, and specifically, the fading coefficients are constant throughout each codeword. This case can be considered as quasi-static as it does not offer time diversity. While for the quasi-static case, one can consider the outage probability, we still consider the average error probability as the performance metric as also done in the space-time coding literature [13].

For both fast and quasi-static fading scenarios, we assume that perfect channel state information (CSI) is available at the receiver, however, no CSI is available at the transmitter. The SNR and INR at receiver i are defined as

$$\text{SNR}_i = \frac{\sigma_{ii}^2}{N_0}, \quad \text{INR}_i = \frac{\sigma_{ji}^2}{N_0}, \quad (4)$$

with $i, j = 1, 2$.

We consider two decoding techniques for ICs. Namely these are JML decoding and single user decoding (SUD).

2.1 JML decoding

In this technique, both desired and interfering signals are decoded based on the ML criterion together at the receiver i , which can be expressed as [14]

$$(\mathbf{c}_1^{(\text{JML})}, \mathbf{c}_2^{(\text{JML})})_i = \arg \min_{(\hat{\mathbf{c}}_1, \hat{\mathbf{c}}_2)} \|\mathbf{y}_i - \boldsymbol{\alpha}_{1i} \hat{\mathbf{c}}_1 - \boldsymbol{\alpha}_{2i} \hat{\mathbf{c}}_2\|^2, \quad i = 1, 2. \quad (5)$$

Here $\|\cdot\|$ is the Euclidean norm of the vector, and the minimisation is taken over both codebooks. We refer to this as 'joint ML' decoding since the codewords of the two users are decoded jointly at each receiver, from which the receivers obtain their own codeword estimates.

2.2 Single user (SU) decoding

In this method, only the desired signal is decoded and the interfering signal is treated as noise. According to the minimum distance criterion, the decoding rule can be expressed as

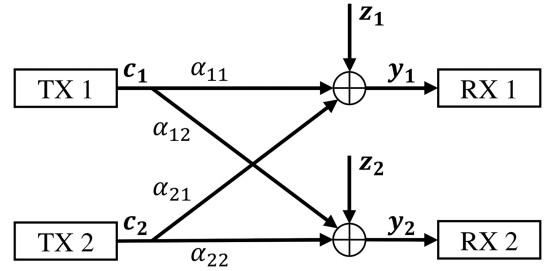


Fig. 1 Block diagram of a two-user GIC

$$\mathbf{c}_i^{(\text{SU})} = \arg \min_{\hat{\mathbf{c}}_i} \|\mathbf{y}_i - \boldsymbol{\alpha}_{ii} \hat{\mathbf{c}}_i\|^2, \quad i = 1, 2, \quad (6)$$

where the minimisation is performed just over the codebook for the desired message. Clearly, the SUD is inferior to JML in terms of performance, however, it is much simpler to implement.

3 Performance analysis

In this section, we derive error-rate bounds on the system performance by utilising the union bound techniques. We note that while there are other more sophisticated and improved bounds (such as those based on the Gallager bound [15, 16] and tangential sphere bound [17]); however, they have higher computational complexities, which make them unsuitable for the code design purposes, hence they are not explored here.

3.1 GIC with strong interference

We first consider GIC with strong interference and assume JML decoding since it is the optimal decoding strategy. We exploit the performance analysis approach utilised in [11] for GMACs under JML decoding and derive performance bounds for our case. We outline the procedure in the following.

Regarding both receivers, the total frame error rate can be upper-bounded using the union bound technique as

$$P_e \leq \frac{1}{|\mathcal{C}|} \sum_{\mathbf{c}} \sum_{\hat{\mathbf{c}} \neq \mathbf{c}} (P_{e,1}(\mathbf{c}, \hat{\mathbf{c}}) + P_{e,2}(\mathbf{c}, \hat{\mathbf{c}})), \quad (7)$$

where \mathcal{C} is the set of codeword pairs \mathbf{c} and $|\cdot|$ is the cardinality of the set. $P_{e,i}(\mathbf{c}, \hat{\mathbf{c}})$ denotes the pairwise error probability for receiver i . This error probability examines the case where the codeword pair $\hat{\mathbf{c}}$ is more likely to be received instead of the transmitted codeword pair \mathbf{c} . We can express $P_{e,i}(\mathbf{c}, \hat{\mathbf{c}})$ as

$$P_{e,i}(\mathbf{c}, \hat{\mathbf{c}}) = Q\left(\sqrt{\frac{E d_i^2(\mathbf{c}, \hat{\mathbf{c}})}{2N_0}}\right), \quad (8)$$

where $Q(x) = (1/\sqrt{2\pi}) \int_x^\infty \exp(-t^2/2) dt$, and $E d_i^2(\cdot, \cdot)$ is the squared Euclidean distance function computed at receiver i , i.e.

$$\begin{aligned} E d_i^2(\mathbf{c}, \hat{\mathbf{c}}) &= \boldsymbol{\alpha}_i \mathbf{D}_{\mathbf{c}, \hat{\mathbf{c}}} \boldsymbol{\alpha}_i^\dagger \\ &= |\alpha_{ii}|^2 d_{11} + 2(\alpha_{1i}^R \alpha_{2i}^R + \alpha_{1i}^I \alpha_{2i}^I) d_{12} + |\alpha_{2i}|^2 d_{22}. \end{aligned}$$

Here ' \dagger ' is the Hermitian conjugate and $\mathbf{D}_{\mathbf{c}, \hat{\mathbf{c}}}$ denotes the 2-by-2 codeword difference matrix, which is given by

$$\mathbf{D}_{\mathbf{c}, \hat{\mathbf{c}}} = (\mathbf{c} - \hat{\mathbf{c}})(\mathbf{c} - \hat{\mathbf{c}})^T = \begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \end{bmatrix}. \quad (9)$$

In the computation of (7) one major difficulty is that, the codeword difference matrix should be enumerated for all possible $\hat{\mathbf{c}}$ and \mathbf{c} pairs. However, this matrix can be calculated efficiently for certain cases, including the systems that employ convolutional coding. To count the multiplicities of different $\mathbf{D}_{\mathbf{c}, \hat{\mathbf{c}}}$ terms in the bound calculations, we follow the approach provided in [11]. Then we

perform the code optimisation by searching over the code pairs minimising (7) calculated at a specific SNR.

We consider a joint trellis diagram having states labelled as (s_1, s_2) , where s_i denotes the trellis state for the i th user's code. It follows that the joint trellis diagram has $n_{s_1} \times n_{s_2}$ states where n_{s_i} represents the number of states for the i th user's code. To take into account all possible \hat{c} and c pairs, we form a product trellis diagram having states labelled as $(s_1, s_2, \hat{s}_1, \hat{s}_2)$, where s_i and \hat{s}_i denote the states of the trellises for the codewords c_i and \hat{c}_i , respectively. To count all possible $D_{c,\hat{c}}$ matrices, we define a state transition matrix $S_{1,2}$ for the product trellis. If the transition from state k to state l is not allowed, the element of the state transition matrix for the k th row and the l th column is zero. Otherwise the element is in the form

$$[S_{1,2}]_{k,l} = D_{11}^{q_{11}^{k,l}} \times D_{12}^{q_{12}^{k,l}} \times D_{22}^{q_{22}^{k,l}}, \quad k, l = 1, \dots, (n_{s_1} \times n_{s_2})^2. \quad (10)$$

Here D_{11} , D_{12} , D_{22} are auxiliary variables, which are used to list various kinds of errors between the two codeword pairs [13]. For the transition from state k to state l , the exponent $q_{i,i}^{k,l}$ denotes the number of indices in which the i th users' correct and erroneous labels disagree with each other. On the other hand, the exponent $q_{i,j}^{k,l}$ with $i \neq j$ denotes the number of agreements between the previous indices for the i th and j th users. All these variables are utilised to evaluate the contributions of the state transitions to the entries of the $D_{c,\hat{c}}$.

We can obtain tighter bounds by performing expurgation of the union bound [18], which takes into account only the basic error events caused by paths diverging from the correct path in only one segment of the trellis diagram. To count such error events effectively, we utilise the notion of an error state which was originally developed in [13]. The transition to the error state occurs only when the two paths that diverged before merge for the first time. Moreover, the transition from the error state is only to itself. By computing the L th power of $S_{1,2}$, the total multiplicities of all possible $D_{c,\hat{c}}$ matrices can be obtained for the L stages of the joint trellis state transitions. We also note that, when trellis termination is taken into account, the final state transition matrix calculations should be modified accordingly [13].

Although the presented approach is rather simple, the exact calculation of the upper bound is still not feasible due to its computational complexity. Hence, we cannot utilise the presented approach directly in the code design. For GMAC, the authors of [11] assume a shorter frame length than the desired frame length in the code design, and this simplification does not disturb the performance significantly since it relies on a traceback length of four to five times the constraint length of the code [19, Ch. 4]. Another simplification is performed to increase the computational efficiency where the number of terms for each entry of $S_{1,2}$ is restricted to those components (q_{ij}) with magnitudes less than a specific threshold knowing that the omitted terms do not affect the error rate bound considerably [13]. We caution that, although this approach greatly helps with the computation, the final computations based on this truncation approach should be considered as an approximation rather than being true upper-bounds on the error probabilities.

3.2 GIC with weak interference

In order to evaluate the performance under weak interference, we assume SUD which is also used in decoding for GBCs [14]. It is also possible to utilise JML decoding in a weak interference case, however, we utilise SUD due to its simplicity and tractability. Also, when the interference level is relatively low, the performance of SUD is already close to that of JML decoding.

Under SUD, the total frame error probability can be upper bounded as

$$P_e \leq P_{e,1} + P_{e,2}. \quad (11)$$

Using the union bound, the individual frame error probabilities can also be upper bounded as

$$P_{e,1} \leq \frac{1}{|\mathcal{C}_1|} \sum_{c_1} \sum_{\hat{c}_1 \neq c_1} P_{e,1}(c_1, \hat{c}_1), \quad (12)$$

where

$$P_{e,1}(c_1, \hat{c}_1) = \frac{1}{|\mathcal{C}_2|} \sum_{c_2} P_{e,1}(c_1, \hat{c}_1 | c_2) \quad (13)$$

with $|\mathcal{C}_i|$ denoting the cardinality of the i th user's codebook. $P_{e,2}$ is obtained similarly. Adopting the technique in [14], we obtain the pairwise error probability $P_{e,i}(c_i, \hat{c}_i)$ using the identity

$$P_{e,i}(c_i, \hat{c}_i | \hat{c}_j) = Q\left(\sqrt{\frac{2f(d_{ii}^{c_i}, d_{ji}^{c_i}, \alpha_{ii}, \alpha_{ji})}{N_0}}\right), \quad i, j = 1, 2, \quad i \neq j. \quad (14)$$

Similar to [14], we introduce f as

$$f(d_{ii}^{c_i}, d_{ji}^{c_i}, \alpha_{ii}, \alpha_{ji}) = \frac{(|\alpha_{ii}|^2 d_{ii}^{c_i} + (\alpha_{ii}^R \alpha_{ji}^R + \alpha_{ii}^I \alpha_{ji}^I)(d_{ii}^{c_i} - 2d_{ji}^{c_i}))^2}{|\alpha_{ii}|^2 d_{ii}^{c_i}}, \quad (15)$$

$d_{ii}^{c_i}$ is the number of indices in which c_i differs from \hat{c}_i , $d_{ji}^{c_i}$ is the number of indices in which c_i differs from both \hat{c}_i and c_j , respectively. To take into account all possible values $d_{ii}^{c_i}$ and $d_{ji}^{c_i}$, we introduce a product trellis with two state transition matrices $S'_{1,2}$ and $S'_{2,1}$. For the k th row and l th column, the entry of the $S'_{i,j}$ is computed as

$$[S'_{i,j}]_{k,l} = D_{11}^{d_{ii}^{k,l}} \times D_{12}^{d_{ji}^{k,l}}, \quad k, l = 1, \dots, (n_{s_1} \times n_{s_2})^2, \quad (16)$$

where $i, j \in \{1, 2\}, i \neq j$. The evaluated state transition matrices are employed towards listing the possible values of $d_{ii}^{c_i}$ and $d_{ji}^{c_i}$. Then by using the upper bound in (11), the performance bound is attained similar to the approach presented in [11] and in the previous subsection. To mitigate with the memory limitations, simplifications are also performed as explained in the previous section. Hence the upper bound is only approximate. Finally, the code design is performed by searching for the code pair which minimises (11).

Our design methodology can be extended to the case of $K > 2$ users. For instance, assuming that the users employ SUD, the upper bound on total frame error probability can be calculated by introducing a new product trellis with K state transition matrices and following the same design procedure as before. Regarding the practical decoding approaches, one can resort to reduced complexity trellis implementations (such as sequential decoding and M -algorithm [20]) to reduce the computational complexity with the increasing number of users.

3.3 I.I.D. fading IC

So far we have derived the union bound on the error probability for the case of GICs with strong and weak interference levels. We now consider i.i.d. fading with JML decoding and derive the union bound for this case. The channel is neither strong nor weak interference due to the fading. As we will see in Section 4, JML decoding has a better decoding performance in all Gaussian channel examples regardless of the interference level, hence we chose to utilise this approach for the fading case as well. We follow the same approach as in Section 3.1, and in addition perform averaging over the fading coefficients to obtain error rate bounds.

We can write the union bound on the probability of frame error at the i th receiver as follows:

$$\begin{aligned}
P_{\varepsilon,i} &\leq E_{\alpha_i} \left[\frac{1}{|\mathcal{C}|} \sum_{\mathbf{c}} \sum_{\hat{\mathbf{c}} \neq \mathbf{c}} P_{\varepsilon,i}(\mathbf{c}, \hat{\mathbf{c}} | \alpha_i) \right] \\
&= \frac{1}{|\mathcal{C}|} \sum_{\mathbf{c}} \sum_{\hat{\mathbf{c}} \neq \mathbf{c}} E_{\alpha_i} [P_{\varepsilon,i}(\mathbf{c}, \hat{\mathbf{c}} | \alpha_i)],
\end{aligned} \quad (17)$$

where the averaging is over the fading coefficients from both transmitters to the receiver. The average pairwise error probability, in this case, can be upper bounded as (see equation below), where $\alpha_{i,t} = \alpha_{i,t}^R + \sqrt{-1}\alpha_{i,t}^I$ is the value of α_i vector at time instant t , \mathbf{D}_t is the codeword difference matrix at time instant t , i.e.

$$\mathbf{D}_t = (\mathbf{c}_t - \hat{\mathbf{c}}_t)(\mathbf{c}_t - \hat{\mathbf{c}}_t)^T = \begin{bmatrix} d_{11,t} & d_{12,t} \\ d_{21,t} & d_{22,t} \end{bmatrix}$$

with

$$\mathbf{c}_t = \begin{bmatrix} c_{1,t} \\ c_{2,t} \end{bmatrix} \quad \text{and} \quad \Sigma = E[(\alpha_{i,t}^R)^T \alpha_{i,t}^R] = \begin{bmatrix} \frac{\sigma_{1i}^2}{2} & 0 \\ 0 & \frac{\sigma_{2i}^2}{2} \end{bmatrix}.$$

We note that $d_{12,t} = d_{21,t}$ and $d_{11,t}d_{22,t} = d_{12,t}^2$. To count the multiplicities of the codeword difference matrix for all correct-errorneous codeword pairs, we use the same procedure as in Section 3.1.

3.4 Quasi-static fading IC

In this channel model, we assume that the fading is slow, and specifically, the fading coefficients are constant throughout each

codeword. While it is possible to consider the outage probability for this case, we still consider the average error probability as the performance metric as done in the space-time coding literature [13]. Considering the quasi-static fading IC and using the JML decoding, the union bound on the probability of frame error at the i th receiver can be written as in (17) where the expectation is performed over the distribution of the fading coefficient. The corresponding average pairwise error probability, in this case, can be bounded as (see equation below) where $\mathbf{D}_{\varepsilon,\hat{\varepsilon}}$ is defined in (9) and

$$\Sigma = E[(\alpha_i^R)^T \alpha_i^R] = \begin{bmatrix} \frac{\sigma_{1i}^2}{2} & 0 \\ 0 & \frac{\sigma_{2i}^2}{2} \end{bmatrix}.$$

Using numerical examples, we have observed that although the derived union bound on the error probability under quasi-static fading is much tighter than the bound in [21], it is still not tight enough for code design purposes. We note that a similar observation has been made in [13] in the context of space-time trellis codes over a quasi-static fading channel. Therefore, we use the technique proposed in [22] to limit the conditional union bound on the error probability before averaging over the fading coefficient distribution and obtain a new upper bound as

$$P_{\varepsilon,i} \leq E_{\alpha_i} \left[\min \left[1, \frac{1}{|\mathcal{C}|} \sum_{\mathbf{c}} \sum_{\hat{\mathbf{c}} \neq \mathbf{c}} P_{\varepsilon,i}(\mathbf{c}, \hat{\mathbf{c}} | \alpha_i) \right] \right]. \quad (18)$$

Obviously, we cannot move the averaging operation inside the min function, hence, (18) needs to be calculated either by numerical

$$\begin{aligned}
E_{\alpha_i} [P_{\varepsilon,i}(\mathbf{c}, \hat{\mathbf{c}} | \alpha_i)] &= E_{\alpha_i} \left[Q \left(\sqrt{\frac{Ed_i^2(\mathbf{c}, \hat{\mathbf{c}})}{2N_0}} \right) \right] \\
&\leq E_{\alpha_i} \left[e^{-Ed_i^2(\mathbf{c}, \hat{\mathbf{c}})/4N_0} \right] \\
&= E_{\alpha_i^R} \left[e^{-\sum_{t=1}^n ((\alpha_{1i,t}^R)^2 d_{11,t} + (\alpha_{2i,t}^R)^2 d_{22,t} + 2\alpha_{1i,t}^R \alpha_{2i,t}^R d_{12,t})/4N_0} \right] E_{\alpha_i^I} \left[e^{-\sum_{t=1}^n ((\alpha_{1i,t}^I)^2 d_{11,t} + (\alpha_{2i,t}^I)^2 d_{22,t} + 2\alpha_{1i,t}^I \alpha_{2i,t}^I d_{12,t})/4N_0} \right] \\
&= \left(\prod_{t=1}^n \frac{1}{\pi \sigma_{1i} \sigma_{2i}} \int e^{-\alpha_{1i,t}^R \mathbf{D}_t (\alpha_{1i,t}^R)^T / 4N_0} \times e^{-\alpha_{1i,t}^I \Sigma^{-1} (\alpha_{1i,t}^I)^T / 2} d\alpha_{1i,t}^R \right)^2 \\
&= \prod_{t=1}^n \left(\frac{1}{\pi \sigma_{1i} \sigma_{2i}} \int e^{-\frac{1}{4N_0} \alpha_{1i,t}^R \begin{bmatrix} d_{11,t} + \frac{4N_0}{\sigma_{1i}^2} & d_{12,t} \\ d_{12,t} & d_{22,t} + \frac{4N_0}{\sigma_{2i}^2} \end{bmatrix} \alpha_{1i,t}^R} d\alpha_{1i,t}^R \right)^2 \\
&= \prod_{t=1}^n \left(\frac{4N_0}{\sigma_{1i} \sigma_{2i} \sqrt{(d_{11,t} + (4N_0/\sigma_{1i}^2))(d_{22,t} + (4N_0/\sigma_{2i}^2)) - d_{12,t}^2}} \right)^2 \\
&= \prod_{t=1}^n 4N_0 (d_{11,t} \sigma_{1i}^2 + d_{22,t} \sigma_{2i}^2 + 4N_0)^{-1},
\end{aligned}$$

$$\begin{aligned}
E_{\alpha_i} [P_{\varepsilon,i}(\mathbf{c}, \hat{\mathbf{c}} | \alpha_i)] &= E_{\alpha_i} \left[Q \left(\sqrt{\frac{Ed_i^2(\mathbf{c}, \hat{\mathbf{c}})}{2N_0}} \right) \right] \\
&\leq E_{\alpha_i} \left[e^{-Ed_i^2(\mathbf{c}, \hat{\mathbf{c}})/4N_0} \right] \\
&= E_{\alpha_i^R} \left[e^{-((\alpha_{1i}^R)^2 d_{11} + (\alpha_{2i}^R)^2 d_{22} + 2\alpha_{1i}^R \alpha_{2i}^R d_{12})/4N_0} \right] E_{\alpha_i^I} \left[e^{-((\alpha_{1i}^I)^2 d_{11} + (\alpha_{2i}^I)^2 d_{22} + 2\alpha_{1i}^I \alpha_{2i}^I d_{12})/4N_0} \right] \\
&= \left(\frac{1}{\pi \sigma_{1i} \sigma_{2i}} \int e^{-\alpha_{1i}^R \mathbf{D}_{\varepsilon,\hat{\varepsilon}} (\alpha_{1i}^R)^T / 4N_0} \times e^{-\alpha_{1i}^I \Sigma^{-1} (\alpha_{1i}^I)^T / 2} d\alpha_{1i}^R \right)^2 \\
&= \left(\frac{1}{\pi \sigma_{1i} \sigma_{2i}} \int e^{-\frac{1}{4N_0} \alpha_{1i}^R \begin{bmatrix} d_{11} + \frac{4N_0}{\sigma_{1i}^2} & d_{12} \\ d_{12} & d_{22} + \frac{4N_0}{\sigma_{2i}^2} \end{bmatrix} \alpha_{1i}^R} d\alpha_{1i}^R \right)^2 \\
&= \left(\frac{4N_0}{\sigma_{1i} \sigma_{2i} \sqrt{(d_{11} + (4N_0/\sigma_{1i}^2))(d_{22} + (4N_0/\sigma_{2i}^2)) - d_{12}^2}} \right)^2 \\
&= 16N_0^2 \left((d_{11} \sigma_{1i}^2 + 4N_0)(d_{22} \sigma_{2i}^2 + 4N_0) - d_{12}^2 \sigma_{1i}^2 \sigma_{2i}^2 \right)^{-1},
\end{aligned}$$

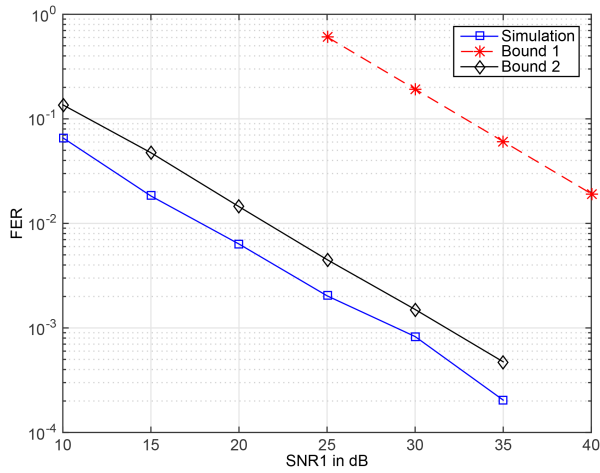


Fig. 2 Simulation and bound results of the code (5,7)/(7,5) for the quasi-static fading IC with, $\text{SNR}_1 - \text{SNR}_2 = 1$ dB, $\text{INR}_1 - \text{SNR}_2 = 2$ dB, and $\text{INR}_2 - \text{SNR}_1 = 1.5$ dB

evaluation of the multidimensional integration or by the use of Monte Carlo techniques. Although the bound in (18) is tight, due to the high complexity of its calculation, it is not a suitable metric for the code design purposes. Therefore, it is provided as a way of theoretical performance characterisation.

To verify the tightness of these two bounds, we consider an example with quasi-static fading IC with $\text{SNR}_1 - \text{SNR}_2 = 2$ dB, $\text{INR}_1 - \text{SNR}_2 = 1$ dB, and $\text{INR}_2 - \text{SNR}_1 = 2$ dB. The simulation and bound results of the optimal P2P convolutional code (5,7)/(7,5) are shown in Fig. 2. The result clearly shows that by utilising the more sophisticated bound in (18) we can obtain much a tighter bound on the error probability.

4 Code design examples

In this section, we illustrate the performance of our trellis-based code designs in comparison with that of P2P optimal trellis-based codes and off-the-shelf LDPC codes. We assume a code rate of $1/2$, a code block length of $N = 96$ and a code memory of 2. For each user, the last two information bits are utilised for trellis termination. We represent the trellis-based codes as $(m_1, n_1)/(m_2, n_2)$, where (m_i, n_i) represents the convolutional encoder of transmitter i assuming the octal format. In the code design, we follow a brute-force approach, i.e. the performance of the candidate codes is ordered according to their performance computed through the approximate bounds in (7), (11) and (17) for the cases of GIC with strong and weak interference and i.i.d. fading ICs, respectively. We truncate the number of terms for each entry of the state transition matrix at 25 in order to cope with the memory limitations and efficiently handle the matrix multiplications. For comparison, we consider P2P optimal trellis-based codes and implement (5, 7) and (7, 5) codes for the first and second users, respectively. We note that these codes have the largest minimum distance among the memory 2 codes, and they are *interleaved* versions of one another. We also consider off-the-shelf LDPC codes provided in [23] and implement the code (96.33.964)/(96.33.966). In the following, for most of the GIC examples, we consider real channel coefficients (for simplicity of exposition), which amounts to the special case of complex coefficients when both coefficients of intended and interfering signals have the same phase. On the other hand, for the fading IC examples, we consider complex channel coefficients.

4.1 GIC with strong interference

In the first example, we assume a GIC with parameters $\text{SNR}_1 - \text{SNR}_2 = 2$ dB, $\text{INR}_1 - \text{SNR}_2 = 1$ dB, and $\text{INR}_2 - \text{SNR}_1 = 2$ dB. We perform code design at $\text{SNR}_1 = 8$ dB. The minimum value of the upper-bound (7) is achieved for the code (2, 7)/(7, 5). Fig. 3a shows the decoding performance of our code design in comparison with that of P2P trellis-based codes and off-the-shelf LDPC codes. We illustrate the performance of the LDPC codes with both soft

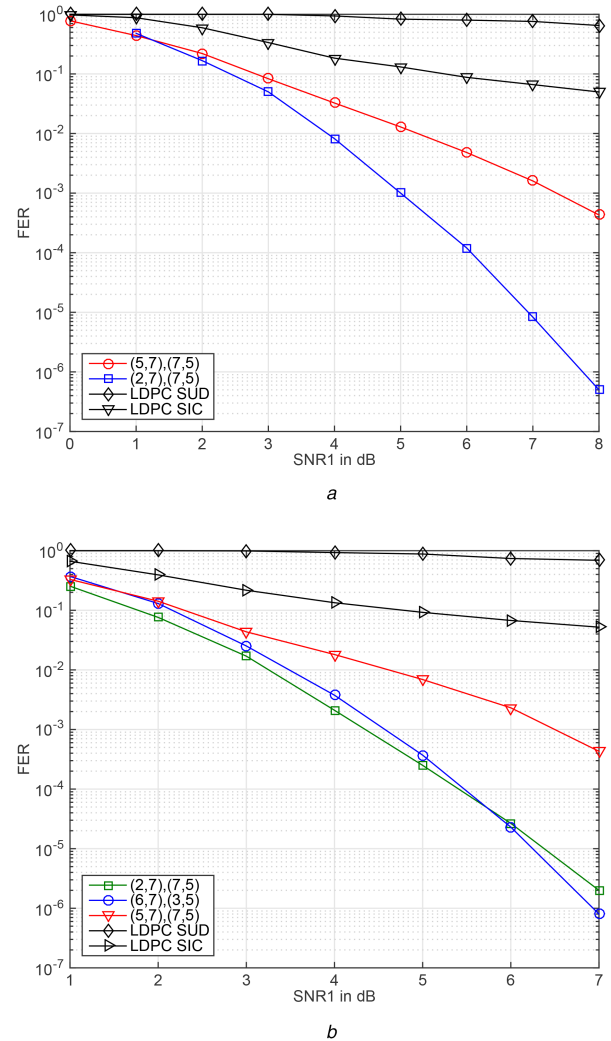


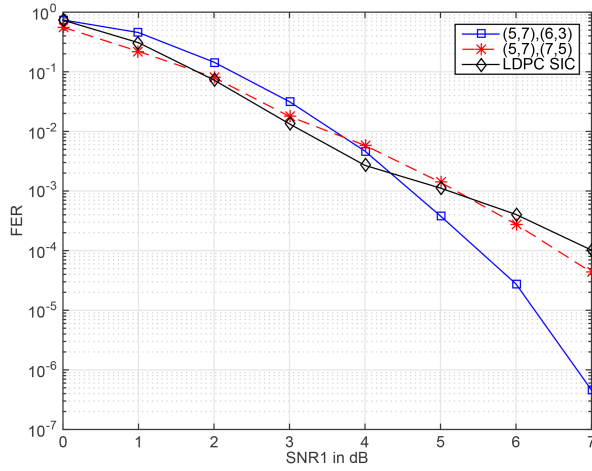
Fig. 3 Total frame error rate of trellis-based codes and LDPC codes employed for a GIC with strong interference (a) $\text{SNR}_1 - \text{SNR}_2 = 2$ dB, $\text{INR}_1 - \text{SNR}_2 = 1$ dB, and $\text{INR}_2 - \text{SNR}_1 = 2$ dB, (b) $\text{SNR}_1 - \text{SNR}_2 = 1$ dB, $\text{INR}_1 - \text{SNR}_2 = 2$ dB, and $\text{INR}_2 - \text{SNR}_1 = 1.5$ dB

interference cancellation (SIC) [24] and SUD. Notice that while the former one assumes joint decoding of both user messages in an iterative fashion, the latter one assumes trivial decoding by treating interfering user message as noise. We observe that the designed codes outperform P2P optimal ones and the LDPC codes even with SIC.

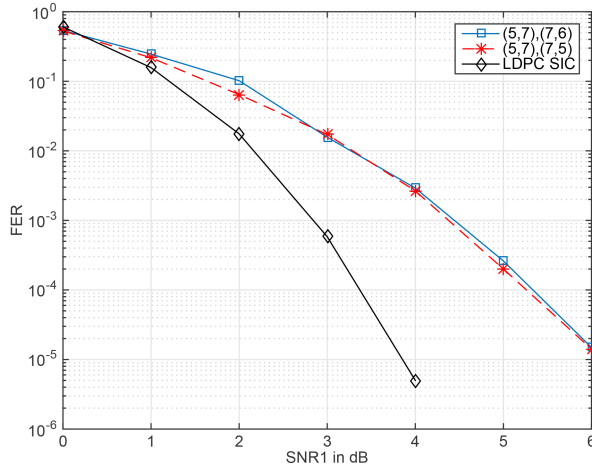
In the second example, we consider a GIC with parameters $\text{SNR}_1 - \text{SNR}_2 = 1$ dB, $\text{INR}_1 - \text{SNR}_2 = 2$ dB, and $\text{INR}_2 - \text{SNR}_1 = 1.5$ dB. We perform code design at $\text{SNR}_1 = 3$ dB and $\text{SNR}_1 = 8$ dB for which (2, 7)/(7, 5) and (6, 7)/(3, 5) minimise the upper bound in (7), respectively. For this scenario, Fig. 3b shows the decoding results for the codes under consideration. We observe that while (2, 7)/(7, 5) codes have the best performance for SNR_1 below 5.7 dB, (6, 7)/(3, 5) codes have the best performance at higher SNRs. Furthermore, both codes significantly outperform P2P optimal trellis-based codes and off-the-shelf LDPC codes.

As the third example for GIC with strong interference, we consider the previous scenario with the only change that the phases of the channel coefficients are $\angle h_{11} = \angle h_{22} = \frac{\pi}{4}$ and $\angle h_{12} = \angle h_{21} = \frac{\pi}{3}$. We perform the code design at $\text{SNR}_1 = 8$ dB for which (5, 7)/(6, 3) minimise the upper bound in (7). Fig. 4a shows the decoding performance of the codes adopted for this example. The results demonstrate that for the same SNR and INR values depending on the phases of the channel coefficients, different codes might perform better.

Finally, we consider an example with $\text{SNR}_1 - \text{SNR}_2 = 2$ dB, $\text{INR}_1 - \text{SNR}_2 = 1$ dB, $\text{INR}_2 - \text{SNR}_1 = 2$ dB, $\angle h_{11} = \angle h_{22} = \frac{\pi}{4}$, and



a



b

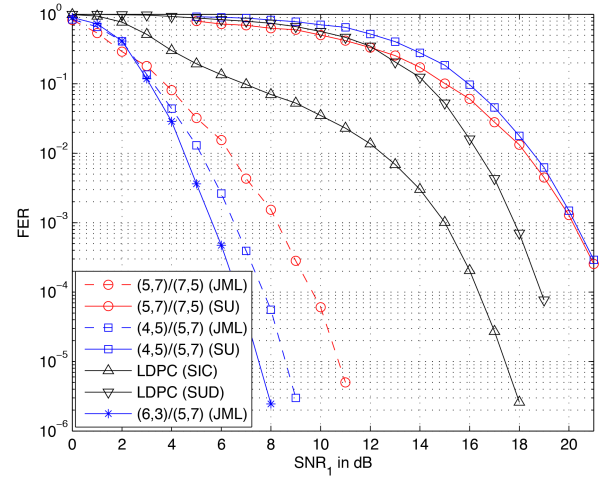
Fig. 4 Total frame error rate of trellis-based codes and LDPC codes employed for a GIC with strong interference

(a) $\text{SNR}_1 - \text{SNR}_2 = 1$ dB, $\text{INR}_1 - \text{SNR}_2 = 2$ dB, $\text{INR}_2 - \text{SNR}_1 = 1.5$ dB, $\angle h_{11} = \angle h_{22} = \frac{\pi}{4}$, and $\angle h_{12} = \angle h_{21} = \frac{\pi}{3}$, (b) $\text{SNR}_1 - \text{SNR}_2 = 2$ dB, $\text{INR}_1 - \text{SNR}_2 = 1$ dB, $\text{INR}_2 - \text{SNR}_1 = 2$ dB, $\angle h_{11} = \angle h_{22} = \frac{\pi}{4}$, and $\angle h_{12} = \angle h_{21} = \frac{\pi}{2}$

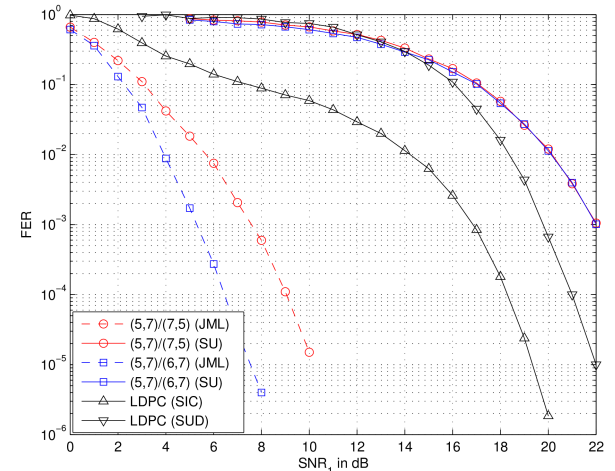
$\angle h_{12} = \angle h_{21} = \frac{\pi}{2}$. For this example, we perform code design at $\text{SNR}_1 = 8$ dB and the minimum value of the upper-bound in (7) is achieved for the code (5,7)/(7,5) and the second best one is the code with polynomial (5,7)/(7,6). In Fig. 4b, we compare the FER results of these two codes under joint decoding and LDPC codes with SIC decoding. For this example, we observe that the best performance is attained with the LDPC codes with SIC decoding. All these four examples demonstrate that, for the case of GIC with strong interference, performance enhancements over off-the-shelf codes are possible. However, there are also instances where P2P trellis-based codes or LDPC codes perform the best. We also notice that as also pointed out in [11], the poor performance of LDPC codes can be credited to the lack of joint ML decoding and suboptimal belief propagation-based decoding.

4.2 GIC with weak interference

In this section, first we consider a GIC with parameters $\text{SNR}_1 - \text{SNR}_2 = 0.5$ dB, $\text{INR}_1 - \text{SNR}_2 = -1$ dB, and $\text{INR}_2 - \text{SNR}_1 = -1.5$ dB satisfying the weak interference condition provided in Section 2. We pursue code design by minimising the upper bound in (11) at $\text{SNR}_1 = 20$ dB over the codes with memory 2 and obtain (4,5)/(5,7) codes. We also perform code optimisation by minimising the upper bound in (7) at the same SNR level, which results in the (6,3)/(5,7) code pair. In Fig. 5a, we illustrate the performance of the optimised trellis-based codes in comparison



a



b

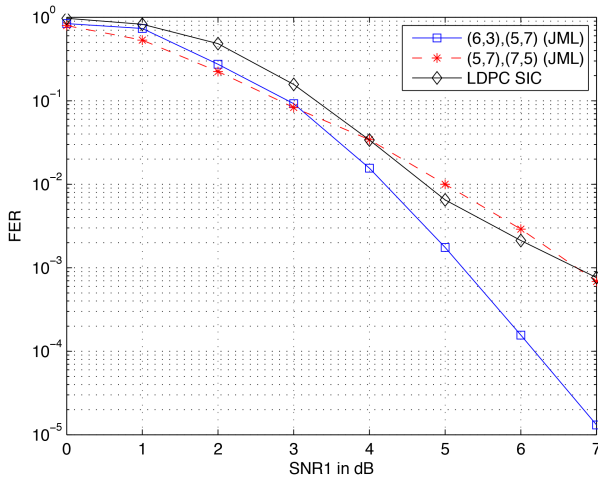
Fig. 5 Total frame error rate of trellis-based codes and LDPC codes employed for a GIC with weak interference

(a) $\text{SNR}_1 - \text{SNR}_2 = 0.5$ dB, $\text{INR}_1 - \text{SNR}_2 = -1$ dB, and $\text{INR}_2 - \text{SNR}_1 = -1.5$ dB, (b) $\text{SNR}_1 - \text{SNR}_2 = -0.75$ dB, $\text{INR}_1 - \text{SNR}_2 = -1.5$ dB, and $\text{INR}_2 - \text{SNR}_1 = -0.5$ dB

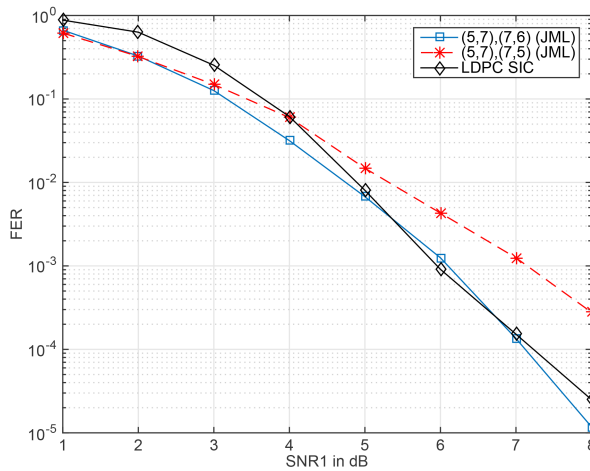
with that of the P2P optimal ones and the off-the-shelf LDPC codes. It is observed that for trellis-based codes under SUD, the design advantage is negligible. However, under JML decoding, the designed trellis-based codes provide significant enhancements when compared with the P2P optimal ones and the off-the-shelf LDPC codes.

As another example, we carry out code optimisation for a GIC with parameters $\text{SNR}_1 - \text{SNR}_2 = -0.75$ dB, $\text{INR}_1 - \text{SNR}_2 = -1.5$ dB, and $\text{INR}_2 - \text{SNR}_1 = -0.5$ dB. Assuming $\text{SNR}_1 = 20$ dB, the minimum of the upper bound in (11) is attained with the P2P optimal codes. The second ranked codes in the minimisation are (5,7)/(6,7) codes. In Fig. 5b we show the performance of these codes for the GIC under consideration. Under SUD, trellis-based codes have comparable performance and they are inferior to the off-the-shelf LDPC codes. However, under the JML decoding, optimised trellis-based codes exhibit better performance when compared with the off-the-shelf LDPC codes.

We now consider the same example as in Fig. 5a with different channel phases given by $\angle h_{11} = \angle h_{22} = \frac{\pi}{4}$ and $\angle h_{12} = \angle h_{21} = \frac{\pi}{3}$. We perform code design by utilising the upper bound in (7), which is (by using the joint decoding metric at $\text{SNR}_1 = 8$ dB) minimised by the (6,3)/(5,7) code. Fig. 6a shows the decoding results for the resulting codes. We observe that considering the same SNR and INR values but different channel phases, results in different optimised codes.



a



b

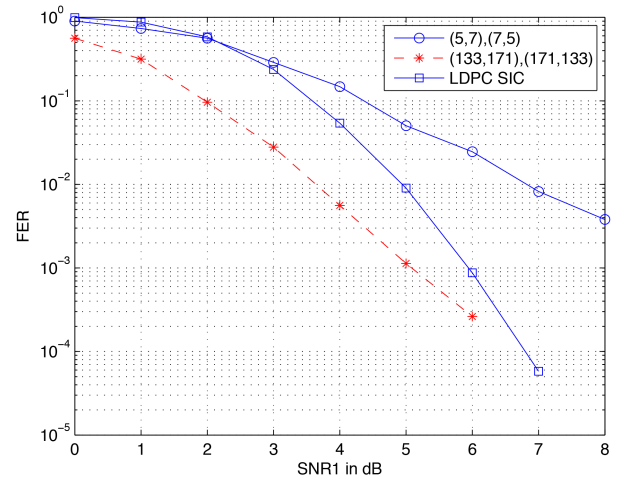
Fig. 6 Total frame error rate of trellis-based codes and LDPC codes employed for a GIC with weak interference

(a) $\text{SNR}_1 - \text{SNR}_2 = 0.5$ dB, $\text{INR}_1 - \text{SNR}_2 = -1$ dB, $\text{INR}_2 - \text{SNR}_1 = -1.5$ dB, $\angle h_{11} = \angle h_{22} = \frac{\pi}{4}$, and $\angle h_{12} = \angle h_{21} = \frac{\pi}{3}$, (b) $\text{SNR}_1 - \text{SNR}_2 = 1$ dB, $\text{INR}_1 - \text{SNR}_2 = -2$ dB, $\text{INR}_2 - \text{SNR}_1 = -3$ dB, $\angle h_{11} = \angle h_{22} = \frac{\pi}{4}$, and $\angle h_{12} = \angle h_{21} = \frac{\pi}{3}$

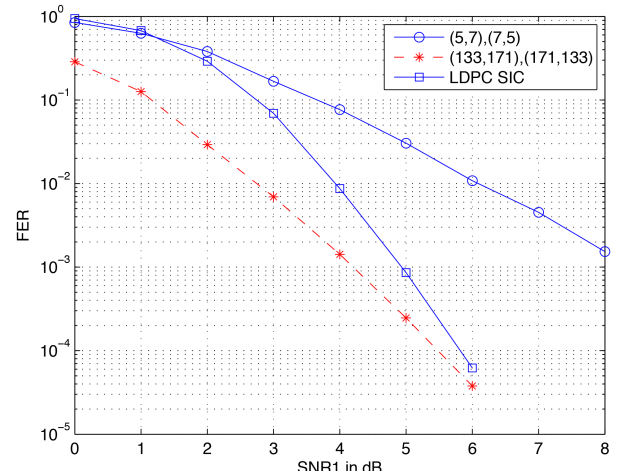
As the fourth example, we consider an instance of weak GIC with $\text{SNR}_1 - \text{SNR}_2 = 1$ dB, $\text{INR}_1 - \text{SNR}_2 = -2$ dB and $\text{INR}_2 - \text{SNR}_1 = -3$ dB. The phases are the same with the previous example. We perform code design at $\text{SNR}_1 = 8$ dB. The minimum value of the upper-bound in (7) is achieved for the code $(5, 7)/(7, 6)$. We compare the FER results of the selected code with that of best P2P code under joint decoding and LDPC codes with SIC decoding in Fig 6b. Now we observe that with the same channel phases but different SNR and INR values, different optimised codes are obtained. We also notice that for the case of weaker interference, it is hard to beat the off-the-shelf codes since the advantage of joint/successive decoding disappears as the interference weakens, and the channel more resembles like a P2P channel.

4.3 i.i.d. fading IC

As the first example, we consider an i.i.d. fading IC with $\text{SNR}_1 - \text{SNR}_2 = 2$ dB, $\text{INR}_1 - \text{SNR}_2 = 1$ dB, and $\text{INR}_2 - \text{SNR}_1 = 2$ dB. We perform code design by minimising the performance bound $P_e \leq P_{e,1} + P_{e,2}$ where $P_{e,i}$ is given by (17). We assume $\text{SNR}_1 = 8$ dB and the memory size 2. The minimum value of the upper-bound is attained with the $(5, 7)/(7, 5)$ code. We observe that the P2P optimal code among all memory 2 codes is the best one, which suggests that P2P optimal codes perform well in i.i.d. fading scenarios. Fig. 7a shows the error-rate performance of the trellis-



a



b

Fig. 7 Total frame error rate of trellis-based codes and LDPC codes employed for an i.i.d. fading IC

(a) $\text{SNR}_1 - \text{SNR}_2 = 2$ dB, $\text{INR}_1 - \text{SNR}_2 = 1$ dB, and $\text{INR}_2 - \text{SNR}_1 = 2$ dB, (b) $\text{SNR}_1 - \text{SNR}_2 = 1$ dB, $\text{INR}_1 - \text{SNR}_2 = 2$ dB, and $\text{INR}_2 - \text{SNR}_1 = 1.5$ dB

based codes and the LDPC codes. As is observed the performance of the optimal convolutional codes with memory 2 is worse than the LDPC codes, however, by increasing the memory of the code and utilising optimal P2P convolutional code with memory 6, the performance improves and beats that of the LDPC code considerably.

As another example, we carry out code optimisation for an i.i.d. fading IC with $\text{SNR}_1 - \text{SNR}_2 = 1$ dB, $\text{INR}_1 - \text{SNR}_2 = 2$ dB, and $\text{INR}_2 - \text{SNR}_1 = 1.5$ dB. As in the previous example, code design is done targeting an SNR value of $\text{SNR}_1 = 8$ dB. As in the previous example, the $(5, 7)/(7, 5)$ code minimises the upper bound. Fig. 7b shows the decoding results for the codes adopted. The LDPC code performs better than the optimal trellis based code with memory 2, however, we observe that by increasing the memory of the convolutional code we can beat the LDPC code performance. We again attribute the good performance of the LDPC code to the observation that the optimal P2P codes perform well for the i.i.d. fading case.

We also consider an i.i.d. fading IC with $\text{SNR}_1 - \text{SNR}_2 = 0.5$ dB, $\text{INR}_1 - \text{SNR}_2 = -1$ dB, and $\text{INR}_2 - \text{SNR}_1 = -1.5$ dB where the average level of the interference is lower than the signal level. The code design is pursued by minimising the union bound at $\text{SNR}_1 = 8$ dB over memory 2 codes. The optimisation process results in the code pair $(5, 7)/(7, 5)$. The performance of the optimised codes is compared with the performance of the LDPC code. Fig. 8a shows the decoding results. We observe that as in the case of the previous example while the performance of the LDPC code beats the best trellis-

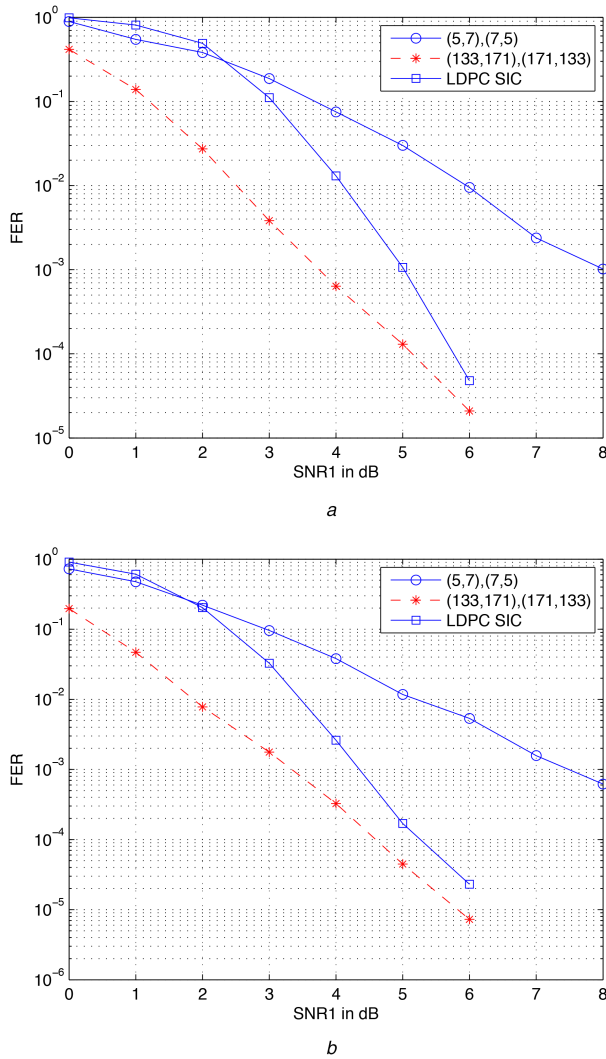


Fig. 8 Total frame error rate of trellis-based codes and LDPC codes employed for an i.i.d. fading IC
(a) $\text{SNR}_1 - \text{SNR}_2 = 0.5$ dB, $\text{INR}_1 - \text{SNR}_2 = -1$ dB, and $\text{INR}_2 - \text{SNR}_1 = -1.5$ dB,
(b) $\text{SNR}_1 - \text{SNR}_2 = -0.75$ dB, $\text{INR}_1 - \text{SNR}_2 = -1.5$ dB, and $\text{INR}_2 - \text{SNR}_1 = -0.5$ dB

based code with memory 2, a significant gain can be obtained by using a convolutional code of higher memory.

As the final example, code optimisation is carried out for an i.i.d. fading IC with $\text{SNR}_1 - \text{SNR}_2 = -0.75$ dB, $\text{INR}_1 - \text{SNR}_2 = -1.5$ dB, and $\text{INR}_2 - \text{SNR}_1 = -0.5$ dB. For this case, $(5, 7)/(7, 5)$ pair achieve the minimum of the expression $P_e \leq P_{e,1} + P_{e,2}$, where $P_{e,i}$ is given by (17). We consider all the codes with memory 2 where the bounds are computed at $\text{SNR}_1 = 8$ dB. For comparison, we also consider an LDPC code and the optimal P2P convolutional code with memory 6. Fig. 8b shows the performance of the employed codes demonstrating (as in the previous examples) that the performance of the optimal P2P convolutional code with memory 6 beats the other alternatives.

5 Conclusion

In this study, code design methods are proposed for the two-user Gaussian and fading ICs. Performance bounds based on the union bounding technique are derived and then they are utilised for optimising short block length trellis-based codes. It is shown that for GIC with strong interference, designed trellis-based codes offer performance enhancements when compared with the P2P optimal ones, and both codes outperform the off-the-shelf LDPC codes at these block lengths. For GIC with weak interference and SUD, we observe that the optimised and the P2P optimal trellis-based codes have relatively close performance, and both codes perform worse

than the off-the-shelf LDPC codes. However, with JML decoding, the optimised trellis-based codes outperform both the P2P optimal trellis-based and the off-the-shelf LDPC codes significantly. The P2P optimal trellis-based codes are found to have the best performance for i.i.d. fading ICs and perform superior to the off-the-shelf LDPC codes if their memory sizes are increased. Finally, we note that, for the case of quasi-static fading IC, a tight performance bound has been proposed for theoretical performance characterisation, which can be evaluated by using Monte Carlo techniques or numerical multidimensional integration.

6 Acknowledgments

This work was presented in part at the 2016 IEEE International Symposium on Information Theory, Barcelona, July 2016. Some of this work was done as part of S. Sharifi's PhD thesis [25]. This work was supported by the Turkish Scientific and Technological Research Council of Turkey (TUBITAK) under the grant 114E601.

7 References

- [1] Han, T., Kobayashi, K.: 'A new achievable rate region for the interference channel', *IEEE Trans. Inf. Theory*, 1981, **27**, pp. 49–60
- [2] Kramer, G.: 'Review of rate regions for interference channels'. Proc. Int. Zurich Seminar on Communications, Zurich, Switzerland, 2006, pp. 162–165
- [3] Bannatan, A., Shamaï, S., Calderbank, A.: 'Soft-decoding-based strategies for relay and interference channels: analysis and achievable rates using LDPC codes', *IEEE Trans. Inf. Theory*, 2014, **60**, pp. 1977–2009
- [4] Sharifi, S., Tanc, A.K., Duman, T.M.: 'On LDPC codes for Gaussian interference channels'. Proc. of IEEE Int. Symp. on Information Theory, Honolulu, HI, 2014, pp. 1992–1996
- [5] Sharifi, S., Tanc, A.K., Duman, T.M.: 'Implementing the Han-Kobayashi scheme using low density parity check codes over Gaussian interference channels', *IEEE Trans. Commun.*, 2015, **63**, pp. 337–350
- [6] Mao, Y., Banihashemi, A.: 'A heuristic search for good low-density parity-check codes at short block lengths'. Proc. of IEEE Int. Conf. on Communications, Helsinki, Finland, 2001, pp. 41–44
- [7] Hu, X., Eleftheriou, E., Arnold, D.M.: 'Progressive edge-growth Tanner graphs'. Proc. IEEE Global Telecommunications Conf., San Antonio, TX, 2001, pp. 995–1001
- [8] Tian, T., Jones, C.R., Villaseñor, J.D., et al.: 'Selective avoidance of cycles in irregular LDPC code construction', *IEEE Trans. Commun.*, 2004, **52**, pp. 1242–1247
- [9] Ramamoorthy, A., Wesel, R.D.: 'Construction of short block length irregular low-density parity-check codes'. Proc. IEEE Int. Conf. on Communications, Paris, France, 2004, pp. 410–414
- [10] Tralli, V., Conti, A., Chiani, M.: 'Pragmatic space-time trellis codes: GTF-based design for block fading channels', *IEEE Trans. Signal Process.*, 2011, **56**, pp. 2809–2823
- [11] Özcelikkale, A., Duman, T.M.: 'Short length trellis-based codes for Gaussian multiple-access channels', *IEEE Signal Process. Lett.*, 2014, **21**, pp. 1177–1181
- [12] Gamal, A.E., Young-Han, K.: 'Network information theory' (Cambridge University Press, Cambridge, MA, 2011)
- [13] Stefanov, A., Duman, T.M.: 'Performance bounds for space-time trellis codes', *IEEE Trans. Inf. Theory*, 2003, **49**, pp. 2134–2140
- [14] Bhat, U., Fertonani, D., Duman, T.M.: 'Approximate performance analysis for linear codes in superposition schemes over Gaussian broadcast channels', *IEEE Trans. Commun.*, 2010, **58**, pp. 2177–2182
- [15] Duman, T.M., Salehi, M.: 'New performance bounds for turbo codes', *IEEE Trans. Commun.*, 1998, **46**, pp. 717–723
- [16] Sason, I., Shamaï, S.: 'Performance analysis of linear codes under maximum-likelihood decoding: a tutorial', *Found. Trends Commun. Inf. Theory*, 2006, **3**, pp. 1–222
- [17] Polytyrev, G.: 'Bounds on the decoding error probability of binary linear codes via their spectra', *IEEE Trans. Inf. Theory*, 1994, **40**, pp. 1284–1292
- [18] Verdu, S.: 'Maximum likelihood sequence detection for inter-symbol interference channels: A new upper bound on error probability', *IEEE Trans. Inf. Theory*, 1987, **33**, pp. 62–68
- [19] Ryan, W., Lin, S.: 'Channel codes: classical and modern' (Cambridge University Press, Cambridge, MA, 2009)
- [20] Schlegel, C.B., Perez, L.C.: 'Trellis and turbo coding: iterative and graph-based error control coding' (IEEE Press, Hoboken, NJ, 2015)
- [21] Dharmawansa, K.D.P., Rajatheva, R.M.A.P.: 'On the pairwise error probability bounds of STTC over Nakagami-m fading channels'. Proc. IEEE Int. Conf. on Communications, Seoul, South Korea, 2005, pp. 2927–2931
- [22] Malkamaki, E., Leib, H.: 'Evaluating the performance of convolutional codes over block fading channels', *IEEE Trans. Inf. Theory*, 1999, **45**, pp. 1643–1646
- [23] MacKay, D.J.C.: 'Encyclopedia of sparse graph codes', 2015. Available at <http://www.inference.phy.cam.ac.uk/mackay/Codes-Files.html>
- [24] Berlin, P., Tuninetti, D.: 'LDPC codes for fading Gaussian broadcast channels', *IEEE Trans. Inf. Theory*, 2005, **51**, pp. 2173–2182
- [25] Sharifi, S.: 'On code design for interference channels'. PhD dissertation, Arizona State University, Tempe, AZ, 2015