Network-Coded Cooperation Over Time-Varying Channels

Hana Khamfroush, Student Member, IEEE, Daniel E. Lucani, Member, IEEE, João Barros, Senior Member, IEEE, and Peyman Pahlevani, Student Member, IEEE

Abstract—In this paper, we investigate the optimal design of cooperative network-coded strategies for a three-node wireless network with time-varying half-duplex erasure channels. To this end, we formulate the problem of minimizing the total cost of transmitting $M$ packets from source to two receivers as a Markov decision process (MDP). The actions of the MDP model include the source and the type of transmission to be used in a given time slot given perfect knowledge of the system state. The cost of packet transmission is defined such that it can incorporate the difference between broadcast and unicast transmissions, e.g., in terms of the rate of packet transmission or the energy consumption. A comprehensive analysis of the MDP solution is carried out under different network conditions to extract optimal rules of packet transmission. Inspired by the extracted rules, we propose two near-optimal heuristics that are suitable for practical systems. We use two wireless channel models to analyze the performance of the proposed heuristics in practical wireless networks, namely, 1) an infrastructure-to-vehicle communication in a highway scenario considering Rayleigh fading and 2) real packet loss measurements for WiFi using Aalborg University’s Raspberry Pi testbed. We compare our results with random linear network coding broadcasting schemes showing that our heuristics can provide up to $2 \times$ gains in completion time and up to $4 \times$ gains in terms of reliably serviced data packets.

Index Terms—Network coding, wireless networks, cooperative system.

I. INTRODUCTION

Motivated by the growing use of wireless networks to support a variety of services that require the transmission of blocks of data to multiple users, e.g., video streaming or progressive download of videos in youtube, there is a significant interest in minimizing the cost of such data transmission. Particularly, mechanisms that can offload infrastructure networks (e.g., cellular networks) have gathered interest in academia [1]–[3] and steps are on the way to introduce it to 3rd Generation Partnership Project (3GPP) as a part of device-to-device (D2D) communication [4]. On the other hand, the time-varying nature of wireless communication channels due to mobility, increased traffic, and a variety of other factors, poses a challenging task for delivering such large amounts of content to multiple users in a reliable and timely fashion. Therefore, the key focus has been on fully utilizing the capacity available in wireless systems, as well as providing robust strategies that can minimize the cost of reliable packet transmission to multiple users in a wireless network. To this end, cooperative data exchange have been proposed to exploit the spatial diversity gains of multi-user networks to improve network connectivity, communication reliability, and to enhance power and spectrum efficiency [5], [6]. Cooperative communication allows terminals in a network to hear and help the information transmission of each other, by taking advantage of the broadcast nature of wireless communications. Broadcast channels with cooperating receivers have been extensively studied in the literature, e.g., [7]–[11]. Dabora et al. [8], investigated the achievable rates of two static cooperative data exchange schemes without looking at the optimal time of starting cooperation, namely, (a) single step cooperation, where each receiver sends a cooperative message to the other receiver for every received packet from the broadcast channel, and (b) two-step cooperation, where after reception one receiver generates a single cooperation message based on its channel input, but the second receiver generates its cooperation message only after decoding. The work in [9] computes the upper and lower bounds for the number of broadcast transmissions required to deliver a set of $n$ packets to a set of $k$ clients, assuming that each client initially holds a subset of packets and cooperative data exchange is allowed. Authors of [10] considered a similar problem with a certain transmission cost associated to each client, and presented an efficient algorithm to minimize the total transmission cost. In [11], a coded cooperative data exchange scheme for multiple unicasts to a set of $n$ clients is proposed, where each client $c_i$ is interested in a specific message $x_i$.

On the other hand, network coding (NC) [12] constitutes a key technology that can improve the performance of cooperative communication in terms of transmission reliability, efficiency, and security by allowing the nodes to use algebraic operations to mix packets they receive or generate. Recent works in [13]–[27] have revealed the benefits of NC for wireless cooperative communications in a variety of scenarios. Mainly, two general network coded cooperative scenarios have been investigated in the literature. First, scenarios with one/multiple sources transmitting data to multiple users with/without the help of relays and cooperative users attempting to receive...
packets from the source(s). Second, scenarios with multiple nodes working together to deliver their packets to a common destination. We call the first set of scenarios downstream cooperation and the second upstream cooperation. For downstream cooperation, the state of the art has considered evaluating the performance of network coding cooperation in terms of diversity multiplexing, and outage probability [13], as well as the application of network coding cooperation in (i) exploiting route selection strategies in multi-rate networks [14], (ii) proposing cluster-based routing protocols [15], (iii) improving user’s perceived QoS in multimedia broadcast/multicast services (MBMS) [16], and (iv) session grouping and relay node selection [17]–[20]. For upstream cooperation, the literature has focused on (i) developing adaptive strategies [21] and constructing distributed network codes [22], (ii) evaluating performance of coded cooperation in a network with two cooperating users, in terms of cooperative diversity and outage probability [23]–[25], and maximal throughput [26], and (iii) determining implementation requirements and deploying cooperative strategies [27]. Despite of the extensive efforts to evaluate the performance of NC in cooperative scenarios, a more in-depth analysis of time-varying scenarios, and particularly, the design of optimal NC cooperative policies and protocols is missing in the literature. In fact, most of the previous works focus only on a predefined packet transmission policy and not on determining the optimal policy given protocol design considerations.

We break from this trend by not assuming a transmission policy a priori. Instead, we seek an optimal policy to minimize the total cost of packet transmission. We exploit some of the intuitions in [28] to provide an in-depth analysis of the optimal solution for this optimization problem in a time-varying scenario. More precisely, we focus on the problem of minimizing the total cost required to complete the transmission of $M$ packets to two receivers from a source [see Fig. 1(a)], exploiting different communication options and constraints as well as assuming independent erasure channels between nodes. The problem is modeled as a Markov Decision Process (MDP). We also propose two close-to-optimal heuristics and test them for two relevant practical scenarios, namely, I2V communication in a high-way scenario and a WiFi scenario using a Raspberry Pi test-bed at Aalborg University. Our results show that the proposed heuristics have close-to-optimal performance in both scenarios and they are able to provide up to two-fold gain in the time required to deliver all packets to the terminals (completion time) and up to a four-fold gain in the maximum number of reliably transmitted packets with respect to random linear network coding (RLNC) [29] broadcasting. Although we look at the case of one source and two receivers, the extracted rules are beneficial to more complex cooperative scenarios and a wide range of network coding multi-hop routing protocols to provide a local optimization of their performance. We make the following contributions.

**Fundamental analysis of a multi-user network for time-varying and time-invariant channels:** We solve the problem of minimizing the completion cost of transmitting $M$ packets from a common source to two receivers for time-varying channels by using an MDP model in Section III. Since the number of meaningful states for the MDP model increases by the number of packets we are transmitting, we consider a finite time horizon and a finite number of packets to minimize the complexity of the MDP model. Given this complexity, we use our MDP solution as a way to evaluate performance of our proposed heuristics for small and moderate $M$.

**Development of efficient heuristics:** We evaluate the distribution of the selected actions by our MDP solution for a time-invariant channel scenario under various system and network conditions (see Section IV). We use this representation to extract meaningful rules for transmissions as a function of time and network conditions. Inspired by the extracted rules, we propose two near-optimal heuristics, namely, Intermediate-Feedback (IF) heuristics and Minimum-Feedback (MF) heuristics with a minimal feedback requirement (see Section V).

**Performance evaluation of the proposed heuristics for time-varying channels:** Two relevant time-varying setups are considered for evaluating performance of our heuristics. In Section VI, we show that the IF and MF heuristics can outperform the performance of RLNC broadcasting in terms of completion time by a factor of 2 and 1.75, respectively. We define a metric called percentage of reliability that shows the percentage of completed transmissions of $M$ packets in $T$ time slots and it could be considered as a proxy to measure the maximum $M$ that can be transmitted reliably. Our results show that the IF is able to increase the reliability of packet transmission in a finite time horizon by a factor of four with respect to the RLNC broadcasting.

**II. PROBLEM STATEMENT**

We consider a network with one source, $S$, and two receivers, $R_1$, $R_2$, as shown in Fig. 1. We assume that the receivers can share their knowledge with each other using unicast transmissions over the available channel between them or receive data directly from source. We define all possible ways of packet transmission as actions $a_1$–$a_5$ in Section III. A time-slotted system is considered and it is assumed that one of the 5 possible actions can be selected per time slot. We consider independent time-varying erasure channels for each receiver with $e_i(t)$ representing the erasure probability of channel $i$ at time $t$. For simplicity, we assume that the channel between receivers is symmetric, later in Section VI-B we will show the validity of our analysis for the asymmetric channels as well. When chosen to transmit, the source or one of the receivers generate RLNC coded packets by linearly combining packets available in their buffer. The source mixes $M$ packets, namely, $p_1, p_2, \ldots, p_M$ from the buffer and creates coded packets as a linear combination of these with some coding coefficients $\alpha_1, \ldots, \alpha_M$. 

![Fig. 1. Network model; solid lines represent unicast, dotted lines represent broadcast.](image-url)
i.e., \( \sum_{i=1}^{M} \alpha_i p_i \). The coding coefficients are independently and randomly selected from a Galois field of size \( q \), i.e., \( GF(q) \). The receivers perform similar operations but linearly combining coded packets in their buffers at the time of transmission. In general, less than \( M \) coded packets are available. It is assumed that \( q \) is large enough so that any RLNC packet received from the source is independent from previously received packets with very high probability. However, this is not necessarily the case for transmissions between \( R_1, R_2 \) because they may share common linear combinations. Our goal is to find an optimal/near-optimal transmission policy that can minimize the total cost of successful transmission of \( M \) packets from source to the two receivers over time-varying channels. The cost of each transmission is defined as a function of different parameters depending on the application, e.g., if the application is not delay tolerant, then the completion time of packet transmission should be the cost we aim to minimize. To differentiate between the cost of packet transmission for unicast and broadcast actions, the cost of broadcast is defined as \( \beta \geq 1 \) times the cost of unicast and one unit of cost is assigned to each unicast transmission.

### III. The MDP Model of the Problem

To determine the optimal policy for minimizing the cost, we assume a Genie aided system (GS), where each node in the network has perfect knowledge of the system state and also the erasure probabilities of the channels per time slot. Thus, the MDP policy has all required information to choose an optimal action per time slot. Note that optimality does not refer to the method we use to solve the MDP, but to the underlying assumptions of the model. We will drop the assumption of having a GS, later for our proposed heuristics. At each time step, the process is in a state \( s \), and the system may choose an action \( a \) that is possible in that state. The action chosen will determine the probability of transition into a new state \( s' \) and giving the system a corresponding reward or adding a corresponding cost. In the following, we specify the state, possible actions, transition probabilities, and optimization algorithm in our model.

**State definition:** each state \( s \) is defined as \( s = (i_1, i_2, c, t) \), where \( i_k \) is the number of degrees of freedom (dof)\(^1\) for the received packets at receiver \( R_k \), and \( c \) represents the dimension of the common knowledge between \( R_1 \) and \( R_2 \). \( t \) represents the instance of time that the observation is made and could have an integer value in \( \{0, 1, \ldots, T\} \), where \( T \) is a finite time horizon that is defined large enough so that the transmission of \( M \) packets can be finished before \( T \) is expired. Since the complexity of the MDP model depends on the number of meaningful states, we will look at the network for a finite horizon of time to reduce the complexity. As a simple example, assume that we are aiming at transmitting 4 packets \( \{P_1, P_2, P_3, P_4\} \) from a source to two receivers \( R_1, R_2 \) and the set of packets received by \( R_1 \) and \( R_2 \) at time \( t \) are respectively, \( \{P_1, P_1 + P_2, P_3 + P_4\} \) and \( \{P_3, P_1 + P_2 + P_3\} \). The state of network in this case is shown as \( s = (3, 3, 2, t) \), because the dof of the received packets by \( R_1 \) and \( R_2 \) are 3 and the common knowledge between receivers is shown as \( \{P_1 + P_2, P_3 + P_4\} \) that has a dimension of 2.

We call a state a meaningful state if and only if the elements of the state satisfy all of the following three conditions: I) \( t \geq i_1 + i_2 - c \), II) \( c \leq \min(i_1, i_2) \), III) \( i_1 + i_2 - c \leq M \). Condition (I) indicates the minimum number of time slots that is required to have \( i_1 \) dof at \( R_1 \) and \( i_2 \) dof at \( R_2 \). Condition (II) states that the dimension of the common knowledge between two receivers cannot be greater than the minimum of the knowledge received by the two receivers, and condition (III) states that the total knowledge received by both receivers cannot be greater than \( M \). In the previous example, the set of states defined as \( s = \{(3, 3, 2, t), \forall t \in \{4, 5, \ldots, T\}\} \) are the meaningful states. The set of absorbing states of our MDP model is defined as \( s_{abs} = (M, M, M, t), \forall t \in \{M, M + 1, \ldots, T\} \), meaning that if the network is in one of these states it cannot leave the state and a self-transition is performed for every selected action. In other words, the packet transmission process is completed when the network is in an absorbing state.

**Possible Actions** \( (a_j) \): we define five actions, \( a_1 - a_5 \) that cover all possible ways of packet transmission in the network of Fig. 1, assuming fixed transmission rate and the same modulation for all nodes. We may be able to define more actions if we are allowing for various slot sizes for different modulations and coding schemes, but these are beyond the scope of this paper. Action \( a_1 \) is defined as broadcast from \( S \) to \( R_1, R_2 \). Actions \( a_2, a_3 \) define the unicast transmissions from \( S \) to \( R_1 \) and \( R_2 \), respectively. Actions \( a_4, a_5 \) define cooperation between receivers. More precisely, \( a_4 \) defines unicast transmission of RLNC packets from \( R_1 \) to \( R_2 \) and \( a_5 \) defines unicast transmission of RLNC packets from \( R_2 \) to \( R_1 \). To allow the system to stop transmission after \( R_1 \) and \( R_2 \) have both received the \( M \) packets, action \( a_6 \) is defined as “do not transmit”.

**Transition Probabilities:** the possible states to which state \( (i_1, i_2, c, t) \) can transit to with non-zero probability depends on the action chosen and the total knowledge \( (K) \) that is available to both receivers at time \( t \). \( K \) indicates the dof of the union of the received packets by both receivers and is calculated as \( K = i_1 + i_2 - c \). We refer to \( p_{z \rightarrow y} \) as the probability of transition from state \( x \) to state \( y \). Note that there are two cases where the state of the network does not change: 1) the packet is not received correctly (is erased by the channel), and 2) the packet is received correctly but it is noninnovative, i.e., the received packet is not linearly independent from previously received packets. We refer to \( I_{z \in X} \) as the indicator function that is one when \( x \in X \) and zero otherwise. For simplicity, \( I_{z_1 \rightarrow z_2} = I_{i_1 + k_1, i_2 + k_2, c = c + k_3, t = t + k_4} \) is denoted by \( I_{(k_1, k_2, k_3, k_4)} \) and \( \epsilon(t) = 1 - \epsilon_i(t) \). The non-zero transition probabilities for the six actions are summarized as follows.

**Action** \( a_1 \) (Broadcast): when the source broadcasts, it creates different possible state transitions. These can be obtained by combinatorial arguments and we explain the more surprising cases.

On the one hand, assuming that the packet is received without erasure at \( R_1, R_2 \) and depending on the total knowledge \( (K) \) that is available to both. If \( K < M \) and the packet is not erased by any one of the channels, then the dimension of the common knowledge between \( R_1, R_2 \) is increased by one since both \( R_1, R_2 \),

\(^1We say that a node has \( k \) degrees of freedom if the dimension of its knowledge space is \( k \), where the knowledge space of a receiver is defined as the linear span of the linear combinations of symbols received by that receiver.
Fig. 2. Schematic of the MDP model; \( a_j \) is the selected action.

\( R_2 \) have received the same packet that is innovative to both of them. If \( K = M \) while none of the receivers has \( M \) dof, and the packet is not erased, the dimension of the common knowledge between \( R_1, R_2 \) is increased by two. Let us illustrate this with an example. Assuming that \( M = 3 \) and the sets of packets received by \( R_1, R_2 \) until time \( t \) are \( \{ P_1, P_3 \} \) and \( \{ P_2 + P_3 \} \), respectively. The network state is then \( s = (2,1,0,t) \). Now assume that source broadcasts a new coded packet \( P_1 + P_2 + P_3 \), which adds one dof to both \( R_1 \) and \( R_2 \). However, the dimension of the common knowledge is increased by two and the system then transits to a new state \( s' = (3,2,1,t+1) \). The time instance, \( t \), is increased by one, since the transition between the two states is performed in one time slot.

On the other hand, if only one of the receivers has \( M \) dof and the other one has less than \( M \) dof, then any new coded packet sent by source that is not erased by the channels adds one dof to the receiver with \( dof < M \) and also increases the dimension of the common knowledge by one. This is because the receiver with \( M \) dof already has enough number of linear independent coded packets to decode the original packets and therefore, any new coded packet transmitted by source is non-innovative to the set of packets received by that receiver. We now summarize all possible transitions with non-zero probabilities for source broadcasting as:

- If \( K < M \), \( i_1 < M \), \( i_2 < M \), then \( P_{(i_1,i_2,c,t)} \rightarrow (i'_1,i'_2,c',t') = e_1(t)e_2(t)I_{(0,0,0,1)} + e_1(t)e_2(t)I_{(1,1,1,1)} + e_1(t)e_2(t)I_{(1,0,0,1)} + e_1(t)e_2(t)I_{(0,0,0,1)} + e_1(t)e_2(t)I_{(1,1,1,1)}. \)

- If \( K = M \), \( i_1 < M \), \( i_2 < M \), then \( P_{(i_1,i_2,c,t)} \rightarrow (i'_1,i'_2,c',t') = e_1(t)e_2(t)I_{(0,0,0,1)} + e_1(t)e_2(t)I_{(0,1,1,1)} + e_1(t)e_2(t)I_{(0,1,1,1)} + e_1(t)e_2(t)I_{(1,0,0,1)} + e_1(t)e_2(t)I_{(0,1,1,1)}. \)

- If \( K = M \), \( i_1 = M \), \( i_2 < M \), then \( P_{(i_1,i_2,c,t)} \rightarrow (i'_1,i'_2,c',t') = e_1(t)I_{(0,0,0,1)} + e_2(t)I_{(0,1,1,1)}. \)

- If \( K = M \), \( i_1 < M \), \( i_2 = M \), then \( P_{(i_1,i_2,c,t)} \rightarrow (i'_1,i'_2,c',t') = e_1(t)I_{(0,0,0,1)} + e_1(t)I_{(1,0,0,1)}. \)

- If \( K = B_1 = i_2 = M \), then \( P_{(i_1,i_2,c,t)} \rightarrow (i'_1,i'_2,c',t') = I_{(0,0,0,0)}. \)

**Action \( a_2 \) (unicast from \( S \) to \( R_1 \)):** could be seen as a special case of broadcasting, \( a_1 \), in the sense that the erasure probability of the link between \( S \) and \( R_2 \) is one, i.e., \( \epsilon_2 = 1 \).

**Action \( a_3 \) (unicast from \( S \) to \( R_2 \)):** could be seen as a special case of broadcasting, \( a_1 \), in the sense that the erasure probability of the link between \( S \) and \( R_1 \) is one, i.e., \( \epsilon_1 = 1 \).

**Action \( a_4 \) (unicast from \( R_1 \) to \( R_2 \)):** On the one hand, if the number of dof at \( R_1 \) is equal to the common knowledge of \( R_1, R_2 \), then the coded packets created by \( R_1 \) cannot add a dof to \( R_2 \). On the other hand, if the number of dof at \( R_1 \) is greater than the common knowledge of \( R_1, R_2 \), then any coded packet created by \( R_1 \) adds one dof to the set of received packets by \( R_2 \) under our high field size assumption. Thus,

- If \( \epsilon_2 < M, i_1 > c, \) then \( P_{(i_1,i_2,c,t)} \rightarrow (i'_1,i'_2,c',t') = \epsilon_1(t)I_{(0,0,0,1)} + \epsilon_2(t)I_{(0,0,0,1)}. \)

- If \( \epsilon_2 < M, i_1 = c \) or \( i_2 = M, i_2 \neq M, \) then \( P_{(i_1,i_2,c,t)} \rightarrow (i'_1,i'_2,c',t') = I_{(0,0,0,0)}. \)

**Action \( a_5 \) (unicast from \( R_2 \) to \( R_1 \)):** is similar to \( a_4 \), replacing \( i_1 \) by \( i_2 \) and vice versa.

**Action \( a_6 \) (do not transmit):** if the system is not in an absorbing state, the time instance is increased by one while the dof of the receivers does not change, therefore, \( P_{(i_1,i_2,c,t)} \rightarrow (i'_1,i'_2,c',t') = I_{(0,0,0,0)}. \)

For a better understanding, we show a schematic of the possible transitions among states in Fig. 2. Assuming that the network state at time \( t_0 \) is \( s_1 = (i_1,i_2,c,t_0) \), there are 7 possible transitions depending on the action selected by the MDP, as shown in Fig. 2. The possible states to which state \( s_1 = (i_1,i_2,c,t_0) \) may transit to could be summarized as:

- \( s_{i+1} = (i_1,i_2,c,t_0+1) \), \( s_{i+2} = (i_1+1,i_2+1,c,t_0+1) \), \( s_{i+3} = (i_1,i_2+1,c,t_0+1) \), \( s_{i+4} = (i_1+1,i_2+1,c+1,t_0+1) \), \( s_{i+5} = (i_1+1,i_2+1,c+1,t_0+1) \), \( s_{i+6} = (i_1+1,i_2+1,c+1,t_0+1) \), \( s_{i+7} = (i_1+1,i_2+1,c+2,t_0+1) \).
For non-absorbing states, the probability of having a self-transition by using any action is zero since the time is always increasing by one and therefore, the network state is changing. The transitions are continued until the network is reached to an absorbing state, \( s_{abs} \). A cost function and an optimization algorithm is defined in the following to complete the MDP model.

Cost Function: we define a general, parametric cost function that could be applied to any wireless network with different requirements. To address the differences between broadcast and unicast communications, we assume that each unicast transmission incurs a cost of one unit while each broadcast transmission incurs \( \beta \geq 1 \) units of cost. For example, if we think of total energy consumption as the cost metric, the cost of each broadcast transmission could be calculated as \( E_{tx} + 2E_{rx} \), and the cost of each unicast transmission is \( E_{tx} + E_{rx} \), where \( E_{tx} \) is the energy consumed by transmitter, \( S_T \) to transmit data and \( E_{rx} \) is the energy consumed by each receiver to receive data. By defining the cost of unicast as one unit of cost, we conclude that the cost of broadcast is \( \beta = E_{tx} + 2E_{rx} \) that is clearly greater than one. Thus, we define the cost of a broadcast transmission to be greater or equal to that of a unicast transmission. Therefore, the cost of each transition from state \( s \) to state \( s' \) by choosing action \( a_j \) is defined as

\[
C(s, a_j, s') = \begin{cases} 
1, & \forall s \in S_T | s \neq (M, M, M, t), j = 2, \ldots, 5 \\
\beta, & \forall s \in S_T | s \neq (M, M, M, t), j = 1 \\
D, & \text{if } s = (M, M, M, t), j = 1, \ldots, 5 \\
D, & \forall s \in S_T | s \neq (M, M, M, t), j = 6 \\
0, & \text{if } s = (M, M, M, M, t), j = 6,
\end{cases} \tag{1}
\]

where \( C(s, a_j, s') \) is the cost of transition from state \( s \) to state \( s' \) by choosing action \( a_j \) and \( S_T \) is the set of all possible states. \( D \) is an arbitrary large number that is much greater than \( \beta \). By defining \( D \geq \beta \), we make sure that the MDP does not choose any one of the actions \( a_1, a_2, \ldots, a_5 \) if the system is in an absorbing state \( s = (M, M, M, t) \) and it chooses \( a_6 \) that has the minimum cost. This leads to having no additional cost once you reach the absorbing state.

Optimization Algorithm: we use the value iteration algorithm (Bellman equations) \( [32] \) to solve the optimization problem and to minimize the total cost of the transmission of \( M \) packets. A value function is defined as \( V_k : S_T \rightarrow \mathbb{R}^+ \) for iteration \( k \) that associates to each state \( s \) of \( S_T \) a lower bound on the minimal total cost \( V^*(s) \) that should be paid starting from that state. We can summarize the steps to find an optimal policy as \( V_{k+1}(s) \leftarrow \min_{a_j} \{ E(C(s, a_j, s') + \gamma V_k(s')) \} \), where \( V_0(s) = 0 \forall s \in S_T \), \( E(X) \) shows the expected value of \( X \), and the discount factor \( \gamma \in (0, 1) \) is used to ensure that the equation converges when \( k \) goes to infinity. This algorithm iterates until \( \max_s [V_{k+1}(s) - V_k(s)] < \delta \) is satisfied. \( \delta \) has a very small value greater than zero (e.g., 0.01) and shows the degree of optimality of the result. We set \( \delta = 0.0001 \) for our analysis. The algorithm chooses an action that minimizes the cost of transition into a new state, \( C(s, a_j, s') \), and the total cost that we pay starting from that state, \( V_k(s') \). Thus, the selected action by the algorithm takes into account both the current state and the future state of channels.

IV. MDP Analysis and Heuristics Development

Since the complexity of the MDP model for a time-varying scenario is high, we investigate the case of a time-invariant channel to extract meaningful rules to develop simple yet efficient heuristics. Section VI shows that the proposed heuristics have near-optimal performance by comparing to the full MDP solution for time-varying channels.

A. The MDP Model for Time-Invariant Channels

This could be seen as a special case of the time-varying channels, in the sense that the erasure probabilities of the channels remain constant over time. Therefore, each state is defined with three elements as \( s = (i_1, i_2, c) \), and we have a single absorbing state, \( s_{abs} = (M, M, M) \). The actions are defined as we defined for time-varying model. The transition probabilities and the cost function in this case can be easily extracted from the previous model for time-varying channels by removing the time element, \( t \), from all equations and replacing \( c_i(t) \) with \( c_i \). In case of time-invariant channels, the system always does a self-transition if the selected action is \( a_0 \). The value iteration algorithm is used to solve the optimization problem.

B. Analysis of the MDP Solution for Time-Invariant Channels

We analyze the relationship between the selected actions by the MDP and different characteristics of the network, e.g., the erasure probabilities, \( \beta \), and \( M \). The percentage of usage for each action \( a_j \) is calculated as percentage of usage = \( (N_{a_j} / N_S) \times 100\% \), where \( N_{a_j} \) is the number of states that MDP chooses to do action \( a_j \) and \( N_S \) is the total number of possible states. We neglect the role of action \( a_0 \), since it is selected only once and when the packet transmission process is finished. We also investigate the distribution of the selected actions for different instants of time to understand the time-evolution of the actions.

The effect of the erasure probabilities: we assume that \( M = 20 \), \( \epsilon_1 = 0.9 \), \( \epsilon_2 = 0.7 \), \( \beta = 1 \), and \( \epsilon_3 \) is changing over time. Fig. 3(a) shows the distribution of the selected actions by the MDP at the time that all 20 packets are received by both receivers. The x-axis of the graph shows the selected action by the MDP and the y-axis shows the percentage of usage of each receiver. We can see that the selected actions by the MDP depend on the erasure probabilities of the channels. For instance, if the channel between \( R_1, R_2 \) is very good (e.g., \( \epsilon_3 = 0.1 \)), only in 48% of the states the MDP chooses to do broadcast and in the remaining 52% two receivers are cooperating to help each other. By increasing the erasure probability of the channel between receivers, broadcast is shown to be used with high probability. For example, if \( \epsilon_3 = 0.7 \), the MDP chooses to do broadcast in 98% of states.

The effect of \( M \): we assume that the erasure probabilities are fixed at \( \epsilon_1 = 0.8 \), \( \epsilon_2 = 0.6 \), \( \epsilon_3 = 0.3 \) and \( \beta = 1 \). \( M \) is
We split the case of actions by the MDP for different values of receivers at the time of observation. Therefore, if the system age of selected actions by the MDP when both receivers get changing from 10 to 25. We evaluate the distribution of the selected actions in terms of the knowledge at the receivers: Fig. 4(a) and (b) shows that for both values of $\gamma$, $\epsilon_1=0.9$, $\epsilon_2=0.7$, $M=20$ and varying $\epsilon_3$, a) $\epsilon_1=0.8$, $\epsilon_2=0.6$, $\epsilon_3=0.3$ and varying $M$.

The time-evolution of the actions: to understand this evolution, we define a new parameter, $\gamma$, that shows the percentage of $M$ packets that were received by at least one of the receivers at the time of observation. Therefore, if the system is in state $s(t_1, t_2, c)$ at time $t$, then $\gamma$ is defined as $\gamma = (K/M) \times 100\%$. We evaluate the distribution of the selected actions by the MDP for different values of $\gamma$ ($0\% < \gamma \leq 100\%$). We split the case of $\gamma=100\%$ into two sub-cases: (a) $K=M$ but $i_1 \neq M$, $i_2 \neq M$ that is shown as $\gamma=100\%$ in our numerical results, and (b) $i_1 = M$ or $i_2 = M$ that are shown as $i_1 = M$ and $i_2 = M$, respectively. The parameters of the network are fixed at $\epsilon_1=0.8$, $\epsilon_2=0.6$, $\epsilon_3=0.3$, $M=20$ and a similar experiment is done for $\beta=1$ and $\beta=1.5$. Fig. 4(a) and (b) shows that for both values of $\beta$, if $\gamma=100\%$, the MDP chooses to do unicast from $R_1$ to $R_2$ and no broadcast is selected. Meaning that, the source can stop transmitting at $K=M$, and the two receivers can exchange their data. It is also seen that for 100% of the states where $i_1 = M$ (or $i_2 = M$), the MDP chooses one of the actions $a_4$ or $a_5$. This means that if one of the receivers collects $M$ dofs, the selected action for the current state of the network could be used until the end of the packet transmission process, assuming that the channel erasure probabilities are not changing very fast. In this specific setup, the MDP chooses to do unicast between receivers, since $\epsilon_3 < \epsilon_1, \epsilon_2$. Clearly, if one of the receivers has $M$ dofs and $\epsilon_3 > \epsilon_1, \epsilon_2$, the MDP does not choose $a_4$ or $a_5$, since the cost of packet transmission from source to receivers would be less than that for transmission between receivers. We repeated the same experiment for different values of $\beta$ and $\epsilon_1$. Based on our observations we conclude that if $\frac{\beta}{(1-\epsilon_1)+\epsilon_2}$ is less than $\frac{1}{1-\epsilon_1}$ and $\frac{1}{1-\epsilon_2}$, the MDP chooses broadcast, and otherwise no broadcast action is selected. $\frac{1}{1-\epsilon_1}$ represents the cost of successful reception of one packet either by $R_1$ or $R_2$, per broadcast transmission and $\frac{1}{1-\epsilon_2}$ represents the cost of successful reception of the packet that is sent by unicast transmission from source to receiver $R_i$. 
C. Lessons Learned From the MDP Solution for Time-Invariant Channels

For a network of three nodes with time-invariant channels, we extract a general packet transmission policy that can be used to achieve close to optimal performance with respect to minimizing the cost of packet transmission under the GS assumption. We define the expected cost of successful transmission of one packet from a sender to a receiver by using action $a_j$ as

$$C(a_j) = \left[ \frac{\beta}{(1 - \epsilon_1) + (1 - \epsilon_2)} \right] \times I_{(j=1)} + \frac{1}{1 - \epsilon_1} \times I_{(j=2)}$$

$$+ \frac{1}{1 - \epsilon_2} \times I_{(j=3)} + \frac{1}{1 - \epsilon_3} \times \left[ I_{(j=4)} + I_{(j=5)} \right], \quad (2)$$

where $I_{(x \in X)}$ is an indicator function, $\beta$ is the cost of one broadcast transmission, and $\epsilon_k$ represents the erasure probability of the channel $k$. From the MDP solution, we extract the relationship between $C(a_j)$ and the optimum selected action, $a^*$. The obtained packet transmission policy is summarized in Algorithm 1. For example, if $K < M$ and the action that minimizes $C(a_j)$ for $a_j \in \{a_1, a_2, a_3\}$ is $a_1$, the selected action would be $a_1$. On the other hand, if $R_1$ has $M$ dof while $R_2$ has less than $M$ dof, the selected action would be either $a_3$ or $a_4$ depending on their cost. By looking at Algorithm 1, we conclude that the transmission policy is changed at two main points: a) when $K = M$, and b) when the dof of one of the receivers is equal to $M$, i.e., $i_1 = M$ or $i_2 = M$. This means that we only need to know the state of the network at the time that one of these main points is happening to be able to change the packet transmission policy.

V. PROPOSED HEURISTICS

Since real world systems cannot provide a full knowledge of the state of the network without incurring in large signaling overhead, we now focus on policies inspired by our MDP model, but that require limited signaling. Thus, the state information of the system is not known at every time slot. We propose two heuristics, Minimum-Feedback (MF) and Intermediate-Feedback (IF), using only a limited number of feedback packets from each receiver per generation of $M$ packets. We will refer to $a^*$ as the selected action and $a'$ as the set of solutions that represent the minimum cost actions for a given subset of actions. In both heuristics, it is assumed that the erasure probabilities of the channels are estimated at the beginning of each generation transmission.

A. Minimum-Feedback (MF)

In this heuristic, a receiver can start sending recoded packets to its neighbor only when it receives $M$ dof and is able to decode the original packets. Therefore, a receiver sends feedback only if $M$ dof are received successfully. The packet transmission process is divided into two phases. First, the source selects one of the three actions $a_1, a_2, a_3$ that has the minimum cost to start the transmission with. To this end, it computes $a' = \arg \min_{a \in \{a_1, a_2, a_3\}} C(a)$, where $C(a)$ is calculated from (2) and depends on the erasure probabilities of the channels. Source starts sending RLNC packets using $a' = a'$ and continues sending. If $a' = \{a_1, a_2\}$ then $a^* = a_1$, i.e., broadcast is favored over unicast when they represent the same cost. Second, upon receiving feedback from either $R_1$ or $R_2$, the second phase makes the following decisions:

- If the received feedback has come from $R_1$, then the new selected action is $a^* = \arg \min_{a \in \{a_3, a_4\}} C(a)$.
- If the received feedback has come from $R_2$, then the new selected action is $a^* = \arg \min_{a \in \{a_2, a_5\}} C(a)$.
- If the received feedback has come from both $R_1, R_2$, the packet transmission is stopped.

Algorithm 1 Packet Transmission Policy Extracted from the MDP Model

$$s = (i_1, i_2, c) = \text{network state at time } t; K = i_1 + i_2 - c$$

$$M = \text{number of packets}$$

$$C(a_j) = \text{cost of action } a_j \text{ that is calculated using (2)};$$

$$a^* = \text{selected action at time } t$$

if $i_1, i_2, K < M$ then

$a^* = \arg \min_{a \in \{a_1, a_2, a_3\}} C(a)$

else if $i_1, i_2 < M$ and $K = M$ then

$a^* = \arg \min_{a \in \{a_1, a_2, a_3, a_4\}} C(a)$

else if $i_1 = M$ and $i_2 < M$ then

$a^* = \arg \min_{a \in \{a_1, a_4\}} C(a)$

else if $i_1 < M$ and $i_2 = M$ then

$a^* = \arg \min_{a \in \{a_2, a_5\}} C(a)$

else if $i_1 = i_2 = M$ then

$a^* = a_6$

else if

B. Intermediate-Feedback (IF)

Considering the rules in Algorithm 1, we propose IF heuristic which allows receivers to start exchanging recoded packets among each other when the total knowledge of the received packets by two receivers is equal to $M$, i.e., $K = M$. Therefore, cooperation between receivers may be started earlier compared to the MF heuristic. The IF heuristic uses feedback not only when a receiver gets $M$ dof successfully, but also when $K = M$. Sending feedback whenever $K = M$ needs some level of signaling among receivers to exchange the information about the number of packets received by two receivers. To minimize the number of signaling, we calculate the expected number of RLNC packets ($M$) that the source has to broadcast to expect that the total number of packets received by two receivers is equal to $M$. Under our high field size assumption for RLNC, this leads to have $K = M$. We calculate $M$ as $M = M \times \frac{1}{1 - \epsilon_1 \epsilon_2}$, where $1 - \epsilon_1 \epsilon_2$ is the probability of adding one degree of freedom to the set of packets received by two receivers.
per broadcast transmission. Now each receiver can calculate the expected number of packets it needs to receive such that the expected total number of packets received by the two receivers at that time is equal to \( M \). For \( R_1 \), this means \( ms_1 = M \times (1 - e_1) \) and for \( R_2 \), this means \( ms_2 = M \times (1 - e_2) \).

Before starting packet transmission, \( R_1, R_2 \) calculate \( ms_1, ms_2 \), respectively, as the milestone they need to reach before starting exchanging data with each other. We can summarize the process of packet transmission in two phases. Phase 1 is similar to the first phase of the MF. Clearly, if the selected action for the phase 1 is not broadcast, the previous scheme of sending feedback can be used without the need for signaling among receivers. Because in that case, only one of the receivers is receiving feedback in phase 1. If the selected action in phase 1 is broadcast, the first receiver to reach its milestone sends a feedback stating the dimensions they possess. For example, after getting \( ms_1 \) packets at \( R_1 \), it will signal to both \( R_2 \) and \( S \) that it has reached that specific milestone and sends extra information in terms of the dimensions it has. To reduce the size of the feedback packets, we can use the compressed format of feedback that was proposed in [33] to convey the required information instead of sending all dimensions explicitly. Then, the other receiver (in this case \( R_2 \)) can determine if \( R_1 \) and \( R_2 \) have enough dimensions jointly. If they do, then \( R_2 \) sends a feedback packet that informs \( S \) to stop phase 1 of the process and informs \( R_1 \) that they can begin the exchange process. Otherwise, \( R_2 \) will wait until getting what is missing and then feedback is sent. This way, the number of signaling packets sent is small, although the scheme is suboptimal as \( R_1 \) will have accumulated additional coded packets by that time. Instead of waiting to get the missing dofs, \( R_2 \) can send a signal to \( R_1 \) telling the number of the coded packets that it needs to reach the milestone. By having this information, both \( R_1, R_2 \) can recalculate a new milestone as before. Once the first has reached the new milestone, it will feedback as before. This approach trades-off accuracy with the use of feedback. A similar process occurs if \( R_2 \) is the first to reach the milestone. Phase 2 is started upon receiving feedback that ACKs one of the three cases, \( i_1 = M \) or \( i_2 = M \) or \( K = M \). The source calculates \( d' = \arg \min_{a \in \{a_1, a_2, a_3, a_4, a_5\}} C(a) \), and the decision on the new action is made based on what the feedback acknowledges and \( d' \). If the feedback acknowledges

- \( K = M \), while \( i_1 \neq M, i_2 \neq M \), and \( d' = \{a_4, a_5\} \), then \( a^* = a_4 \) for the case of \( i_1 > i_2 \) and \( a^* = a_5 \) otherwise. If \( a' \neq \{a_4, a_5\} \), then the action selected by the source is one of the three actions \( a_1 - a_3 \) that minimizes the transmission cost, i.e., \( a^* = \arg \min_{a \in \{a_1, a_2, a_3\}} C(a) \).

- \( i_1 = M \), then the selected action would be \( a^* = \arg \min_{a \in \{a_3, a_4\}} C(a) \).

- \( i_2 = M \), then the selected action would be \( a^* = \arg \min_{a \in \{a_2, a_3\}} C(a) \).

- \( i_1 = i_2 = M \), then \( a^* = a_6 \) and the packet transmission is stopped.

If the channel between receivers is not symmetric and the feedback acknowledges \( K = M \) while \( i_1 \neq M, i_2 \neq M \), the selected action would be \( a^* = a_4 \) if \( a' = a_4 \), and \( a^* = a_5 \) if \( a' = a_5 \). We emphasize that for both MF and IF heuristics, the received dofs determine when to start cooperation, while the selected action depends on the erasure probabilities of channels.

In highly dynamic environments, a receiver can send a wake up message to ask source to restart transmission if it does not receive anything from its neighbor after a given waiting time, \( T_{wait} \).

C. Generalization of the Proposed Heuristics

For a network with more than two receivers, the MDP model requires high computational complexity, because the number of meaningful states increases exponentially with the number of receivers. For such networks, the state definition needs to incorporate the dof of the received packets by each receiver, and also the common dofs between every pair of the receivers, that leads to have a complex model. Therefore, instead of finding the optimal transmission policies, we extend the proposed heuristics for a network of one source and \( N \) receivers. Our generalized heuristics include one extra phase that is executed before the packet transmission is started. In this phase, \( N \) receivers are divided into \( N/2 \) clusters such that each cluster includes exactly two receivers. The clustering of receivers could be performed by the source, and depending on different metrics, such as distance between receivers. We assume that the receivers are already clustered. A similar approach as the MF and IF heuristics is used for packet transmission toward the receivers of each cluster. The first phase of both heuristics is started by broadcasting RLNC packets from the source to the receivers. The source stops transmission whenever it receives feedback from at least one receiver of each cluster.

Generalized MF: when a receiver collects \( M \) dof, it sends a feedback to the source and at the same time, starts transmitting recoded RLNC packets to its neighbor inside the same cluster. Intra-cluster transmissions are continued until all receivers have \( M \) dof. Since the erasure probabilities of the channels are changing over time, we may need to ask the source to re-start sending packets.

Generalized IF: similar to the IF, we define a milestone for receivers in each cluster. To minimize the cost of signaling, upon a receiver reaching its milestone, it starts sending recoded RLNC packets to its neighbor in the cluster. Thus, cooperation between receivers in a cluster is started while the source is still transmitting. Finally, each receiver sends a feedback when it has \( M \) dof. As in the generalized MF, we may need to ask the source to wake-up and restart transmission.

VI. HEURISTICS EVALUATION

We define two set-ups to analyze the performance of the proposed heuristics in a time-varying scenario in terms of throughput and reliability. As a relevant example of time-varying environments, we consider an I2V communication in a highway scenario as our first set-up, which reflects the gain of our heuristics in a highly dynamic environment. The second set-up is a three-node wireless test-bed that reflects the performance of our proposed heuristics in a less dynamic
environment. Our analysis is divided into two parts. First, an analysis for small number of packets, where we compare the performance of the proposed heuristics with the performance of the optimal MDP solution in terms of expected completion cost. This is to validate our claim about near-optimality of the proposed heuristics for time-varying environment. The optimality of these heuristics for time-invariant scenarios was evaluated in [28]. Second, an analysis for large number of packets, where we use mean-field analysis to show the gain of the proposed heuristics compared with RLNC broadcast in terms of expected completion cost and reliability. Assuming that we are aiming to transmit a set of packets in 

\[ N \]

packets from a fixed access point (AP) representing a source (S), to two moving vehicles \[ V_1, V_2 \]. We use a realistic model of vehicle distances, and their respective traveling speeds, based on stereoscopic aerial photography as proposed in [34]. Authors of [34] show that the distribution of inter-vehicle spacing can be well fitted with an exponential probability distribution with mean 51.58 m and the speed distribution of vehicles is well approximated by a normal probability distribution with mean 106.98 km/h and standard deviation 21.09 km/h. We assume that, both vehicles are moving in the same direction and \[ \epsilon_1(t) \], \[ \epsilon_2(t) \] show their speeds that are selected randomly and according to the normal distribution. The initial coordinates of the AP and the two vehicles that are selected randomly and according to the normal distribution. A time-slotted system is considered with one transmission per time slot. We use a Rayleigh fading channel model and BPSK modulation. According to [30] the average bit error rate (BER) of BPSK modulation is

\[
\text{BER} = \frac{1}{2} \left( 1 - \sqrt{\frac{\gamma_b}{1 + \gamma_b}} \right),
\]

where \( \gamma_b \) represents the average SNR per bit that is calculated as \( \gamma_b = E_b/N_0 \). \( E_b \) is energy per bit and \( N_0 \) is the noise power. Since the vehicles are moving, the packet loss/erasure probability changes over time. To find a relationship between the distance of transmitter-receiver and the BER, we assume that \( \gamma_b \) is known at a reference distance \( (d_{ref}) \) and is shown as \( \gamma_{ref} \). By using the log-distance path-loss model [31], we can find a relationship between the path loss \( (P_L) \), the average received power, and the distance between transmitter and receiver. In other words, \( P_L = P_{L_0} + 10\alpha \log \left( \frac{d}{d_{ref}} \right) + X_\gamma \), where \( P_{L_0} \) is the path-loss at a reference distance \( (d_{ref}) \) in dB, \( \alpha \) is the path loss exponent, and \( X_\gamma \) is a random variable reflecting the attenuation caused by fading in dB. By calculating the expected value of path loss for an arbitrary distance \( d \), we will have \( P_L = \frac{P_{L_0}}{P_{R_0}} = k \left( \frac{d}{d_{ref}} \right)^\alpha \),

where \( k \) is a constant value. Assuming that the transmission power is constant for all nodes, the received power by a receiver at arbitrary distance \( d \) from the transmitter is calculated as \( P_R = P_{R_0} \left( \frac{d_{ref}}{d} \right)^\alpha \), where \( P_{R_0} \) is the received power by the receiver at \( d_{ref} \) and \( P_R \) is the received power by the receiver at distance \( d \). Therefore, the average SNR per bit \( (\gamma_b) \) for a pair of transmitter-receiver with distance \( d \) is calculated as \( \gamma_b = \gamma_{ref} \times \left( \frac{d_{ref}}{d} \right)^\alpha \). We set \( \gamma_{ref} = 25 \) dB, \( d_{ref} = 250 \) m, \( \alpha = 2 \) for our numerical analysis. By substituting \( \gamma_b \) into (3), we can calculate the BER for a pair of transmitter-receiver with distance \( d \).

**A. First Network Set-Up (I2V)**

Considering the network defined in Fig. 5(a), we want to calculate the cost of transmitting \( M \) packets from a fixed access point (AP) representing a source (S), to two moving vehicles \( V_1, V_2 \). We use a realistic model of vehicle distances, and their respective traveling speeds, based on stereoscopic aerial photography as proposed in [34]. Authors of [34] show that the distribution of inter-vehicle spacing can be well fitted with an exponential probability distribution with mean 51.58 m and the speed distribution of vehicles is well approximated by a normal probability distribution with mean 106.98 km/h and standard deviation 21.09 km/h. We assume that, both vehicles are moving in the same direction and \( \epsilon_1(t), \epsilon_2(t) \) show their speeds that are selected randomly and according to the normal distribution. The initial coordinates of the AP and the two vehicles that are selected randomly and according to the normal distribution. A time-slotted system is considered with one transmission per time slot. We use a Rayleigh fading channel model and BPSK modulation. According to [30] the average bit error rate (BER) of BPSK modulation is

\[
\text{BER} = \frac{1}{2} \left( 1 - \sqrt{\frac{\gamma_b}{1 + \gamma_b}} \right),
\]

where \( \gamma_b \) represents the average SNR per bit that is calculated as \( \gamma_b = E_b/N_0 \). \( E_b \) is energy per bit and \( N_0 \) is the noise power. Since the vehicles are moving, the packet loss/erasure probability changes over time. To find a relationship between the distance of transmitter-receiver and the BER, we assume that \( \gamma_b \) is known at a reference distance \( (d_{ref}) \) and is shown as \( \gamma_{ref} \). By using the log-distance path-loss model [31], we can find a relationship between the path loss \( (P_L) \), the average received power, and the distance between transmitter and receiver. In other words, \( P_L = P_{L_0} + 10\alpha \log \left( \frac{d}{d_{ref}} \right) + X_\gamma \), where \( P_{L_0} \) is the path-loss at a reference distance \( (d_{ref}) \) in dB, \( \alpha \) is the path loss exponent, and \( X_\gamma \) is a random variable reflecting the attenuation caused by fading in dB. By calculating the expected value of path loss for an arbitrary distance \( d \), we will have \( P_L = \frac{P_{L_0}}{P_{R_0}} = k \left( \frac{d}{d_{ref}} \right)^\alpha \),

where \( k \) is a constant value. Assuming that the transmission power is constant for all nodes, the received power by a receiver at arbitrary distance \( d \) from the transmitter is calculated as \( P_R = P_{R_0} \left( \frac{d_{ref}}{d} \right)^\alpha \), where \( P_{R_0} \) is the received power by the receiver at \( d_{ref} \) and \( P_R \) is the received power by the receiver at distance \( d \). Therefore, the average SNR per bit \( (\gamma_b) \) for a pair of transmitter-receiver with distance \( d \) is calculated as \( \gamma_b = \gamma_{ref} \times \left( \frac{d_{ref}}{d} \right)^\alpha \). We set \( \gamma_{ref} = 25 \) dB, \( d_{ref} = 250 \) m, \( \alpha = 2 \) for our numerical analysis. By substituting \( \gamma_b \) into (3), we can calculate the BER for a pair of transmitter-receiver with distance \( d \).

**Performance evaluation for small M:** assuming the network in Fig. 5(a), we compare the performance of the MDP, and the proposed heuristics in terms of expected completion time for \( M = 5, \beta = 1, T = 100 \). A time slot is 0.1 s. We repeat the same experiment for 200 different random pairs of \( \Delta, \epsilon_1, \epsilon_2 \) that are selected randomly according to the model we explained before. We calculate \( X = \frac{CT_{MDP}}{CT_{Heu}} \) as the ratio between the expected completion times of packet transmission by using a heuristic and the MDP solution. Fig. 6(a) shows the Cumulative Distribution Function (CDF) of \( X \) for 200 random tests. It is seen that both heuristics can perform close to the optimal. For example, in case of the IF and for \( X = 1.25 \), the calculated CDF is 0.95. This means the probability that the completion time by using the IF is within 1 dB of the optimal completion time is 0.95. Also in case of the MF, the probability of being
within 1.3 dB of the optimal solution is 0.95. These observations state that the proposed heuristics can provide a close-to-optimal performance in highly dynamic environment.

Performance evaluation for large $M$: considering the network defined in Fig. 5(a), we compare the performance of the IF, the MF, and RLNC broadcasting for $50 \leq M \leq 3500$ and $\beta \geq 1$ in terms of the total cost of packet transmission and the reliability. The results are shown in Fig. 7. We use $T = 5000$ time slots, where each time slot is 4 ms. We repeat the same experiment for 1000 randomly selected pairs of vehicle speeds and inter-vehicle distances and our results show the average of these samples. Note that by increasing the generation size up to 3500, we just want to see where the performance of our heuristics breaks, and obviously we are not advocating for using generations of this size in practice. The reason is that the overhead per packet used to transmit the coding coefficients increases dramatically, e.g., 3500 bytes for $\text{GF}(2^8)$ according to [35], and the computational complexity would be high. Therefore in practice, we divide a large generation into smaller chunks of packets and use our cooperative approach for each chunk of data. For $M = 50$, $\beta = 1$, the IF and MF heuristics, respectively, reduce the completion time by a factor of 2 and 1.75, compared to RLNC broadcasting [see Fig. 7(a)]. Although for larger number of packets, e.g., $M \geq 750$, the gain of heuristics is reduced, the IF heuristic still provides a gain in terms of the percentage of reliability. The reason of having less gain in case of larger generation size is that the period of time that we are running using a channel estimate that we have made at the beginning of the transmission of the generation is increased. Since our heuristics make decision without knowing the future state of the channels and only based on their current state, the larger the generation size becomes, the less accurate the estimate becomes as time progresses. Ultimately, this results in a lower gain. Fig. 7(b) shows that in case of the IF, the percentage of reliability for transmitting 2500 packets is 63%, while in case of RLNC broadcasting it is only 16%. Therefore, in this case, the IF increases the percentage of reliability by a factor of 3.98 with respect to RLNC broadcasting. For $\beta > 1$, we calculate the gain of heuristics with respect to RLNC broadcasting (see Fig. 8). The network characteristics are defined as we defined for $\beta = 1$. Fig. 8 shows that the MF achieves less gain compared with the IF, while the stability of the gain achieved by the MF is more compared to that of the IF by increasing the values of $\beta$. This is because the packet transmission between receivers in MF heuristics is started later than IF heuristics, so in case of the MF having a lower cost for the transmission between receivers has a smaller impact on the total cost compared with the IF. For example, if $M = 50$ and $\beta$ is changed from 1 to 5, the gain of the IF increases from 1.75 to 2.25, while the gain of the MF is changed from 1.65 to 1.8.
We also evaluated the gain of the generalized versions of heuristics for the network defined in Fig. 5(b), where four receivers are divided into two clusters, $C_1$, $C_2$. The initial positions of the source and four vehicles are shown in Fig. 5(b). $\Delta_1$, $\Delta_2$, $\Delta_3$ are three randomly selected inter vehicle distances. The speed of each vehicle is selected randomly as we explained before. A time horizon of $T = 10000$ time slots is considered, where each time slot is 0.01 sec. Fig. 9 shows the results of this experiment for 1000 random pairs of vehicle speeds and inter-vehicle distances. We see that the IF and MF heuristics are able to reduce the completion time by a factor of respectively, 1.41, 1.26 with respect to RLNC broadcast. Fig. 9(b) illustrates that the reliability of transmitting 800 packets in 10 000 time slots for RLNC broadcast is zero, while in case of IF and MF, it is respectively, 93%, 53%.

**B. Second Network Set-Up (Raspberry Pi Test-Bed)**

We use a wireless network coding testbed implemented on Raspberry Pi’s at Aalborg University to measure packet losses over time. Detailed information about this test-bed could be found in [36]. The goal is to collect statistics about the channel loss as a function of time and to use these measurements to compare the performance of our heuristics versus RLNC broadcasting. All three nodes are fixed during our tests and they are located inside two buildings, as shown in Fig. 10(a), but the channels suffer wide variations over time. Two receivers are located inside the same room and the source is located inside a room in another building, as shown in Fig. 10(a). $R_1$ has line-of-sight (LOS) to the source, while $R_2$ does not have LOS to the source but there is LOS between $R_1$, $R_2$. To ensure that our measurements keep track of the correlation between erasure probability of the channels, a node transmits for 10 s while the others record the packet losses. After the 10 s, another node transmits and so on. Each sender broadcasts packets of 1 KB every 0.1 s. We record the sequence number of the received packets by each receiver. A moving average filter with window size $w$ is applied to the collected data set to model the erasure probability of the channels. We do not make any assumption about having symmetric channel between receivers, and instead we use the real measurement for the loss probability of the channel between nodes. Therefore, we may have $C(a_4) \neq C(a_5)$. Fig. 10(b)–(e) shows the erasure probabilities obtained by our measurements for $T = 1000$, $w = 5$, and $w = 50$.

**Performance evaluation for small M:** considering the second set-up, we compare the performance of the MDP solution and the heuristics for $M = 5$. Fig. 6(b) shows the completion time for the optimal MDP solution and the proposed heuristics. We see that the performance of the IF is very close to the performance of the MDP solution for all values of $w$. In case of the MF, its performance is getting closer to the optimal MDP solution performance by increasing the window size. Meaning that the MF has better performance in less dynamic environments.
Fig. 10. (a) Deployment of nodes in the wireless test-bed, (b) erasure prob. between $S$, $R_1$, (c) erasure prob. between $s$, $R_2$, (d) erasure prob. between $R_1$, $R_2$, (e) erasure prob. between $R_2$, $R_1$.

Fig. 11. Comparison between IF, MF, and RLNC broadcasting for time-varying scenario, second set-up, and $w=5$; (a) completion time comparison, (b) reliability comparison.

Performance evaluation for large $M$: Fig. 11 shows the mean of completion time and the reliability of the two heuristics and RLNC broadcasting for the second setup. We assume $w=5$ and $\beta=1$. For $M=3000$, the percentage of reliability of the IF is 46%, while for RLNC broadcasting it is only 15%. Meaning that the IF increases the percentage of reliability by a factor of 3 compared with RLNC broadcasting. For smaller $M$, we see that both IF and MF heuristics decrease the expected completion time with respect to RLNC broadcasting. For example, for $M=50$ the expected completion time by using the IF, the MF and the RLNC broadcasting are respectively, 115.22, 127.01, and 161.71. This leads to have a 40% gain for the IF and a 27% gain for the MF in terms of the expected completion time.

VII. CONCLUSION

The problem of minimizing the total cost of transmitting $M$ packets from a source to two receivers has been solved for a time-varying wireless network by taking advantage of network-coded cooperation between receivers. We modeled the problem as an MDP problem. Inspired by the optimal MDP solution, two simple yet powerful heuristics have been proposed that are shown to have close-to-optimal performance in both time-varying and time-invariant wireless environments. A comparison with RLNC broadcasting reveals that the proposed heuristics are able to decrease the completion time by a factor of 2 and increase the percentage of reliability by a factor of 3.98. Although our analysis was done for a network of three nodes, the results of this analysis have significant impact on improving the design of routing protocols in wireless networks. In fact, the proposed heuristics could be applied to the traditional multi-hop protocols to improve their performance in terms of the cost of packet transmission per single hop. As for the future works, we can consider other channel models rather than the general erasure model, e.g., Gilbert Elliot model, to provide more structure and, potentially, simpler solutions. Other directions for future work may include the case of multiple unicasts, index coding and finding optimal code structures.
REFERENCES


Hana Khamfroush received the B.Sc. degree in electrical engineering from Urmia University, Urmia, Iran, in 2005 and the M.Sc. degree in telecommunications engineering from the University of Yazd, Yazd, Iran, in 2009. She is currently working toward the Ph.D. degree in telecommunications engineering with the University of Porto, Porto, Portugal. She is also a Researcher with the Networking and Information Processing Group (NIP), Institute for Telecommunications, Porto. Her research interests include network coding, wireless communications, and communication networks. Ms. Khamfroush was a recipient of a Ph.D. scholarship from the Portuguese Foundation for Science and Technology.

Daniel E. Lucani (S’04–M’10) received the B.S. and M.S. degrees in electronics engineering from the Universidad Simon Bolivar, Caracas, Venezuela, in 2005 and 2006, respectively, and the Ph.D. degree in electrical engineering from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 2010. He is currently an Associate Professor with the Department of Electronic Systems, Aalborg University (AAU), Aalborg, Denmark. He was an Assistant Professor with the University of Porto, Porto, Portugal, from 2010 to 2013, before joining AAU. His research focuses on communications, network theory, and network coding theory and applications.

João Barros (S’98–M’04–SM’11) received the undergraduate education in electrical and computer engineering from the University of Porto, Porto, Portugal, and the Universität Karlsruhe, Karlsruhe, Germany, and the Ph.D. degree in electrical engineering and information technology from the Technische Universität München, München, Germany. He is a Full Professor of electrical and computer engineering at the University of Porto and the Founding Director of the Institute for Telecommunications, Porto. He also teaches at the Porto Business School and co-founded two recent start-ups, Streambolico and Veniam, commercializing wireless video and vehicular communications technology, respectively.

Peyman Pahlevani received the B.S. degree in information technology from the Institute for Advanced Studies in Basic Science, Zanjan, Iran, in 2008 and the M.Sc. degree in computer networks from the University of Yazd, Yazd, Iran, in 2010. He is currently working toward the Ph.D. degree with the Department of Electronic Systems, Aalborg University, Aalborg, Denmark. His research interests include wireless networks and network coding theory and its applications.