This paper was presented as part of the main technical program at IEEE INFOCOM 2011

Optimizing Network-Coded Cooperative Communications via Joint Session Grouping and Relay Node Selection

Sushant Sharma⋆ Yi Shi⋆ Y. Thomas Hou⋆ Hanif D. Sherali⋆ Sastry Kompella†
⋆Virginia Polytechnic Institute and State University, Blacksburg, VA, USA
†U.S. Naval Research Laboratory, Washington D.C., USA

***Abstract*—Network-coded cooperative communications (NC-CC) is a new paradigm in wireless networks that employs network coding (NC) to improve the performance of CC. The core mechanism to harness the benefits of NC-CC is to appropriately combine sessions into separate groups, and then have each group select the most beneficial relay node for NC-CC. In this paper, we study this joint grouping and relay node selection problem for NC-CC. Due to NP-hardness of problem, we propose a distributed and online algorithm that offers near-optimal solution to this problem. The key idea in our algorithm is to have each neighboring relay node of a new session determine and offer its best local group; and then to have the source node of the new session select the best group among all offers. We show that our distributed algorithm has polynomial complexity. Using extensive numerical results, we show that our distributed algorithm adapts well to online network dynamics.**

I. Introduction

Network-coded cooperative communications (NC-CC) [3], [19], [20], [24], [25] is a new paradigm in wireless networks that employs network coding (NC) to improve the performance of CC. To see how NC-CC works, let's first understand the potential issue with CC in a multi-session network and see how NC can help.

- Under CC, a source node exploits its neighboring node to relay data and to achieve path diversity and possible gain in achievable rate [13]. In the simple three-node model [17], [22] shown in Fig. 1, the source node s_0 can exploit a neighboring relay node r when it sends data to its destination node d_0. Here, a time frame is divided into two slots. In the first time slot, the source node s_0 transmits data to node d_0, which is also overheard by r. In the second time slot, r re-transmits its overheard signal to d_0 (with amplify-and-forward (AF), decode-and-forward (DF), compress-and-forward (CF), or other schemes [13]). The destination node can now combine these two signals from different paths. It was shown in [17] that such CC scheme can improve the achievable rate of the channel over direct transmission.
- When the same relay node is being used by multiple sessions, say N_s sessions, one would divide the time frame into $2N_s$ time slots, as shown in Fig 2(a). Note than among the $2N_s$ time slots, N_s time slots are used for relaying data for each of the N_s sessions. This is clearly wasteful, and is precisely the place where NC can be leveraged to improve efficiency. It was recognized in [19], [20] that by using NC's capability to combine/aggregate signals inside the network, one can consolidate the N_s time slots used for relaying into just a single time slot as shown in Fig. 2(b). Here, a frame is divided into only $N_s + 1$ time slots. The first N_s time slots are used for transmission by each of the N_s source nodes. Then, the relay node combines all the signals it overhears in the previous N_s time slots and transmits the combined signal in the (N_s+1)-th time slot. This combined signal is then received by all the destination nodes, which can subtract the unwanted signals that were overheard in the first N_s time-slots, thereby extracting their desired signal [20]. The potential benefits of using NC-CC are two-fold. First, the number of time-slots used by the relay node is reduced to one (from N_s), which significantly increases time slot efficiency. Second, due to the reduction in the number of time slots, the bandwidth of each time slot for transmission is increased.

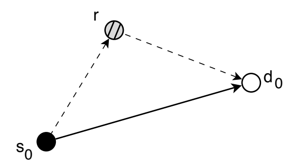
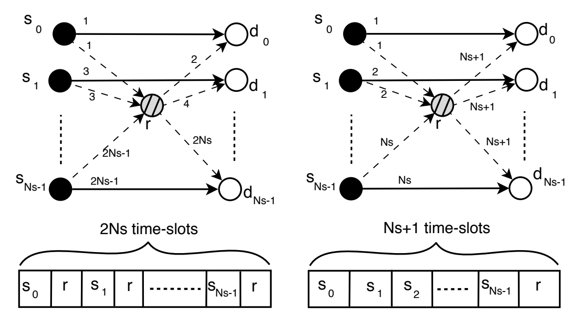


Fig. 1. A reference model for three-node CC.



(a) Using CC without NC. (b) Using CC with NC.

Fig. 2. An example illustrating how NC-CC may improve over CC.

Based on our discussion so far, it may appear that for a single relay node, we can group as many sessions as we want. But, in a recent study [20], Sharma *et al.* showed that there exists a so-called "NC noise" at a destination node when

978-1-4244-9921-2/11/$26.00 ©2011 IEEE

extracting the desired signal from the network coded signal. Further, it was shown that as the group size (i.e., the number of sessions in a group) increases, the NC noise also increases, thereby decreasing the achievable rates. That is, there exists a trade-off between the time slot efficiency and the NC noise. As a result, instead of grouping all the sessions in a single group, it may be necessary to put sessions into different groups in order to keep NC noise under control. However, how to perform session grouping is not a trivial problem.

In this paper, we are interested in a more general setting where there are multiple relay nodes in the network. In this setting, a session has the option to select a relay node from different available relay nodes. So, in addition to session grouping, we also have a relay node selection problem. We study a join session grouping and relay node selection problem in NC-CC. The goal is to maximize the sum of weighted session rates in the network. Our main contributions can be described as follows. To solve this joint problem, we propose a distributed and online algorithm called D-GRS (Distributed Grouping and Relay node Selection). We show that D-GRS has polynomial complexity. Using extensive numerical results, we show that D-GRS can offer near-optimal solutions when compared to a centralized solver (CPLEX). Further, it adapts well to online network dynamics.

The rest of this paper is organized as follows. In Section II, we describe the joint session grouping and relay selection problem. In Section III, we propose our D-GRS algorithm to solve the joint session grouping and relay node selection problem. Section IV presents simulation results to demonstrate the performance and time complexity of D-GRS. In Section V, we discuss related work, and Section VI concludes this paper.

II. Problem Statement

In this section, we first give some mathematical background for achievable rate calculation in NC-CC (Section II-A). Then in Section II-B, we describe the joint session grouping and relay node selection problem.

A. Preliminaries

We start with the simple case where all sessions are in the same group and share the same relay node. By identifying the NC noise, we introduce the grouping mechanism and discuss the case where sessions sharing the same relay node can be put into different groups. Finally, we consider the general case where there are multiple relay nodes in the network.

The Single-Group Single-Relay Case. Consider the simple case in Fig. 2(b) where all sessions share the same relay node. Denote $\mathcal{S}_r$ the set of source nodes $\{s_0, s_1, \cdots, s_{N_s-1}\}$ for all sessions in the network. Denote W (in Hz) the total bandwidth. Let h_{uv} capture the effect of path-loss, shadowing, and fading within the channel between two nodes u and v. Denote P_u as the transmission power at node u. Assume the background noise at node v has zero mean and a variance of σ_v^2. Denote SNR_{uv} the signal-to-noise ratio at receiving node v (for the signal from node u).

Under this setting, the achievable rate for a session (s_i, d_i) can be written as [20]

$$R_{\text{NC-CC}}(s_i, r, \mathcal{S}_r) = \frac{W}{N_s + 1} \cdot I_{\text{NC-CC}}(s_i, r, \mathcal{S}_r), \quad (1)$$

where $I_{\text{NC-CC}}(s_i, r, \mathcal{S}_r)$ is the mutual information between nodes s_i and d_i. Under analog NC with AF CC, $I_{\text{NC-CC}}(s_i, r, \mathcal{S}_r)$ can be written as [20]

$$I_{\text{NC-CC}}(s_i, r, \mathcal{S}_r) = \log_2 \Bigg(1 + \text{SNR}_{s_i d_i} + \frac{\text{SNR}_{s_i r}\ \text{SNR}_{r d_i}}{|\mathcal{S}_r| \frac{\sigma^2_{z^{NC}_{d_i}}}{\sigma^2_{d_i}} + \text{SNR}_{r d_i} + \frac{\sigma^2_{z^{NC}_{d_i}}}{\sigma^2_{d_i}} \sum_{s_k \in \mathcal{S}_r} \text{SNR}_{s_k r}} \Bigg), \quad (2)$$

where $\text{SNR}_{s_i d_i} = \frac{P_{s_i}}{\sigma^2_{d_i}} |h_{s_i d_i}|^2$, $\text{SNR}_{s_i r} = \frac{P_{s_i}}{\sigma^2_r} |h_{s_i r}|^2$, $\text{SNR}_{r d_i} = \frac{P_{r_j}}{\sigma^2_{d_i}} |h_{r d_i}|^2$, and $\sigma^2_{z^{NC}_{d_i}}$ denotes the variance of the NC noise at node d_i. The value of $\sigma^2_{z^{NC}_{d_i}}$ is given in [20] and can be written as

$$\sigma^2_{z^{NC}_{d_i}} = \sigma^2_{d_i} + (|\mathcal{S}_r| - 1)(\alpha_r h_{r d_i})^2 \sigma^2_r + \sigma^2_{d_i} \sum_{s_k \in \mathcal{S}_r}^{s_k \neq s_i} \left(\frac{\alpha_r h_{s_k r} h_{r d_i}}{h_{s_k d_i}} \right)^2, \quad (3)$$

where α_r is the amplification factor at the relay node r and is

$$\alpha_r^2 = \frac{P_r}{|\mathcal{S}_r| \sigma^2_r + \sum_{s_i \in S_r} P_{s_i} |h_{s_i r}|^2}. \quad (4)$$

A closer look at the NC noise in (3) shows that, as more sessions share the same relay node (i.e., $|\mathcal{S}_r|$ increases), the NC noise also increases monotonically. Further, as NC noise increases, the value of mutual information for each session (s_i, d_i) in (2) decreases.

The Multi-Group Single-Relay Case. Recognizing the above NC noise problem associated with a single group, one can introduce multiple groups to control the NC noise. This is illustrated in an example in Fig. 3. Here, instead of putting all six sessions in the same group, one can put them into three separate groups. From (3), we find that NC noise is directly tied to the number of sessions in group $\mathcal{S}_r$. When the number of sessions in a group is reduced, the NC noise for the sessions in the group is also reduced. As a result, the value of mutual information in (2) will increase.

To support multiple groups sharing the same relay node, we need to re-organize time slot structure in a frame. Figure 3(d) shows the proposed time slot structure for multiple groups sharing the same relay. Time slot structure of Fig. 3(d) is based on that in Fig. 3(b), where each session is allocated equal time for direct transmission (i.e., NC-CC is not used). Since s_0, s_1, and s_2 are now in one group, the total time available to them is $3t$. Under NC-CC, as one additional time slot is needed for the relay node, we divide $3t$ into 4 equal-sized time slots, and thus the length of each time slot is $3t/4$. Following the same token, each time slot for $\mathcal{G}_1 = \{s_3, s_4\}$ is $2t/3$. Finally, each time slot for $\mathcal{G}_2 = \{s_5\}$ is $t/2$.

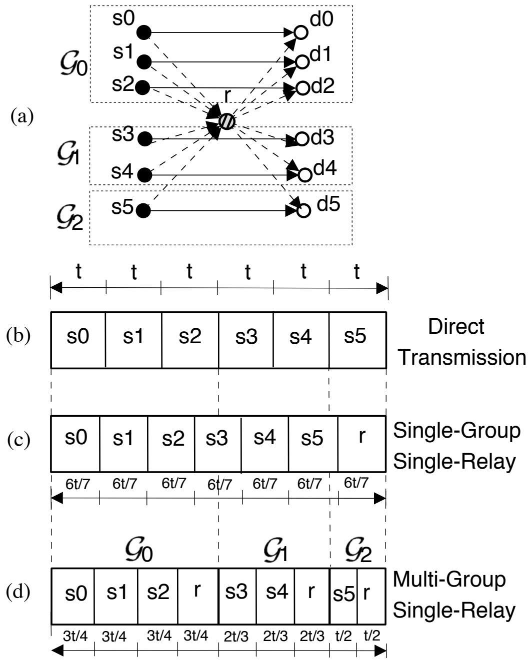


Fig. 3. An example illustrating the time slot structures for single-group single-relay case and multi-group single-relay case in NC-CC.

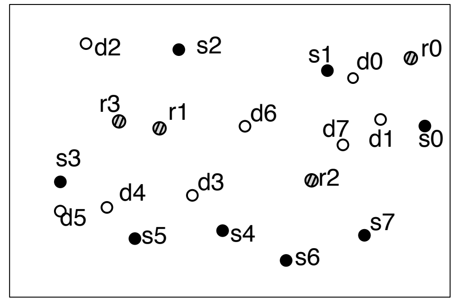
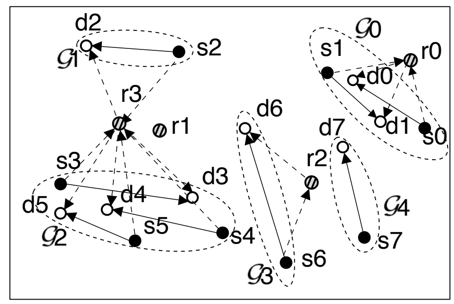
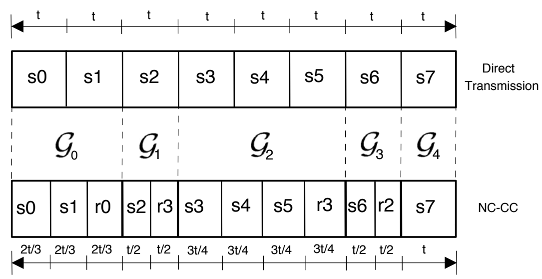


Fig. 4. A 20-node network.



(a) A possible solution.



(b) Time slot structure.

Fig. 5. An example showing grouping and relay node selection.

A naive approach to structure time slots in Fig. 3(d) is to set equal time slot size across all groups, i.e., $6t/9$ (or $2t/3$) for each time slot. We argue this is not a fair way to allocate time, as the total time allocated to $\mathcal{G}_0 = \{s_0, s_1, s_2\}$ will be $4 \cdot 2t/3 = 8t/3$, which is less than its fair share of $3t$ in Fig. 3(b). Likewise, s_5 will have a total time of $2 \cdot 2t/3 = 4t/3$, which is greater than its fair share of t.

We now show that this multi-grouping mechanism affects the achievable rate for a session (s_i, d_i) in a group. Denote $\mathcal{G}_r^{s_i}$ the group that contains s_i and uses relay node r for NC-CC. Under our approach, there are a total of $|\mathcal{G}_r^{s_i}|$ time slots (each of size t) for this group. Then the size of each time slot for this group under NC-CC will be $\frac{|\mathcal{G}_r^{s_i}| \cdot t}{|\mathcal{G}_r^{s_i}|+1}$. The achievable rate for session (s_i, d_i) in this group is

$$R_{\text{NC-CC}}(s_i, r, \mathcal{G}_r^{s_i}) = \left(\frac{\frac{|\mathcal{G}_r^{s_i}| \cdot t}{|\mathcal{G}_r^{s_i}|+1}}{t} \right) \cdot \frac{W}{N_s} \cdot I_{\text{NC-CC}}(s_i, r, \mathcal{G}_r^{s_i})$$
$$= \frac{|\mathcal{G}_r^{s_i}|}{|\mathcal{G}_r^{s_i}|+1} \cdot \frac{W}{N_s} \cdot I_{\text{NC-CC}}(s_i, r, \mathcal{G}_r^{s_i}) \; . \quad (5)$$

Comparing (5) to (1), we find that when we use multiple groups, the effective session bandwidth $\left(\frac{|\mathcal{G}_r^{s_i}|}{|\mathcal{G}_r^{s_i}|+1} \cdot \frac{W}{N_s} \right)$ will always be less than the effective session bandwidth in single-group case $\left(\frac{W}{N_s+1} \right)$, as $|\mathcal{G}_r^{s_i}| < N_s$. This reduction in effective session bandwidth can also be observed by comparing the time slot size in Fig. 3(c) and Fig. 3(d). On one hand, multiple groups can increase a session's mutual information. But on the other hand, multiple groups also reduces the effective bandwidth for a session. Therefore, we have a trade-off between effective bandwidth and mutual information of a session. In light of this trade-off, there is a need to explore an optimal grouping for a given objective.

The Multi-Group Multi-Relay Case. Our previous discussion on multi-group single-relay case shows the significance of putting sessions into different groups even when there is only a single relay node. In general, there may be multiple relay nodes available in the network. In this case, in addition to the grouping problem, we also need to address the relay node selection problem. That is, we have a joint problem of session grouping and relay node selection. This is the focus of this paper.

We use the network in Fig. 4 to illustrate our problem. In this network, we have eight sessions $\{(s_0, d_0), (s_1, d_1), \cdots, (s_7, d_7)\}$ and four relay nodes $\{r_0, r_1, r_2, r_3\}$. All nodes are within the interference range of each other and therefore simultaneous transmissions by two or more sessions are not allowed. Figure 5(a) shows a possible grouping and relay node selection. In this solution, there are five groups: $\mathcal{G}_0 = \{s_0, s_1\}, \mathcal{G}_1 = \{s_2\}, \mathcal{G}_2 = \{s_3, s_4, s_5\}$, $\mathcal{G}_3 = \{s_6\}$, and $\mathcal{G}_4 = \{s_7\}$. The group $\mathcal{G}_0$ uses relay node r_0. Groups $\mathcal{G}_1$ and $\mathcal{G}_2$ both use the same relay node r_3. Group $\mathcal{G}_3$, which contains only source s_6, uses the relay node r_2, and group $\mathcal{G}_4$ with source node s_7 does not use any relay node (i.e., direct transmission). We can also see that relay node r_1 is not being used by any group. For the session grouping and relay node selection in Fig. 5(a), the time-slot structure in a frame is shown in the lower portion of Fig. 5(b). In Fig. 5(b), we also show the time-slot structure when only direct transmission is employed in the network. It is not hard to see there are many other possible ways to do grouping and relay node selection for this network.

In general, the achievable rate for session (s_i, d_i) in group $\mathcal{G}_{r_j}^{s_i}$ is

$$R_{\text{NC-CC}}(s_i, r_j, \mathcal{G}_{r_j}^{s_i}) = \frac{|\mathcal{G}_{r_j}^{s_i}|}{|\mathcal{G}_{r_j}^{s_i}|+1} \cdot \frac{W}{N_s}\ I_{\text{NC-CC}}(s_i, r_j, \mathcal{G}_{r_j}^{s_i}), \quad (6)$$

which is similar to (5), with the only difference being that r (single relay) is now replaced by r_j (one of the relays).

B. Problem Statement

Denote $\mathcal{N}_s = \{s_0, s_1, \cdots, s_{N_s-1}\}$ the set of source nodes, $\mathcal{N}_d = \{d_0, d_1, \cdots, d_{N_d-1}\}$ the set of destination nodes, and $\mathcal{N}_r = \{r_0, r_1, \cdots, r_{N_r-1}\}$ the set of relay nodes. We assume $N_s = N_d$ and all source and destination nodes are distinct.[1] Each source node is expected to transmit data to its destination node, either with or without the assistance of a relay node. Further, a session (or a group of sessions) may use at most one relay node for NC-CC.

We now define our objective function. A number of objective functions can be used for our problem. In this paper, we choose the objective of maximizing the sum of weighted data rates of all sessions, where the weight w_i for session (s_i, d_i) is a pre-defined constant. We can write the weighted rates for session (s_i, d_i) under NC-CC and direct transmission as

$$R_{\text{NC-CC}}^{w}(s_i, r_j, \mathcal{G}_{r_j}^{s_i}) = w_i \cdot\ R_{\text{NC-CC}}(s_i, r_j, \mathcal{G}_{r_j}^{s_i})\ , \quad (7)$$

$$R_{\text{D}}^{w}(s_i, d_i) = w_i \cdot \frac{W}{N_s} \cdot \log_2(1 + \text{SNR}_{s_i d_i})\ . \quad (8)$$

Our session grouping and relay node selection problem can now be formally defined as follows: How to put all the sessions into different groups at different relay nodes such that the sum of the weighted data rates for all the sessions is maximized?

Note that a solution to the above optimization problem does not exclude a session from employing direct transmission (e.g., $\mathcal{G}_4$ in Fig. 5(b)). Further, in the special case when a group using a relay node has only one session, then only CC is employed for that session (e.g., $\mathcal{G}_3$ in Fig. 5(b)). In other words, both direct transmission and CC without NC are allowed in our solution. As a result, an optimal solution cannot be worse than a solution that only employs CC (without NC) or direct transmission.

[1]In the case that a node is serving multiple roles, we can logically partition this node and visualize it as multiple nodes.

III. A Distributed Algorithm

In [21], we show that our joint session grouping and relay node selection (GRS) is an NP-hard problem. In this section, we present D-GRS, a distributed and online algorithm for the GRS problem that produces near-optimal results. By "online", we mean that network dynamics are unknown *a priori*. That is, sessions can join and leave the network as time progresses. Further, we allow a relay node to be active ("on") and inactive ("off") over time. The goal of D-GRS is to accomplish session grouping and relay node selection via local computation and distributed message exchange among the nodes so as to maximize the sum of the weighted rates of all sessions. In our distributed algorithm, we separate the control plane used for executing the D-GRS algorithm from the data plane used for data transport by the sessions. That is, we assume the execution of the D-GRS algorithm is done on a separate control channel, which is independent from the data frame carrying sessions' data.

In Section III-A, we describe the information that needs to be maintained at each source and relay node. In Section III-B, we give the description of the three core subroutines of D-GRS. Section III-C presents how D-GRS handles session arrivals and departures assuming the set of relay nodes are always active. In Section III-D, we show how D-GRS works in a setting where the relay nodes are also allowed to switch between active and inactive status over time. In Section III-E, we discuss the stability of D-GRS, and in Section III-F, we analyze D-GRS' complexity.

A. Information Maintained at Nodes

We first describe the information that needs to be maintained at each node.

Source node. Each source node s_i in the network maintains the following information.

- Channel state information (CSI) (i) between s_i and its destination node d_i[2], and (ii) between the source nodes of other sessions and d_i. Information in (ii) can be obtained by having d_i hear the other source nodes' transmissions over a time frame. Then d_i can inform its source node about this information. A source node needs this information so that it can forward this information to the relay node. The relay node in-turn will use this information to calculate the data rates that the relay node will include in its offers.
- The number of active sessions in the network, and the number of sessions in its current session group. This information is needed by the source to determine time slot structure. To acquire this information, the source node of a new session sends a broadcast request (`REQ--ACT-SESSIONS`). Upon hearing this request, one of the relay nodes will reply with this information (`RAS-REPLY`). Note that only one relay needs to reply. This can be achieved by setting a random timer at every

[2]A number of mechanisms can be employed to obtain this information. Discussion of these mechanisms is beyond the scope of this paper.

relay, and have those relay nodes refrain their response once they hear that some other relay node has already responded. For the active source nodes in the network, they also update the information regarding the number of active sessions in the network when they hear REQ-ACT-SESSIONS message.

Relay node. Every relay node in the network maintains the following information.

- (i) The CSI between the source nodes using this relay and this relay node, (ii) the CSI between the relay node and the destination nodes of the above source nodes, and (iii) the CSI maintained at the above source nodes. The information in (i) can be obtained when a new session initiates and starts direct transmission. The information in (ii) can be obtained from the destination nodes.
- The number of active sessions in the network, and the number of sessions in its local groups. This information is updated whenever a new session initiates or an existing session terminates, and after a session selects a group.

B. Core Subroutines

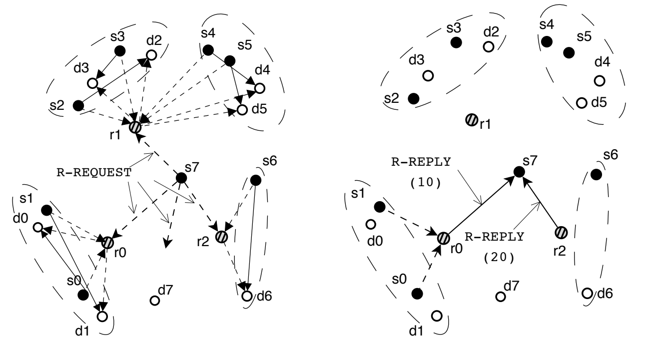
The core subroutines in D-GRS can be summarized as "SEEK-OFFER-SELECT" (or SOS), and can be described as follows.

SEEK. Initiated by a source node, which sends a broadcast request in the hope of finding a new/better relay node.

The source node seeks offers from relay nodes by broadcasting a message for relay node selection (R-REQUEST). The transmitted request includes the CSI information maintained at the source node. This information is required by the neighboring relay nodes to determine beneficial assignments. The message also includes the weight associated with this session, and its current data transmission rate. The source node will then wait to hear any offer from the relay nodes. This is the "SEEK" operation of SOS.

OFFER. Performed by the relay nodes. Each relay node makes an offer to the requesting source node regarding grouping based on its local computation.

When a relay node r_j receives the R-REQUEST from some source s_i, it will compose an offer for s_i. Initially, the relay node r_j uses the CSI information that it maintains to determine the data rate it can offer to session (s_i, d_i). The relay node r_j is aware of the groups of sessions it is currently supporting, and can determine local "candidate" groups for the new session (s_i, d_i). A candidate group for a relay node is defined as the one in which the weighted sum of achievable rates of all sessions is not less than that under direct transmission. Note that one candidate group could be an empty group (i.e., if the new session joins this group, only CC will be used). Now, the relay node determines the local group for (s_i, d_i) that has the potential to maximize the objective function. That is, the relay now considers the new group $\hat{\mathcal{G}}_k = \mathcal{G}_k + s_i$. Denote $\hat{U}_k$ as the sum of weighted session rates in $\hat{\mathcal{G}}_k$. Denote $w_i R_i$ the current weighted rate of s_i, which is available in the R-REQUEST message. Denote U_k the sum of weighted session rates in a



(a) Source node s_7 broadcasting the R-REQUEST message.

(b) Relay nodes r_2 and r_{10} sending back R-REPLY messages with OFFERs.

Fig. 6. An example illustrating core SOS operations.

local group $\mathcal{G}_k$ that is supported by r_j. Then by comparing $\hat{U}_k - U_k - w_i R_i$ among all local groups $\mathcal{G}_k$ at this relay node, the relay can identify the group that offers the largest gain, which we call LOCAL_GAIN.

This calculated LOCAL_GAIN is then included in the relay node's offer (R-REPLY) to the source s_i. In the case that the relay node cannot find any candidate group for (s_i, d_i), or if the LOCAL_GAIN is negative, then the relay node does not reply to the requesting source node. This completes the "OFFER" operation of SOS.

SELECT. The source node selects the best offer among the relay nodes.

To accomplish this, the source node waits for a prescribed time after transmitting the R-REQUEST message. Among all the offers that it receives, the source node of the session selects the relay node that offers the largest LOCAL_GAIN. The source node then transmits a message (CONFIRMATION) informing the relay node of its selection. Upon receiving this confirmation message, the time slot structure is updated accordingly. This completes the "SELECT" operation of SOS.

As an example, Fig. 6 shows the core SOS operations. Source node s_7 broadcasts an R-REQUEST message (SEEK). In reply (see Fig 6(b)), the relay nodes r_0 and r_2 offer LOCAL_GAINs of 10 and 20 respectively (OFFER); relay node r_1 does not reply because it does not find a candidate group for session (s_7, d_7). Finally, s_7 selects relay r_2 with the largest LOCAL_GAIN (SELECT).

In the rest of this section, we will show how D-GRS uses these core subroutines during different events.

C. Session Initiation or Termination

We first consider a network scenario where new session initiates or an existing session terminates in the network. Here, the set of relay nodes are assumed to remain active in the network. The case of a relay node's on/off behavior will be discussed in Section III-D.

A new session initiates. When a session (s_i, d_i) initiates, the source node s_i broadcasts a message requesting the number of active sessions in the network (REQ-ACT-SESSIONS). This request serves two purposes: (i) the relay nodes and other source nodes in the network will know about this new session, and can adjust their time slots appropriately to accommodate the new session, and (ii) the source s_i will get a response (RAS-REPLY) from one of the relay nodes, and will start direct transmission in its time slot based on the new frame structure. Meanwhile, upon hearing the REQ-ACT-SESSIONS, the corresponding destination node d_i starts to collect the CSI between the other active source nodes and itself. This CSI information is necessary so that the source node can begin the SEEK operation. After one time frame, the destination node of this new session will transfer the collected CSI information back to the new source node. The source node of this new session will broadcast a request message for relay nodes (R-REQUEST) (i.e., SEEK). Upon receiving this message, each relay node will find a best local group for this session to maximize the objective. Then the relay node will reply to the new source node with this information (i.e., OFFER). Upon receiving the replies from all the relay nodes, the source node selects the relay node with the best offer and sends a confirmation message (CONFIRMATION) to the selected relay node (i.e., SELECT). Subsequently, the new session joins the group in the chosen relay.

A session departs. When a session (s_i, d_i) departs, the source node of the session broadcasts a message (LEAVING) indicating its new status. Other sessions will update their knowledge about the number of sessions in the network, and can adjust their time slot structure accordingly.

Due to the session's departure, additional adjustments in the group of the departing session may be necessary to ensure that the remaining group remains a candidate group. The corresponding relay node r_j again calculates the new data rates for the remaining sessions in the group. If the weighted sum of new data rates of all the remaining sessions is above their weighted direct transmission rates, then no other operation is performed. Otherwise, the remaining group is considered inferior (i.e., not a candidate group). Relay node will now offload some sessions from the group (starting from the session with the largest rate drop below its direct transmission rate). To offload a session (s_k, d_k) from this group, the relay node will send a message (REMOVE-SRC) announcing to the source node s_k of this removal. Upon receiving this message, source node s_k falls back to direct transmission. The relay node will repeat this process for the remaining sessions in this group until the group becomes either a candidate group or empty.

The sessions that are being offloaded in the above process need to wait for a random amount of time before performing the core SOS operations to seek other relays. Note that this time, the SEEK operation (i.e., the R-REQUEST message) should contain a flag indicating that this request is from an ongoing session, instead of a new one. This flag is required to indicate that there is no change needed in the time slot structure, which is unlike the case of a new session.

D. Relay Activation and Deactivation

We now consider the scenario where the relay nodes can also become active and inactive as time progresses.

A relay node becomes active. Upon activation, the relay node broadcasts a message (REQ-ACT-SESSIONS). The purpose of this message is twofold: (i) to inform other sessions in the network regarding its activation, and (ii) to request information regarding the number of active sessions in the network. The information in (ii) is required to construct an OFFER in response to some SEEK request. Upon receiving this request, one of the active relay nodes will reply with the latter information (RAS-REPLY). Note that only one reply is needed. Other relay nodes can hear the first reply and then refrain from transmitting the same information again.

Upon hearing the activation of a new relay, each source will attempt to seek this new relay after a random amount of time if allowed. A source node s_i is allowed to seek the new relay node only if its current group remains a candidate group should the source leaves the current group. The source node will ask for this permission from its current relay node. This relay node may or may not grant such permission. Note that the permissions will be granted by a relay node in sequence to only one source node at a time. If the permission is granted, then the source node can start the SEEK operation. After a source node selects some relay node, it transmits a confirmation message (CONFIRMATION) to its new relay node. This confirmation message will be used to re-adjust the existing time slot structure.

A relay node becomes inactive. If a relay node r_j decides to go inactive, it broadcasts a message (R-LEAVING). The source nodes that are using r_j will adjust their time slots and fall back to direct transmission. Subsequently, these source nodes will wait for a random amount of time and then perform the three core SOS operations to seek beneficial relay nodes.

E. Stability

We now discuss the stability of our D-GRS algorithm. We show that the D-GRS algorithm remains stable under various situations.

A session departs. When a session departs, the relay node may decide to offload some of the remaining sessions one by one from the group of the departing session. While the offloading is in progress, some other session in the network may broadcasts an R-REQUEST message (the SEEK operation). Now, the question is what will the relay node (that is currently offloading the sessions) do? We propose that in this scenario, the relay node will not construct a new OFFER for this session until it finishes its offloading process.

Multiple sessions become active. When multiple sessions become active at the same time, there will be multiple SEEK operations initiated in the network. Here, we will exploit the fact that the transmitted R-REQUEST messages will be transmitted sequentially and not simultaneously. All the relay nodes will construct the OFFERs sequentially. That is, initially the R-REPLY for the first session will be constructed. Only after the first session is finished (e.g., it has accepted an offer), the relay nodes will construct an OFFER in response to the second R-REQUEST message. Further, the R-REPLY message will contain the identity of the session for which this message is constructed.

F. Overhead Analysis

Since the D-GRS algorithm is activated by various events in the network, we will analyze the number of messages exchanged (i.e., the overhead) associated with each event in the network.

A session (s_i, d_i) **initiates.** The source node of the session broadcasts an R-REQUEST message, and can get at most N_r R-REPLY messages in reply. After receiving the R-REQUEST message, each relay node may further request CSI values from the source nodes it is currently supporting (see the information maintained at a relay node in Section III-A). As a session uses only one relay node, the total message exchanges in the network in this event cannot be more than $O(N_r + N_S)$.

A session (s_i, d_i) **terminates.** The source node of the departing session broadcasts a LEAVING message. This LEAVING message may result in at most N_s REMOVE-SRC messages (to offload the other sessions in the group). When a session is offloaded, it may want to seek other relay nodes. In this case, one R-REQUEST message from every offloaded source node will be transmitted. We know that a single R-REQUEST message can result in at most $O(N_r + N_s)$ messages. Thus, the total messages exchanged in the network due to a session termination cannot exceed $O(N_s \cdot (N_r + N_s))$.

A relay node becomes active. The relay node will broadcast a message requesting the number of active sessions in the network. This will result in a single reply from one of the existing relay nodes. Next, active relay nodes may transmit permission messages to their source nodes. This can result in at most N_s permission messages in the network. Every permission message will allow a source node to search for another relay node. The search for a relay node requires at most $O(N_r + N_s)$ messages as explained earlier. Thus, the total message exchanges in this case cannot exceed $O(N_s \cdot (N_r + N_s))$.

A relay node becomes inactive. The relay node broadcasts a single message indicating its deactivation. This will result in every source node using this relay node to seek other relay nodes. There are at most N_s source nodes in the network, and search for another relay node requires at most $O(N_r + N_s)$ messages as explained earlier. This can result in at most $O(N_s \cdot (N_r + N_s))$ messages in the network.

IV. SIMULATION RESULTS

In this section, we present simulation results to demonstrate the performance and complexity of the proposed D-GRS algorithm. As a benchmark, we also formulate the problem as an integer linear program [21], and use a centralized optimization solver CPLEX [6] to solve it. We will compare results from D-GRS with the optimal results from CPLEX. As expected, the running time of CPLEX is exponential due to NP-hardness and the integer linear nature of the formulated problem.

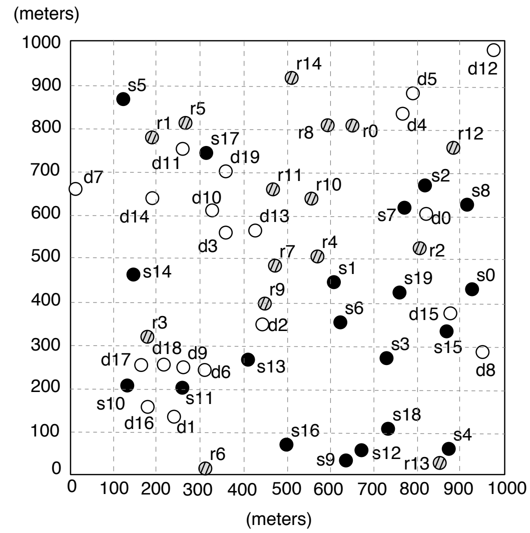
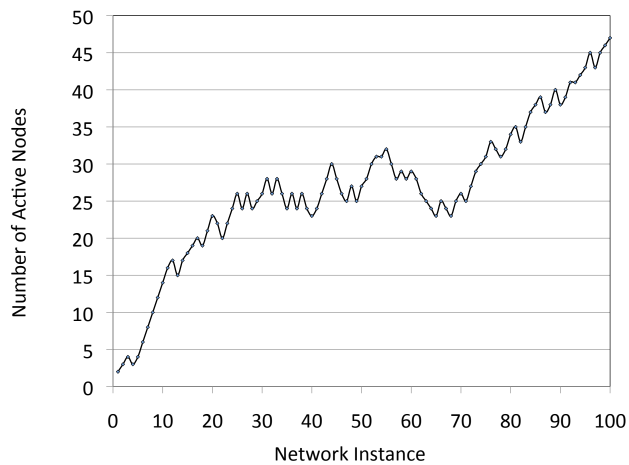
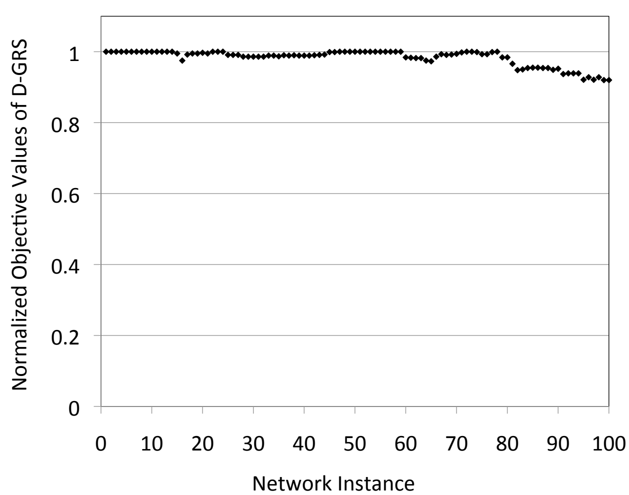


Fig. 7. A 55-node network.

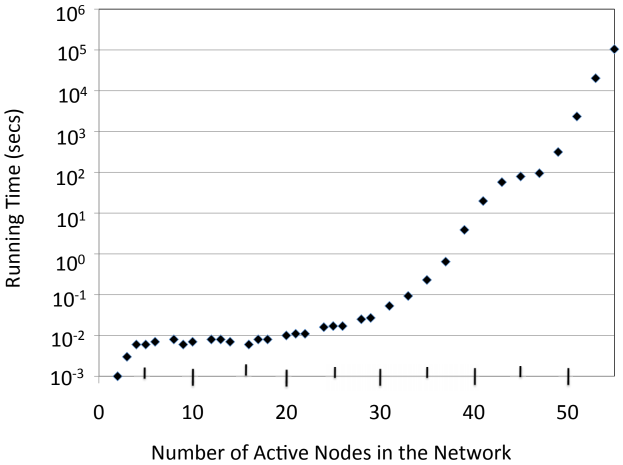


(a) Number of active nodes in the network.

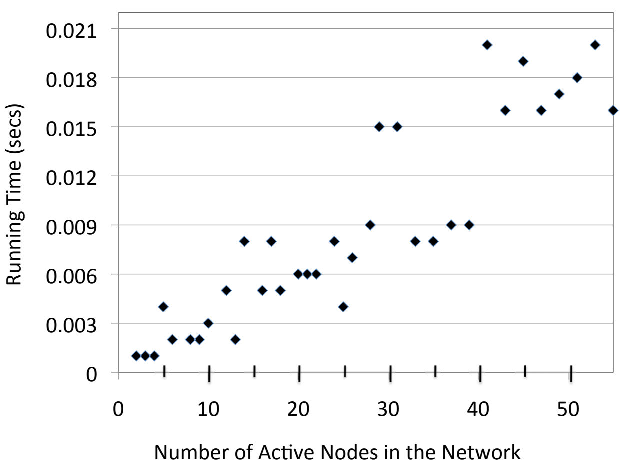


(b) Ratio of objective values from D-GRS and CPLEX.

Fig. 8. Results showing near-optimality of D-GRS under online dynamics.



(a) Solutions from CPLEX (exponential growth).



(b) Solutions from D-GRS.

Fig. 9. Time to obtain solutions.

A. Simulation Settings

For all network instances in the simulation, we assume the transmission power at each node is 1 W and the channel bandwidth is $W = 22$ MHz [9]. We assume the channel gain between two nodes s and d is $||s - d||^{-4}$, where $||s - d||$ is the distance (in meters) between s and d and 4 is the path loss index. We assume the white Gaussian noise at all the nodes has a variance of 10^{-10} W. The weight for each session is assumed to be 1.

B. Results for Online Dynamics

In this set of results, we consider a 55-node network consisting of 20 sessions and 15 relay nodes. The location of each node in the network is shown in Fig. 7. We show results from this 55-node network when nodes join and leave the network at random.

Initially, all sessions and relay nodes are assumed to be inactive. Then we allow new sessions to initiate and ongoing sessions to terminate, as well as activation/deactivation of relay nodes. The sequence of these online dynamics is chosen to be random. Figure 8(a) shows the number of active nodes (including both source/destination nodes and relay nodes) in the network for each of the 100 events.

The D-GRS runs continuously over the 100 events. Under each event, we compare the results from D-GRS and those from CPLEX. Figure 8(b) shows the normalized objective values of D-GRS (over those from CPLEX) under each event. We find that the performance of D-GRS is highly competitive (98.3% optimal on average).

We now compare the complexity (in terms of running time) of D-GRS and CPLEX. Figure 9(a) shows the time required to obtain the optimal solutions as the number of nodes in the network increases. Note that the y-axis in Fig. 9(a) is in log-scale, indicating the exponential running time of CPLEX. On the other hand, Fig. 9(b) shows that the running time of D-GRS algorithm is orders of magnitude smaller than that under CPLEX.

V. Related Work

We review related work on CC and NC separately, followed by related work on NC-CC.

(a) CC. The concept of CC can be traced back to the introduction of a three-terminal communication channel (or a relay channel) by van der Meulen [23]. Subsequently, Cover and El Gamal [5] developed a lower bound on the capacity of a general relay channel. Recent research on CC at the physical layer was motivated by these early results, and led to a number of CC protocols at the physical layer (e.g., [1], [13], [14]). These physical layer protocols proposed different ways in which distributed antennas could cooperate with each other, and aimed at improving the mutual information between transmitters and receivers. As the choice of a relay node in CC directly affects its performance, several researchers studied the problem of relay node assignment in single-hop networks (see e.g., [4], [22], and the references within). For multi-hop networks, the relay node problem was shown to be coupled tightly with flow-routing (see e.g., [10], [16], and the references within).

(b) NC. The concept of NC [8] was first introduced by Ahlswede *et al.* in [2], where they showed that NC can save bandwidth in a wired network with multicast flows. The core idea of NC is to reduce the number of time slots required to transmit packets by combining multiple packets at the physical layer. Due to this important property, NC has quickly found its applications in wireless networks, and can be categorized into two types: digital network coding (DNC) [11] and analog network coding (ANC) [12]. The reduction in the required time slots due to NC makes it an ideal candidate to improve the performance of CC.

(c) NC-CC. The benefits of employing NC in CC were recognized in [3], [19], [20], [24], [25], [26]. Due to the usage of NC with CC at physical layer, relay node selection is tightly coupled with session grouping. Most of the existing studies on NC-CC are information theoretic and limited in illustrating only the mechanism to combine NC with CC and the benefits of the combined approach, i.e., the time slot advantage. They do not address the issue of NC noise and how this tradeoff could be leveraged in a general network through appropriate session grouping and relay node selection.

In [3], Bao *et al*. showed how to use NC with CC to improve the outage probability in a network with multiple source nodes and single destination node. In [19], Peng *et al*. performed an analysis of outage probability in a network where NC is used by a single relay to enable CC for multiple sessions. In [24], Xiao *et al*. showed how NC could be used with CC to reduce the packet error rate in a simple two-source single-destination wireless network. In [20], Sharma *et al*. considered NC-CC with only one relay node. Their analysis showed that NC is not always good for CC, and introduced an important concept of NC noise. They showed that data rates of individual sessions in NC-CC are directly tied to the NC noise, which depends on individual sessions and the relay node. This motivated the study of joint grouping and relay node selection problem in this paper.

In [25], Xu and Li presented a CC framework for cellular networks that exploited NC opportunities but only worked in the presence of *bi-directional* traffic between two transmitters. It was not clear if/how their framework can be extended in a general network setting with unicast traffic and/or multiple destination nodes. In [26], the analysis of NC-CC was shown to improve throughput in a multi-hop wireless network. Again, the analysis in [26] was limited to bi-directional traffic, and the simple scenarios of two transmitters exploiting one relay node.

VI. Conclusion

NC-CC is a powerful paradigm that uses NC to improve the performance of CC in a multi-session network. However, the benefits of NC-CC can only be fully exploited by appropriate session grouping and relay node selection. In this paper, we studied this problem with the goal of maximizing the sum of weighted rates among all the sessions in the network. Due to NP-hardness of the problem, we developed a distributed and online algorithm that offers near-optimal solution to this problem. We showed that our distributed algorithm has polynomial complexity, and demonstrated that D-GRS adapts well to online network dynamics.

Acknowledgments

This work was supported in part by the National Science Foundation under Grant CCF 0946273 (Y.T. Hou, S. Sharma, and Y. Shi) and ECCS 0925719 (H.D. Sherali), and the Office of Naval Research (S. Kompella).

References

[1] B. Aazhang, R.S. Blum, J.N. Laneman, K.J.R. Liu, W. Su, and A. Wittneben (Editors), *IEEE Journal on Selected Areas in Communications – Special Issue on Cooperative Communications and Networking,* vol. 25, no. 2, Feb. 2007.

[2] R. Ahlswede, N. Cai, S.R. Li, and R.W. Yeung, "Network information flow," *IEEE Trans. on Information Theory,* vol. 46, issue 4, pp. 1204–1216, July 2000.

[3] X. Bao and J. Li, "Adaptive network coded cooperation (ANCC) for wireless relay networks: Matching code-on-graph with network-on-graph," *IEEE Trans. on Wireless Communications,* vol. 7, no. 2, pp. 574–583, February 2008.

[4] A. Bletsas, A. Khisti, D. Reed, and A. Lippman, "A simple cooperative diversity method based on network path selection," *IEEE Journal on Selected Areas in Communications,* vol. 24, no. 3, pp. 659–672, March 2006.

[5] T.M. Cover and A. EL Gamal, "Capacity theorems for the relay channel," *IEEE Trans. on Information Theory,* vol. 25, issue 5, pp. 572–584, Sept. 1979.

[6] IBM ILOG CPLEX: http://www.ibm.com/software/integration/optimization/cplex/

[7] M.R. Garey and D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness,* W.H. Freeman & Co., 1990.

[8] The network coding home page: http://www.ifp.uiuc.edu/~koetter/NWC/index.html

[9] IEEE 802.11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications: http://standards.ieee.org/getieee802/download/802.11-2007.pdf

[10] G. Jakllari, S.V. Krishnamurthy, M. Faloutsos, P.V. Krishnamurthy, and O. Ercetin, "A cross-layer framework for exploiting virtual MISO links in mobile ad hoc networks," *IEEE Trans. on Mobile Computing,* vol. 6, no. 5, pp. 579–594, June 2007.

[11] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, "XORs in the air: Practical wireless network coding," in *Proc. of ACM SIGCOMM,* pp. 497–510, Pisa, Italy, September 11–15, 2006.

[12] S. Katti, S. Gollakotta, and D. Katabi, "Embracing wireless interference: Analog network coding," in *Proc. of ACM SIGCOMM,* pp. 397–408, Kyoto, Japan, August 27–31, 2007.

[13] G. Kramer, I. Maric, and R.D. Yates, *Foundations and Trends in Networking: Cooperative Communications, NOW publishers,* vol. 1, nos. 3–4, 2006.

[14] G. Kramer, R. Berry, A. El Gamal, H. El Gamal, M. Franceschetti, M. Gastpar, and J.N. Laneman (Editors), *IEEE Trans. on Information Theory – Special Issue on Models, Theory, and Codes for Relaying and Cooperation in Communication Networks,* vol. 53, no. 10, Oct. 2007.

[15] M. Kurth, A. Zubow, and J-P. Redlich, "Cooperative opportunistic routing using transmit diversity in wireless mesh networks," in *Proc. of IEEE INFOCOM,* pp. 1984–1992, Phoemix, AZ, April 15–17, 2008.

[16] S. Lakshmanan and R. Sivakumar, "Diversity routing for multi-hop wireless networks with cooperative transmissions," in *Proc. of IEEE SECON,* Rome, Italy, June 22–26, 2009.

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

[18] E.L. Lawler, *Combinatorial Optimization: Networks and Matroids,* Holt, Rinehart and Winston, New York, 1976.

[19] C. Peng, Q. Zhang, M. Zhao, and Y. Yao, "On the performance analysis of network-coded cooperation in wireless networks," in *Proc. of IEEE INFOCOM,* pp. 1460–1468, Anchorage, Alaska, May 6–12, 2007.

[20] S. Sharma, Y. Shi, J. Liu, and Y.T. Hou, "Is network coding always good for cooperative communications," in *Proc. IEEE INFOCOM,* San Diego, CA, March 15–19, 2010.

[21] S. Sharma, Y. Shi, Y.T. Hou, H.D. Sherali, and S. Kompella "Optimizing network-coded cooperative communications via joint session grouping and relay node selection," *Technical Report,* Dept. of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA. Jan. 2011, http://people.cs.vt.edu/~sushant/papers/TR-DGRS.pdf.

[22] Y. Shi, S. Sharma, Y.T. Hou, and S. Kompella, "Optimal relay assignment for cooperative communications," in *Proc. of ACM MobiHoc,* pp. 3–12, Hong Kong, China, May 27–30, 2008.

[23] E.C. van der Meulen, "Three terminal communication channels," *Advances in Applied Probability,* vol. 3, pp. 120–154, 1971.

[24] L. Xiao, T.E. Fuja, J. Kliewer, and D.J. Costello, "A network coding approach to cooperative diversity," *IEEE Trans. on Information Theory,* vol. 53, no. 10, pp. 3714–3722, Oct. 2007.

[25] H. Xu and B. Li, "XOR-assisted cooperative diversity in OFDMA wireless networks: Optimization framework and approximation algorithms," in *Proc. of IEEE INFOCOM,* pp. 2141–2149, Rio de Janeiro, Brazil, April 19–25, 2009.

[26] J. Zhang and Q. Zhang, "Cooperative network coding-aware routing for multi-rate wireless networks," in *Proc. of IEEE INFOCOM,* pp. 181–189, Rio de Janeiro, April 19–25, 2009.