This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

Cooperative Communications in Multi-hop Wireless Networks: Joint Flow Routing and Relay Node Assignment

Sushant Sharma* Yi Shi† Y. Thomas Hou† Hanif D. Sherali‡ Sastry Kompella§
*Department of Computer Science, Virginia Tech, Blacksburg, VA, USA
†Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA, USA
‡Grado Department of Industrial and Systems Engineering, Virginia Tech, Blacksburg, VA, USA
§Information Technology Division, U.S. Naval Research Laboratory, Washington, DC, USA

***Abstract*—It has been shown that cooperative communications (CC) have the potential to significantly increase the capacity of wireless networks. However, most of the existing results are limited to single-hop wireless networks. To illustrate the benefits of CC in multi-hop wireless networks, we solve a joint optimization problem of relay node assignment and flow routing for concurrent sessions. We study this problem via mathematical modeling and solve it using a solution procedure based on the branch-and-cut framework. We design several novel components to speed-up the computation time of branch-and-cut. Via numerical results, we show the significant rate gains that can be achieved by incorporating CC in multi-hop networks.**

I. Introduction

The potential of spatial diversity, in the form of employing multiple antennas, to cope fading in wireless channels has been well recognized. Multiple antennas can be equipped on each node in the network (i.e., MIMO) or distributed at different nodes in the network. For MIMO, there is a physical limitation on the number of antennas that can be packed on the same node, as the separation between the antennas is dictated by the operating radio wavelength. On the other hand, distributed antenna systems employing *cooperative communications* (CC) do not have this constraint and thus have received much attention in recent years [5], [6], [7], [11], [15], [17], [18]. Under CC, each node is equipped with only a *single* antenna and spatial diversity is achieved by exploiting the antennas on *other* nodes in the network.

Although there has been active research on CC at the physical layer or for single-hop communications, results on CC in *multi-hop* wireless networks remain very limited. In this paper, we illustrate the benefits of using CC in multi-hop wireless networks by investigating a joint problem of relay node assignment and multi-hop flow routing. The objective of this problem is to maximize the minimum rate among a set of concurrent communication sessions. For each session, the data from the source node may need to traverse multiple hops before reaching its destination node. Further, CC can be exploited along any link of the path to increase a session's rate. The main difficulties here are (1) the assignment of relay nodes (either for CC or as a multi-hop relay) to each user session, and (2) the coupling problem of multi-hop flow routing and relay node assignment.

To solve the problem, we first develop a mathematical characterization for cooperative relay node assignment and multi-hop flow routing for a set of concurrent user communication sessions. The formulated problem initially has non-linear constraints, which we show can be converted into linear constraints using problem specific properties. As a result, the final problem formulation is a mixed integer linear programming (MILP) problem. We then develop a solution procedure based on *branch-and-cut* framework. We propose three novel components that makes the solution procedure highly efficient. First, we develop an efficient polynomial time local search algorithm to generate feasible flow-routes that exploit CC along individual hops. Second, based on our problem structure, we establish a clever strategy to generate cutting planes that significantly decreases the number of branches in our branch-and-cut tree. Third, we present an innovative approach to perform branching operations that exploits problem specific properties to choose superior branches and reduce overall computation time. Our solution procedure provides $(1-\epsilon)$-optimal solutions, with ϵ being the desired approximation error bound.

The remainder of this paper is organized as follows. Section II presents related work. Section III describes how CC work in a three-node model, which will be the basic building block for multi-hop study in this paper. In Section IV, we give a mathematical characterization for joint cooperative relay node assignment and multi-hop routing for a set of concurrent user communication sessions. We also present a problem formulation, with the goal of maximizing the minimum rate among user sessions. In Section V, we devise a solution based on branch-and-cut framework to solve the optimization problem. Section VI presents numerical results to demonstrate the rate gains that can be achieved by incorporating CC in multi-hop wireless networks. Section VII concludes this paper.

II. Related Work

The concept of CC can be traced back to the pioneering work done by Van Der Meulen [21] and Cover and El Gamal [3]. In [21], Van Der Meulen first introduced the three-terminal communication channel (or a relay channel) and gave capacity bounds for various ways of sending information on

978-1-4244-5837-0/10/$26.00 ©2010 IEEE

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

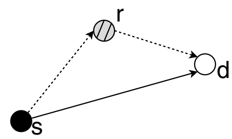


Fig. 1. A three-node schematic for CC.

this channel. Cover and El Gamal [3] studied general relay channel and established an achievable lower bound. These early works on relay channels laid the foundation for CC.

In recent years, there is growing interest on exploiting distributed antennas for wireless networks, which leads to research on CC protocols at physical layer (see e.g., [5], [6], [7], [15], [17], [18]). These physical layer CC protocols have recently found their application in ad hoc networks, either single-hop networks [2], [19], [23], [25] or multi-hop networks [8], [9], [14], [16], [24]. In single-hop networks, the focus has mostly been on relay node selection (assignment) between each source and destination pair in the network.

For multi-hop networks, Khandani et al. [9] studied minimum energy routing problem (for a single message) by exploiting both wireless broadcast advantage and CC (called wireless cooperative advantage in the paper). They developed a dynamic programming based solution and two heuristic algorithms to find the minimum energy route for a single message. However, their approach is limited to individual messages as opposed to flows that we have considered in this paper. In [24], Yeh and Berry aimed to generalize the well known maximum differential backlog policy [20] in the context of CC. They formulated a challenging nonlinear program that characterizes the network stability region, but only provided solutions for a few simple cases. In [16], Scaglione et al. proposed two architectures for multi-hop cooperative wireless networks. Under these architectures, nodes in the network can form multiple cooperative clusters. They showed that the network connectivity can be improved by using such cooperative clusters. However, problems related with flow routing and relay node assignment are not the focus for their work. Additionally, prior works [8], [14] have proposed heuristics that separately develop routing solutions before addressing relay node assignment for CC. In contrast, this paper considers joint flow routing and relay node assignment for concurrent sessions, and presents a solution with performance guarantee.

III. Reference Three-Node CC Models

The essence of CC is to exploit (1) the wireless broadcast advantage and (2) the relaying capability of other nodes so as to achieve higher throughput, lower transmission error, or other objectives in transmission [9], [12]. Figure 1 shows the well known three-node model for CC, where node s is the source node, node d is the destination node, and node r is a relay node.

Although the three-node model is simple, the specific mechanism to accomplish CC is not unique. In [11], the authors use a frame/time slot based model for transmission. In particular, they model transmission from s on a frame-by-frame basis. Within a frame, there are two time slots. In the first time slot, source s makes a transmission to d, which is also overheard by node r. In the second time slot, relay node r forwards the data received in the first time slot to destination node d [11]. Node d can now apply any diversity combining technique on two copies of the data from two different paths, thereby achieving higher capacity gains.

In this paper, instead of limiting ourselves to the time-slot model, we employ an orthogonal channel model for CC in multi-hop wireless network. The orthogonal channel model is more analytically tractable than the time-slot model in a multi-hop environment and can be transformed into a model that uses frequency division, time division, or code division multiplexing. Usage of orthogonal channels has been widely accepted for CC (see e.g. [2], [6], [7], [15], [16]). In our setting, each node uses separate channels for transmission and reception and thus can transmit and receive data on different channels at the same time without self-interfering.

In the following, we present the achievable rate between s and d under CC. We consider both the amplify-and-forward (AF) and decode-and-forward (DF) modes [11], as well as direct transmission.

CC with Amplify-and-Forward (AF) Under this mode, relay node r receives, amplifies, and forwards the signal from source node s (all in analog form) to destination node d [11]. Let h_{sd}, h_{sr}, h_{rd} capture the effects of path-loss, shadowing, and fading within its respective channel between nodes s and d, s and r, and r and d, respectively. Also denote z_d and z_r the zero-mean background noise at nodes d and r, with variance σ_d^2 and σ_r^2, respectively. For simplicity, we assume the background noise at a node has the same stochastic property on different channels. Denote P_s and P_r the transmission powers at nodes s and r, respectively.

Following the same token for deriving the rate under AF mode in [11], it can be shown that the achievable rate between s and d with r as a relay is

$$C_{\mathrm{AF}}(s,r,d) = W \cdot I_{\mathrm{AF}}(s,r,d) \ ,$$

where

$$I_{\mathrm{AF}}(s,r,d) = \log_2\left(1 + \mathrm{SNR}_{sd} + \frac{\mathrm{SNR}_{sr}\cdot\mathrm{SNR}_{rd}}{\mathrm{SNR}_{sr}+\mathrm{SNR}_{rd}+1}\right),$$

$\mathrm{SNR}_{sd} = \frac{P_s}{\sigma_d^2}|h_{sd}|^2$, $\mathrm{SNR}_{sr} = \frac{P_s}{\sigma_r^2}|h_{sr}|^2$, $\mathrm{SNR}_{rd} = \frac{P_r}{\sigma_d^2}|h_{rd}|^2$, and W is the available bandwidth of channels at nodes s and r.

CC with Decode-and-Forward (DF) Under this mode, relay node r first decodes and estimates the received signal from source node s, and then transmits the estimated data to destination node d [11]. The achievable rate under DF mode can be developed by following the same token in [11], which is

$$C_{\mathrm{DF}}(s,r,d) = W \cdot I_{\mathrm{DF}}(s,r,d) \ ,$$

where

$$I_{\mathrm{DF}}(s,r,d) = \min\{\log_2(1+\mathrm{SNR}_{sr}), \\ \log_2(1+\mathrm{SNR}_{sd}+\mathrm{SNR}_{rd})\}.$$

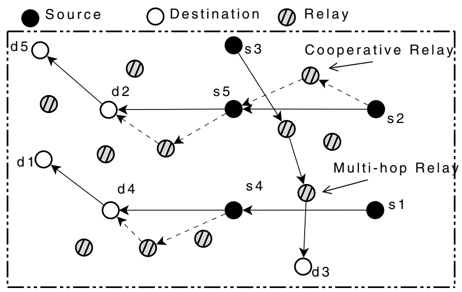


Fig. 2. A multi-hop ad hoc network consisting of source nodes, destination nodes and relay nodes.

Direct Transmission (without CC) When CC is not used, the achievable rate from source node s to destination node d is simply

$$C_{\text{D}}(s,d) = W \log_2(1 + \text{SNR}_{sd}) \ .$$

A number of comments follow from the above discussion. First, note that $I_{\text{AF}}(\cdot)$ and $I_{\text{DF}}(\cdot)$ are increasing functions of P_s and P_r, respectively. This suggests that, in order to achieve the maximum rate under either AF or DF, both source node and relay node should transmit at the maximum power P. Thus, we set $P_s = P_r = P$. Due to the use of orthogonal channels, this will not create interference for other concurrent transmissions on different channels.

Second, based on the rate expressions, one can see that although AF and DF are different physical layer mechanisms, the achievable rates for both of them have the same mathematical form, i.e., a function of SNR_{sd}, SNR_{sr}, and SNR_{rd}. Therefore, any solution procedure designed for AF can be readily extended for DF. As a result, it is sufficient to focus on the development of solution for one of them, for which we choose AF in this paper.

IV. CC in Multi-hop Wireless Networks

A. Network Setting

We consider a multi-hop wireless network where there are multiple concurrent unicast sessions. The data for each communication session may traverse multiple hops in the network. We assume the presence of orthogonal channels in the network, and consider that the transmitters are able to transmit concurrently without any interference.

We distinguish two types of relay nodes in the network based on their functions. We call a relay node used for CC purpose (i.e., node r in Fig. 1) as *Cooperative Relay* (CR) and a relay node used for multi-hop relaying in the traditional sense as *Multi-hop Relay* (MR). Note that a CR operates at the physical layer while MR operates at the network layer.

Physical limitations of many transceivers prohibit them from transmitting data on multiple channels at the same time. As a result, a relay node can serve as either CR or MR, but not both at the same time. For the same reason, a source node (or destination node) cannot serve as a CR, and a single CR can not serve more than one transmitter and receiver pair.

In [25], Zhao et al. have shown that for a single hop, the diversity gain obtained by exploiting multiple relay nodes is marginally higher than the diversity gain that can be obtained by selecting the *best* relay. As a result, we only consider at most one relay node for CC between each sender and receiver in this paper.

Figure 2 shows an example ad hoc network where there are five communication sessions (s_i, d_i), $i = 1, 2, \cdots, 5$. In this example, dashed line involving a relay node represents CC (and thus the relay node is serving the role of CR) while solid line involving a relay node represents traditional multi-hop relay (and thus the relay node is serving as a MR). Note that session 1 (from node s_1 to node d_1) traverses nodes s_4 and d_4 as its intermediate hops. Between nodes s_4 and d_4, a CR is used. Here, source node s_4 and destination node d_4 are being used as MRs for session 1. As a result, the hop from s_4 to d_4 is carrying traffic for both session 1 and session 4. As another example, session 2 (from node s_2 to node d_2) traverses node s_5. Node s_5 is the source node of session 5 and is also being used as a MR for session 2. As a result, the hop from s_5 to d_2 is carrying traffic for both session 2 and session 5.

B. Mathematical Modeling

In this section, we present a mathematical model for our joint flow routing and relay node assignment problem. Denote $\mathcal{N}$ the set of nodes in the network, with $|\mathcal{N}| = N$. In set $\mathcal{N}$, there are three subsets of nodes, namely, (i) the set of source nodes, $\mathcal{N}_s = \{s_1, s_2, \cdots, s_{N_s}\}$, with $|\mathcal{N}_s| = N_s$, (ii) the set of destination nodes, $\mathcal{N}_d = \{d_1, d_2, \cdots, d_{N_d}\}$, with $|\mathcal{N}_d| = N_d = N_s$, and (iii) the set of remaining nodes that are available for serving as CR or MR nodes, $\mathcal{N}_r = \{r_1, r_2, \cdots, r_{N_r}\}$, with $|\mathcal{N}_r| = N_r$. For clarity, we assume all source and destination nodes are distinct. In this context, for unicast communication sessions, we have $N = N_s + N_d + N_r = 2N_s + N_r$.

Role of Relay Node. Due to the existence of CRs, it is necessary to introduce integer variables to characterize whether or not an available relay node will be used as CR. A binary variable A^w_{uv} is defined for this purpose. Specifically,

$$A^w_{uv} = \begin{cases} 1 & \text{if node } w \text{ is used as a CR on hop } (u,v), \\ 0 & \text{otherwise.} \end{cases}$$

We also introduce another binary variable B_{uv} to specify whether or not the link from u to v is active in the routing solution. That is,

$$B_{uv} = \begin{cases} 1 & \text{if } v \text{ is the next hop node of node } u, \\ 0 & \text{otherwise.} \end{cases}$$

For each relay node $w \in \mathcal{N}_r$, since we assume that it may be used as either a CR or MR at most once, we can characterize this with the following two constraints.

$$\textstyle\sum_{u\in\mathcal{N}}^{u\neq w, u\neq v} \sum_{v\in\mathcal{N}}^{v\neq w} A^w_{uv} + \sum_{t\in\mathcal{N}}^{t\neq w} B_{tw} \leq 1 \quad (w \in \mathcal{N}_r) \ ,(1)$$
$$\textstyle\sum_{u\in\mathcal{N}}^{u\neq w, u\neq v} \sum_{v\in\mathcal{N}}^{v\neq w} A^w_{uv} + \sum_{t\in\mathcal{N}}^{t\neq w} B_{wt} \leq 1 \quad (w \in \mathcal{N}_r) \ .(2)$$

In both (1) and (2), if the first term is 1 (i.e., node w is used as a CR), then the second term must be 0 (i.e., w cannot be used as a MR). Similarly, if the second term is 1 (i.e., node w is used as a MR), then the first term must be 0 (i.e., w cannot be used as a CR).

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

For a relay node $w \in \mathcal{N}_r$ that is being used as a MR, since w is not the destination node of any communication session, the traffic entering node w must also exit. This can be written as follows.

$$\sum_{u \in \mathcal{N}}^{u \neq w} B_{uw} = \sum_{v \in \mathcal{N}}^{v \neq w} B_{wv} \quad (w \in \mathcal{N}_r) \; . \tag{3}$$

Note that (3) also holds when w is not a MR. In this case, all B variables in (3) are 0.

It can be shown that it is sufficient to include either (1) or (2) once we have (3). As a result, we only include (1) in the problem formulation.

Further, we may assign a relay node as a CR to hop (u, v) only if it is active (i.e., if $B_{uv} = 1$). Otherwise, no relay node should be assigned as CR to hop (u, v). This constraint can be characterized as follows.

$$B_{uv} - \sum_{w \in \mathcal{N}_r}^{w \neq u, w \neq v} A_{uv}^w \geq 0 \quad (u \in \mathcal{N}, v \in \mathcal{N}, v \neq u) \; . \tag{4}$$

Flow Routing. As explained earlier, due to transceiver limitations, a node can only transmit on one channel at any given time. As a result, we limit the transmission and reception of data at the network layer to only one transmitter and one receiver. This can be mathematically characterized by the following constraints.

$$\sum_{v \in \mathcal{N}}^{v \neq s_i} B_{s_i v} = 1 \quad (s_i \in \mathcal{N}_s) \; , \tag{5}$$

$$\sum_{v \in \mathcal{N}}^{v \neq u} B_{uv} \leq 1 \quad (u \notin \mathcal{N}_s) \; , \tag{6}$$

$$\sum_{u \in \mathcal{N}}^{u \neq v} B_{uv} \leq 1 \quad (v \notin \mathcal{N}_d) \; , \tag{7}$$

$$\sum_{v \in \mathcal{N}}^{v \neq d_i} B_{v d_i} = 1 \quad (d_i \in \mathcal{N}_d) \; , \tag{8}$$

where (5) says that a source node must transmit data to some other node and (8) says that a destination node must receive data from some node.

We note that there are some redundant constraints in (6) and (7). Constraint (6) can be divided into the following two sets of constraints.

$$\sum_{v \in \mathcal{N}}^{v \neq w} B_{wv} \leq 1 \quad (w \in \mathcal{N}_r) \; , \tag{9}$$

$$\sum_{v \in \mathcal{N}}^{v \neq d_i, s_i} B_{d_i v} \leq 1 \quad (d_i \in \mathcal{N}_d) \; . \tag{10}$$

Similarly, constraint (7) can also be divided into the following two sets of constraints

$$\sum_{u \in \mathcal{N}}^{u \neq w} B_{uw} \leq 1 \quad (w \in \mathcal{N}_r), \tag{11}$$

$$\sum_{u \in \mathcal{N}}^{u \neq s_i, d_i} B_{u s_i} \leq 1 \quad (s_i \in \mathcal{N}_s) \; . \tag{12}$$

Note that due to (3), (9) is equivalent to (11). Thus, instead of using (6), we will use (10) in the final problem formulation.

Denote $f_{uv}(s_i)$ the flow rate on link (u, v) that is attributed to source-destination pair (i.e., communication session) (s_i, d_i). The flow balance (i.e., the incoming flow rate must be equal to the outgoing flow rate at an intermediate node w along the path between s_i and d_i) is formulated as follows.

$$\sum_{u \in \mathcal{N}}^{u \neq w, u \neq d_i} f_{uw}(s_i) = \sum_{v \in \mathcal{N}}^{v \neq w, v \neq s_i} f_{wv}(s_i)$$
$$(s_i \in \mathcal{N}_s, w \in \mathcal{N}, w \neq d_i, w \neq s_i) \; . \tag{13}$$

Using (13), it is easy to show that $\sum_{w \in \mathcal{N}}^{w \neq s_i} f_{s_i w}(s_i) = \sum_{w \in \mathcal{N}}^{w \neq d_i} f_{w d_i}(s_i)$, which states that all the data generated by a source node s_i will reach its destination node d_i.

Capacity. To ensure the feasibility of routing solution, we must consider the capacity constraint on each hop in the network. That is, the aggregate flow rates traversing link (u, v) must not exceed the capacity on this link, i.e.,

$$\begin{aligned} \sum_{s_i \in \mathcal{N}_s}^{s_i \neq v} f_{uv}(s_i) \;\; \leq \;\; & \left(1 - \sum_{w \in \mathcal{N}_r}^{w \neq u, w \neq v} A_{uv}^w \right) C_{\text{D}}(u, v) B_{uv} \\ & + \sum_{w \in \mathcal{N}_r}^{w \neq u, w \neq v} A_{uv}^w C_{\text{AF}}(u, w, v) B_{uv} \\ & (u \in \mathcal{N}, v \in \mathcal{N}, v \neq u) \; . \end{aligned} \tag{14}$$

Note that on the right-hand-side (RHS) of (14), there can be at most one non-zero term, depending on whether direct transmission or CC is employed. If direct transmission is employed, then the first term on the RHS of (14) is non-zero and the second term is 0; the converse is true when CC is employed.

C. Problem Formulation

We consider a set of N_s communication sessions in the network, denoted by $\mathcal{N}_s$. The goal is to maximize the minimum flow rate among all active sessions via optimal multi-hop flow routing and cooperative relay assignment. More formally, for a given communication session (s_i, d_i), denote the end-to-end flow rate (or throughput) as R_{s_i}, where $R_{s_i} = \sum_{v \in \mathcal{N}}^{v \neq s_i} f_{s_i v}(s_i)$. Denote R_{min} the minimum flow rate among all sessions, i.e.,

$$R_{\text{min}} \leq \sum_{v \in \mathcal{N}}^{v \neq s_i} f_{s_i v}(s_i) \quad (s_i \in \mathcal{N}_s) \; . \tag{15}$$

Then our objective is to maximize R_{min}.

Before we present the final formulation, we would like to convert the nonlinear constraint (14) into a linear constraint. The constraint in (14) contains the product of two variables A_{uv}^w and B_{uv} and is thus in non-linear form. We can reformulate it into a linear constraint by exploiting the following property for A_{uv}^w and B_{uv}.

Property 1: $B_{uv} \cdot A_{uv}^w = A_{uv}^w$ for any $u \in \mathcal{N}, v \in \mathcal{N}, v \neq u, w \in \mathcal{N}_r, w \neq u, w \neq v$.

Proof: This property is proved by considering both cases of B_{uv}.

(i) When $B_{uv} = 1$, the equality holds trivially.

(ii) When $B_{uv} = 0$, it means the link (u, v) is not active. As a result, no CR should be assigned to (u, v), i.e., $A_{uv}^w = 0$ for $w \in \mathcal{N}_r, w \neq u, w \neq v$ by (4). The equality still holds. ■

By using Property 1, we can rewrite the constraint in (14) as follows.

$$\begin{aligned} \sum_{s_i \in \mathcal{N}_s}^{s_i \neq v} f_{uv}(s_i) \;\; \leq \;\; & \left(B_{uv} - \sum_{w \in \mathcal{N}}^{w \neq u, w \neq v} A_{uv}^w \right) C_{\text{D}}(u, v) \\ & + \sum_{w \in \mathcal{N}}^{w \neq u, w \neq v} A_{uv}^w C_{\text{AF}}(u, w, v) \\ & (u \in \mathcal{N}, v \in \mathcal{N}, v \neq u) \; , \end{aligned} \tag{16}$$

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

which is now a linear constraint.

All of our constraints are now linear. Using the above definition of objective function, we are now ready to present the problem formulation in its entirety, with the necessary constraints described in the earlier part of this section.

$$\begin{array}{ll} \text{Max} & R_{\min} \\ \text{s.t.} & (1), (3), (4), (5), (7), (8), (10), (13), (15), (16) \end{array}$$

$$R_{\min}, f_{uv}(s_i) \geq 0 \quad (s_i \in \mathcal{N}_s, \quad u \in \mathcal{N},\ v \in \mathcal{N}, \quad u \neq v, d_i,\ v \neq s_i)$$

$$A^w_{uv}, B_{uv} \in \{0, 1\} \quad (w \in \mathcal{N}_r, u \in \mathcal{N}, v \in \mathcal{N}, u \neq v \neq w)$$

where $R_{\min}$, $f_{uv}(s_i)$, A^w_{uv} and B_{uv} are optimization variables.

It is not hard to see that this optimization formulation is in the form of *mixed integer linear program* (MILP), which is NP-hard in general [4], [22].

V. A Proposed Solution

For the MILP problem formulation, we devise a solution procedure based on the so-called *branch-and-cut* framework [13]. Branch-and-cut is an enhancement of *branch-and-bound* with the so called *cutting plane* method to deal with integer variables [1], [13]. We further enhance the branch-and-cut framework with several novel problem-specific components.

In Section V-A, we provide a brief overview of the branch-and-cut framework. For a comprehensive understanding of branch-and-cut procedure, readers are referred to [13]. Then in Section V-B to V-D, we give details on our proposed components to the solution procedure.

A. Algorithm Overview

The branch-and-cut solution procedure consists of a set of iterative steps. During the first iterative step, an upper bound on the objective value is obtained by solving a relaxed version of the MILP problem. Due to relaxation, the values of A^w_{uv} and B_{uv} in the solution may become fractional. Therefore, a local search algorithm, called *Feasible Solution Construction* (FSC), is proposed to obtain a feasible solution from the relaxed solution. The feasible solution obtained from FSC provides a lower bound on the objective value. If the gap between the lower bound and the upper bound is greater than the desired gap (depends upon the value of ϵ), cutting planes are added to the problem. A cutting plane is a linear constraint that reduces the feasible region of the relaxed problem (but not the original MILP), thereby improving the values of upper and lower bound. The cutting planes are added to the problem as long as they are improving the upper and lower bounds.

After cutting planes can no longer improve the bounds, the problem is branched into two subproblems. The relaxed version of these two subproblems is then solved and FSC is used to obtain the upper and lower bounds. This step finishes the iteration.

After an iteration, if the gap between the largest upper bound (among all the subproblems), and the largest lower bound (among all the subproblems) is more than ϵ, another iteration step (similar to the first step) is performed on the subproblem with largest upper bound. Note that after every iteration, the chosen subproblem is branched into two subproblems, increasing the total number of subproblems in the system.

The iterations of branch-and-cut continues until the largest upper bound (among all the current subproblems) and largest lower bound among all the subproblems (i.e., best feasible solution) are within ϵ of each other. At this point, the best feasible solution is $(1 - \epsilon)$-optimal. As one can see, the key challenge in implementing a branch-and-cut framework is to develop several problem-specific components. For our problem, we propose the following components.

1) An efficient polynomial time local search algorithm, which we call *Feasible Solution Construction* (FSC) algorithm. FSC algorithm generates feasible flow-routes that exploit CC along individual hops.
2) Based on our problem structure, we establish a clever strategy to generate cutting planes that significantly decreases the number of branches in our branch-and-cut tree.
3) A helpful approach to perform branching operations that exploits problem specific properties to choose superior branches and reduce overall computation time.

Although the worst case complexity of our solution remains exponential (due to MILP), the actual run time is in fact reasonable. This reasonable running time can be attributed to our proposed new components in the branch-and-cut framework. In the rest of this section, we offer the details for these components.

B. FSC Algorithm

After solving the relaxed MILP, the solution may have fractional values for some of the A^w_{uv}'s or B_{uv}'s, which is clearly infeasible. The proposed FSC is a *local search* algorithm that constructs a feasible solution based on a given infeasible solution. The algorithm will determine feasible routing, CR assignment, and flow rates for all sessions in the network.

Our proposed FSC algorithm is an efficient polynomial time algorithm, and addresses the solution construction process in three phases, namely, *Path Determination, CR Assignment,* and *Flow Re-calculation.* There are number of subtle technical details that need to be addressed in these phases. In the following, we give details on these three phases. We omit the detailed pseudo-code for FSC Algorithm due to paper length limitation.

1) Phase 1: Path Determination. The goal of this first phase is to find a feasible and potentially high capacity paths for each session in the network. In this phase, our algorithm starts by assuming no prior paths exist for any session in the network. Among the sessions whose paths are yet to be determined, the algorithm performs path determination for a session (chosen at random) iteratively.

When determining the next-hop node, we take the following approach. Suppose we are searching the next hop node for a

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

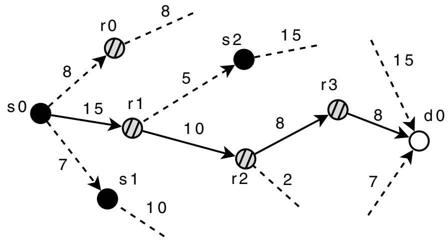


Fig. 3. A simple path between source and destination nodes where intermediate nodes are all in $\mathcal{N}_r$.

node r_i. In the relaxed solution, it is possible that r_i may have multiple next-hop nodes. Here, among these candidate next-hop nodes, we select the node r_j to which r_i is transmitting the largest amount of data (in the relaxed solution) for source node s_i. This "widest pipe" approach, although heuristic, has the potential of finding a high capacity path. Please note that once a node is included in a path, it will not be considered for inclusion in the other paths during subsequent iterations.

Case 1 – Simple Path We consider the simple case first, where after the widest-pipe approach, the intermediate nodes between source and destination nodes of a session turn out to be all from the set $\mathcal{N}_r$. An example is shown in Fig. 3, where the final path between s_0 and d_0 will go through r_1, r_2 and r_3, with r_1, r_2 and r_3 all in set $\mathcal{N}_r$. In this case, phase 1 for the selected session is considered complete, and the algorithm will move on to Phase 2 for the selected session.

Case 2 – Overlapping Path In this case, based on the widest-pipe approach, we have encountered an intermediate node from the set of $\mathcal{N}_s$ or $\mathcal{N}_d$, i.e., a source or destination node. So the path under consideration may overlap with the path of another session. This is the main complicating situation that we need to deal with in the path determination phase. Depending on the type of this intermediate node (source or destination), different mechanisms need to be devised.

Sub-case 2.1: The encountered intermediate node is the source node of another session. In this case, this encountered intermediate node (say s_j) is included in the path as the next-hop node. At the same time, s_j is recorded in a special list (denoted as $\mathcal{L}$) to keep track of such source nodes of other sessions that have not yet found their own paths to their corresponding destination nodes but are included in the path during the current iteration. Note that the source node s_i for the path under construction is not listed in $\mathcal{L}$.

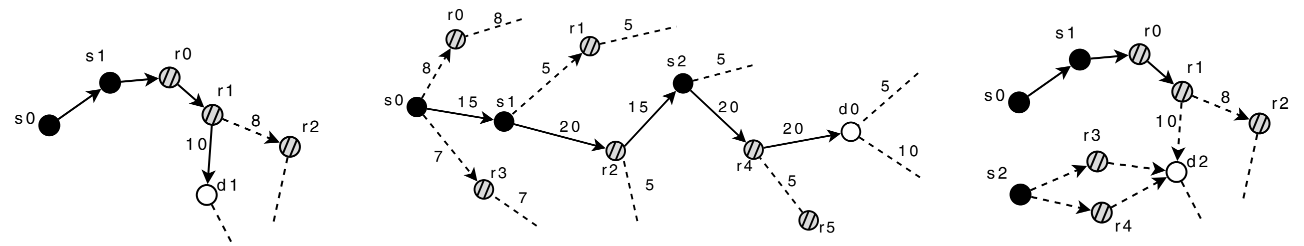
Sub-case 2.2: The encountered intermediate node is a destination node. This is the most complex case and a number of scenarios need to be considered.

A. **This node is the destination node of a source node in $\mathcal{L}$:** In this case, this node will be included in the current path under construction. Its corresponding source node will be removed from $\mathcal{L}$, since the path for that source node is complete. An an example, in Fig. 4(a), the path under construction is for source node s_0. Currently, node d_1 is considered as the next hop node along the path and the corresponding source node s_1 is in list $\mathcal{L}$. In this case, d_1 is added to the path and s_1 is removed from list $\mathcal{L}$.

B. **This node is the destination node of the current path under construction.** In this case, this node is included in the path and the path construction for the intended source node is complete. If list $\mathcal{L}$ is empty, the current iteration finishes and the algorithm will move on to the next iteration for the remaining nodes.
But list $\mathcal{L}$ may not be empty at this point, meaning that some source nodes of other overlapping paths still do not have a complete path to their corresponding destination nodes. As an example, in Fig. 4(b), the source node for which path is being constructed is s_0. Source nodes s_1 and s_2 are included in the path and thus are in $\mathcal{L}$. When the destination node d_0 (for s_0) is included in the path, the path construction for s_0 is complete. But the paths for s_1 and s_2 remain incomplete.
If $\mathcal{L}$ is not empty, the iteration continues by taking a source node from $\mathcal{L}$ as the current intended source node and continues path construction for this node. In our algorithm, if there are multiple source nodes in $\mathcal{L}$, we pick the source node that has the largest share of out-going flow at the current encountered node. Once chosen, this source node is removed from $\mathcal{L}$ and the path construction continues.
In the case that the current encountered node is not carrying any flow for any of the source nodes in $\mathcal{L}$, then all the source nodes in $\mathcal{L}$ will be removed from the current path as well as from the list $\mathcal{L}$. This is done by removing a source node from current path (which was in list $\mathcal{L}$) and directly connecting its preceding and succeeding nodes in the path. For example, in Fig. 4(b), the resulting path by removing s_1 and s_2 will be s_0-r_2-r_4-d_0. At this point, list $\mathcal{L}$ is empty and the current iteration finishes. The algorithm will move on to the next iteration for remaining nodes.

C. **This node is the destination node whose source node is not on the current path under construction.** In this case, this node must not be included in the path and the node receiving the next largest flow will be considered. This scenario is illustrated in Fig. 4(c).
In Fig. 4(c), when d_2 is considered for the next node of r_1, d_2 will not be included in the path because s_2 is not on the current path. As a result, another node r_2 will be considered and included as the next node for r_1. This will ensure that a different path can be set up for s_2 in a future iteration.

Upon the completion of an iteration of path determination, the list $\mathcal{L}$ must be empty. The algorithm will then moves on to the next iteration of path determination for the remaining source nodes whose paths are not yet determined. Note that all the source nodes, destination nodes, and relay nodes that are already included in a path are removed from further consideration, due to the physical layer constraint we discussed earlier. The iteration continues until all the paths for all the source nodes are determined. The pseudo-code for this phase is omitted due to space limit.



(a) Case 2.2(A): The source of the encountered destination node is in $\mathcal{L}$.

(b) Case 2.2(B): The encountered node is the destination node of the current path under construction.

(c) Case 2.2(C): Source node for d_2 is not on the path under construction.

Fig. 4. Examples illustrating different scenarios in Sub-case 2.2 during Phase 1 of FSC algorithm.

2) Phase 2: CR Assignment. After Phase 1, there may still be some nodes in $\mathcal{N}_r$ that are not yet used and are thus available to serve as CR nodes. The goal of Phase 2 is to consider how to assign these remaining relay nodes as CR nodes to increase capacity.

In our algorithm, we use the results for A^w_{uv}'s in the relaxed solution to assign these remaining relay nodes. First, we introduce a term called *capacity-flow-ratio* (CFR) for a hop as the ratio of the hop's capacity (assuming direct transmission without CC) to the number of overlapping sessions (as determined by Phase 1) on that hop. Then, we order the hops among all paths in Phase 1 in non-decreasing order of CFR. The assignment of CRs start with the hop with the minimum CFR and works in increasing order. For a particular hop, say (u, v), under consideration, we choose the CR node with the largest A^w_{uv} value among all CR nodes from the relaxed solution. In the case that the largest A^w_{uv} is 0 for this hop in the relaxed solution, no CR node will be assigned to this hop. The iteration continues until all the hops are considered for CR node assignment or the remaining available CR nodes are all assigned.

3) Phase 3: Flow Re-calculation. Upon the completion of Phases 1 and 2, all the integer variables $A^w_{u,v}$'s and B_{uv}'s are now fixed (either 0 or 1). Consequently, the original MILP (in Section IV-C) is now reduced to an LP with variables R_{min} and $f_{uv}(s_i)$'s. In Phase 3, which we call Flow Re-calculation, we solve this LP and obtain a feasible solution for $f_{uv}(s_i)$'s. The value of the objective function obtained via this LP can be used as lower bound in the branching process.

C. Generating a Cutting Plane

The process of adding a cutting plane, which is a linear constraint, begins by examining the values of A^w_{uv}'s and B_{uv}'s in the solution to a relaxed MILP problem. If there are multiple A^w_{uv}'s or B_{uv}'s with fractional values, then a decision must be made on which one of the fractional A^w_{uv}'s or B_{uv}'s will be chosen to generate a cutting plane. Such decision should exploit some problem-specific property and our algorithm is based on the following observation.

In our problem, if any A^w_{uv} or B_{uv} variable is assigned to 1, then some other relevant variables can immediately be assigned to 0. For example, if A^w_{uv} is assigned to 1, then node w cannot be used as MR on any path. So, if we can increase the chances of a variable getting assigned to 1, then we may quickly fix a number of other relevant variables and reduce the problem size for future branching process. Thus in our algorithm, we propose to choose a variable that is closest to the value of 1 in the relaxed solution when generating a cutting plane.

After selecting the variable for the cutting plane, the next step is to generate a linear constraint as the cutting plane, for which we can use a standard procedure in [13].

D. Selection of Branching Variable

When the addition of cutting planes is no longer able to offer much improvement in upper and lower bounds for a relaxed problem, we move on to the branching process in the algorithm. The candidate variables for branching are those A^w_{uv}'s and B_{uv}'s with fractional values in the relaxed solution. Although the choice of A^w_{uv}'s or B_{uv}'s will not affect the convergence of the branch-and-cut algorithm, a wise choice of branching variable can significantly speed up the convergence time.

In our solution procedure, we choose an A^w_{uv} or B_{uv} that is nearest to either 0 or 1 for branching. Once chosen, the current problem is then divided into two subproblems, with the value of branching variable fixed as 0 in one subproblem and 1 in the other subproblem. Our choice of the branching variable is based on the following reasoning. If the variable that is closest to 0 is chosen, then in the two subproblems, it will have the value 1 in one subproblem and 0 in the other. For the subproblem with its value of 1, the new upper bound may be reduced significantly. Consequently, this subproblem (with the branching variable of 1) has the potential of having an upper bound lower than the current lower bound, making it eligible to be removed from the problem list for further consideration. Similar argument also holds for the case when the variable chosen is closest to 1.

VI. Numerical Results

In this section, we present some numerical results to demonstrate the rate gains that can be achieved by jointly optimizing relay node assignment and flow routing in multi-hop wireless networks. We also compare the results under our solution with that when CC is not used.

A. Simulation Setting

In our simulations, we set $W = 22$ MHz bandwidth for each channel. The maximum transmission power at each node

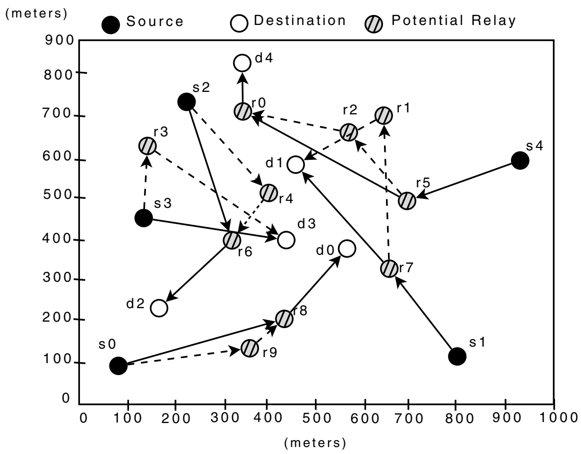


Fig. 5. Jointly optimal CR node assignment and flow routing for the 20-node network.

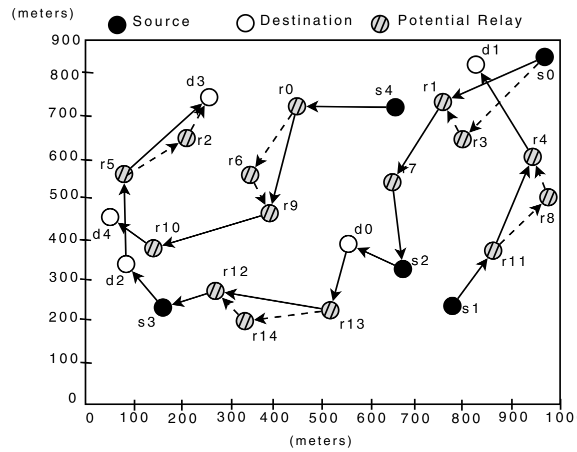


Fig. 7. Jointly optimal CR node assignment and flow routing for the 25-node network.

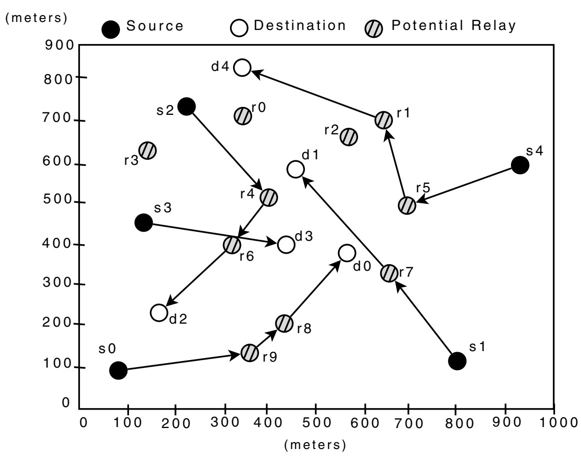


Fig. 6. Optimal flow routing solution for the 20-node network (CC is not employed).

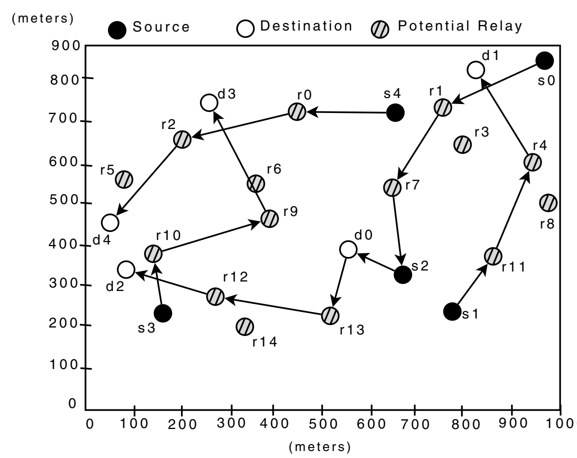


Fig. 8. Optimal flow routing solution for the 25-node network (CC is not employed).

TABLE I
THROUGHPUT COMPARISON FOR THE 20-NODE NETWORK

Session	With CC (Mb/s)	Without CC (Mb/s)
$s_0 - d_0$	36.1	29.2
$s_1 - d_1$	30.3	20.5
$s_2 - d_2$	32.3	22.0
$s_3 - d_3$	28.3	23.0
$s_4 - d_4$	32.2	27.3

TABLE II
THROUGHPUT COMPARISON FOR THE 25-NODE NETWORK

Session	With CC (Mb/s)	Without CC (Mb/s)
$s_0 - d_0$	44.5	31.9
$s_1 - d_1$	47.4	34.8
$s_2 - d_2$	50.4	42.1
$s_3 - d_3$	49.0	31.1
$s_4 - d_4$	46.6	34.8

is set to 1 W. For simplicity, we assume that h_{sd} only includes the propagation gain between nodes s and d and is given by $|h_{sd}|^2 = ||s-d||^{-4}$, where $||s-d||$ is the distance (in meters) between nodes s and d and path loss index is 4. For the AWGN channel, we assume the variance of noise is 10^{-10} W at all nodes. For our $(1-\epsilon)$-optimal solution, we set $\epsilon = 0.1$ in all cases.

B. Results

We present our results for two different network settings, one with 20 nodes and another with 25 nodes.

20-node Network. The location of each node in this 20-node network can be inferred from Fig. 5. There are five sessions in the network, with five source nodes ($N_s = 5$), five destination nodes ($N_d = 5$), and ten potential relay nodes ($N_r = 10$) in the network.

Figure 5 shows our solution for CR node assignment and multi-hop routing for the 20-node network setting. The rate of each session is given in Table I (second column), with the minimum rate among all sessions being 28.3 Mb/s.

For comparison, Fig. 6 shows the solution for the 20-node network when CC is not employed (i.e., only flow routing is optimized). The problem formulation in this case is a simplified version of the formulation given in Section IV-C, i.e., all A_{uv}^{w} variables are set to 0. The solution for this problem only consists of flow routing for each session. The rate of each session when CC is not employed is given in Table I (third column), with the minimum rate among all sessions being 20.5 Mb/s, which is less than the case when CC is employed (28.3 Mb/s).

25-node Network. The location of each node in a 25-node network can be inferred from Fig. 7. There are five sessions in the network, with five source nodes ($N_s = 5$), five destination nodes ($N_d = 5$), and fifteen potential relay nodes ($N_r = 15$).

For this 25 node network, Fig. 7 shows our solution for joint CR node assignment and flow routing. The rate of each session is shown in Table II (second column), with the minimum rate among all sessions being 44.5 Mb/s. As a comparison, Fig. 8 shows the optimal flow routing solution when CC is not used. The rate of each session when CC is not used is shown in Table II (third column) with the minimum rate among all sessions being 31.1 Mb/s, which is less than 44.5 Mb/s.

VII. CONCLUSION

In this paper, we demonstrated the benefits of using CC in multi-hop wireless networks by performing joint optimization of cooperative relay node assignment and multi-hop flow routing for concurrent sessions. This optimization problem is inherently difficult due to its mixed-integer nature and very large problem space. To solve the problem, we developed an efficient solution procedure based on branch-and-cut framework with several novel components to speed up the computation. Our results demonstrated the significant rate gains that can be achieved by incorporating CC in multi-hop wireless networks.

For future work, we believe that many enhanced models such as those described in [10] can be explored in multi-hop context.

ACKNOWLEDGMENTS

The work of Y.T. Hou, Y. Shi, and S. Sharma was supported in part by NSF Grant CCF–0946273. The work of H.D. Sherali was supported in part by NSF Grant CMMI–0754236. The work of S. Kompella was supported in part by the ONR.

REFERENCES

[1] M.S. Bazaraa, H.D. Sherali, and C.M. Shetty, Nonlinear programming: Theory and algorithms, 3rd edition. *Wiley,* New York, 2006.

[2] J. Cai, S. Shen, J. W. Mark, and A. S. Alfa. Semi-distributed user relaying algorithm for amplify-and-forward wireless relay networks. *IEEE Transactions on Wireless Communications*, 7(4):1348–1357, April 2008.

[3] T.M. Cover and A.E. Gamal. Capacity theorems for the relay channel. *IEEE Transactions on Information Theory*, 25(5):572–584, 1979.

[4] M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman and Company, New York, 1979.

[5] D. Gunduz and E. Erkip. Opportunistic cooperation by dynamic resource allocation. *IEEE Transactions on Wireless Communications*, 6(4):1446–1454, April 2007.

[6] O. Gurewitz, A. de Baynast, and E.W. Knightly. Cooperative strategies and achievable rate for tree networks with optimal spatial reuse. *IEEE Transactions on Information Theory*, 53(10):3596–3614, October 2007.

[7] T.E. Hunter and A. Nosratinia. Diversity through coded cooperation. *IEEE Transactions on Wireless Communications*, 5(2):283–289, February 2006.

[8] 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 Transactions on Mobile Computing*, 6(5):579–594, June 2007.

[9] A.E. Khandani, J. Abounadi, E. Modiano, and L. Zheng. Cooperative routing in static wireless networks. *IEEE Transactions on Communications*, 55(11):2185–2192, November 2007.

[10] G. Kramer, I. Maric, and R.D. Yates, *Cooperative communications.* Foundations and Trends in Networking, Now Publishers, June 2007.

[11] 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*, 50(12):3062–3080, December 2004.

[12] F. Li, K. Wu, and A. Lippman. Energy-efficient cooperative routing in multi-hop wireless ad hoc networks. In *Proc. 25th IEEE International Performance, Computing, and Communications Conference*, pages 215–222, April 10–12 2006.

[13] Y. Pochet and L.A. Wolsey, *Production Planning by Mixed Integer Programming.* Springer, New York, 2006.

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

[15] S. Savazzi and U. Spagnolini. Energy aware power allocation strategies for multihop-cooperative transmission schemes. *IEEE Journal on Selected Areas in Communications*, 25(2):318–327, February 2007.

[16] A. Scaglione, D.L. Goeckel, and J.N. Laneman. Cooperative communications in mobile ad hoc networks. *IEEE Signal Processing Magazine*, 23(5):18–29, September 2006.

[17] A. Sendonaris, E. Erkip, and B. Aazhang. User cooperation diversity – part i: system description. *IEEE Transactions on Communications*, 51(11):1927–1938, November 2003.

[18] A. Sendonaris, E. Erkip, and B. Aazhang. User cooperation diversity – part ii: implementation aspects and performance analysis. *IEEE Transactions on Communications*, 51(11):1939–1948, November 2003.

[19] Y. Shi, S. Sharma, Y.T. Hou, and S. Kompella. Optimal relay assignment for cooperative communications. In *Proceeding of ACM MobiHoc*, pages 3–12, Hongkong, China, May 27–30 2008.

[20] L. Tassiulas and A. Ephremides. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. *IEEE Transactions on Automatic Control*, 37(12):1936–1948, 1992.

[21] E.C. van der Meulen. Three terminal communication channels. *Advances in Applied Probability*, 3:120–154, 1971.

[22] V.V. Vazirani. *Approximation Algorithms*. Springer Verlag, Berlin, Germany, 2001.

[23] B. Wang, Z. Han, and K.J.R. Liu. Distributed relay selection and power control for multiuser cooperative communication networks using buyer/seller game. In *Proc. IEEE INFOCOM*, pages 544–552, Anchorage, Alaska, May 6–12 2007.

[24] E.M. Yeh and R.A. Berry. Throughput optimal control of cooperative relay networks. *IEEE Transactions on Information Theory*, 53(10):3827–3833, October 2007.

[25] Y. Zhao, R. Adve, and T.J. Lim. Improving amplify-and-forward relay networks: optimal power allocation versus selection. In *Proc. IEEE International Symposium on Information Theory*, pages 1234–1238, Seattle, USA, July 9–14 2006.