Multi-Layer HARQ with Delayed Feedback

Alaa Khreis, Francesca Bassi, Philippe Ciblat, Pierre Duhamel

Multi-Layer HARQ with Delayed Feedback

Alaa Khreis *Member IEEE*, Francesca Bassi *Member IEEE*, Philippe Ciblat *Senior Member IEEE*, and Pierre Duhamel *Fellow IEEE*

Abstract

In order to improve the transmission reliability in current wireless communication systems, the Hybrid Automatic ReQuest (HARQ) protocol is employed to manage the unknown time-varying channel. The acknowledgments are fed back with delay on the return link. To fill up the idle time between a transmission and its acknowledgment, parallel HARQ streams associated with different messages are carried out. In this paper we improve on parallel HARQ by proposing a multi-layer HARQ protocol (also called superposition coding or multi-packet HARQ), where a single transmission may carry information on multiple messages. The multi-layer HARQ protocol works in presence of delay on the return link as parallel HARQ does, and does not require additional feedback such as the channel state information. It aims at improving the accuracy as well as the user's delay distribution, thus achieving throughput increase. Assuming capacity-achieving codes, we show that the proposed protocol outperforms parallel HARQ in throughput, message error rate, and delay distribution. Using practical codes and decoding algorithms the gains are as well significant, at the expense of the receiver's complexity.

I. Introduction

Hybrid Automatic Repeat reQuest (HARQ) has become an important research field in the wireless digital communications area during the last decade [2]–[9] since it enables to improve the robustness of communication over fading channels. In multi-stream communication (where different message streams may belong to the same user or not) orthogonal multiple access techniques such as Orthogonal Frequency Division Multiple Access (OFDMA) may be used, as

A. Khreis was with Télécom ParisTech, Paris, France. He is now with Huawei Technologies, France. F. Bassi was with ESME-Sudria and UniversitÃl' Paris-Saclay, CNRS/L2S, Gif-sur-Yvette, France. She is now with IRT SystemX, Palaiseau, France. P. Ciblat is with Télécom ParisTech, Institut Polytechnique de Paris, Palaiseau, France. P. Duhamel is with Université Paris-Saclay, CNRS/L2S, Gif-sur-Yvette, France. Contact: alaa_khreis@ieee.org, francesca.bassi@irt-systemx.fr, philippe.ciblat@telecom-paristech.fr, pierre.duhamel@l2s.centralesupelec.fr. Part of this work has been published in IEEE PIMRC 2018 conference [1]. This work was supported by the Labex Digicosme PhD scholarship from Université Paris-Saclay under the grant called "Coccinelle"

in 4G and 5G, to share the radio resources. The solution to manage HARQ over the different streams is generally to mimic Time Division Multiple Access (TDMA). For instance, if the streams belong to the same user, parallel Stop-And-Wait (SAW) HARQ [10], [11] (shortened by parallel HARQ in the rest of the paper) is carried out, which processes in turn multiple independent streams (up to 8 parallel streams in Long Term Evolution (LTE)). Some efforts have been made for jointly designing HARQ and the multiple access technique. Assuming time-slotted communications, the main idea lies in sharing the time slot in a smart way for example between a stream and a retransmission belonging to another stream, instead of just applying TDMA slot-by-slot. Different strategies are listed below.

- *Time sharing of the time-slot:* each message is separately encoded and modulated using a specific rate in order to share a time-slot in the time domain; This policy enables to be less rigid than a slot-by-slot allocation since the size of the retransmission packets are adapted and tuned according to feedback information [12].
- *Joint encoding:* different messages are jointly encoded, then modulated into a single packet sent in a time-slot. In this case, the messages are interleaved into each time-slot. Once again, this approach is less rigid than a slot-by-slot allocation [13]–[15].
- *Superposition coding:* each message is separately encoded and modulated using a fixed rate. The packets associated with different messages are then superposed in one time-slot using a specific power allocation. This approach relies on Non Orthogonal Multiple Access (NOMA) principle which has proven its superiority to the TDMA. This leads to the so-called *multi-layer* based HARQ [16]–[21].

A combination of the previously-mentioned strategies can be proposed, as done in [21].

In this paper, we build on parallel HARQ in a single-user context. We propose a multi-packet HARQ protocol which keeps parallel HARQ as a baseline, and on top implements the superposition coding approach by adding layers that carry retransmissions. This aims at improving the accuracy measured by the Message Error Rate (MER). In practical systems the HARQ feedback messages (ACKnowledgment (ACK)/Negative ACKnowledgment (NACK)) arrive at the transmitter with a delay which may be important because of the propagation times, the decoding processing time, and the reverse link scheduling (8 time-slots in LTE [11]). The multi-layer HARQ protocol allows to anticipate a necessary retransmission before receiving the NACK feedback, and this aims at improving the distribution of the delay at the receiver. As

a consequence of improved accuracy and delay, the throughput, here defined as the average number of correctly received information bits per channel use, is expected to improve as well. Our protocol requires only ACK/NACK feedback, and no additional Channel State Information (CSI) at the transmitter (which would be otherwise necessary with the time sharing approach). Moreover, in contrast to the solutions based on the superposition coding approach in the literature (see Section I-A), which assume instantaneous feedback, our protocol works also for delayed feedback, as the parallel HARQ does.

Although arbitrarily many layers could be considered for the multi-layer HARQ protocol, we restrict our attention to the case of two layers. A practical receiver needs in fact to manage the interference between the superposed layers in order to decode the streams, with complexity increasing with the number of layers. Restricting to two layers allows us to consider feasible practical decoders, and to characterize analytically the achievable performance of the system.

The rest of the paper is organized as follows. In Section I-A we present a non-exhaustive overview on multi-packet based HARQ protocols. In Section II we introduce the system model and in Section III we describe and justify the proposed HARQ protocol based on superposition coding. The achievable performance of two receiver's decoders, one based on interference cancellation and one considering interference as noise, is characterized in Section IV. Numerical evaluations based on either capacity-achieving or practical codes of the proposed protocol are conducted in Section V. The multi-layer HARQ improves over parallel HARQ in terms of accuracy, delay and throughput. The numerical results explore the effects of the power assignment on the layers, and assess the performance loss of a receiver considering interference as noise in comparison with a receiver implementing interference cancellation. The performance gain of the proposed protocol over parallel HARQ is maintained in the case of practical decoders as well. Finally, concluding remarks are drawn in Section VI.

A. Related works

Before going further, we present a non-exhaustive overview of multi-packet based HARQ protocols. In [16], a multi-layer based HARQ with superposition coding is proposed to improve the throughput. Assumption is that the feedback is without delay. Assuming capacity-achieving codes and a decoder treating the layer-interference as additional noise, a slight gain in throughput for a given rate is observed while the rate adaptation coupled with a multi-layer transmission enables a huge gain. In [17], a practical decoder based on Successive Interference Cancellation

(SIC) with real codes is implemented for a similar setup to [16]. In [19], simulations close to those of [17] have been carried out when only one retransmission and only one additional layer are allowed and Quadrature Phase-shift Keying (QPSK) modulation is considered.

In [13], multi-layer based HARQ is done with the joint encoding for the retransmission phase. The first transmission is done as usual by stacking the different streams. An extension is proposed in [14] where perfect CSI is available to the transmitter side.

In [12], they consider multi-layer based HARQ with a time sharing approach for the retransmission phase. The rate adaptation per stream is done for each retransmission (in order to fit the slot portion devoted to the dedicated stream) and relies on perfect past CSI, i.e., the accumulated mutual information (of each HARQ stream). Once again, the feedback is without delay. Extensions are proposed in [20], [21] where the transmitter may choose between the approaches (time-sharing, superposition coding and no additional layer) according to the instantaneous ACK/NACK and accumulated mutual information. The problem is solved via a Markov Decision Process (MDP).

In [15], a multi-layer based HARQ with joint encoding is optimized based on the knowledge of the ACK/NACK without delay and additional CSI of the last transmissions. Note that similar ideas have been applied in other contexts such as Transport Control Protocol (TCP). For instance, in [18], a multi-layer HARQ with superposition coding is done when the number of pre-assigned slots is equal to the maximum number of allowed transmissions. If the streams are decoded with few transmissions, some slots are empty, hence wasted. With multi-layer technique, more streams can be sent with fewer empty slots.

II. System model

We consider point-to-point transmission, slotted, with a time-slot corresponding to N channel uses. During time-slot t the transmitter sends the N symbols $\mathbf{x}_t$. The vector $\mathbf{x}_t$ may represent a single packet or a superposition of two packets, as will be explained in Section III-A. It is sent over a Rayleigh flat fading channel with coherence time equal to the time-slot duration. Let $h(t)$ denote the channel realization at time-slot t. The received signal at time-slot t is:

$$\mathbf{y}_t = h(t)\mathbf{x}_t + \mathbf{w}_t, \tag{1}$$

where $\mathbf{w}_t$ is an additive white Gaussian noise vector, with zero-mean and variance per component equal to N_0. The channel-to-noise gain is denoted by $g(t)$ where $g(t) = \frac{|h(t)|^2}{N_0}$. The channel gain

$h(t)$ is assumed to be known by the receiver. The average channel gain $\sigma_h^2 := \mathbb{E}[|h(t)|^2]$ is assumed to be known by the transmitter since it can be fed back only when modified which is not often as the coherence time of the average gain is much larger than the time-slot duration. The value of σ_h^2 is actually related to the pathloss.

To enable the use of Incremental Redundancy (IR)-HARQ, each message $\mathbf{m}_k$, containing RN information bits, is encoded via a mother code of rate R_0, and then punctured into C codeword chunks of index ℓ, $\ell \in \{1, 2, \ldots, C\}$. The ℓ-th codeword chunk is modulated into a packet of length N, denoted by $\mathbf{p}_k(\ell)$. Packets relative to the same message may occupy C time-slots at most, using the truncated HARQ mechanism.

The feedback is error-free and only composed of ACK or NACK of the considered messages. We assume a feedback delay of T time-slots, which means that the feedback of the transmission at time-slot t gets to the transmitter just before the beginning of time-slot $t+T$. The case $T = 1$ thus corresponds to a no-delay feedback. For the clarity of the presentation, we assume that the receiver knows at the end of time-slot t if the considered messages at time-slot t are successfully decoded or not, i.e. the decoding time has been assumed to be null. Extension and discussion to non-vanishing decoding time is done in Section V-C. Moreover, a message is said in *timeout* if it has not been ACKed after CT time-slots starting from its first transmission, corresponding to the timeout in parallel HARQ.

III. Proposed Protocol

We propose a protocol that enables the transmitter to anticipate the HARQ feedback by sending, in advance to its correct reception, packets related to unacknowledged messages using superposition coding. In contrast to previous works, where HARQ is considered without a delayed feedback of multiple time-slots, the proposed protocol is designed to counteract this feedback delay as well as to improve the throughput.

A. Transmitter strategy

In the proposed protocol, at each time-slot the transmitter selects a packet $\mathbf{p}_k(\ell)$, based on the ACK/NACK feedbacks, as in parallel HARQ. The transmitter may superpose to $\mathbf{p}_k(\ell)$ a second packet $\mathbf{p}_{k'}(\ell')$, with $k' \neq k$, even before receiving any feedback corresponding to previous transmissions of message $\mathbf{m}_{k'}$. The idea is to send a redundant packet without waiting for the feedback to arrive at the transmitter side, which enables the receiver to possibly decode $\mathbf{m}_{k'}$

without waiting for the next HARQ round. Accordingly, the transmission occurs in two layers where:

- Layer 1 acts as the parallel HARQ protocol;
- Layer 2 corresponds to the transmission of additional redundant packets.

Note that layer 2 does not transmit a packet associated with a new message but rather a redundant packet associated with an already-sent message. The idea is to improve the standard parallel HARQ by ensuring a smaller delay, a better MER, and finally a better throughput. Sending new messages on layer 2 while the other messages in layer 1 are not acknowledged would degrade the delay and the MER since messages on layer 1 would not be better protected.

In order to keep the same energy at each time-slot, the superposed packet, belonging to the second layer, uses the portion $(1-\alpha)$ of the predefined energy per time-slot, while the packet sent by the first layer uses the portion α of the energy, with $\alpha \in [0,1]$. The influence of α will be investigated in Section V. The transmit vector $\mathbf{x}_t$ is given by:

$$\mathbf{x}_t = \begin{cases} \mathbf{p}_k(\ell), & \text{if no superposition,} \\ \sqrt{\alpha}\mathbf{p}_k(\ell) + \sqrt{1-\alpha}\mathbf{p}_{k'}(\ell'), & \text{if superposition.} \end{cases} \tag{2}$$

Note that the case of $\alpha = 1$ corresponds to the parallel HARQ. Therefore, tuning α can only result in improvements over parallel HARQ Note also that superposing many HARQ packets in the same time-slot is limited by interference between the superposed packets. Although more layers could be superposed in theory, we superpose two layers at most to avoid increasing the decoder complexity in practical systems, since the decoder has to manage the interference between the superposed layers.

At the beginning of time-slot t the transmitter knows the ACK/NACK related to the messages sent up to time-slot $t-T$ (because of the feedback delay). According to this knowledge, the transmitter selects the packets to be included in $\mathbf{x}_t$. As anticipated, the first layer acts as parallel HARQ. Therefore, if packet $\mathbf{p}_k(\ell)$ was sent in the first layer at time-slot $t-T$, the reception of a NACK relative to message $\mathbf{m}_k$ just before time-slot t triggers the transmission of another redundancy packet $\mathbf{p}_k(\ell+1)$, as long as $\ell < C$. Otherwise, the reception of an ACK of $\mathbf{m}_k$ triggers the transmission of a packet $\mathbf{p}_{k''}(1)$ associated with a new message $\mathbf{m}_{k''}$ (never transmitted before). The selection of the superposed packet in the second layer is done according to the following principles: i) superposing packets related to the most recent messages of the first layer to reduce the delay, ii) superposing unsent redundant packets to reduce the message error by

using transmit diversity. Based on these principles, we describe the selection strategy by the following rules (ordered by priority), which determine the choice of the superposed packet in the second layer:

1) A packet $\mathbf{p}_{k'}(\ell')$ cannot be superposed if message $\mathbf{m}_{k'}$ is in timeout or previously ACKed.
2) As long as there are unacknowledged messages with unsent packets, the superposed packet is the unsent packet of the lowest index ℓ' of the most recent message $\mathbf{m}_{k'}$, with $k' \neq k$ (different messages in the two layers).
3) If the transmitter already sent all the packets of all the unacknowledged messages that are not in timeout, the superposed packet is the packet with the lowest index ℓ' that was not previously sent in the second layer. (Notice that this packet has been already sent once, in the first layer).
4) No packet is superposed to a packet of the first layer that has $\ell = C$.

The first rule prevents larger delays than those provided by conventional parallel HARQ. The second rule reduces the delay furthermore by sending redundant packets related to unacknowledged messages in advance to the receiver's feedback, and provides a diversity gain. Likewise, the third rule provides more diversity gains by superposing packets related to different messages at each time-slot, in addition to sending different Redundancy Version (RV)s corresponding to each message in the second layer. Moreover, the fourth rule reduces the probability to drop messages by forbidding interference during the last retransmission. Notably, this last rule is necessary in order to simplify the decoding at the receiver side by limiting the number of messages (to be decoded) in the buffer. In other words, one can check that, at each time-slot, at most T messages are not previously ACKed nor in timeout, which means also that the feedback at each time-slot contains at most T feedback bits (ACKs/NACKs).

Note that our protocol still works if Chase-Combining (CC) HARQ (also called Repetition Time Diversity (RTD)) is carried out. As the redundant packet is then identical to the initial one, the proposed protocol simplifies as follows: the rule 2) now just corresponds to send the packet of the most recent message; the rule 3) vanishes since it does not make sense anymore.

In Section III-A1 we provide an example of the protocol, with $C = 3$ and $T = 3$. We remind that we consider instantaneous decoding at the end of the time-slot t, although the feedback will be available at the transmitter side after T time-slots.

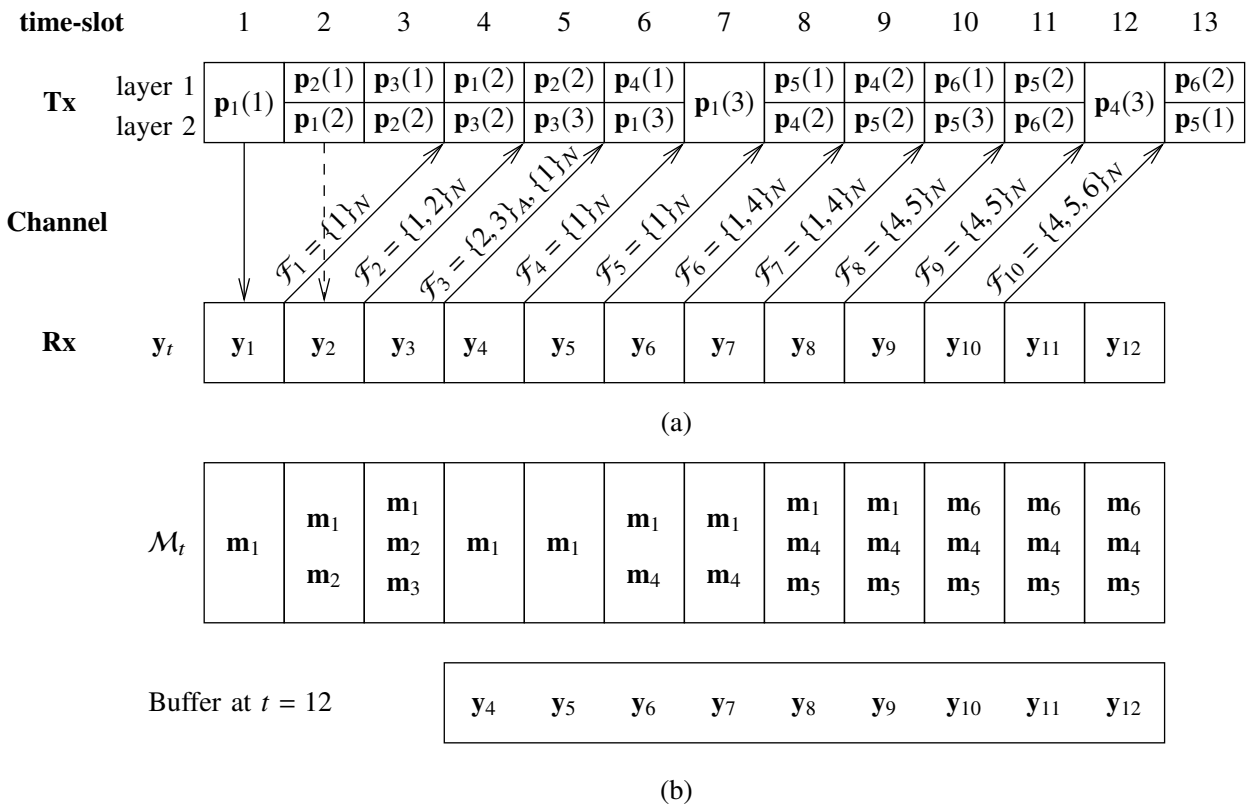

Figure 1: A realization of the proposed protocol (a) and the corresponding receiver buffer (b).

1) Example: This section presents a realization of the proposed protocol, illustrated in Fig. 1, with parameters $T = 3$ time-slots (feedback delay) and $C = 3$ (HARQ transmission credits). Hence, layer 1 acts as a HARQ mechanism with $T = 3$ parallel processes, while layer 2 represents the superposed packets, chosen according to the rules detailed in Section III-A.

In layer 1, message $\mathbf{m}_1$ is sent at time-slot 1 through packet $\mathbf{p}_1(1)$. The receiver fails to decode and the NACK feedback $\mathcal{F}_1 = \{1\}_N$ is available to the transmitter just before time-slot 4, so that the transmitter sends $\mathbf{p}_1(2)$ at time-slot 4. Another NACK feedback $\mathcal{F}_4 = \{1\}_N$ on message $\mathbf{m}_1$ is available to the transmitter just before time-slot 7, and the transmitter sends $\mathbf{p}_1(3)$ at time-slot 7. The message $\mathbf{m}_1$ is dropped at the beginning of time-slot 10 due to time-out. In time-slots 2 and 3 the two other parallel HARQ processes corresponding to messages $\mathbf{m}_2$ and $\mathbf{m}_3$ are initiated. The feedback ($\mathcal{F}_3 = \{2,3\}_A, \{1\}_N$) corresponding to correct decoding of $\mathbf{m}_2$ and $\mathbf{m}_3$ but not of $\mathbf{m}_1$ is available to the transmitter at the beginning of time-slot 6, where the transmission of a new message $\mathbf{m}_4$ starts. Another new process is initiated at time-slot 8 with message $\mathbf{m}_5$. At

the end of time-slot 9, $\mathbf{m}_1$ is in time-out, therefore, the transmission of a new message $\mathbf{m}_6$, via $\mathbf{p}_6(1)$, starts at $t = 10$, and so on. Note that the number of parallel HARQ processes in layer 1 is always equal to the feedback delay $T = 3$.

Layer 2 contains the packets to be superposed to layer 1 according to Eq. (2). These redundant packets are selected according to the four rules in Section III-A. According to rule 2), there is no superposition in time-slot 1. At time-slot 2, according to rule 2), packet $\mathbf{p}_1(2)$ is superposed with power fraction $(1-\alpha)$. The transmit vector in time-slot 2 is hence $\mathbf{x}_2 = \sqrt{\alpha}\mathbf{p}_2(1) + \sqrt{1-\alpha}\mathbf{p}_1(2)$. Similarly, the layer 2 packets for time-slots 3, 4, and 5 are determined by rule 2). At time-slot 6 the only unsent packet corresponding to an unacknowledged message is $\mathbf{p}_1(3)$. This packet is chosen according to rules 1) and 2) and it is transmitted in advance to its transmission in layer 1. At time-slot 7, according to rule 4), no packet is superposed. During the next time-slots, the superposition of packets in the second layer continues according to these rules, as depicted in Fig. 1.

Notice that, due to the feedback delay of $T = 3$, $\mathbf{m}_2$ is retransmitted in layer 1 at time-slot 5 through packet $\mathbf{p}_2(2)$, although this message was successfully decoded by the receiver at the end of time-slot 3. The same remark applies to message $\mathbf{m}_3$ which is retransmitted in layer 2 via the packets $\mathbf{p}_3(2)$ and $\mathbf{p}_3(3)$ at $t = 4$ and $t = 5$, respectively. These *useless* retransmissions are inevitable since the transmitter takes decisions in advance to the reception of the receiver's feedback. However, the *early* retransmission of $\mathbf{p}_2(2)$ in layer 2 enabled the decoding of $\mathbf{m}_2$ at time-slot 3. In other words, $\mathbf{m}_2$ is delivered to the receiver with a small delay of 2 time-slots only (which is not possible using parallel HARQ protocol).

B. Receiver analysis

Due to multi-packet transmission, the received signals share common information. The receiver attempts to decode multiple messages at each time-slot, using the current and previous observations. In the following we discuss the receiver's observation window and buffer size. Then, we explain the multi-bit ACK/NACK feedback vector $\mathcal{F}_t$, and we specify the set of messages that the receiver attempts to decode at time-slot t, denoted by $\mathcal{M}_t$.

1) Buffer size: A received signal at time-slot t could share common information with another received signal at any previous time-slot.

For instance, in the protocol's realization in Section III-A1, decoding $\mathbf{m}_4$ at time-slot 12 could benefit from the observations in all the previous time-slots. More precisely, at time-slot 6, $\mathbf{p}_4(1)$

is superposed to $\mathbf{p}_1(3)$. Therefore, decoding $\mathbf{m}_1$ would help in decoding $\mathbf{m}_4$ by removing the interference at time-slot 6, hence increasing the accumulated mutual information at the receiver corresponding to $\mathbf{m}_4$, which helps in decoding $\mathbf{m}_4$. Vice versa, decoding $\mathbf{m}_4$ at time-slot 12 would help in decoding $\mathbf{m}_1$, since it removes the interference due to $\mathbf{p}_4(1)$ at time-slot 6, hence it increases the accumulated mutual information corresponding to $\mathbf{m}_1$ at the receiver, which helps in decoding $\mathbf{m}_1$.

As a result, an optimal decoder would require an unlimited buffer size, however the buffer size should be fixed for the following reasons:

- Decoding a message in timeout is not useful in most applications. If needed, a retransmission of this message could be handled by upper layer protocols.
- If a message is in timeout, it is more likely that the accumulated mutual information associated with this message at the receiver at time-slot t, which is provided by the transmissions before $t - CT$ time-slots, is low. In other words, the benefit of considering more than CT observations is low.
- In addition, the decoder becomes very complex if we consider an unlimited buffer size (both in practice and using information theoretic analysis).
- Moreover, the buffer size is limited in practice.

Therefore, the receiver's buffer consists of the last CT received signals. For decoding purposes, the receiver would consider the undecoded messages in this observation window as interference. Whereas, the decoded messages are removed from the observations, which enhances the decoding performance. Keeping in the buffer the most recent CT observations, which correspond to the most recent CT time-slots, is a trade-off between the decoder's performance and the buffer size.

In the example in Section III-A1, only the observations from $t = 4$ to $t = 12$ (included) are kept in the buffer at $t = 12$, which corresponds to the most recent CT observations ($C = 3$ and $T = 3$). Moreover, decoding $\mathbf{m}_1$ at time-slot 12 is not useful, or could be assigned to upper layer protocols, since $\mathbf{m}_1$ is timeout at $t = 12$. Moreover, decoding $\mathbf{m}_1$ failed before time-slot 12, hence it is more likely that the accumulated mutual information at the receiver related to $\mathbf{m}_1$ that is provided by the transmissions before time-slot 4, is low.

In summary, the buffer size is fixed to CT observations which induces sub-optimal decoding. However, the performance degradation due to the limited buffer size is marginal.

2) Feedback vector: At the end of time-slot t, the receiver considers the observations of the most recent CT time-slots. Since there are T parallel HARQ processes in this observation window,

there are at most T undecoded messages (that are not in timeout). Therefore, the receiver attempts to decode these messages. Considering the observation window and superposition coding, the system is equivalent to a Multiple Input Single Output (MISO) channel with T (virtual) users, where each user is associated with a message. The output of the receiver is the feedback vector $\mathcal{F}_t$, which will be available at the transmitter at the beginning of time-slot $t+T$. The feedback vector $\mathcal{F}_t$ contains the ACK/NACK bits corresponding to the messages that i) are object of decoding at time-slot t, and ii) will not be in timeout at time-slot $t+T$. Hereafter, we show the realization of the proposed protocol that was explained in Section III-A1 from the receiver's perspective. The set of messages to decode $\mathcal{M}_t$ at time-slot t and the receiver's buffer at $t = 12$ are shown in Fig. 1.

In this instance, $\mathbf{m}_1$ and $\mathbf{m}_4$ are object of decoding at time-slot 7, *i.e.* , $\mathcal{M}_7 = \{\mathbf{m}_1, \mathbf{m}_4\}$. Also, $\mathcal{F}_7$ contains the ACK/NACK bits corresponding to these messages. However, $\mathbf{m}_1$ will be in timeout by time-slot 10. Hence, $\mathcal{F}_8$ does not contain feedback information corresponding to $\mathbf{m}_1$. Notice that $\mathcal{F}_8$ will be available to the transmitter just before the start of time-slot 11. Moreover, attempting to decode all messages, including the ones that will be in timeout, is beneficial because it removes the interference that is introduced by superposition. This can be seen at time-slot 8 where the receiver attempts to decode $\mathbf{m}_1$, $\mathbf{m}_4$ and $\mathbf{m}_5$, *i.e.* , $\mathcal{M}_8 = \{\mathbf{m}_1, \mathbf{m}_4, \mathbf{m}_5\}$. Since $\mathbf{m}_1$ will be in timeout by time-slot 10, $\mathcal{F}_8$ contains only information about $\mathbf{m}_4$ and $\mathbf{m}_5$. However, attempting to decode $\mathbf{m}_1$ (which is in timeout) is beneficial since it allows to remove the interference on message $\mathbf{m}_4$ at time-slot 6.

IV. Information theoretic characterization of the receiver

In this Section, our objective is to characterize the conditions on the rate and on the channel realizations to know the acknowledged or non-acknowledged messages at each time, i.e., to describe $\mathcal{F}_t$. The conditions depend on the selected decoder. We propose to consider two different decoders:

- the Multi-layer based Decoder (MD): each involved layer at time t is seen as a signal to be decoded (even if in time-out). As each layer can be seen as a flow/user, we apply the optimal joint information-theoretic decoder coming from Multiple Access Channel (MAC) rate region.
- the Single-layer based Decoder (SD): each interfering layer to the layer of interest is seen as an additional Gaussian noise. We apply the optimal information-theoretic decoder when

interference is treated as noise.

We consider that

- $\mathcal{M}_t$ is the set of messages that the receiver is attempting to decode at time-slot t.
- $\mathcal{T}_t$ is the set of messages in time-out belonging to $\mathcal{M}_t$. The messages do not contribute to $\mathcal{F}_t$ but may help the decoder to work in a better way.
- $\mathcal{D}_t$ is the set of successfully decoded messages belonging to $\mathcal{M}_t$.
- Given $\mathcal{D}_t$, we define $\mathcal{R}_t(\mathcal{D}_t)$ as the achievable rate region where all messages in $\mathcal{D}_t$ are successfully decoded and none of the messages in $\mathcal{M}_t \setminus \mathcal{D}_t$ is.

According to the transmit protocol, if $\mathcal{D}_t$ is the set of successfully decoded messages at time t, then we are able to describe $\mathcal{F}_t$. For instance, at $t = 8$ in Fig. 1, we have $\mathcal{M}_8 = \{\mathbf{m}_1, \mathbf{m}_4, \mathbf{m}_5\}$ and $\mathcal{T}_8 = \{\mathbf{m}_1\}$. More precisely, if $\mathcal{D}_8 = \{\mathbf{m}_1, \mathbf{m}_4, \mathbf{m}_5\}$ or $\mathcal{D}_8 = \{\mathbf{m}_4, \mathbf{m}_5\}$, then we obtain the same $\mathcal{F}_8$ equal to $\{4, 5\}_A$.

In order to characterize $\mathcal{F}_t$, we need to evaluate the rate region $\mathcal{R}_t(\mathcal{D}_t)$ for each $\mathcal{D}_t \subseteq \mathcal{M}_t$. When $\mathcal{D}_t \neq \{\emptyset, \mathcal{M}_t\}$, $\mathcal{R}_t(\mathcal{D}_t)$ is obtained, by definition, taking the union of the rate regions where the messages in $\mathcal{D}_t$ are successfully decoded (alone or simultaneously with other messages in $\mathcal{M}_t \setminus \mathcal{D}_t$), and excluding the regions where the messages in $\mathcal{D}_t$ are simultaneously decoded with at least another message in $\mathcal{M}_t \setminus \mathcal{D}_t$. We first denote by $\mathcal{Q}_t(\mathcal{S})$ the rate region at time t where the messages in a set $\mathcal{S}$ are successfully decoded (by the considered decoder) and the messages outside $\mathcal{S}$ are modeled as noise. The region where the messages in $\mathcal{D}_t$, and possibly other messages in $\mathcal{M}_t$, are successfully decoded is the union of $\mathcal{Q}_t$ of any set of messages that includes $\mathcal{D}_t$, *i.e.*, $\bigcup_{\mathcal{D}_t \subseteq \mathcal{S}} \mathcal{Q}_t(\mathcal{S})$ [22]. The region where the messages in $\mathcal{D}_t$ are successfully decoded, simultaneously with at least another message in $\mathcal{M}_t$, is the union of $\mathcal{Q}_t$ of any set that includes $\mathcal{D}_t$ and at least another message from $\mathcal{M}_t \setminus \mathcal{D}_t$, *i.e.*, $\bigcup_{\mathcal{D}_t \subset \mathcal{S}, \mathcal{S} \neq \mathcal{D}_t} \mathcal{Q}_t(\mathcal{S})$ [22]. Therefore we obtain

$$\begin{aligned}
\mathcal{R}_t(\mathcal{D}_t) &= \left(\bigcup_{\mathcal{D}_t \subseteq \mathcal{S} \subseteq \mathcal{M}_t} \mathcal{Q}_t(\mathcal{S}) \right) \setminus \left(\bigcup_{\mathcal{D}_t \subset \mathcal{S}' \subseteq \mathcal{M}_t, \mathcal{S}' \neq \mathcal{D}_t} \mathcal{Q}_t(\mathcal{S}') \right) \\
&= \left(\bigcup_{\mathcal{D}_t \subseteq \mathcal{S} \subseteq \mathcal{M}_t} \mathcal{Q}_t(\mathcal{S}) \right) \bigcap \left(\overline{\bigcup_{\substack{\mathcal{D}_t \subset \mathcal{S}' \subseteq \mathcal{M}_t, \\ \mathcal{S}' \neq \mathcal{D}_t}} \mathcal{Q}_t(\mathcal{S}')} \right) = \mathcal{Q}_t(\mathcal{D}_t) \bigcap \left(\bigcap_{\mathcal{D}_t \subset \mathcal{S} \subseteq \mathcal{M}_t, \mathcal{S} \neq \mathcal{D}_t} \overline{\mathcal{Q}_t(\mathcal{S})} \right). \quad (3)
\end{aligned}$$

When $\mathcal{D}_t = \mathcal{M}_t$, Eq. (3) is replaced with the following equation since the second term in the

Right Hand Side (RHS) does not exist. So we have

$$\mathcal{R}_t(\mathcal{M}_t) = \mathcal{Q}_t(\mathcal{M}_t). \tag{4}$$

When $\mathcal{D}_t = \emptyset$, we use the fact that the set of $\mathcal{R}_t(\mathcal{S})$ with any possible $\mathcal{S} \subseteq \mathcal{M}_t$ is a partition of the rate space. Thus, we get that $\mathcal{R}_t(\emptyset)$ is the complementary of the union of all rate regions for $\mathcal{S} \neq \emptyset$, *i.e.*,

$$\mathcal{R}_t(\emptyset) = \overline{\bigcup_{\mathcal{S}\subseteq\mathcal{M}_t,\mathcal{S}\neq\emptyset} \mathcal{R}_t(\mathcal{S})} = \bigcap_{\mathcal{S}\subseteq\mathcal{M}_t,\mathcal{S}\neq\emptyset} \overline{\mathcal{R}_t(\mathcal{S})}. \tag{5}$$

We remark that characterizing $\mathcal{R}_t(\mathcal{D}_t)$, for any $\mathcal{D}_t \subseteq \mathcal{M}_t$, is equivalent to characterize each $\mathcal{Q}_t(\mathcal{S})$ for any $\mathcal{S}$ involved in Eqs. (3)-(4)-(5).

If we apply Eqs. (3)-(4)-(5) on Fig. 1, we obtain, for instance, the following sets for $t = 1, 2, 3$.

- Case $t = 1$: $\mathcal{M}_1 = \{\mathbf{m}_1\}$. So we have an unique $\mathcal{D}_1 = \{\mathbf{m}_1\}$. Therefore we have

$$\mathcal{R}_1(\mathbf{m}_1) = \mathcal{Q}_1(\mathbf{m}_1). \tag{6}$$

- Case $t = 2$: $\mathcal{M}_2 = \{\mathbf{m}_1, \mathbf{m}_2\}$. So we have $\mathcal{D}_2 = \{\mathbf{m}_1\}$, or $\mathcal{D}_2 = \{\mathbf{m}_2\}$, or $\mathcal{D}_2 = \{\mathbf{m}_1, \mathbf{m}_2\}$ or $\mathcal{D}_2 = \emptyset$. This leads to

$$\begin{aligned}
\mathcal{R}_2(\mathbf{m}_1) &= \mathcal{Q}_2(\mathbf{m}_1) \bigcap \overline{\mathcal{Q}_2(\mathbf{m}_1, \mathbf{m}_2)}, && (7)\\
\mathcal{R}_2(\mathbf{m}_2) &= \mathcal{Q}_2(\mathbf{m}_2) \bigcap \overline{\mathcal{Q}_2(\mathbf{m}_1, \mathbf{m}_2)}, && (8)\\
\mathcal{R}_2(\mathbf{m}_1, \mathbf{m}_2) &= \mathcal{Q}_2(\mathbf{m}_1, \mathbf{m}_2), && (9)\\
\mathcal{R}_2(\emptyset) &= \overline{\mathcal{Q}_2(\mathbf{m}_1)} \bigcap \overline{\mathcal{Q}_2(\mathbf{m}_2)} \bigcap \overline{\mathcal{Q}_2(\mathbf{m}_1, \mathbf{m}_2)}. && (10)
\end{aligned}$$

- Case $t = 3$: $\mathcal{M}_3 = \{\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3\}$. So we have $\mathcal{D}_3 = \{\mathbf{m}_1\}$, or $\mathcal{D}_3 = \{\mathbf{m}_2\}$, or $\mathcal{D}_3 = \{\mathbf{m}_3\}$, or $\mathcal{D}_3 = \{\mathbf{m}_1, \mathbf{m}_2\}$, or $\mathcal{D}_3 = \{\mathbf{m}_1, \mathbf{m}_3\}$, or $\mathcal{D}_3 = \{\mathbf{m}_2, \mathbf{m}_3\}$, or $\mathcal{D}_3 = \{\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3\}$, or $\mathcal{D}_3 = \emptyset$. This leads to

$$\begin{aligned}
\mathcal{R}_3(\mathbf{m}_1) &= \mathcal{Q}_3(\mathbf{m}_1) \bigcap \overline{\mathcal{Q}_3(\mathbf{m}_1, \mathbf{m}_2)} \bigcap \overline{\mathcal{Q}_3(\mathbf{m}_1, \mathbf{m}_3)} \bigcap \overline{\mathcal{Q}_3(\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3)}, && (11)\\
\mathcal{R}_3(\mathbf{m}_2) &= \mathcal{Q}_3(\mathbf{m}_2) \bigcap \overline{\mathcal{Q}_3(\mathbf{m}_1, \mathbf{m}_2)} \bigcap \overline{\mathcal{Q}_3(\mathbf{m}_2, \mathbf{m}_3)} \bigcap \overline{\mathcal{Q}_3(\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3)}, && (12)\\
\mathcal{R}_3(\mathbf{m}_3) &= \mathcal{Q}_3(\mathbf{m}_3) \bigcap \overline{\mathcal{Q}_3(\mathbf{m}_1, \mathbf{m}_3)} \bigcap \overline{\mathcal{Q}_3(\mathbf{m}_2, \mathbf{m}_3)} \bigcap \overline{\mathcal{Q}_3(\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3)}, && (13)\\
\mathcal{R}_3(\mathbf{m}_1, \mathbf{m}_2) &= \mathcal{Q}_3(\mathbf{m}_1, \mathbf{m}_2) \bigcap \overline{\mathcal{Q}_3(\mathbf{m}_1, \mathbf{m}_2\, \mathbf{m}_3)}, && (14)\\
\mathcal{R}_3(\mathbf{m}_1, \mathbf{m}_3) &= \mathcal{Q}_3(\mathbf{m}_1, \mathbf{m}_3) \bigcap \overline{\mathcal{Q}_3(\mathbf{m}_1, \mathbf{m}_2\, \mathbf{m}_3)}, && (15)
\end{aligned}$$

$$\mathcal{R}_3(\mathbf{m}_2, \mathbf{m}_3) = \mathcal{Q}_3(\mathbf{m}_2, \mathbf{m}_3) \bigcap \overline{\mathcal{Q}_3(\mathbf{m}_1, \mathbf{m}_2\, \mathbf{m}_3)}, \tag{16}$$

$$\mathcal{R}_3(\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3) = \mathcal{Q}_3(\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3), \tag{17}$$

$$\mathcal{R}_3(\emptyset) = \overline{\mathcal{Q}_3(\mathbf{m}_1)} \bigcap \overline{\mathcal{Q}_3(\mathbf{m}_2)} \bigcap \overline{\mathcal{Q}_3(\mathbf{m}_3)} \bigcap \overline{\mathcal{Q}_3(\mathbf{m}_1, \mathbf{m}_2)} \tag{18}$$

$$\bigcap \overline{\mathcal{Q}_3(\mathbf{m}_1, \mathbf{m}_3)} \bigcap \overline{\mathcal{Q}_3(\mathbf{m}_2, \mathbf{m}_3)} \bigcap \overline{\mathcal{Q}_3(\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3)}. \tag{19}$$

$$\tag{20}$$

In Fig. 1, at $t = 3$, no message is in time out which implies that all of them take part to $\mathcal{F}_3$. More precisely, we have assumed that the rate belongs to $\mathcal{R}_3(\mathbf{m}_2, \mathbf{m}_3)$ which leads to $\mathcal{F}_3 = \{2, 3\}_A, \{1\}_N$.

In order to pursue the rate region characterization, we need to describe the constraints on the rate R to be in $\mathcal{R}_t(\mathcal{D}_t)$. The characterization will depend on the decoder used at the receiver side. In Section IV-A, we assume that the decoder jointly decodes all involved messages. In Section IV-B, we assume that the decoder decodes each message of interest by considering all the other ones as additive noise.

We mention that our analysis assumes that the codewords are large enough to allow the use of standard information-theoretic results on capacity-achieving codes. This is often true for practical systems employing parallel HARQ, as in current cellular networks. For systems using short timeslots an adapted analysis may be performed using the results in [23].

A. Region characterization for the Multi-layer based Decoder (MD)

To describe $\mathcal{R}_t(\mathcal{D}_t)$ for any involved $\mathcal{D}_t$, we need to describe in the rate region $\mathcal{Q}_t(\mathcal{S})$, for any set of messages $\mathcal{S}$. We remind that R is the rate of each message in $\mathcal{S}$. According to [22], if $R \in \mathcal{Q}_t(\mathcal{S})$, then the rate satisfies

$$|\mathcal{U}|.R \leq I(X_{\mathcal{U}}; Y_t | X_{\mathcal{S}\setminus\mathcal{U}}), \ \forall \mathcal{U} \subseteq \mathcal{S}, \tag{21}$$

where

- Y_t is the set of observations (received signals) during the current time-slot and the most recent CT time-slots, *i.e.* , $Y_t = [\mathbf{y}_{t-CT}, \ldots, \mathbf{y}_t]$.
- $X_{\mathcal{U}}$ represents the sent packets relative to the messages in $\mathcal{U}$, and $X_{\mathcal{S}\setminus\mathcal{U}}$ is interpreted likewise.
- $I(X_{\mathcal{U}}; Y_t | X_{\mathcal{S}\setminus\mathcal{U}})$ is the mutual information between $X_{\mathcal{U}}$ and Y_t when $X_{\mathcal{S}\setminus\mathcal{U}}$ are assumed to be known.

We remind that the packets taking part to Y_t but whose the associated messages are not in $\mathcal{S}$ are treated as additive noise. We also have to keep in mind that messages sent but already decoded (which may occur due to the feedback delay) are directly removed at the receiver side, hence do not take part to Y_t, and are not present in $\mathcal{M}_t$ anymore.

In the following, we describe the rate inequalities for the involved sets $\mathcal{Q}_t(\mathcal{S})$ with $t = 1, 2, 3$. Extension to $t > 3$ is tedious but straightforward.

- Case $t = 1$: $\mathcal{M}_1 = \{\mathbf{m}_1\}$. According to Eq. (6), we just have to describe $\mathcal{Q}_1(\mathbf{m}_1)$. At $t = 1$, the available observations are

$$\mathbf{y}_1 = h(1)\mathbf{p}_1(1) + \mathbf{w}_1. \tag{22}$$

  Therefore, according to Eq. (21), we have

$$\mathcal{Q}_1(\mathbf{m}_1) = \{R \mid R \leq \log(1 + g(1))\}. \tag{23}$$

  In Fig. 1, we have assumed that this inequality was not satisfied which leads to $\mathcal{F}_1 = \{1\}_N$.

- Case $t = 2$: $\mathcal{M}_2 = \{\mathbf{m}_1, \mathbf{m}_2\}$. According to Eqs. (7)-(10), we have to describe $\mathcal{Q}_2(\mathbf{m}_1)$, $\mathcal{Q}_2(\mathbf{m}_2)$, and $\mathcal{Q}_2(\mathbf{m}_1, \mathbf{m}_2)$. The available observations to decode at $t = 2$ correspond to the received signals at $t = 2$ and $t = 1$. Thus we obtain

$$\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix} = \begin{bmatrix} h(1)\mathbf{1}_N & 0 \\ 0 & \sqrt{1-\alpha}h(2)\mathbf{1}_N \end{bmatrix} \begin{bmatrix} \mathbf{p}_1(1) \\ \mathbf{p}_1(2) \end{bmatrix} + \begin{bmatrix} 0 \\ \sqrt{\alpha}h(2)\mathbf{1}_N \end{bmatrix} \mathbf{p}_2(1) + \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \end{bmatrix}. \tag{24}$$

  According to Eq. (21) and [24], we have

  - $\mathcal{Q}_2(\mathbf{m}_1) = \{R \mid R \leq \log(1 + g(1)) + \log(1 + \frac{(1-\alpha)g(2)}{1+\alpha g(2)})\}$,
  - $\mathcal{Q}_2(\mathbf{m}_2) = \{R \mid R \leq \log(1 + \frac{\alpha g(2)}{1+(1-\alpha)g(2)})\}$,
  - $\mathcal{Q}_2(\mathbf{m}_1, \mathbf{m}_2) = \left\{ R \;\middle|\; \begin{array}{rcl} R & \leq & \log(1 + g(1)) + \log(1 + (1-\alpha)g(2)) \\ R & \leq & \log(1 + \alpha g(2)) \\ 2R & \leq & \log(1 + g(1)) + \log(1 + g(2)) \end{array} \right\}$.

  We deduce from these inequalities and Eqs. (7)-(10) that the $\mathcal{R}_2$ are as in Fig. 2 where the hatched parts belong to the red one. In Fig. 1, we have assumed that the rate R belongs to $\mathcal{R}_2(\emptyset)$ which leads to $\mathcal{F}_2 = \{1, 2\}_N$.

- Case $t = 3$: $\mathcal{M}_3 = \{\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3\}$. According to Eqs. (11)-(18), we have to describe $\mathcal{Q}_3(\mathbf{m}_1)$, $\mathcal{Q}_3(\mathbf{m}_2)$, $\mathcal{Q}_3(\mathbf{m}_3)$, $\mathcal{Q}_3(\mathbf{m}_1, \mathbf{m}_2)$, $\mathcal{Q}_3(\mathbf{m}_1, \mathbf{m}_3)$, $\mathcal{Q}_3(\mathbf{m}_2, \mathbf{m}_3)$, and $\mathcal{Q}_3(\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3)$. The available

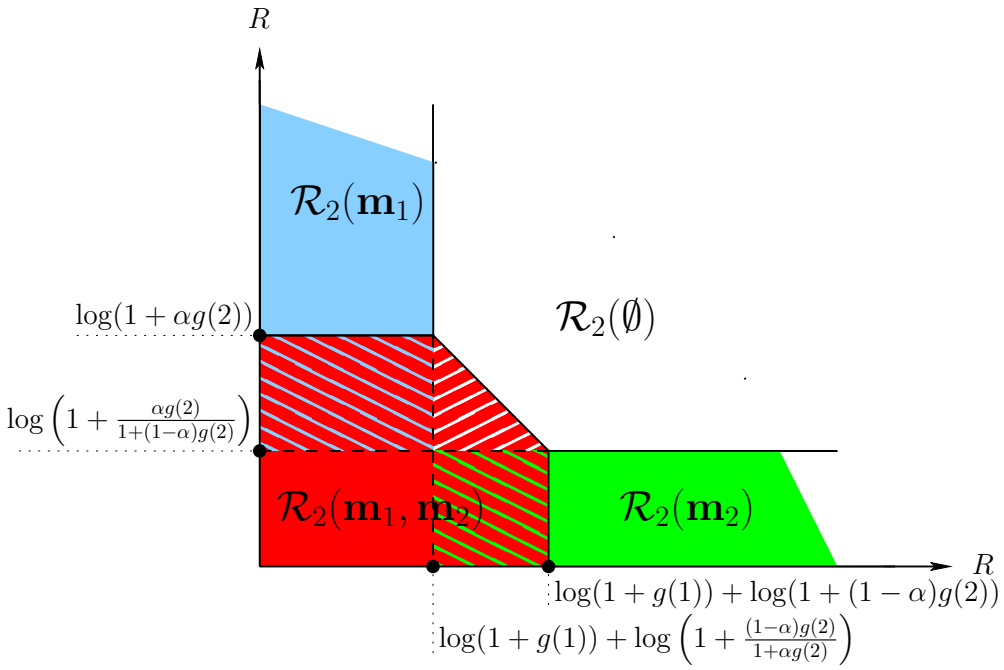

Figure 2: The regions $\mathcal{R}_2$ for the protocol described in Fig. 1.

observations to decode at $t = 3$ correspond to the received signals at $t = 3$, $t = 2$, and $t = 1$. Thus we obtain

$$\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \mathbf{y}_3 \end{bmatrix} = \begin{bmatrix} h(1)\mathbf{1}_N & 0 \\ 0 & \sqrt{1-\alpha}h(2)\mathbf{1}_N \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{p}_1(1) \\ \mathbf{p}_1(2) \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ \sqrt{\alpha}h(2)\mathbf{1}_N & 0 \\ 0 & \sqrt{1-\alpha}h(3)\mathbf{1}_N \end{bmatrix} \begin{bmatrix} \mathbf{p}_2(1) \\ \mathbf{p}_2(2) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \sqrt{\alpha}h(3)\mathbf{1}_N \end{bmatrix} \mathbf{p}_3(1) + \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \\ \mathbf{w}_3 \end{bmatrix}. \tag{25}$$

According to Eq. (21) and [24], we have

- $\mathcal{Q}_3(\mathbf{m}_1) = \{R \mid R \le \log(1+g(1)) + \log(1+\frac{(1-\alpha)g(2)}{1+\alpha g(2)})\}$,
- $\mathcal{Q}_3(\mathbf{m}_2) = \{R \mid R \le \log(1+\frac{\alpha g(2)}{1+(1-\alpha)g(2)}) + \log(1+\frac{(1-\alpha)g(3)}{1+\alpha g(3)})\}$,
- $\mathcal{Q}_3(\mathbf{m}_3) = \{R \mid R \le \log(1+\frac{\alpha g(3)}{1+(1-\alpha)g(3)})\}$,
- $\mathcal{Q}_3(\mathbf{m}_1,\mathbf{m}_2) = \left\{ R \;\middle|\; \begin{array}{rcl} R & \le & \log(1+g(1)) + \log(1+(1-\alpha)g(2)) \\ R & \le & \log(1+\alpha g(2)) + \log(1+\frac{(1-\alpha)g(3)}{1+\alpha g(3)}) \\ 2R & \le & \log(1+g(1)) + \log(1+g(2)) + \log(1+\frac{(1-\alpha)g(3)}{1+\alpha g(3)}) \end{array} \right\}$,

$$\circ\ \mathcal{Q}_3(\mathbf{m}_1,\mathbf{m}_3)=\left\{R\ \middle|\ \begin{array}{rcl} R & \leq & \log(1+g(1))+\log(1+\frac{(1-\alpha)g(2)}{1+\alpha g(2)}) \\ R & \leq & \log(1+\frac{\alpha g(3)}{1+(1-\alpha)g(3)}) \\ 2R & \leq & \log(1+g(1))+\log(1+\frac{(1-\alpha)g(2)}{1+\alpha g(2)})+\log(1+\frac{\alpha g(3)}{1+(1-\alpha)g(3)}) \end{array}\right\},$$

$$\circ\ \mathcal{Q}_3(\mathbf{m}_2,\mathbf{m}_3)=\left\{R\ \middle|\ \begin{array}{rcl} R & \leq & \log(1+\frac{\alpha g(2)}{1+(1-\alpha)g(2)})+\log(1+(1-\alpha)g(3)) \\ R & \leq & \log(1+\alpha g(3)) \\ 2R & \leq & \log(1+\frac{\alpha g(2)}{1+(1-\alpha)g(2)})+\log(1+g(3)) \end{array}\right\},$$

$$\circ\ \mathcal{Q}_3(\mathbf{m}_1,\mathbf{m}_2,\mathbf{m}_3)=\left\{R\ \middle|\ \begin{array}{rcl} R & \leq & \log(1+g(1))+\log(1+(1-\alpha)g(2)) \\ R & \leq & \log(1+\alpha g(2))+\log(1+(1-\alpha)g(3)) \\ R & \leq & \log(1+\alpha g(3)) \\ 2R & \leq & \log(1+g(1))+\log(1+g(2))+\log(1+(1-\alpha)g(3)) \\ 2R & \leq & \log(1+g(1))+\log(1+(1-\alpha)g(2))+\log(1+g(3)) \\ 2R & \leq & \log(1+\frac{\alpha g(2)}{1+(1-\alpha)g(2)})+\log(1+g(3)) \\ 3R & \leq & \log(1+g(1))+\log(1+g(2))+\log(1+g(3)) \end{array}\right\}.$$

In Fig. 1, we have assumed that the rate R belongs to $\mathcal{R}_3(\mathbf{m}_1)$ which leads to $\mathcal{F}_3 = \{2,3\}_N, \{1\}_A$.

B. Region characterization for the Single-layer based Decoder (SD)

In many practical receivers, the various packets associated with different messages are decoded separately by considering each other as a noise for the sake of simplicity. In this Section, we will describe the rate regions $\mathcal{R}_t(\mathcal{D}_t)$ when the decoder attempts to decode each involved message separately by assuming the other ones as additive noise. We continue to assume capacity-achieving codes.

Compared to Subsection IV-A, the unique difference is the way to characterize $\mathcal{Q}_t$. So Eqs. (3)-(18) remain valid. We just modify Eq. (21) and the inequalities describing any $\mathcal{Q}_t$.

As in Subsection IV-A, we describe the rate inequalities for the involved sets $\mathcal{Q}_t(\mathcal{S})$ with $t = 1, 2, 3$. Extension to $t > 3$ is tedious but straightforward.

- Case $t = 1$: According to Eq. (6), we just have to describe $\mathcal{Q}_1(\mathbf{m}_1)$. At $t = 1$, the available observations are given by Eq. (22). As there is an unique message involved in this observation, we obtain the same result as in previous Subsection. So

$$\mathcal{Q}_1(\mathbf{m}_1) = \{R \mid R \leq \log(1+g(1))\}. \tag{26}$$

- Case $t = 2$: $\mathcal{M}_2 = \{\mathbf{m}_1, \mathbf{m}_2\}$. According to Eqs. (7)-(10), we have to describe $Q_2(\mathbf{m}_1)$, $Q_2(\mathbf{m}_2)$, and $Q_2(\mathbf{m}_1, \mathbf{m}_2)$. The available observations to decode at $t = 2$ correspond to the received signals at $t = 2$ and $t = 1$, and are given by Eq. (24). We have
  - $Q_2(\mathbf{m}_1) = \{R \mid R \leq \log(1 + g(1)) + \log(1 + \frac{(1-\alpha)g(2)}{1+\alpha g(2)})\}$,
  - $Q_2(\mathbf{m}_2) = \{R \mid R \leq \log(1 + \frac{\alpha g(2)}{1+(1-\alpha)g(2)})\}$,
  - $Q_2(\mathbf{m}_1, \mathbf{m}_2) = \left\{ R \;\middle|\; \begin{array}{rcl} R & \leq & \log(1 + g(1)) + \log(1 + \frac{(1-\alpha)g(2)}{1+\alpha g(2)}) \\ R & \leq & \log(1 + \frac{\alpha g(2)}{1+(1-\alpha)g(2)}). \end{array} \right\}$

  For obtaining this $Q_2(\mathbf{m}_1, \mathbf{m}_2)$, we have assumed that the packets related to $\mathbf{m}_2$ were seen as noise when the decoder attempts to decode $\mathbf{m}_1$ and vice-versa. We deduce from these inequalities and Eqs. (7)-(10) that the $\mathcal{R}_2$ are as in Fig. 2 where the hatched blue/red part belongs to the blue one, the hatched green/red part belongs to the green one, and the hatched white/red part belongs to the white one.
- Case $t = 3$: $\mathcal{M}_3 = \{\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3\}$. According to Eqs. (11)-(18), we have to describe $Q_3(\mathbf{m}_1)$, $Q_3(\mathbf{m}_2)$, $Q_3(\mathbf{m}_3)$, $Q_3(\mathbf{m}_1, \mathbf{m}_2)$, $Q_3(\mathbf{m}_1, \mathbf{m}_3)$, $Q_3(\mathbf{m}_2, \mathbf{m}_3)$, and $Q_3(\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3)$. The available observations to decode at $t = 3$ correspond to the received signals at $t = 3$, $t = 2$, and $t = 1$, and are given by Eq. (25). We have
  - $Q_3(\mathbf{m}_1) = \{R \mid R \leq \log(1 + g(1)) + \log(1 + \frac{(1-\alpha)g(2)}{1+\alpha g(2)})\}$,
  - $Q_3(\mathbf{m}_2) = \{R \mid R \leq \log(1 + \frac{\alpha g(2)}{1+(1-\alpha)g(2)}) + \log(1 + \frac{(1-\alpha)g(3)}{1+\alpha g(3)})\}$,
  - $Q_3(\mathbf{m}_3) = \{R \mid R \leq \log(1 + \frac{\alpha g(3)}{1+(1-\alpha)g(3)})\}$,
  - $Q_3(\mathbf{m}_1, \mathbf{m}_2) = \left\{ R \;\middle|\; \begin{array}{rcl} R & \leq & \log(1 + g(1)) + \log(1 + \frac{(1-\alpha)g(2)}{1+\alpha g(2)}); \\ R & \leq & \log(1 + \frac{\alpha g(2)}{1+(1-\alpha)g(2)}) + \log(1 + \frac{(1-\alpha)g(3)}{1+\alpha g(3)}) \end{array} \right\}$,
  - $Q_3(\mathbf{m}_1, \mathbf{m}_3) = \left\{ R \;\middle|\; \begin{array}{rcl} R & \leq & \log(1 + g(1)) + \log(1 + \frac{(1-\alpha)g(2)}{1+\alpha g(2)}); \\ R & \leq & \log(1 + \frac{\alpha g(3)}{1+(1-\alpha)g(3)}) \end{array} \right\}$,
  - $Q_3(\mathbf{m}_2, \mathbf{m}_3) = \left\{ R \;\middle|\; \begin{array}{rcl} R & \leq & \log(1 + \frac{\alpha g(2)}{1+(1-\alpha)g(2)}) + \log(1 + \frac{(1-\alpha)g(3)}{1+\alpha g(3)}); \\ R & \leq & \log(1 + \frac{\alpha g(3)}{1+(1-\alpha)g(3)}) \end{array} \right\}$,
  - $Q_3(\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3) = \left\{ R \;\middle|\; \begin{array}{rcl} R & \leq & \log(1 + g(1)) + \log(1 + \frac{(1-\alpha)g(2)}{1+\alpha g(2)}); \\ R & \leq & \log(1 + \frac{\alpha g(2)}{1+(1-\alpha)g(2)}) + \log(1 + \frac{(1-\alpha)g(3)}{1+\alpha g(3)}); \\ R & \leq & \log(1 + \frac{\alpha g(3)}{1+(1-\alpha)g(3)}) \end{array} \right\}$.

The different areas described by Q_t are smaller when SD is employed instead of MD. The difference corresponds to the loss in performance due to the suboptimality of the SD based receiver.

V. Numerical results

In this section, we evaluate the performance of the proposed protocol with both decoders when capacity-achieving codes (CAC) are employed and also when practical coding schemes are carried out. In each case, conventional parallel HARQ is considered as a benchmark as well.

Except otherwise stated, the simulation setup is as follows: IR-HARQ with $R = 0.8$ is implemented as described in Section II. Each component of the transmit vector $\mathbf{x}_t$ is sent with energy E_s. According to Eq. (2), this energy E_s is shared between superposed packets with the power proportion α devoted to the first layer. The average channel gain is normalized, *i.e.*, $\mathbb{E}[|h(t)|^2] = 1$.

The considered performance metrics are the following ones: the throughput defined as the average number of correctly received information bits per channel use, the MER defined as the average ratio of dropped messages over the number of sent messages, and the average delay (shortened by delay) defined as the average number of elapsed time-slots between the first transmission and the last one for the messages correctly decoded.

Before going further, we analyze the influence of the parameter α in the performance. We consider the multi-layer decoder with capacity-achieving codes with $C = 3$ and $T = 3$. In Fig. 3, we plot the throughput (on left), MER (on left), and the delay (on right) versus α, for $E_s/N_0 = -4$dB, $E_s/N_0 = 0$dB, and $E_s/N_0 = 4$dB. For a given value of E_s/N_0, we observe that the

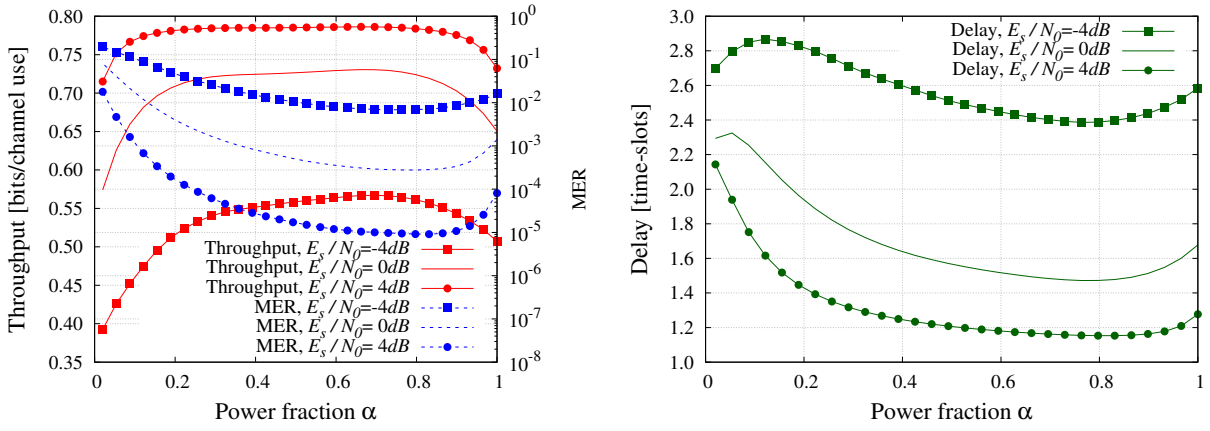

Figure 3: Throughput (on left), MER (on left), and Delay (on right) versus α.

optimal values for α are different for each performance metric (for example, for $E_s/N_0 = 0$dB, we have that $\alpha = 0.67$ maximizes the throughput, $\alpha = 0.79$ minimizes the MER, and $\alpha = 0.78$

minimizes the delay), but there is a plateau around each optimal value. This implies that applying the optimal value of α associated with one metric does not disadvantage the other ones strongly. However, α can be tuned according to the application requirements based on the sole knowledge of the average channel gain. As visible in Fig. 3, the optimum values of α depend on the considered E_s/N_0, although their variation is modest in the considered range. The value of α maximizing the throughput is $\alpha = 0.75$ for $E_s/N_0 = -4$dB, $\alpha = 0.68$ for $E_s/N_0 = 0$dB, and $\alpha = 0.67$dB for $E_s/N_0 = 4$dB. *In the remainder of this Section, for the sake of simplicity, at each E_s/N_0, we select the value of α leading to the highest throughput even if we are looking at the other performance metrics. In addition the optimal α is adapted to each decoder.*

A. Performance for multi-layer decoder with capacity-achieving codes

We analyze the performance of our proposed protocol with the multi-layer decoder (applied on capacity-achieving codes) and compare them with the conventional parallel HARQ. In this Subsection, we consider $C = 3$ and $T = 3$, as done for the example provided in Fig 1 on Section III-A1.

In Fig. 4, we plot the throughput versus E_s/N_0. The proposed protocol offers a significant

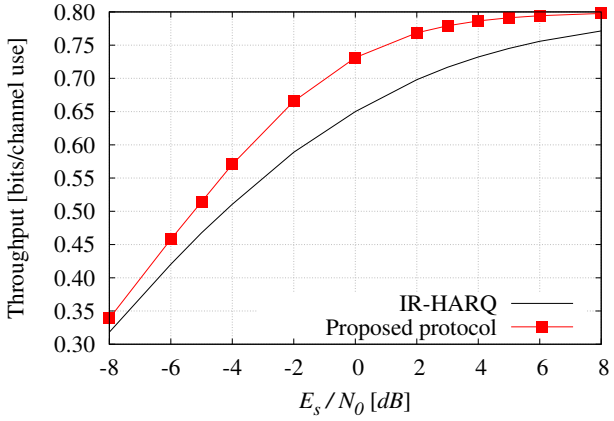

Figure 4: Throughput versus E_s/N_0 (MD, CAC, $C = 3$, $T = 3$).

throughput gain compared to parallel HARQ at any Signal-to-Noise Ratio (SNR). For instance, the gain in SNR is around 2dB at moderate SNR. In $E_s/N_0 = 10$dB, the gain in throughput is about 10%. In addition, the throughput of the proposed protocol goes faster to the asymptotic

value (here equal to $R = 0.8$) than the parallel HARQ. Consequently, our proposed protocol offers a significant gain in throughput.

In Fig. 5, we plot MER versus E_s/N_0. We observe that the proposed protocol achieves much

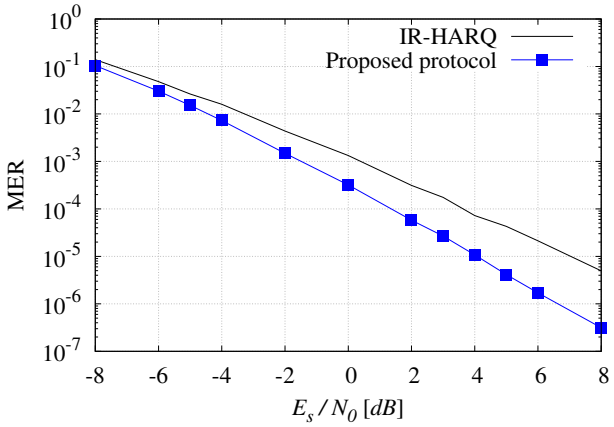

Figure 5: MER versus E_s/N_0 (MD, CAC, $C = 3$, $T = 3$).

lower MER than the parallel HARQ protocol since at MER $= 10^{-5}$ (typical value for ultra-reliable communications), the gain in SNR is around 3dB. This can be explained by the diversity order achieved by our protocol. With $C = 3$, the parallel HARQ yields a diversity of 3 since each message is transmitted at most C times. This diversity order is indeed obtained in Fig. 5. For our proposed protocol, if the message is not well decoded, it is sent C times plus at least once on the second layer. Consequently, the diversity order is at least $C+1$. When $C = 3$ and $T = 3$ as in the example of Fig 1, the diversity order is actually 4 since the message $\mathbf{m}_4$ (which is never positively acknowledged) is sent at time-slots 6, 8, 9, and 12. This diversity order is effectively obtained in Fig. 5. Notice also that if the MER is the performance metric of interest for a specific application, an other protocol could be advocated by forcing more retransmissions in order to offer a higher diversity order at the expense of the delay.

In Fig. 6, we plot the delay versus E_s/N_0. The gain is slight especially at low and high SNR. To better analyze the behavior of our proposed protocol, we suggest to look at the delay distribution rather than the average delay. For each correctly decoded message, we evaluate the number of elapsed time-slots denoted by d. The delay distribution is defined as the histogram of the variable d.

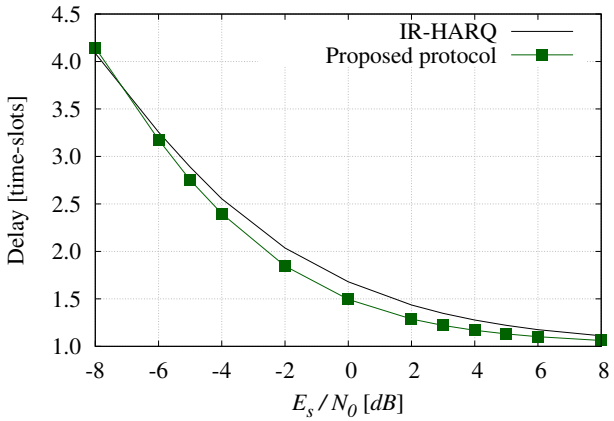

Figure 6: Delay versus E_s/N_0 (MD, CAC, $C = 3$, $T = 3$).

In Fig. 7, we plot the delay distribution versus E_s/N_0. In the figure, one bar corresponds to one delay distribution. In the bar, the height of each box corresponds to the proportion (between 0 and 1) of the mentioned value for d in the legend. We consider that the delay distribution is

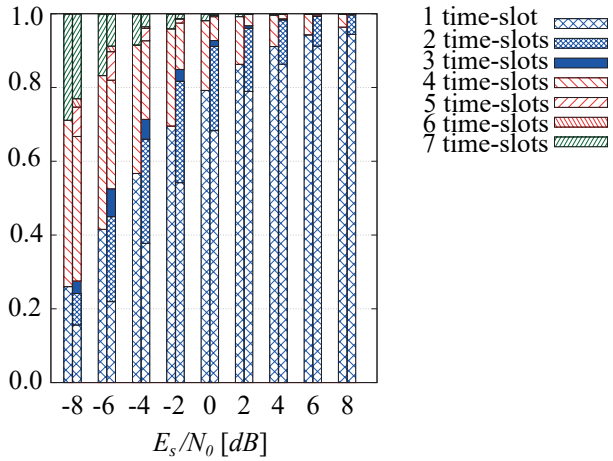

Figure 7: Delay distribution versus E_s/N_0 (MD, CAC, $C = 3$, $T = 3$). At each SNR, the left bar stands for the parallel HARQ while the right bar stands for the proposed protocol.

better for our proposed protocol because the probability to receive a message with large delay is smaller. The main reason lies in the granularity in d offered by our protocol. Indeed, the parallel HARQ allows only the values 1 (first transmission), 4 (second transmission), and 7 (third and

so last transmission) for d, while our protocol offers any value between 1 and 7 due to the superposed layer. For both protocols, the maximum value for d is $(C-1)T+1$ leading to 7.

Notice that similar behaviors have been obtained for other set of parameters C, T, and R, but not reported here due to space limitation.

B. Performance for single-layer decoder with capacity-achieving codes

We hereafter analyze the performance of our proposed protocol when SD is carried out with related CAC. We compare it to our protocol when MD is used also with CAC and the parallel HARQ. Hereafter, we consider $C = 3$ but $T = 8$ in order to be closer to LTE standards [11].

In Fig. 8, we display the throughput versus E_s/N_0. The proposed protocol with SD still offers

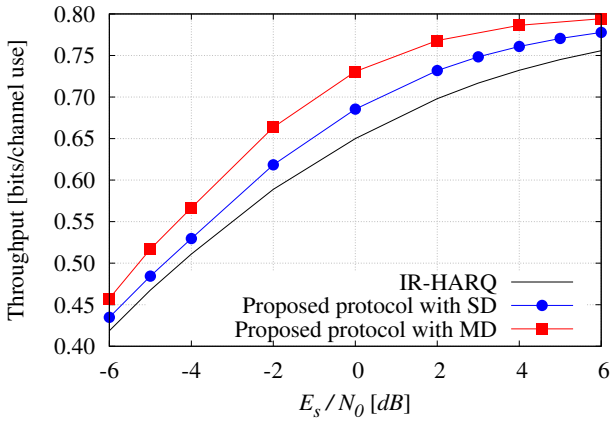

Figure 8: Throughput versus E_s/N_0 (MD/SD, CAC, $C = 3$, $T = 8$).

a significant throughput gain with respect to the parallel HARQ. The gain in throughput for SD is almost half the gain achieved with MD, and so is around 1dB. Once again, the throughput even with SD goes faster to the asymptotic value.

In Fig. 9, we display MER versus E_s/N_0. Once again, the MER even with SD is better than in parallel HARQ. Actually the diversity order is the same for both decoders related to our protocol, explaining the same performance. But, as the throughput is smaller for SD than for MD, the SD needs more retransmissions and may stay more time in the HARQ mechanism to achieve the same MER. Next Figure devoted to delay confirms this statement.

The average delay is not reported here since as seen in Section V-A the delay distribution is more relevant to analyze our protocol. In Fig. 10, we display the delay distribution versus

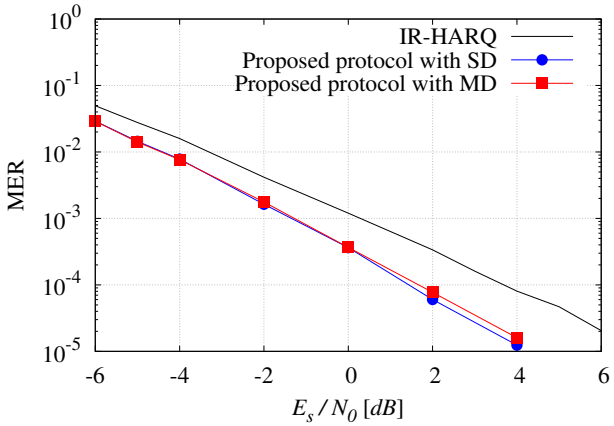

Figure 9: MER versus E_s/N_0 (MD/SD, CAC, $C = 3$, $T = 8$).

E_s/N_0. Once again, the delay distribution even with SD is better since the probability to receive

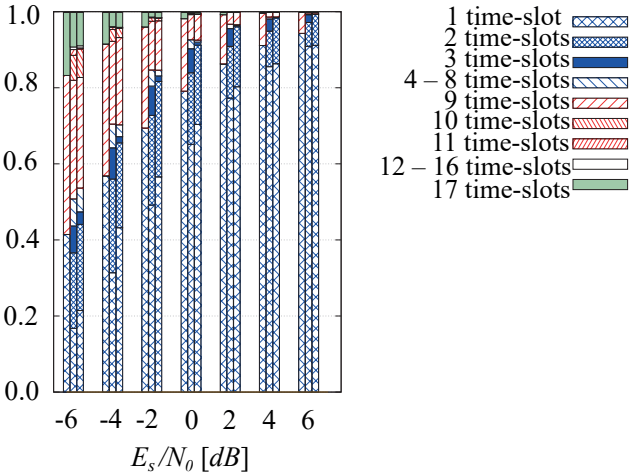

Figure 10: Delay distributions versus E_s/N_0 (CAC, $C = 3$, $T = 8$). At each SNR, the left bar stands for the parallel HARQ, the central bar stands for the proposed protocol with SD while the right bar stands for the proposed protocol with MD.

the message with a large delay is smaller. The distribution for MD is slightly better than for SD which explains the degradation in the throughput observed in Fig. 8. Nevertheless, the proposed protocol is robust to a worse decoder than the optimal one since it still offers better performance than the parallel HARQ. When SD is employed, the complexity is of the same order of magnitude

as the parallel HARQ. Indeed, at the transmitter side, our protocol just needs negligible extra additions, and at the receiver side, the complexity of SD per message and time-slot is the same as in parallel HARQ since inter-layer interference is seen as noise. Only the number of time-slots per message increases since we attempt to decode each message more often.

C. Performance with capacity-achieving codes in more general setups

As the end of time-slot t corresponds to the beginning to time-slot $(t+1)$, in case of low mobility, $h(t)$ may be not independent of $h(t-1)$. Therefore, in Fig. 11, we plot the throughput versus E_s/N_0 when MD is used and the channel has a Markov model given by

$$h(t) = \beta.h(t-1) + \sqrt{1-\beta^2}.e(t)$$

where $e(t)$ is an additive white Gaussian noise with zero-mean and variance σ_h^2. We fix $\beta = 0.5$.

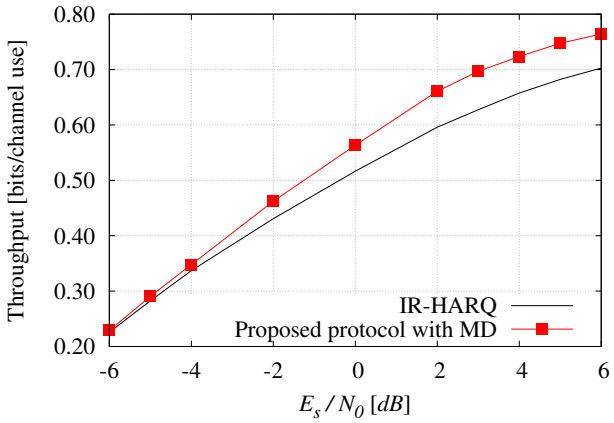

Figure 11: Throughput versus E_s/N_0 (MD, CAC, $C = 3$, $T = 8$, correlated channel).

We observe that the gain in performance is maintained although the packets related to a given message encounter fewer independent channel realizations due to their time correlation.

We also tested our protocol for MD with $C = 3$, $T = 8$ but a non-vanishing decoding time denoted by T_p. We fix $T_p = 3$ as in LTE standard [11]. The presence of a non-vanishing decoding time leads to modify the receiver design into two ways

- At time-slot t, the messages selected to be decoded are those present in the buffer $[y_{t-CT}, \cdots y_t]$ such that they are not in timeout when the decoder has completed its decoding. As the

decoding time is T_p, the messages involved for the first time in $[y_{t-CT+T_p}, \cdots y_t]$ will be considered of interest (we remind that CT is the timeout value). Other messages will be considered as noise. Consequently, the set of messages seen as noise may increase compared to the case $T_p = 0$.

- We remind that the contributions of a correctly decoded message are removed for the future observations. For, $T_p = 0$, once a message in $[y_{CT}, \cdots, y_t]$ is decoded, it is removed from $y_{t'}$ with $t' > t$. When $T_p \neq 0$, the message can be removed only from $y_{t'}$ with $t' > t + T_p$. Between time-slot t and $t + T_p$, the message is still under consideration for decoding if not in timeout and seen as noise if in timeout. Once again, the set of messages seen as noise may increase compared to the case $T_p = 0$.

As we observe only an insignificant difference with the throughput given in Fig. 8, we do not report the simulation results.

D. Performance for a practical decoder with practical codes

We analyze the performance of our proposed protocol with a practical decoder (PD) (described later) when practical convolutional codes (PCC) are used. We still continue to assume $C = 3$, $T = 8$. The bits associated with one message are encoded using Rate-Compatible Punctured Convolutional (RCPC) codes with a mother code's coding rate $R_0 = 1/4$. In order to ensure a information rate $R = 0.8$ (in the first transmission), we consider puncturing tables of memory 4 and period 8, as in [25]. This leads to 3 codeword chunks of equal size. The codeword chunks are then modulated using Binary Phase Shift Keying (BPSK) prior to transmission and stack into the packet sent over the channel.

In the following, we describe the implemented PD. The suggested decoder is just slightly more complex than the SD and can be applied in practical devices.

- As one packet $\mathbf{p}_k(\ell)$ may be spread over several time-slots (according to the protocol realization where one packet transmitted on layer 1 can be also available in other time-slots in layer 2), we first perform a Chase combining of the received samples located in these time-slots. Let $\mathbf{z}_k(\ell) = \{z_{k,n}(\ell)\}_{n=1,\ldots,N}$ be the output of the Chase combiner related to the packet $\mathbf{p}_k(\ell)$. It can be written as

$$\mathbf{z}_k(\ell) = \beta \mathbf{p}_k(\ell) + \sum_{k' \in \mathcal{K}_{k,\ell}} \beta_{k'} \mathbf{p}_{k'}(\ell_{k'}) + \mathbf{w} \tag{27}$$

with $\mathbf{w}$ a Gaussian noise with zero-mean and variance N_0, and $\mathcal{K}_{k,\ell}$ the set of other packets involved in the time-slots where $\mathbf{p}_k(\ell)$ is present. The scalars β and $\{\beta_{k'}\}_{k'\in\mathbf{K}_{k,\ell}}$ stand for the weights obtained after the Chase combiner.

- Then we calculate the Log Likelihood Ratio (LLR) of each bit of the packet $\mathbf{k}(\ell)$ based on $\mathbf{z}_k(n)$ without using the structure of the code. As BPSK is employed, the bit $b_{k,n}(\ell)$ (corresponding to the n-th bit of packet $\mathbf{p}_k(\ell)$ is only involved in the sample $z_{k,n}(\ell)$. According to Eq. (27), the LLR of $b_{k,n}(\ell)$, denoted by $\mathrm{LLR}(b_{k,n}(\ell))$, is given by

$$\mathrm{LLR}(b_{k,n}(\ell)) = \log \frac{P(z_{k,n}(\ell)|b_{k,n}(\ell)=0)}{P(z_{k,n}(\ell)|b_{k,n}(\ell)=1)} \tag{28a}$$

$$= \log \frac{\sum_{\{d_{k'}\}\in\{-\sqrt{E_s},\sqrt{E_s}\}^{|\mathcal{K}_{k,\ell}|}} \exp\left(-\frac{|z_{k,n}(\ell)-\beta-\sum_{k'\in\mathcal{K}_{k,\ell}}\beta_{k'}d_{k'})|^2}{N_0}\right)}{\sum_{\{d_{k'}\}\in\{-\sqrt{E_s},\sqrt{E_s}\}^{|\mathcal{K}_{k,\ell}|}} \exp\left(-\frac{|z_{k,n}(\ell)+\beta-\sum_{k'\in\mathcal{K}_{k,\ell}}\beta_{k'}d_{k'}|^2}{N_0}\right)}, \tag{28b}$$

  We so obtain a sequence of LLR for each coded bit of each message (by doing the previous derivations for any k and ℓ).
- Then the sequence of LLR associated with one message is passed into a channel coding decoder (Viterbi's decoder in our case since the code is convolutional). This operation is done for each message involved at the current time.
- Then a SIC is applied in such a way: the messages involved at the current time correctly decoded in the previous set are removed from Eq. (27) associated with the uncoded messages. And the LLR of the uncoded messages are then updated once and also passed into their Viterbi's decoder.
- Finally, a ACK is sent if the message is correctly decoded, and a NACK otherwise.

Let us now analyze the performance. In Fig. 12, we show the throughput versus E_s/N_0. We still observe a gain for our proposed protocol, typically, around 1dB for the throughput of interest. Once again, the asymptotic value is reached much faster. Finally Fig. 12 is very close to Fig. 8 with a 4dB shift in SNR when the PD is assimilated to the SD.

In Fig. 13, we show the MER versus E_s/N_0. For medium MER (around 10^{-3}), we have a 2dB gain in SNR which is significant. The expected diversity orders are also achieved despite our suboptimal decoder. And we also observe that Fig. 13 is close to Fig. 9 with about a 4dB shift in SNR when the PD is assimilated to the SD.

In Fig. 14, we show the delay distribution versus E_s/N_0. Once again, the delay distribution is better with our proposed protocol since the probability to deliver with a high delay is smaller. As

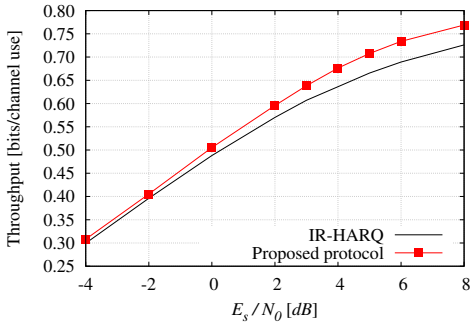

Figure 12: Throughput versus E_s/N_0 (PD, PCC, $C = 3$, $T = 8$).

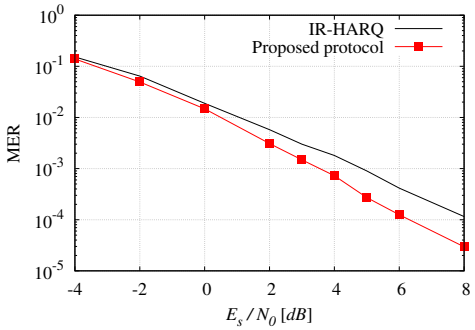

Figure 13: MER versus E_s/N_0 (PD, PCC, $C = 3$, $T = 8$).

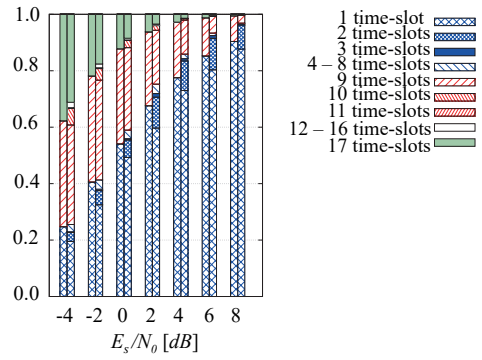

Figure 14: Delay distributions versus E_s/N_0 (PD, PCC, $C = 3$, $T = 8$). At each SNR, the left bar stands for the parallel HARQ while the right bar stands for the proposed protocol.

a conclusion, even with a low complex PD (since it requires roughly one extra SIC iteration), the proposed protocol outperforms the conventional parallel HARQ. These observed performance

validate the proof of concept for this protocol.

VI. Conclusion

This paper has introduced an enhanced version of parallel HARQ by allowing multi-packet transmissions. The main idea is to allow a superposition of the current packet with a packet of another message even before having received any feedback about it. With extensive numerical simulations (with various receivers, various channel codes), we showed that the proposed protocol outperforms the conventional parallel HARQ for the considered metrics (throughput, MER, average delay, delay distribution) at the expense of a slightly higher complexity but without additional signalling (just standard ACK/NACK) and without any modification of the communication system infrastructure.

Future works related to this topic are numerous: *i)* more complex practical receivers may be designed in order to increase the gap between the proposed protocol and the conventional one. *ii)* The proposed protocol is well adapted to throughput and delay distribution but may be modified to highlight other performance metrics. More layers can be also suggested. More importantly, we do not know what is the best protocol according to a certain metric. *iii)* Therefore more theoretical analysis of the protocol and the potential benchmarks have to be conducted.

References

[1] A. Khreis, P. Ciblat, F. Bassi, and P. Duhamel, "Multi-Packet HARQ with delayed feedback," in *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Sept. 2018.

[2] S. Lin, D. J. Costello, and M. J. Miller, "Automatic-repeat-request error-control schemes," *IEEE Communications Magazine*, vol. 22, pp. 5–17, Dec. 1984.

[3] D. Chase, "Code combining - a maximum-likelihood decoding approach for combining an arbitrary number of noisy packets," *IEEE Trans. Commun.*, vol. 33, pp. 385–393, May 1985.

[4] D. Mandelbaum, "An adaptive-feedback coding scheme using incremental redundancy," *IEEE Trans. Inf. Theory*, vol. 20, pp. 388–389, May 1974.

[5] M. Zorzi and R. Rao, "On the use of renewal theory in the analysis of ARQ protocols," *IEEE Trans. Commun.*, vol. 44, pp. 1077–1081, Sept. 1996.

[6] L. Badia, M. Levorato, and M. Zorzi, "Markov analysis of selective repeat type ii hybrid arq using block codes," *IEEE Trans. Commun.*, vol. 56, pp. 1434–1441, Sept. 2008.

[7] D. Tuninetti and G. Caire, "The throughput of Hybrid ARQ protocols for the Gaussian collision channel," *IEEE Trans. Inf. Theory*, vol. 47, pp. 1971–1988, July 2001.

[8] E. Soljanin, N. Varnica, and P. Whiting, "Incremental redundancy Hybrid ARQ with LDPC and raptor codes," *IEEE Trans. Inf. Theory*, Sept. 2005.

[9] C. Lott, O. Milenkovic, and E. Soljanin, "Hybrid ARQ: theory, state of the art and future directions," in *International Workshop on Information Theory (ITW)*, IEEE, July 2007.

[10] S. Lin and D. J. Costello, *Error Control Coding, Second Edition*. Prentice-Hall, 2004.

[11] S. Sesia, I. Toufik, and M. Baker, *LTE, The UMTS Long Term Evolution: From Theory to Practice*. Wiley, 2009.

[12] L. Szczecinski, S. R. Khosravirad, P. Duhamel, and M. Rahman, "Rate Allocation and Adaptation for Incremental Redundancy Truncated HARQ," *IEEE Trans. Commun.*, vol. 61, pp. 2580–2590, June 2013.

[13] M. E. Aoun, R. L. Bidan, X. Lagrange, and R. Pyndiah, "Multiple-packet versus single-packet incremental redundancy strategies for type-II Hybrid ARQ," in *IEEE International Symposium on Turbo Codes Iterative Information Processing*, Sept. 2010.

[14] M. E. Aoun, X. Lagrange, R. L. Bidan, and R. Pyndiah, "Analysis and optimization of hybrid single packet and multiple-packets incremental redundancy in the presence of channel state information," in *IEEE International Symposium on Wireless Personal Multimedia Communications (WPMC)*, Oct. 2011.

[15] K. F. Trillingsgaard and P. Popovski, "Generalized HARQ Protocols with Delayed Channel State Information and Average Latency Constraints," *IEEE Trans. Inf. Theory*, vol. 64, pp. 1262–1280, Feb. 2018.

[16] A. Steiner and S. Shamai, "Multi-layer broadcasting hybrid-ARQ strategies for block fading channels," *IEEE Trans. Wireless Commun.*, vol. 7, pp. 2640–2650, July 2008.

[17] A. N. Assimi, C. Poulliat, and I. Fijalkow, "Packet combining for multi-layer hybrid-ARQ over frequency-selective fading channels," in *European Signal Processing Conference (EUSIPCO)*, Aug. 2009.

[18] R. Zhang and L. Hanzo, "Superposition-Coding-Aided Multiplexed Hybrid ARQ Scheme for Improved End-to-End Transmission Efficiency," *IEEE Trans. Veh. Technol.*, vol. 58, pp. 4681–4686, Oct. 2009.

[19] F. Takahashi and K. Higuchi, "HARQ for Predetermined-Rate Multicast Channel," in *IEEE Vehicular Technology Conference (VTC)*, May 2010.

[20] A. E. Hamss, L. Szczecinski, and P. Piantanida, "Increasing the throughput of HARQ via multi-packet transmission," in *IEEE Global Communications Conference (GLOBECOM)*, Dec. 2014.

[21] M. Jabi, A. E. Hamss, L. Szczecinski, and P. Piantanida, "Multipacket Hybrid ARQ: Closing Gap to the Ergodic Capacity," *IEEE Trans. Commun.*, vol. 63, pp. 5191–5205, Dec. 2015.

[22] B. Bandemer, A. E. Gamal, and Y. H. Kim, "Simultaneous nonunique decoding is rate-optimal," in *Allerton Conference on Communication, Control, and Computing*, Oct. 2012.

[23] B. Makki, T. Svensson, and M. Zorzi, "Finite block-length analysis of the Incremental Redundancy HARQ," *IEEE Commun. Lett.*, vol. 3, pp. 529–532, Oct. 2014.

[24] A. Goldsmith, S. A. Jafar, N. Jindal, and S. Vishwanath, "Capacity limits of MIMO channels," *IEEE J. Sel. Areas Commun.*, vol. 21, pp. 684–702, May 2003.

[25] J. Hagenauer, "Rate-compatible punctured convolutional codes (RCPC codes) and their applications," *IEEE Transactions on Communications*, vol. 36, pp. 389–400, Apr. 1988.