Research Article

Network Coding-Based Retransmission for Relay Aided Multisource Multicast Networks

Quoc-Tuan Vien,[1] Le-Nam Tran,[2] and Een-Kee Hong[3]

[1] *School of Engineering & Computing, Glasgow Caledonian University, Cowcaddens Road, Glasgow G4 0BA, UK*
[2] *Signal Processing Laboratory, ACCESS Linnaeus Center, KTH Royal Institute of Technology, 100 44 Stockholm, Sweden*
[3] *School of Electronics and Information, Kyung Hee University, Yongin, Gyeonggi-do 446-701, Republic of Korea*

Correspondence should be addressed to Een-Kee Hong, ekhong@khu.ac.kr

This paper considers the reliable transmission for wireless multicast networks where multiple sources want to distribute information to a set of destinations with assistance of a relay. Basically, the reliability of a communication link is assured via automatic repeat request (ARQ) protocols. In the context of multisource multicast networks, the challenge is how to retransmit the lost or erroneous packets efficiently. In traditional approaches, the retransmission of lost packets from a single source is treated separately, and thus it may cause a considerable delay. To solve this problem, we propose the relay detects, combines, and forwards the packets which are lost at destinations using network coding. In the proposed ARQ protocol, the relay detects packets from different sources and combines the lost packets using NC. In particular, two packet-combination algorithms are developed to guarantee that all lost packets are retransmitted with the smallest number of retransmissions. Furthermore, we analyze the transmission bandwidth and provide the numerical results to demonstrate the superior performance of the proposed ARQ protocol over some existing schemes.

1. Introduction

Recently, relay communication has been extensively investigated as a promising technique to extend the coverage of wireless networks by exploiting the spatial diversity gains [1–3]. Generally, the use of relays does not immediately increase the network throughput since packets traverse along the relays via store-and-forward manner. For some particular network topologies such as two-way relay channels, relay-assisted broadcast channels, and multicast channels, the network throughput can be dramatically improved by applying network coding (NC) at the relays [4–9]. The basic idea of NC is that the relays are allowed to perform algebraic linear operations on the received packets from multiple sources and forward the combined packet in the subsequent transmission.

In this paper, we consider the reliable transmission over multisource-multicast networks [10] with assistance of a relay, where multiple sources want to transmit their messages to a set of intended destinations. This network model is widely applicable in various scenarios, in particular wireless ad hoc networks, where a set of sources needs to transmit data to a set of destinations through relays. One way to deliver information reliably over error-prone channels is to employ automatic repeat request (ARQ) protocols [11], in which, if a packet cannot be decoded, it is discarded and retransmitted. Applying the traditional ARQ techniques to multicast or broadcast networks may cause considerable delay for two reasons: (i) the lost packets of different destinations are retransmitted individually and (ii) the retransmission will be repeated until all destinations receive all packets correctly. To reduce the number of retransmissions, ARQ schemes based on NC have been proposed in [12, 13]. The relay may XOR the disjointedly erroneous packets of different destinations and retransmit them to all the involving destinations.

The existing NC-based ARQ strategies for reliable wireless multicast networks are devised for the deployment
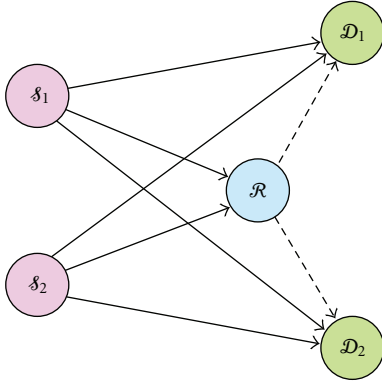
FIGURE 1: Multisource-multicast network model.

FIGURE 2: Retransmission with RT, traditional NC-based ARQ, and our proposed protocol.

scenario where a source distributes information to multiple intended destinations, as in [5, 14]. The problem of designing a retransmission mechanism for multisource-multicast networks that can achieve a high network throughput efficiency has received less interest. The NC-based ARQ strategies for multicast networks proposed in [12] can be reused for multisource-multicast network by viewing a multisource-multicast network as a superposition of several multicast networks. More specifically, the lost packets in the same information flow can be XORed using the NC-based ARQ strategies for multicast networks. Here, The information flow is defined as the data transmission from a source to multiple destinations. However, this traditional NC-based ARQ may result in a poor throughput efficiency since the information flows from distinct sources are treated independently.

In this paper, we propose a new NC-based ARQ protocol for multisource-multicast networks, in which the relay detects, combines, and sends the lost packets from different sources to the destinations. To achieve the best performance, multiuser detection (MUD) techniques, such as optimum detector, linear decorrelating detector, decision-feedback detector, and successive interference cancellation. [15–19] are employed at the relay and destinations.

Thus, many lost packets from different sources can be combined and retransmitted. We need to develop an ARQ protocol to retransmit these lost packets in a systematic and efficient way. First, we classify lost packets into two types: Type-I includes the packets that are lost at the destinations but successfully received at the relay, and Type-II packets are lost at both destinations and the relay. Obviously, the sources must handle the retransmission of Type-II packets. The retransmission algorithm based on NC for sources can be easily designed since these packets are in the same flow, and thus it becomes the classical application of NC. The problem is how the relay efficiently retransmits Type-I packets that can come from different sources.

Dealing with that problem, we propose an algorithm at the relay to retransmit Type-I packets and an algorithm at the source to retransmit Type-II packets. Particularly, the algorithm for the retransmission at the relay is proposed based on an integration of NC and packet detection from two different sources.

Unlike the traditional NC-based ARQ, the proposed method can combine the packets from different flows and thus can improve the network throughput for multisource-multicast networks. As we show later in an example (Figure 2), with our proposed ARQ protocol, the number of retransmissions is significantly reduced, comparing with other ARQ-based protocols. As a second contribution of the paper, we compare our proposed method with other ARQ-based protocols for multisource-multicast networks by evaluating the transmission bandwidth with theoretical analysis and numerical results. In fact, three protocols are taken into account: direct transmission (DT), relaying transmission (RT), and the traditional NC-based ARQ. DT protocol denotes the case when multiple sources simultaneously transmit information to destination without relaying technique. RT protocol represents the model where relay participates in the transmission but NC is not performed at the relay (e.g., decode-and-forward relaying technique [2]).

The rest of this paper is organized as follows. In Section 2, we describe the system model of a multisource-multicast network. Different retransmission protocols are also presented in this section, and their transmission bandwidths are derived in Section 3. We provide the numerical results in Section 4 and Section 5 concludes this paper.

2. System Model and Transmission Protocols

The system model under investigation is shown in Figure 1. The data delivery from two sources $\mathcal{S}_1$ and $\mathcal{S}_2$ to two destinations $\mathcal{D}_1$ and $\mathcal{D}_2$ is assisted by a relay $\mathcal{R}$. This is a specific case of multisource-multicast networks where the

numbers of sources and destinations are 2 and 2, respectively. The generalization to cope with more than two sources and two destinations is straightforward.

In this paper, we assume that the sources send data in the form of packets (i.e., packet-based transmission) and each packet must be received correctly by all destinations after several transmissions and retransmissions. The packet loss of transmission from $\mathcal{S}_i$ to $\mathcal{R}$, from $\mathcal{S}_i$ to $\mathcal{D}_j$, and from $\mathcal{R}$ to $\mathcal{D}_j$ follows Bernoulli trial with parameters $p_{s_i r}$, $p_{s_i d_j}$, and $p_{r d_j}$, respectively. We also assume that the sources and the relay are equipped with sufficient signal processing modules to be able to perform NC, that is, algebraic operation such as bitwise XOR operation.

Receiving the information data from multiple sources along with the feedback from the destinations, the relay knows what destinations are waiting for the lost packets to be retransmitted and then decides how to combine and forward the data to the intended destinations. In the following, we introduce some protocols that allow the relay to resend the lost packets to the destinations. The two fundamental DT and RT protocols are presented first, and our proposed protocol is followed.

2.1. DT Protocol. In this protocol, the sources send data directly to two destinations. The transmission is carried out with the traditional ARQ scheme and completed if two destinations receive correctly the data from two sources.

2.2. RT Protocol. This protocol is different from DT protocol in that the relay participates in the transmission. When one or two destinations do not receive the packet correctly, whereas the relay receives this packet successfully, the relay can help the source to forward the correctly received packet to the erroneously received destinations in the next time slot. The retransmission at the relay will be continued until its transmitted packet is correctly received by the intended destinations. In case that one or two destinations and the relay fail to receive the same packet from the sources, it is obvious that the sources need to resend that lost packet.

2.3. Proposed Protocol. Instead of resending the lost packet as soon as the destinations fail to receive it, the retransmission in the proposed ARQ protocol will happen after N packets.

The buffer of length N is required at two sources, whereas the buffer of length $2N$ is required at the relay and two destinations since they receive packets from two different sources. To improve the network throughput, the relay retransmits the packets of Type-I, and the sources deal with the retransmission of Type-II packets. What is particular in the proposed ARQ strategy is that the relay can mix the packets from different data flows.

Let us describe the proposed protocol by examining an example shown in Figure 2, where $\mathcal{S}_i$ wishes to deliver 10 packets $s_i[1], s_i[2], \ldots, s_i[10]$ to $\mathcal{D}_1$ and $\mathcal{D}_2$.

The packets with a crossover sign represent the lost or erroneous packets. Without loss of generality, we assume that, for the data flow from $\mathcal{S}_1$, the erroneous packets received at $\mathcal{R}$, $\mathcal{D}_1$, and $\mathcal{D}_2$ are $\{s_1[4], s_1[9]\}$,

$\{s_1[5], s_1[9], s_1[10]\}$, and $\{s_1[1], s_1[2], s_1[3], s_1[4], s_1[8]\}$, respectively. Similarly, the erroneous packets received at $\mathcal{R}$, $\mathcal{D}_1$, and $\mathcal{D}_2$ from $\mathcal{S}_2$ are assumed to be $\{s_2[3], s_2[5]\}$, $\{s_2[1], s_2[2], s_2[5], s_2[6], s_2[8]\}$, and $\{s_2[3], s_2[6], s_2[7]\}$, respectively.

As shown in Figure 2, $\mathcal{R}$ needs to retransmit 11 packets if applying RT protocol. The number of retransmissions can be significantly reduced with the traditional NC-based ARQ scheme where $\mathcal{R}$ would transmit $\{s_1[1] \oplus s_1[5], s_1[2] \oplus s_1[10], s_1[3], s_1[8]\}$, and $\{s_2[1] \oplus s_2[7], s_2[2], s_2[6], s_2[8]\}$. In this scheme, we could not combine $s_1[3]$ and $s_1[8]$ since both of them are corrupted at $\mathcal{D}_2$. Similarly, there is no way to combine $s_2[2]$, $s_2[6]$, and $s_2[8]$ since $s_2[2]$ and $s_2[8]$ are simultaneously lost at $\mathcal{D}_2$ and $s_2[6]$ is lost at both $\mathcal{D}_1$ and $\mathcal{D}_2$. Thus, 8 retransmissions are totally required for the traditional NC-based ARQ scheme. That helps $\mathcal{R}$ save 3 retransmissions. Not stopping at that, we can further reduce the number of retransmissions with our proposed scheme if packets from different data flows are detected in parallel at $\mathcal{R}$, $\mathcal{D}_1$, and $\mathcal{D}_2$. As in our above definition, packets $\{s_1[1], s_2[1], s_1[2], s_2[2], s_1[3], s_1[5], s_2[6], s_2[7], s_1[8], s_2[8], s_1[10]\}$ are Type-I packets. To improve the network throughput, $\mathcal{R}$ forwards $\{s_1[1] \oplus s_2[1], s_1[2] \oplus s_2[2], s_2[7] \oplus s_2[8], s_1[8] \oplus s_1[10], s_1[3] \oplus s_1[5], s_2[6]\}$ in the retransmission phase. The details of this combination algorithm are presented in Algorithm 1. That means, the proposed ARQ requires 6 retransmissions and thus can save 2 further retransmissions comparing with the traditional NC-based ARQ scheme. $\mathcal{R}$ retransmits these packets until they are successfully received by both $\mathcal{D}_1$ and $\mathcal{D}_2$.

We can see that $s_1[4]$, $s_1[9]$, $s_2[3]$, and $s_2[5]$ are lost at $\mathcal{R}$ and also lost at $\mathcal{D}_1$ and/or $\mathcal{D}_2$. These packets are classified as Type-II packets. Obviously, the relay has no way to resend such packets, and thus the sources must resend these packets. In the example, $\mathcal{S}_1$ and $\mathcal{S}_2$ retransmit the combined packets $\{s_1[4] \oplus s_1[9]\}$ and $\{s_2[3] \oplus s_2[5]\}$, respectively. The destinations are able to recover the corrupted packets by XORing their correctly received packets with the XORed packets received from the relay or sources.

The generalization of the above example for the arbitrary buffer size is summarized in Figure 3. In this protocol, the combination algorithms at the relay and sources are presented in Algorithms 1 and 2, respectively.

3. Transmission Bandwidth Analysis

In this section, we study the transmission bandwidth of different transmission protocols in multisource-multicast networks consisting of two sources, one relay, and two destinations. The transmission bandwidth is defined as the average number of transmissions that is required to successfully transmit two packets from two sources to two destinations.

3.1. DT Protocol. This protocol is the simplest in which two sources directly send packets to two destinations without
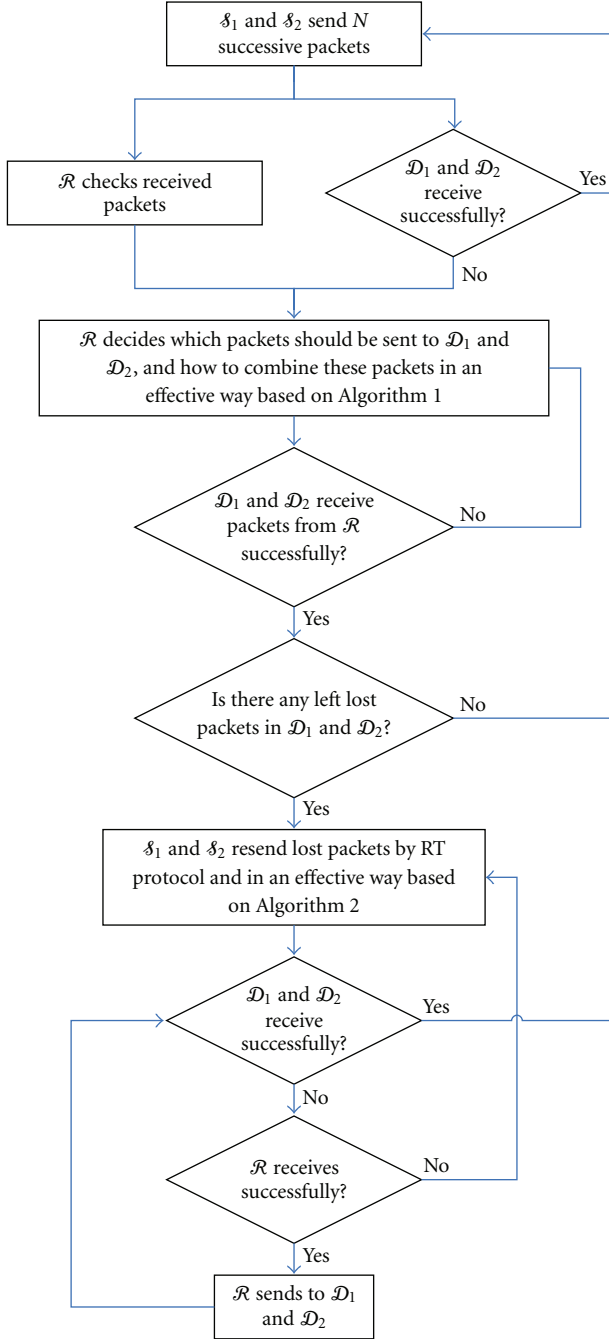
FIGURE 3: Block diagram of proposed protocol.

relay and network coding. The transmission bandwidth is given by

$$n_{\mathrm{DT}} = \max\left\{ n_{\mathrm{DT}}^{(s_1)}, n_{\mathrm{DT}}^{(s_2)} \right\}, \tag{1}$$

where $n_{\mathrm{DT}}^{(s_i)}$, $i = 1, 2$, denotes the average number of transmissions required for $\mathcal{S}_i$ to send data to both $\mathcal{D}_1$ and $\mathcal{D}_2$ and is calculated as [13]

$$n_{\mathrm{DT}}^{(s_i)} = \frac{1}{1 - p_{s_i d_1}} + \frac{1}{1 - p_{s_i d_2}} - \frac{1}{1 - p_{s_i d_1} p_{s_i d_2}}. \tag{2}$$

3.2. RT Protocol. In this protocol, the relay helps two sources in sending data to two destinations, but no NC is applied at the relay. The transmission bandwidth required to successfully transmit two packets from $\mathcal{S}_1$ and $\mathcal{S}_2$ to $\mathcal{D}_i$, $i = 1, 2$, is given by

$$\begin{aligned}
n_{\mathrm{RT}}^{(d_i)} &= \frac{1}{1 - p_{s_1 r} p_{s_2 r} p_{s_1 d_i} p_{s_2 d_i}} \\
&\quad \times \Big[1 + p_{s_1 r} p_{s_1 d_i} (1 - p_{s_2 d_i}) n_{\mathrm{RT}}^{(s_1, d_i)} \\
&\qquad + p_{s_2 r} (1 - p_{s_1 d_i}) p_{s_2 d_i} n_{\mathrm{RT}}^{(s_2, d_i)} \\
&\qquad + (1 - p_{s_1 r}) p_{s_1 d_i} (1 - p_{s_2 d_i}) n_{r d_i} \\
&\qquad + (1 - p_{s_2 r}) (1 - p_{s_1 d_i}) p_{s_2 d_i} n_{r d_i} \\
&\qquad + 2(1 - p_{s_1 r})(1 - p_{s_2 r}) p_{s_1 d_i} p_{s_2 d_i} n_{r d_i} \\
&\qquad + (1 - p_{s_1 r}) p_{s_2 r} p_{s_1 d_i} p_{s_2 d_i} \left(n_{r d_i} + n_{\mathrm{RT}}^{(s_2, d_i)} \right) \\
&\qquad + p_{s_1 r} (1 - p_{s_2 r}) p_{s_1 d_i} p_{s_2 d_i} \left(n_{r d_i} + n_{\mathrm{RT}}^{(s_1, d_i)} \right) \Big],
\end{aligned} \tag{3}$$

where $n_{\mathrm{RT}}^{(s_i, d_j)}$ denotes the average number of transmissions to send a packet from $\mathcal{S}_i$ to $\mathcal{D}_j$ with the help of $\mathcal{R}$ and is found as [12]

$$n_{\mathrm{RT}}^{(s_i, d_j)} = \frac{1 + p_{r d_j} + p_{s_i d_j} (1 - p_{s_i r})}{\left(1 - p_{s_i r} p_{s_i d_j}\right)\left(1 - p_{r d_j}\right)}. \tag{4}$$

Finally, the transmission bandwidth of this protocol is

$$n_{\mathrm{RT}} = \max\left\{ n_{\mathrm{RT}}^{(d_1)}, n_{\mathrm{RT}}^{(d_2)} \right\}. \tag{5}$$

3.3. Traditional NC-Based ARQ Protocol. The relay in this protocol combines the lost packets in the same flow based on NC in the retransmission phase. The transmission bandwidth is given by

$$n_{\mathrm{NCA}} = \max\left\{ n_{\mathrm{NCA}}^{(s_1)}, n_{\mathrm{NCA}}^{(s_2)} \right\}, \tag{6}$$

where $n_{\mathrm{NCA}}^{(s_i)}$, $i = 1, 2$, denotes the average number of transmissions to transmit from $\mathcal{S}_i$ to both $\mathcal{D}_1$ and $\mathcal{D}_2$ with the help of $\mathcal{R}$ in the traditional NC-based ARQ (NCA) protocol. Note that the data delivery from $\mathcal{S}_i$ to $\mathcal{D}_1$ and $\mathcal{D}_2$ through $\mathcal{R}$ resembles the system model in [12]. However, the analysis of transmission bandwidth presented in [12] is difficult to follow. Here, we introduce a simple way to calculate the transmission bandwidth.

In this protocol, there are three steps to transmit data from $\mathcal{S}_i$ to $\mathcal{D}_1$ and $\mathcal{D}_2$ through $\mathcal{R}$.

Step 1. $\mathcal{S}_i$ transmits N packets.

Step 2. $\mathcal{R}$ retransmits Type-I packets.

Step 3. $\mathcal{S}_i$ retransmits Type-II packets.

Let $n_{\mathrm{NCA}}^{(s_i, j)}$, $i \in \{1, 2\}$, $j \in \{1, 2, 3\}$, denote the number of transmissions in the jth step of the traditional NCA protocol.

(1) Let $\mathfrak{G}_1$ and $\mathfrak{G}_2$ denote the ordered set of correctly received packet at $\mathcal{R}$ transmitted from $\mathcal{S}_1$ and $\mathcal{S}_2$, respectively: $\mathfrak{G}_1 = \{s_1[i_1], s_1[i_2], \ldots, s_1[i_m]\}$, where $i_1 < i_2 < \cdots < i_m \in \{1, 2, \ldots, N\}$, $\mathfrak{G}_2 = \{s_2[j_1], s_2[j_2], \ldots, s_2[j_n]\}$, where $j_1 < j_2 < \cdots < j_n \in \{1, 2, \ldots, N\}$.
Define $\Omega = \mathfrak{G}_1 \cup \mathfrak{G}_2$ and divide Ω into 3 groups:
  (i) Group Ω_1 includes packets that $\mathcal{R}$ receives successfully from both $\mathcal{S}_1$ and $\mathcal{S}_2$, that is, $\Omega_1 = \{(s_1[i], s_2[j]) \mid i = j\}$. In the preceding example, $(s_1[1], s_2[1])$, $(s_1[2], s_2[2])$, $(s_1[6], s_2[6])$, $(s_1[7], s_2[7])$, $(s_1[8], s_2[8])$, and $(s_1[10], s_2[10])$ belongs to Ω_1,
  (ii) Group Ω_2 includes packets that $\mathcal{R}$ receives successfully from $\mathcal{S}_1$ but fails to receive from $\mathcal{S}_2$, that is, $\Omega_2 = \{s_1[i] \mid i \notin \{j_1, j_2, \ldots, j_n\}\}$. In Figure 2, Ω_2 includes $s_1[3]$ and $s_1[5]$.
  (iii) Group Ω_3 includes packets that $\mathcal{R}$ receives successfully from $\mathcal{S}_2$ but fails to receive from $\mathcal{S}_1$: $\Omega_3 = \{s_2[j] \mid j \notin \{i_1, i_2, \ldots, i_m\}\}$. In Figure 2, Ω_3 includes $s_2[4]$ and $s_2[9]$.
(2) For packets in Ω_1, if one packet is correctly received at $\mathcal{D}_1$, lost at $\mathcal{D}_2$, while another packet is correctly received at $\mathcal{D}_2$, lost at $\mathcal{D}_1$, we can combine these two packets.
Thus, there are 3 possibilities: $s_1[k_1] \oplus s_2[k_2]$ or $s_1[m_1] \oplus s_1[m_2]$ or $s_2[n_1] \oplus s_2[m_2]$.
Start from left to right in the group of packets in Ω_1, and choose the suitable combination (e.g., $s_1[1] \oplus s_2[1]$, $s_1[2] \oplus s_2[2]$, $s_2[7] \oplus s_2[8]$, and $s_1[8] \oplus s_1[10]$ in the above example)
(3) For packets in Ω_2 and Ω_3, similarly if one packet is correctly received at $\mathcal{D}_1$, lost at $\mathcal{D}_2$, while another packet is correctly received at $\mathcal{D}_2$, lost at $\mathcal{D}_1$, we can combine these two packets in only one way $s_1[m_1] \oplus s_1[m_2]$ (for Ω_2) or $s_2[n_1] \oplus s_2[m_2]$ (for Ω_3). (e.g., $s_1[3] \oplus s_1[5]$ in the above example)
(4) For the remaining lost packets at $\mathcal{D}_1$ and $\mathcal{D}_2$ that $\mathcal{R}$ receives successfully but cannot perform the combination, they are normally resent without using NC (e.g., $s_1[6]$ in the above example)

ALGORITHM 1: Algorithm at relay to retransmit Type-I packets.

(1) Through the feedback from $\mathcal{D}_1$, $\mathcal{D}_2$, and $\mathcal{R}$, $\mathcal{S}_i$ determines the number and the position of remaining lost packets at destinations that $\mathcal{R}$ also fails in receiving them.
(2) Combine the packets for retransmission by NC with the condition that only one packet in the combined packet should be correctly received by only one destination, similar to the combination performed for packets in Ω_2 and Ω_3 explained in Algorithm 1. In the preceding example, $\mathcal{S}_1$ resends $s_1[4] \oplus s_1[9]$ and $\mathcal{S}_2$ resends $s_2[3] \oplus s_2[5]$.
(3) For the remaining lost packets at $\mathcal{D}_1$ and $\mathcal{D}_2$ that $\mathcal{S}_i$ cannot perform the combination, they are resent without using NC.

ALGORITHM 2: Algorithm at source to retransmit Type-II packets.

The average number of transmissions to send from $\mathcal{S}_i$ to both $\mathcal{D}_1$ and $\mathcal{D}_2$ is calculated by

$$n_{\mathrm{NCA}}^{(s_i)} = \frac{n_{\mathrm{NCA}}^{(s_i,1)} + n_{\mathrm{NCA}}^{(s_i,2)} + n_{\mathrm{NCA}}^{(s_i,3)}}{N}, \tag{7}$$

where the number of transmissions in Step 1 is simply given by

$$n_{\mathrm{NCA}}^{(s_i,1)} = N. \tag{8}$$

The number of transmissions in Step 2 and in Step 3 is calculated by

$$n_{\mathrm{NCA}}^{(s_i,2)} = \sum_{k=0}^{N} C_k^N p_{s_i r}^{N-k} (1 - p_{s_i r})^k E\left[n_{\mathrm{NCA}}^{(s_i,2)} \mid K = k\right], \tag{9}$$

$$n_{\mathrm{NCA}}^{(s_i,3)} = \sum_{k=0}^{N} C_k^N p_{s_i r}^{N-k} (1 - p_{s_i r})^k E\left[n_{\mathrm{NCA}}^{(s_i,3)} \mid K = k\right], \tag{10}$$

respectively, where $E[\cdot]$ denotes the expectation, C_k^N is the total number of subsets consisting of k elements in a set of N elements, and K is a random variable representing the number of packets that $\mathcal{R}$ successfully receives in the first step.

Given that $K = k$ packets are successfully received at $\mathcal{R}$ in the first step, the number of transmissions at $\mathcal{R}$ using traditional NCA protocol in the second step can be computed by

$$E\left[n_{\mathrm{NCA}}^{(s_{i_s},2)} \mid K = k\right]$$

$$= \sum_{i=0}^{k} \sum_{j=0}^{k} C_i^k p_{s_{i_s} d_1}^i \left(1 - p_{s_{i_s} d_1}\right)^{k-i} C_j^k p_{s_{i_s} d_2}^j \left(1 - p_{s_{i_s} d_2}\right)^{k-j}$$

$$\times \left[\min\{i, j\} n_{\mathrm{DT}}^{(r)} + |i - j| n_{r d_a}\right], \tag{11}$$

where $i_s \in \{1, 2\}$, $n_{\mathrm{DT}}^{(r)}$ is the average number of transmissions required at $\mathcal{R}$ to send a packet to both $\mathcal{D}_1$ and $\mathcal{D}_2$, and $n_{r d_a}$ is the average number of transmissions for a direct transmission from $\mathcal{R}$ to $\mathcal{D}_a$, where $a = 1$ if $i > j$, and $a = 2$

otherwise. The term $[\min\{i, j\} \, n_{\text{DT}}^{(r)} + |i - j|n_{rd_a}]$ in (11) is derived based on the fact that there are $\min\{i, j\}$ packets that the relay can combine with NC, and thus the number of transmissions is given by $\min\{i, j\} \, n_{\text{DT}}^{(r)}$. Then, the relay transmits the remaining $|i - j|$ packets to the corresponding destination depending on the relation of i and j, that is, if $i > j$, then $\mathcal{R}$ transmits $(i - j)$ packets to $\mathcal{D}_1$, otherwise $\mathcal{R}$ transmits $(j - i)$ packets to $\mathcal{D}_2$. With these packets, the number of transmissions is given by $|i - j|n_{rd_a}$. Thus, n_{rd_a} and $n_{\text{DT}}^{(r)}$ are, respectively, given by

$$n_{rd_a} = \frac{1}{1 - p_{rd_a}}, \tag{12}$$

$$n_{\text{DT}}^{(r)} = \frac{1}{1 - p_{rd_1}} + \frac{1}{1 - p_{rd_2}} - \frac{1}{1 - p_{rd_1} p_{rd_2}}. \tag{13}$$

In the third step where $\mathcal{R}$ fails to receive $(N - k)$ packets in the first step, $\mathcal{S}_i$ is required to retransmit with the number of transmissions

$$E\left[n_{\text{NCA}}^{(s_{i_s}, 3)} \mid K = k\right]$$
$$= \sum_{i=0}^{N-k} \sum_{j=0}^{N-k} C_i^{N-k} p_{s_{i_s} d_1}^i \left(1 - p_{s_{i_s} d_1}\right)^{N-k-i}$$
$$\times C_j^{N-k} p_{s_{i_s} d_2}^j \left(1 - p_{s_{i_s} d_2}\right)^{N-k-j} \tag{14}$$
$$\times [\min\{i, j\} n_{\text{RT}} + |i - j| \, n_{rd_a}],$$

where $i_s \in \{1, 2\}$, $a = 1$ if $i > j$, and $a = 2$ otherwise. n_{RT} and n_{rd_a} are given by (5) and (12), respectively.

3.4. Proposed Protocol.
The relay in the proposed protocol combines the lost packets of different flows. Since the total of $2N$ packets is transmitted from $\mathcal{S}_1$ and $\mathcal{S}_2$, the transmission bandwidth is expressed as

$$n = \frac{n^{(1)} + n^{(2)} + n^{(3)}}{2N}, \tag{15}$$

where $n^{(i)}$ denotes the number of transmissions in the ith step of the proposed protocol including the following steps.

Step 1. Each of $\mathcal{S}_1$ and $\mathcal{S}_2$ transmits N packets.

Step 2. $\mathcal{R}$ retransmits Type-I packets.

Step 3. $\mathcal{S}_1$ and/or $\mathcal{S}_2$ retransmit Type-II packets.

In Step 1, both $\mathcal{S}_1$ and $\mathcal{S}_2$ send N packets to $\mathcal{R}$, $\mathcal{D}_1$, and $\mathcal{D}_2$, and thus

$$n^{(1)} = 2N. \tag{16}$$

The number of transmissions in Step 2 and Step 3 are given by

$$n^{(2)} = \sum_{k=0}^{N} \left\{ C_k^N p_{s_1 r}^{N-k} \left(1 - p_{s_1 r}\right)^k \right.$$
$$\times p_{s_2 r}^{N-k} \left(1 - p_{s_2 r}\right)^k E\left[n^{(2)} \mid K = k\right]$$
$$+ \sum_{l=0}^{N-k} \left\{ C_l^{N-k} p_{s_1 r}^{N-k-l} \left(1 - p_{s_1 r}\right)^l \right.$$
$$\times p_{s_2 r}^l \left(1 - p_{s_2 r}\right)^{N-k-l} E\left[n^{(2)} \mid L = l\right]$$
$$+ \sum_{m=0}^{N-k-l} \left\{ C_m^{N-k-l} p_{s_1 r}^m \left(1 - p_{s_1 r}\right)^{N-k-l-m} \right.$$
$$\times \left(1 - p_{s_2 r}\right)^m p_{s_2 r}^{N-k-l-m}$$
$$\left. \left. \left. \times E\left[n^{(2)} \mid M = m\right] \right\} \right\} \right\}, \tag{17}$$

$$n^{(3)} = \sum_{k=0}^{N} \left\{ C_k^N p_{s_1 r}^{N-k} \left(1 - p_{s_1 r}\right)^k \right.$$
$$\times p_{s_2 r}^{N-k} \left(1 - p_{s_2 r}\right)^k E\left[n^{(3)} \mid K = k\right]$$
$$+ \sum_{l=0}^{N-k} \left\{ C_l^{N-k} p_{s_1 r}^{N-k-l} \left(1 - p_{s_1 r}\right)^l p_{s_2 r}^l \left(1 - p_{s_2 r}\right)^{N-k-l} \right.$$
$$\times E\left[n^{(3)} \mid L = l\right]$$
$$+ \sum_{m=0}^{N-k-l} \left\{ C_m^{N-k-l} p_{s_1 r}^m \left(1 - p_{s_1 r}\right)^{N-k-l-m} \right.$$
$$\times \left(1 - p_{s_2 r}\right)^m p_{s_2 r}^{N-k-l-m}$$
$$\left. \left. \left. \times E\left[n^{(3)} \mid M = m\right] \right\} \right\} \right\}, \tag{18}$$

respectively, where K, L, M are random variables representing the number of packets that $\mathcal{R}$ successfully receives from Ω_1, Ω_2, and Ω_3, respectively.

Given that $K = k$ packets are successfully received at $\mathcal{R}$ in the first group, the number of transmissions at $\mathcal{R}$ based on the proposed algorithm (i.e., **Algorithm 1**) for the packets in Ω_1 in the second step can be computed by

$$E\left[n^{(2)} \mid K = k\right]$$
$$= \sum_{i=0}^{k} \sum_{j=0}^{k} \sum_{u=0}^{k} \sum_{v=0}^{k} C_i^k p_{s_1 d_1}^i \left(1 - p_{s_1 d_1}\right)^{k-i}$$
$$\times C_j^k p_{s_2 d_1}^j \left(1 - p_{s_2 d_1}\right)^{k-j} C_u^k p_{s_1 d_2}^u \left(1 - p_{s_1 d_2}\right)^{k-u}$$

$$\times C_v^k p_{s_2 d_2}^v (1 - p_{s_2 d_2})^{k-v}$$

$$\times \left[\min\{i+j, u+v\} n_{\mathrm{DT}}^{(r)} + |(i+j) - (u+v)| n_{r d_a} \right],$$

$$(19)$$

where $n_{\mathrm{DT}}^{(r)}$ is given by (13), $n_{r d_a}$ is given by (12) with $a = 1$ if $i + j > u + v$, and $a = 2$ otherwise.

For the packets in Ω_2 and Ω_3 in Step 2, the number of transmissions is calculated by

$$E\left[n^{(2)} \mid L = l \right]$$

$$= \sum_{i=0}^{l} \sum_{j=0}^{l} C_i^l p_{s_1 d_1}^i (1 - p_{s_1 d_1})^{l-i} C_j^l p_{s_1 d_2}^j (1 - p_{s_1 d_2})^{l-j} \quad (20)$$

$$\times \left[\min\{i, j\} n_{\mathrm{DT}}^{(r)} + |i - j| n_{r d_a} \right],$$

$$E\left[n^{(2)} \mid M = m \right]$$

$$= \sum_{i=0}^{m} \sum_{j=0}^{m} C_i^m p_{s_2 d_1}^i (1 - p_{s_2 d_1})^{m-i} C_j^m p_{s_2 d_2}^j (1 - p_{s_2 d_2})^{m-j}$$

$$\times \left[\min\{i, j\} n_{\mathrm{DT}}^{(r)} + |i - j| n_{r d_a} \right],$$

$$(21)$$

respectively, where $a = 1$ if $i > j$, and $a = 2$ otherwise.

In Step 3 where the relay fails to receive packets of the first group in the first step, the sources are required to retransmit these remaining lost packets with the number of transmissions

$$E\left[n^{(3)} \mid K = k \right]$$

$$= \sum_{i=0}^{N-k} \sum_{j=0}^{N-k} \sum_{u=0}^{N-k} \sum_{v=0}^{N-k} C_i^{N-k} p_{s_1 d_1}^i (1 - p_{s_1 d_1})^{N-k-i}$$

$$\times C_j^{N-k} p_{s_2 d_1}^j (1 - p_{s_2 d_1})^{N-k-j} C_u^{N-k} p_{s_1 d_2}^u$$

$$\times (1 - p_{s_1 d_2})^{N-k-u} C_v^{N-k} p_{s_2 d_2}^v (1 - p_{s_2 d_2})^{N-k-v}$$

$$\times \left[\min\{i+j, u+v\} n_{\mathrm{RT}} + |(i+j) - (u+v)| n_{\mathrm{RT}}^{(d_a)} \right],$$

$$(22)$$

where $a = 1$ if $i + j > u + v$, and $a = 2$ otherwise. n_{RT} and $n_{\mathrm{RT}}^{(d_a)}$ are given by (5) and (3), respectively.

For the second group and the third group in Step 3, the number of transmissions is computed by

$$E\left[n^{(3)} \mid L = l \right]$$

$$= \sum_{i=0}^{N-k-l} \sum_{j=0}^{N-k-l} C_i^{N-k-l} p_{s_1 d_1}^i (1 - p_{s_1 d_1})^{N-k-l-i}$$

$$\times C_j^{N-k-l} p_{s_1 d_2}^i (1 - p_{s_1 d_2})^{N-k-l-j}$$

$$\times \left[\min\{i, j\} n_{\mathrm{RT}}^{(s_1)} + |i - j| n_{\mathrm{RT}}^{(s_1, d_a)} \right],$$

$$(23)$$

$$E\left[n^{(3)} \mid M = m \right]$$

$$= \sum_{i=0}^{N-k-l-m} \sum_{j=0}^{N-k-l-m} C_i^{N-k-l-m} p_{s_2 d_1}^i (1 - p_{s_2 d_1})^{N-k-l-m-i}$$

$$\times C_j^{N-k-l-m} p_{s_2 d_2}^j (1 - p_{s_2 d_2})^{N-k-l-m-j}$$

$$\times \left[\min\{i, j\} n_{\mathrm{RT}}^{(s_2)} + |i - j| n_{\mathrm{RT}}^{(s_2, d_a)} \right],$$

$$(24)$$

respectively, where $n_{\mathrm{RT}}^{(s_i, d_a)}$, $i = 1, 2$, is given by (4); $a = 1$ if $i > j$, and $a = 2$ otherwise. In the above (23) and (24), $n_{\mathrm{RT}}^{(s_i)}$, $i = 1, 2$, denotes the number of transmissions to transmit data from $\mathcal{S}_i$ to both $\mathcal{D}_1$ and $\mathcal{D}_2$ through $\mathcal{R}$ that can be computed by

$$n_{\mathrm{RT}}^{(s_i)} = \frac{1}{1 - p_{s_i r} p_{s_i d_1} p_{s_i d_2}}$$

$$\times \left[1 + p_{s_i r} p_{s_i d_1} (1 - p_{s_i d_2}) n_{\mathrm{RT}}^{(s_i, d_1)} \right.$$

$$+ p_{s_i r} (1 - p_{s_i d_1}) p_{s_i d_2} n_{\mathrm{RT}}^{(s_i, d_2)}$$

$$+ (1 - p_{s_i r}) p_{s_i d_1} (1 - p_{s_i d_2}) n_{r d_1}$$

$$+ (1 - p_{s_i r})(1 - p_{s_i d_1}) p_{s_i d_2} n_{r d_2}$$

$$\left. + (1 - p_{s_i r}) p_{s_i d_1} p_{s_i d_2} n_{\mathrm{DT}}^{(r)} \right],$$

$$(25)$$

where $n_{\mathrm{DT}}^{(r)}$ is given by (13).

4. Numerical Results

In this section, we compare the transmission bandwidth of different protocols considered in our work by analytically evaluating the expressions in Section 3. In fact, the simulation and analytical results drawn in Figure 4 demonstrate a strong agreement. Consequently, it is sufficient to show the analytical results in Figures 5–7.

Figure 4 plots the transmission bandwidth of various ARQ protocols versus $p_{s_1 r}$, the packet error rate (PER) of the wireless link from $\mathcal{S}_1$ to $\mathcal{R}$. Both numerical and analytical results are included. The range of $p_{s_1 r}$ is from 0.04 to 0.2 to characterize a wide range of wireless applications. To study the effect of the channels from the sources to the relay on the overall performance, we assume that $p_{s_1 r} = p_{s_2 r}$. The value of other PERs is arbitrarily set at $p_{s_1 d_1} = p_{s_2 d_2} = 0.2$, $p_{s_1 d_2} = p_{s_2 d_1} = 0.3$, and $p_{r d_1} = p_{r d_2} = 0.1$. We can see that the proposed protocol outperforms other existing schemes since it can combine the lost packets from different flows in the retransmission phase. In particular, when $p_{s_1 r}$ is small (e.g., $p_{s_1 r} = 0.04$), the proposed scheme shows a remarkable gain over the traditional ARQ method. In fact,
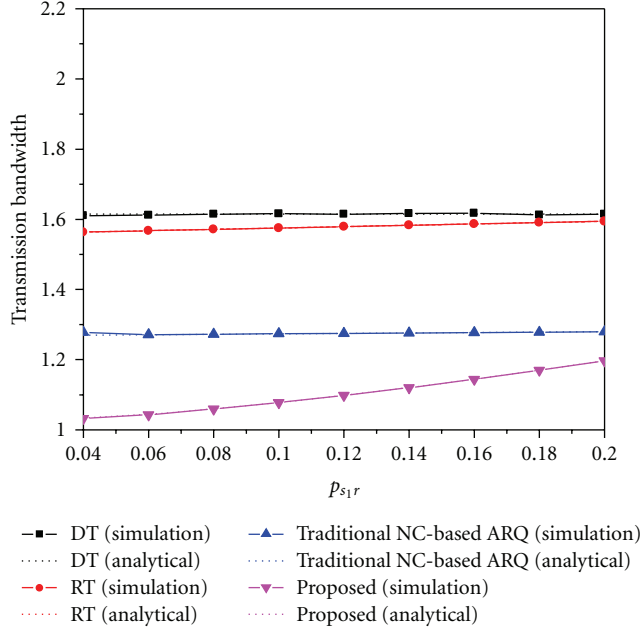
FIGURE 4: Transmission bandwidth of different protocols over $p_{s_1 r} = p_{s_2 r}$ with $p_{s_1 d_1} = p_{s_2 d_2} = 0.2$, $p_{s_1 d_2} = p_{s_2 d_1} = 0.3$, and $p_{rd_1} = p_{rd_2} = 0.1$.

FIGURE 5: Transmission bandwidth comparison of traditional NC-based ARQ and our proposed protocol for different values of $p_{s_i d_j}$.

if $p_{s_1 r}$ is small, we have more Type-I packets in Ω_1 than in Ω_2 and Ω_3. For packets in Ω_1, our proposed scheme can save the number of retransmissions by mixing the packets from different flows. When $p_{s_1 r}$ is high (i.e., the channel from sources to relay is in bad condition), the relay mostly obtains the lost packets of Type-II. Consequently, the sources should retransmit these packets. In other words, there is little benefit in applying NC at the relay in this scenario. As a result, the performance of our proposed protocol converges to that of the traditional approach. Additionally, it can be seen that the analytical results are quite matched with the simulation results, and thus, in the following, we only show the analytical results.

Figure 5 compares the transmission bandwidth of our proposed protocol with that of the traditional NC-based ARQ protocol for different values of PER of the channels from sources to destinations. The simulation setting is similar to that in Figure 4. For a given value of $p_{s_i d_j}$, our proposed approach always shows better performance than the traditional NC-based ARQ protocol. What is particular in Figure 5 is that the transmission bandwidth of our proposed protocol converges to a certain value at low $p_{s_1 r}$ regardless of $p_{s_1 d_1}$ and $p_{s_1 d_2}$. That means, when $p_{s_1 r}$ is small, the reliability of the channels from sources to destinations has little impact on the transmission bandwidth of the proposed protocol. In fact, a small value of $p_{s_1 r}$ denotes the case where $\mathcal{R}$ certainly detected the packets from $\mathcal{S}_1$ and $\mathcal{S}_2$ successfully. As a result, the lost packets at the destinations belong to Type-I. Thus, the relay handles the retransmission of these packets, and the impact of the direct channels from sources to destinations is negligible.
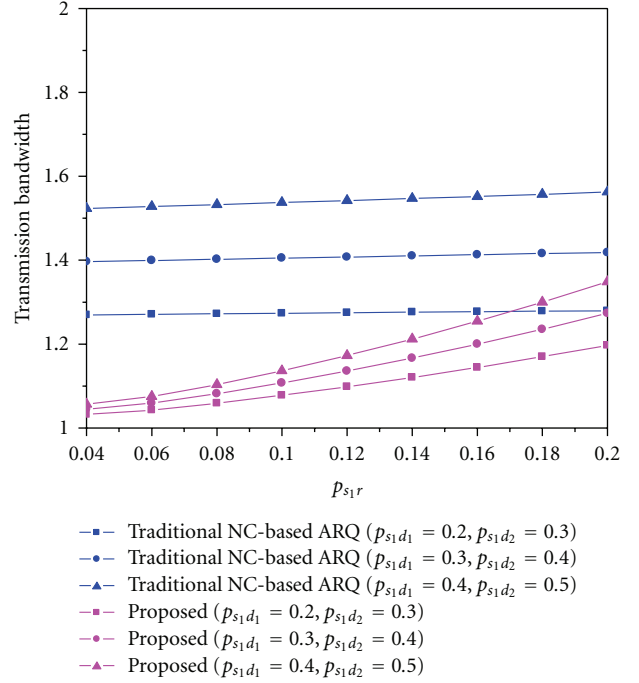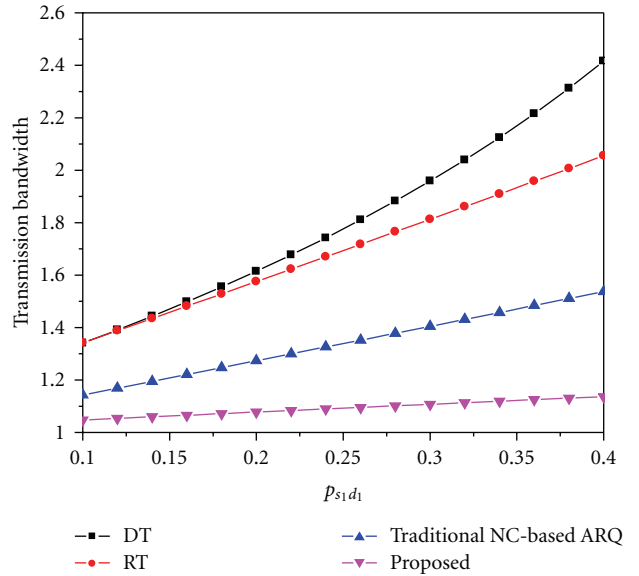

FIGURE 6: Transmission bandwidth of different protocols over $p_{s_1 d_1} = p_{s_2 d_2}$ with $p_{s_1 d_2} = p_{s_2 d_1} = p_{s_1 d_1} + 0.1$, $p_{s_1 r} = p_{s_2 r} = 0.1$, and $p_{rd_1} = p_{rd_2} = 0.1$.

In Figures 6 and 7, we show the transmission bandwidth of various ARQ protocols with respect to $p_{s_1 d_1}$. The PERs of the channels from the sources to the relay and from the relay to the destinations are fixed. The values of $p_{s_1 d_2}$ and $p_{s_2 d_1}$ are adjusted accordingly to $p_{s_1 d_1}$ by the relation $p_{s_1 d_2} = p_{s_2 d_1} = p_{s_1 d_1} + 0.1$. It is observed that the transmission bandwidth curve of the proposed protocol has the smallest slope. Thus,
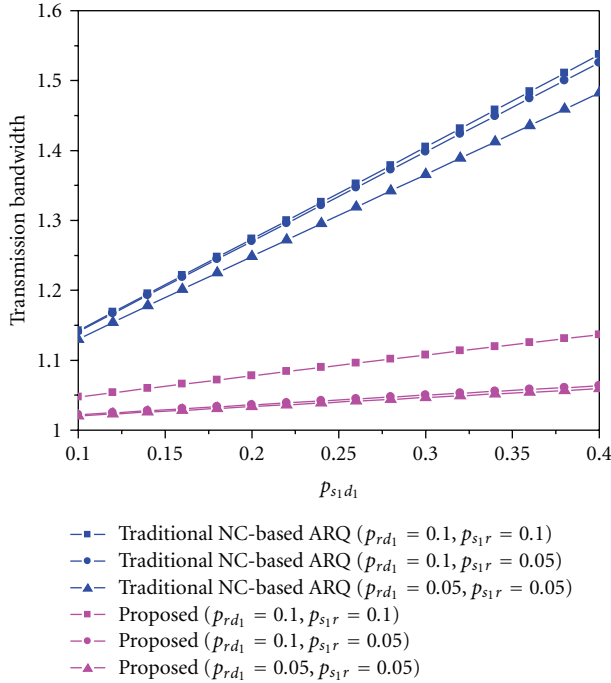
FIGURE 7: Transmission bandwidth of traditional NC-based ARQ and our proposed protocol over $p_{s_1 d_1} = p_{s_2 d_2}$ with $p_{s_1 d_2} = p_{s_2 d_1} = p_{s_1 d_1} + 0.1$ and different $p_{s_1 r} = p_{s_2 r}$, $p_{rd_1} = p_{rd_2}$.

we conclude that the performance of our developed ARQ scheme is not sensitive to the quality of the channels from the sources to the destinations. The channels $\mathcal{S}_i \to \mathcal{R}$ and $\mathcal{R} \to \mathcal{D}_i$ are more important than the channels $\mathcal{S}_i \to \mathcal{D}_j$.

5. Conclusion

In this paper, we propose a new reliable transmission scheme for wireless multisource-multicast networks based on NC. For a specific case of multisource-multicast networks with two sources and two destinations, we present two packet-combination algorithms to retransmit the lost packets efficiently. The transmission bandwidth of various ARQ protocols is analyzed. Furthermore, we provide numerical results with different simulation settings to demonstrate the effectiveness of our proposed scheme in saving the transmission bandwidth. For future works, one could possibly investigate the performance of fading channels with path loss and placement of the nodes.

References

[1] A. Sendonaris, E. Erkip, and B. Aazhang, "User cooperation diversity—part I: system description," *IEEE Transactions on Communications*, vol. 51, no. 11, pp. 1927–1938, 2003.

[2] J. N. Laneman, D. N. C. Tse, and G. W. Wornell, "Cooperative diversity in wireless networks: efficient protocols and outage behavior," *IEEE Transactions on Information Theory*, vol. 50, no. 12, pp. 3062–3080, 2004.

[3] B. Nazer and M. Gastpar, "Compute-and-forward: harnessing interference through structured codes," in *Proceedings of IEEE International Symposium on Information Theory (ISIT '08)*, pp. 772–776, Toronto, Ontario, Canada, July 2008.

[4] R. Ahlswede, N. Cai, S. Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1204–1216, 2000.

[5] R. Koetter and M. Médard, "An algebraic approach to network coding," *IEEE/ACM Transactions on Networking*, vol. 11, no. 5, pp. 782–795, 2003.

[6] S. Katti, D. Katabi, W. Hu, H. Rahul, and M. Medard, "The importance of being opportunistic: practical network coding for wireless environments," in *Proceedings of the 43rd Annual Allerton Conference on Communication, Control and Computing (Allerton '05)*, Montecillo, Ill, USA, September 2005.

[7] S. Zhang, S. C. Liew, and P. P. Lam, "Hot topic: physical-layer network coding," in *Proceedings of the 12th Annual International Conference on Mobile Computing and Networking (MOBICOM '06)*, pp. 358–365, New York, NY, USA, September 2006.

[8] S. Katti, S. Gollakota, and D. Katabi, "Embracing wireless interference: analog network coding," in *Proceedings of the ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '07)*, pp. 397–408, Kyoto, Japan, August 2007.

[9] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, "XORs in the air: practical wireless network coding," *IEEE/ACM Transactions on Networking*, vol. 16, no. 3, pp. 497–510, 2008.

[10] P. Larsson, "Multicast multiuser ARQ," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '08)*, pp. 1985–1990, Las Vegas, Nev, USA, March 2008.

[11] S. S. L. Chang, "Theory of information feedback systems," *IEEE Transactions on Information Theory*, vol. 2, no. 3, pp. 29–40, 1956.

[12] P. Fan, C. Zhi, C. Wei, and K. B. Letaief, "Reliable relay assisted wireless multicast using network coding," *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 5, pp. 749–762, 2009.

[13] D. Nguyen, T. Tran, T. Nguyen, and B. Bose, "Wireless broadcast using network coding," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 2, pp. 914–925, 2009.

[14] S. Y. R. Li, R. W. Yeung, and N. Cai, "Linear network coding," *IEEE Transactions on Information Theory*, vol. 49, no. 2, pp. 371–381, 2003.

[15] S. Verdu, *Multiuser Detection*, Cambridge University Press, Cambridge, UK, 1998.

[16] W. J. Huang, Y. W. Peter Hong, and C. C. Jay Kuo, "Relay-assisted decorrelating multiuser detector (RAD-MUD) for cooperative CDMA networks," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 3, pp. 550–560, 2008.

[17] R. Lupas and S. Verdu, "Linear multiuser detectors for synchronous code-division multiple-access channels," *IEEE Transactions on Information Theory*, vol. 35, no. 1, pp. 123–136, 1989.

[18] A. Duel-Hallen, "Decorrelating decision-feedback multiuser detector for synchronous code-division multiple-access channel," *IEEE Transactions on Communications*, vol. 41, no. 2, pp. 285–290, 1993.

[19] P. Patel and J. Holtzman, "Analysis of a simple successive interference cancellation scheme in a DS/CDMA system," *IEEE Journal on Selected Areas in Communications*, vol. 12, no. 5, pp. 796–807, 1994.