Multirate Packet Delivery in Heterogeneous Broadcast Networks

Mohammad A. Zarrabian, Foroogh S. Tabataba, Member, IEEE, Sina Molavipour, and Parastoo Sadeghi, Senior Member, IEEE

Abstract—In this paper, we study the problem of multirate packet delivery in heterogeneous packet erasure broadcast networks. The technical challenge is to enable users receive packets at different rates, as dictated by the quality of their individual channel. We present a new analytical framework for characterizing the delivery rate and delivery delay performance of a previously proposed non-block-based network coding scheme in the literature. This scheme was studied in homogeneous network settings. We show for the first time, via new theoretical analysis and simulations that it can actually achieve multirate packet delivery. Using acknowledgments from each user, we show that the user with the highest link capacity achieves the maximum possible throughput. Also, a non-zero packet delivery rate is possible for other users, and the delivery rate depends on the difference between the packet arrival rate at the sender and the link capacity of each user. The accuracy of our analytical framework is confirmed by comparing the results with simulations for different settings of packet arrival rate at the sender and link capacities.

Index Terms—Broadcasting, heterogeneous networks, network coding, multirate packet delivery, delivery delay.

I. INTRODUCTION

NOWADAYS with the pervasive development of wireless communication networks, the real-time applications such as broadcast multimedia and video streaming with high quality are in high demand [1]–[4]. In wireless broadcast streaming, an identical message is intended to be delivered in the form of ordered data packets to each user. An appropriate model for such a system is a packet erasure model, where a single sender aims to deliver ordered data packets to some users over independent wireless packet erasure channels. When all the links, connecting to all the users have identical erasure probabilities, the network is called homogeneous, otherwise it is called heterogeneous [5]. In both homogeneous and heterogeneous networks, packet erasure events occur independently among users. Therefore, at any given time, the packets already received and still wanted at each user vary. This makes the design of transmission schemes challenging.

An efficient method to accommodate multiple users’ demands, achieving high throughput and decreasing delay is network coding [6]–[9]. This method has been studied in different types of networks such as multicast and unicast networks [9]–[13], multiple access and relay networks [14]–[17] as it is used to exploit the broadcast property of wireless channels and also to combat the packet erasure problem in networks [18]–[23]. Most of the network coding methods are block-based [24]–[28], where a block of packets is considered and a linear combination of the packets is constructed as the transmitted packet. In some cases, the transmitted packet is encoded in such a way that it provides new information for the most possible number of users. It is called innovation guarantee property. However, since the decoding of such methods depends on the reception of the whole block, it may cause long delay in real-time packet streaming. Besides, in heterogeneous networks, another challenge is to provide packets for each user with respect to the quality of its channel, which is known as multirate packet delivery [5]. In block-based codes, encoded blocks with the length of \( n \) packets convey \( k \leq n \) packets of information and the rate of encoding would be \( k/n \). In heterogeneous networks, the users with the link capacity lower than the encoding rate (weak users) cannot decode the packets and if the sender decreases the encoding rate, users with the higher bandwidth will experience long delays and their delivery rates decrease to the rate of the weak ones. To achieve multirate packet delivery in block-based codes, the sender must change the encoding rate which is inefficient and hard to implement when the number of users increases [29].

For the purpose of real-time broadcasting, a coding method is preferred that would allow intermediate decoding of the packets prior to the reception of the whole block [30]–[32]. In these applications, average per packet delay is more important and due to the necessity of applying packets in-order, the performance of the system can be mainly measured by the delivery rate (which is proportional to the throughput) and delivery delay rather than the decoding rate and decoding delay. A packet is said to be delivered to a user when all the previous packets with lower indices in the user’s buffer have been decoded.

In addition to block-based codes, there also exist non-block-based network codes [33]–[40]. Similar to [30], [32], the goal in papers [34]–[38] is to increase the chance of decoding the packets prior to receiving all the information sent from the sender. They use the users’ feedback to determine which packets...
should be encoded together and transmitted. For the purpose of real-time streaming, an ARQ (automatic repeat request) online network coding has been introduced in [34] that combines the benefits of ARQ and network coding for broadcast networks. It achieves the maximum throughput of one hop multicast networks but suffers from large delay for weak users. For the delay mitigation problem, some solutions have been proposed in [33], [35], [36]. In [35], a non-block-based algorithm for three users has been suggested and then it has been improved for any number of users in [36]. However, both [35] and [36] only considered homogeneous networks. The authors in [36] conjectured that their approach is asymptotically optimal in the decoding delay and delivery delay in the limit when the packet arrival rate at the sender approaches the capacity (or the load factor approaches one). A non-asymptotic analysis (with respect to the load factor) of the works in [34] and [36] has been done in [37], [38], [41] for homogeneous networks. In [41], the authors have shown that the coding scheme of [36] is more practical than the one in [34], because it provides more opportunities to transmit uncoded packets, which results in better decoding for the users. Based on the observations of [41], a dynamic rate adaptation scheme was proposed in [37] to improve system throughput and delay.

A. Approach and Contributions

In this paper, we provide an in-depth modeling and analysis of multirate packet delivery of non-block-based network coding of [36] in heterogeneous networks. In the following we summarize our contributions and highlight distinctions with earlier works.

- We demonstrate that the coding scheme proposed in [36], can indeed achieve multirate packet delivery in heterogeneous broadcast networks. Ensuring innovation guarantee property for all users and instantaneous delivery for some, this coding scheme achieves maximum possible throughput for the user with the highest link capacity and a non-zero delivery rate for the others. The system model and the coding scheme is presented in Section II.

- Then, an analysis of the delivery rate of the coding scheme is proposed. Although the analysis of non-block-based codes is a challenging problem, using a reasonable approximation, we develop a tractable model to estimate the delivery rate. To validate the analysis, our results are compared with extensive simulations for the different settings of the packet arrival rate and the channel capacities. Due to the existence of a transmission queue, this coding scheme is not deterministic as the one in [38] (which also uses a different non-block-based network coding scheme) and the demand of the users is restricted by the packet arrival rate at the sender. This difference underpins our analysis and the model is completely different from the ones used in [38]. The analytical model of the delivery rate is discussed in Section III.

- Finally, we analyze the delivery delay of the system based on a different definition from the previous works. In the literature, the delivery delay of a packet have been considered as the time when the packet enters the transmission queue to the time it is delivered. However, we define the delivery delay as the time between the first request of a packet and its delivery. We believe this new definition is more suitable for the heterogeneous case and better characterizes the delay of the users. Consequently, it results in a simple closed-form delivery delay. To estimate and calculate the delivery delay, our delivery rate model is used for different cases, which is shown the consistency of our assumptions and approximation. Furthermore, the accuracy of the delivery rate and delivery delay analysis is confirmed by comparing the results with simulations. The delivery delay analysis is described in Section IV, the simulation results in Section V and the paper is concluded in Section VI.

II. System Model

A single transmitter aims to broadcast a set of packets $p_1, p_2, \ldots, p_n$ ($n$ arbitrary large) to $\nu$ users $U_i$ ($1 \leq i \leq \nu$) via heterogeneous broadcast packet erasure channels. Here, a time-slotted scheme ($t = 1, 2, \ldots$) is assumed in which the sender uses linear network coding to construct the encoded packet for transmitting one coded packet in each time slot. Packets enter an infinite-length buffer, or the transmission queue at the sender according to a Bernoulli process of rate $\lambda$. Assuming independent channels between the transmitter and the users, each packet is correctly received by a user $U_i$ with a probability $c_i$ which is called the channel capacity, i.e., packets are erased in each channel independently with the probability of $\bar{c}_i = 1 - c_i$. Due to the heterogeneous property of the channels, the capacities are unique. Hence, without loss of generality, it is possible to assume that $c_1 > c_2 > \cdots > c_\nu$, which is shown by the vector $e = [c_1, c_2, \ldots, c_\nu]$. Here, we indicate the strength of a user by its link capacity. We refer to $U_1$ as the strongest user (i.e., with the highest link capacity). The purpose of the system is to achieve multirate in-order packet delivery such that, more packets are delivered to the stronger users.

Each transmission is a linear combination of the packets along with a coefficient vector that determines the coefficient of each packet. The users store the received packets and the coefficient vectors in their buffers to apply Gaussian elimination for decoding. The coefficients are chosen from a Galois field $\mathbb{F}_q^\nu$. For simplicity, it is considered that each packet is a single symbol in $\mathbb{F}_q$.

**Definition 1:** A packet $p_i$ corresponds to $n$’th packet that has entered the transmission queue. A packet $p_n$ is older than $p_m$ if $n < m$, otherwise it is newer.

**Definition 2:** A packet $p_n$ is decoded by a user $U_i$ if the individual value of $p_n$ has been revealed by applying Gaussian elimination on the already received network coded packets.

**Definition 3:** A packet $p_n$ is delivered to a user if all older packets $p_1, p_2, \ldots, p_{n-1}$ have been decoded by that user. The number of delivered packets by $U_i$ at time slot $t$ is shown with $d_i(t)$.

\[^1\]For simplicity, we use the notation $\bar{p}$ for $1 - x$.\[^2\]
TABLE I
AN EXAMPLE OF A USER’S BUFFER

| $p_1$ | $p_2$ | $p_3$ | ... | $p_5$ | $p_6 + p_4$ | ... |

**Definition 4:** The user $U_i$ has seen a packet $p_n$, if it can compute a linear combination of the form $(p_n + q)$, where $q$ is a linear combination of the packets older than $p_n$.

**Example 1:** Consider Table I as an example of a user’s buffer. By Definitions 2, 3 and 4 $p_1, p_2, p_3$ are delivered packets thus they are decoded and seen too. $p_5$ is decoded and seen however, it is not delivered. $p_6$ is just a seen packet and $p_4$ is neither a seen packet nor decoded.

Seeing, decoding and delivering are the situations of a packet in the users’ buffers with different level of strength. Note that a delivered packet is also a decoded and seen packet and a decoded packet is a seen packet but the opposite is not true necessarily. Seeing a packet is an important concept for the queue management and decoding process [36]. When all the users have seen a packet, that packet is dropped from the transmission queue and the users save it until they receive older packets to decode it.

**Definition 5:** At time $t$, the next required packet of $U_i$ is the oldest unseen packet in its buffer, and it is denoted by $N_i(t)$.

**Example 2:** In Table I the next required packet will be $p_4$.

There is full feedback from the users to the transmitter so that in each time slot the sender has complete information about what packets the users have correctly received or lost and their next required packets. The sender uses this information to determine the combination of the packets for the next transmission [36].

**Definition 6:** A transmission $s(t)$ is a symbol in $F_q$ and comprises the next required packets of the users along with the coding coefficients at time slot $t$, which is given by

$$s(t) = \sum_{i=1}^{\nu} \alpha_i(t)N_i(t),$$

where $\alpha_i(t)$’s are chosen from $F_q$ using a non-block-based coding scheme, which will be defined shortly.

**Definition 7:** A transmission $s(t)$ is innovative for $U_i$, if it cannot be computed from the information stored in the buffer of $U_i$.

**Definition 8:** The delivery rate of a user $U_i$ is given by $R_i(t) = \frac{d_i(t)}{t}$, and the average rate at which packets are delivered to the user is $R_i = \lim_{t \to \infty} R_i(t)$.

### A. Coding Scheme

The sender employs linear network coding that was proposed in [36]. The method of packet encoding is given in Algorithm 1 [36]. In each time slot, the transmitter makes a list of the next requested packets $p_j$ by the users in descending order of the packet indices, excluding those users whose required packets have not yet arrived into the transmission queue. Let $G_j$ be the group of the users whose next requested packet is $p_j$. Starting with group $G_j$ with the highest index, it will add the packet $p_j$ into $s(t)$ only if the user(s) in $G_j$ do not otherwise receive an innovative packet. Furthermore, to ensure that we can always find an innovative transmission for all the users using this coding scheme, the field size should be $q \geq \nu$ [36]. To check if $s(t)$ is innovative, Gaussian elimination on $s(t)$ with the transmissions stored in $U_i$’s buffer.

**Algorithm 1:** Coding algorithm [36].

1: Organize users $U_1, \ldots, U_\nu$ into groups $G_j$, so that $G_j$ contains at least one user.
2: Initialize $s(t) = 0$.
3: for each group $G_j$, from high to low $j$, do
4: Initialize the empty veto list $v_j = \{}$.
5: for each user $U_i \in G_j$ do
6: Calculate $\rho_i$, the residual of performing Gaussian elimination on $s(t)$ with the transmissions stored in $U_i$’s buffer.
7: if $\rho_i = 0$ then
8: $v_j \leftarrow v_j \cup \{\}$.
9: else if $r_i = \alpha p_j$ for some field element $\alpha$ then
10: $v_j \leftarrow v_j \cup \{\alpha\}$.
11: end if
12: end for
13: if $0 \in v_j$ then
14: $a_j \equiv \min(F_q \setminus v_j)$.
15: Set $s(t) = s(t) + a_j p_j$.
16: end if
17: end for

**Example 3:** Consider a system with three users such that $c_1 > c_2 > c_3$. For simplicity, let $N_1(0) > N_2(0) > N_3(0)$ so that there is only a single user in each group $G_j$ corresponding to the next required packet $p_j$. An example of the transmission scheme is given in Table II.

At $t = 0$, 10 packets have arrived into the transmission queue. $U_1$ has $p_1 - p_{10}$ as delivered packets and $p_1 - p_{9}$ have been delivered to $U_2$ and also it has received $p_5$. $U_3$ has $p_1, p_2$ as delivered packets and it has received the combination $p_6 + p_3$. We have $d_1(0) = 10, d_2(0) = 5$ and $d_3(0) = 2$. The next required packets of the users are $N_1(0) = p_{11}, N_2(0) = p_6$ and $N_3(0) = p_1$. At $t = 1$, the sender checks the next required packets of the users and starts encoding by the highest index packet $p_{11}$. The sender sets $s(1) = p_{11}$ at the first step, then because $s(1)$ is innovative for all the users, it is sent without adding any other packet to it. After the transmission of $s(1)$, $U_2$ receives the packet successfully while $U_1$ and $U_3$ have erasures. At the next time slot $t = 2$, again $p_{11}$ is the next required packet with the highest index and encoding starts with $s(2) = p_{11}$. After that, the sender checks the users’ buffer information and finds out that $p_{11}$ is not innovative for $U_2$, thus it adds $N_2(2) = p_6$ to $s(2)$. Now $s(2)$ is innovative for all the users and it can be transmitted. After transmission $U_1$ receives the packets and decodes $p_{11}$, but $U_2$ and $U_3$ cannot receive the packet. At $t = 3$, the required
packet with the highest index is $N_1(3) = p_{12}$. However it has not entered the transmission queue, so $U_1$ is not considered for encoding. Therefore, the encoding starts with $N_2(3) = p_6$ and it is innovative for $U_2$ and $U_3$ thus it is transmitted. All the users receive this packet. It is not innovative for $U_1$ but $U_2$ receives its needed packet and $U_3$ uses it to decode the combination $p_6 + p_3$ and reveal $p_3$ as its required packet.

**Definition 9:** A user with the highest next requested packet index $N_i(t)$ in $s(t)$ is named the leader, also this transmission is called a leader transmission for $N_i(t)$. In Example 3, $U_1$ is the leader at time slots $t = 1, 2$ and $U_2$ is the leader at $t = 3$. Thus, the transmissions are leader transmissions for $p_{11}$ in $t = 1, 2$ and for $p_{06}$ in $t = 3$.

**Definition 10:** At time slot $t$, we call $s(t)$ a differential knowledge transmission for a user $U_j$ if $s(t)$ leads to the delivery of its next required packet, otherwise it is called a non-differential transmission. If $U_i$ is the leader and $s(t)$ is a differential knowledge for $U_j$, we say that $U_j$ has a differential knowledge from $U_i$ and the probability of this event is shown by $D_j^i$. In each time slot, the probability of transmitting differential knowledge for the leader, $D_1^i$, is one (In fact, leader transmission is a differential knowledge for the leader).

**Example 4:** In Table II, $s(1)$ and $s(2)$ are leader and differential knowledge transmissions for $U_1$, while $s(2)$ is a differential knowledge transmission for $U_2$. Moreover, $s(3)$ is a leader transmission (and also differential knowledge) for $U_2$. Thus, if they receive these corresponding transmissions their next required packets are delivered. Note that $s(1)$ is a non-differential transmission for $U_2$ and all transmissions are non-differential for $U_3$.

### III. Delivery Rate Analysis

In this section, the analysis of the delivery rate is proposed. First, let us introduce the following sets:

- $S(t)$: Set of packets arrived at the sender until end of time slot $t$. The size of $S(t)$ is denoted by $|S(t)|$.
- $S_i(t)$: Set of packets that user $U_i$ has seen until end of time slot $t$.

Based on the coding scheme 1, the sender drops a packet from the transmission queue when all the users have seen that packet.

In Table II, $s(1)$ and $s(2)$ are leader and differential knowledge transmissions for $U_1$, while $s(2)$ is a differential knowledge transmission for $U_2$. Moreover, $s(3)$ is a leader transmission (and also differential knowledge) for $U_2$. Thus, if they receive these corresponding transmissions their next required packets are delivered. Note that $s(1)$ is a non-differential transmission for $U_2$ and all transmissions are non-differential for $U_3$.

### III. Delivery Rate Analysis

In this section, the analysis of the delivery rate is proposed. First, let us introduce the following sets:

- $S(t)$: Set of packets arrived at the sender until end of time slot $t$. The size of $S(t)$ is denoted by $|S(t)|$.
- $S_i(t)$: Set of packets that user $U_i$ has seen until end of time slot $t$.

Based on the coding scheme 1, the sender drops a packet from the transmission queue when all the users have seen that packet.

In Table II, $s(1)$ and $s(2)$ are leader and differential knowledge transmissions for $U_1$, while $s(2)$ is a differential knowledge transmission for $U_2$. Moreover, $s(3)$ is a leader transmission (and also differential knowledge) for $U_2$. Thus, if they receive these corresponding transmissions their next required packets are delivered. Note that $s(1)$ is a non-differential transmission for $U_2$ and all transmissions are non-differential for $U_3$.

#### TABLE II

<table>
<thead>
<tr>
<th>$t$</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source buffer</td>
<td>$p_1 - p_{10}$</td>
<td>✓</td>
<td>$p_1 - p_{11}$</td>
<td>✓</td>
</tr>
<tr>
<td>$s(t)$</td>
<td>$p_{11}$</td>
<td>$p_{11} + p_6$</td>
<td>$p_6$</td>
<td></td>
</tr>
<tr>
<td>$U_1$ buffer</td>
<td>$p_1 - p_{10}$</td>
<td>✓</td>
<td>$p_1 - p_{10}$</td>
<td>✓</td>
</tr>
<tr>
<td>$U_2$ buffer</td>
<td>$p_1 - p_5, p_3$</td>
<td>✓</td>
<td>$p_1 - p_5, p_9, p_{11}$</td>
<td>✓</td>
</tr>
<tr>
<td>$U_3$ buffer</td>
<td>$p_1 - p_2, p_6 + p_3$</td>
<td>✓</td>
<td>$p_1 - p_2, p_6 + p_3$</td>
<td>✓</td>
</tr>
</tbody>
</table>

**Fig. 1.** Markov chain for the size of the virtual queue $Q_i(t)$. Thus, the size of the transmission queue is given by

$$Q(t) = |S(t)| - |C_{i=1}^n S_i(t)|.$$  \(2\)

Simlar to the physical transmission queue, we can define a virtual queue for each $U_i$ as the difference of $S(t)$ and $S_i(t)$. The size of such a virtual queue is given by

$$Q_i(t) = |S(t)| - |S_i(t)|.$$  \(3\)

$Q_i(t)$ can be modeled by a Markov chain (Fig. 1). The transition between the states depends on both the incoming packets to the virtual queue and seeing the packets by the user. If $c_i > \lambda$, the Markov chain is positive recurrent and has steady states and if $c_i \leq \lambda$ it is transient.

We can use the steady state distribution for the positive recurrent case, however, for the users with a transient Markov chain we use another approach. Thus, we separate users in two groups i.e., $\mathcal{H}$ and $\mathcal{L}$ due to the different behavior of their Markov chains. Accordingly, we use the following notation

- $\mathcal{H} = \{U_i : c_i > \lambda\}$,
- $\mathcal{L} = \{U_i : c_i \leq \lambda\}$,
- $|\mathcal{H}| = \nu_{\mathcal{H}}$.

For $U_i \in \mathcal{H}$, the steady state distribution of $Q_i(t)$ with the Markov chain in Fig. 1 is given by [37]

$$\pi_i(n) = \left(1 - \frac{\lambda c_i}{\lambda c_i^+}\right)^n \frac{\lambda c_i}{\lambda c_i^+}.$$  \(4\)

For $1 \leq i \leq \nu_{\mathcal{H}}$, when $Q_i(t) = 0$, $U_i \in \mathcal{H}$ is in the zero state and all the packets in the sender have been seen by $U_i$. Thus, all of them are decoded and delivered. In this situation, if no new packet arrives at the sender, the transmitted packet has no new information for $U_i$. The probability of this event is $\tilde{\lambda} \pi_i(0)$, otherwise, by the innovation guarantee property, the transmitted packet has new information for the user (i.e., if the user receives
this transmission, it can see a new packet) with probability
\[ 1 - \bar{\lambda} \pi_i(0) = \frac{\lambda}{c_i}. \]  

(5)

**Theorem 1:** In steady state, the delivery rate of each \( U_i \in \mathcal{H} \) asymptotically tends to \( \lambda \).

**Proof:** As \( t \to \infty \), the size of \( \mathcal{S}(t) \) tends to \( \lambda t \). Using (5), the transmitted packets have new information for a \( U_i \in \mathcal{H} \) in \( (\frac{\lambda}{c_i})t \) fraction of time and \( U_i \) receives \( \frac{\lambda}{c_i}t \) number of these transmissions on average. Because all of these received transmissions are innovative, \( U_i \) has seen all the packets and it can decode them, so the packets are delivered. Thus, the asymptotic delivery rate of the users in \( \mathcal{H} \) will be
\[ \lim_{t \to \infty} \frac{d_i(t)}{t} = \frac{\lambda t}{t} = \lambda. \] 

(6)

The asymptotic delivery rate of the users in \( \mathcal{H} \) is given by Theorem 1. To estimate the delivery rate of the users in \( \mathcal{L} \), we need the following arguments on the leadership probability and differential knowledge.

**A. Probability of Becoming the Leader**

In each time slot, at least one user is the leader (i.e., the next required packet is the highest index packet in \( s(t) \)), and encoding is done based on its requested packet. In the following, we analyze the probability of the leadership for the users in \( \mathcal{H} \) and \( \mathcal{L} \).

1) \( \mathcal{H} = \emptyset \): If \( \mathcal{H} = \emptyset \), all the users are in \( \mathcal{L} \) with the link capacities smaller than \( \lambda \). After some time slots, users are left behind the transmission queue and their next required packets always exist at the sender. On the other hand, the first user with the highest link capacity \( U_1 \equiv U_t \) receives more packets, so that after some time slots its next requested packet has the highest index in \( s(t) \). In this case, by ignoring some time slots from the beginning of the transmission, \( U_1 \) is always the leader and its leadership probability is assumed to be 1.

2) \( |\mathcal{H}| = 1 \): In this case, the strongest user \( U_1 \) is the only member of \( \mathcal{H} \). Because \( c_1 > \lambda \), there are some time slots that \( \bar{N}_1(t) \) has not arrived at the sender. Using (5), the probability of this event is \( 1 - \frac{1}{c_1} \). In these time slots, because the sender has no new information for \( U_1 \), the second user \( U_2 \equiv U_1 \) is the leader. Note that the only user in \( \mathcal{L} \) that can be the leader is \( U_1 \) based on the argument in the previous item. Thus, in this case, there are only two leaders \( U_1 \) and \( U_2 \). \( U_1 \) is the leader with probability of \( \frac{\lambda}{c_1} \) and \( U_2 \) is the leader with the complement probability.

3) \( |\mathcal{H}| > 1 \): In this case, since all the users in \( \mathcal{H} \) have capacities greater than \( \lambda \), there are situations where more than one user are leaders at the same time. Thus, analysis of the leadership probability is not trivial. However, we can make an estimation of the \( U_1 \) leadership probability and the probability that the leader is in \( \mathcal{H} \).

Based on the Markov chain analysis, each user in \( \mathcal{H} \) comes back to the zero state in steady state. It can happen for more than one user or even all the users in \( \mathcal{H} \) at the same time. When all the users in \( \mathcal{H} \) are in the zero state and no new packet arrives at the sender, \( U_l \) will be the leader based on the previous arguments. Now we estimate the probability of \( U_l \) leadership in this case.

Let us define a new virtual queue as
\[ Q_h(t) = |\mathcal{S}(t)| - |\mathcal{S}_h(t)|. \] 

(7)

When all the users in \( \mathcal{H} \) are in the zero state \( Q_h(t) = 0 \). Since each \( \mathcal{S}_h(t) \) is a subset of \( \mathcal{S}(t) \), we have [34]
\[ |\mathcal{S}(t)| - |\mathcal{S}_h(t)| \leq \sum_{i=1}^{\nu_h} (|\mathcal{S}(t)| - |\mathcal{S}_i(t)|), \] 

thus,
\[ Q_h(t) \leq \sum_{i=1}^{\nu_h} \mathcal{Q}_i(t). \] 

(9)

Using Markov chain, the steady state expected size of the virtual queue is given by
\[ \lim_{t \to \infty} \mathbb{E}[Q_h(t)] = \sum_{n=0}^{\infty} n \pi_i(n) = \frac{\bar{c}_i \lambda}{c_i - \lambda}. \] 

(10)

We consider a Markov chain for \( Q_h(t) \) and a corresponding parameter \( c_i \) as \( c_i \) in \( \mathcal{Q}_i(t) \). With this consideration and taking the expectation on both sides of (9), we obtain
\[ \frac{\bar{c}_j \lambda}{c_j - \lambda} \leq \sum_{i=1}^{\nu_h} \frac{\bar{c}_i \lambda}{c_i - \lambda}. \] 

(11)

Using (11) we have
\[ \frac{1 + \lambda \eta}{1 + \eta} \leq c_h, \] 

(12)

where, \( \eta = \sum_{i=1}^{\nu_h} \frac{\bar{c}_i}{c_i - \lambda} \).

The probability of \( U_1 \) leadership is \( \bar{\lambda} \Pr(Q_h = 0) \). Using (12) and (5), if we set
\[ \frac{1 + \lambda \eta}{1 + \eta} = c_h, \] 

(13)

an upper bound for \( \bar{\lambda} \Pr(Q_h = 0) \) will be given by
\[ \bar{\beta}_h = 1 - \frac{\lambda}{c_h}. \] 

(14)

\( \bar{\beta}_h \) is a possible minimum fraction of the time that \( U_l \) is the leader. Consequently, the fraction of the time that the leader is in group \( \mathcal{H} \) is given by the complement probability \( \bar{\beta}_h \).

4) **Leadership Probability Model:** First, we define the following parameter to summarize the previous results.
\[ \beta \triangleq \begin{cases} \bar{\beta}_h & \mathcal{H} \neq \emptyset, \\ 0 & \mathcal{H} = \emptyset. \end{cases} \] 

(15)

It shows the leadership probability of the users in \( \mathcal{H} \). If \( \mathcal{H} = \emptyset \), using (14) the leader is in group \( \mathcal{H} \) with the probability of \( \beta_h \), so \( \beta = \beta_h \). Note that if \( |\mathcal{H}| = 1 \) we have \( c_h = c_1 \). For the case of \( \mathcal{H} = \emptyset \), there is no user in \( \mathcal{H} \), so \( \beta = 0 \) and \( U_l \) is the leader with probability of \( \beta = 1 \).

In the case where \( |\mathcal{H}| > 1 \), due to the simultaneity of the leadership for the users in \( \mathcal{H} \) in some time slots, analysis of leadership probability for each individual user in this group is
very complicated. However, to estimate the delivery rate of the users in $\mathcal{L}$, we need the leadership probability. Thus, we omit exact analysis and use (13) and (14) to make an estimation for differential knowledge and delivery rate of the users in $\mathcal{L}$.

Suppose a virtual user $U_h$ with capacity $c_h$ corresponds to $Q_h$. To estimate differential knowledge and delivery rate of the users in $\mathcal{L}$, we consider $U_h$ as the representative of the users in $\mathcal{H}$. Using (15) the leader is in $\mathcal{H}$ in $\beta$ fraction of the time and is $U_l$ with the complement probability $\overline{\beta}$. When one or some of the users in $\mathcal{H}$ are the leaders, we consider $U_h$ as the leader and we use $c_h$ in our analysis instead of the capacity of the users in $\mathcal{H}$. Although it is not an exact model, it helps us to make a tractable model for the delivery rate of the users in $\mathcal{L}$. We summarize the above argument with the following approximation.

**Approximation 1:** In the rest of this paper, we consider $U_h$ as the representation for the leadership of the users in $\mathcal{H}$. We say $U_h$ is the leader if any user in $\mathcal{H}$ is the leader. We also use (13) in our analysis instead of the capacity of the users in $\mathcal{H}$. Thus, for simplicity we consider only two leaders $U_h$ and $U_l$. $U_h$ is the leader with priority of $\beta$ and $U_l$ with complement probability $\overline{\beta}$.

In the following, the indices $h$ and $l$ will be used for the corresponding parameters for $U_h$ and $U_l$, respectively. For instance, $d_h(t)$ and $d_l(t)$ are used for the number of delivered packets to $U_h$ and $U_l$ at time slot $t$, respectively.

**B. Probability of Differential Knowledge for the Users in $\mathcal{L}$**

In each time slot, there is just a leader $U_h$ or $U_l$ and by Definition 10 at least one user can receive a differential knowledge from one of these leaders. A differential knowledge transmission for $U_i$ is sent in a time slot $t$, if $N_i(t)$ has been encoded in the transmission packet. When $U_h$ is the leader the encoding is started with its requested packet and if it is the first request of a packet $p_x$ by $U_h$ it will be the first transmission of $p_x$ and we have $s(t) = p_x$. The transmission of $p_x$ will continue until it is received by $U_h$ and during this process, the request of another user $U_l$ is added to $s(t)$ if it receives $p_x$ while $U_h$ has not received it. Here, $U_l$ has a differential knowledge from $U_h$.

When $U_l$ is the leader there is a similar explanation, however, note that only the users in $\mathcal{L}$ can have differential knowledge from $U_l$.

Before calculating the probability of differential knowledge, the probability of the leader transmissions for $U_h$ and $U_l$ are determined. $L_h^k(k)$ and $L_l^k(k)$ are the probabilities of $k$ unsuccessful leader transmissions of a requested packet by $U_h$ and $U_l$, respectively. $L_h^k(k)$ is the probability of $k$ leader transmissions of a delivered packet by $U_h$. Remember that we replaced $U_l$ for the all members in group $\mathcal{H}$.

We start by deriving $L_h^k(k)$. When the transmission of a new packet for $U_h$ is started, it will continue until the packet is received. Note that transmissions for the other users have no new information for $U_h$ and they have no effect on $L_h^k(k)$, so it is given by

$$L_h^k(k) = c_h^k c_h,$$

where $c_h^k$ is probability of $k$ unsuccessful leader transmissions of a packet for $U_h$.

To determine $L_l^k(k)$, note that $U_l$ can deliver a packet by the leader transmissions or differential knowledges from $U_h$. The probability of the leader transmission for $U_l$ is $\beta h$ and the probability of receiving differential knowledge is $\beta D_h c_l$. We normalize the probabilities to $\overline{\beta} h + \beta D_h c_l$, in order to restrict our probability space to these events. Accordingly, $L_l^k(k)$ is given by

$$L_l^k(k) = \left( \frac{\overline{\beta} c_l}{\beta + \beta D_h c_l} \right)^k \left( \frac{\beta c_l + \beta D_h c_l}{\beta + \beta D_h c_l} \right).$$

To calculate $L_h(k)$, we consider two cases of $k = 0$ and $k > 0$. With the given Definition of $L_h(k)$, $k = 0$ denotes that $U_h$ received a packet using no leader transmission. This would mean $U_h$ received a differential knowledge transmission which is not possible by our model, so $L_h(0) = 0$. For $k > 0$, to have exactly $k$ leader transmissions there should be $k - 1$ unsuccessful leader transmissions followed by a successful one. Thus we have

$$L_h(k) = \begin{cases} 0, & k = 0, \\ c_h^{(k-1)} c_h, & k \neq 0. \end{cases}$$

Now the probability of differential knowledge for a user $U_i$ is calculated using the complement of the probability that it has not seen the required packet of the leaders, i.e.,

$$D_i^h = 1 - \left( \sum_{k=0}^{\infty} L_h^k(k) c_h^k \right),$$

$$D_i^l = \begin{cases} 1, & U_i = U_l, \\ 1 - \left( \sum_{k=0}^{\infty} L_h^k(k) c_h^k \right), & \mathcal{H} = \emptyset, \\ 1 - \left( \sum_{k=0}^{\infty} L_l^k(k) c_h^k \right) \times \left( \sum_{k=0}^{\infty} L_h^k(k) c_h^k \right), & \mathcal{H} \neq \emptyset. \end{cases}$$

Note that the summation in (19) represents the probability that $U_i$ has not seen the requested packet by $U_h$ while $U_h$ has been the leader. Similarly, in (20) when $\mathcal{H} = \emptyset$, the summation is the same probability while $U_i$ has been the leader. In the case of $\mathcal{H} \neq \emptyset$, there is the second summation which is the probability of transmitting the packet prior to the leadership of $U_l$ and they are independently multiplied.

**C. Delivery Rate**

**Theorem 2.** The asymptotic delivery rate of the users in this system is given by

$$R_i = \begin{cases} \lambda, & U_i \in \mathcal{L}, \\ \frac{\lambda (\beta D_h + \beta D_l) c_i}{1 - B_i}, & U_i \in \mathcal{H}. \end{cases}$$

In the above, we have

$$B_i = \frac{\beta D_h c_i}{\lambda} + \frac{\beta D_l c_i}{R_i}.$$
Corollary 1: The coding method in Algorithm 1 can achieve multicast packet delivery in the described system model to the users in $\mathcal{L}$ whenever it is non-empty.

According to (21), in a system with non-empty set $\mathcal{L}$, users experience different delivery rates. On the other hand, although users in $\mathcal{H}$ have equal delivery rate, users in group $\mathcal{L}$ experience delivery rates proportional to their capacity. This is the desired multicast packet delivery property of the coding method.

Proof: The average delivery rate of $U_i$ is given by $R_i = \lim_{t \to \infty} d_i(t)/t$. If $U_i \in \mathcal{H}$ the asymptotic delivery rate is $\lambda$ using Theorem 1. For the users in $\mathcal{L}$, the value of $d_i(t)$ is estimated by accumulation of the number of the packets delivered to $U_i$ from different types of transmissions. $U_i$ receives differential knowledge from users in $\mathcal{H}$ with the probability $\beta D_i^c c_{i1}$, and from $U_l$ with the probability $\beta D_i^c c_{i2}$. Furthermore, there are packets that $U_i$ receives from non-differential transmissions, which are distributed in the buffer of $U_i$ between $p_i$ to $p_{d_i(t)}$ and $p_1$ to $p_{d_i(t)}$. Assuming that these packets are uniformly distributed, we have

$$d_i(t) = \beta \left( D_i^h c_i t + \bar{D}_i^h c_i \frac{d_i(t)}{d_i(t)} t + \beta \left( D_i^c c_i t + \bar{D}_i^c c_{i1} \frac{d_i(t)}{d_i(t)} t, \right) \right),$$

where $\beta D_i^h c_i t$ is the fraction of the received packets from non-differential transmissions while $U_h$ is the leader. Similarly $\beta D_i^c c_{i1} t$ is the received non-differential packets while $U_l$ is the leader. These packets are in the delivered region of $U_i$’s buffer $p_1$ to $p_{d_i(t)}$. From (23) we have

$$d_i(t) = \beta D_i^h c_i t + \bar{D}_i^h c_i t + d_i(t) B_i. \quad (24)$$

Using (23) and (24) $B_i$ is given by (22). Note that $\lim_{t \to \infty} \frac{d_i(t)}{t} = \lambda$. Finally, using (24) the delivery rate is given by (21).

IV. DELAY ANALYSIS

Different types of delay analysis have been studied in [36], [37], [41] for this coding scheme in homogeneous networks. They considered both the decoding delay and delivery delay. However, in such a system where the packets can be used only if they are delivered, the decoding delay is less important than the delivery delay; because it is possible that a user decodes a packet but it must wait until it is actually delivered in order to the application layer. In the literature, the delivery delay has been considered as the time between when a packet enters the transmission queue and its delivery to the application at each user [36]. Using this definition of the delivery delay in heterogeneous networks, there may be a large difference between delivery times of a packet for different users. On the other hand, weak users left behind from the transmission queue, still seek older packets to complete their delivery. Therefore, we study the delivery delay using a new definition, which is based on the time that each user waits for a packet after it is first requested by that user. Using this new definition, the delivery delay of the users is measured independently of each other and with respect to their capability of delivering packets. We believe that this new definition is more suitable for heterogeneous networks and moreover, it leads to a closed form for the delivery delay.

Definition 11: The delivery delay of a packet for a user $U_i$ is the time between the first request of that packet and its delivery which is shown by $\theta_i$.

Theorem 3: Suppose that $d_{u_i}$ is the probability of delivering a packet by $U_i$ in each time slot. For $U_i$, the probability that a packet has $T$ time slots delivery delay $\theta_i = T$ is given by

$$\Pr(\theta_i = T) = \begin{cases} d_{u_i} (1 - d_{u_i})^{(T-1)}/R_i, & T > 0, \\ 1 - \sum_{T=1}^{\infty} \Pr(\theta_i = T), & T = 0. \end{cases} \quad (25)$$

Proof: A packet is delivered to a user $U_i$ with the probability of $d_{u_i}$, then $U_i$ requests the next packet and it can deliver that packet after $T > 0$ time slots with the probability of $(1 - d_{u_i})^{(T-1)}$. Therefore, the number of packets with delivery delay of $T > 0$ is $T d_{u_i} (1 - d_{u_i})^{(T-1)}$, and $\Pr(\theta_i = T)$ is given by $\lim_{T \to \infty} T d_{u_i} (1 - d_{u_i})^{(T-1)}/d_i(t)$. By summation on $\Pr(\theta_i = T)$ for $T > 0$, all packets with the non-zero delivery delay in $U_i$ buffer are considered and the probability of the rest of them is given by complement probability that is given in (25).

A. $|\mathcal{H}| = 0$

In this case, the strongest user is always the leader and the other ones receive their packets only via differential knowledge transmissions. Thus, according to Section III-C the probability of packet delivery is given by

$$d_{u_i} = \begin{cases} c_i, & i = 1, \\ D_i^h c_i, & i > 1. \end{cases} \quad (26)$$

For $U_1$ (the strongest user), $d_{u_1}$ is the same as the channel capacity because it is the strongest user and all the packets in its buffer are assumed to be delivered. However, for the other users $d_{u_i}$ is different, since they also receive non-differential packets which affects the number of delivered packets and the delivery rate, while $d_{u_1}$ is the probability of receiving a requested packet and its delivery at the same time. Now, $\Pr(\theta_i = T)$ can be determined using (25).

B. $|\mathcal{H}| = 1$

In this case, there are two leaders, the user in $\mathcal{H}$, $U_1 \equiv U_h$ and $U_2 \equiv U_l$. For $d_{u_i}$ we have

$$d_{u_i} = \begin{cases} \beta c_1 = \lambda, & i = 1 \ (U_1 \equiv U_h), \\ (\beta D_i^h + \beta c_1), & i = 2 \ (U_2 \equiv U_l), \\ \beta D_i^h + \beta D_i^c c_{i2}, & i > 2. \end{cases} \quad (27)$$

According to the delivery rate analysis in Section III, $\beta$ is the fraction of time that $N_i(t)$ is in the transmission queue, and $U_1$ receives it with the probability of $c_1$. Since all the packets
received by $U_1$ are delivered, $d_{u_1}$ is given by (27). On the other hand, the portion of time that $U_1^i$ is the leader is given by $\beta$, and this user delivers the packets via leader transmissions with the probability of $\beta c_i$ and differential knowledge transmissions with the probability of $\beta D_i^h c_i$. Moreover, other users deliver the packets via differential knowledge transmissions from these two leaders with the probability given in (27). Again, the delay probability is given by (25).

### C. $|\mathcal{H}| > 1$

When there are more than one user in $\mathcal{H}$, all members of $\mathcal{H}$ have a chance to be the leader and they receive differential knowledges from each other. Furthermore, using our analytical model, we cannot calculate the probability of being the leader and differential knowledge for the users in $\mathcal{H}$. However, if we had the leader and differential knowledge probabilities in $\mathcal{H}$, then $d_{u_1}$ would be given by

$$d_{u_1} = \begin{cases} \sum_{k: U_k \in \mathcal{H}} \beta_k D_k^i c_i, & U_i \in \mathcal{H}, \\ (\beta_h D_h^i + \beta_l D_l^i) c_i, & U_i \in \mathcal{L}. \end{cases}$$

(28)

Where $\beta_k$ is the probability of $U_k \in \mathcal{H}$ being the leader and $D_k^i$ is the probability of differential knowledge for $U_i$ when $U_k$ is the leader (Note that $D_i^l = 1$ which corresponds to the leader transmission for $U_i$). Because we cannot calculate the values of $\beta_k$ and $D_k^i$s for the users in $\mathcal{H}$, to evaluate the accuracy of (28), we extract these values from simulations and after calculating $d_{u_1}$, we use them in (25) and compare the results with simulations in Section V (see Fig. 5).

### D. Expected Value of the Delay

Another parameter for comparing the delay of the users is the expected value of the delivery delay. From (25), we have

$$\mathbb{E}\{\theta_i\} = \sum_{T=0}^{\infty} T \Pr(\theta_i = T) = \frac{1}{R_i}. \quad (29)$$

This is a reasonable result that the average delay of each user has an inverse relation to its delivery rate.

### V. Simulation Results

In this section, we compare the simulation results with the analysis to validate the results given in the previous sections. In our simulation setup, we have considered different packet arrival rates at the sender, number of users and channel erasure probabilities. These different settings are shown in Table III.

**TABLE III**

<table>
<thead>
<tr>
<th>Setting</th>
<th>$\lambda$</th>
<th>$c$</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.85</td>
<td>$[0.8, 0.6, 0.4, 0.2]$</td>
</tr>
<tr>
<td>B</td>
<td>0.85</td>
<td>$[0.9, 0.8, 0.7, 0.5, 0.3]$</td>
</tr>
<tr>
<td>C</td>
<td>0.6</td>
<td>$[0.8, 0.7, 0.5, 0.3, 0.2]$</td>
</tr>
<tr>
<td>D</td>
<td>0.6</td>
<td>$[0.9, 0.8, 0.7, 0.5, 0.4]$</td>
</tr>
<tr>
<td>E</td>
<td>0.8</td>
<td>$[0.9, 0.85, 0.8, 0.75, 0.7, 0.65, 0.6, 0.5]$</td>
</tr>
</tbody>
</table>

As described in Algorithm 1, at each time slot a Gaussian elimination is performed to construct the transmitted packet $s(t)$. After each transmission, Gaussian elimination is again performed on the users’ buffer to decode the packets. Then, the next required packet of each user (which is the oldest unseen packet) is determined. The newer seen packets are also stored in the buffer of each user until they receive older packets to decode them. Furthermore, for the purpose of the delivery delay analysis, the critical time slots for each packet such as arrival at the sender, seeing, decoding and delivering by each user are traced. Finally, we have measured the delivery rate and the delivery delay of the users after the delivery of 10000 packets to $U_1$.

### A. Delivery Rate

The comparison of simulation and analysis for the delivery rates are depicted in Fig. 2. To analyze delivery rate, (15) is used for the leader probability and (16)–(18) have been used for the probability of leader transmissions, then the differential knowledge probabilities are given by (19), (20) and the delivery rate is given by (21). In all settings, it is observed that the users in group $\mathcal{H}$ have reached a delivery rate very close to $\lambda$, and for the users in $\mathcal{L}$, there is a reasonable match between the simulation and analytical results.

Among simulation settings, it can be observed that settings A and D have the minimum error margin. In setting A, since $\mathcal{H} = \emptyset$ and there is only one leader, there is no need for approximation and the simulation result is very close to analysis. In setting D, the error decreases due to the low value of $\lambda$ and the number of the users in $\mathcal{L}$. In setting C, $\lambda$ has the same value as D, however, there is one more user in $\mathcal{L}$ and it causes the error to increase.
In setting B, since there is only one user in $\mathcal{H}$, the probability of being the leader is determined more accurately and the error observed for the last users is due to the high value of $\lambda$. Except setting E, the pattern of the analyzed delivery rate is very similar to the simulations. These observations show that the accuracy of our model decreases when the number of users and the value of $\lambda$ increase.

To have more accurate characterization of the delivery rates, as the number of users increases, more accurate analysis of leadership probability and differential knowledge is required. On the other hand, a key part of the analytical framework is Approximation 1, which treats the leaders in group $\mathcal{H}$ as a single user, subsequently it affects the precision of analysis. In summary, to have a more accurate model, finding a way to determine the leader probability of the users in $\mathcal{H}$ seems necessary.

B. Delivery Delay

Here, the derived expressions for the delay probabilities are compared with the values of simulations. Fig. 3 and Fig. 4 illustrate the simulation and analysis results for the cases of Section IV-A and IV-B respectively. For these cases, the settings A and B of Table III have been used for the simulation and (25), (26) and (27) have been used for the analysis. As it is observed, the analysis shows perfect match with simulation for $U_1$ and $U_2$ and loses its accuracy for the other users in the both settings due to the error in the calculation of differential knowledge probability. It is noteworthy that $\Pr(\theta_i = 0)$ is zero, that shows $U_1$ did not receive non-differential packets, because in setting A, $U_1$ has been the leader in all time slots, and in setting B, $U_2$ could be the leader only when $N_1(t)$ is not in the transmission queue. Moreover, for $U_1$ in the both settings most of the packets are delivered with the delay of $T = 1$ i.e., in the next time slot after requesting the packets. However, for $U_1$ in the both settings most of the packets are delivered with the delay of $T = 1$ i.e., in the next time slot after requesting the packets. However, for the other users $\Pr(\theta_i = 0)$ is maximum, since they receive most of their packets by the non-differential transmissions from the leaders. Although $\Pr(\theta_i = 0)$ is maximum, it does not mean that the users experience low delay. In order to compare the users in terms of the delay they experience, we should look at the range of $T$ for each user. For instance, in setting A, $U_1$ has the range of $0 \leq T \leq 6$ while for $U_2$ the range is $0 \leq T \leq 90$ (the ranges in

In setting B, since there is only one user in $\mathcal{H}$, the probability of being the leader is determined more accurately and the error observed for the last users is due to the high value of $\lambda$. Except setting E, the pattern of the analyzed delivery rate is very similar to the simulations. These observations show that the accuracy of our model decreases when the number of users and the value of $\lambda$ increase.

To have more accurate characterization of the delivery rates, as the number of users increases, more accurate analysis of
the figures are limited to have better illustration). The maximum value of \( T \) increases for \( U_i \) to 202 and for the last user to 1100. Furthermore, for \( U_i \)’s with \( i > 1 \) the probability of delay has a slow decline for \( T > 0 \) that shows the number of packets for each value of \( T \) is close to each other.

The simulation results for Section IV-C is depicted in Fig. 5. Setting C of Table III, (25) and (28) have been used for the simulation and analysis. Note that in this case \( \Pr(\theta_i = 0) \) is not zero, because the other members in \( H \) could be the leader and all of the users in \( H \) can receive differential knowledge and also non-differential transmissions from each other. Figure 6 illustrates the simulation and analysis results for 29. Since expectation of the delay is the inverse of the delivery rate, error margin of the delay expectation increases for weaker users because the delivery rate value of these users is small and a little error in its analysis affects the delay expectation considerably. Using this comparison, we conclude that the stronger users have less delay.

VI. CONCLUSION

In this paper, we have shown that a previously proposed network coding scheme can achieve efficient multirate packet delivery in heterogeneous broadcast packet erasure networks. Also, we have introduced an appropriate model to estimate the delivery rate and the delivery delay of the system. Using this coding scheme, the strongest user, receives packets with the maximum possible throughput and the other users have a non-zero delivery rate according to their link capacities. Moreover, we have introduced a new definition for the delivery delay and analyzed the system based on it. The number of time slots between the first request of a packet and its delivery is counted as the delay. Using this definition, the delivery delays of the users have been compared and a simple expression has been derived. Similar to the delivery rate analysis, the numerical results for the delivery delay have shown a reasonable match between our analysis and the simulation results, especially for stronger users. Although achieving multirate packet delivery is possible for a number of users, it seems by increasing the number of users, the delivery rate of the weaker users tends to zero. Designing a coding method to support multirate packet delivery for a large number of users in a heterogeneous case can be considered in future works. Furthermore, considering other performance measures like fairness might also be useful.

ACKNOWLEDGEMENT

The authors would like to acknowledge helpful discussions with Prof. M. Médard in early parts of this work.

REFERENCES


Fig. 6. Delay expectation of different users, analysis and simulation for the settings of Table III. For analysis (29) is used.


Mohammad A. Zarrabian was born in Mashhad, Iran, in 1990. He received the B.Sc. degree in electrical engineering from the Ferdowsi University of Mashhad, Mashhad, Iran, in 2013, and the M.Sc. degree in electrical engineering from the Isfahan University of Technology, Isfahan, Iran, in 2016. His research interests include wireless communication, data networks, network coding, and information theory.

Foroogh S. Tahatab (S’09–M’12) received the B.S. degree (with honors) in electrical engineering from the Isfahan University of Technology, Isfahan, Iran, in 2004, and the M.S. and the Ph.D. degrees in electrical engineering from the Sharif University of Technology, Tehran, Iran, in 2006 and 2011, respectively. In February 2012, she joined the communication group in the Department of Electrical and Computer Engineering, Isfahan University of Technology, where she is currently an Assistant Professor. She was a recipient of the Young Professor Grant of the National Elite Foundation in September 2012. Her research interests include communication systems theory, wireless communications, information theory applications, network coding, optical networks, and molecular communication.

Sina Molavipour was born in Kerman, Iran, in 1990. He received the B.Sc. degree in electrical engineering from the Ferdowsi University of Mashhad, Mashhad, Iran, in 2013, and the M.Sc. degree in electrical engineering from the Sharif University of Technology, Tehran, Iran, in 2015. He is currently working toward the Ph.D. degree with the School of Electrical Engineering and Computing Science, KTH Royal Institute of Technology, Stockholm, Sweden. His research interests include information theory, and statistical inference. His specific research areas include estimation theory, graphical models, causality, and machine learning.

Parastoo Sadeghi received the bachelor’s and master’s degrees in electrical engineering from the Sharif University of Technology, Tehran, Iran, in 1995 and 1997, respectively, and the Ph.D. degree in electrical engineering from the University of New South Wales, Sydney, NSW, Australia, in 2006. She is currently an Associate Professor with the College of Engineering and Computer Science, Australian National University, Canberra, VIC, Australia. She has co-authored a book and more than 160 journal articles and conference papers in the areas of network coding, information theory, signal processing, and wireless communications.