Energy-Efficient Topology Control in Cooperative Ad Hoc Networks

Ying Zhu, Minsu Huang, Siyuan Chen, and Yu Wang, *Senior Member*, *IEEE*

Abstract—Cooperative communication (CC) exploits space diversity through allowing multiple nodes cooperatively relay signals to the receiver so that the combined signal at the receiver can be correctly decoded. Since CC can reduce the transmission power and extend the transmission coverage, it has been considered in topology control protocols [1], [2]. However, prior research on topology control with CC only focuses on maintaining the network connectivity, minimizing the transmission power of each node, whereas ignores the energy efficiency of paths in constructed topologies. This may cause inefficient routes and hurt the overall network performance in cooperative ad hoc networks. In this paper, to address this problem, we introduce a new topology control problem: *energy-efficient topology control problem with cooperative communication*, and propose two topology control algorithms to build *cooperative energy spanners* in which the energy efficiency of individual paths are guaranteed. Both proposed algorithms can be performed in distributed and localized fashion while maintaining the globally efficient paths. Simulation results confirm the nice performance of all proposed algorithms.

Index Terms—Cooperative communication, topology control, energy efficiency, greedy algorithm, spanner.

1 Introduction

Wireless ad hoc networks have various civilian and military applications which have drawn considerable attentions in recent years. One of the major concerns in designing wireless ad hoc networks is to reduce the energy consumption as the wireless nodes are often powered by batteries only. Topology control [3], [4], [5], [6] is one of the key energy saving techniques which have been widely studied and applied in wireless ad hoc networks. Topology control lets each wireless node to select certain subset of neighbors or adjust its transmission power in order to conserve energy meanwhile maintain network connectivity.

Chen and Huang [7] first studied the strongly connected topology control problem, which aims to find a connected topology such that the total energy consumption is minimized. They proved such problem is NP-complete. Several following works [8], [9], [10], [11], [12], [13], [14] have focused on finding the minimum power assignment so that the induced communication graph has some "good" properties in terms of network tasks such as disjoint paths, connectivity or fault-tolerance. On the other hand, several localized geometrical structures [15], [16], [17], [18], [19], [20] have been proposed to be used as underlying topologies for wireless ad hoc networks. These geometrical structures are usually kept as few link as possible from the original communication graph and can be easily constructed using location information.

Recently, a new class of communication techniques, cooperative communication (CC) [21], [22], has been introduced to allow single antenna devices to take the advantage of the multiple-input-multiple-output (MIMO) systems. This cooperative communication explores the broadcast nature of the wireless medium and allows nodes that have received the transmitted signal to cooperatively help relaying data for other nodes. Recent study has shown significant performance gain of cooperative communication in various wireless network applications: energy efficient routing [23], [24], [25], [26], broadcasting [27], [28], [29], multicasting [30], connectivity/coverage improvement [31], [32], and relay selection for throughput maximization or energy conservation [33], [34], [35], [36].

The cooperative communication techniques can also be used in topology control to further reduce the transmission energy consumption [1] or to improve the network connectivity [2]. In [1], Cardei et al. first studied the topology control problem under CC model which aims to obtain a strongly connected topology with minimum total energy consumption. They first showed that this problem is NP-complete and then proposed two algorithms that start from a connected topology (the output of a traditional topology control algorithm) and further reduce the energy consumption using CC model. In [2], Yu et al. applied CC model in topology control to improve the network connectivity as well as reduce transmission power. Their algorithm first constructs all candidates of bidirectional links using CC model to connect different disconnected components, then generates MST structure to further reduce the energy consumption.

Even though the proposed solutions in [1], [2] can guarantee the network connectivity and reduce the *energy consumption* by constructing a sparse structure under CC model, they do not consider the *energy efficiency* of paths among nodes in the constructed structure. Fig. 1 shows an example network with n nodes. Assume that the link v_1v_n needs exactly the maximal transmission power at node v_1, and all other links need slightly smaller transmission power. In addition, we assume that there is no CC transmission possible due to the value of maximal transmission power and

- *The authors are with the Department of Computer Science, the University of North Carolina at Charlotte, 9201 University City Blvd., Charlotte, NC 28223. E-mail: {yzhu17, mhuang4, schen40, yu.wang}@uncc.edu.*

Manuscript received 8 July 2011; revised 2 Nov. 2011; accepted 14 Nov. 2011; published online 29 Nov. 2011.
Recommended for acceptance by E. Li.
For information on obtaining reprints of this article, please send e-mail to: tpds@computer.org, and reference IEEECS Log Number TPDS-2011-07-0453.
Digital Object Identifier no. 10.1109/TPDS.2011.293.

1045-9219/12/$31.00 © 2012 IEEE Published by the IEEE Computer Society

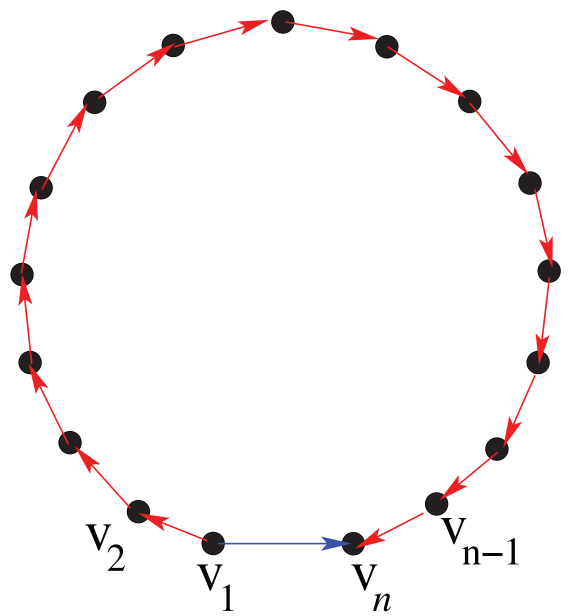


Fig. 1. Inefficiency of Current Topology Control Methods with CC: Solutions in [1], [2] will remove link v_1v_n to minimize the total energy consumption, thus will lead to an inefficient path from v_1 to v_n.

the signal-to-noise ratio (SNR) threshold. Then, if we apply the topology control algorithms in [1], [2], all of them will remove link v_1v_n to minimize the energy consumption. However, such operation will hurt the route between v_1 and v_n. The least energy path between them now (v_1 sends via $v_2 \ldots v_{n-1}$ to v_n) could be arbitrarily large compared with the optimal one (v_1 directly sends to v_n). It is also easy to construct a similar example where v_1v_n is a link via cooperative communication. Therefore, we hope that the sparseness of the constructed topology under CC model should not compromise too much on energy efficiency of communication paths. Such problem has been studied in traditional topology control algorithms (without CC) [17], [18], [20], [37], [38], [39]. In [17], Li et al. first introduced the concept of *energy spanner* into topology control. Here, a subgraph G' is called an *energy spanner* of a graph G if there is a positive real constant t such that for any two nodes, the energy consumption of the least energy cost path in G' is at most t times of the energy consumption of the least energy cost path in G. Several geometric structures [17], [18], [20] have been proved to be energy spanners. Recently, [37], [38], [39] also studied how to assign the transmission power for each node such that the induced communication graph is an energy spanner of the original communication graph meanwhile the total power level of all nodes is minimized. However, so far there is no energy spanner proposed under CC model yet.

Hereafter, we differentiate the terms of *energy consumption* and *energy efficiency* over a topology to represent two important but different energy concepts in topology control for wireless networks. *Energy consumption* of a topology is usually the total transmission power of all nodes or links in the topology; while *energy efficiency* of a topology is defined as whether the topology is an energy spanner to support energy efficient routing. Both concepts are important aspects for saving energy in wireless networks (one focuses on maintaining costs, while the other focuses on routing costs). These two concepts are contradict in topology control. A denser topology usually achieves better energy efficiency but may lead to higher energy consumption. Our energy efficiency topology control problem defined in this paper looks for the tradeoff between these two energy aspects.

In this paper, we study the energy efficient topology control problem with CC model by taking the energy efficiency of routes into consideration. Taking advantage of physical layer design that allows combining partial signals to obtain the complete data, we formally define *cooperative energy spanner* in which the least energy path between any two nodes is guaranteed to be energy efficient compared with the optimal one in the original cooperative communication graph. We then introduce the *energy-efficient topology control problem with CC* (ETCC), which aims to obtain a cooperative energy spanner with minimum total energy consumption, and prove its NP-completeness. Therefore, as solutions for ETCC, we propose two topology control algorithms to build energy-efficient cooperative energy spanners. Both algorithms can guarantee the bounded energy stretch factor and are easy to be implemented in a distributed and localized fashion. Our simulation results confirm the nice performance of these proposed algorithms.

The rest of this paper is organized as follows: In Section 2, we summarize related works in topology control for wireless networks. In Section 3, we introduce the network model used by our methods and formally define the cooperative energy spanner and the new corresponding topology control problem under CC model. Two topology control algorithms for constructing energy-efficient cooperative energy spanners are then proposed in Section 4. Section 5 discusses how to perform the proposed algorithms in localized fashion while maintaining the stretch factor bounds. Section 6 presents the simulation results. Finally, Section 7 concludes the paper by pointing out some possible future directions. A preliminary conference version of this paper appeared in [40].

2 Related Work

Topology control has drawn a significant amount of research interests in wireless ad hoc networks [8], [9], [10], [11], [12], [13], [14], [17], [18], [19], [20]. Primary topology control algorithms aim to maintain network connectivity and conserve energy by selecting certain subset of neighbors and adjusting the transmission power of wireless nodes. Comprehensive surveys of topology control can be found in [3], [4], [5], [6].

2.1 Topology Control for Connectivity

The strongly connected topology problem with a minimum total energy consumption was first studied by Chen and Huang [7]. They proved that to find a connected topology such that the total energy consumption is minimized is NP-complete. An approximation algorithm with a performance ratio of 2 is given when the links are symmetric. In last decade, variations of this problem (with symmetric or asymmetric links) have been studied and many approximation algorithms have been proposed [8], [9], [10], [11], [12]. In addition, several localized geometrical structures [15], [16], [17], [18], [19], [20] have been proposed to be used as topologies for wireless ad hoc networks. These geometrical structures can be constructed using neighbors' location information in order to remove as many links as possible from the original communication graph. Each node only selects certain neighbors (neighbors in the constructed structure) for communication.

2.2 Topology Control for Energy Efficiency

Besides connectivity guarantee, a good network topology should also be *energy efficient*, i.e., the total energy consumption of the least energy cost path between any two nodes in the final topology should not exceed a constant factor of the power consumption of the least energy cost path in the original network. In [17], Li et al. first introduced the concept of *energy spanner* into topology control. A subgraph G' is called an *energy spanner* of a graph G if there is a positive real constant t such that for any two nodes, the energy consumption of the least energy cost path in G' is at most t times of the energy consumption of the least energy cost path in G. The constant t is called the *energy stretch factor* and G' is called an energy t-spanner of G. A power spanner of the original communication graph is usually energy efficient for routing, since it guarantees that there is an efficient path between each pair of nodes. Several geometrical topologies [17], [18], [20] are energy spanners of unit disk graph (where every node uses its maximum transmission power) while others [15], [16], [19] are not. In [37], Wang and Li introduced a new topology control problem: the *minimum power energy spanner problem*, which aims to find the optimum transmission power of each individual node such that 1) the induced communication graph is an energy t-spanner of the original communication graph; and 2) the total power level of all nodes is minimized. They first proved that the problem is NP-complete and then presented two heuristics for the construction of a low cost power assignment with an energy spanner property for unit disk graphs. Recently, Shpungin and Segal [38] provided the first approximation algorithm for this problem. They first presented a basic method to construct an energy t-spanner such that the total power consumption is at most $\beta \cdot n$ times of the optimal solution, for any $t > 1$ and $\beta \geq 1 + \frac{2}{t-1}$. Then, they generalized the basic method for a randomly distributed network to build an energy $O((1+a)\frac{n-m}{m}\log n + a)$-spanner with high probability such that the total power is at most $(\beta \cdot m + 2)$ times of the optimal, for any $a > 1$, $\beta \geq 1 + \frac{2}{a-1}$, and any positive integer $m \leq n$. Abu-Affash et al. [39] presented a constant approximation algorithm for the minimum power energy spanner problem. Their method can build a planar energy t-spanner such that the total power is at most $2(1 + \frac{2}{t-1})$ times of the optimal when the underlying communication graph is a completed graph or a unit disk graph.

2.3 Topology Control with Cooperative Communication

The appearance of cooperative communication models also injects a new element in the topology control area. In [1], Cardei et al. first studied the topology control problem under cooperative communication model (denoted by TCC) which aims to obtain a strongly connected topology with minimum total energy consumption. They first showed that this problem is NP-complete and then proposed two algorithms that start from a connected topology assumed to be the output of a traditional (without using CC) topology control algorithm and reduce the energy consumption using CC model. The first algorithm (DTCC) uses 2-hop neighborhood information where each node tries to reduce the overall energy consumption within its 2-hop neighborhood without hurting the connectivity under CC model. The second algorithm (ITCC) starts from a minimum transmission power, and iteratively increases its power until all nodes within its 1-hop neighborhood are connected under CC model. Observing that the CC technique can also extend the transmission range and thus link disconnected components, Yu et al. [2] applied CC model in topology control to improve the network connectivity as well as reduce transmission power. Their algorithm first constructs all candidates of bidirectional links using CC model (called *cooperative bridges*) which can connect different disconnected components in the communication graph with maximum transmission power. Then, they apply a 2-layer MST structure (one MST over the CC links to connect the components, the other is inside each component) to further reduce the energy consumption.

To the best of our knowledge, [1] and [2] are the only papers to address the topology control problem under CC model. However, both work do not consider energy efficiency of the constructed topology. Even though the proposed solutions in [1], [2] can guarantee the network connectivity and reduce the energy consumption of the topology, they may hurt the energy efficiency of paths among nodes. As shown by the example in Fig. 1, the least energy path between two nodes could be arbitrarily large compared with the optimal one. This is unacceptable for many energy-critical applications of cooperative ad hoc networks. Therefore, in this paper, we study the energy efficient topology control problem with CC model.

3 Model and Problem Formulation

In this section, we first describe the cooperative communication model and corresponding network model used by our topology control problem and algorithms. Then, we formally define the *cooperative energy spanners* and introduce the *energy-efficient topology control problem with cooperative communication*.

3.1 Cooperative Communication Model

Our cooperative communication model is similar to those of [1], [2], [21], [22]. Every node v_i can adjust its transmission power P_i which is limited by a maximum value P_{MAX}. In the traditional communication model, without cooperative communication, a sending node v_i can successfully communicate with a receiving node v_j directly, only when the transmission power of v_i satisfy

$$P_i \cdot (d_{ij})^{-\alpha} \geq \tau \qquad (P_i \leq P_{MAX}). \tag{1}$$

Here, α is the path loss exponent (usually between 2 and 4), τ is the minimum average signal-to-noise ratio for decoding received data, and d_{ij} is the distance between node v_i and node v_j. In this paper, we focused on the interference-limited regime, where the noise is small compared to signals.

Cooperative communication model takes advantage of the physical layer design [21] that combines partial signals to obtain the complete information. Thus, a complete communication from node v_i to node v_j can be achieved

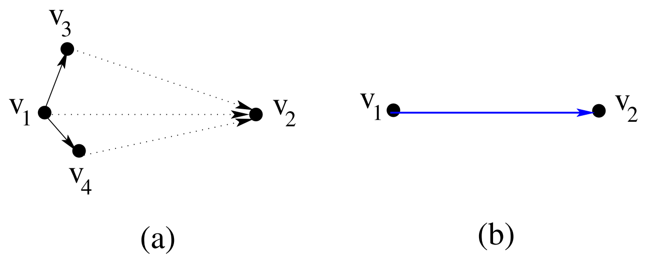


Fig. 2. Cooperative Communication: (a) v_1 together with its helpers v_3 and v_4 transmit to v_2 using cooperative communication; (b) a CC link $\widetilde{v_1v_2}$ represents such cooperative communication.

with cooperative communication if v_i transits the signal with a set of helper nodes H_{ij} and the summation of their transmission power satisfies

$$\sum_{v_k \in v_i \cup H_{ij}} P_k \cdot (d_{kj})^{-\alpha} \geq \tau \qquad (P_k \leq P_{MAX}). \qquad (2)$$

Fig. 2a shows such an example. Node v_1 wants to transmit data to node v_2. Using cooperative communication, it can first transmit to its nearby neighbors v_3 and v_4, then the three nodes together transmit to v_2. If the combined SNR is larger than τ, v_2 can successfully decode the data. Notice that, when node v_1 transmits data to its helpers v_3 and v_4 in the first phase, v_2 also receives a partial data from v_1. Thus, in some CC models, such partial data are used by v_2 to jointly decode the original data during the second phase of CC transmission. In this paper, we ignore such partial data during the first phase for simplicity (as [2] did). However, the proposed algorithms can be easily adopted to the model where the partial data are considered during decoding. Physical layer techniques for implementation of CC could be found in [23]. Carefully selecting the helper set and using cooperative communication can reduce the transmission power and extend the transmission coverage. For example, if v_1 cannot directly reach v_2 (i.e., $P_{MAX} \cdot (d_{12})^{-\alpha} < \tau$), its transmission coverage can be extended with v_3 and v_4's help. On the other hand, even v_1 can directly reach v_2, if the transmission power it needs is larger than the sum of transmission power consumes when using CC, i.e., $P_1 + P_3 + P_4 < \tau \cdot (d_{12})^{\alpha}$, CC transmission can still save energy from the direct transmission.

3.2 Network Model

We consider a wireless ad hoc network with n nodes which are capable of receiving and combining partial received packets in accordance with the CC model. The network topology is modeled as a 2D directed graph: $G = (V, E)$, where $V = (v_1, \ldots, v_n)$ denotes the set of wireless nodes and E denotes a set of directed communication links. A directed link $v_iv_j \in E$ denotes that node v_i can transmit data to node v_j either directly or using CC. $N(v_i)$ is the set of *direct* neighbor nodes of v_i within its maximum transmission range R_{max}, i.e., for all $v_k \in N(v_i)$, there exists $P_i \leq P_{MAX}$ such that $P_i \cdot (d_{ik})^{-\alpha} \geq \tau$. In other words, node v_i is able to communicate with its neighbor v_k directly. We assume that each node has a unique ID and knows its own location information. Node ID and location information are exchanged among all nodes. Then, we can define several important concepts.

Definition 1 (Direct Link). *A direct link $\overline{v_iv_j}$ is a link in E representing that node v_i can transit the information to node v_j directly (i.e., without CC). We use a solid black line to denote a direct link.*

Definition 2 (Helper Node Set). *H_{ij} symbolizes the* helper node set *including all helper nodes of node v_i for its CC transmission to v_j. In this paper, we assume all helper nodes of v_i must be a direct neighbor of v_i, i.e., $H_{ij} \subset N(v_i)$. In other words, all elements in $N(v_i)$ are the candidates of helper nodes of v_i.*

Definition 3 (Helper Link). *A* helper link *is a direct link $\overline{v_iv_k}$ between a source node v_i and its helper node v_k. For example, the link $\overline{v_1v_3}$ is a helper link in Fig. 2a.*

Definition 4 (Cooperative Communication Link (CC-link)). *A CC-link $\widetilde{v_iv_j}$ is a link in E representing node v_i can transit the information to node v_j cooperatively with a set of helper nodes H_{ij}. We use a solid blue line to denote a CC-link. As shown in Fig. 2b, link $\widetilde{v_1v_2}$ is a CC-link.*

The unions of all direct links and CC-links are $\overline{E}$ and $\widetilde{E}$, respectively. Similarly, we define the direct communication graph and CC communication graph as $\overline{G} = (V, \overline{E})$ and $\widetilde{G} = (V, \widetilde{E})$, respectively. Notice that $E = \overline{E} + \widetilde{E}$ and $G = \overline{G} + \widetilde{G}$.

In traditional topology control problem, the direct communication graph $\overline{G}$ is assumed to be *strongly connected* (i.e., any two nodes $v_i, v_j \in V$ are mutually reachable by direct links). Similarly, we can assume the strongly connectivity of the network G under CC model for our new cooperative topology control problem.

Definition 5 (Strongly Connected under CC Model). *The cooperative network G is* strongly connected under CC model *if and only if for any two nodes v_i and v_j there exists a directed path to connect them in G. In other words, by using the combination of direct links and CC-links the packet from any source node can reach any other node in the network.*

3.3 Cooperative Energy Spanners

In [1], given a strongly connected direct communication graph $\overline{G}$, the authors studied how to use a few CC-links replacing some direct links such that the resulting graph G' is strongly connected under CC model and the total energy consumption is minimized. In [2], given a strongly connected communication graph G under CC model (the corresponding $\overline{G}$ may be disconnected), the authors studied how to build a sparse subgraph G' such that G' is strongly connected under CC model and the total energy consumption is minimized. Both existing works on topology control under CC model [1], [2] do not consider the energy efficiency of paths inside the generated topologies. Notice that the method in [1] can take the output of a traditional energy-efficient topology control (without using CC) as its input and construct an energy spanner of the direct communication graph $\overline{G}$, however, it may not be an energy spanner of the overall communication graph G. In the example of Fig. 1, if the link v_1v_n is a CC-link, the method in [1] will not add it in the final topology, thus, it will cause an arbitrarily large energy stretch factor between v_1 and v_n. To address the energy efficient problem, we formally define the energy consumption of each link and each path in cooperative ad hoc networks.

Definition 6 (Link Weight). *For each link $v_iv_j \in E$, we define the link weight $w(v_iv_j)$ represents the minimum total energy consumption(i.e., total transmission power) of all nodes*

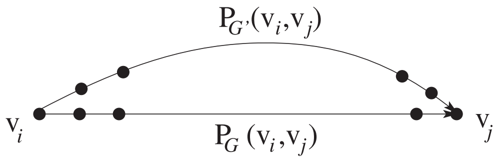


Fig. 3. Energy stretch factor: the energy stretch factor of a node pair (v_i, v_j) in G' with respect to G, $\rho_G^{G'}(v_i,v_j) = \frac{p(P_{G'}(v_i,v_j))}{p(P_G(v_i,v_j))}$, represents the difference between the costs of least-energy paths in G' and G.

participated in maintaining a directed link from v_i to v_j. For a direct link $\overline{v_iv_j}$, $w(\overline{v_iv_j}) = \tau d_{ij}^{\alpha}$. For a CC-link $\widetilde{v_iv_j}$, to select a set of helper nodes H_{ij} for v_i from its one-hop neighbors $N(v_i)$, such that the total energy consumption of the constructed CC-link is minimized, is a challenging problem itself. Therefore, we use an estimated total energy cost as follows, for $\widetilde{v_iv_j}$ with a selected helper node set H_{ij}

$$w(\widetilde{v_iv_j}) = \frac{\tau}{(\max_{v_k \in H_{ij}} d_{ik})^{-\alpha}} + \frac{(|H_{ij}|+1)\cdot\tau}{\sum_{v_k \in v_i \cup H_{ij}} (d_{kj})^{-\alpha}}. \quad (3)$$

Here, $|H_{ij}|$ denotes the number of elements in set H_{ij}. The link weight of a CC-link includes two parts: the first one is the energy consumption for node v_i to communicate with its helper node set H_{ij} directly, and the second part is the total energy consumption for v_i and its helpers in H_{ij} to cooperatively communicate with v_j. Notice that here we simplify the CC-model by assuming the transmission powers of node v_i and its helper node set H_{ij} during CC transmission are the same. In addition, we only consider the transmission power at each sender here. However, it is easy to extend our model to take the receiver's power into consideration. If the receiving power at each receiver is denoted by P_r, we can simply add P_r and $(|H_{ij}|+1)P_r$ to $w(\overline{v_iv_j})$ and $w(\widetilde{v_iv_j})$, respectively. This will not affect our proposed algorithms and analysis.

Consider a unicast path $\pi(v_i,v_j)$ in a cooperative network G from node v_i to node v_j under CC model, the total transmission energy consumed by this path $\pi(v_i,v_j)$ is $p(\pi(v_i,v_j)) = \sum_{e\in\pi(v_i,v_j)} w(e)$. Here, e can be either a direct link or a CC-link. Let $P_G(v_i,v_j)$ be the path consuming the least energy among all paths connecting v_i and v_j in G. We call $P_G(v_i,v_j)$ the least-energy path in G for v_i and v_j. Let $p(P_G(v_i,v_j))$ be the total energy of the least-energy path. Then, we can define the stretch factor as follows:

Definition 7 (Energy Stretch Factor). *Let G' be a subgraph*[1] *of G. The energy stretch factor of a node pair (v_i,v_j) in G' under CC model with respect to G is defined as $\rho_G^{G'}(v_i,v_j) = \frac{p(P_{G'}(v_i,v_j))}{p(P_G(v_i,v_j))}$. See* Fig. 3 *for illustration. The* energy stretch factor *of G' under CC model with respect to G is the defined as*

$$\rho_G(G') = \max_{v_i,v_j\in V} \rho_G^{G'}(v_i,v_j) = \max_{v_i,v_j\in V} \frac{p(P_{G'}(v_i,v_j))}{p(P_G(v_i,v_j))}. \quad (4)$$

Definition 8 (Cooperative Energy t-Spanner). *A subgraph G' of the cooperative network G is a* cooperative energy t-spanner *of G if its stretch factor under CC model with respect to G is no larger than a constant t, i.e., $\rho_G(G') \le t$.*

1. Here, a subgraph G' has the same vertex set V with the graph G but with less links than G.

If a topology is a cooperative energy spanner of the original communication graph G, then we guarantee there is a path between each pair of nodes whose energy consumption is similar to the original optimal one when all possible direct and CC links are used. This will benefit energy efficient routing performance on the network topology.

3.4 Problem Formulation

Now we can define the new topology control problem we will study in this paper.

Energy-Efficient Topology Control Problem with CC. Given a wireless multihop network $G = (V, E)$ which is strongly connected under CC model, assign transmission power P_i to every node v_i such that 1) the induced topology G' from this power assignment is a cooperative energy t-spanner of G, i.e., $\rho_G(G') \le t$; and 2) the sum of transmission power of all nodes, $\sum_{v_i\in V} P_i$, is minimized.

Notice that the spanner property also guarantees that the induced topology G' is strongly connected under CC model. The above topology control problem is equivalent to the following one: given the strongly connected G, construct a substructure G' spanning all nodes such that 1) G' is a cooperative energy t-spanner of G; and 2) the induced power assignment from G' minimizes the total transmission energy $\sum_{v_i\in V} P_i$. Here, the induced transmission power of node v_i from G' is the maximum power to support all of its outgoing links in G'.

Hardness of ETCC. In [1], [2], the topology control problem under cooperative communication model (TCC), which aims to minimize the total energy while maintaining the connectivity, has been proved to be NP-complete. Now we also prove that ETCC is also NP-complete. Given the power assignment for each node in the network, it is easily to verify in polynomial time whether the energy stretch factor of the induced topology G' is less than or equal to t and whether the total cost of this assignment is less than a fixed value. Thus, ETCC belongs to the NP-class. In addition, TCC problem is a special case of ETCC, where the stretch factor constrain t is set to an arbitrarily large constant. In such case, the spanner property reduces to the basic connectivity requirement. As TCC is NP-complete, ETCC is also NP-complete. Therefore, in this paper, we will focus on heuristic algorithms to guarantee the spanner property while minimizing the transmission energy.

Table 1 lists all the symbols used in the paper.

4 Energy-Efficient Topology Control with Cooperative Communication

In this section, we propose two topology control algorithms which build energy-efficient cooperative energy spanners. To keep the proposed algorithms simple and efficient, we only consider its one-hop neighbors as possible helper nodes for each node when CC is used. Thus, the original cooperative communication graph G contains all direct links and CC-links with one hop helpers, instead of all possible direct links and CC-links. In addition, for each pair of nodes v_i and v_j, we only maintain one link with least weight if there are multiple links connecting them. Here, all links are directional links. Both proposed algorithms are greedy algorithms. The major difference between them is the processing order of links. The first algorithm deletes links from the original graph G greedily, while the second algorithm adds links into G''

TABLE 1
Summary of Notations

Symbol	Definition
P_{MAX}	maximum transmission power
P_i	transmission power of node v_i
α	path loss exponent
τ	minimum average SNR
d_{ij}	distance between node v_i and node v_j
$N(v_i)$	one-hop neighbor nodes of node v_i
$\overline{v_iv_j}$	direct link from node v_i to node v_j
$\widetilde{v_iv_j}$	CC-link from node v_i to node v_j
H_{ij}	helper nodes set of node v_i for $\widetilde{v_iv_j}$
$\pi(v_i, v_j)$	path from node v_i to node v_j
$w(v_iv_j)$	link weight of link v_iv_j
G	orignal underlying communication graph including both direct links and CC-links
G'	constructed topology by topology control algorithm, which is a subgraph of G
$P_G(v_i, v_j)$	least energy path connecting v_i and v_j in G
$\rho_{G'}(G)$	stretch factor of G' under CC model with respect to the original cooperative graph G
$\rho_G^{G'}(v_i, v_j)$	stretch factor of a node pair (v_i, v_j) in G' under CC model with respect to G
$p(\pi)$	energy consumption of path π
t	energy stretch factor requirement
n, m	number of nodes/links in G
$P_i^d(j)$	energy consumption at v_i for a direct link $\overline{v_iv_j}$
$P_i^{cc}(j)$	energy consumption at v_i for a CC-link $\widetilde{v_iv_j}$
$g_G^{G'}(v_iv_j)$	stretch-factor-gain of a link v_iv_j by adding it to G'
$N_k(v_i)$	k-hop local neighborhood around v_i in G
$G_k(v_i)$	local graph around v_i with all links among $N_k(v_i)$ in G
$G'_k(v_i)$	local energy spanner generated by our localized algorithm which is a subgraph of $G_k(v_i)$

greedily. Here, G'' is a basic connected subgraph of G. Both algorithms can guarantee the cooperative energy spanner property of the constructed graph G'.

4.1 Greedy Algorithm 1—Deleting Links

We first propose a simple greedy algorithm for energy-efficient topology control, which is inspired by the classical greedy algorithm [41], [42] for low-weight spanner. The general idea is to start with the original cooperative communication graph G as described above. Then, gradually delete the least energy-efficient link (the link with highest weight) from G if doing so does not break the energy stretch factor requirement. Finally, the transmission power of each node is decided from the constructed topology G'. The details of these three steps are as follows:

Step 1: Construction of G. Initially, G is an empty graph. First, add every direct links $\overline{v_iv_j}$ into G, if node v_i can reach node v_j when it operates with P_{MAX}. Then, for every pair of nodes v_i and v_j, we select a set of helper nodes H_{ij} for node v_i from its one-hop neighbors $N(v_i)$, such that the link weigh $w(\widetilde{v_iv_j})$ of the constructed CC-link is minimized. Notice that this helper node decision problem is challenging even under our assumption that the transmission powers of v_i and its helper node set to maintain CC-link are the same. If we try all combinations of the helper sets to find the optimal helper set which minimizes the total energy consumption of v_i and its helpers, the computational complexity is exponential to the size of the one-hop neighborhood $N(v_i)$. It is impractical to do so in case of a large number of neighbors. Therefore, we directly use the greedy heuristic algorithm1 in [2], Greedy Helper Set Selection$(v_i, N(v_i), v_j)$, to select the helper set H_{ij}. For details, please refer to [2]. Then, we compare $w(\widetilde{v_iv_j})$ with $p(P_G(v_i, v_j))$ which is the current shortest path from node v_i to node v_j in G. If $w(\widetilde{v_iv_j}) < p(P_G(v_i, v_j))$ and

$$\frac{\tau}{\sum_{v_k \in v_i \cup H_{ij}} (d_{kj})^{-\alpha}} \leq P_{MAX},$$

add this CC-link $\widetilde{v_iv_j}$ into G. If there already exists a direct link $\overline{v_iv_j}$, delete it after the new CC-link $\widetilde{v_iv_j}$ is added (since it costs more energy than the CC-link). Notice that if

$$\frac{\tau}{\sum_{v_k \in v_i \cup H_{ij}} (d_{kj})^{-\alpha}} \geq P_{MAX},$$

node v_i cannot communicate with node v_j within one-hop even in CC model.

Step 2: Construction of G'. Copy all links in G to G', and sort them in the descending order of their weights. Start to process all links one by one and delete the link v_iv_j from G' if $G' - v_iv_j$ is still a cooperative energy t-spanner of G. Hereafter, we use $G - e$ or $G + e$ to denote the graph generated by removing link e from G or adding link e into G, respectively. In addition, when a CC-link $\widetilde{v_iv_j}$ is kept in G', all its helper links must be kept in G' too.

Step 3: Power Assignment from G'. For each node v_i, its transmission power is decided by the following equation:

$$P_i = \max\left\{ \max_{\overline{v_iv_j} \in G'} P_i^d(j), \max_{\widetilde{v_iv_j} \in G'} P_i^{cc}(j) \right\}. \quad (5)$$

Here, $P_i^d(j) = \frac{\tau}{d_{ij}^{-\alpha}}$ and

$$P_i^{cc}(j) = \frac{\tau}{\sum_{v_k \in v_i \cup H_{ij}} (d_{kj})^{-\alpha}}$$

are the energy consumption at v_i for a direct link $\overline{v_iv_j}$ and a CC-link $\widetilde{v_iv_j}$, respectively.

Algorithm 1 gives the detailed topology control algorithm. The guarantee of the energy t-spanner property is straightforward, since links are deleted from G' only when the deletion does not break the t-spanner requirement. The time complexity of Algorithm 1 is $O(n^2\Delta\log\Delta + mn(n\log n + m)) = O(mn(n\log n + m))$, where n, m, and Δ are the number of nodes, the number of links (both direct links and CC-links), and the maximum node degree of the original cooperative communication graph G, respectively. Detailed time complexity of each part of this algorithm is as follow: Lines 2-6 cost $O(n^2)$; Lines 7-15 cost $O(n^2\Delta\log\Delta)$; Line 17 needs perform ordering with cost of $O(m\log m)$; and Lines 18-23 cost $O(mn(n\log n + m))$ for testing the stretch factor requirement.

Algorithm 1. Greedy Algorithm 1 - Deleting Links
Input: A set of n wireless nodes V and nodes' locations; the maximum transmission power P_{MAX}; and the energy stretch factor requirement t.
Output: A topology G' and its induced power assignment P_i for each node v_i.

1: $G = (V, \phi)$;
2: **for all** v_i and $v_j \in V$ **do**
3: **if** $P_{MAX}(d_{ij})^{-\alpha} \geq \tau$ **then**
4: Add direct link $\overline{v_iv_j}$ into G;
5: **end if**
6: **end for**

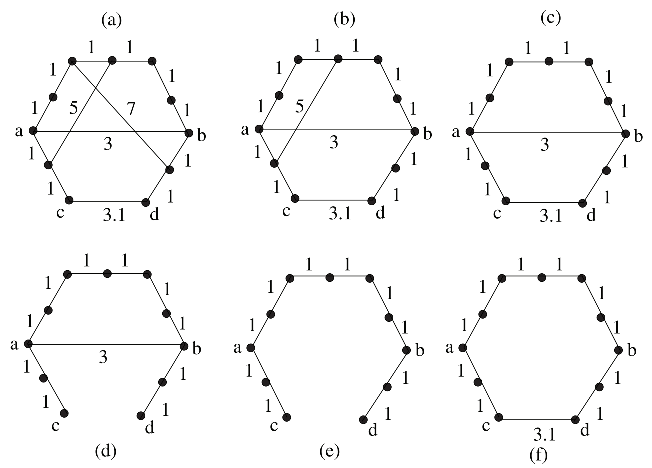


Fig. 4. Examples of two greedy algorithms: (a) communication graph G; (b)-(d) deleting links from G using Algorithm 1; (e)-(f) adding links from the connected sparse topology G'' using Algorithm 2. Link weights are labeled on links, and the stretch factor requirement $t = 3$.

7: **for all** v_i and $v_j \in V$ **do**
8: $H_{ij} =$ Greedy Helper Set Selection $(v_i, N(v_i), v_j)$;
9: **if** $w(\widetilde{v_iv_j}) < p(P_G(v_i, v_j))$ and $\frac{\tau}{\sum_{v_k \in v_i \cup H_{ij}} (d_{kj})^{-\alpha}} \leq P_{MAX}$ **then**
10: Add CC-link $\widetilde{v_iv_j}$ into G;
11: **if** direct link $\overline{v_iv_j} \in G$ **then**
12: Delete $\overline{v_iv_j}$ from G;
13: **end if**
14: **end if**
15: **end for**
16: Copy G to G';
17: Sort all links $v_iv_j \in G'$ in the descending order of their weights and mark them deletable;
18: **for all** link $v_iv_j \in G'$ processed in the sorted order **do**
19: **if** $\rho_G(G' - v_iv_j) \leq t$ and v_iv_j is deletable **then**
20: Delete link v_iv_j from G';
21: **else if** link v_iv_j is a CC-link **then**
22: Make all helper links v_iv_k for $v_k \in H_{ij}$ undeletable;
23: **end if**
24: **end for**
25: **for all** $v_i \in V$ **do**
26: $P_i = \max\{\max_{\overline{v_iv_j} \in G'} P_i^d(j), \max_{\widetilde{v_iv_j} \in G'} P_i^{cc}(j)\}$;
27: **end for**

Figs. 4a, 4b, 4c, and 4d show the process of Algorithm 1 on a simple CC network. Fig. 4a gives all possible links (both direct links and CC links) in the network G. Algorithm 1 tries to remove links with high costs without hurting the stretch factor beyond the requirement. In this example, assume that the stretch factor requirement $t = 3$. Removing two links with cost 5 and 7 (shown in Figs. 4b and 4c) does not affect the stretch factor. Removing link cd (shown in Fig. 4d) raises the stretch factor to $\frac{7}{3.1} \approx 2.26$ which still satisfies the requirement. However, further removing link ab will lead to too large stretch factor ($\frac{10}{3.1} \approx 3.23$). Therefore, the output topology of Algorithm 1 is Fig. 4d.

4.2 Greedy Algorithm 2—Adding Links

The second topology control algorithm starts with a sparse topology G'' which is strongly connected under CC model. We can use the output of the algorithm in [2] as the initial topology. Then, we gradually add the most energy-efficient link into G''. Here, the energy-efficiency of a link is defined as the gain on reducing energy stretch factors by adding this link. Our algorithm will terminate until the constructed graph G' satisfies the energy stretch factor requirement. The detail steps are summarized as follows:

Step 1: Construction of G and G''. The step of constructing G is the same as the one in Algorithm 1. Then, we call the algorithm in [2] to generate G'', a connected sparse subgraph of G.

Step 2: Construction of G'. Initialize $G' = G''$, for every link $v_iv_j \in G$ but $\notin G'$, compute its stretch-factor-gain $g_G^{G'}(v_iv_j)$ as follows:

$$g_G^{G'}(v_iv_j) = \sum_{v_p, v_q \in V} \left(\rho_G^{G'}(v_p, v_q) - \rho_G^{G'+v_iv_j}(v_p, v_q)\right). \quad (6)$$

In other words, the total gain of a link v_iv_j is the summation of the improvement of stretch factors of every pair of nodes in G' after adding this link. In each step, we greedily add the link with the largest stretch-factor-gain into G'. If there is a tie, we use the link weight to break it by adding the link with the least weight. We repeat this procedure until G' meets the stretch factor requirement t.

Step 3: Power Assignment from G'. For every node v_i, assign its power level P_i using (5).

Algorithm 2 gives the detailed topology control algorithm. The guarantee of the energy t-spanner property is straightforward since the algorithm terminates adding links until the t-spanner requirement is satisfied. In the worst case, every links are added into G', then the t-spanner requirement must be satisfied. The time complexity of Algorithm 2 is also $O(mn(n \log n + m))$. Detailed time complexity of each part of this algorithm is as follows: Lines 2-6 cost $O(n^2)$; Lines 7-15 cost $O(n^2 \Delta \log \Delta)$; Line 16 costs $O(m + n \log n)$ for building the minimum spanning tree; and Lines 18-23 cost $O(mn(n \log n + m))$ for testing the stretch factor gain.

Algorithm 2. Greedy Algorithm 2 - Adding Links
Input: A set of n wireless nodes V and nodes' locations; the maximum transmission power P_{MAX}; and the energy stretch factor requirement t.
Output: A topology G' and its induced power assignment P_i for each node v_i.
1: $G = (V, \phi)$;
2: **for all** v_i and $v_j \in V$ **do**
3: **if** $P_{MAX}(d_{ij})^{-\alpha} \geq \tau$ **then**
4: Add direct link $\overline{v_iv_j}$ into G;
5: **end if**
6: **end for**
7: **for all** v_i and $v_j \in V$ **do**
8: $H_{ij} =$ Greedy Helper Set Selection $(v_i, N(v_i), v_j)$;
9: **if** $w(\widetilde{v_iv_j}) < p(P_G(v_i, v_j))$ and $\frac{\tau}{\sum_{v_k \in v_i \cup H_{ij}} (d_{kj})^{-\alpha}} \leq P_{MAX}$ **then**
10: Add CC-link $\widetilde{v_iv_j}$ into G;
11: **if** direct link $\overline{v_iv_j} \in G$ **then**
12: Delete $\overline{v_iv_j}$ from G;
13: **end if**
14: **end if**
15: **end for**

Call the algorithm in [2] to generate a connected sparse topology G'' of G;
Copy G'' to G';
while $\rho_G(G') > t$ **do**
  for all link $v_iv_j \in G$ but $\notin G'$ **do**
    $g_G^{G'}(v_iv_j) = \sum_{v_p,v_q \in V}(\rho_G^{G'}(v_p,v_q) - \rho_G^{G'+v_iv_j}(v_p,v_q))$;
  end for
  Add link v_iv_j with the largest stretch-factor-gain $g_G^{G'}(v_iv_j)$ into G'; If there is a tie, break it by adding the link with the least weight $w(v_iv_j)$; If link v_iv_j is a CC-link, also add all helper links v_iv_k for $v_k \in H_{ij}$ in G'.
end while
for all $v_i \in V$ **do**
  $P_i = \max\{\max_{\overline{v_iv_j} \in G'} P_i^d(j), \max_{\widetilde{v_iv_j} \in G'} P_i^{cc}(j)\}$;
end for

Consider the same network in Fig. 4a. Algorithm 2 first constructs a connected sparse topology G'' as shown in Fig. 4e and copy it to G'. Obliviously, G' cannot satisfies the stretch factor requirement. Thus, Algorithm 2 greedily chooses new links to add to decrease the stretch factor. Based on the stretch-factor-gain defined in (6), we can calculate the gain of adding link ab or cd as follows:

$$g_G^{G'}(a,b) = \left(\rho_G^{G'}(a,b) - \rho_G^{G'+ab}(a,b)\right) + \left(\rho_G^{G'}(c,d) - \rho_G^{G'+ab}(c,d)\right) = \left(\frac{6}{3} - 1\right) + \left(\frac{10}{3.1} - \frac{7}{3.1}\right) \approx 1.97$$

and

$$g_G^{G'}(c,d) = \left(\rho_G^{G'}(a,b) - \rho_G^{G'+cd}(a,b)\right) + \left(\rho_G^{G'}(c,d) - \rho_G^{G'+cd}(c,d)\right) = \left(\frac{6}{3} - \frac{6}{3}\right) + \left(\frac{10}{3.1} - 1\right) \approx 2.23.$$

Therefore, Algorithm 2 adds cd into the topology G' with a larger gain. Since this results stretch factor of 2 which satisfies the requirement, the algorithm exists with Fig. 4f as its output topology.

In Algorithm 2, we consider the summation of all improvements on stretch factors of every pair of nodes in G' as the stretch-factor-gain of a link. However, for certain pairs of nodes, if their stretch factors are already below t, further improvement for these pairs may not be very useful. Therefore, we can change the definition of stretch-factor-gain to

$$g_G^{G'}(v_iv_j) = \sum_{(v_p,v_q) \in B} \left(\rho_G^{G'}(v_p,v_q) - \rho_G^{G'+v_iv_j}(v_p,v_q)\right). \quad (7)$$

Here, B is the set of node pairs in G' whose stretch factor is still larger than t. By doing so, we hope that the algorithm can converge faster. However, this method may also lead to a situation where no link can provide further gain on stretch-factors, meanwhile the entire network's stretch factor is still larger than t. In this case, we can either use the original Algorithm 2 as the backup or directly add one least-energy path in G to G' whose stretch factor in G' is the largest. Similarly, another alternative way to compute the stretch-factor-gain is

$$g_G^{G'}(v_iv_j) = \rho_G(G') - \rho_G(G' + v_iv_j), \quad (8)$$

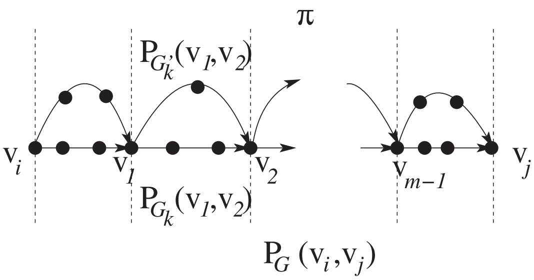


Fig. 5. Localized algorithms can still guarantee the energy stretch factor requirement t, i.e., for any v_i and v_j, there exists a path π connecting them in the constructed graph G' such that $p(\pi) \leq t \cdot p(P_G(v_i,v_j))$.

in which the gain is defined as the improvement of overall stretch factor.

5 Localized Energy-Efficient Topology Control with Local Information

The two proposed topology control algorithms (Algorithms 1 and 2) are both centralized algorithms, since they need the global information of the network. However, such solutions are not very practical for certain applications, since the ad hoc networks are usually self-organized and without centralized control. Thus, we are also interested in extending the proposed methods into distributed or localized algorithms. Here, a distributed topology control algorithm (to construct a graph G') is a localized algorithm if every node v_i can exactly decide all edges incident on v_i based only on the information of all nodes within a constant hops of v_i.

Fortunately, both of our cooperative topology control algorithms can be easily extended to localized algorithms with only local information. For each node $v_i \in V$, we define its k-hop local neighborhood $N_k(v_i)$, which includes all nodes within k-hop in graph G under CC model. Let $G_k(v_i)$ be the local graph around v_i which includes all links among $N_k(v_i)$ in G. Then, our localized topology control algorithm works as follows: First, each node v_i collects the local information from its k-hop local neighborhood (i.e., $G_k(v_i)$). Then, v_i applies either Algorithm 1 or Algorithm 2 on its local graph $G_k(v_i)$ and generates its local energy spanner $G'_k(v_i)$ which is a subgraph of $G_k(v_i)$ and has its stretch factor bounded by t with respect to $G_k(v_i)$. Node v_i notices its k-hop neighbors all edges in $G'_k(v_i)$. The final topology G' is the union of all $G'_k(v_i)$ for all nodes, i.e., $G' = \bigcap_{i=1}^{n} G'_k(v_i)$. We can prove that the final constructed graph G' is still a t-spanner of G.

Theorem 1. *The constructed graph G' by our localized topology control algorithm is a cooperative energy t-spanner of the original communication graph G.*

Proof. To prove the theorem, we only need to prove that for any two nodes $v_i, v_j \in V$ there is a path π in G' such that $p(\pi) \leq t \cdot p(P_G(v_i,v_j))$. As shown in Fig. 5, we now consider the least energy path $P_G(v_i,v_j)$ in G and partition it into serval segments $\pi(v_0 = v_i, v_1), \pi(v_1, v_2) \ldots, \pi(v_{m-2}, v_{m-1}), \pi(v_{m-1}, v_m = v_j)$, where each segment is less than or equal to k-hop. Notice that each segment $\pi(v_q, v_{q+1})$ is basically $P_G(v_q, v_{q+1})$, since $P_G(v_i,v_j)$ is the least energy path. Because $P_G(v_q, v_{q+1})$ is less than or equal to k-hop, v_{q+1} and this segment are inside the local graph $G_k(v_q)$ of v_q. Based on our localized algorithm, the least cost path

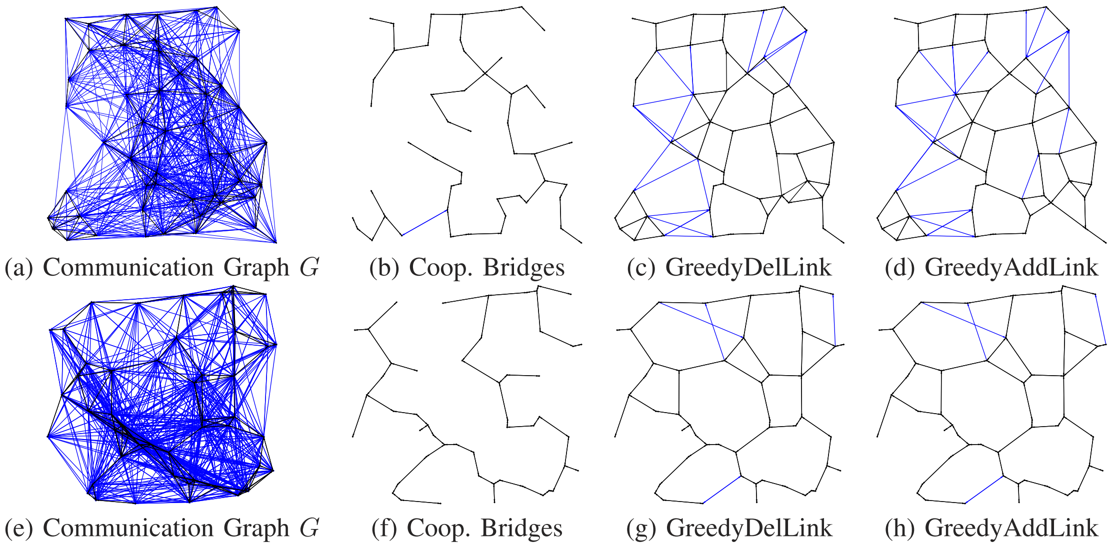


Fig. 6. Topologies generated by different topology control algorithms over the same random network: black/blue links are direct/cooperative links, respectively. Upper row (a-d): $t = 1.2$; Lower row (e-h): $t = 1.4$.

$P_{G'_k(v_q)}(v_q, v_{q+1})$ between v_q and v_{q+1} in $G'_k(v_q)$ costs at most t time of energy than the least cost path $P_{G_k(v_q)}(v_q, v_{q+1})$ (i.e., $P_G(v_q, v_{q+1})$) does. In other words, there is a path $P_{G'_k(v_q)}(v_q, v_{q+1})$ in the constructed local graph $G'_k(v_q)$ such that $p(P_{G'_k(v_q)}(v_q, v_{q+1})) \leq p(P_G(v_q, v_{q+1}))$. Notice that $P_{G'_k(v_q)}(v_q, v_{q+1})$ is also in G' since G' takes the union of all local graphs. Now consider the path π which connects all paths $p(P_{G'_k(v_q)}(v_q, v_{q+1}))$ for $q = 0$ to $m-1$ in G' (see Fig. 5 for illustration), we have

$$p(\pi) = \sum_{p=0}^{m-1} p(P_{G'_k(v_q)}(v_q, v_{q+1}))$$
$$\leq \sum_{p=0}^{m-1} t \cdot p(P_G(v_q, v_{q+1})) = t \cdot P_G(v_i, v_j).$$

This finishes the proof. □

In summary, our localized topology control algorithms only need local information to construct the cooperative energy t-spanner, even though the spanner property is a *global* requirement. This makes them applicable even for large-scale ad hoc networks. However, compared with the centralized versions, they may keep more communication links in G' due to the lack of global information of the network, thus may lead to larger energy cost in G'. Notice that with the increasing of k, the constructed G' of our localized algorithm will converge toward the output of the centralized algorithm. Therefore, there is a clear tradeoff between the communication cost of collecting information during construction and the transmission energy consumption of the constructed topology. We will see such observations in our simulation results.

6 Simulations

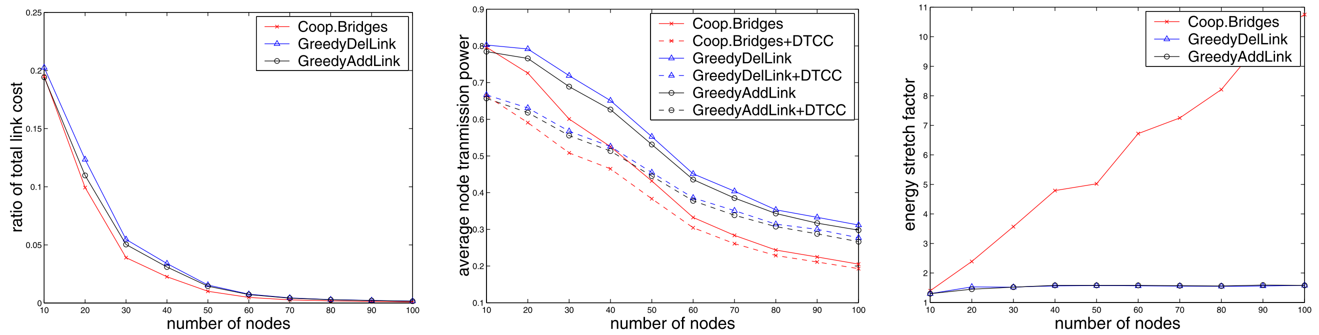
In this section, we will evaluate our proposed topology control algorithms, namely, *Greedy Algorithm 1* (GreedyDelLink) and *Greedy Algorithm 2* (GreedyAddLink), by comparing their performances with *Cooperative Bridges-based Method* (Coop. Bridges) [2]. We implement all these three algorithms in a simulator developed by our group. The underlying wireless networks are randomly generated in an area of 100×100. For convenience, we set the path loss factor $\alpha = 2$ and the SNR threshold $\tau = 1$. The value of P_{MAX} is set to 400 so that the maximum transmission range of a direct link is 20. Then, we perform proposed topology control algorithms on G. Fig. 6 shows two sets of topologies constructed by all three topology control algorithms for two different networks, respectively, with energy stretch factor requirement $t = 1.2$ and $t = 1.4$. In Fig. 6 the number of nodes $n = 50$. Clearly, our methods keep more links than the Coop. Bridges does, since our topologies guarantee the energy stretch factor bound while Coop Bridges doesn't. In addition, the larger t is, the sparser our topologies are.

In the following simulations, we take three metrics as the performance measurement of topology control algorithms:

- **Total Link Cost:** the total link cost of the constructed topology G', i.e., $w(G') = \sum_{e \in G'} w(e)$.
- **Average Node Transmission Power:** the average node transmission power of the constructed topology G', i.e., $\overline{P_i}(G') = \frac{1}{n}\sum_{v_i \in V} P_i$ where P_i is the transmission power of node v_i assigned from G' by (5).
- **Maximum Energy Strength Factor:** the maximum energy strength factor of graph G' under CC model with respect to G, i.e., $\rho_G(G') = \max_{v_i, v_j \in V} \frac{p(P_{G'}(v_i, v_j))}{p(P_G(v_i, v_j))}$.

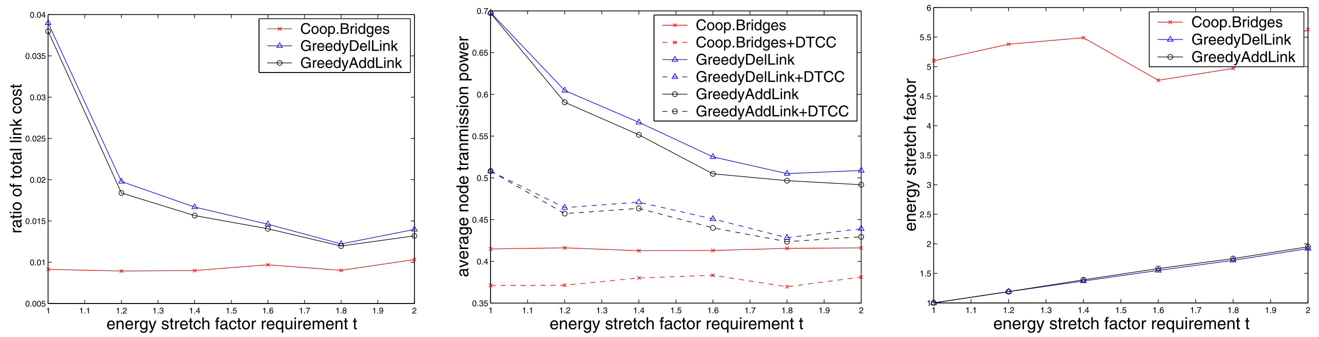
For all the simulations, we repeat the experiment for multiple times and report the average values of these metrics. It is clear that a desired topology should have small total link cost $w(G')$ and small node transmission power $\overline{P_i}(G')$ while the energy strength factor $\rho_G(G')$ should be smaller than the requirement t.

For the first set of simulations, we increase the network density by rising the number of nodes n from 10 to 100, and keep energy stretch factor requirement t at 1.6. Fig. 7a shows the ratios between the total link cost of the generated graph G' and that of the original cooperative communication graph G when n increases. This ratio implies how much cost saving achieved by the topology control algorithm, compared with the original network without topology control. Fig. 7b shows the ratios between average node transmission power



(a) $w(G')/w(G)$ (b) $\overline{P_i}(G')/\overline{P_i}(G)$ (c) $\rho_G(G')$

Fig. 7. Simulation results on random networks with fixed stretch factor requirement $t = 1.6$ and varying number of nodes n.



(a) $w(G')/w(G)$ (b) $\overline{P_i}(G')/\overline{P_i}(G)$ (c) $\rho_G(G')$

Fig. 8. Simulation results on random networks with fixed number of nodes $n = 50$ and varying stretch factor requirement t.

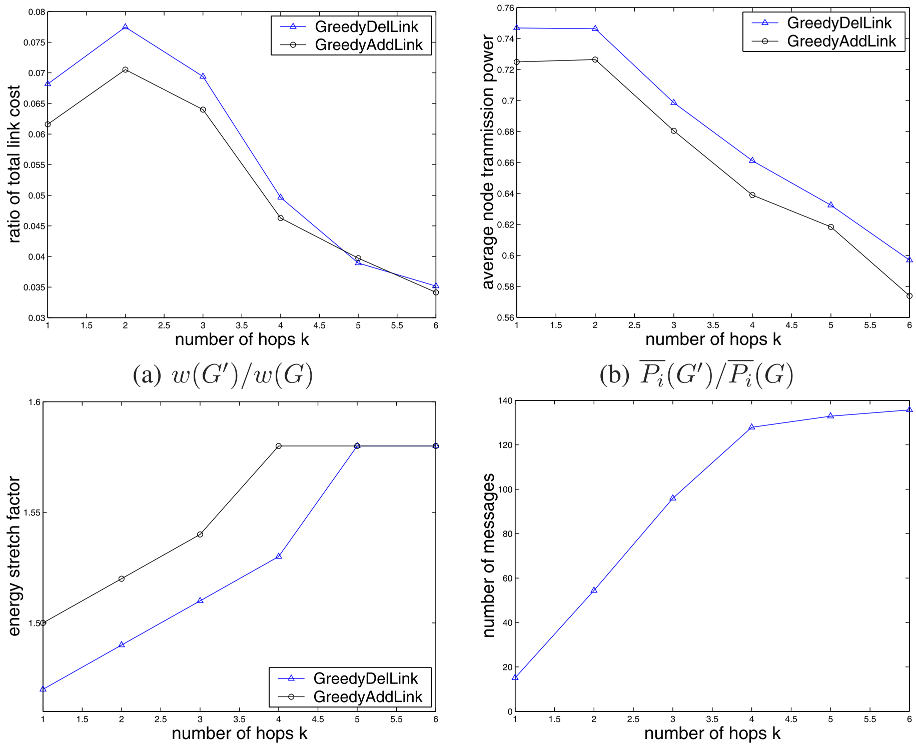
assigned by the topology control algorithm and that assigned from G. Here, for each topology control algorithm, we also apply the DTCC algorithm in [1] to further reduce the transmission power with CC. From these results, all topology control algorithms can significantly reduce the total cost of maintaining the connectivity and save the node transmission energy. When the network is denser, the saving of topology control is larger. Compared with Coop Bridges, our two methods generate a denser topology. However, their costs and transmission power are just slightly larger than that of Coop Bridges. DTCC algorithm can indeed further reduce the transmission power. Fig. 7c shows the energy stretch factor $\rho_G(G')$ of all methods. Clearly, Coop Bridges cannot satisfy the energy stretch factor requirement $t = 1.6$, and its stretch factor increases with the number of nodes. When the network is dense, it could be more than 10, which means some paths could have huge energy consumption! From observing above, we can conclude that both of our proposed methods can achieve similar small energy stretch factors, all of them are bounded by the requirement t, with the costs of very slightly larger total link costs and transmission power. This demonstrates the energy efficiency of our energy cooperative spanners.

In the second set of simulations, we fix the number of nodes $n = 50$ and run our algorithms with different energy stretch factor requirement t increasing from 1.0 to 2.0. From results shown in Figs. 8a and 8b, we can observe that tighter (i.e., smaller) stretch factor requirement results in higher cost and larger node transmission power of all topology algorithms. The reason is when the stretch factor requirement becomes larger, the restriction on removing links becomes weaker. In the extreme case with infinite t, our energy-efficient topology control problem converges to TCC which only preserves the connectivity. Clearly, there is a tradeoff between stretch factor and link cost, transmission power. We can choose a suitable stretch factor bound to match the particular application needs. From the results in Fig. 8c, we can see that the stretch factor of Coop Bridges does not affect by the stretch factor requirement, however both of our methods can perfectly satisfy the stretch factor requirement. Again it proves our methods can efficiently satisfy the stretch factor requirement while Coop Bridges cannot.

In the last set of simulations, we implement our localized algorithms with k-hop local information. We fix the number of nodes $n = 100$, the maximum power $P_{MAX} = 144$, and the energy stretch factor requirement $t = 1.6$. Fig. 9 shows the results when $k = 1$ to 6. It is clear that with larger neighborhood information (larger value of k) our algorithms can achieve better energy conservation (i.e., with smaller total cost and lower average node power) but with larger stretch factor. However, all of our localized algorithms can still satisfy the stretch factor requirement. As we expect, larger value of k also leads to higher number of messages need to collect the neighborhood information (as shown in Fig. 9d). Therefore, there is a tradeoff between the communication cost of collecting information during construction and the transmission energy consumption of the constructed topology. For different applications, we may have different setting of k.

7 Conclusion

In this paper, we introduced a new topology control problem: *energy-efficient topology control problem with cooperative communication*, which aims to keep the energy-efficient paths in the constructed topology. We proved that this problem is very challenging (NP-complete), and proposed two new topology control algorithms using cooperative communications. Both algorithms can build a



(a) $w(G')/w(G)$ (b) $\overline{P_i}(G')/\overline{P_i}(G)$ (c) $\rho_G(G')$ (d) number of messages

Fig. 9. Simulation results of localized algorithms on random networks with $n = 100$, $t = 1.6$, and various k values.

cooperative energy spanner in which the energy efficiency of individual paths are guaranteed even with only local information. Simulation results confirm the nice performance of both proposed algorithms.

Possible future works include: 1) design more efficient algorithms with lower complexity to achieve cooperative energy spanner with CC model; 2) investigate how to adapt the constructed spanners to unexpected changes in the network such as link and node failures. Notice that even though the proposed methods mainly work for static cooperative ad hoc networks, they do have potential to handle changes in dynamic networks. Especially, for localized versions of the proposed algorithms, any node or link change will only affect the final topology within its k-hop neighborhood. However, with the existence of cooperative communication links, such neighborhood could be large.

Acknowledgments

This work is supported in part by the US National Science Foundation (NSF) under Grant No. CNS-0915331 and CNS-1050398.

References

[1] M. Cardei, J. Wu, and S. Yang, "Topology Control in Ad Hoc Wireless Networks Using Cooperative Communication," *IEEE Trans. Mobile Computing*, vol. 5, no. 6, pp. 711-724, June 2006.
[2] J. Yu, H. Roh, W. Lee, S. Pack, and D.-Z. Du, "Cooperative Bridges: Topology Control in Cooperative Wireless Ad Hoc Networks," *Proc. IEEE INFOCOM*, 2010.
[3] R. Rajaraman, "Topology Control and Routing in Ad Hoc Networks: A Survey," *SIGACT News*, vol. 33, pp. 60-73, 2002.
[4] X.-Y. Li, "Topology Control in Wireless Ad Hoc Networks," *Ad Hoc Networking*, S. Basagni, M. Conti, S. Giordano, and I. Stojmenovic, eds., IEEE Press, 2003.
[5] C.-C. Shen and Z. Huang, "Topology Control for Ad Hoc Networks: Present Solutions and Open Issues," *Handbook of Theoretical and Algorithmic Aspects of Sensor, Ad Hoc Wireless and Peer-to-Peer Networks*, J. Wu, ed., CRC Press, 2005.
[6] A.E. Clementi, G. Huiban, P. Penna, G. Rossi, and Y.C. Verhoeven, "Some Recent Theoretical Advances and Open Questions on Energy Consumption in Ad-Hoc Wireless Networks," *Proc. Workshop Approximation and Randomization Algorithms in Comm. Networks*, 2002.
[7] W.-T. Chen and N.-F. Huang, "The Strongly Connecting Problem on Multihop Packet Radio Networks," *IEEE Trans. Comm.*, vol. 37, no. 3, pp. 293-295, Mar. 1989.
[8] L.M. Kirousis, E. Kranakis, D. Krizanc, and A. Pelc, "Power Consumption in Packet Radio Networks," *Theoretical Computer Science*, vol. 243, nos. 1/2, pp. 289-305, 2000.
[9] A.E.F. Clementi, P. Penna, and R. Silvestri, "On the Power Assignment Problem in Radio Networks," *Proc. Electronic Colloquium on Computational Complexity (ECCC)*, 2000.
[10] D. Blough, M. Leoncini, G. Resta, and P. Santi, "On the Symmetric Range Assignment Problem in Wireless Ad Hoc Networks," *Proc. Second IFIP Int'l Conf. Theoretical Computer Science*, 2002.
[11] E. Althaus, G. Câlinescu, I. Mandoiu, S. Prasad, N. Tchervenski, and A. Zelikovsly, "Power Efficient Range Assignment in Ad-Hoc Wireless Networks," *Proc. IEEE Wireless Comm. and Networking (WCNC)*, 2003.
[12] R. Ramanathan and R. Hain, "Topology Control of Multihop Wireless Networks Using Transmit Power Adjustment," *Proc. IEEE INFOCOM*, 2000.
[13] M. Hajiaghayi, N. Immorlica, and V.S. Mirrokni, "Power Optimization in Fault-Tolerant Topology Control Algorithms for Wireless Multi-Hop Networks," *Proc. ACM Mobicom*, 2003.
[14] J. Cheriyan, S. Vempala, and A. Vetta, "Approximation Algorithms for Minimum-Cost K-Vertex Connected Subgraphs," *Proc. Ann. ACM Symp. Theory of Computing (STOC)*, 2002.
[15] P. Bose, P. Morin, I. Stojmenovic, and J. Urrutia, "Routing with Guaranteed Delivery in Ad Hoc Wireless Networks," *Proc. Int'l Workshop Discrete Algorithms and Methods for Mobile Computing and Comm.*, 1999.

[16] X.-Y. Li, Y. Wang, and W.Z. Song, "Applications of K-local MST for Topology Control and Broadcasting in Wireless Ad Hoc Networks," *IEEE Trans. Parallel and Distributed Systems,* vol. 15, no. 12, pp. 1057-1069, Dec. 2004.

[17] X.-Y. Li, P.-J. Wan, and Y. Wang, "Power Efficient and Sparse Spanner for Wireless Ad Hoc Networks," *Proc. 10th Int'l Conf. Computer Comm. and Networks (ICCCN),* 2001.

[18] R. Wattenhofer, L. Li, P. Bahl, and Y.-M. Wang, "Distributed Topology Control for Wireless Multihop Ad-Hoc Networks," *Proc. IEEE INFOCOM,* 2001.

[19] N. Li, J.C. Hou, and L. Sha, "Design and Analysis of a MST-Based Topology Control Algorithm," *Proc. IEEE INFOCOM,* 2003.

[20] Y. Wang and X.-Y. Li, "Localized Construction of Bounded Degree and Planar Spanner for Wireless Ad Hoc Networks," *Mobile Networks and Applications,* vol. 11, no. 2, pp. 161-175, 2006.

[21] N. Laneman, D. Tse, and G. Wornell, "Cooperative Diversity in Wireless Networks: Efficient Protocols and Outage Behavior," *IEEE Trans. Information Theory,* vol. 50, no. 12, pp. 3062-3080, Dec. 2004.

[22] A. Nosratinia, T.E. Hunter, and A. Hedayat, "Cooperative Communication in Wireless Networks," *IEEE Comm. Magazine,* vol. 42, no. 10, pp. 74-80, Oct. 2004.

[23] G. Jakllari, S.V. Krishnamurthy, M. Faloutsos, P.V. Krishnamurthy, and O. Ercetin, "A Framework for Distributed Spatio-Temporal Communications in Mobile Ad Hoc Networks," *Proc. IEEE Infocom,* 2006.

[24] A. Khandani, J. Abounadi, E. Modiano, and L. Zheng, "Cooperative Routing in Static Wireless Networks," *IEEE Trans. Comm.,* vol. 55, no. 11, pp. 2185-2192, Nov. 2007.

[25] J. Zhang and Q. Zhang, "Cooperative Routing in Multi-Source Multi-Destination Multi-Hop Wireless Networks," *Proc. IEEE INFOCOM,* 2008.

[26] A. Ibrahim, Z. Han, and K. Liu, "Distributed Energy-efficient Cooperative Routing in Wireless Networks," *IEEE Trans. Wireless Comm.,* vol. 7, no. 10, pp. 3930-3941, Oct. 2008.

[27] M. Agarwal, J. Cho, L. Gao, and J. Wu, "Energy Efficient Broadcast in Wireles Ad Hoc Networks with Hitch-hiking," *Proc. IEEE INFOCOM,* 2004.

[28] J. Wu, M. Cardei, F. Dai, and S. Yang, "Extended Dominating Set and Its Applications in Ad Hoc Networks Using Cooperative Communication," *IEEE Trans. Parallel and Distributed Systems,* vol. 17, no. 8, pp. 851-864, Aug. 2006.

[29] G. Jakllari, S. Krishnamurthy, M. Faloutsos, and P. Krishnamurthy, "On Broadcasting with Cooperative Diversity in Multi-Hop Wireless Networks," *IEEE J. Selected Area in Comm.,* vol. 25, no. 2, pp. 484-496, Feb. 2007.

[30] F. Hou, L.X. Cai, P.H. Ho, X. Shen, and J. Zhang, "A Cooperative Multicast Scheduling Scheme for Multimedia Services in IEEE 802.16 Networks," *IEEE Trans. Wireless Comm.,* vol. 8, no. 3, pp. 1508-1519, Mar. 2009.

[31] L. Wang, B. Liu, D. Goeckel, D. Towsley, and C. Westphal, "Connectivity in Cooperative Wireless Ad Hoc Networks," *Proc. ACM Mobihoc,* 2008.

[32] A.K. Sadek, Z. Han, and K.J.R. Liu, "Distributed Relay-Assignment Protocols for Coverage Expansion in Cooperative Wireless Networks," *IEEE Trans. Mobile Computing,* vol. 9, no. 4, pp. 505-515, Apr. 2010.

[33] Y. Shi, S. Sharma, and Y. Hou, "Optimal Relay Assignment for Cooperative Communications," *Proc. ACM Mobihoc,* 2008.

[34] Q. Zhang, J. Jia, and J. Zhang, "Cooperative Relay to Improve Diversity in Cognitive Radio Networks," *IEEE Comm. Magazine,* vol. 47, no. 2, pp. 111-117, Feb. 2009.

[35] B. Wang, Z. Han, and K.J.R. Liu, "Distributed Relay Selection and Power Control for Multiuser Cooperative Communication Networks Using Stackelberg Game," *IEEE Trans. Mobile Computing,* vol. 8, no. 7, pp. 975-990, July 2009.

[36] M. Veluppillai, L. Cai, J.W. Mark, and X. Shen, "Maximizing Cooperative Diversity Energy Gain for Wireless Networks," *IEEE Trans. Wireless Comm.,* vol. 6, no. 7, pp. 2530-2539, July 2007.

[37] Y. Wang and X.-Y. Li, "Minimum Power Assignment in Wireless Ad Hoc Networks with Spanner Property," *J. Combinatorial Optimization,* vol. 11, no. 1, pp. 99-112, 2006.

[38] H. Shpungin and M. Segal, "Near Optimal Multicriteria Spanner Constructions in Wireless Ad-Hoc Networks," *Proc. IEEE INFOCOM,* 2009.

[39] A.K. Abu-Affash, R. Aschner, P. Carmi, and M. Katz, "Minimum Power Energy Spanners in Wireless Ad Hoc Networks," *Proc. IEEE INFOCOM,* 2010.

[40] Y. Zhu, M. Huang, S. Chen, and Y. Wang, "Cooperative Energy Spanners: Energy-Efficient Topology Control in Cooperative Ad Hoc Networks," *Proc. IEEE INFOCOM '11,* 2011.

[41] I. Althofer, G. Das, D. Dobkin, D. Joseph, and J. Soares, "On Sparse Spanners of Weighted Graphs," *Discrete and Computational Geometry,* vol. 9, no. 1, pp. 81-100, 1993.

[42] B. Chandra, G. Das, G. Narasimhan, and J. Soares, "New Sparseness Results on Graph Spanners," *Int'l J. Computational Geometry and Applications,* vol. 5, pp. 125-144, 1995.

Ying Zhu received the BEng degree in automatic control and MEng degree in pattern recognition from the University of Electronic Science and Technology of China in 2002 and 2005, respectively. She is currently working toward the PhD degree in the University of North Carolina at Charlotte, majoring in computer science. Her current research focuses on wireless networks, ad hoc and sensor networks, delay tolerant networks, and algorithm design.

Minsu Huang received the BEng degree in mechanical engineering from Central South University in 2003 and the MS degree in computer science from Tsinghua University in 2006. He is currently working toward the PhD degree in the University of North Carolina at Charlotte, majoring in computer science. His current research focuses on wireless networks, ad hoc and sensor networks, delay tolerant networks, and algorithm design.

Siyuan Chen received the BS degree from Peking University, China in 2006. He is currently working toward the PhD degree in the University of North Carolina at Charlotte, majoring in computer science. His current research focuses on wireless networks, ad hoc and sensor networks, delay tolerant networks, and algorithm design.

Yu Wang received the BEng degree (1998), the MEng degree (2000) in computer science from Tsinghua University, China, and the PhD degree (2004) in computer science from Illinois Institute of Technology. He is an associate professor of Computer Science at the University of North Carolina at Charlotte. His research interest includes wireless networks, ad hoc and sensor networks, delay tolerant networks, mobile computing, and algorithm design. He has published more than 100 papers in peer-reviewed journals and conferences. He has served as program chair, publicity chair, and program committee member for several international conferences (such as IEEE INFOCOM, IEEE IPCCC, IEEE GLOBECOM, IEEE ICC, IEEE MASS, etc.). He is a recipient of Ralph E. Powe Junior Faculty Enhancement Awards from Oak Ridge Associated Universities in 2006 and a recipient of Outstanding Faculty Research Award from College of Computing and Informatics at UNC Charlotte in 2008. He is a senior member of the ACM, the IEEE, and the IEEE Communications Society.