How to Boost the Throughput of HARQ with Off-the-Shelf Codes

Mohammed Jabi, Étienne Pierre-Doray, Leszek Szczecinski, and Mustapha Benjillali

Abstract

In this work, we propose a coding strategy designed to enhance the throughput of hybrid ARQ (HARQ) transmissions over i.i.d. block-fading channels with the channel state information (CSI) unknown at the transmitter. We use a joint packet coding where the same channel block is logically shared among many packets. To reduce the complexity, we use a two-layer coding where, first, packets are first coded by the binary compressing encoders, and the results are then passed to the conventional channel encoder. We show how to optimize the compression rates on the basis of the empirical error-rate curves. We also discuss how the parameters of the practical turbo-codes may be modified to take advantage of the proposed HARQ scheme. Finally, simple and pragmatic rate adaptation strategies are developed. In numerical examples, our scheme is compared to the conventional incremental redundancy HARQ (IR-HARQ), and it yields a notable gain of $1-2$dB in the region of high throughput, where HARQ fails to provide any improvement.

Index Terms

Block Fading Channels, Coding, Dynamic Programming, HARQ, Hybrid Automatic Repeat reQuest, Rate Adaptation.

M. Jabi and L. Szczecinski are with INRS-EMT, Montreal, Canada. [e-mail: {jabi, leszek}@emt.inrs.ca].

É. Pierre-Doray is with Polytechnique de Montreal, Canada. He was with INRS-EMT when this work was carried out. [e-mail: etipdoray@gmail.com].

M. Benjillali is with the Communication Systems Department, INPT, Rabat, Morocco. [e-mail: benjillali@ieee.org].

Project partially financed by NSERC, Canada under ENGAGE grant EGP 490796–15.

I. Introduction

In this work, we propose and analyze a Hybrid ARQ protocol based on practical ("off-the-shelf") codes whose parameters are optimized to maximize the throughput for transmission over block-fading channels.

HARQ protocols are used to guarantee a reliable communication over error-prone channels, where the receiver uses the feedback to inform the transmitter about the decoding success (via positive acknowledgment (ACK) messages) or failure (via negative acknowledgment (NACK) messages). After each NACK, the transmitter starts a new HARQ round (or, a *retransmission*); this continues till the ACK message is received or the maximum allowed number of rounds is attained.

In this work, we assume that the transmitter operates without the instantaneous CSI, so the retransmissions in HARQ can be considered as an implicit adaptation to the channel states: each NACK triggers the transmission of additional parts of the codewords, and hence reduces the effective coding rate which in turn facilitates the decoding of the packet. Such a setup became "canonical" with the work [1] which demonstrated that the throughput of HARQ can approach the ergodic capacity, and this, despite a binary and per-block feedback. However, to attain the ergodic capacity, [1] assumes a very high coding rate per round, R, and a very large number of transmission rounds; since large memories at the transmitter and the receiver are then necessary, this approach is impractical.

The practical problem is thus to increase the throughput for a *given* and *finite* rate R. This problem is particularly challenging for the throughput in the vicinity of R, where the conventional HARQ fails to provide any improvement [2], [3].

To address this issue, two main venues have been explored in the literature. The first relies on the explicit reduction of the required transmission time, see e.g., [4]–[9]. However, the throughput increase is obtained with variable-length channel blocks which may be a challenge in those systems which have to keep the block size constant. The second venue harnesses the channel coding to overcome this very difficulty: the works [3], [10]–[14] keep the block size constant but increase the coding rate, i.e., the number of bits encoded in each HARQ round. This may be seen as a joint encoding of various packets into a single channel block. Then, the challenge is to define a simple (joint) encoding/decoding strategy and to optimize the coding rates.

In this work, we pursue the second venue with two main objectives, namely 1) To use off-the-shelf encoders and decoders, and 2) To optimize the transmission parameters (rates) of truncated HARQ. In fact, both objectives are interconnected since the “off-the-shelf” (i.e., simple to implement) encoders/decoders must also be accompanied by simple tools allowing us to optimize the coding rates; more on that in Sec. II-B.

The contributions of this work are the following:

- We compare the implementation feasibility of various joint coding strategies in the light of the implementation/optimization simplicity and we propose to use layer-coded HARQ (L-HARQ) which is a modified version of HARQ proposed in [11].
- We show how to calculate the throughput of truncated L-HARQ based on the off-the-shelf encoders/decoders. Our approach is applicable to any scenario where the empirical error-rate curves characterizing the decoders are known. This is different from [11] which assumed an infinite number of rounds and an idealized coding/decoding.
- We formulate and solve the problem of rate adaptation using a dynamic programming (DP) and compare the throughputs of L-HARQ to those of conventional IR-HARQ. While [13], [14] addressed the issue of rate optimization for idealized-decoding scenarios and explicitly joint (i.e., non layer) decoding, to the best of our knowledge, none of the previous works addressed the issue of rate optimization with off-the-shelf encoders/decoders.
- We show the throughput achievable with L-HARQ based on (turbo)-codes, where the optimal solution is found using solely the empirical error-rate curves of the decoder. We also discuss the issue of choosing the encoder parameters (puncturing pattern) and its relationship with the performance of L-HARQ.
- We propose and optimize a simplified version of L-HARQ.

The rest of the paper is organized as follows. We define the system model and introduce the considered retransmission schemes in Sec. II. The proposed layer-coded HARQ is defined in Sec. III, the rate optimization procedure is explained in Sec. III-D and illustrated with numerical results shown in Sec. IV. Next, we discuss the sub-optimal rate adaptation policies in Sec. V. Conclusions are drawn in Sec. VI.

II. Incremental Redundancy HARQ

In conventional IR-HARQ, a packet $\mathsf{m} \in \{0,1\}^{RN_{\mathrm{s}}}$ is encoded into K subcodewords $\boldsymbol{x}_k = \Phi_k[\mathsf{m}] \in \mathcal{X}^{N_{\mathrm{s}}}$, each composed of N_{s} complex symbols drawn from a constellation $\mathcal{X}$, where $\Phi_k[\cdot]$ are the encoders generating complementary/incremental redundancy symbols; here R denotes the coding rate per block.[1]

We consider a point-to-point transmission over a block fading channel. Each packet may require many transmission *rounds*. The kth round carries a subcodeword $\boldsymbol{x}_k$ and the received signal is given by

$$\boldsymbol{y}_k = \sqrt{\mathsf{snr}_k}\boldsymbol{x}_k + \boldsymbol{z}_k, \quad k = 1, \ldots, K, \tag{1}$$

where $\boldsymbol{z}_k$ is a zero mean, unit-variance, complex Gaussian variable modeling the noise, K is the maximum number of rounds; fixing the average energy of $\boldsymbol{x}_k$ to unity, and snr_k is the signal-to-noise ratio (SNR) at the receiver, which we assume to be perfectly known/estimated at the receiver but unknown at the transmitter.

We will model snr_k by independent, identically distributed (i.i.d.) random variables SNR_k. The derivations will be done in abstraction of a particular fading type, but in the numerical examples we consider the Rayleigh fading model, hence, SNR_k follow exponential distributions

$$p_{\mathsf{SNR}_k}(\mathsf{snr}) = \frac{1}{\overline{\mathsf{snr}}} \exp(-\mathsf{snr}/\overline{\mathsf{snr}}), \tag{2}$$

where $\overline{\mathsf{snr}}$ is the average SNR.

After the transmission in the kth round, the receiver tries to decode the packet m using all the received channel outcomes

$$\hat{\mathsf{m}}_k = \mathrm{DEC}[\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{k-1}, \boldsymbol{y}_k], \tag{3}$$

and, using a binary feedback channel, informs the transmitter whether the decoding succeeded, i.e., $\{\hat{\mathsf{m}}_k = \mathsf{m}\}$ (through an ACK) or failed (through a NACK). The transmission rounds continue until an ACK is received or the Kth round is reached.

[1] As the number of used subcodewords is random, we find it more convenient to define the rate per channel block (or per subcodeword), instead of the rate per the entire codeword R/K because transmission with such a rate is a random event.

A. Throughput

The HARQ *cycle* is a sequence of D transmission rounds related to the same packet m. In truncated HARQ, $\mathsf{D} \leq K$. Each round may be seen as a state of a Markov chain. At the end of the cycle (the "renewal", in the language of Markov processes), the receiver obtains a "reward" $\mathsf{R} \in \{0, R\}$, which is the number of correctly received bits normalized by the number of symbols in the block, N_{s}.

Since D and R are random, the long-term average throughput is calculated from the reward-renewal theorem, as the ratio between the expected reward and the expected duration [1],

$$\eta_K^{\text{ir}} = \frac{\mathbb{E}[\mathsf{R}]}{\mathbb{E}[\mathsf{D}]} = \frac{R(1-f_K)}{\sum_{k=0}^{K-1} f_k}, \tag{4}$$

which we specialized for the case of truncated HARQ [15, Sec. III] using the probability of the decoding failure after k rounds

$$f_k = \Pr\{\mathsf{NACK}_k\}, \tag{5}$$

where

$$\mathsf{NACK}_k \triangleq \Big\{\mathsf{ERR}_1 \wedge \mathsf{ERR}_2 \wedge \ldots \wedge \mathsf{ERR}_k\Big\} \tag{6}$$

and $\mathsf{ERR}_k \triangleq \{\hat{\mathsf{m}}_k \neq \mathsf{m}\}$ denotes the event of a decoding error in the kth round.

Therefore, to evaluate the throughput, which is our metric of interest, we need to calculate f_k.

In the idealized model of [1], [2], [15], it is assumed that $\mathsf{ERR}_k = \{\sum_{l=1}^{k} I(\mathsf{snr}_l) < R\}$, where $I(\mathsf{snr}_k)$ is the mutual information (MI) between the channel input and output in the kth block; then, $\mathsf{NACK}_k \iff \mathsf{ERR}_k$ is deterministically defined by the values of the SNRs.

In practice, however, the decoding errors depend also on the information sequence and the realizations of the noise. The expectation taken with respect to these variables yields the packet error rate (PER) curve of the decoder,

$$\text{PER}(\mathsf{snr}_1, \ldots, \mathsf{snr}_k; R) \triangleq \Pr\{\mathsf{ERR}_k | \mathsf{snr}_1, \ldots, \mathsf{snr}_k, R\}, \tag{7}$$

which may be obtained with Monte-Carlo simulations, keeping the SNRs and the transmission rate R fixed.

Under such a model, the events ERR_k and NACK_k are not identical. Nevertheless, we may use the approximate relation of backward decoding error implication $\mathsf{ERR}_k \implies \mathsf{ERR}_{k-1} \implies \ldots \implies \mathsf{ERR}_1$ [16], [17], which allows to write $\Pr\{\mathsf{NACK}_k\} \approx \Pr\{\mathsf{ERR}_k\}$.

B. Cross-packet coding for HARQ

As observed before, e.g., in [2], [3], [15], HARQ is particularly useful when the probability of error in the first round f_1 is high, as then the throughput can be notably increased with K. On the other hand, HARQ has negligible impact on the throughput when $f_1 \ll 1$; this is because $f_k < f_1^k \ll f_1$, and then

$$\eta_K^{\text{ir}} = \frac{R(1-f_K)}{1+f_1+\sum_{k=2}^{K-1} f_k} \approx \frac{R}{1+f_1} \approx R(1-f_1) = \eta_1,$$

where η_1 is the throughput of one-round (non-HARQ) transmission. Thus, we cannot expect any improvement in the throughput deploying conventional IR-HARQ for relatively small f_1, or—alternatively—for η_1 close to R [2], [3], [15]. In our model it also means that IR-HARQ is not useful for high average SNR.

The reason is that, due to predefined coding, the reward R is not allowed to grow even if D increases throughout the HARQ rounds. Thus, to improve the throughput, the coding should be modified so as to increase the attainable reward as the rounds advance. To this end we let the transmitter to jointly encode multiple packets into the same codeword as shown in Fig. 1

$$\boldsymbol{x}_k = \Phi_k[\mathsf{m}_{[k]}] \in \mathcal{X}^{N_\text{s}} \tag{8}$$

$$\mathsf{m}_{[k]} = [\mathsf{m}_1, \ldots, \mathsf{m}_k] \in \{0,1\}^{N_\text{s}R_{[k]}}, \tag{9}$$

where $R_{[k]}$ denotes the joint coding rate in the kth round. The throughput of such Cross-packet HARQ (XP-HARQ) is calculated as [14]

$$\eta_K^{\text{xp}} = \frac{\sum_{k=0}^{K-1} R_{[k]}(f_{k-1}-f_k)}{\sum_{k=0}^{K-1} f_k}, \tag{10}$$

where f_k is defined by (5) with $\mathsf{ERR}_k = \{\hat{\mathsf{m}}_{[k]} \neq \mathsf{m}_{[k]}\}$ being the error of the joint packet decoding, i.e.,

$$\hat{\mathsf{m}}_{[k]} = \text{DEC}[\boldsymbol{y}_1, \ldots, \boldsymbol{y}_k]. \tag{11}$$

Comparing to (4), the throughput can be increased by increasing the numerator of (10) if values of $R_{[k]}$ are optimized.

To attain (10) two main venues are adopted in the literature: i) *direct* encoding/decoding [3], [12]–[14], and ii) *layer* encoding/decoding [3], [10], [11], which have different impact on the encoding/decoding complexity.

The direct encoding considers (8) without any constraints on $\Phi_k[\cdot]$; it is thus entirely general but raises some practical concerns regarding its implementation. Namely

1) The encoder Φ_k must accept inputs $\mathsf{m}_{[k]}$ with increasing lengths, $N_\text{s}R_1 < N_\text{s}R_{[2]} < \ldots < N_\text{s}R_{[k]}$, while practical encoders are limited with regard to the input length (e.g., due to the available encoding matrix in the low-density parity-check (LDPC) codes or the way the interleavers are defined in turbo-codes);
2) Since the coding rates $R_{[k]}$ grow with k (and may even exceed $|\mathcal{X}|$), the customized design of the encoder $\Phi_k[\cdot]$ is necessary to take into account the encoders used in the previous rounds $\Phi_l[\cdot], l = 1, \ldots, k-1$.
3) The joint decoding (11) must consider concatenation of the decoders and has implementation issues of its own as can be seen, for example, in [18], [19].
4) The multi-dimensional PER curves (7), depending on the coding rates, $R_{[k]}$, would be very cumbersome to measure and store.

These issues make the direct encoding unfit to be used with “off-the-shelf” codes and thus, we will not follow this approach. Instead, we address the practical aspects with the layer-coded HARQ (L-HARQ) we explain in the following.

III. Layer-coded HARQ

L-HARQ intends to remedy the difficulties steaming from the direct application of the joint coding principle. Since we cannot escape the encoding of the message $\mathsf{m}_{[k]}$ into the codeword of length N_s, we will split it into simpler steps.

To understand the principle of L-HARQ, it is convenient to analyze a simple case of HARQ with two rounds, $K = 2$, which we next generalize to arbitrary K.

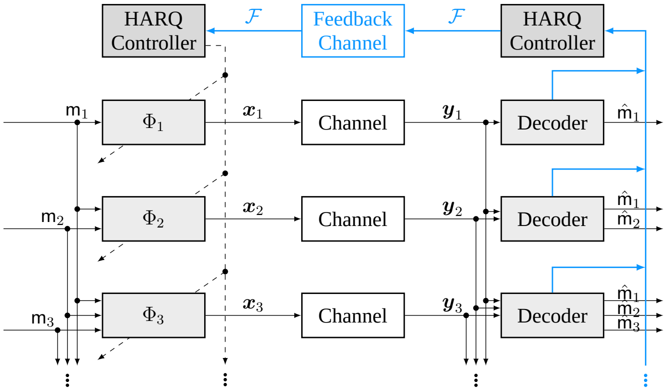

Fig. 1. Model of the joint coding/decoding HARQ transmission. The HARQ controller has to adjust the coding rates using feedback information.

A. The principle via example, $K = 2$

The first transmission is done in the same way as before. If the packet m_1 is decoded correctly, the earned reward (normalized by N_s) is given by $\mathsf{R} = R$, and a new HARQ cycle starts.

However, if the decoding fails, i.e., we observe the error event, $\mathsf{ERR}_1 = \{\hat{\mathsf{m}}_1 \neq \mathsf{m}_1\}$, the reward equals to $\mathsf{R} = 0$ and in the second round we transmit a codeword $\boldsymbol{x}_2$ obtained as

$$\boldsymbol{x}_2 = \Phi[\mathsf{m}_{[2]}] \tag{12}$$

$$\mathsf{m}_{[2]} = [\mathsf{m}_1', \mathsf{m}_2] \in \{0,1\}^{RN_\mathrm{s}}, \tag{13}$$

where $\mathsf{m}_2 \in \{0,1\}^{N_\mathrm{s}(R-\rho_1)}$ is a new packet and $\mathsf{m}_1' \in \{0,1\}^{N_\mathrm{s}\rho_1}$ is composed of $N_\mathrm{s}\,\rho_1$ bits of m_1 (we can say that m_1' is a "punctured" version of m_1).

Although, per (12), $\boldsymbol{x}_2$ is a result of a joint encoding of packets m_1 and m_2, we do not decode them jointly (which would imply using $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$). Instead, we decode the packet $\mathsf{m}_{[2]}$ using only the observation $\boldsymbol{y}_2$

$$\hat{\mathsf{m}}_{[2]} = \mathrm{DEC}[\boldsymbol{y}_2]. \tag{14}$$

If decoding error, $\mathsf{ERR}_2 = \{\hat{\mathsf{m}}_{[2]} \neq \mathsf{m}_{[2]}\}$ occurs, a zero reward, $\mathsf{R} = 0$, is earned and a new HARQ cycle starts. However, if $\mathsf{m}_{[2]}$ is decoded correctly, we know perfectly m_1', see (13). Knowing these $N_\mathrm{s}\,\rho_1$ bits of m_1, the decoder has to decode the remaining $N_\mathrm{s}(R-\rho_1)$ unknown bits using observation $\boldsymbol{y}_1$

$$\hat{\mathsf{m}}_1^{\mathsf{b}} = \text{DEC}[\boldsymbol{y}_1; \mathsf{m}_1'], \tag{15}$$

where the notation $\hat{\mathsf{m}}_1^{\mathsf{b}}$ is introduced to make difference with $\hat{\mathsf{m}}_1$ obtained via the direct decoding in the first round. This "backtrack" decoding (15) was introduced in [11]; a similar idea of successive decoding was also exploited in [3]. We define here the backtrack decoding error by $\mathsf{ERR}_1^{\mathsf{b}} = \{\hat{\mathsf{m}}_1^{\mathsf{b}} \neq \mathsf{m}_1\}$.

If the decoding si successful, $\hat{\mathsf{m}}_1^{\mathsf{b}} = \mathsf{m}_1$, the total reward is $\mathsf{R} = 2R - \rho_1$. Since $\rho_1 < R$ there is a potential for improvement over the reward $\mathsf{R} = R$ attainable in the conventional HARQ. This is because, the spirit of joint coding is followed and the second round is not merely used to convey redundancy for the packet m_1 but also to transmit a new packet m_2.

Let us generalize this approach.

B. General case

Encoding

The encoding in each round is done as follows:

$$\mathsf{m}_{[l]}' = \Phi_l^{\mathsf{b}}[\mathsf{m}_{[l]}] \in \{0,1\}^{\rho_l N_{\mathsf{s}}} \tag{16}$$

$$\mathsf{m}_{[k]} = [\mathsf{m}_{[k-1]}', \mathsf{m}_k] \in \{0,1\}^{R N_{\mathsf{s}}}, \tag{17}$$

$$\boldsymbol{x}_k = \Phi[\mathsf{m}_{[k]}], \tag{18}$$

where $\Phi_l^{\mathsf{b}}[\cdot], l = 1, \ldots, k-1$ are binary compressing encoders with binary rate $R/\rho_l > 1$, that is, we cannot recover $\mathsf{m}_{[l]}$ knowing solely $\mathsf{m}_{[l]}'$.

Since we use the channel encoder Φ which operates with a fixed coding rate R, it remains agnostic of the encoding in the step (17); this may be contrasted with the encoding using the variable rates $R_{[k]}$ required in the direct encoding. We thus remedied the two first difficulties related to encoding which are shown in the list in Sec. II-B.

We introduced in (16) the notion of the compressing encoders $\Phi_k^{\mathsf{b}}[\cdot]$ to discuss the difference with [11], where the bits $\mathsf{m}_{[k]}'$ are "parity" bits of the packet $\mathsf{m}_{[k]}$. In many practical cases, $\Phi[\cdot]$ is implemented via bit-interleaved coded modulation (BICM), i.e., it combines a binary encoder and the non-binary mapper to the symbols from the constellation $\mathcal{X}$ [20, Sec. 2.3]. Therefore the parity bits $\mathsf{m}'_{[k]}$ might be obtained as a byproduct of the binary encoding. This also means that, as an intermediate step, the encoder $\Phi[\cdot]$ must produce binary codewords longer than those necessary to produce the codewords $\boldsymbol{x}_k$. We can thus again enter into conflict with the first item in the list of practical considerations we enumerated in Sec. II-B. To avoid this pitfall we thus use the simplest possible compressor, that is the puncturer, i.e., $\mathsf{m}'_{[k]}$ is composed of the "systematic" bits of $\mathsf{m}_{[k]}$.

Beside eliminating the need for the actual binary encoding by $\Phi^{\mathsf{b}}_{k}[\cdot]$, there are other arguments in favour of the systematic $\Phi^{\mathsf{b}}[\cdot]$ we propose. First, if the message $\mathsf{m}_{[k]}$ is successfully decoded and $\mathsf{m}_{[k-1]}$ is not, we collect the reward $\mathsf{R} = R$, while with the parity encoding the reward would be only $\mathsf{R} = R - \rho_{k-1}$. Second, the backtrack decoding of the message $\mathsf{m}_{[l]}$ benefits from the presence of systematic bits, more than it would from parity bits. This is particularly true for turbo-codes that we will consider, especially that current standards recommend to puncture some of the systematic bits while encoding $\mathsf{m}_{[l]}$. These punctured bits may then be included in $\mathsf{m}'_{[l]}$ but these technical details will be discussed in Sec. IV-B.

Decoding

As for the decoding, we need of course all the observations $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_k$ to recover the messages $\mathsf{m}_1, \ldots, \mathsf{m}_k$. However, instead of explicit joint decoding that is necessary in the direct encoding/decoding, we may use a simplified layer-by-layer decoding, defined as follows:

- In the kth round, we try to decode the packet

$$\hat{\mathsf{m}}_{[k]} = \mathrm{DEC}[\boldsymbol{y}_k] \tag{19}$$

and if we succeed (i.e., $\hat{\mathsf{m}}_{[k]} = \mathsf{m}_{[k]}$), we recover the message m_k and $\mathsf{m}'_{[k-1]}$, see (17).

- With $\mathsf{m}'_{[k-1]}$ at hand, we backtrack decode the packet $\mathsf{m}_{[k-1]}$

$$\hat{\mathsf{m}}^{\mathsf{b}}_{[k-1]} = \mathrm{DEC}[\boldsymbol{y}_{k-1}, \mathsf{m}'_{[k-1]}], \tag{20}$$

where we use the fact that $\mathsf{m}'_{[k-1]}$ is now known and should be used to improve the decoding results. The decoding (20) based on $\boldsymbol{y}_{k-1}$ and $\mathsf{m}'_{[k-1]}$ is stil necessary because i) the decoding $\mathrm{DEC}[\boldsymbol{y}_{k-1}]$ failed – that is why we are in the backtrack decoding of the kth round, and ii) knowing $\mathsf{m}'_{[k-1]}$ we cannot recover $\mathsf{m}_{[k-1]}$, see the comment after (18).

- If there is no error, i.e., $\hat{\mathsf{m}}^{\mathsf{b}}_{[k-1]} = [\mathsf{m}'_{[k-2]}, \mathsf{m}_{k-1}]$, we recover the packet m_{k-1} but also can

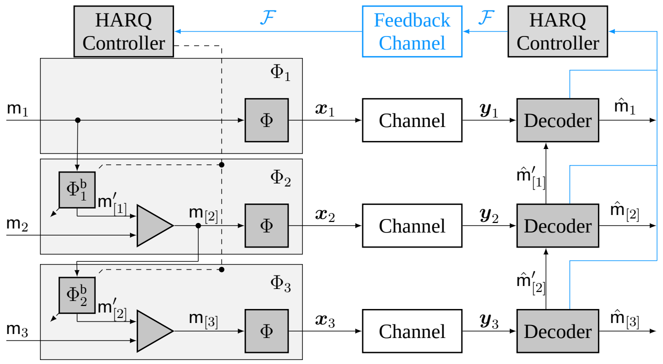

Fig. 2. Encoding and decoding in L-HARQ. The HARQ controller adjusts the rates of the puncturer $\Phi^{\mathsf{b}}_k[\cdot]$.

go back and repeat the decoding (20) with $k \leftarrow k-1$.

If the decoding steps are successful for $k-1, k-2, \dots, 1$ we recover all the packets $\mathsf{m}_{k-1}, \dots, \mathsf{m}_1$

From the implementation point of view, the receiver operation is very simple: the decoding of $\mathsf{m}_{[k-1]}$ in (20) is done using a channel outcome $\boldsymbol{y}_k$ and a priori information about $\mathsf{m}_{[k-1]}$ contained in $\mathsf{m}'_{[k-1]}$. Also, the decoding result of (20), depending on snr_{k-1} and ρ_{k-1}, is simple to describe with the PER curves as we will shown later. This is very different from the decoding (11) which depends on $\mathsf{snr}_1, \dots, \mathsf{snr}_k$ and $R_1, R_{[2]}, \dots, R_{[k]}$.

The two last issues from the list in Sec. II-B, related to the decoding, are now solved. The proposed encoding/decoding schemes are illustrated in Fig. 2, where we emphasize that the adaptation of the rate of the encoder Φ_k is done adjusting the rate of the binary compressor/puncturer Φ^{b}_k.

C. *Throughput*

To calculate the throughput

$$\eta^{\mathrm{L}}_K = \frac{\mathbb{E}[\mathsf{R}]}{\mathbb{E}[\mathsf{D}]} \tag{21}$$

we start with $K=2$.

The expected reward of L-HARQ can be obtained analyzing three events which produce non-zero reward:

- Decoding success in the first round: $\{\overline{\mathsf{ERR}}_1\}$, where $\overline{\mathsf{ERR}}$ denotes the complement of ERR; the corresponding reward is $\mathsf{R} = R$,

- Decoding success in the second round and decoding failure in the backtrack decoding: $\{\mathsf{ERR}_1 \wedge \overline{\mathsf{ERR}}_2 \wedge \mathsf{ERR}_1^{\mathsf{b}}\}$; the reward is $\mathsf{R} = R$, and
- Decoding success in the second round and decoding success in the backtrack decoding: $\{\mathsf{ERR}_1 \wedge \overline{\mathsf{ERR}}_2 \wedge \overline{\mathsf{ERR}_1^{\mathsf{b}}}\}$; the reward is $\mathsf{R} = 2R - \rho_1$.

The average reward can thus be calculated as

$$\begin{aligned}\mathbb{E}[\mathsf{R}] = \mathbb{E}\Big[& R\ \mathbb{I}\big[\overline{\mathsf{ERR}}_1\big] + R\ \mathbb{I}\big[\mathsf{ERR}_1 \wedge \overline{\mathsf{ERR}}_2\big] \\ & + (R-\rho_1)\mathbb{I}\big[\mathsf{ERR}_1 \wedge \overline{\mathsf{ERR}}_2 \wedge \overline{\mathsf{ERR}_1^{\mathsf{b}}}\big]\Big] \qquad (22)\\ = \mathbb{E}\Big[& R(1-\Pr\{\mathsf{ERR}_1\}) + \Pr\{\mathsf{ERR}_1\}(1-\Pr\{\mathsf{ERR}_2\}) \\ & \Big(R + (R-\rho_1)\big(1-\Pr\{\mathsf{ERR}_1^{\mathsf{b}}|\mathsf{ERR}_1\}\big)\Big)\Big], \qquad (23)\end{aligned}$$

where $\mathbb{I}\big[x\big] = 1$ if x is true, and $\mathbb{I}\big[x\big] = 0$ otherwise. The expectations in (22) are taken with respect to all variables affecting the decoding errors (including the message and the realizations of the noise), while (23) takes expectation with respect to SNRs $\mathsf{SNR}_1, \mathsf{SNR}_2$.

The expected number of transmissions is given by $\mathbb{E}[\mathsf{D}] = 1 + f_1$, where $f_1 = \Pr\{\mathsf{ERR}_1\}$.

For $K > 2$ we enumerate the decoding success/failure events in various rounds, we obtain the following generalization of (23)

$$\begin{aligned}\mathbb{E}[\mathsf{R}] = \mathbb{E}\Big[&\sum_{k=1}^{K}(1-\Pr\{\mathsf{ERR}_k\})\prod_{t=1}^{k-1}\Pr\{\mathsf{ERR}_t\} \\ &\Big(R + \sum_{l=1}^{k-1}(R-\rho_l)\prod_{z=l}^{k-1}\big(1-\Pr\{\mathsf{ERR}_z^{\mathsf{b}}|\mathsf{ERR}_z\}\big)\Big)\Big], \qquad (24)\end{aligned}$$

which can be expressed in a nested form as

$$\begin{aligned}\mathbb{E}[\mathsf{R}] =& \mathbb{E}_{\mathsf{SNR}_1}\Big[(1-\Pr\{\mathsf{ERR}_1\})R + \Pr\{\mathsf{ERR}_1\} \\ & \cdot \mathbb{E}_{\mathsf{SNR}_2}\Big[(1-\Pr\{\mathsf{ERR}_2\})\big(R + (R-\rho_1) \\ & \quad \cdot \big(1-\Pr\{\mathsf{ERR}_1^{\mathsf{b}}|\mathsf{ERR}_1\}\big)\big) + \Pr\{\mathsf{ERR}_2\} \\ & \cdot \mathbb{E}_{\mathsf{SNR}_3}\Big[\ \ldots\Big]\Big]\Big]. \qquad (25)\end{aligned}$$

Further we note that, due to (14), $\Pr\{\mathsf{ERR}_l\}$ depends only on the value of snr_l. Thus, the events $\mathsf{ERR}_1, \ldots, \mathsf{ERR}_l$ are independent, and f_l can be calculated as

$$f_l = \Pr\{\mathsf{ERR}_1\} \ldots \Pr\{\mathsf{ERR}_l\} = (f_1)^l. \tag{26}$$

Thus, the average number of transmission rounds is given by

$$\mathbb{E}[\mathsf{D}] = 1 + f_1 + f_1^2 + \ldots + f_1^{K-1} = \frac{1-f_1^K}{1-f_1}. \tag{27}$$

D. Optimal Rates

We are interested in finding the optimal throughput of the L-HARQ scheme, and we have to find the backtrack rates $\rho_1, \rho_2, \ldots, \rho_{K-1}$ which maximize the throughput for a given transmission rate R.

Coming back to the simple two-transmission example, the “backtrack” rate of the first round, $\rho_1 \in (0, R)$ can be defined once the decoding of m_1 fails. Consequently, it may be adapted to the *known*, but outdated, SNR snr_1.

This idea is not new, the adaptation to the outdated channel state was already proposed in previous works, e.g., [7], [9], [21], and will be exploited in Sec. III-D to optimize the throughput. Therefore, the rates ρ_k are functions of SNRs $\mathsf{snr}_1, \mathsf{snr}_2, \ldots, \mathsf{snr}_{k1-}$ and eventually of other parameters defining the transmission process.

The expected number of transmissions in (27) is independent of the backtrack rates. Consequently, maximizing the throughput is equivalent to maximizing the expected reward in (25). Denoting its optimal value by $\overline{\mathsf{R}}$, we have

$$\begin{aligned}\overline{\mathsf{R}} =& \mathbb{E}_{\mathsf{SNR}_1}\Big[\max_{\rho_1}\ (1-\Pr\{\mathsf{ERR}_1\})R + \Pr\{\mathsf{ERR}_1\} \\ & \cdot \mathbb{E}_{\mathsf{SNR}_2}\Big[\max_{\rho_2}\ (1-\Pr\{\mathsf{ERR}_2\})\Big(R + (R-\rho_1) \\ & \cdot \big(1-\Pr\big\{\mathsf{ERR}_1^{\mathsf{b}}|\mathsf{ERR}_1\big\}\,\big)\Big) + \Pr\{\mathsf{ERR}_2\} \\ & \cdot \mathbb{E}_{\mathsf{SNR}_3}\Big[\max_{\rho_3}\ \ldots\Big]\Big]\Big],\end{aligned} \tag{28}$$

and the optimum throughput of L-HARQ is thus given by

$$\eta_K^{\mathrm{L}} = \frac{(1-f_1^K)\overline{\mathsf{R}}}{1-f_1}. \tag{29}$$

$$\overline{\mathsf{R}} = \mathbb{E}_{\mathsf{SNR}_1}\big[V_1(\mathsf{SNR}_1, 0)\big], \tag{31}$$

$$V_1(\mathsf{snr}_1, J_0) = \max_{\rho_1}\big\{\big(R + J_0\big)\mathrm{PER}^{\mathsf{c}}(\mathsf{snr}_1; R) + \mathrm{PER}(\mathsf{snr}_1; R)\mathbb{E}_{\mathsf{SNR}_2}\big[V_2(\mathsf{SNR}_2, J_1)\big]\big\}, \tag{32}$$

$$\vdots$$

$$\begin{aligned} V_{K-2}(\mathsf{snr}_{K-2}, J_{K-3}) = \max_{\rho_{K-2}}\big\{\big(R{+}J_{K-3}\big)\mathrm{PER}^{\mathsf{c}}(\mathsf{snr}_{K-2}; R) &+ \mathrm{PER}(\mathsf{snr}_{K-2}; R) \\ &\times \mathbb{E}_{\mathsf{SNR}_{K-1}}\big[V_{K-1}(\mathsf{SNR}_{K-1}, J_{K-2})\big]\big\}, \end{aligned} \tag{33}$$

$$\begin{aligned} V_{K-1}(\mathsf{snr}_{K-1}, J_{K-2}) = \max_{\rho_{K-1}}\big\{\big(R{+}J_{K-2}\big)\mathrm{PER}^{\mathsf{c}}(\mathsf{snr}_{K-1}; R) &+ \mathrm{PER}(\mathsf{snr}_{K-1}; R)\mathbb{E}_{\mathsf{SNR}_K}\big[\mathrm{PER}^{\mathsf{c}}(\mathsf{SNR}_K; R)\big] \\ &\times \big(R + (R + J_{K-2} - \rho_{K-1})\mathrm{PER}^{\mathsf{c}}(\mathsf{snr}_{K-2}; R, \rho_{K-1})\big)\big\}. \end{aligned} \tag{34}$$

The nested structure of (28) allows us to rewrite it in the recursive form that is characteristic of DP in (32)–(34), where $J_0 \triangleq 0$ and

$$J_k = (R + J_{k-1} - \rho_k)\big(1 - \mathrm{PER}(\mathsf{snr}_k; \rho_k)\big) \tag{30}$$

has the meaning of an expected reward that may be collected thanks to the backtrack decoding.

We also used $\mathrm{PER}(\mathsf{snr}_k; R) = \Pr\{\mathsf{ERR}_k\}$ and $\mathrm{PER}(\mathsf{snr}; R, \rho_k) \triangleq \Pr\big\{\mathsf{ERR}^{\mathsf{b}}_k | \mathsf{ERR}_k\big\}$ to emphasize that the whole optimization depends solely on the PER curves of the decoder. For compactness, we define $\mathrm{PER}^{\mathsf{c}}(\cdot) \triangleq 1 - \mathrm{PER}(\cdot)$.

The optimization process starts with (34) and continues via a backward recursion to (31). In this way, thanks to the DP formulation, the multi-dimensional global optimization in (28) is reduced to a series of one-dimensional optimizations, and the overall computational complexity grows linearly with K. The optimization is done point-by-point over the discretized values of the variables $(\mathsf{snr}_k, J_{k-1})$, with $J_{k-1} \in \big(0, (k-1)\cdot R\big)$, and $\mathsf{snr}_k \in \mathbb{R}^+$. In the DP vocabulary, the variables $(\mathsf{snr}_k, J_{k-1})$ form a “state” at time k, the backtrack rates ρ_k are “actions” and depend on the state.

For the numerical implementation, it is convenient to truncate the PER function: we set $\mathrm{PER}(\mathsf{snr}_k) = 0$ if $\mathsf{snr}_k > \mathsf{snr}_\epsilon$; where snr_ϵ satisfies $\mathrm{PER}(\mathsf{snr}_\epsilon) = \epsilon$. In the numerical examples, we set $\epsilon = 10^{-6}$. Thus, $\rho_k(\mathsf{snr}_k, J_{k-1})$ is a 2-dimensional function, and it is non-zero only when $0 \leq J_{k-1} < (k-1)R$ and $0 \leq \mathsf{snr}_k < \mathsf{snr}_\epsilon$.

Since, in practice, only a limited number of rates is available, and by construction $\rho_k \leq R$, we use a discrete set of backtrack rates $\mathcal{A} = \{\Delta, 2\Delta_R, \ldots, R\}$, where $\Delta = R/T_R$, where the number of the available rates, T_R, may be adjusted to find a suitable compromise between the performance and the feedback requirements : only $\lceil \log_2(T_R) \rceil$ bits of feedback are needed even if the arguments $(\mathsf{snr}_k, J_{k-1})$ may be discretized with an arbitrary resolution when solving (32)–(34).

The backtrack rate functions $\rho_k(\mathsf{snr}_k, J_{k-1})$ calculated off-line using DP are stored at the receiver: after each round, the receiver observes snr_k, computes J_{k-1} via (30), and transmits the index of the optimal $\rho_k(\mathsf{snr}_k, J_{k-1}) \in \mathcal{A}$.

IV. Numerical examples

Numerical results illustrating the optimization procedure explained in Sec. III-D are here shown in two cases. First, we will use synthetic decoder curves which will allow the reader to reproduce the results. Next, we will use experimental PER curves obtained using turbo-codes to show the throughput gains in a realistic scenario and shed some light on the practical aspects of the encoding.

A. Synthetic PER curves

We will use the well-known model for the PER curve [22]

$$\mathrm{PER}(\mathsf{snr}, R) = \begin{cases} 1 & \text{if} \quad \mathsf{snr} < \mathsf{snr}_{\text{th}} \\ \exp\big(-\tilde{a}(\mathsf{snr}/\mathsf{snr}_{\text{th}} - 1)\big) & \text{if} \quad \mathsf{snr} \geq \mathsf{snr}_{\text{th}} \end{cases}; \tag{35}$$

where $I(\mathsf{snr}_{\text{th}}) = R$ and $I(x) = \log_2(1+x)$; as indicated in [23], $\tilde{a} = 4$ may be fitted to empirical curves.

To characterize the decoding errors in IR-HARQ, we use the simplified approach proposed in [24], [25], where we apply the PER curve (35)

$$\Pr\{\mathsf{ERR}_k\} \approx \mathrm{PER}(\mathsf{snr}_k^{\Sigma}, R), \tag{36}$$

and use the *aggregate* SNR given by

$$\mathsf{snr}_k^{\Sigma} = I^{-1}\Big(\sum_{l=1}^{k} I(\mathsf{snr}_l)\Big). \tag{37}$$

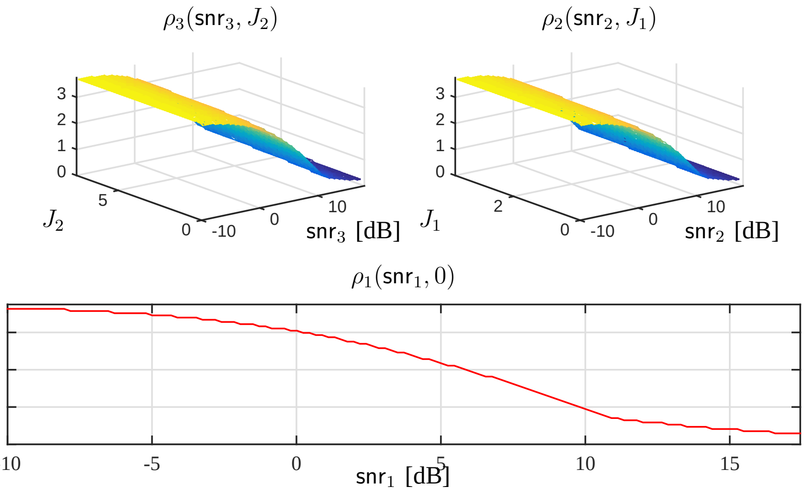

Fig. 3. L-HARQ optimal policies $\rho_k(\mathsf{snr}_k, J_{k-1})$ obtained for $R = 3.75$, $K = 4$, $\overline{\mathsf{snr}} = 15$dB, and the synthetic PER curves defined in Sec. IV-A.

Note that, setting $\tilde{a} = \infty$, we conveniently fall back on the idealized threshold decoding of [1], [2], [15].

Regarding L-HARQ, we need to characterize the decoder PER curve in the backtrack decoding. Since the effective rate of the message is decreased, we use

$$\Pr\{\mathsf{ERR}_k^{\mathsf{b}}\} = \mathrm{PER}(\mathsf{snr}_k; R - \rho_k). \tag{38}$$

From the assumption of backward errors implication [16], [17], $\mathsf{ERR}_k^{\mathsf{b}} \Rightarrow \mathsf{ERR}_k$ (which means that if the decoding fails in the backtrack phase, it must have failed in the original transmission), we have

$$\Pr\left\{\mathsf{ERR}_k \wedge \mathsf{ERR}_k^{\mathsf{b}}\right\} \approx \Pr\left\{\mathsf{ERR}_k^{\mathsf{b}}\right\}, \tag{39}$$

$$\Pr\left\{\mathsf{ERR}_k^{\mathsf{b}} | \mathsf{ERR}_k\right\} \approx \frac{\mathrm{PER}(\mathsf{snr}_k; R - \rho_k)}{\mathrm{PER}(\mathsf{snr}_k; R)}. \tag{40}$$

Furthermore, with the backward errors implication assumption, $\mathsf{ERR}_k \Rightarrow \mathsf{ERR}_{k-1} \Rightarrow \ldots \Rightarrow \mathsf{ERR}_1$, f_k is calculated as

$$f_k \approx \mathbb{E}\big[\mathrm{PER}(\mathsf{SNR}_k^{\Sigma}, R)\big], \tag{41}$$

where the expectation is taken over the channel SNRs which contribute to SNR_k^{Σ} via (37).

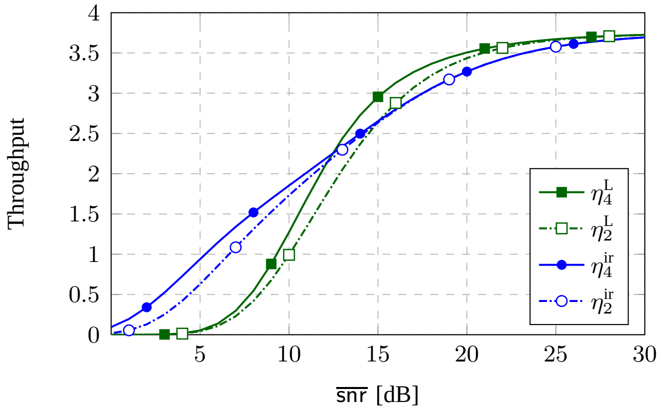

Fig. 4. Throughput of the proposed L-HARQ, η_K^{L}, is compared to the throughput of IR-HARQ, η_K^{ir}; $R = 3.75$, $\log_2(T_R) = 6$, and the synthetic PER curves defined in Sec. IV-A.

The optimal backtrack rates, $\rho_k(\mathsf{snr}_k, J_{k-1})$, obtained with the DP formulation are shown in Fig. 3. The rates ρ_k decrease with the observed snr_k because they are optimized to increase the chances of success in the backtrack decoding, and yet not to penalize the throughput. Thus, as snr_k increases, the number of bits needed to *guarantee* the backtrack decoding decreases. We also observe that the optimal policy varies little in terms J_{k-1}, which indicates the possibility of using a suboptimal policy independent of J_{k-1} as we will discuss in Sec. V-B.

The throughputs of L-HARQ and IR-HARQ are compared in Fig. 4. As already mentioned in Sec. II-B, we are mostly interested in the throughput close to R where the conventional IR-HARQ fails to provide gains even when increasing the number of retransmissions [2]. Indeed, this is where the improvement from L-HARQ materializes. For instance, around a throughput of $\eta = 3$, L-HARQ offers a gain of approximately $1\,$dB compared to IR-HARQ with $K = 2$, and up to 2.5dB with $K = 4$. On the other hand, L-HARQ is outperformed by IR-HARQ for small values of the throughput, where f_1 is high. This is not a serious drawback because, knowing the average SNR, we may switch to IR-HARQ if necessary or, if possible, use a different rate R. Performing a joint decoding, i.e., decoding $\mathsf{m}_{[2]}$ from $\boldsymbol{y}_2$ and $\boldsymbol{y}_1$ would also improve the performance at the cost of increased complexity, as we discussed in Sec. II-B.

Finally, Fig. 5 provides an insight into the additional feedback required to make L-HARQ operational. We note that with only two additional feedback bits, L-HARQ practically attains its maximum potential and ensures notable gains over the conventional IR-HARQ.

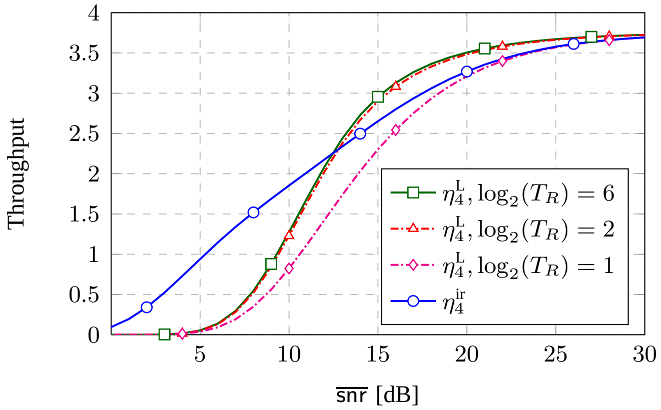

Fig. 5. Throughput of the L-HARQ, η_K^{L}, is compared to the throughput of IR-HARQ, η_K^{ir}, for $R = 3.75$ and different numbers of feedback bits $\log_2(T_R)$. The synthetic PER curves defined in Sec. IV-A are used.

B. Rate Adaptation with Turbo-Codes

In order to perform the optimization steps (32)–(34) for practical encoders/decoders, we only need the PER curves $\mathrm{PER}(\mathsf{snr}; R)$ and $\mathrm{PER}(\mathsf{snr}; R, \rho)$. These are obtained by simulating/measuring $\Pr\{\mathsf{ERR}_k\}$ and $\Pr\{\mathsf{ERR}_k \wedge \mathsf{ERR}_k^{\mathrm{b}}\}$, and the results obtained for different values of ρ_k are shown in Fig. 6; of course, if $\rho_k = 0$ we have $\Pr\{\mathsf{ERR}_k\} = \Pr\{\mathsf{ERR}_k \wedge \mathsf{ERR}_k^{\mathrm{b}}\}$.

We used here a turbo-code specified by 3rd generation partnership project (3GPP) in [26], comprising two constituent convolutional encoders with generating polynomials $[13/15]_8$ and the 3GPP pseudo-random interleaver defined in [26, Sec. 5.1.3.2.3]. The result of the encoding, after the interleaving of subblocks as prescribed by the 3GPP rate matching algorithm [26, Sec. 5.1.4.1] is denoted by $\mathsf{c} = [\mathsf{m}, \mathsf{m}^{\mathsf{p}}]$, where m^{p} and m are interleaved versions of the parity bits and systematic bits, respectively.

Since we use $R \in \{2.25, 3.75\}$ and the nominal coding rate of the 3GPP encoder is $r_{\mathrm{o}} = 1/3$, we need to puncture c to obtain the binary coding rate $r = R/m \in \{0.5625, 0.9375\}$, where $m = 4$ is the rate of the 16-quadrature amplitude modulation (QAM) modulation. We thus take $N_{\mathrm{c}}' = r_{\mathrm{o}} N_{\mathrm{c}}/r$ bits from c and map them with a Gray mapping [20, Sec. 2.5.2] onto $N_{\mathrm{s}} = 1024$ symbols $\boldsymbol{x}_k$ taken from a 16-QAM constellation, which are next transmitted over the channel (1). The receiver calculates the logarithmic likelihood ratios (LLRs) using exact expressions [20, Sec. 3.3] and feeds them to the Bahl–Cocke–Jelinek–Raviv (BCJR) decoder [27] implemented in the log-domain; the interested reader can refer to the library at [28].

As for the puncturing, we take N'_{c} bits starting with the offset of R_{m} $[\%]$ defining the percentage of the systematic bits being punctured. In this way, the codeword $\boldsymbol{x}_k$ in the kth round contains $100\% - R_{\mathsf{m}}$ of the bits in the message $\mathsf{m}_{[k]}$. The interesting question now is: which bits $\mathsf{m}'_{[k]}$ from the message $\mathsf{m}_{[k]}$ should be taken to construct the message $\mathsf{m}_{[k+1]} = [\mathsf{m}'_{[k]}, \mathsf{m}_{k+1}]$?

The interplay between the coding and the HARQ scheme becomes, indeed, interesting: for $R_{\mathsf{m}} > 0$, it is beneficial to construct $\mathsf{m}'_{[k]}$ using the *first* bits of $\mathsf{m}_{[k]}$ because some of these bits are punctured to construct $\boldsymbol{x}_k$ in round k; thus, knowing these bits (after a successful decoding in round $k+1$) improves the performance of the decoder in the backtrack phase. On the other hand, if we construct $\mathsf{m}'_{[k]}$ using the *last* bits of $\mathsf{m}_{[k]}$, their perfect knowledge (after a successful decoding of $\mathsf{m}_{[k+1]}$) will eliminate the channel-related LLRs during the backtrack decoding, removing thus some of the available information.

We show the PER curves of the turbo-decoder in Fig. 6 for $R_{\mathsf{m}} = 0\%$ and $R_{\mathsf{m}} = 6.25\%$, where the latter offset value is, in fact, recommended by the 3GPP. The important observation is that while the results of $\mathrm{PER}(\mathsf{snr}_k; R)$ (circles) deteriorate due to the puncturing of the systematic bits (solid lines, $R_{\mathsf{m}} = 6.25\%$), the results of the backtrack decoding are significantly improved in this case. There is thus a tradeoff between decreasing the decoding error probability and decreasing the probability of backtrack decoding error $\Pr\{\mathsf{ERR}^{\mathsf{b}}_k \wedge \mathsf{ERR}_k\}$. This tradeoff becomes even clearer as the nominal transmission rate R increases.

The above mentioned tradeoff becomes evident with the throughput results shown in Fig. 7 based on the same turbo-code PER curves shown in Fig. 6. For $R = 3.75$, and using $R_{\mathsf{m}} = 6\%$, the gain of L-HARQ over IR-HARQ is $\sim 0.5\,$dB for $K = 2$, and $\sim 2.5\,$dB for $K = 4$ (measured at $\eta = 3$). On the other hand, a similar gain is obtained for $K = 2$ with $R_{\mathsf{m}} = 0\%$, but no further improvement is observed when the number of transmissions is increased to $K = 4$. However, the effect of changing R_{m} on the results of L-HARQ is less notable when $R = 2.25$ as can be seen in Fig. 7(b). This is not too surprising, since the difference between $\Pr\{\mathsf{ERR}_k \wedge \mathsf{ERR}^{\mathsf{b}}_k\}$ curves of $R_{\mathsf{m}} = 0\%$ and $R_{\mathsf{m}} = 6.25\%$ is less important when $R = 2.25$; see Fig. 6b.

V. Sub-optimal rate adaptation policies

We will now discuss adaptation strategies aiming i) to streamline the way the backtrack errors are handled, and ii) to simplify the rates adaptation.

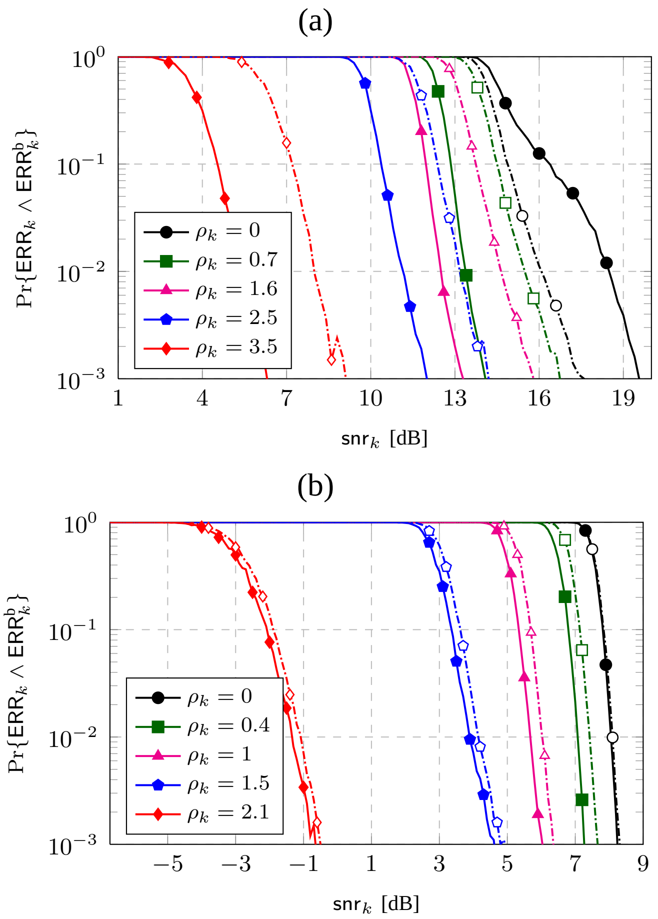

Fig. 6. $\Pr\{\mathsf{ERR}_k \wedge \mathsf{ERR}^{\mathrm{b}}_k\}$ as a function of the instantaneous snr_k for different values of ρ_k when a turbo-code and a 16-QAM modulation are used with (a) $R = 3.75$ and (b) $R = 2.25$. Dashed curves correspond to the case where systematic bits are not punctured, i.e., $R_{\mathsf{m}} = 0\%$, while solid lines correspond to the results obtained by puncturing systematic bits with $R_{\mathsf{m}} = 6.25\%$.

A. *All-or-none decoding*

In the example of two rounds, presented in Sec. III-A, if the message $\mathsf{m}_{[2]}$ is decoded successfully and the backtrack decoding of m_1 fails, L-HARQ does not discard the correctly received $N_{\mathrm{s}}\,\rho_1$ bits of m_1 (meaning that only a part of m_1 is received correctly). This complicates the buffer management, and may not be suitable for some applications in which only the packet m_1 is critical and the packets $\mathsf{m}_2, \ldots, \mathsf{m}_k$ are piggybacked on the ongoing HARQ process to not waste the ressources.

We thus want to evaluate a different strategy, where a non-zero reward is collected only if both $\mathsf{m}_{[2]}$ and m_1 are decoded successfully. In the resulting all-or-none L-HARQ (AoN-HARQ)

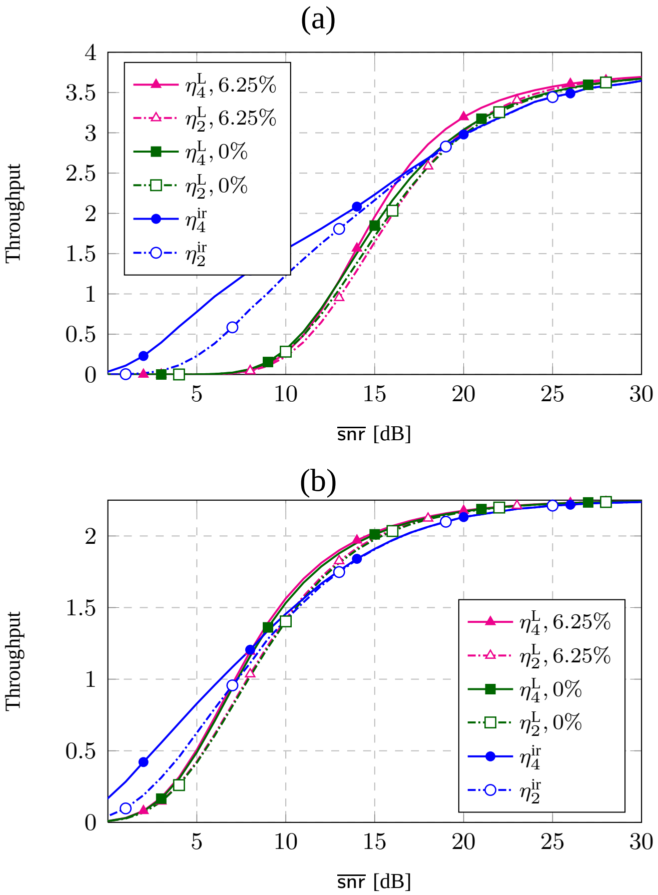

Fig. 7. The throughput of L-HARQ and IR-HARQ obtained for turbo-coded 16QAM transmissions with the puncturing defined by $R_{\mathsf{m}} = 0\%$ and $R_{\mathsf{m}} = 6.25\%$for (a) $R = 3.75$, and (b) $R = 2.25$; $\log_2(T_R) = 4$.

the average reward (22) is modified as

$$\begin{aligned}\mathbb{E}[\mathsf{R}] &= \mathbb{E}\Big[R\,\mathbb{I}\big[\overline{\mathsf{ERR}_1}\big] + (2R-\rho_1)\;\mathbb{I}\big[\mathsf{ERR}_1 \wedge \overline{\mathsf{ERR}_2} \wedge \overline{\mathsf{ERR}^{\mathsf{b}}_1}\big]\Big]\\ &= \mathbb{E}\Big[R(1-\Pr\{\mathsf{ERR}_1\}) + (2R-\rho_1)(1-\Pr\{\mathsf{ERR}_2\})\\ &\qquad \Pr\{\mathsf{ERR}_1\}\big(1-\Pr\{\mathsf{ERR}^{\mathsf{b}}_1|\mathsf{ERR}_1\}\big)\Big].\end{aligned} \tag{46}$$

$$
\begin{aligned}
\overline{\mathsf{R}}^{\text{AoN}} &= R \cdot \mathbb{E}_{\mathsf{SNR}_1}\big[V_1(\mathsf{SNR}_1, R)\big], && (42)\\
V_1(\mathsf{snr}_1, J_0) &= \max_{\rho_1}\Big\{\text{PER}^{\mathsf{c}}(\mathsf{snr}_1; R) + \frac{J_0 + R - \rho_1}{J_0}\text{PER}(\mathsf{snr}_1; R)\text{PER}^{\mathsf{c}}(\mathsf{snr}_1; R, \rho_1)\\
&\qquad \times \mathbb{E}_{\mathsf{SNR}_2}\big[V_2(\mathsf{SNR}_2, J_1)\big]\Big\}, && (43)\\
&\ \vdots\\
V_{K-2}(\mathsf{snr}_{K-2}, J_{K-3}) &= \max_{\rho_{K-2}}\Big\{\text{PER}^{\mathsf{c}}(\mathsf{snr}_{K-2}; R) + \frac{J_{K-3} + R - \rho_{K-2}}{J_{K-3}}\text{PER}(\mathsf{snr}_{K-2}; R)\text{PER}^{\mathsf{c}}(\mathsf{snr}_{K-2}; R, \rho_{K-2})\\
&\qquad \times \mathbb{E}_{\mathsf{SNR}_{K-1}}\big[V_{K-1}(\mathsf{SNR}_{K-1}, J_{K-2})\big]\Big\}, && (44)\\
V_{K-1}(\mathsf{snr}_{K-1}, J_{K-2}) &= \max_{\rho_{K-1}}\Big\{\text{PER}^{\mathsf{c}}(\mathsf{snr}_{K-1}; R) + \frac{J_{K-2} + R - \rho_{K-1}}{J_{K-2}}\text{PER}(\mathsf{snr}_{K-1}; R)\text{PER}^{\mathsf{c}}(\mathsf{snr}_{K-1}; R, \rho_{K-1})\\
&\qquad \times \mathbb{E}_{\mathsf{SNR}_K}\big[\text{PER}^{\mathsf{c}}(\mathsf{SNR}_K; R)\big]\Big\}. && (45)
\end{aligned}
$$

In a case of arbitrary K the expected reward of AoN-HARQ (46) generalizes as follows:

$$
\begin{aligned}
\mathbb{E}[\mathsf{R}] &= \mathbb{E}\Big[\sum_{k=1}^{K}(kR - \sum_{l=1}^{k-1}\rho_l) \cdot \big(1 - \Pr\{\mathsf{ERR}_k\}\big)\\
&\qquad \times \prod_{z=1}^{k-1}\Pr\{\mathsf{ERR}_z\}\big(1 - \Pr\{\mathsf{ERR}_z^{\mathsf{b}} | \mathsf{ERR}_z\}\big)\Big], && (47)\\
&= R\ \mathbb{E}_{\mathsf{SNR}_1}\Bigg[(1 - \Pr\{\mathsf{ERR}_1\}) + \frac{(2R - \rho_1)}{R}\Pr\{\mathsf{ERR}_1\}\\
&\qquad \big(1 - \Pr\{\mathsf{ERR}_1^{\mathsf{b}} | \mathsf{ERR}_1\}\big)\mathbb{E}_{\mathsf{SNR}_2}\Big[\big(1 - \Pr\{\mathsf{ERR}_2\}\big) +\\
&\qquad \frac{(3R - \rho_1 - \rho_2)}{(2R - \rho_1)}\Pr\{\mathsf{ERR}_2\}\big(1 - \Pr\{\mathsf{ERR}_2^{\mathsf{b}} | \mathsf{ERR}_2\}\big)\\
&\qquad \mathbb{E}_{\mathsf{SNR}_3}\Big[\big(1 - \Pr\{\mathsf{ERR}_3\}\big) + \ldots\Big]\Big]\Bigg], && (48)
\end{aligned}
$$

while the expected number of rounds is the same as in (27). Thus, the optimal throughput of AoN-HARQ, denoted as η_K^{AoN}, is given by

$$
\eta_K^{\text{AoN}} = \frac{(1 - f_1^K) \cdot \overline{\mathsf{R}}^{\text{AoN}}}{1 - f_1}, \qquad (49)
$$

where $\overline{\mathsf{R}}^{\text{AoN}}$ denotes the optimum expected reward (47) with respect to $\{\rho_k\}_{k=1}^{K-1}$. Again, profiting

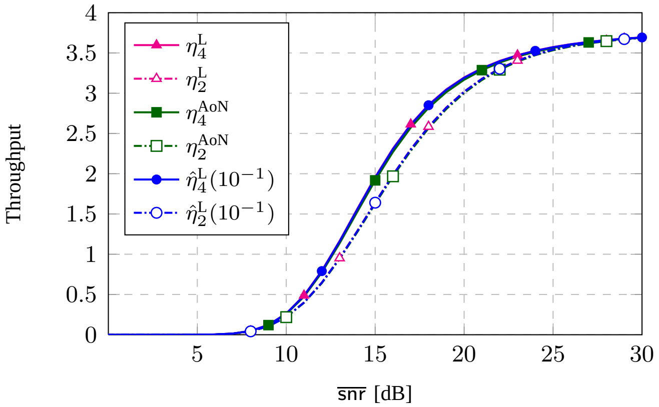

Fig. 8. The throughputs of AoN-HARQ and the heuristic policy (51) when $\epsilon = 0.1$ are compared with L-HARQ results obtained for turbo-coded 16QAM transmissions with the puncturing defined by $R_{\mathsf{m}} = 6.25\%$ for $R = 3.75$; $\log_2(T_R) = 4$.

from the nested structure of (48), the $\overline{\mathsf{R}}^{\mathrm{AoN}}$ can be found by solving the recursive equations (42)–(45), where $J_k \in \big(R, (k+1)\cdot R\big)$, and it is related to J_{k-1} and ρ_k through

$$J_k = J_{k-1} + R - \rho_k, \tag{50}$$

where, by definition, $J_0 = R$.

The results of the proposed AoN-HARQ are compared with L-HARQ in Fig. 8. We can clearly see that imposing the constraint that all backtrack decoding actions are successful does not penalize the final throughput of AoN-HARQ, which is practically equal to the optimal throughput of L-HARQ. We thus conclude that the optimal backtrack rates of L-HARQ are such to guarantee a high probability of successful backtrack decoding. This observation will be exploited in the following to simplify the rate adaptation policy.

B. Fixed-outage policy

The rate adaptation policies $\rho_k(\mathsf{snr}_k, J_{k-1})$ determined by solving (32)–(34) or (43)–(45) are sufficient to optimize the throughput but they have two drawbacks, namely

1) The rates are three-dimensional functions of snr_k, J_{k-1} and the transmission round k, see Fig. 3; this is inconvenient from the point of view of storage requirement.
2) The rate depend on the distribution of SNR, which not only adds to the storage and optimization complexity, but makes the solution potentially sensitive to the changes in the channel model.

To address the above issues, we propose a simple one-dimensional adaptation policy, independent of J_{k-1}, k, and $p_{\mathsf{SNR}}(\mathsf{snr})$, which is partially inspired by the form of the optimal policy in Fig. 3 that varies little in terms of J_{k-1} and k. Moreover, motivated by the results of AoN-HARQ, which provide results with very reliable backtrack decoding and this, without penalizing the throughput, we propose the rate adaptation policy, which will guarantee successful instantaneous backtrack decoding. Thus we take into account solely the outdated channel SNR

$$\rho(\mathsf{snr}_k) = \underset{\rho\in\mathcal{A}}{\operatorname{argmin}} \big\{ \rho \ \mid \mathrm{PER}(\mathsf{snr}_k; R, \rho) \leq \epsilon \big\}, \tag{51}$$

where $\epsilon \in \mathbb{R}_+$ is a design parameter.

The throughput obtained with the policy $\rho(\mathsf{snr}_k)$, we denote by $\hat{\eta}_K^{\mathrm{L}}(\epsilon)$, can be evaluated via (25) to determine the optimal values of ϵ

$$\hat{\epsilon} = \underset{\epsilon}{\operatorname{argmax}} \ \hat{\eta}_K^{\mathrm{L}}(\epsilon) \tag{52}$$

which we show in Fig. 9. Alternatively, we might use simulations to evaluate the throughput with different values of ϵ; the direct advantage of such an approach is that it would free us from the channel-model dependence.

Here, we observe that while $\hat{\epsilon}$ is a function of the average SNR, it varies little in the region of high $\overline{\mathsf{snr}}$. And since this region of operation is of main interest, we further fix $\epsilon = 10^{-1}$ eliminating the dependence of the policy on the channel statistics.[2] The throughput $\hat{\eta}_K^{\mathrm{L}}(10^{-1})$ is shown in Fig. 8, where it is clear that the penalty incurred with respect to the optimal solution is negligible.

This is quite a remarquable result which indicates that the throughput obtained with a very simple adaptation strategy (51) that is agnostic to the channel statistics as well as to the past and the future of the HARQ process, is very close to the optimal solution.

VI. Conclusions

In this work, we proposed an HARQ transmission scheme and showed how its throughput can be optimized using PER curves of the practical decoder. Compared to the conventional IR-HARQ protocol, the proposed solution yields notable gains in the high throughput regime. In

[2] This value is arbitrary, but we wanted a "round" number close to what the results indicated.

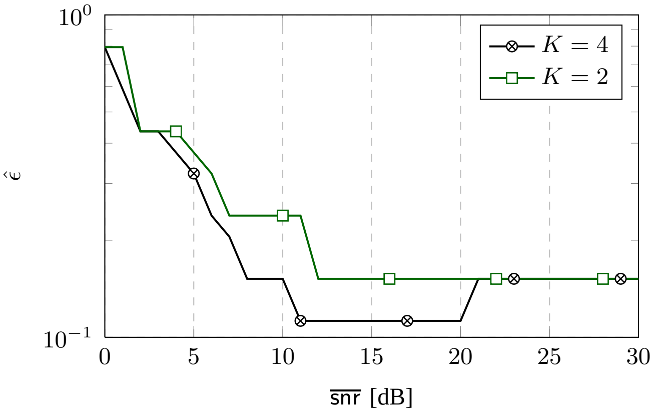

Fig. 9. The optimal $\hat{\epsilon}$ which solves (52) for turbo-coded 16QAM transmissions with the puncturing defined by $R_{\mathsf{m}} = 6.25\%$ for $R = 3.75$; $\log_2(T_R) = 4$.

wireless systems, these gains may translate into energy savings, reduced intercell interference, or coverage extension.

To illustrate our findings, we used turbo-codes to demonstrate the possibility of boosting HARQ throughput with off-the-shelf codes, and we discussed the importance of a code design (here–the puncturing) to see the gains materialize. We only need the simulated/measured PER curves $\mathrm{PER}(\mathsf{snr}; R)$ and $\mathrm{PER}(\mathsf{snr}; R, \rho)$ to perform the rate adaptation. Thus, our approach is well suited to the case of finite block-length, a promising feature for 5G systems which was studied recently in a similar context in [12], [13].

Furthermore, we developed suboptimal but very simple rate adaptation strategies, and showed that the inflicted performance loss is negligible compared to the optimal schemes.

REFERENCES

[1] G. Caire and D. Tuninetti, “The throughput of hybrid-ARQ protocols for the Gaussian collision channel,” *IEEE Trans. Inf. Theory*, vol. 47, no. 5, pp. 1971–1988, Jul. 2001.

[2] P. Larsson, L. K. Rasmussen, and M. Skoglund, “Throughput analysis of ARQ schemes in Gaussian block fading channels,” *IEEE Trans. Commun.*, vol. 62, no. 7, pp. 2569–2588, Jul. 2014.

[3] M. Jabi, A. El Hamss, L. Szczecinski, and P. Piantanida, “Multi-packet hybrid ARQ: Closing gap to the ergodic capacity,” *IEEE Trans. Commun.*, vol. 63, no. 12, pp. 5191–5205, Dec. 2015.

[4] J.-F. Cheng, Y.-P. Wang, and S. Parkvall, “Adaptive incremental redundancy,” in *IEEE Veh. Tech. Conf. (VTC Fall)*, Orlando, Florida, USA, Oct. 2003, pp. 737–741.

[5] E. Uhlemann, L. K. Rasmussen, A. Grant, and P.-A. Wiberg, “Optimal incremental-redundancy strategy for type-II hybrid ARQ,” in *IEEE Intern. Symp. Inf. Theory (ISIT)*, 2003, p. 448.

[6] E. Visotsky, V. Tripathi, and M. Honig, “Optimum ARQ design: a dynamic programming approach,” in *IEEE Intern. Symp. Inf. Theory (ISIT)*, Jun. 2003, p. 451.

[7] E. Visotsky, Y. Sun, V. Tripathi, M. Honig, and R. Peterson, “Reliability-based incremental redundancy with convolutional codes,” *IEEE Trans. Commun.*, vol. 53, no. 6, pp. 987–997, Jun. 2005.

[8] S. Pfletschinger and M. Navarro, “Adaptive HARQ for imperfect channel knowledge,” in *2010 International ITG Conference on Source and Channel Coding (SCC)*, Jan. 2010, pp. 1–6.

[9] L. Szczecinski, S. R. Khosravirad, P. Duhamel, and M. Rahman, “Rate allocation and adaptation for incremental redundancy truncated HARQ,” *IEEE Trans. Commun.*, vol. 61, no. 6, pp. 2580–2590, June 2013.

[10] P. Larsson, B. Smida, T. Koike-Akino, and V. Tarokh, “Analysis of network coded HARQ for multiple unicast flows,” *IEEE Trans. Commun.*, vol. 61, no. 2, pp. 722–732, Feb. 2013.

[11] P. Popovski, “Delayed channel state information: Incremental redundancy with backtrack retransmission,” in *IEEE Inter. Conf. Comm. (ICC)*, June 2014, pp. 2045–2051.

[12] K. Trillingsgaard and P. Popovski, “Block-fading channels with delayed CSIT at finite blocklength,” in *IEEE Intern. Symp. Inf. Theory (ISIT)*, June 2014, pp. 2062–2066.

[13] K. D. Nguyen, R. Timo, and L. K. Rasmussen, “Causal-CSIT rate adaptation for block-fading channels,” in *IEEE Intern. Symp. Inf. Theory (ISIT)*, Jun. 2015, pp. 351–355.

[14] A. Benyouss, M. Jabi, L. T. Treust, and L. Szczecinski, “Joint coding/decoding for multi-message HARQ,” in *IEEE Wireless Communications and Networking Conference (WCNC’16), 3-6 April, Doha, Qatar*, 2016.

[15] M. Jabi, M. Benjillali, L. Szczecinski, and F. Labeau, “Energy efficiency of adaptive HARQ,” *IEEE Trans. Commun.*, vol. 64, no. 2, pp. 818–831, Feb. 2016.

[16] J. Gu, Y. Zhang, and D. Yang, “Modeling conditional FER for hybrid ARQ,” *IEEE Commun. Lett.*, vol. 10, no. 5, pp. 384–386, May 2006.

[17] H. Long, W. Wang, K. Zheng, and F. Wang, “Performance analysis on conditional error ratio in HARQ transmission,” *3rd IEEE International Symposium on Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications (MAPE), 2009*, vol. 297-302, Oct. 2009.

[18] C. Hausl and A. Chindapol, “Hybrid ARQ with cross-packet channel coding,” *IEEE Commun. Lett.*, vol. 11, no. 5, pp. 434–436, May 2007.

[19] D. Duyck, D. Capirone, C. Hausl, and M. Moeneclaey, “Design of diversity-achieving LDPC codes for H-ARQ with cross-packet channel coding,” in *IEEE Inter. Symp. Pers. Indoor and Mob. Comm. (PIMRC)*, Sept. 2010, pp. 263–268.

[20] L. Szczecinski and A. Alvarado, *Bit-Interlaved Coded Modulation : Fundamentals, Analysis and Design*. Wiley, 2015.

[21] S. Pfletschinger, D. Declercq, and M. Navarro, “Adaptive HARQ with non-binary repetition coding,” *IEEE Trans. Wireless Commun.*, vol. 13, no. 8, pp. 4193–4204, Aug. 2014.

[22] Q. Liu, S. Zhou, and G. B. Giannakis, “Cross-layer combining of adaptive modulation and coding with truncated ARQ over wireless links,” *IEEE Trans. Wireless Commun.*, vol. 3, no. 5, pp. 1746–1755, Sep. 2004.

[23] R. Sassioui, L. Szczecinski, L. B. Le, and M. Benjillali, “AMC and HARQ: Effective capacity analysis,” in *IEEE Wireless Communications and Networking Conference (WCNC’16), 3-6 April, Doha, Qatar*, 2016.

[24] M. Pauli, U. Wachsmann, and S. Tsai, “Quality determination for a wireless communications link,” Jun. 2007. [Online]. Available: https://www.google.com/patents/US7231183

[25] L. Wan, S. Tsai, and M. Almgren, “A fading-insensitive performance metric for a unified link quality model,” in *IEEE Wireless Communications and Networking Conference WCNC’06*, vol. 4, 2006, pp. 2110–2114.

[26] 3GPP, “Evolved universal terrestrial radio access (E-UTRA); multiplexing and channel coding,” 3GPP, Tech. Rep. V12.5.0, 2015-07.

[27] L. J. Bahl, J. Cocke, F. Jelinek, and J. Raviv, “Optimum decoding of linear codes for minimizing symbol error rate,” *IEEE Trans. Inf. Theory*, vol. 20, no. 2, pp. 284–287, Mar. 1974.

[28] E. Pierre-Doray and L. Szczecinski. (2015) “FeCl channel coding library”. [Online]. Available: https://github.com/eti-p-doray/FeCl/wiki