



## Delay Reduction for Instantly Decodable Network Coding in Persistent Channels With Feedback Imperfections

Item Type	Article
Authors	Douik, Ahmed S.;Sorour, Sameh;Al-Naffouri, Tareq Y.;Alouini, Mohamed-Slim
Citation	Delay Reduction for Instantly Decodable Network Coding in Persistent Channels With Feedback Imperfections 2015, 14 (11):5956 IEEE Transactions on Wireless Communications
Eprint version	Post-print
DOI	<a href="https://doi.org/10.1109/TWC.2015.2445338">10.1109/TWC.2015.2445338</a>
Publisher	Institute of Electrical and Electronics Engineers (IEEE)
Journal	IEEE Transactions on Wireless Communications
Rights	(c) 2015 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other users, including reprinting/ republishing this material for advertising or promotional purposes, creating new collective works for resale or redistribution to servers or lists, or reuse of any copyrighted components of this work in other works.
Download date	2024-04-17 02:25:08
Link to Item	<a href="http://hdl.handle.net/10754/594982">http://hdl.handle.net/10754/594982</a>

# Delay Reduction for Instantly Decodable Network Coding in Persistent Channels with Feedback Imperfections

Ahmed Douik, *Student Member, IEEE*, Sameh Sorour, *Member, IEEE*,  
Tareq Y. Al-Naffouri, *Member, IEEE*, and Mohamed-Slim Alouini, *Fellow, IEEE*

**Abstract**—This paper considers the multicast decoding delay reduction problem for generalized instantly decodable network coding (G-IDNC) over persistent erasure channels with feedback imperfections. The feedback scenario discussed is the most general situation in which the sender does not always receive acknowledgments from the receivers after each transmission, and the feedback communications are subject to loss. The decoding delay increment expressions are derived and employed to express the decoding delay reduction problem as a maximum weight clique problem in the G-IDNC graph. This paper provides a theoretical analysis of the expected decoding delay increase at each time instant. Problem formulations in simpler channel and feedback models are shown to be special cases of the proposed generalized formulation. Since finding the optimal solution to the problem is known to be NP-hard, a sub-optimal greedy algorithm is designed and compared to blind approaches proposed in the literature. Through extensive simulations, the proposed algorithm is shown to outperform the blind methods in all situations and to achieve significant improvement, especially for high time-correlated channels.

**Index Terms**—Multicast channels, persistent erasure channels, G-IDNC, decoding delay, lossy intermittent feedback.

## I. OVERVIEW

### A. Introduction

In the past decade, *Network Coding* (NC) emerged as a promising technique to improve throughput and delay over wireless erasure channels [2]–[6]. Fundamental research has been conducted to approach the network capacity. However, a better use of channel or throughput is not equivalent, in general, to a lower delay at the application level [7]. This throughput-delay tension [8] is mainly caused by the erasure nature of links that affects the delivery of meaningful data.

A part of this paper [1] appeared in proc. of IEEE International Conference on Wireless and Mobile Computing, Networking, and Communications (WiMob'13), Lyon, France.

This work was done while the first author was conducting his master thesis at King Abdullah University of Science and Technology (KAUST), Thuwal, Makkah Province, Saudi Arabia.

Ahmed Douik is with the Electrical Engineering Department at the California Institute of Technology (Caltech), Pasadena, CA, USA. e-mail: ahmed.douik@caltech.edu.

Sameh Sorour is with the Electrical Engineering Department at King Fahd University of Petroleum and Minerals (KFUPM), Dhahran, Eastern Province, Saudi Arabia, email:samehsorour@kfupm.edu.sa

Tareq Y. Al-Naffouri is with both King Abdullah University of Science and Technology (KAUST), Thuwal, Makkah Province, Saudi Arabia, and King Fahd University of Petroleum and Minerals (KFUPM), Dhahran, Eastern Province, Saudi Arabia, e-mail: tareq.alnaffouri@kaust.edu.sa.

Mohamed-Slim Alouini is with Computer, Electrical and Mathematical Sciences and Engineering (CEMSE) Division at King Abdullah University of Science and Technology (KAUST), Thuwal, Makkah Province, Saudi Arabia, email: slim.alouini@kaust.edu.sa

This work is supported by KAUST project no. EE002355 at the Research Institute, King Fahd University of Petroleum and Minerals.

$$\mathbf{F} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix} \text{ IDNC packets: } 1 \oplus 2 \text{ and } 3 \oplus 4$$

Fig. 1. Benefits of IDNC in a network composed of 3 receivers and 4 packets.  $F_{ij}$  is 1 if user  $i$  needs packet  $j$ . In IDNC, the 2 packets are instantly decodable by all receivers hence there is no delay. In LRNC, the mix of all the packets is sent twice, and the decoding is possible for receivers 2 and 3 only after receiving both combinations which translates into a delay of 2.

As a consequence, receivers are unable to decode the mixed information flows synchronously. Designing NC algorithms that enable a trade-off between throughput and delay has been shown to be challenging [9]. This paper considers the optimization of the individual delays, known as the decoding delay [10], [11], experienced when delivered packets are not useful at their reception instant.

Recently, an important subclass of network coding, called Instantly Decodable Network Coding (IDNC) [12]–[17], gained much attention thanks to its immediate decodability property. Such feature makes received packets useful at their reception moment resulting in a lower decoding delay as shown in Figure 1. Moreover, unlike some NC schemes, e.g., linear random network coding (RLNC) in which operations over large Galois field are required [18], IDNC can be implemented using simple binary XOR to encode and decode the mixed packets [19]. Besides, no buffers are needed at the receivers since each non-instantly decodable packet is discarded. These low complexity decoding and no buffer properties allow the design of simple and power efficient receivers.

IDNC can be divided into two schemes named Strict (S-IDNC) and Generalized IDNC (G-IDNC). The S-IDNC scheme, proposed in [20], restrict the sender to generate only packets that are instantly decodable by all receivers. The constraint, even though it makes the problem more mathematically tractable, limits the number of receivers targeted simultaneously. To overcome this limitation, G-IDNC is introduced in [21]. This scheme allows the sender to generate any combination of packets but forces the receivers to discard every not instantly decodable packet. G-IDNC was shown to outperform S-IDNC and to achieve a better decoding delay [21].

### B. Related Works

Earlier works on G-IDNC considered ideal problems in which an accurate and prompt feedback from all receivers is available at the sender. Moreover, the forward channel is deemed to be a memoryless erasure channel (MEC). These

assumptions are too idealistic given the severe and highly correlated impairments on both forward and feedback channels of wireless networks, due to shadowing, high interference, and fading. In addition, many wireless networks standards employ a time division duplex (TDD) structure for downlink and uplink transmissions. In such systems, the sender (usually a base station or an access point) receives feedback only during the uplink frame. Consequently, it needs to transmit several subsequent packets without having any information about their reception status at the different receivers. Due to such uncertainties about prior packet reception, the sender is no longer be sure on whether the selected combination instantly provide a new packet for the indented receivers. Therefore, he is not be confident about the resulting decoding delays.

Some recent works started to address these concerns in IDNC. In the perfect feedback scenario, reference [22] study the decoding delay problem in persistent erasure channels (PECs). The authors in [23] consider the throughput minimization problem for intermittent and lossy feedback scenarios. Inspired by the system properties, the authors solve the problem using blind approaches. In the context of video streaming, Li et al. [5], [24] optimize the throughput for lossy intermittent feedback scenario. The authors show that, for a sufficiently large video file and users experiencing hard deadline constraints, their proposed solution asymptotically achieves throughput optimality. Sorour et al. [25] address the problem of decoding delay minimization in G-IDNC with MECs for both lossy and intermittent feedback scenarios.

### C. Contribution

The main contribution of this paper is to extend the study [1] of the decoding delay reduction in limited feedback situation when both the forward and backward channels are considered to be PECs. Moreover, the feedback loss scenario discussed in this paper is the most general one that combines the scenarios studied separately in [25]. Furthermore, whereas earlier works assumed that receivers are requiring the same set packets, this paper considers the multicast session in which the demand of receivers may differ. Finally, in contrast with [23], analytical results are provided.

Finding a schedule of packet combinations that optimally minimize the decoding delay for the whole recovery phase, prior to its start, is intractable even for only three receivers and erasure free scenario [26] or the off-line case (i.e., case in which erasure events' timings are known in advance) [27]. A commonly adopted strategy [1], [5], [21], [23]–[25] is to reduce the decoding delay on-line, i.e., at each recovery transmission.

To reduce the decoding delay on-line, the probability distributions of the decoding delay increments in the different levels of feedback uncertainty are derived. Afterward, the decoding delay reduction problem is formulated as a maximum weight clique problem in the G-IDNC graph. Earlier obtained results for memoryless erasure channels and/or prompt feedback are shown to be special cases of our proposed generalized formulation. Since finding the maximum weight clique is NP-hard

[28], [29], a low complexity greedy algorithm is designed to solve the problem. The performance of our adaptive algorithm is compared to both the perfect feedback scenario and the blind approaches proposed in [23].

The rest of the paper is organized as follows: Section II introduces the system model. In Section III, the channel and feedback models are presented. In Section IV, the probability distributions of decoding delay increments are derived, and the problem formulated. Section V gives the problem formulations for simpler channel and feedback environments as special cases of our proposed generalized formulation. In Section VI, a heuristic algorithm is designed. Simulation results are presented in Section VII before concluding the paper in Section VIII.

## II. SYSTEM MODEL AND PARAMETERS

The model considered in this paper consists of a wireless sender that is required to deliver different (but possibly overlapping) portions of a frame (denoted by  $\mathcal{N}$ ) of  $N$  source packets to a set (denoted by  $\mathcal{M}$ ) of  $M$  receivers. Each receiver is interested in receiving a subset  $\mathcal{W}_i$  of packets from the frame (i.e.,  $\mathcal{W}_i \subseteq \mathcal{N}$ ,  $\forall i \in \mathcal{M}$ ). The scenarios in which  $\mathcal{W}_i = \mathcal{N}$ ,  $\forall i \in \mathcal{M}$  is known as a broadcast session. Packets requested by receiver  $i$  (packets in  $\mathcal{W}_i$ ) are called its primary (or wanted) packets and those not requested (packets in  $\mathcal{N} \setminus \mathcal{W}_i$ ) are called its secondary (or unwanted) packets.

Initially, the sender transmits the  $N$  packets of the frame uncoded to all the receivers. Each receiver listens to the transmitted packets and feeds back to the sender an acknowledgment for each successfully received packet. Both data and feedback packets are subject to erasures. Since receivers send feedback only when they successfully receive a packet, then an unheard feedback at the sender makes it uncertain on whether the packet or the feedback were erased. At the end of this *initial phase*, four sets of packets are attributed to each receiver  $i$ :

- The Has set (denoted by  $\mathcal{H}_i$ ): set of packets successfully received and acknowledged by receiver  $i$ .
- The Lack set (denoted by  $\mathcal{L}_i$ ): set of packets that are not in the Has set. In other words,  $\mathcal{L}_i = \mathcal{N} \setminus \mathcal{H}_i$ .
- The Wants set (denoted by  $\mathcal{W}_i$ ): set of primary packets in the Lack set. We have  $\mathcal{W}_i \subseteq \mathcal{L}_i$ .
- The Uncertain set (denoted by  $\mathcal{X}_i$ ): set of packets whose state is uncertain. We have  $\mathcal{X}_i \subseteq \mathcal{L}_i$ .

**Remark 1.** *The variables defined above should be all a function of the transmission time  $t$  (i.e., we should write  $\mathcal{H}_i(t)$ ). However, for notation convenience, the time index will be dropped unless it is required.*

The sender stores the information in a *sender feedback matrix* (SFM)  $\mathbf{F} = [f_{ij}]$ ,  $\forall i \in \mathcal{M}$ ,  $\forall j \in \mathcal{N}$  such that:

$$f_{ij} = \begin{cases} 0 & \text{if } j \in \mathcal{H}_i \\ -1 & \text{if } j \in \mathcal{L}_i \setminus \mathcal{W}_i \\ 1 & \text{if } j \in \mathcal{W}_i \setminus \mathcal{X}_i \\ x & \text{if } j \in \mathcal{W}_i \cap \mathcal{X}_i. \end{cases} \quad (1)$$

After the *initial phase*, a recovery one begins at time  $t = 1$ . In this stage, the sender transmits XOR combination of the

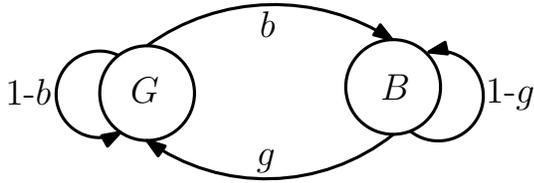


Fig. 2. The two state Gilbert-Elliott channel. The Good state ( $G$ ) results in a successful transmission whereas the Bad state ( $B$ ) results in an erasure of the transmitted packet.

source packets, using information available in the SFM and the expected erasure patterns of the links. After each transmission, each receiver that received and successfully decoded a packet acknowledges the reception of all previously received packets. This process is repeated until all receivers obtain their requested packets. The targeted receivers by a transmission are those that can instantly decode a packet from it.

In the recovery phase, each transmitted coded packet satisfies one of the following three options for each receiver  $i$ :

- *Non-innovative*: A packet is non-innovative for receiver  $i$  if all the source packets it encodes were successfully received previously, or they are secondary packets.
- *Instantly Decodable*: A packet is instantly decodable for receiver  $i$  if it contains *only one source packet* from  $\mathcal{L}_i$  that was not received previously.
- *Non-Instantly Decodable*: A packet is non-instantly decodable for receiver  $i$  if it contains two or more source packets from  $\mathcal{L}_i$ .

The decoding delay [21], [22] is defined as follows:

**Definition 1.** *At any recovery transmission, a user with non-empty Wants set, experiences one unit of decoding delay increase if it receives a packet that is either non-instantly decodable or both instantly decodable and non-innovative.*

### III. FORWARD AND FEEDBACK CHANNELS MODELS

#### A. Forward Channel Model and Parameters

The persistent erasure channel is modelled by the well-known Gilbert-Elliott channel (GEC) [30], [31]. GEC is a time-varying channel, whose crossover probabilities are determined by the current state of a discrete-time stationary binary Markov process (see Figure 2). The good state  $G$  results in a successful transmission whereas the bad state  $B$  results in an erasure. Due to the underlying Markov nature of the channel, it has memory that depends on the transition probabilities between the states, which can be defined for receiver  $i$  as follows:

$$\begin{aligned} \mathbb{P}(C_i(t) = G | C_i(t-1) = B) &= g_i^f \\ \mathbb{P}(C_i(t) = B | C_i(t-1) = G) &= b_i^f, \end{aligned} \quad (2)$$

where  $C_i(t)$  is the state of the channel for receiver  $i$  at time  $t$  and the superscript  $f$  refers to the forward channel. From a practical concerns [31], the system is more likely to stay in the same state than to switch between states. Thus, the values of  $g_i^f$  and  $b_i^f$ ,  $\forall i \in \mathcal{M}$  are, in general, both less than or equal to 0.5.

Let the memory factor of the channel be defined as  $\mu_i \triangleq 1 - g_i^f - b_i^f$  and so  $0 \leq \mu_i \leq 1$ ,  $\forall i \in \mathcal{M}$ . A high value of

$\mu_i > 0.5$  means that the channel is highly correlated (highly persisting channel), and thus the channel is likely to stay in the same state during the following transmission. In the particular case  $\mu_i = 0$  (i.e.,  $g_i^f = 1 - b_i^f$ ), the state of the channel changes in an independent manner and thus it represents a memoryless channel. Consequently, the proposed channel model is a more general model of which the MEC is a special case.

The receivers' channels are assumed to be independent, and the sender is assumed to have perfect knowledge of the state transition probabilities of the channels of receivers and the initial state for each receiver.

#### B. Feedback Channel Model and Parameters

Let the time be divided into frames of length  $T$  time-slots. Each frame is composed of a downlink sub-frame of length  $T_d$  and an uplink sub-frame of length  $T_u$  (i.e.,  $T = T_d + T_u$ ). Let  $n^+(t) = \lceil t/T \rceil$  be the index of the frame at time  $t$ , where  $\lceil \cdot \rceil$  is the ceiling function. In the considered intermittent lossy feedback scenario, the sender transmits packets during the downlink sub-frame and receives feedback during the uplink sub-frame. Scheduled receivers in the downlink sub-frame that successfully decoded a packet send acknowledgement of all their received packets in the following uplink sub-frame.

Let  $T_{u_i}$  be the time-slot in the uplink sub-frame in which receiver  $i$  sends its feedback (i.e.,  $1 \leq T_{u_i} \leq T_u$ ,  $\forall i \in \mathcal{M}$ ). In other words, receiver  $i$  can transmit, during frame number  $n$ , at time  $t = nT - T_u + T_{u_i} = u_i(n)$ .

Following a model similar to the forward channel, the feedback channel is subject to persistent loss, modelled by a GEC where  $g_i^b$  and  $b_i^b$  are the transition probabilities for each receiver  $i$ . Superscript  $b$  refers to the feedback channel. The memory factor of the feedback channel, for receiver  $i$ , is  $\psi_i \triangleq 1 - g_i^b - b_i^b$ ,  $\forall i \in \mathcal{M}$ .

Figure 3 shows a potential transmission in a lossy intermittent feedback environment with all the system variables. Note that since  $T_u = 1$ , then  $T_{u_i} = 1$ ,  $i = 1, 2, 3$ . At time slot  $t = 4$ , the sender do not transmit and listen to the feedback from receivers. Receiver 2 has empty Wants set at transmission  $t = 5$  and thus is not monitored any more and its state not estimated.

In order to accurately estimate the state of the channel at the sender, for each targeted receiver, needing more than one packet, there is a packet that is attempted only once from the last time a feedback is heard from that receiver. Since only targeted receivers send feedback, the sender will not get any feedback from a receiver until the next transmission in which it is targeted.

### IV. DECODING DELAY REDUCTION FORMULATION

Finding a schedule of packet combinations that optimally minimize the decoding delay for the whole recovery phase, prior to its start, is intractable. A commonly adopted strategy [1], [5], [21], [23]–[25] is to reduce the decoding delay on-line, i.e., at each recovery transmission.

Let  $\mathcal{D}(\kappa, t)$  be the sum of the on-line decoding delay increases of all receivers after the transmission  $\kappa$ . Unlike the scenario of [1], in which the on-line decoding delay is a

$$M = 3, N = 4, L = 1, T_d = 3, T_u = 1$$

Time ( $t$ )	1	2	3	4	5	...
Frame Number	1				2	...
Frame Type	D			U	D	...
Receiver 1 channel	<b>B</b>	<b>G</b>	<b>G</b>	<b>G</b>	<b>G</b>	...
Receiver 2 channel	<b>G</b>	<b>B</b>	<b>G</b>	<b>G</b>	<b>X</b>	...
Receiver 3 channel	<b>G</b>	<b>G</b>	<b>B</b>	<b>B</b>	<b>G</b>	...
Packet Combination	4	$1 \oplus 2$	$1 \oplus 3$	<b>X</b>	4	...

$$\mathbf{F}(0) = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{pmatrix}, \quad \mathbf{F}(1) = \begin{pmatrix} 1 & 0 & 0 & x \\ 0 & 0 & 1 & x \\ 0 & 1 & 1 & x \end{pmatrix}$$

$$\mathbf{F}(2) = \begin{pmatrix} x & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & x & 1 & 1 \end{pmatrix}, \quad \mathbf{F}(3) = \begin{pmatrix} x & 0 & 0 & x \\ 0 & 0 & x & x \\ 0 & x & x & x \end{pmatrix}$$

$$\mathbf{F}(4) = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & x & x & x \end{pmatrix}, \quad \mathbf{F}(5) = \begin{pmatrix} 0 & 0 & 0 & x \\ 0 & 0 & 0 & 0 \\ 0 & x & x & x \end{pmatrix}$$

Fig. 3. Illustration of a potential transmission in lossy intermittent feedback scenario. The frame type D refers to the downlink subframe and the frame type U to the uplink subframe. The channel designation **G** and **B** refers to the Good and Bad state respectively. Symbol **X** is used designation is not applicable

function only of the packet combination, in persistent erasure channel, it also depends on the transmission time slot. This time-dependency makes the algorithm proposed in [1] non-effective to reduce the decoding delay in persistent erasure channels. Following the simplification adopted in Remark 1, the decoding delay may be written as  $\mathcal{D}(\kappa)$ . The on-line decoding delay reduction problem in lossy intermittent feedback scenario can be written as:

$$\min_{\kappa \in \mathcal{P}(\mathcal{N})} (\mathbb{E}[\mathcal{D}(\kappa)]), \quad (3)$$

where the notation  $\mathcal{P}(\mathcal{X})$  refers to the power set of the set  $\mathcal{X}$ .

To optimally solve the optimization problem (3), several probability distributions are derived. Afterwards, the expected decoding delay increase for any arbitrary transmission is expressed. These expressions allow the reformulation of the problem as a maximum weight clique in the G-IDNC graph that can be globally solved using efficient algorithms [28], [32].

#### A. Transmission/Feedback Loss Probabilities at Time $t$

In this subsection, the probability to lose the transmission  $p_i(t)$  and to lose the feedback  $q_i(t)$  for receiver  $i$  at time slot  $t$  are computed. Define  $j_i^\dagger$  as the last sent packet among those which were attempted only once between the most recent two frames in which a feedback is heard from user  $i$ . Let  $t_i^\dagger$  be the time in which the packet is attempted. Let  $t_i^*$  be the time when the latest feedback is heard from receiver  $i$ .

Given these definitions, the following theorem introduces the loss probabilities of the forward and feedback transmissions at any given time  $t$ .

**Theorem 1.** *The probabilities  $p_i(t)$  and  $q_i(t)$  of losing a transmission at receiver  $i$  or a feedback from it at time  $t > t_i^*$*

can be, respectively, expressed as:

$$p_i(t) = \begin{cases} b_i^f \sum_{l=0}^{t-t_i^\dagger-1} \mu_i^l & \text{if } f_{i,j_i^\dagger}(t_i^*) = 0 \\ 1 - g_i^f \sum_{l=0}^{t-t_i^\dagger-1} \mu_i^l & \text{if } f_{i,j_i^\dagger}(t_i^*) = 1 \end{cases} \quad (4)$$

$$q_i(t) = b_i^b \sum_{l=0}^{t-t_i^*-1} \psi_i^l. \quad (5)$$

*Proof:* The state of the forward (backward) channel at time  $t_i^\dagger$  ( $t_i^*$ ) being known, a time regression from  $t$  to  $t_i^\dagger$  ( $t_i^*$ ) using the properties of the Markov chain yield the desired results. A complete proof can be found in Appendix A. ■

Following the simplification adopted in Remark 1, the loss probabilities of the forward and feedback transmissions may be written as  $p_i$  and  $q_i$ . In that case, they refer to the expectations at the transmission slot under investigation.

#### B. Decoding Delay Increment Probabilities

Let  $\mathfrak{J}$  be a combination of packets from  $\mathcal{N}$  and let  $d_i(\mathfrak{J})$  be the decoding delay increase experienced by receiver  $i$  after the transmission of  $\mathfrak{J}$ . For each receiver  $i$ , its Wants set is assumed to be only one of the following three options:

- 1) Non-Uncertain Wants set (i.e.,  $\mathcal{W}_i \cap \mathcal{X}_i = \emptyset$ ).
- 2) Partially Uncertain Wants set (i.e.,  $\mathcal{W}_i \cap \mathcal{X}_i \neq \emptyset$  and  $\mathcal{W}_i \not\subset \mathcal{X}_i$ ). Let  $U$  be the set of targeted receivers for which intended packet state in the transmission  $\mathfrak{J}$  is unknown (i.e.,  $f_{i,\mathfrak{J}_i} = x, \forall i \in U$ ).
- 3) Fully Uncertain Wants set (i.e.,  $\mathcal{W}_i \subset \mathcal{X}_i$ ). Let  $F$  be the set of receivers having entirely uncertain Wants sets.

From the definition of decoding delay in Section II, receiver  $i$  with non-empty Wants set will not experience a decoding delay (i.e.,  $d_i(\mathfrak{J}) = 0$ ) if and only if  $\mathfrak{J}_i$  is both instantly decodable and contains a single source packet from  $\mathcal{W}_i$  (packet is innovative for receiver  $i$ ). Since receivers with empty Wants set will never experience a decoding delay, we will only consider, in the rest of the section, receivers with non-empty Wants set. Let  $M_w$  be the set of these receivers. The decoding delay increase depends not only on the sent packet  $\mathfrak{J}$  but also on the channel states of the targeted receivers. We define  $\tau$  as the set of targeted receivers by the transmission  $\mathfrak{J}$  where the intended packet  $\mathfrak{J}_i$  is a primary packet for receiver  $i, \forall i \in \tau$  ( $\mathfrak{J}_i = \emptyset$  if receiver  $i$  is not targeted by a primary packet). Let  $\hat{\tau} = M_w \setminus \tau$  be the set of non-targeted receivers and those targeted by a secondary packet. The following theorem gives the expected decoding delay for a receiver  $i$ :

**Theorem 2.** *The probability that receiver  $i$  experiences a decoding delay, after the transmission  $\mathfrak{J}$  is:*

$$\mathbb{P}(d_i(\mathfrak{J}) = 1) = \begin{cases} 1 - p_i & i \in (\hat{\tau} \cap \bar{F}) \\ (1 - p_i)(1 - p_{i,f}) & i \in (\hat{\tau} \cap F) \\ 0 & i \in (\tau \cap \bar{U}) \\ (1 - p_i)(1 - p_{i,n}(\mathfrak{J}_i)) & i \in (\tau \cap (U \setminus F)) \\ (1 - p_i)(1 - p_{i,n}(\mathfrak{J}_i) - p_{i,f}) & i \in (\tau \cap F), \end{cases} \quad (6)$$

where  $p_{i,n}(j)$  (referred to as innovative probability) is the probability that packet  $j$  is innovative for receiver  $i$  and  $p_{i,f}$  (referred to as finish probability) is the probability that receiver  $i$  successfully received all its primary packets but  $\mathcal{W}_i \neq \emptyset$  at the sender due to intermittence and loss of feedback.

*Proof:* To prove this theorem, the different events that can occur at the sender are identified, and their probabilities computed. Afterwards, the decoding delay is expressed as a function of these events for each category of receivers, e.g., with or without a fully uncertain wants set. A complete proof can be found in Appendix B. ■

Naturally, the innovative and the finish probability depend on the time instant of the frame. For simplification purposes as in Remark 1, the time index will be used only when necessary. Otherwise, it is implicitly understood that it is the time of the transmission under investigation. In order to derive the expressions for the innovative and finish probabilities, we introduce the following variables:

- $\mathcal{K}_{ij}$ : set of indices of frames in which packet  $j$  was attempted to receiver  $i$  since the last frame a feedback is received, excluding the current one.
- $\lambda_{ij}(n)$ : set of the time indices in which packet  $j$  was attempted to receiver  $i$  during frame number  $n$ .
- $\mathcal{X}_i^d(n)$ : set of times where a packet (any packet) was attempted to receiver  $i$  during frame number  $n$ .

Given these definitions, we can introduce the following theorem and corollary, illustrating the expressions for the innovative and finish probabilities.

**Theorem 3.** *The probability that packet  $j$  is innovative for receiver  $i$ , at time  $t$  is:*

$$p_{i,n}(j, t) = \prod_{k \in \lambda_{ij}(n+(t))} p_i(k) \prod_{k \in \mathcal{K}_{ij}} \left\{ \left( \prod_{s \in \mathcal{X}_i^d(k)} p_i(s) + \prod_{s \in \lambda_{ij}(k)} p_i(s) \left[ 1 - \prod_{s \in \mathcal{X}_i^d(k) \setminus \lambda_{ij}(k)} p_i(s) q_i(u_i(k)) \right] \right) \right. \\ \left. \times \left( \prod_{s \in \mathcal{X}_i^d(k)} p_i(s) + \left[ 1 - \prod_{s \in \mathcal{X}_i^d(k)} p_i(s) q_i(u_i(k)) \right]^{-1} \right) \right\}, \quad (7)$$

*Proof:* To show this theorem, all the states of a given packet are identified, and the probabilities computed. Such expressions allow to express the innovative probability at any time instant using the innovative probability at the beginning of the downlink sub-frame. Finally, the innovative probability at the start of each frame is computed, and the final expression derived. A complete proof can be found in Appendix C. ■

**Corollary 1.** *The probability that receiver  $i$  successfully received all his primary packets but  $\mathcal{W}_i \neq \emptyset$  is:*

$$p_{i,f} = \prod_{j \in \mathcal{W}_i} (1 - p_{i,n}(j)), \quad (8)$$

*Proof:* To express the finish probability, the different events are identified, and their expectation expressed using

the innovative likelihood. A complete proof can be found in Appendix D. ■

### C. Problem Reformulation

In order to reformulate the on-line decoding delay reduction problem (3), we look for all possible packets combinations that are instantly decodable for a subset (possibly all) of receivers, and choose the one that has minimum expected increase in the decoding delay. A convenient way to represent all these XOR packet combination is the G-IDNC graph introduced in [21]. Let  $\mathcal{G}(\mathcal{V}, \mathcal{E})$  be the G-IDNC graph and let  $v_{ij}$  be the vertices such that  $\mathcal{V} = \{v_{ij}, \forall j \in \mathcal{L}_i \text{ and } \forall i \in \mathcal{M}\}$ . In other words, a vertex  $v_{ij}$  is generated for every non-zero entry  $f_{ij}$  in the SFM.

Vertices in this graph are connected to each other only if the resulting packet combination is instantly decodable of both receivers represented by the vertices. In other words,  $v_{ij}$  and  $v_{kl}$  are connected if one of these scenarios occur:

- $j = l$ : The receivers  $i$  and  $k$  are requesting the same packet  $j$ .
- $j \in \mathcal{H}_k$  and  $l \in \mathcal{H}_i$ : The needed packet of each vertex is in the Has set of the receiver represented by the other vertex.

The following theorem reformulates the on-line decoding delay reduction problem (3) as a maximum weight clique problem.

**Theorem 4.** *The on-line decoding delay reduction problem (3) for G-IDNC in lossy intermittent feedback scenario is equivalent to a maximum weight clique in the G-IDNC graph in which the weight of each vertex  $v_{ij}$  can be expressed as follows:*

$$w_{ij}^0 = (1 - p_i)p_{i,n}(j). \quad (9)$$

*Given the NP-hardness of finding the maximum weight clique, the optimization problem (3) is an NP-hard problem.*

*Proof:* To prove this theorem, the optimal packet combination to reduce the decoding delay is expressed using the previously defined innovative and finish probabilities. Further, the mixed integer program is shown to be equivalent to a maximum weight clique in the G-IDNC graph. A complete proof can be found in Appendix E. ■

## V. SPECIAL CASES

In this section, our proposed generalized formulation is used to extract the distributions and problem formulations of seven feedback loss scenarios. These special cases includes configurations studied in [1], [21], [22], [25]. Hence, our formulation is shown to be a general framework that can be used to study multiple feedback loss scenarios. For simplicity, since the problem formulation relies on the expression of the innovative probability, in this section we only provide its expression. By substitution, the on-line decoding delay problem can be formulated. For each special case, the problem formulation is in agreement with the expression derived in the cited references and thus the solution is the same.

### A. Forward and Feedback PECs without Feedback Intermittence

In the forward and feedback PECs without feedback intermittence case, the length of the frame  $T = 2$ . After each sent packet a feedback can be heard upon successful reception (i.e.,  $T_d = T_u = T_{u_i} = 1$ ). The following lemma gives the simplified expression of the packet erasure and innovative probability in lossy feedback without intermittence.

**Lemma 1.** *The packet erasure probability in the lossy feedback scenario is:*

$$p_i = b_i^f \sum_{l=0}^{t-t_i^{(0)}-1} \mu_i^l. \quad (10)$$

The likelihood that packet  $j$  is innovative to receiver  $i$  at time  $t$  in the lossy feedback situation is:

$$p_{i,n}(j, t) = \prod_{k \in \mathcal{K}_{ij}} \frac{p_i(2k-1)}{p_i(2k-1) + (1-p_i(2k-1))q_i(2k)}. \quad (11)$$

*Proof:* The proof can be found in Appendix F. ■

### B. Forward Only PECs with Feedback Intermittence

In the forward only PECs with intermittent feedback, the reception of the feedback occurs in the uplink sub-frame and is considered perfect. In other words  $q_i = 0, \forall i \in \mathcal{M}$ . By substituting this feedback erasure probability in (8), the probability that packet  $j$  is innovative to receiver  $i$  at time  $t$  becomes:

$$p_{i,n}(j, t) = \prod_{k \in \lambda_{ij}(n^+(t))} p_i(k). \quad (12)$$

### C. Forward Only PECs with Perfect Feedback

In forward only PECs with perfect feedback, the length of the frame  $T = 2$  and reception of feedback is considered perfect. In other words, only the data packets are subjected to loss. This case can be derived by setting  $q_i = 0, \forall i \in \mathcal{M}$  in (11). Thus, the probability of a packet to be innovative is 1. The problem can be formulated as a maximum weight clique problem as follows:

$$\kappa^* = \arg \max_{\kappa \in \mathcal{G}} \sum_{i \in \tau(\kappa)} (1 - p_i), \quad (13)$$

in agreement with the expression derived in [22].

### D. Forward and Feedback MECs with Feedback Intermittence

In a memoryless channel, channel state changes in an entirely uncorrelated manner. This can be obtained in our model by setting the channel memory  $\mu_i = 0, \forall i \in \mathcal{M}$ . Since  $\mu = 1 - g_i^f - b_i^f = 0$  then we have  $g_i^f + b_i^f = 1$ . The transition probabilities become independent of time:

$$\begin{aligned} \mathbb{P}(C_i(t) = G | C_i(t-1) = B) &= g_i^f \\ \mathbb{P}(C_i(t) = B | C_i(t-1) = G) &= b_i^f \end{aligned} \quad (14)$$

As a result, the packet and feedback erasure probability becomes independent of the time instant as illustrated in the

following equations:

$$p_i = b_i^f \quad (15)$$

$$q_i = b_i^b. \quad (16)$$

In the forward and feedback MECs with feedback intermittence case, the likelihood that packet  $j$  is innovative for receiver  $i$  in the lossy intermittent feedback, at time  $t$  becomes:

$$p_{i,n}(j) = p_i^{|\lambda_{ij}(n^+(t))|} \times \prod_{k \in \mathcal{K}_{ij}} \frac{\left( p_i^{|\mathcal{X}_i^d(k)|} + p_i^{|\lambda_{ij}(k)|} \times (1 - p_i^{|\mathcal{X}_i^d(k)| - |\lambda_{ij}(k)|}) q_i \right)}{\left( p_i^{|\mathcal{X}_i^d(k)|} + (1 - p_i^{|\mathcal{X}_i^d(k)|}) q_i \right)}, \quad (17)$$

which is the same expression derived in [1].

### E. Forward and Feedback MECs without Feedback Intermittence

In the forward and feedback MECs without feedback intermittence case, the length of the frame  $T = 2$ . In MECs, the transmission and feedback loss probabilities are time-independent. This particular case can be obtained by substitution  $p_i(t) = p_i$  and  $q_i(t) = q_i$  in (11). As a result, the probability that packet  $j$  is innovative to receiver  $i$  at time  $t$  in the lossy feedback without intermittence scenario becomes:

$$p_{i,n}(j) = \left( \frac{p_i}{p_i + (1-p_i)q_i} \right)^{|\mathcal{K}_{ij}|}, \quad (18)$$

in agreement with the expression derived in [25].

### F. Forward Only MECs with Feedback Intermittence

The forward only MECs with feedback intermittence case can be obtained by setting  $q_i(t) = 0$  and substitution  $p_i(t) = p_i$  in (12). Thus, the probability that packet  $j$  is innovative to receiver  $i$  at time  $t$  in the intermittent feedback scenario becomes:

$$p_{i,n}(j) = p_i^{|\lambda_{ij}|}, \quad (19)$$

in agreement with the expression derived in [25].

### G. Forward Only MECs with Perfect Feedback

In the forward only MECs with perfect feedback, the length of the frame  $T = 2$  and reception of the feedback is considered perfect. As for PECs, the innovative probability, in forward only MECs with accurate feedback, is equal to 1. The on-line decoding delay reduction problem, in this particular case, can be obtained by substituting  $p_i(t) = p_i$  in (13) as follows:

$$\kappa^* = \arg \max_{\kappa \in \mathcal{G}} \sum_{i \in \tau(\kappa)} (1 - p_i), \quad (20)$$

in agreement with the expression derived in [21].

## VI. PROPOSED GREEDY ALGORITHM AND PARTIALLY BLIND APPROACHES

The maximum weight clique problem is well known to be NP-hard [28], [29], [33]. To solve problem (3) with low complexity, a simple heuristic algorithm is introduced in this section. The complexity of the proposed solution is  $O(M^2N)$  (i.e., quadratic time in the number of receivers and linear

time in the number of packets). This algorithm follows the same concept as the one proposed in [1], but the way the new weights are computed is different. Furthermore, blind graph update policies, introduced in [23] to heuristically solve the decoding delay reduction in lossy feedback scenario, are extended to the lossy intermittent feedback situation under investigation. These partially blind approaches update the graph by removing uncertain vertices with a predefined policy then perform packet selection using the algorithm proposed in [21].

#### A. Proposed Greedy Algorithm

Define  $A = [a_{ij,kl}]$  as the adjacency matrix associated with the G-IDNC graph  $\mathcal{G}(\mathcal{V}, \mathcal{E})$  defined as follows:

$$a_{ij,kl} = \begin{cases} 1 & v_{ij} \text{ is connected to } v_{kl} \text{ in } \mathcal{G} \\ 0 & \text{otherwise.} \end{cases} \quad (21)$$

Define  $w_{ij}$  as the modified weights of vertex  $v_{ij}$ . The modified weight incorporates the connectivity of  $v_{ij}$  to vertices having high reception and innovative probabilities as follows:

$$w_{ij} = \sum_{v_{kl} \in \mathcal{G}} a_{ij,kl} (1 - p_k) p_{k,n}(l) \frac{\delta(v_{kl})}{|\mathcal{E}|}, \quad (22)$$

where  $\delta(v_{ij}) = \sum_{v_{kl} \in \mathcal{G}} a_{ij,kl}$  is the degree of vertex  $v_{ij}$  (i.e., the number of vertices adjacent to  $v_{ij}$ ), and  $|\mathcal{E}|$  is total number of edges.

In order to well define the weight even if the vertex is not adjacent to any other vertices (i.e.,  $w_{ij} = 0$ ), we set the new weights  $w_{ij}^*$  as follows:

$$w_{ij}^* = (w_{ij} + 1)w_{ij}^0, \quad \forall v_{ij} \in \mathcal{G}. \quad (23)$$

Consequently, a vertex has a significant weight, when:

- It has an important initial weight (i.e., higher probability of reception and less uncertainty).
- It is adjacent to a larger number of vertices with large initial weights.

To solve the maximum weight clique problem in  $O(M^2N)$ , we propose the following method. The primary graph  $\mathcal{G}_p$  (i.e., G-IDNC graph containing only the wanted packets) is generated. The vertex with the highest modified weight is selected. The graph is then updated to remove all vertices not connected to the selected vertex in the previous step. The process is repeated until the graph becomes empty. At this point, the packet combination  $\kappa_p$  contain only the primary packet. Since the inclusion of the secondary packets should not disturb the instant decodability of the primary packets, they are added after selecting the main ones. Hence after selection of  $\kappa_p$ , the secondary graph  $\mathcal{G}_s$  (i.e., G-IDNC graph containing only the unwanted packets) is generated. The graph is updated to keep only the vertices that are connected to all vertices in  $\kappa_p$ . The same steps are applied to output the packet combination  $\kappa_s$  containing the unwanted packets combinable with  $\kappa_p$ . The final served clique is  $\kappa^* = \kappa_p \cup \kappa_s$ . Let  $\mathfrak{R}(\mathcal{G}, v_{ij})$  be the sub-graph in  $\mathcal{G}$  containing only the vertices connected to  $v_{ij}$ . The whole algorithm structure is illustrated in Algorithm 1.

---

#### Algorithm 1 Maximum Weight Vertex Search Algorithm

---

**Require:**  $\mathbf{F}$  and  $p_i, p_{i,n}(j), p_{i,f} \quad \forall i \in \mathcal{M} \quad \forall j \in \mathcal{N}$   
Initialize primary and secondary cliques  $\kappa_p, \kappa_s = \emptyset$ .  
Construct primary graph  $\mathcal{G}_p(\mathcal{V}_p, \mathcal{E}_p)$ .  
**while**  $\mathcal{G}_p \neq \emptyset$  **do**  
    Compute weights  $w_{ij}^*$  using (9),(22) and (23).  
    Select  $v_p = \arg \max_{v_{ij} \in \mathcal{G}_p} \{w_{ij}^*\}$ .  
    sets  $\kappa_p \leftarrow \kappa_p \cup v_p$ .  
    sets  $\mathcal{G}_p \leftarrow \mathfrak{R}(\mathcal{G}_p, v_p)$ .  
**end while**  
Construct secondary graph  $\mathcal{G}_s(\mathcal{V}_s, \mathcal{E}_s)$ .  
**for all**  $v_p \in \kappa_p$  **do**  
    sets  $\mathcal{G}_s \leftarrow \mathfrak{R}(\mathcal{G}_s, v_p)$   
**end for**  
**while**  $\mathcal{G}_s \neq \emptyset$  **do**  
    Compute weights  $w_{ij}^*$  using (9),(22) and (23).  
    Select  $v_s = \arg \max_{v_{ij} \in \mathcal{G}_s} \{w_{ij}^*\}$ .  
    sets  $\kappa_s \leftarrow \kappa_s \cup v_s$ .  
    sets  $\mathcal{G}_s \leftarrow \mathfrak{R}(\mathcal{G}_s, v_s)$ .  
**end while**  
Output  $\kappa^* \leftarrow \kappa_p \cup \kappa_s$ .

---

#### B. Blind Graph Update Policies

1) *No Vertex Elimination (NVE)*: In this approach, all uncertain packets are considered in subsequent transmissions. In a context of lossy feedback, this approach performs well for high erasure channels. However, in an intermittent feedback situation, no acknowledgment is expected during the downlink frame. Hence, the same packet combination is attempted during the whole downlink that lead to unsatisfactory performance. Therefore, this approach is not tested in this paper.

2) *Full Vertex Elimination (FVE)*: In this approach, all uncertain packets are considered received. Hence, their corresponding vertices are removed from the graph. Not reconsidering these packets in the following transmissions gives a greater chance to non-attempted packets to be transmitted. Unless targeted, no acknowledgment is expected from users with entirely uncertain Wants set. Therefore, such users are reconsidered after the uplink subframe.

3) *Stochastic Vertex Elimination (SVE)*: In this approach, all uncertain packets are probabilistically removed from the graph. This approach tends to balance stochastically between reattempting packets and transmitting new ones. In other words, this approaches trade-off the FVE and NVE approaches. Given the unsatisfactory performance of NVE in intermittent feedback scenarios, this approach is expected to perform worst than the FVE one. Therefore, this method is not tested in this paper.

## VII. SIMULATION RESULTS

In this section, the performance of our adaptive algorithm to efficiently reduce the decoding delay in lossy intermittent feedback situation are shown. Our proposed solution is compared against the full vertex elimination approach and the perfect feedback scenario.

In [1], the authors proposed a heuristic to solve the maximum weight clique problem based on the following weight

modification:

$$w_{ij} = \sum_{v_{kl} \in \mathcal{G}} a_{ij,kl} (1 - p_k) p_{k,n}(l). \quad (24)$$

Although the weight modification (24) gives priority to vertices with significant initial weight and degree, it does not consider the degree of adjacent vertices, unlike our proposed weight modification (22). Considering the connectivity index in the selection process allows the selection of cliques with larger degree and thus a better objective function that translates into a lower decoding delay. The solid lines represent the performance of our proposed policy with the proposed weight modification (22). The dashed ones represent the performance using the weight modification (24).

In these simulations, we assume channel reciprocity. In other words, the packet and the feedback erasure probabilities are the same (i.e.,  $g_i^f = g_i^b$  and  $b_i^f = b_i^b$ ,  $\forall i \in \mathcal{M}$ ). We also assume that the packet erasure probability  $b_i^f$ ,  $\forall i \in \mathcal{M}$  for all receivers change, from frame to frame, uniformly in the range  $[0.1 \ 0.3]$ . The average decoding delay is computed over a large number of iterations then the mean value is presented.

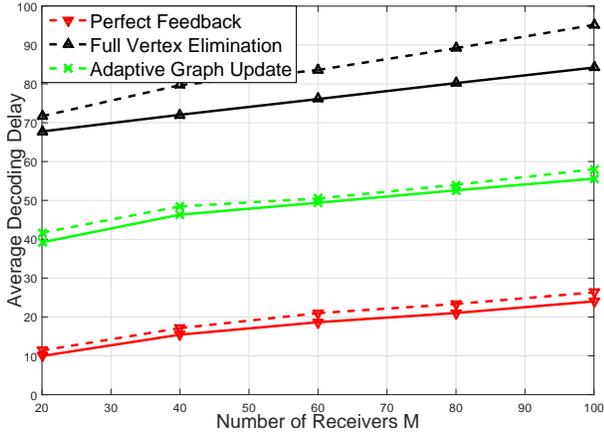


Fig. 4. Mean decoding delays against the number of receivers  $M$  for a network composed of  $N = 30$  packets, a channel memory  $\mu = 0.5$ , and a downlink of 10 time slots. The solid lines represent our proposed weight modification (22) whereas the dash lines represent the weight modification (24).

Figure 4 depicts the comparison of mean decoding delays achieved by the different algorithms against the number of receivers  $M$ . Figure 5 describe the performance versus the number of packets  $N$ . From both figures, we clearly see that our proposed adaptive scheme outperforms the FVE approach. This can be explained by the fact that when the channel memory is low the probability  $p_i \approx b_i^f \ll 1$  is also low but when the persistent becomes higher the probability to lose the transmission  $p_i > \sum_{l=0}^{t-t_i^f-1} \mu^l b_i^f \gg b_i^f$  becomes also higher. Thus removing all the uncertain vertexes is no longer an acceptable approach.

Figure 6 and Figure 7 illustrate the comparison against the channel memory  $\mu$  and the frame length  $T$ . Figure 6 shows that all the algorithms achieve a close decoding delay for a low persistent channel. This result can be explained by the fact that in a low persistent channel, the channel change in an uncorrelated way that makes channel estimation non-effective for all algorithms. For high time-correlation, the channel is

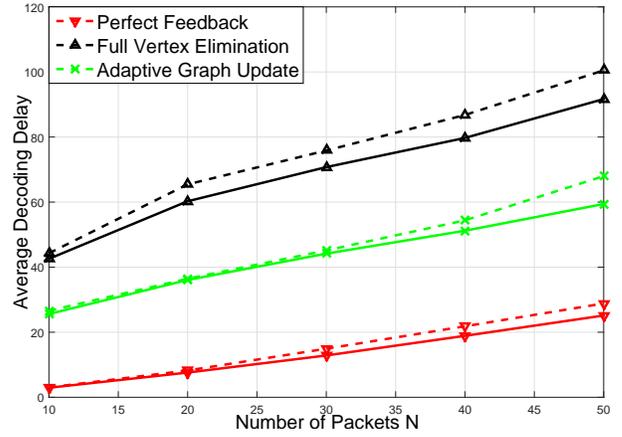


Fig. 5. Mean decoding delays against the number of packets  $N$  for a network composed of  $M = 30$  receivers, a channel memory  $\mu = 0.5$ , and a downlink of 10 time slots. The solid lines represent our proposed weight modification (22) whereas the dash lines represent the weight modification (24).

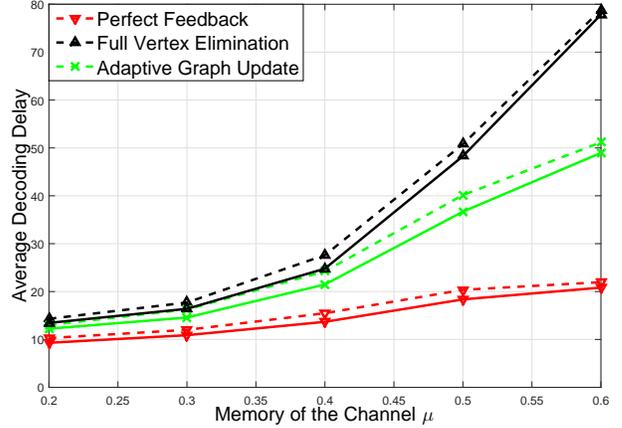


Fig. 6. Mean decoding delays against the average channel memory  $\mu$  for a network composed of  $M = 30$  receivers,  $N = 30$  packets, a demand  $L = 80\%$ , and a downlink of 5 time slots. The solid lines represent our proposed weight modification (22) whereas the dash lines represent the weight modification (24).

more likely to remain in a state than to toggle between states. Therefore, the estimate is more accurate which explain the difference in performance between algorithms. We clearly can see the gap between the algorithm when the persistence of the channel is higher than 0.4 while the probability to be in the bad state still the same. We also see that our adaptive algorithm achieve reasonable degradation in the high persistent channel. The same thinking is applicable for Figure 7. Whereas our algorithm produces a moderate degradation in a high persistent channel ( $\mu = 0.5$ ) and extensive feedback period ( $T = 10$ ), other solutions quickly degrade.

Figure 8 illustrates the performance versus the percentage of needed packet  $L$ . We clearly see that the decoding delay is high when the demand of packet is moderate. The result can be explained by the nature of the multicast scenario. In multicast, the receivers are listening to a lot of unwanted packets before completion that is reflected by a high decoding delay. When this percentage is high, the receivers are listening to fewer unwanted packets and thus the decoding delay is small. When the percentage is tiny (e.g., 20%), the receivers are demanding only a few packets resulting in a little decoding delay. Figure 8

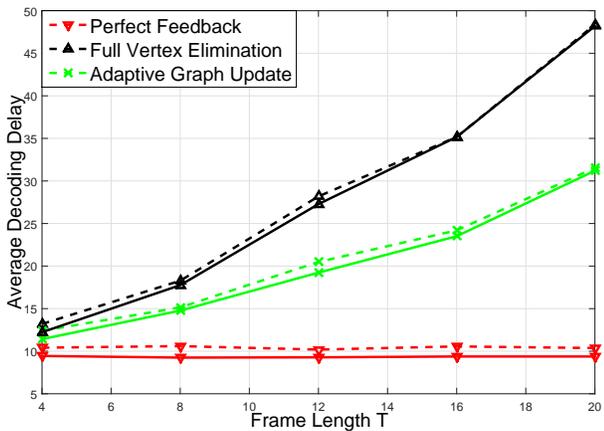


Fig. 7. Mean decoding delays against the downlink frame length  $T$  for a network composed of  $M = 30$  receivers,  $N = 30$  packets, a channel memory  $\mu = 0.2$ , and a demand  $L = 80\%$ . The solid lines represent our proposed weight modification (22) whereas the dash lines represents the weight modification (24).

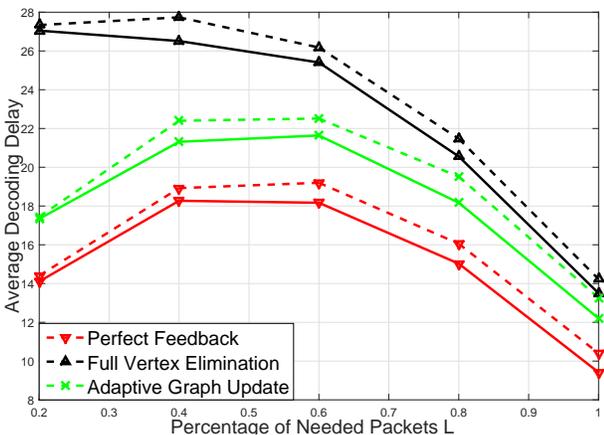


Fig. 8. Mean decoding delays against the percentage of needed packet  $L$  for a network composed of  $M = 30$  receivers,  $N = 30$  packets, a channel memory  $\mu = 0.2$ , and a downlink of 5 time slots. The solid lines represent our proposed weight modification (22) whereas the dash lines represent the weight modification (24).

shows that our algorithm perform a gain of 4% when the demand is less than 60%. For a need greater than 80%, the gain is higher than 9%.

From all the figures, our proposed weight modification algorithm (22) achieves a better decoding delay in all situations. The average gain from Figure 4, Figure 5, and Figure 6 is 9% comparing to the weight modification (24).

## VIII. CONCLUSION

This paper studies the multicast decoding delay reduction problem for generalized instantly decodable network coding over persistent erasure channels with feedback imperfections. The occurring events are identified, and their probabilities computed. The decoding delay increment expressions are derived and employed to express the decoding delay reduction problem as a maximum weight clique problem in the G-IDNC graph. In order to solve the decoding delay reduction problem with low complexity, a sub-optimal greedy algorithm is designed and compared to blind approaches proposed in the literature. Through extensive simulations, the proposed

algorithm is shown to outperform the blind methods in all situations and to achieve significant improvement, especially for high time-correlated channels.

## REFERENCES

- [1] A. Douik, S. Sorour, M.-S. Alouini, and T. Al-Naffouri, "Delay reduction in lossy intermittent feedback for generalized instantly decodable network coding," in *Proc. of IEEE 9th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob' 2013)*, Lyon, France, Oct 2013, pp. 388–393.
- [2] J. Sundararajan, D. Shah, and M. Medard, "Online network coding for optimal throughput and delay - the three-receiver case," in *Proc. of IEEE International Symposium on Information Theory and Its Applications (ISITA' 2008)*, Auckland, New Zealand, Dec 2008, pp. 1–6.
- [3] D. Nguyen, T. Nguyen, and X. Yang, "Multimedia wireless transmission with network coding," in *Proc. of the International Packet Video Workshop (PV' 2007)*, Lausanne, Switzerland, Nov 2007, pp. 326–335.
- [4] D. Nguyen and T. Nguyen, "Network coding-based wireless media transmission using pomdp," in *Proc. of the International Packet Video Workshop (PV' 2009)*, Seattle, WA, USA, May 2009, pp. 1–9.
- [5] X. Li, C.-C. Wang, and X. Lin, "On the capacity of immediately-decodable coding schemes for wireless stored-video broadcast with hard deadline constraints," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 5, pp. 1094–1105, May 2011.
- [6] L. Lu, M. Xiao, and L. K. Rasmussen, "Design and analysis of relay-aided broadcast using binary network codes," *Journal of Communications*, vol. 6, no. 8, pp. 610–617, 2011.
- [7] S. Rayanchu, S. Sen, J. Wu, S. Banerjee, and S. Sengupta, "Loss-aware network coding for unicast wireless sessions: Design, implementation, and performance evaluation," *SIGMETRICS Performance Evaluation Review*, vol. 36, pp. 85–96, 2008.
- [8] M. Medard and A. Sprintson, *Network Coding: Fundamentals and Applications*, ser. Academic Press. Elsevier, 2012.
- [9] R. Costa, D. Munaretto, J. Widmer, and J. Barros, "Informed network coding for minimum decoding delay," in *Proc. of the 5th IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS' 2008)*, Atlanta, Georgia, USA, Sept 2008, pp. 80–91.
- [10] P. Sadeghi, R. Shams, and D. Traskov, "An optimal adaptive network coding scheme for minimizing decoding delay in broadcast erasure channels," *EURASIP Journal on Wireless Communications and Networking*, vol. 2010, pp. 1–14, 2010.
- [11] M. Karim and P. Sadeghi, "Decoding delay reduction in broadcast erasure channels with memory for network coding," in *Proc. of IEEE 23rd International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC' 2012)*, Sydney, Australia, Sept 2012, pp. 60–65.
- [12] M. Muhammad, M. Berioli, G. Liva, and G. Giambene, "Instantly decodable network coding protocols with unequal error protection," in *Proc. of IEEE International Conference on Communications (ICC' 2013)*, Sydney, Australia, June 2013, pp. 5120–5125.
- [13] A. Douik, S. Sorour, T. Al-Naffouri, and M.-S. Alouini, "A lossy graph model for delay reduction in generalized instantly decodable network coding," *IEEE Wireless Communications Letters*, vol. 3, no. 3, pp. 281–284, June 2014.
- [14] Y. Liu and C. W. Sung, "Quality-aware instantly decodable network coding," *IEEE Transactions on Wireless Communications*, vol. 13, no. 3, pp. 1604–1615, March 2014.
- [15] X. Li, C.-C. Wang, and X. Lin, "Optimal immediately-decodable inter-session network coding (idnc) schemes for two unicast sessions with hard deadline constraints," in *Proc of 49th Annual Allerton Conference on Communication, Control, and Computing (Allerton' 2011)*, Monticello, IL, USA, Sept 2011, pp. 784–791.
- [16] L. Lu, M. Xiao, and L. Rasmussen, "Relay-aided broadcasting with instantaneously decodable binary network codes," in *Proc. of 20th International Conference on Computer Communications and Networks (ICCCN' 2011)*, Maui, Hawaii, July 2011, pp. 1–5.
- [17] A. Douik, S. Sorour, M. Alouini, and T. Y. Al-Naffouri, "On minimizing the maximum broadcast decoding delay for instantly decodable network coding," *Proc. of IEEE Vehicular Technology Conference (VTC-Fall' 2014)*, Vancouver, BC, Canada, vol. abs/1404.0265, 2014.
- [18] D. Lucani, M. Medard, and M. Stojanovic, "Random linear network coding for time-division duplexing: Field size considerations," in *Proc. of IEEE Global Telecommunications Conference (GLOBECOM' 2009)*, Honolulu, Hawaii, USA, Nov 2009, pp. 1–6.

- [19] S. Y. El Rouayheb, M. A. R. Chaudhry, and A. Sprintson, "On the minimum number of transmissions in single-hop wireless coding networks," in *Proc. of IEEE Information Theory Workshop (ITW' 2007)*, Bergen, Norway, Sept 2007, pp. 120–125.
- [20] P. Sadeghi, D. Traskov, and R. Koetter, "Adaptive network coding for broadcast channels," in *Proc. of Workshop on Network Coding, Theory, and Applications (NetCod' 2009)*, Lausanne, Switzerland, June 2009, pp. 80–85.
- [21] S. Sorour and S. Valaee, "Minimum broadcast decoding delay for generalized instantly decodable network coding," in *Proc. of IEEE Global Telecommunications Conference (GLOBECOM' 2010)*, Miami, Florida, USA, Dec 2010, pp. 1–5.
- [22] S. Sorour, N. Aboutorab, P. Sadeghi, M. Karim, T. Al-Naffouri, and M.-S. Alouini, "Delay reduction in persistent erasure channels for generalized instantly decodable network coding," in *Proc. of IEEE Vehicular Technology Conference (VTC-Fall' 2013)*, Dresden, Germany, June 2013, pp. 1–5.
- [23] S. Sorour and S. Valaee, "Effect of feedback loss on instantly decodable network coding," in *Proc. of IEEE 7th International Wireless Communications and Mobile Computing Conference (IWCMC' 2011)*, Istanbul, Turkey, July 2011, pp. 21–28.
- [24] X. Li, C.-C. Wang, and X. Lin, "Throughput and delay analysis on uncoded and coded wireless broadcast with hard deadline constraints," in *Proc. of IEEE Conference on Computer Communications (INFOCOM' 2010)*, San Diego, CA, USA, March 2010, pp. 1–5.
- [25] S. Sorour, A. Douik, S. Valaee, T. Al-Naffouri, and M. Alouini, "Partially blind instantly decodable network codes for lossy feedback environment," *IEEE Transactions on Wireless Communications*, vol. 13, no. 9, pp. 4871–4883, Sept 2014.
- [26] A. Tehrani and A. Dimakis, "Finding three transmissions is hard," in *Proc. of IEEE Global Telecommunications Conference (GLOBECOM' 2012)*, Anaheim, California, USA, Dec 2012, pp. 2293–2298.
- [27] E. Drinea, C. Fragouli, and L. Keller, "Delay with network coding and feedback," in *Proc. of IEEE International Symposium on Information Theory (ISIT' 2009)*, Seoul, Korea, June 2009, pp. 844–848.
- [28] K. Yamaguchi and S. Masuda, "A new exact algorithm for the maximum-weight clique problem," in *Proc. Of the 23rd International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC' 2008)*, Yamaguchi, Japan.
- [29] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., 1990.
- [30] M. Mushkin and I. Bar-David, "Capacity and coding for the gilbert-elliott channels," *IEEE Transactions on Information Theory*, vol. 35, no. 6, pp. 1277–1290, Nov 1989.
- [31] P. Sadeghi, R. Kennedy, P. Rapajic, and R. Shams, "Finite-state markov modeling of fading channels - a survey of principles and applications," *IEEE Signal Processing Magazine*, vol. 25, no. 5, pp. 57–80, September 2008.
- [32] P. R. J. Ostergard, "A fast algorithm for the maximum clique problem," *Discrete Appl. Math.*, vol. 120, pp. 197–207.
- [33] G. Ausiello, M. Protasi, A. Marchetti-Spaccamela, G. Gambosi, P. Crescenzi, and V. Kann, *Complexity and Approximation: Combinatorial Optimization Problems and Their Approximability Properties*, 1st ed. Springer-Verlag New York, Inc., 1999.
- [34] A. Le, A. Tehrani, A. Dimakis, and A. Markopoulou, "Instantly decodable network codes for real-time applications," in *Proc of International Symposium on Network Coding (NetCod' 2013)*, Calgary, Canada, June 2013, pp. 1–6.



**Ahmed Douik** (S'13) received the Eng. degree in electronic and communication engineering (with first class honors) from the Ecole Polytechnique de Tunisie, Tunisia, in 2013, the M.S. degree in electrical engineering from King Abdullah University of Science and Technology, Thuwal, Saudi Arabia, in 2015. He is now pursuing his Ph.D. at the California Institute of Technology, Pasadena, CA, USA. His research interests include cloud-radio access networks, network coding, single and multi-hop transmissions, and cooperation communication.



**Sameh Sorour** (S'98-M'11) is an Assistant Professor and King Fahd University of Petroleum and Minerals (KFUPM). He received his B.Sc. and M.Sc. degrees in Electrical Engineering from Alexandria University, Egypt, in 2002 and 2006, respectively. In 2011, he obtained his Ph.D. degree from the Wireless and Internet Research Laboratory (WIRLab), Department of Electrical and Computer Engineering, University of Toronto, Canada. After two postdoctoral fellowships at University of Toronto and King Abdullah University of Science and Technology (KAUST), he joined KFUPM in September 2013. His research interests include network coding, indoor localization, coded cooperative data exchange for machine-to-machine communications, cloud storage networks, online coded caching and smart networks.



**Tareq Y. Al-Naffouri** (M'10) received the B.S. degrees in mathematics and electrical engineering (with first honors) from King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia, in 1994, the M.S. degree in electrical engineering from the Georgia Institute of Technology, Atlanta, in 1998, and the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA, in 2004. He was a visiting scholar at the California Institute of Technology, Pasadena, from January to August 2005 and during summer 2006. He was a Fulbright scholar at the University of Southern California from February to September 2008. He has held internship positions at NEC Research Labs, Tokyo, Japan, in 1998, Adaptive Systems Lab, UCLA in 1999, National Semiconductor, Santa Clara, CA, in 2001 and 2002, and Beceem Communications Santa Clara, in 2004. He is currently an Associate Professor with the Electrical Engineering Department, King Fahd University of Petroleum and Minerals, and jointly at the Electrical Engineering Department, King Abdullah University of Science and Technology (KAUST). His research interests lie in the areas of adaptive, statistical, and sparse signal processing and their applications to wireless communications, seismic signal processing, and in multiuser information theory. He has more than 90 publications in Journal and conference proceedings, nine standard contributions, 6 issued patents, and 6 pending.

Dr. Al-Naffouri is the recipient of a 2001 Best Student Paper Award at the IEEE-EURASIP Workshop on Nonlinear Signal and Image Processing (NSIP) 2001 for his work on adaptive filtering analysis, the IEEE Education Society Chapter Achievement Award in 2008, and Al-Marai Award for innovative research in communication in 2009. He has been an Associate Editor of the IEEE Transaction on Signal Processing since August 2013.



**Mohamed-Slim Alouini** (S'94-M'98-SM'03-F'09) was born in Tunis, Tunisia. He received the Ph.D. degree in electrical engineering from the California Institute of Technology (Caltech), Pasadena, CA, USA, in 1998. He served as a faculty member at the University of Minnesota, Minneapolis, MN, USA, then in the Texas A&M University at Qatar, Education City, Doha, Qatar, before joining King Abdullah University of Science and Technology (KAUST), Thuwal, Makkah Province, Saudi Arabia, as a Professor of electrical engineering in 2009. His current research interests include the modeling, design, and performance analysis of wireless communication systems.

APPENDIX A  
PROOF OF THEOREM 1

**Remark 2.** *In order to simplify the notations, the receiver index will be dropped from all the variables. The final result will be presented with the index.*

To prove the theorem, we first show that the estimation of the channel, giving the feedback history, can be efficiently performed. If an arbitrary packet is transmitted only once, at time  $t^\dagger$ , and feedback is heard at time  $t^*$ , then state of the channel at time  $t^\dagger$  can be surely determined. If multiple packets satisfy the constraint for the same receiver, then using the latest one will give a more accurate estimation. Thus, we first show that such  $t^\dagger$  exists for all receivers. We, then use the most recent one to perform the channel estimation.

We first introduce the following variable: Let  $\alpha$  and  $\beta$  ( $\alpha > \beta$ ) be the indices of the two most recent frame in which a feedback is heard. The last packet among those attempted only once between the frames  $\alpha$  and  $\beta$  can be mathematically expressed as:

$$j^\dagger = \arg \max_{j \in \mathcal{W}} \left[ \bigcup_{k=\beta+1}^{\alpha} \lambda_j(k) \right]$$

subject to  $\left| \bigcup_{k=\beta+1}^{\alpha} \lambda_j(k) \right| = 1,$  (A.1)

where  $\cup_{x \in X} A_x$  is the union of the sets  $A_x$ ,  $\forall x \in X$  and  $|\cdot|$  represents the cardinality operator. The constraint in (A.1) ensures that the packet is attempted only once, whereas the objective function ensures that the latest packet is considered to have a more accurate estimate.

The assumption, in Section III, states that each targeted receiver, from the last time a feedback is heard, there is a packet that is attempted only once. Therefore, we have:

$$\exists j \in \mathcal{W} \text{ such that } \left| \bigcup_{k=\beta+1}^{\alpha} \lambda_j(k) \right| = 1. \quad (\text{A.2})$$

Let  $\mathbb{J}$  be the set of packets, attempted only once from frame  $(\beta + 1)$  to frame  $\alpha$ . The set is expressed as follows:

$$j \in \mathbb{J} \Leftrightarrow \left| \bigcup_{k=\beta+1}^{\alpha} \lambda_j(k) \right| = 1. \quad (\text{A.3})$$

Therefore, the optimization problem (A.1) can be rewritten as follows

$$\arg \max_{j \in \mathbb{J}} \left[ \bigcup_{k=\beta+1}^{\alpha} \lambda_j(k) \right]. \quad (\text{A.4})$$

From (A.2), we have  $\mathbb{J} \neq \emptyset$ , which guarantee the existence of  $j^\dagger$ . Therefore, the existence of time  $t^\dagger$  in which the packet  $j^\dagger$  is attempted, is guaranteed and can be written as:

$$t^\dagger = \bigcup_{k=\beta+1}^{\alpha} \lambda_{j^\dagger}(k). \quad (\text{A.5})$$

Note that since the packet is attempted only once (constraint (A.1)), then the union of all the sets contains a single element indicated by  $t^\dagger$ . The state of the channel at  $t^\dagger$  can be determined using the received feedback at time  $t^*$ . Depending on

the reception statue of  $j^\dagger$ , the channel state of receiver  $i$  is:

$$C_i(t^\dagger) = \begin{cases} G & \text{if } f_{ij^\dagger}(t^*) = 0 \\ B & \text{if } f_{ij^\dagger}(t^*) = 1. \end{cases} \quad (\text{A.6})$$

The probability for a transmission to be erased, at time  $t$ , can then be expressed as:

$$p_i(t) = \mathbb{P}(C_i(t) = B) = \begin{cases} \mathbb{P}(C_i(t) = B | C_i(t^\dagger) = G) & \text{if } f_{ij^\dagger}(t^*) = 0 \\ \mathbb{P}(C_i(t) = B | C_i(t^\dagger) = B) & \text{if } f_{ij^\dagger}(t^*) = 1. \end{cases} \quad (\text{A.7})$$

According to the analysis done in Lemma 2 (Appendix G), (A.7) is equivalent to:

$$p_i(t) = \begin{cases} b_i^f \sum_{l=0}^{t-t^\dagger-1} \mu_i^l & \text{if } f_{ij^\dagger}(t^*) = 0 \\ 1 - g_i^f \sum_{l=0}^{t-t^\dagger-1} \mu_i^l & \text{if } f_{ij^\dagger}(t^*) = 1. \end{cases} \quad (\text{A.8})$$

Since a feedback is heard at  $t^*$ , then the feedback channel is in the good state (i.e.,  $C_i(t^*) = G$ ). Therefore, according to Lemma 2, the probability for a feedback to be erased, at time  $t$ , can then be expressed as:

$$q_i(t) = b_i^b \sum_{l=0}^{t-t^*-1} \psi_i^l. \quad (\text{A.9})$$

APPENDIX B  
PROOF OF THEOREM 2

To prove this theorem, we list all possible events and compute the probability of each. Afterward, we calculate the expected decoding delay increase for an arbitrary receiver  $i$  after the transmission of packet  $\mathfrak{J}$ .

Assume receiver  $i$  is not targeted by a primary packet, two scenarios can occur:

- Want set of receiver  $i$  is Non or Partially Uncertain: delay is experienced upon successful reception of  $\mathfrak{J}$ . Thus, the probability of a delay increase is  $\mathbb{P}(d_i(\mathfrak{J}) = 1) = 1 - p_i$ .
- Want set of receiver  $i$  is Fully Uncertain: delay is experienced if the two following conditions are true:
  - 1)  $i$  received packet  $\mathfrak{J}$ .
  - 2)  $i$  did not obtain all its requested packets.

Since the events are independent, then the probability of a delay increase is  $\mathbb{P}(d_i(\mathfrak{J}) = 1) = (1 - p_i)(1 - p_{i,f})$ .

Assume receiver  $i$  is targeted by a primary packet, three scenarios can occur:

- State of the packet is known (i.e.,  $f_{i\mathfrak{J}_i} = 1$ ): no delay is experienced in this situation.
- State of the packet is unknown (i.e.,  $f_{i\mathfrak{J}_i} = x$ ): delay depends on the state of the Wants set as follows:
  - a) Wants set of receiver  $i$  is Non or Partially Uncertain: delay is experienced if the two following conditions are true:
    - 1)  $i$  received packet  $\mathfrak{J}$ .
    - 2)  $\mathfrak{J}_i$  is not innovative for  $i$ .

Since the events are independent, then the probability of a delay increase is  $\mathbb{P}(d_i(\mathfrak{J}) = 1) = (1 - p_i)(1 - p_{i,n}(\mathfrak{J}_i))$ .

- b) Wants set of receiver  $i$  is Fully Uncertain: delay is experienced if the three following conditions are true:

- 1)  $i$  received packet  $\mathfrak{J}$ .
- 2)  $\mathfrak{J}_i$  is not innovative for  $i$ .
- 3)  $i$  did not obtain all its requested packets.

The first event is independent of the two others. The probability of delay increment can be expressed as:

$$\mathbb{P}(d_i(\mathfrak{J}) = 1) = (1 - p_i)\mathbb{P}(2, 3). \quad (\text{B.1})$$

where  $\mathbb{P}(2, 3)$  is the probability for events 2 and 3 to happen.

To compute  $\mathbb{P}(2, 3)$ , we introduce the following variables. Let  $\mathfrak{J}_i^j$  be the event that packet  $j$  is innovative for receiver  $i$  and let  $\mathcal{F}_i$  be the event that receiver  $i$  gets all its requested packets. According to these definitions, we have:

$$\mathbb{P}(2, 3) = \mathbb{P}(d_i(\mathfrak{J}) = 1 | \overline{\mathfrak{J}_i^{\mathfrak{J}}})\mathbb{P}(\overline{\mathfrak{J}_i^{\mathfrak{J}}}) + \mathbb{P}(d_i(\mathfrak{J}) = 1 | \overline{\mathfrak{J}_i^{\mathfrak{J}}})\mathbb{P}(\overline{\mathfrak{J}_i^{\mathfrak{J}}}). \quad (\text{B.2})$$

Since no delay is experienced when a packet is innovative, then the probability is written as:

$$\mathbb{P}(2, 3) = \mathbb{P}(d_i(\mathfrak{J}) = 1 | \overline{\mathfrak{J}_i^{\mathfrak{J}}})(1 - p_{i,n}(\mathfrak{J}_i)). \quad (\text{B.3})$$

Develop the first term yields:

$$\mathbb{P}(d_i(\mathfrak{J}) = 1 | \overline{\mathfrak{J}_i^{\mathfrak{J}}}) = \mathbb{P}(d_i(\mathfrak{J}) = 1 | \overline{\mathfrak{J}_i^{\mathfrak{J}}}, \mathcal{F}_i)\mathbb{P}(\overline{\mathcal{F}_i} | \overline{\mathfrak{J}_i^{\mathfrak{J}}}) + \mathbb{P}(d_i(\mathfrak{J}) = 1 | \overline{\mathfrak{J}_i^{\mathfrak{J}}}, \overline{\mathcal{F}_i})\mathbb{P}(\overline{\mathcal{F}_i} | \overline{\mathfrak{J}_i^{\mathfrak{J}}}). \quad (\text{B.4})$$

Since no delay is experienced by receivers with empty Wants set, then  $\mathbb{P}(d_i(\mathfrak{J}) = 1 | \overline{\mathfrak{J}_i^{\mathfrak{J}}}, \mathcal{F}_i) = 0$ . From the definition of the decoding delay, receivers with non-empty Wants set receiving a non-innovative packet, experience a delay. In other words,  $\mathbb{P}(d_i(\mathfrak{J}) = 1 | \overline{\mathfrak{J}_i^{\mathfrak{J}}}, \overline{\mathcal{F}_i}) = 1$ . Finally, the probability can be expressed as:

$$\begin{aligned} \mathbb{P}(2, 3) &= (1 - p_{i,n}(\mathfrak{J}_i))\mathbb{P}(\overline{\mathcal{F}_i} | \overline{\mathfrak{J}_i^{\mathfrak{J}}}) \\ &= (1 - p_{i,n}(\mathfrak{J}_i))(1 - \prod_{k \in (\mathcal{W}_i \setminus \mathfrak{J}_i)} (1 - p_{i,n}(k))) \\ &= 1 - p_{i,n}(\mathfrak{J}_i) - p_{i,f}. \end{aligned} \quad (\text{B.5})$$

Given the analysis above, the likelihood that receiver  $i$  experiences a decoding delay, after the transmission  $\mathfrak{J}$  is:

$$\begin{aligned} \mathbb{P}(d_i(\mathfrak{J}) = 1) & \quad (\text{B.6}) \\ &= \begin{cases} 1 - p_i & i \in (\hat{\tau} \cap \overline{F}) \\ (1 - p_i)(1 - p_{i,f}) & i \in (\hat{\tau} \cap F) \\ 0 & i \in (\tau \cap \overline{U}) \\ (1 - p_i)(1 - p_{i,n}(\mathfrak{J}_i)) & i \in (\tau \cap (U \setminus F)) \\ (1 - p_i)(1 - p_{i,n}(\mathfrak{J}_i) - p_{i,f}) & i \in (\tau \cap F), \end{cases} \end{aligned}$$

### APPENDIX C PROOF OF THEOREM 3

**Remark 3.** In order to simplify the notations, the receiver index will be dropped from all the variables. The final result will be presented with the index.

Let  $\mathcal{I}^b(j, n)$  be the event that packet  $j$  is innovative for receiver  $i$  at the beginning of the frame  $n$  and let  $\mathcal{I}(j, t)$  be the event of the packet is innovative at time  $t$  (i.e.,  $\mathbb{P}(\mathcal{I}(j, t)) = p_n(j, t)$ ). Using the total probability theorem, the probability that packet  $j$  is innovative for receiver  $i$ , at time  $t$ , can be expressed as:

$$\begin{aligned} p_n(j, t) &= \mathbb{P}(\mathcal{I}(j, t) | \mathcal{I}^b(j, n^+(t)))\mathbb{P}(\mathcal{I}^b(j, n^+(t))) \\ &+ \mathbb{P}(\mathcal{I}(j, t) | \overline{\mathcal{I}^b(j, n^+(t))})\mathbb{P}(\overline{\mathcal{I}^b(j, n^+(t))}). \end{aligned} \quad (\text{C.1})$$

Clearly, a packet that is not innovative at the beginning of a frame is non-innovative during the whole frame. Therefore,  $\mathbb{P}(\mathcal{I}(j, t) | \overline{\mathcal{I}^b(j, n^+(t))}) = 0$ . The probability for an innovative packet at the beginning the frame to remain innovative at time  $t$  is the probability that the packet is erased each time it is attempted. Therefore, we obtain:

$$\mathbb{P}(\mathcal{I}(j, t) | \mathcal{I}^b(j, n^+(t))) = \prod_{k \in \lambda_j(n^+(t))} p(k). \quad (\text{C.2})$$

Define  $\mathcal{U}(n)$  as the event of an unheard feedback from receiver  $i$  during the frame  $n$  and define  $L(j, n)$  as the event that packet  $j$  is innovative at frame  $n$ :

$$L(j, n) = \mathbb{P}(\mathcal{I}^b(j, n) | \mathcal{U}(n)). \quad (\text{C.3})$$

The probability that a packet  $j$  is innovative at the beginning of frame  $n^+(t)$  can be expressed as:

$$\mathbb{P}(\mathcal{I}^b(j, n^+(t))) = \prod_{k \in \mathcal{K}_j} L(j, k). \quad (\text{C.4})$$

An unheard feedback  $\mathcal{U}(n)$  can occur if one of these scenarios occurs:

- All the packet sent during downlink frame number  $n$  are lost. This happen with probability  $\prod_{s \in \mathcal{X}^d(n)} p(s)$ .
- At least one packet arrived but the feedback is lost. This event happen with probability  $\left(1 - \prod_{s \in \mathcal{X}^d(n)} p(s)\right)q(u(n))$ .

Given that the events are independent, the probability of event  $\mathcal{U}(n)$  can be expressed as:

$$\begin{aligned} \mathbb{P}(\mathcal{U}(n)) &= \prod_{s \in \mathcal{X}^d(n)} p(s) \\ &+ \left(1 - \prod_{s \in \mathcal{X}^d(n)} p(s)\right)q(u(n)). \end{aligned} \quad (\text{C.5})$$

The event that a packet  $j$  is not received after a frame  $n$  in which receiver  $i$  is targeted (i.e.,  $\overline{\mathcal{I}^b(j, n) | \mathcal{U}(n)}$ ), can occur according to theses scenarios:

- All the sent packets during downlink frame number  $n$  are lost. This event happen with probability  $\prod_{s \in \mathcal{X}^d(n)} p(s)$ .
- The considered packet is lost, at least one of the other packet arrived and the feedback is lost. This event happen with probability  $\prod_{s \in \lambda_j(n)} p(s) \times \left(1 - \prod_{s \in \mathcal{X}^d(n) \setminus \lambda_j(n)} p(s)\right)q(u(n))$ .

Given the independence of the events, the expressions of

$L(j, n)$  becomes:

$$\begin{aligned} L(j, n) &= \frac{\mathbb{P}(\mathcal{I}^b(j, n) | \mathcal{U}(n))}{\mathbb{P}(\mathcal{U}(n))} \\ &= \left( \prod_{s \in \mathcal{X}^d(n)} p(s) + \prod_{s \in \lambda_j(n)} p(s) \left[ 1 - \prod_{s \in \mathcal{X}^d(n) \setminus \lambda_j(n)} p(s) \right] q(u(n)) \right) \\ &\quad \times \left( \prod_{s \in \mathcal{X}^d(n)} p(s) + \left[ 1 - \prod_{s \in \mathcal{X}^d(n)} p(s) \right] q(u(n)) \right)^{-1}. \end{aligned} \quad (\text{C.6})$$

Substituting (C.6) and (C.2) in (C.1), the probability that packet  $j$  is innovative for receiver  $i$ , at time  $t$  is:

$$\begin{aligned} p_{i,n}(j, t) &= \prod_{k \in \lambda_{ij}(n^+(t))} p(k) \prod_{k \in \mathcal{K}_{ij}} \left\{ \left( \prod_{s \in \mathcal{X}^d(k)} p(s) + \prod_{s \in \lambda_{ij}(k)} p(s) \left[ 1 - \prod_{s \in \mathcal{X}^d(k) \setminus \lambda_{ij}(k)} p(s) \right] q(u(k)) \right) \right. \\ &\quad \times \left. \left( \prod_{s \in \mathcal{X}^d(k)} p(s) + \left[ 1 - \prod_{s \in \mathcal{X}^d(k)} p(s) \right] q(u(k)) \right)^{-1} \right\}. \end{aligned} \quad (\text{C.7})$$

#### APPENDIX D PROOF OF COROLLARY 1

To prove this corollary, we compute the finish probability for receivers not having Full Uncertain Wants set and those we have. We, then, unify the expression of the finish probability for both types of receivers.

First note that if receiver  $i$  does not have all his primary needed packets in an uncertain state, then naturally, the receiver did not finish receiving all its primary packets. In other words,  $p_{i,f} = 0$ .

Assume now receiver  $i$  have all his primary packets in an uncertain state. The event of receiving all the primary packets with  $\mathcal{W}_i \neq \emptyset$  can occur if these two conditions are true:

- 1) The Uncertain set is equal to the Wants set ( i.e.,  $\mathcal{W} = \mathcal{X}$ ). In other words, all the primary packets were attempted but not acknowledged.
- 2) All the packets in the Uncertain set were successfully received.

Recall that the probability of a correctly received and not acknowledged packet is the inverse its innovative possibility. Hence, the probability receiver  $i$  have all its primary packets but  $\mathcal{W}_i \neq \emptyset$  can be written as:

$$p_{i,f} = \prod_{j \in \mathcal{X}_i} (1 - p_{i,n}(j)) = \prod_{j \in \mathcal{W}_i} (1 - p_{i,n}(j)). \quad (\text{D.1})$$

If receiver  $i$  do not have all his primary needed packets in an uncertain state (i.e.,  $\mathcal{W}_i \setminus \mathcal{X}_i \neq \emptyset$ ), then there is at least one primary packet that is wanted. In mathematical term, we have  $\exists j^* \in (\mathcal{W}_i \setminus \mathcal{X}_i)$ . Given that the finish probability of such

packet  $j^*$  is  $p_{i,n}(j^*) = 1$ , we have:

$$\begin{aligned} \prod_{j \in \mathcal{W}_i} (1 - p_{i,n}(j)) &= (1 - p_{i,n}(j^*)) \prod_{j \in \mathcal{W}_i \setminus j^*} (1 - p_{i,n}(j)) \\ &= 0 = p_{i,f}. \end{aligned} \quad (\text{D.2})$$

Finally, we write the expression of the finish probability for any arbitrary receiver  $i$  as follows:

$$p_{i,f} = \prod_{j \in \mathcal{W}_i} (1 - p_{i,n}(j)). \quad (\text{D.3})$$

#### APPENDIX E PROOF OF THEOREM 4

Let  $\mathcal{D}(\kappa)$  be the sum of the decoding delay after the transmission  $\kappa$  (i.e.,  $\mathcal{D}(\kappa) = \sum_{i=1}^M d_i(\kappa)$ ). According to the analysis done in Section IV, the expected decoding delay increase can be expressed as:

$$\begin{aligned} \mathbb{E}[\mathcal{D}(\kappa)] &= \sum_{i \in \tilde{\tau}(\kappa) \cap \bar{F}} (1 - p_i) \\ &\quad + \sum_{i \in \tilde{\tau}(\kappa) \cap F} (1 - p_i)(1 - p_{i,f}) \\ &\quad + \sum_{i \in \tau(\kappa) \cap (U \setminus F)} (1 - p_i)(1 - p_{i,n}(\mathfrak{J}_i(\kappa))) \\ &\quad + \sum_{i \in \tau(\kappa) \cap F} (1 - p_i)(1 - p_{i,n}(\mathfrak{J}_i(\kappa)) - p_{i,f}), \end{aligned} \quad (\text{E.1})$$

where  $\mathfrak{J}_i(\kappa)$  is the intended packet of receiver  $i$  in the transmission  $\kappa$ . The optimal packet combination  $\kappa^*$  that reduces the on-line decoding delay (E.1) is the following:

$$\begin{aligned} \kappa^* &= \arg \min_{\kappa \in \mathcal{P}(\mathcal{N})} \{\mathbb{E}[\mathcal{D}(\kappa)]\} \\ &= \arg \max_{\kappa \in \mathcal{P}(\mathcal{N})} \left\{ \sum_{i \in \tau(\kappa) \cap \bar{F}} (1 - p_i) \right. \\ &\quad + \sum_{i \in \tau(\kappa) \cap F} (1 - p_i)(1 - p_{i,f}) \\ &\quad - \sum_{i \in \tau(\kappa) \cap (U \setminus F)} (1 - p_i)(1 - p_{i,n}(\mathfrak{J}_i(\kappa))) \\ &\quad \left. - \sum_{i \in \tau(\kappa) \cap F} (1 - p_i)(1 - p_{i,n}(\mathfrak{J}_i(\kappa)) - p_{i,f}) \right\}. \end{aligned} \quad (\text{E.2})$$

The first and the second term of (E.2) can be rearranged as follows:

$$\begin{aligned} &\sum_{i \in \tau(\kappa) \cap \bar{F}} (1 - p_i) + \sum_{i \in \tau(\kappa) \cap F} (1 - p_i)(1 - p_{i,f}) \\ &= \sum_{i \in \tau(\kappa)} (1 - p_i) - \sum_{i \in \tau(\kappa) \cap F} (1 - p_i)p_{i,f}. \end{aligned} \quad (\text{E.3})$$

The last term of (E.2) can be developed to yield:

$$\begin{aligned} &\sum_{i \in \tau(\kappa) \cap F} (1 - p_i)(1 - p_{i,n}(\mathfrak{J}_i(\kappa)) - p_{i,f}) \\ &= \sum_{i \in \tau(\kappa) \cap F} (1 - p_i)(1 - p_{i,n}(\mathfrak{J}_i(\kappa))) \\ &\quad - \sum_{i \in \tau(\kappa) \cap F} (1 - p_i)p_{i,f}. \end{aligned} \quad (\text{E.4})$$

Substituting the equations (E.3) and (E.4) in (E.2), the optimal packet combination to reduce the on-line decoding delay can be expressed as:

$$\begin{aligned}
\kappa^*(t) &= \arg \max_{\kappa \in \mathcal{P}(\mathcal{N})} \left\{ \sum_{i \in \tau(\kappa)} (1 - p_i) \right. \\
&\quad - \sum_{i \in \tau(\kappa) \cap (U \setminus F)} (1 - p_i)(1 - p_{i,n}(\mathfrak{J}_i(\kappa))) \\
&\quad \left. - \sum_{i \in \tau(\kappa) \cap F} (1 - p_i)(1 - p_{i,n}(\mathfrak{J}_i(\kappa))) \right\} \\
&= \arg \max_{\kappa \in \mathcal{P}(\mathcal{N})} \left\{ \sum_{i \in \tau(\kappa)} (1 - p_i) \right. \\
&\quad \left. - \sum_{i \in \tau(\kappa) \cap U} (1 - p_i)(1 - p_{i,n}(\mathfrak{J}_i(\kappa))) \right\}. \quad (\text{E.5})
\end{aligned}$$

If the state of a given packet is known, then the probability that it is innovative is 1 (i.e.,  $p_{i,n}(\mathfrak{J}_i(\kappa)) = 1$ ,  $\forall i \in (\tau(\kappa) \setminus U)$ ). This last equality allows the reformulation of the last term of (E.5) as follows:

$$\begin{aligned}
&\sum_{i \in \tau(\kappa) \cap U} (1 - p_i)(1 - p_{i,n}(\mathfrak{J}_i(\kappa))) \\
&= \sum_{i \in \tau(\kappa)} (1 - p_i)(1 - p_{i,n}(\mathfrak{J}_i(\kappa))). \quad (\text{E.6})
\end{aligned}$$

Finally, the optimal packet combination to reduce the on-line decoding delay is:

$$\kappa^* = \arg \max_{\kappa \in \mathcal{P}(\mathcal{N})} \sum_{i \in \tau(\kappa)} (1 - p_i)p_{i,n}(\mathfrak{J}_i(\kappa)). \quad (\text{E.7})$$

According to the analysis done in [34], the optimization problem (E.7) is equivalent to a maximum weight clique problem in the G-IDNC graph where the weights of vertex  $v_{ij}$  can be expressed as:

$$w_{ij}^0 = (1 - p_i)p_{i,n}(j). \quad (\text{E.8})$$

## APPENDIX F

### PROOF OF LEMMA 1

In a situation without feedback intermittence, the length of the downlink frame is equal to 1, in other words,  $T = 2$ . Each receiver sends a feedback upon successful reception. The erasure probability can be simplified as follows:

$$p_i = b_i^f \sum_{l=0}^{t-t_i^{(0)}-1} \mu_i^l. \quad (\text{F.1})$$

Without loss of generality, we assume that packets are attempted at times  $t = 2k + 1$ ,  $\forall k \in \mathbb{N}$  and feedback is received at times  $t = 2k$ ,  $\forall k \in \mathbb{N}^+$ . Therefore, the innovative probability can be written as:

$$p_{i,n}(j, t) = p_{i,n}(j, 2n^+(t) + 1). \quad (\text{F.2})$$

Note that only one packet can be attempted during a frame then  $|\lambda_{ij}(n)| \leq 1$  and  $|\mathcal{X}_i^d(n)| \leq 1, \forall n$ . More specifically, we have:

$$\lambda_{ij}(k) = \begin{cases} \mathcal{X}_i^d(k) = 2k - 1 & \text{if } k \in \mathcal{K}_{ij} \\ \emptyset & \text{if } k \notin \mathcal{K}_{ij}. \end{cases} \quad (\text{F.3})$$

The likelihood that packet  $j$  is innovative to receiver  $i$  at time  $t$  can be simplified as follows:

$$p_{i,n}(j, t) = \prod_{k \in \mathcal{K}_{ij}} \frac{p_i(2k - 1)}{p_i(2k - 1) + (1 - p_i(2k - 1))q_i(2k)}. \quad (\text{F.4})$$

## APPENDIX G

### AUXILIARY LEMMA

To estimate the channel and feedback erasure probability, we first introduce the following lemma:

**Lemma 2.** Let  $(X_n)_{n \geq 1}$  be a two state ( $x$  and  $y$ ) Markov chain, with  $P_{tr_{x \rightarrow y}}$  and  $P_{tr_{y \rightarrow x}}$  the transition probability from state  $x$  to  $y$  and  $y$  to  $x$ , respectively. Let  $\mu = (1 - P_{tr_{x \rightarrow y}} - P_{tr_{y \rightarrow x}})$  be the memory of the chain. Define  $f(n) = \mathbb{P}(X_n = y | X_{n^0} = x)$ ,  $\forall n \geq n^0$ . We have:

$$f(n) = P_{tr_{x \rightarrow y}} \sum_{i=0}^{n-n^0-1} \mu^i \quad (\text{G.1})$$

*Proof:* If  $n = n^0$ , then it is clear that:

$$f(n) = \mathbb{P}(X_{n^0} = y | X_{n^0} = x) = 0 \quad (\text{G.2})$$

If  $n > n^0$ , then the relationship linking  $f(n)$  and  $f(n - 1)$  can be obtained by applying the total probability theorem as follows:

$$\begin{aligned}
f(n) &= \mathbb{P}(X_n = y | X_{n^0} = x) \\
&= \mathbb{P}(X_n = y | X_{n-1} = x, X_{n^0} = x) \mathbb{P}(X_{n-1} = x | X_{n^0} = x) \\
&\quad + \mathbb{P}(X_n = y | X_{n-1} = y, X_{n^0} = x) \mathbb{P}(X_{n-1} = y | X_{n^0} = x) \\
&= \mathbb{P}(X_n = y | X_{n-1} = x) \mathbb{P}(X_{n-1} = x | X_{n^0} = x) \\
&\quad + \mathbb{P}(X_n = y | X_{n-1} = y) \mathbb{P}(X_{n-1} = y | X_{n^0} = x) \\
&= (1 - P_{tr_{y \rightarrow x}})f(n - 1) + P_{tr_{x \rightarrow y}}(1 - f(n - 1)) \\
&= \mu f(n - 1) + P_{tr_{x \rightarrow y}}. \quad (\text{G.3})
\end{aligned}$$

By a simple computation of the previous sequence, the expression of  $f(n)$  becomes:

$$\begin{aligned}
f(n) &= \mu^{n-n^0} f(n^0) + P_{tr_{x \rightarrow y}} \sum_{i=0}^{n-n^0-1} \mu^i \\
&= P_{tr_{x \rightarrow y}} \sum_{i=0}^{n-n^0-1} \mu^i \quad (\text{G.4})
\end{aligned}$$

■