HAL open science

End-to-End Throughput with Cooperative Communication in Multi-channel Wireless Networks

Zheng Huang, Xin Wang, Baochum Li

To cite this version:

Zheng Huang, Xin Wang, Baochum Li. End-to-End Throughput with Cooperative Communication in Multi-channel Wireless Networks. 9th International IFIP TC 6 Networking Conference (NETWORKING), May 2010, Chennai, India. pp.327-338, 10.1007/978-3-642-12963-6_26 . hal-01059109

HAL Id: hal-01059109

https://inria.hal.science/hal-01059109v1

Submitted on 29 Aug 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution 4.0 International License

Work in Progress: End-to-End Throughput with Cooperative Communication in Multi-channel Wireless Networks

Zheng Huang†, Xin Wang†, and Baochun Li‡

†School of Computer Science
Shanghai Key Laboratory of Intelligent Information Processing
Fudan University, Shanghai, 200433, P. R. China
‡Department of Electrical and Computer Engineering
University of Toronto
Corresponding author's mail: xinw@fudan.edu.cn

Abstract. Although cooperative communication has been proposed at the physical layer to address multi-path fading effects, how physical layer gains with cooperative communication can translate to tangible performance benefits in end-to-end flows remains to be an open problem. This paper represents one step forward towards a deeper understanding of the interplay between end-to-end throughput and physical layer cooperative communication, in the general context of multi-hop multi-channel wireless networks. Based on a decode-and-forward physical layer design with rateless codes, we reformulate the problem of routing and channel assignment to account for physical layer cooperation. We design a distributed protocol to solve the new problem. Our simulation results have validated the effectiveness of our protocol to offer a substantial gain with respect to stabilizing the offered aggregate throughput in the network.

Key words: cooperative communication, channel assignment, multi-hop multi-channel wireless networks

1 Introduction

Cooperative communication has been proposed as a powerful physical layer technique to combat fading [1] or to increase the physical layer capacity [2] in wireless relay networks. A basic model studied in the research of cooperative strategies is a "triangle" network [3], which consists of a source S, a destination D, and a relay node R. In the transmission from S to D, the relay R can cooperate with S to jointly forward the packets to D. Cooperation can improve the channel capacity when the channel (S, R) and (R, D) have a higher quality than the channel (S, D).

While there is a large body of literature focusing on various cooperative communication strategies, most of these results are limited to the triangle network or its generalizations. Such triangle networks have a two-hop topology, which is difficult for these cooperative communication strategies to be extended directly

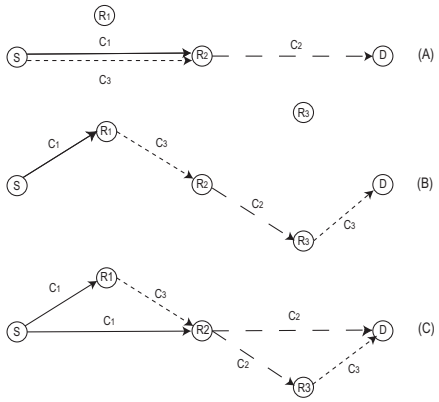

Fig. 1. An example showing the benefits of cooperative communication in multi-hop wireless networks. (A) (B) Conventional channel assignment with no cooperation in two fading cases. (C) Channel assignment with cooperation.

to its multi-hop counterpart. On the other hand, as pointed out in [4], while there exists a large volume of literature on cooperative communication strategies in the physical layer, there are very few higher layer protocols that can take advantage of physical layer cooperation in a multi-hop network setting.

In this paper, our objective is to make use of cooperative communication strategies in *multi-hop* and *multi-channel* wireless networks. In multi-channel networks, each node is equipped with several wireless interfaces, each of which can be tuned to a channel from an orthogonal set of channels. We seek to improve the aggregate end-to-end network throughput by allowing nodes to cooperate in the physical layer, which is a challenge not fully explored in the literature.

As a preamble of our work, a motivating example that involves a multi-hop and multi-channel network can be described as follows. Consider a wireless network with five nodes, labeled as S, R_1, R_2, R_3 and D in Fig. 1. We assume that there is a unicast traffic from S to D. We have three orthogonal channels C_1, C_2 and C_3 to assign on nodes' interfaces, with an objective of maximizing the unicast throughput. Channel capacity is dependent on which link the channel is placed, *i.e.*, channel diversity is considered. Nodes can transmit simultaneously via orthogonal channels without interference. By conventional channel assignment with no cooperation, we may assign C_1, C_3 on (S, R_2) and C_2 on (R_2, D), as illustrated in Fig. 1(A). The throughput is thus bounded by the minimum of the channel capacity of C_2 and the sum of that of C_1 and C_3 on respective links. However, it can occur that the channel (S, R_2) or (R_2, D) has a very poor quality. In this case, a better alternative may exist to assign channels on (S, R_1), (R_2, R_3), (R_1, R_2) and $(R_3 - D)$ as in Fig. 1(B).

Cooperative communication in the physical layer, on the other hand, provides a new insight to this example scenario. As in Fig. 1(C), S, R_1 and R_2 can form a local "triangle" for cooperation, while R_2, R_3 and D forming another.

In particular, S (R_2) can broadcast the packets to R_1 (R_3) and R_2 (D) by a common channel C_1 (C_2) at first, and then R_1 (R_3) helps to forward the packets to R_2 (D) by another channel. Because the capacity from S to R_2 and R_2 to D can both increase via cooperation, the unicast throughput from S to D is thus improved.

From this example, it is clear that, although physical layer cooperation improves end-to-end throughput in multi-hop and multi-channel networks, it is non-trivial to design distributed protocols to realize such a performance gain. There is a tradeoff between throughput improvement and temporary increase of network congestion.

In this paper, we seek to make use of physical layer cooperative communication strategies to improve the aggregate end-to-end throughput in multi-hop and multi-channel networks. To our knowledge, this has not been fully explored in prior work. Towards this objective, we reformulate the routing and channel assignment problem to account for our physical layer model for cooperation. In this context, we propose a new concept, *cooperative link*, as the component of a cooperative routing path. We have designed a decentralized protocol to maximize aggregate end-to-end throughput.

2 Related Work

Our work builds upon cooperative communication strategies that have been studied thoroughly at the physical layer, such as amplify-and-forward [5], decode-and-forward [6], compress-and-forward [7] and compute-and-forward [8]. Most of these studies are from an information-theoretic perspective. In contrast, the objective of this paper is to translate the physical-layer capacity improvement via cooperation to network-layer throughput benefits. In this sense, our work is not directly related to recent advances in cooperative diversity (e.g., [1]).

From the perspective of the network layer, we select decode-and-forward rather than the other three strategies as the underlying strategy for cooperative communication, because with decode-and-forward it is flexible enough to incorporate cooperation into the multi-hop and multi-channel network model. Specifically, we use a coding scheme based on rateless codes in [9] to formulate the underlying cooperative model.

We note that there has been existing work on translating physical layer gains of cooperative communication to the high layer performance benefits, all of which uses the decode-and-forward strategy in the system model. In [4], a cross-layer approach has been proposed to exploit cooperative diversity in single-channel ad hoc networks. It provides few insights on how cooperative communication could be used to improve the network performance. In contrast, we identify a clear underlying cooperative model that could be analyzed quantitatively from the perspective of a higher layer. In addition, we consider multi-channel rather than a single-channel networks.

One common feature in most existing works in the area of multi-radio or multi-channel networks is that packets are forwarded along a chain of point-to-

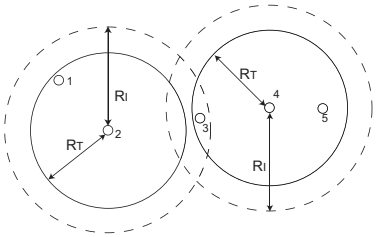

Fig. 2. An example exhibiting the interference relationship. The solid and dash circles indicate the transmission range and interference range, respectively. Link $(3, 4)$ interferes with link $(1, 2)$. However, link $(4, 5)$ can operate simultaneously with link $(1, 2)$ via a common channel since there is no interference.

point links. We believe that considering cooperation in multi-channel networks is inherently attractive. More recently, there has been work that considered network coding in multi-channel networks [10]. Compared with its system model, we used a different underlying cooperative communication strategy, and assume that a channel can have different capacities on different links due to fading effects at different locations. In addition, our use of dynamic channel assignment is more flexible than static channel assignment in [10].

3 System Model

3.1 Network Model

We consider a wireless network $N = (V, E)$ with a set of stationary wireless nodes. There are a total of K orthogonal channels denoted by $\mathcal{C} = \{c_1, c_2, ..., c_K\}$ in the network. Each wireless node $v \in V$ is equipped with κ_v channel interfaces. A *channel assignment* $\mathcal{A}$ assigns a collection A_v of κ_v channels to node v, with each interface on v tuned to a channel from A_v, $A_v \subset \mathcal{C}$. A half-duplex model is assumed on each channel.

We assume each wireless node uses a fixed transmission power. It follows that there is a fixed *transmission range* R_T and *interference range* $R_I > R_T$ associated with each node. As each channel may have a different capacity at different locations in the network, we denote R_{ij}^c as the capacity of channel c on the link (i, j), provided that there is no interference. We use the disk model [11] to account for the interference relationships (Fig. 2).

For traffic flows, we assume that there is a collection $\mathcal{S}$ of elastic unicast flows in which each session $m \in \mathcal{S}$ runs concurrently between a pair of wireless nodes m_s and m_t. We denote the achieved throughput of a unicast flow m by σ_m.

3.2 Cooperative Communication Model

A basic cooperative communication opportunity involves the operation of three nodes, namely, s, r and t as in Fig. 3. Let node s communicate with node t.

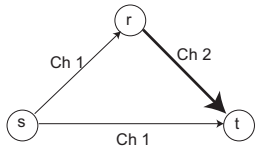

Fig. 3. The underlying cooperative communication model. Node s broadcasts the packets via a common channel 1. Node r, once decoding the message, helps to re-encode and forward to node t via another channel 2. Rateless codes are used as the coding scheme.

If no cooperation is present, node s uses a block coding scheme to encode the information, and forwards the coded packets on channel 1. At the other side of channel 1, node t decodes the packet and recovers the original information. The achievable rate of traffic is denoted as R^1_{st}. Alternatively, when node s forwards a packet to t, node r can overhear this packet since channel 1 is a "broadcast" channel. If we assume $R^1_{st} < R^1_{sr}$, it follows that node r manages to decode the packet before node t. In the remaining time, node r can help node s to re-encode and forward the packet to node t via channel 2.

The achievable rate with this cooperative communication strategy has been proved in [12] and later extended in [9]: Let $f := \begin{cases} \frac{R^1_{st}+R^2_{rt}}{R^1_{sr}+R^2_{rt}} & \text{if } R^1_{st} < R^1_{sr}, \\ 1 & \text{otherwise.} \end{cases}$ and let $R := fR^1_{sr} = fR^1_{st} + (1-f)(R^1_{st} + R^2_{rt}) = R^1_{st} + (1-f)R^2_{rt}$. Then for any $\delta > 0$, there exists a block coding scheme at rate $R - \delta$ such that with increasing block length, the decoding error probability is driven arbitrarily close to 0.

Rateless code has been argued in [9] to facilitate node s to choose such a rate, without channel state information. *Fountain codes*, for example Raptor [13] and LT [14] codes, are typical forms of rateless codes.

4 Routing and Channel Assignment with Cooperative Communication

4.1 Cooperative Routing

We propose the notion of *cooperative links* to describe the cooperative routing path. A *cooperative link* consists of three parts: there is a single source s, a single receiver t, and a potential set B of relay nodes. We denote the set of cooperative links as $E_c = \{(i, B, j), i \in V, j \in V, B \subset V\}$. For the sake of simplicity, in the following analysis, we restrict to the case when $|B| = 1$.

For each unicast session $m \in \mathcal{S}$, a *cooperative path* between m_s and m_t can be expressed as a chain of links from $E \cup E_c$. For example, the cooperative path from node 1 to 7 as in Fig. 4 can be written as $\{(1, 2), (2, \{3\}, 4), (4, 5), (5, \{6\}, 7)\}$.

We assume that the entire system operates according to a "universal" clock, which divides time into slots of unit length. Each link in $E \cup E_c$ is associated with a queue. Let $\{q_l(t) : l \in E \cup E_c\}$ denote the number of packets queued at link l

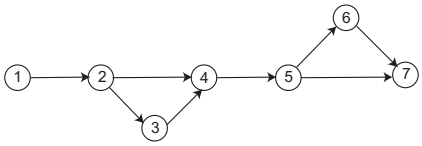

Fig. 4. A routing path from node 1 to 7 can be expressed as $\{(1,2),(2,\{3\},4),(4,5),(5,\{6\},7)\}$. Here $(2,\{3\},4)$ and $(5,\{6\},7)$ are cooperative links.

waiting to be served before time t. Suppose we allow for multi-path routing for each unicast session m. Let $K(m)$ denote the set of routing paths between m_s and m_t, and P_{mk} denote the fraction of traffic of m that is routed on the path $k \in K(m)$. Here, $\sum_{k \in K(m)} P_{mk} = 1$ for all m. We denote the routing matrix as G. If the kth route of the unicast session m passes through link l, then $G^l_{mk} = 1$. Otherwise, $G^l_{mk} = 0$.

The cooperative routing problem considered in multi-channel and multi-hop networks is to determine the cooperative path of each unicast flow and determine the routing matrix G^l_{mk} and the fraction P_{mk}, if multi-path routing is considered.

4.2 Cooperative Channel Assignment

We adapt a dynamic channel assignment model as in [15–17]. That is, the interface on nodes can switch to a different channel dynamically from one time slot to the next. We use $\mathcal{A}(t)$ to denote the channel assignment in time slot t. It is required that under each $\mathcal{A}(t)$, there is no interference in the network.

5 A Distributed Cooperative Protocol

Our distributed protocol consists of two parts. The first part is cooperative routing, which facilitates wireless nodes to discover local cooperative communication opportunities in a distributed manner. The output of this part is a chain of direct and cooperative links that could be used as a routing path. Also, for multi-path routing, the traffic fraction P_{mk} and routing matrix G^l_{mk} are determined. The second part is to assign channels on direct and cooperative links that have been determined in the first part.

5.1 Cooperative Routing Protocol

Our routing protocol involves two stages. The first stage is a direct link routing discovery that operates similar to a traditional single-path routing scheme. Specifically, for a given unicast session m, we use the hop-count as our routing metric, and assume that each wireless node can measure its distance to the destination m_t. The source node m_s broadcasts a probing packet which all of the neighbors of the source in the transmission range could overhear. A neighbor

chooses to forward (in a broadcast manner) the probing packet if its distance to the destination is less than that of the last-hop predecessor. This process continues until the destination node is reached. It follows that a single path of a chain of direct links can thus be established between the source m_s and destination m_t.

The second stage is to find the potential cooperative opportunities, which is the key feature of the cooperating routing discovery. In doing so, each node along the path (formed in the first stage) picks up the node from its neighbors with the shortest distance from the current node, and selects this neighbor to form a cooperative link. We describe the detailed steps of these two stages in Algorithm 1.[1]

Algorithm 1 Cooperative Routing Discovery

Input: A unicast session m with source node m_s and destination node m_t.
Output: A chain P of direct or cooperative links as a single routing path.
Stage 1: Generate the direct link routing path.
Stage 2: Discover local cooperative communication opportunities.
current $= m_s$
loop
  if *current* $== m_t$ **then**
    return
  end if
  search in the neighborhood of *current* node $N(current)$ for a neighbor i with the smallest distance $d(i, m_t)$ to the destination m_t.
  next-hop $= i$
  Add (*current, next-hop*) to P.
  search in $N(current)$ for a neighbor j with the smallest distance $d(current, j)$ to the *current* node
  if j exists and $j \neq$ *next-hop* **then**
    Replace (*current,* i) with the cooperative link (*current,* $\{j\}$, i) in P
  end if
  current $=$ *next-hop*
end loop.

5.2 Cooperative Channel Assignment Protocol

The difficulty in generalizing the distributed algorithm of channel assignment under the direct link scenario to the cooperative link scenario arises from the inherent feature of cooperative links. In the direct link scenario, there is one channel assigned on one direct link. In contrast, in order to let one cooperative link, say link (i, B, j), be active, one needs to assign 2 channels on this link if

[1] Although we use the hop-count as the metric here, the algorithm is easily extended to other metrics such as the ETX model.

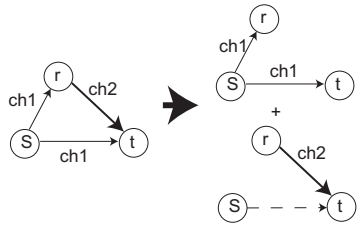

Fig. 5. A cooperative link $(s, \{r\}, t)$ can be decomposed into two "virtual" links. One is a virtual broadcast link $(s, \{\{r\}, t\})$, and the other is a virtual multiple-access link $(\{s, \{r\}\}, t)$. Only one channel is needed for each virtual link.

$|B| = 1$, as indicated in Section 3. This 2 (channel)-to-1 (link) mapping relationship makes it almost impossible to directly generalize the existing algorithm.

Our key idea to address this difficulty is to decompose a cooperative link (i, B, j) into two "virtual" links: one is a virtual broadcast link $(i, \{B, j\})$ and the other is a virtual multiple access link $(\{i, B\}, j)$. As in Fig. 5, initially a cooperative link $(s, \{j\}, k)$ needs two channels ($ch1$ and $ch2$) to perform the cooperation. Now we decompose it into two virtual channels: a virtual broadcast link $(s, \{\{r\}, t\})$ and a virtual multiple access link $(\{s, \{r\}\}, t)$. Each link now needs only one channel. Note that on the virtual link $(\{s, \{r\}\}, t)$, we do not need to assign an extra channel on the link from s to t (indicated by the dash line) since once we have assigned the channel on the virtual broadcast counterpart, we can use the same channel on s to t for the virtual multiple access link.

By the decomposition of cooperative links in E_c, we now have three kinds of links E, E_{cb} and E_{cm} in the network. E is the direct link set, E_{cb} is the virtual broadcast link set decomposed from E_c, and E_{cm} is the virtual multiple access link set from E_c. The interference relationship among these links can be generalized directly from that under the case of conventional direct links.

We define that a link l in $E \cup E_{cb} \cup E_{cm}$ interferes with a link $(i, \{B, j\})$ in E_{cb} or $(\{i, B\}, j)$ in E_{cm}, if at least one of the end points of l (there are two end points if $l \in E$ and three end points for $l \in E_{cb} \cup E_{cm}$) is located in the interference disk formed by $D_i \cup D_j \cup (\cup_{k \in B} D_k)$. We denote $\mathcal{I}$ as the *interference degree* of the network [15]. The interference degree $\mathcal{I}(i, j)$ defined on link (i, j) is the largest number of links in the interference range I_{ij} that do not interfere with each other. The interference degree $\mathcal{I}$ is then the largest $\mathcal{I}(i, j)$ over all links in the network.

Through the definition of the interference relationship, we can derive an *cooperative interference degree* (denoted by $\mathcal{I}_c$) of the network with cooperative links. Compared with $\mathcal{I}$, which is the interference degree of the same network but without cooperative links, we have $\mathcal{I}_c \leq \mathcal{I}$, since cooperative links replace direct links in the routing path, and one cooperative link itself is made up of several direct links in E.

Our algorithm is a generalization of the algorithm in [15] with the concept of virtual broadcast and multiple access links. We let each link $l \in E \cup E_{cb} \cup E_{cm}$

maintain $K+1$ queues. There is a queue q_l and a total of K channel queues u_l^c, one for each channel $c \in \mathcal{C}$ at link l. The channel assignment algorithm involves two steps.

We focus on the case in which there is only one single cooperative routing path for each session $m \in \mathcal{S}$. Since for multi-path cooperative routing, the set of routing paths has been generated in the cooperating routing phase, what we need to do is just to run the channel assignment in single path case iteratively on each of the cooperative link. Thus the focus on single-path case is sufficient.

Cooperative Channel Assignment Protocol with Single-path Routing

- Step 1: Let $x_l^c(t)$ be the maximum number of packets that could be assigned on link l through channel c in time slot t. For each link $l \in E \cup E_{cb} \cup E_{cm}$,

$$x_l^c(t) = \begin{cases} R_l^c \text{ if } \frac{q_l}{\alpha_l} \geq \frac{1}{R_l^c}(CA_l^c(\frac{u}{R}) + CE_l(\frac{u}{R})), \\ 0 \quad \text{otherwise.} \end{cases} \tag{1}$$

  Here, α_l is a positive constant chosen for link l, R_l^c is the capacity of link l when channel c is assigned on it, $CA_l^c(\frac{u}{R})$ is the *congestion cost* [15] at link l to use channel c, and $CE_l(\frac{u}{R})$ is the *interface cost* [15] at link l. For link $l = (i,j) \in E$, $CA_{ij}^c(\frac{u}{R}) = \sum_{e \in I_{ij}} \frac{u_e^c}{R_e^c}$. Note that link e, which interferes with link (i,j) may come from $E_{cb} \cup E_{cm}$. $CA_l^c(\frac{u}{R})$ for $l \in E_{cb} \cup E_{cm}$ can be defined in a similar manner.
  The calculation of R_l^c for link $l \in E_{cb} \cup E_{cm}$ is a little tricky. For $l = (i, \{\{j\}, k\}) \in E_{cb}(B = \{j\})$, we have $R_{i,\{\{j\},k\}}^c = R_{ij}^c$, since during the first phase of broadcast in the underlying cooperative communication strategy, only node j finishes decoding. For $l = (\{i, \{j\}\}, k) \in E_{cm}(B = \{j\})$, we have $R_{\{i,\{j\}\},k}^c = R_{jk}^c$ since there is actually one channel c assigned between j and k in the virtual multiple access link. The interface cost $CE_l(\frac{u}{R})$ for link $(i,j) \in E$ is defined as $\frac{1}{\kappa_i} \sum_{e \in E(i)} \sum_{d=1}^{K} \frac{u_e^d}{R_e^d} + \frac{1}{\kappa_j} \sum_{e \in E(j)} \sum_{d=1}^{K} \frac{u_e^d}{R_e^d}$. Here, the set $E(i)$ represents the set of links in $E \cup E_{cb} \cup E_{cm}$ that are adjacent with node i. For link $l = (i, \{\{j\}, k\}) \in E_{cb}$ or $l = (\{i, \{j\}\}, k) \in E_{cm}$, $CE_l(\frac{u}{R})$ is defined as $\frac{1}{\kappa_i} \sum_{e \in E(i)} \sum_{d \in \mathcal{C}} \frac{u_e^d}{R_e^d} + \frac{1}{\kappa_j} \sum_{e \in E(j)} \sum_{d \in \mathcal{C}} \frac{u_e^d}{R_e^d} + \frac{1}{\kappa_k} \sum_{e \in E(k)} \sum_{d \in \mathcal{C}} \frac{u_e^d}{R_e^d}$.
  In each time slot t, link l assigns $y_l^c(t) \in [0, x_l^c(t)]$ to each link channel u_l^c. The queueing dynamics of $q_l (l \in E \cup E_{cb} \cup E_{cm})$ is thus expressed as:

$$q_l(t+1) = q_l(t) + \sum_{m \in \mathcal{S}} G_m^l \sigma_m - \sum_{c \in \mathcal{C}} y_l^c(t). \tag{2}$$

  where $\sum_{c \in \mathcal{C}} y_l^c(t) = \min\{q_l(t), \sum_{c \in \mathcal{C}} x_l^c(t)\}$.
- Step 2: Based on the queue length at u_l^c, a maximal schedule $\mathcal{A}^c(t)$ is calculated by the distributed algorithm in [18]. Then at the end of time slot t, the queueing dynamics of $u_l^c(t)$ is expressed as:

$$u_l^c(t+1) = u_l^c(t) + y_l^c(t) - R_l^c 1_{\{l \in \mathcal{A}^c(t)\}}. \tag{3}$$

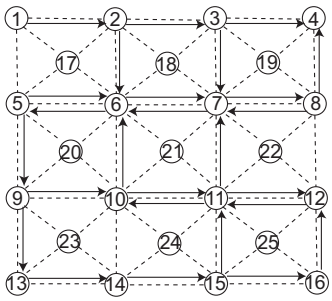

Fig. 6. The grid topology.

The following main result demonstrates the efficiency ratio of the above two-step algorithm.

Proposition 1: Assume each session $m \in \mathcal{S}$ uses a single cooperative routing path, and the routing matrix is given by G_m^l. The efficiency ratio of the proposed algorithm is $\frac{1}{\mathcal{I}_c+2}$, where $\mathcal{I}_c$ is the cooperative interference degree of the network.

The proof could be generated as a direct extension of that in [15]. We omit it here due to space constraints.

6 Simulations

To evaluate the performance of our protocol, we compare it with the best non-cooperative distributed algorithm in the literature [15]. To make a fair comparison, our simulation is based on a similar grid topology as shown in Fig. 6. There are 25 nodes, represented by the circles, and 60 direct links, represented by the dash lines. Compared with the topology in [15], we add nodes 17 to 25 to facilitate cooperative communication. As for interference relationship, we use the node-interference model as in [15, 19].

There are 8 orthogonal channels ($K = 8$) in the system. Each node is equipped with 8 interfaces ($\kappa_i = 8, i \in V$). The parameters α_l in Equation (1) are set as 10 and 100, for the cooperative and non-cooperative protocol, respectively. The capacity R_l^c is randomly chosen from 1 to 5, according to a uniform distribution. For traffic patterns, the unicast sessions are represented by the arrows in Fig. 6. Each session is assumed to have a uniform throughput of σ. The chain of arrows in Fig. 6 could be regarded as the conventional routing path generated in the first stage of our cooperative routing discovery protocol.

We plot the comparison of per-link mean queue backlog in Fig. 7. In our data statistics, we do not include those links with no traffic passing through, *e.g.*, link $(1, 5)$. From Fig. 7, two protocols behave at the same level when the offered throughput σ ranges from 3 to 5.6. However, from $\sigma = 6$, the backlog of the non-cooperative protocol in [15] increases to infinity quickly. As indicated in [15], this throughput σ could be viewed as the boundary of the capacity region. In contrast, as the offered throughput increases, the queue backlog with our

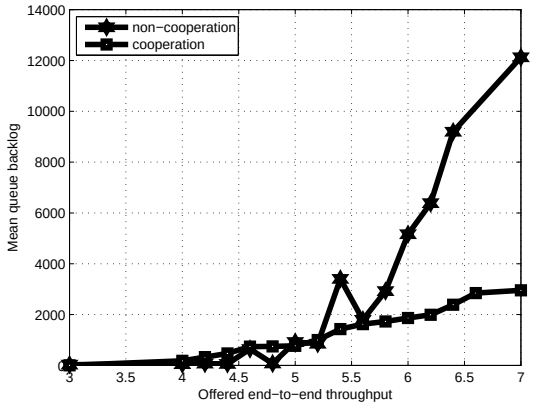

Fig. 7. Comparison of our cooperative protocol with the non-cooperative protocol.

cooperative protocol increases in a much slower manner. At $\sigma = 6$, the backlog is 5162 (vs. 1866 in the non-cooperative protocol); at $\sigma = 6.4$, the backlog is 9197 (vs. 2386); and at $\sigma = 7$, the backlog is 12126 (vs. 2954). The improvement thus ranges from 100% to 300%.

7 Conclusion

This paper explores how physical layer gains using cooperative communication strategies can translate to tangible performance for end-to-end flows at higher layers, in a general context of multi-channel networks. Based on a specific physical layer cooperative communication model with rateless codes, we reformulate the conventional routing and channel assignment problem. Moreover, we provide an efficient distributed protocol to solve these problems. Our simulation results have shown that, by using physical layer cooperation, our protocol performs 100% to 300% better, with respect to stabilizing the offered aggregate throughput, as compared to the best non-cooperative distributed protocol in the literature.

8 Acknowledgement

This work was supported in part by NSFC under Grant No. 60702054, Shanghai Municipal R&D Foundation under Grant No. 09511501200, the Shanghai Rising-Star Program under Grant No. 08QA14009.

References

1. J. N. Laneman, D. N. C. Tse, and G. W. Wornell, "Cooperative diversity in wireless networks: Efficient protocols and outage behavior," *IEEE Trans. Inf. Theory*, vol. 50, pp. 3062–3080, Dec. 2004.

2. G. Kramer, M. Gastpar, and P. Gupta, "Cooperative strategies and capacity theorems for relay networks," *Information Theory, IEEE Transactions on*, vol. 51, no. 9, pp. 3037–3063, 2005.
3. E. C. V. der Meulen, "Three-terminal communication channels," *Adv. Appl. Probab.*, vol. 3, pp. 120–154, 1971.
4. G. Jakllari, S. V. Krishnamurthy, and M. Faloutsos, "A framework for distributed spatio-temporal communications in mobile ad hoc networks," in *Proc. IEEE INFOCOM*, 2006.
5. A. Nosratinia, T. E. Hunter, and A. Hedayat, "Cooperative communication in wireless networks," *Communications Magazine, IEEE*, vol. 42, no. 10, pp. 74–80, 2004.
6. J. N. Laneman, G. W. Wornell, and D. Tse, "An efficient protocol for realizing cooperative diversity in wireless networks," in *Proc. IEEE Inter Symp Inform. Theory*, Jun. 2001.
7. T. Cover and A. E. Gamal, "Capacity theorems for the relay channel," *Information Theory, IEEE Transactions on*, vol. 25, no. 5, pp. 572–584, 1979.
8. B. Nazer and M. Gastpar, "Compute-and-forward: Harnessing interference with structured codes," in *Proceedings of the IEEE International Symposium on Information Theory (ISIT 2008)*, Toronto, Canada, July 2008.
9. J. Castura and Y. Mao, "Rateless coding over fading channels," *IEEE Comm. Lett.*, vol. 10, Jan. 2006.
10. X. Zhang and B. Li, "On the benefits of network coding in multi-channel wireless networks," in *Proc. IEEE SECON*, 2008.
11. J. Tang, G. Xue, and W. Zhang, "Interference-aware topology control and qos routing in multi-channel wireless mesh networks," in *Proc. ACM MobiHoc*, 2005.
12. P. Mitran, H. Ochiai, and V. Tarokh, "Space-time diversity enhancements using collaborative commnications," *IEEE Trans. Inf. Theory*, vol. 51, no. 6, pp. 2041–2057, Jun. 2005.
13. A. Shokrollahi, "Raptor codes," in *Proc. IEEE Inter Symp Inform. Theory*, 2004, p. 36.
14. M. Luby, "LT codes," in *Proc. 43rd Annual IEEE Symp. Fundations Computer Science (FOCS)*, 2002, pp. 271–282.
15. X. Lin and S. Rasool, "A distributed joint channel-assignment, scheduling and routing algorithm for multi-channel ad hoc wireless networks," in *Proc. IEEE INFOCOM*, 2007.
16. M. Kodialam and T. Nandagopal, "Characterizing the capacity region in multi-radio multi-channel wireless mesh networks," in *MobiCom '05: Proceedings of the 11th annual international conference on Mobile computing and networking.* ACM Press, 2005, pp. 73–87.
17. P. Bahl, R. Chandra, and J. Dunagan, "SSCH: slotted seeded channel hopping for capacity improvement," in *in IEEE 802.11 Ad-Hoc Wireless Networks, in ACM Mobicom*, 2004, pp. 216–230.
18. M. Hańćkowiak, M. Karoński, and A. Panconesi, "On the distributed complexity of computing maximal matchings," in *SODA '98: Proceedings of the ninth annual ACM-SIAM symposium on Discrete algorithms.* Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 1998, pp. 219–225.
19. B. Hajek and G. Sasaki, "Link scheduling in polynomial time," *IEEE Trans. Inf. Theory*, vol. 34, no. 5, pp. 910–91, 1998.