Efficient Topology Control for Ad-hoc Wireless Networks with Non-uniform Transmission Ranges

Xiang-Yang Li, Wen-Zhan Song, Yu Wang
Department of Computer Science, Illinois Institute of Technology
Email: xli@cs.iit.edu, songwen@iit.edu, wangyu1@iit.edu

July 22, 2003

Abstract

Wireless network topology control has drawn considerable attention recently. Priori arts assumed that the wireless ad hoc networks are modeled by unit disk graphs (UDG), i.e., two mobile hosts can communicate as long as their Euclidean distance is no more than a threshold. However, practically, the networks are never so perfect as unit disk graphs: the transmission ranges may vary due to various reasons such as the device differences, the network control necessity, and the perturbation of the transmission ranges even the transmission ranges are set as the same originally. Thus, we assume that each mobile host has its own transmission range. The networks are modeled by mutual inclusion graphs (MG), where two nodes are connected iff they are within the transmission range of each other. Previously, no method is known for topology control when the networks are modeled as mutual inclusion graphs.

The paper proposes the first distributed mechanism to build a sparse power efficient network topology for ad hoc wireless networks with non-uniform transmission ranges. We first extend the Yao structure to build a spanner with a constant length and power stretch factor for mutual inclusion graph. We then propose two efficient localized algorithms to construct connected sparse network topologies. The first structure, called extended Yao-Yao, has node degree at most $O(\log \gamma)$, where $\gamma = \max_u \max_{uv \in MG} \frac{r_u}{r_v}$. The second structure, called extended Yao and Sink, has node degree bounded by $O(\log \gamma)$, and is a length and power spanner. The methods are based on a novel partition strategy of the space surrounded each mobile host. Both algorithms have communication cost $O(n)$ under a local broadcasting communication model, where each message has $O(\log n)$ bits.

Keywords: Wireless ad hoc networks, topology control, non-uniform transmission ranges, power consumption, degree-bounded structure.

1 Introduction

Ad hoc wireless networks comprise mobile nodes that communicate via multi-hop wireless channels, which are usually deployed in unattended environments. Though single hop wireless networks (or infrastructured networks) are common, there are a growing number of applications which require multi-hop wireless infrastructure which does not necessarily depend on any fixed base-station, i.e., *ad-hoc*. It has a lot of promising applications, such as emergency search-and-rescue operations, meetings, law enforcement or military applications in which persons wish to quickly share information and data acquisition operations in inhospitable terrain.

Multi-hop structures in wireless networks provide enhanced capacity and fault-tolerance. This capacity allows the use of wireless nodes as repeaters and thus not only enhances the range of communication at low power levels, but also causes less spatial interference and allows reuse of the bandwidth available on the frequency channels at the same time. An important requirement of these networks is that they should be self-organizing, i.e., data paths or routers are dynamically restructured with changing topology.

Ad hoc wireless network needs some special treatment as it intrinsically has its own special characteristics and some unavoidable limitations compared with other wired or wireless network. For example, a transmission by a wireless device is often received by all nodes within its vicinity, which possibly causes signal interferences at these neighboring nodes. On the other hand, we can utilize the property to save the communications for some application. Wireless devices are usually powered by batteries only and have limited memories, which demands high communication efficiency and small routing table. Also, unlike most traditional static communication devices, the wireless devices are often moving or adjusting its transmission range during the communication, which could change the network topology in some extent. Therefore, it is more challenging to design a network topology for ad hoc wireless networks.

In the past several years, topology control algorithms for ad hoc networks have drawn significant research interest. Centralized algorithms can achieve optimality or its approximation, which are more applicable to static networks due to the lack of adaptability to topology changes. In contrast, distributed algorithms are more suitable for mobile ad hoc networks since the environment is inherently dynamic and they are adaptive to topology changes at the cost of possible less optimality. Furthermore, these algorithms only attempt to selectively choose some neighbors of each node. The primary distributed topology control algorithms for ad hoc networks aims to maintain network connectivity, optimize network throughput with power-efficient routing, conserve energy and increase the fault tolerance.

However, priori arts [2, 5, 8, 9, 10, 14, 13, 16] on network topology control assumed that the wireless ad hoc networks are modeled by unit disk graphs (UDG), i.e., two mobile hosts can communicate as long as their Euclidean distance is no more than a threshold. However, practically, the networks are never so perfect as unit disk graphs: the transmission ranges may vary due to various reasons such as the device differences, the network control necessity, and the perturbation of the transmission ranges even the transmission radii are set as the same originally. Thus, we assume that each mobile host has its own transmission range. The networks are modeled by mutual inclusion graphs (MG), where two nodes are connected iff they are within the transmission range of each other. Previously, no method is known for topology control when the networks are modeled as mutual inclusion graphs.

In this paper, we concentrate on designing distributed topology control methods, aiming to build a sparse power efficient network topology in MG. Our methods also works for wireless ad hoc networks with directional antennas. Directional antennas have recently been studied in [12, 15, 4], which have the property that its peak gain is higher than that of a similar antenna with an omni-directional pattern in addition to the advantage of reducing unwanted interference. Ad hoc networks with directional antennas can transmit in specific antenna pattern (direction(s)) to create the desired topology. The network nodes also rely on the discovered topology to communicate by using the least transmission power possible. Observe that, the Yao graph is closely matching ad hoc networks with directional antennas and has some other nice properties which are important for constructing wireless network topology.

The paper proposes the first distributed mechanism to build a sparse power efficient network topology for non-uniform ad hoc wireless networks. The method also works for networks with directional antennas. We first extend the Yao structure to build a spanner with a constant length and power stretch factor for mutual inclusion graph. We then propose two efficient localized algorithms to construct connected sparse network topologies. The first structure, called extended Yao-Yao, has node degree at most $O(\log \gamma)$, where $\gamma = \max_u \max_{uv \in MG} \frac{r_u}{r_v}$. The second structure, called extended Yao and Sink, has node degree bounded by $O(\log \gamma)$, and is a length and power spanner. The methods are based on a novel partition strategy of the space surrounded each mobile host. Both algorithms have communication cost $O(n)$, where each message has $O(\log n)$ bits.

The following sections provide further details of the proposed approach. Preliminaries are presented in Section 2. The proposed approach is described in Section 3. We describe three distributed methods for topology control and analyze their communication complexities, and the stretch factors. We conclude the paper in Section 4 with the discussion of possible future works.

2 Preliminaries

2.1 Network Model

We consider a wireless ad hoc network composed of nodes distributed in a two-dimensional plane. Assume that all wireless nodes have distinctive identities and each static wireless node knows its position information [1] either through a low-power Global Position System (GPS) receiver or through some other way. By one-hop broadcasting, each node u can tell its location information to all nodes within its transmission range. Notice, throughout this paper, a *broadcast* by a node u means u sends the message to all nodes within its transmission range. The main communication cost in wireless networks is to send out the signal while the receiving cost of a message is neglected here. Consequently, throughout this paper, we are interested in designing a protocol with small total number of messages.

All previous known structures are defined solely on the given point set or the unit disk graph. However, graphs representing communication links are rarely so completely specified as the unit disk graph. For example, for wireless communications, different nodes may have different transmission radius. Consequently, two nodes can communicate directly only if they are within the transmission range of each other. Assume each wireless node u has a fixed transmission range r_u. A mutual inclusion graph, denoted by MG, used in wireless ad hoc networks,

[1] More specifically, it is enough for our protocol when each node knows the relative position of its one-hop neighbors. The relative position of neighbors can be estimated by the *direction of arrival* and the *strength of signal*.

has an edge uv if and only if $\|uv\| \leq \min(r_u, r_v)$, as shown in Figure 1 (a). Hereafter, let $D(u, r_u)$ be the disk centered at node u with radius r_u.

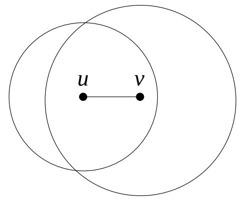


Figure 1: Mutual inclusion graph MG.

2.2 Yao Graph

The *Yao graph* [17] with an integer parameter $k \geq 6$, denoted by $\overrightarrow{YG}_k(G)$, is defined as follows. At each node u, any k equal-separated rays originated at u define k cones. In each cone, choose the shortest edge uv among all edges from u, if there is any, and add a directed link $\overrightarrow{uv}$. Ties are broken arbitrarily or by ID. The resulting directed graph is called the *Yao graph*. Let $YG_k(G)$ be the undirected graph by ignoring the direction of each link in $\overrightarrow{YG}_k(G)$. Some researchers used a similar construction named θ-graph [11, 7], the difference is that it chooses the edge which has the shortest projection on the axis of each cone instead of the shortest edge in each cone.

2.3 Spanners and Stretch Factors

Spanners have been studied intensively in recent years [1, 3, 6, 11]. Let $G = (V, E)$ be a n-vertex weighted connected graph. The distance in G between two vertices $u, v \in V$ is the length of the shortest path between u and v and it is denoted by $d_G(u, v)$. A subgraph $H = (V, E')$, where $E' \subseteq E$, is a *t-spanner* of G if for every $u, v \in V$, $d_H(u, v) \leq t \cdot d_G(u, v)$. The value of t is called the *stretch factor*. When the graph is a geometry graph and the weight is the Euclidean distance between two vertices, the stretch factor t is called the *length stretch factor*, denoted by $\ell_H(G)$. For wireless networks, the mobile devices are usually powered by battery only. We thus pay more attention to the power consumptions. The power, denoted by $p_G(u, v)$, needed to support the communication between a link uv in G is often assumed to be $\|uv\|^\beta$, where $2 \leq \beta \leq 5$ is a constant depending on the transmission environment, and $\|uv\|$ is the Euclidean distance between u and v. When the weight of the geometry graph G is defined as the power to support the communication of the link, the stretch factor of H is called the *power stretch* *factor*, denoted by $\rho_H(G)$ hereafter.

Obviously, for any weighted graph G and a subgraph $H \subseteq G$, we have

Lemma 1 *Graph H has stretch factor δ if and only if for any link $uv \in G$, $d_H(u,v) \leq \delta \cdot d_G(u,v)$.*

2.4 Sparseness and Bounded Degree

The sparseness of all well-known proximity graphs implies that the average node degree is bounded by a constant. We prefer the node degree be bounded by a constant, because wireless nodes have limited resources. Unbounded degree (or in-degree) at node u will often cause large overhead at u, whereas bounded degree increases the network throughput. On the other hand, bounded degree will also give us advantages when apply several routing algorithms. Therefore, it is often imperative to construct a sparse network topology such that both the in-degree and the out-degree are bounded by a constant while it is still power-efficient.

However, in all known primitive proximity graphs, Li *et al.* [9] showed that the maximum node degree could be as large as $n-1$ as shown in Figure 2. The instance consists of $n-1$ points lying on the unit circle centered at a node $u \in V$. Then each edge uv_i belongs to the $RNG(V)$, $GG(V)$ and $\overrightarrow{YG}_k(V)$. Thus, node u has degree $n-1$ (in-degree for $\overrightarrow{YG}_k(V)$) in $RNG(V)$, $GG(V)$ and $\overrightarrow{YG}_k(V)$, although $\overrightarrow{YG}_k(V)$ has a bounded out-degree k.

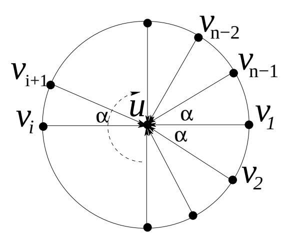


Figure 2: Node u has degree (or in-degree) $n-1$.

Recently, some improved or combined primitive proximity graphs [10] have been proposed to build degree-bounded sparse power efficient topology for UDG.

3 Proposed Approaches

Usually, simple extension of the Yao structure from UDG to MG even does not guarantee the connectivity. Remember that, in UDG, Li *et al.* [9] uses induction to prove the Yao structure on UDG is connected. For any link

$uv \in UDG$, if uv is not in Yao structure, then there is a node w such that uw is in Yao structure and link wv is in UDG, and with length less than uv. The property that link $wv \in UDG$ and $\|wv\| < \|uv\|$ is essential there. However, as shown in Figure 3, for MG this property does not hold anymore since node w could have transmission range smaller than $\|wv\|$, so deleting link uv will violate connectivity. Thus we need more sophisticated extensions of the Yao structure to MG. Notice that UDG is a special case of MG.

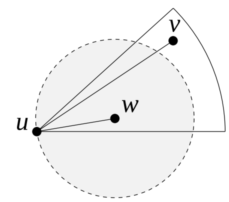


Figure 3: Simple extension of the Yao structure does not guarantee the connectivity.

In following sections, we present several new algorithms that constructs a sparse and power efficient topology for MG.

3.1 Extended Yao Graph

Assume that each node v_i of MG has a unique identification number $ID(v_i) = i$. The identity of a bidirectional link uv is defined as $ID(uv) = (\|uv\|, ID(u), ID(v))$. Please note that we use the bidirectional links instead of the directional links to enhance connectivity. In other words, we require that both node u and node v can communicate with each other through this link. In this paper, all proofs about connectivity or stretch factors take the notation uv and vu as same, which is meaningful. Only in the topology building algorithm or proofs about bounded-degree, uv is different than vu: the former is initiated and built by u, whereas the latter is by node v. Sometimes we denote a directional link from v to u as $\overrightarrow{vu}$ if necessary. Then we can order all bidirectional links (at most $n(n-1)$ such links) in an increasing order of their identities. Here the identities of two links are ordered based on the following rule: $ID(uv) > ID(pq)$ if

1. $\|uv\| > \|pq\|$ or
2. $\|uv\| = \|pq\|$ and $ID(u) > ID(p)$ or
3. $\|uv\| = \|pq\|$, $u = p$ and $ID(v) > ID(q)$.

Correspondingly, the rank of each link uv, denoted by $rank(uv)$, is its order in the sorted bidirectional links. Notice that, we actually only have to consider the links in MG. For the remainder of the subsection, we present our network topology control algorithm and then show that the constructed network topology is a connected spanner.

Algorithm 1 *Constructing-EYG*

1. Each node u divides the disk centered at u with radius r_u by k equal-sized cones centered at u. We generally assume that the cone is half open and half-close. Let $C_i(u)$, $1 \leq i \leq k$, be the set of nodes v inside the ith cone of node u with a larger radius than u. Initially, $C_i(u)$ is empty.

2. In the beginning, each node u broadcasts a message with $ID(u)$, r_u and its position (x_u, y_u) to all nodes in its transmission range.

3. At the same time, each node processes the incoming broadcast messages from some node v. If v is inside the ith cone of node u and $r_v \geq r_u$, then set $C_i(u) = C_i(u) \bigcup \{v\}$. If $v \notin D(u, r_u)$, v is not considered here.

4. Node u chooses a node v from each cone $C_i(u)$ so that the link uv has the smallest $ID(uv)$ among all links uv_j with v_j in $C_i(u)$, if there is any.

5. Finally, each node u informs all 1-hop neighbors of its chosen links through a broadcast message.

Let $\overrightarrow{EYG}_k(G)$ be the union of all chosen links. In other words, the above method computes the Extended Yao graph $\overrightarrow{EYG}_k(G)$ for MG. Since the symmetric communications are required, let $EYG_k(G)$ be the undirected graph by ignoring the direction of each link in $\overrightarrow{EYG}_k(G)$. Graph $EYG_k(G)$ is the final network topology. Since node u chooses a node $v \in D(u, r_u)$ with $r_v \geq r_u$, link uv is indeed a bidirectional link, i.e., u and v are within the transmission range of each other. Additionally, this strategy could avoid the possible disconnectivity by simple Yao extension we mentioned before.

Obviously, each node only broadcasts twice: one for broadcasting its ID, radius and position; and the other one for broadcasting the selected neighbors. Remember that it selects at most k neighbors. Thus, each node sends messages at most $O((k+1) \cdot \log n)$ bits. Here, we assume that the node ID and its position can be represented using $O(\log n)$ bits for a n-node wireless network.

Theorem 2 *The length stretch factor of the Yao graph $EYG_k(G)$, $k > 6$, is at most $\ell = \frac{1}{1-2\sin(\frac{\pi}{k})}$.*

PROOF. From Lemma 1, it is sufficient to show that for any nodes u and v with $\|uv\| \leq \min(r_u, r_v)$, i.e. $uv \in MG$, there is a path connecting u and v in $EYG_k(G)$ with length at most $\ell\|uv\|$. We construct a path $u \leftrightarrow v$ connecting u and v in $EYG_k(G)$ as follows.

Assume that $r_u \leq r_v$. If link $uv \in EYG_k(G)$, then set the path $u \leftrightarrow v$ as the link uv. Otherwise, there must exist another node w in the same cone as v, which is a neighbor of u in $EYG_k(G)$. Then set $u \leftrightarrow v$ as the concatenation of the link uw and the path $w \leftrightarrow v$. Notice that the angle θ of each cone section is $\frac{2\pi}{k}$. When $k > 6$, then $\theta < \frac{\pi}{3}$. It is easy to show that $\|wv\| < \|uv\|$. Consequently, the path $u \leftrightarrow v$ is a simple path, i.e., each node appears at most once.

We prove by induction that the path $u \leftrightarrow v$ has total length at most $\ell\|uv\|$.

Obviously, if there is only one edge in $u \leftrightarrow v$, $d(u \leftrightarrow v) = \|uv\| < \ell\|uv\|$. Assume that the claim is true for any path with l edges. Then consider a path $u \leftrightarrow v$ with $l+1$ edges, which is the concatenation of edge uw and the path $w \leftrightarrow v$ [2] with l edges, as shown in Figure 4 where $\|wv\| = \|xv\|$.

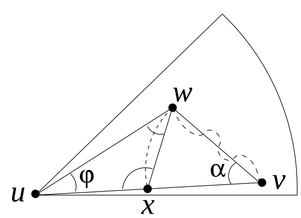


Figure 4: The length stretch factor of the Extended Yao graph is at most $\ell = \frac{1}{1-2\sin(\frac{\pi}{k})}$.

[2]In the procedure of induction, if $r_w \leq r_v$ then we induct on path $w \leftrightarrow v$, otherwise we induct on path $v \leftrightarrow w$. In fact, here $w \leftrightarrow v$ is same as $v \leftrightarrow w$ since the path is bidirectional for communication. Directional link is only considered in building process and is meaningless when we talk about the path. This induction rule is applied throughout the remainder of the paper.

Notice, from induction, $d(w \leftrightarrow v) \leq \ell \|wv\|$. Then, let $\varphi = \angle wuv$ and $\alpha = \angle uvw$, we have

$$
\begin{aligned}
\frac{\|uw\|}{\|ux\|} &= \frac{\sin(\angle uxw)}{\sin(\angle xwu)} = \frac{\sin(\frac{\pi}{2} + \frac{\alpha}{2})}{\sin(\frac{\pi}{2} + \frac{\alpha}{2} + \varphi)} \\
&= \frac{\cos(\frac{\alpha}{2})}{\cos(\frac{\alpha}{2} + \varphi)} = \frac{1}{\cos\varphi - \sin\varphi \tan\frac{\alpha}{2}} \\
&\leq \frac{1}{\cos\varphi - \sin\varphi \tan(\frac{\pi}{4} - \frac{\varphi}{4})} \quad (Since\ \ 0 \leq \alpha \leq \frac{\pi}{2} - \frac{\varphi}{2}) \\
&= \frac{cos(\frac{\pi}{4} - \frac{\varphi}{4})}{cos(\frac{\pi}{4} + \frac{3}{4}\varphi)} \leq \frac{cos(\frac{\pi}{4} - \frac{\pi}{2k})}{cos(\frac{\pi}{4} + \frac{3\pi}{2k})} \quad (Since\ \ 0 \leq \varphi \leq \frac{2\pi}{k}) \\
&= \frac{1}{1 - 2\sin(\frac{\pi}{k})}
\end{aligned}
$$

Define $\ell = \frac{1}{1-2\sin(\frac{\pi}{k})}$. Consequently,

$$d(u \leftrightarrow v) = \|uw\| + d(w \leftrightarrow v) < \ell\|ux\| + \ell\|wv\| = \ell\|uv\|.$$

That is to say, the claim is also true for the path $w \leftrightarrow v$ with $l+1$ edges.

So, the length stretch factor of the Extended Yao graph is at most $\ell = \frac{1}{1-2\sin(\frac{\pi}{k})}$. This finishes the proof. ◻

Theorem 3 *The power stretch factor of the extended Yao graph $EYG_k(G)$, $k > 6$, is at most $\rho = \frac{1}{1-(2\sin\frac{\pi}{k})^{\beta}}$.*

Proof. The proof is similar to that in UDG [9, 10] except the induction procedure. For the completeness of presentation, we shall give the detail here.

From Lemma 1, it is sufficient to show that for any nodes u and v with $\|uv\| \leq \min(r_u, r_v)$, i.e. $uv \in MG$, there is a path connecting u and v in $EYG_k(G)$ with power consumption at most ρ. We construct a path $u \leftrightarrow v$ connecting u and v in $EYG_k(G)$ as follows.

Assume that $r_u \leq r_v$. If link $uv \in EYG_k(G)$, then set the path $u \leftrightarrow v$ as the link uv. Otherwise, there must exist another node w in the same cone as v such that the directed link uw is in $\overrightarrow{EYG}_k(G)$ from Algorithm 1. Then set the path $u \leftrightarrow v$ as the concatenation of the undirected link uw and path $w \leftrightarrow v$. Remember that if $r_w > r_v$, we actually construct path $v \leftrightarrow w$. Notice that the angle θ of each cone is $\frac{2\pi}{k}$. When $k > 6$, then $\theta < \frac{\pi}{3}$. It is easy to show that $\|wv\| < \|uv\|$. Consequently, the path $u \leftrightarrow v$ is a simple path, i.e., each node appears at most once.

We then prove by induction, on the number of its edges, that the path $u \leftrightarrow v$ has power cost, denoted by $p(u \leftrightarrow v)$, at most $\rho\|uv\|^\beta$.

Obviously, if there is only one edge in $u \leftrightarrow v$, $p(u \leftrightarrow v) = \|uv\|^\beta < \rho\|uv\|^\beta$. Assume that the claim is true for any path with l edges. Then consider a path $u \leftrightarrow v$ with $l+1$ edges, which is the concatenation of edge uw and the path $w \leftrightarrow v$ with l edges. We consider two cases.

Case 1: the angle $\angle uwv$ is not acute. See Figure 5 (a). We have $\|uw\|^2 + \|wv\|^2 \leq \|uv\|^2$. Notice that $\frac{\|uw\|}{\|uv\|} \leq 1$ and $\frac{\|wv\|}{\|uv\|} \leq 1$. It implies that

$$\left(\frac{\|uw\|}{\|uv\|}\right)^\beta + \left(\frac{\|wv\|}{\|uv\|}\right)^\beta \leq \left(\frac{\|uw\|}{\|uv\|}\right)^2 + \left(\frac{\|wv\|}{\|uv\|}\right)^2 \leq 1$$

Therefore,

$$\|uw\|^\beta + \|wv\|^\beta \leq \|uv\|^\beta$$

for any $\beta \geq 2$. Since $\|wv\| < \|uv\|$, we can apply induction on the path $w \leftrightarrow v$ also. Therefore, $p(w \leftrightarrow v) \leq \rho\|wv\|^\beta$ by induction. Then

$$p(u \leftrightarrow v) = \|uw\|^\beta + p(w \leftrightarrow v) \leq \|uw\|^\beta + \rho\|wv\|^\beta \leq \rho\|uv\|^\beta.$$

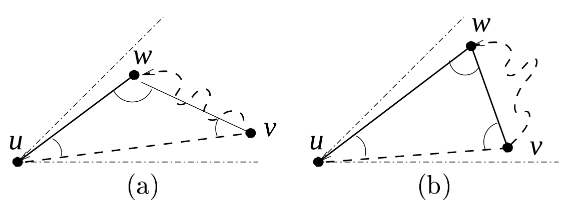


Figure 5: (a) The angle $\angle uwv$ is not acute. (b)The angle $\angle uwv$ is acute.

Case 2: the angle $\angle uwv$ is acute. See Figure 5 (b). We bound the length $\|wv\|$ respecting to $\|uv\|$. Notice that $\|uw\| \leq \|uv\|$ and $\angle wuv < \theta$. The maximum length of vw is achieved when $\|uw\| = \|uv\|$ because the angle $\angle uwv$ is acute. Therefore

$$\|wv\| \leq 2\sin\frac{\theta}{2}\|uv\| = 2\sin\frac{\pi}{k}\|uv\|.$$

By induction, we have

$$p(u \leftrightarrow v) = \|uw\|^\beta + p(w \leftrightarrow v) \leq \|uw\|^\beta + \rho\|wv\|^\beta \leq \|uv\|^\beta + \rho \cdot (2\sin\frac{\pi}{k})^\beta \|uv\|^\beta = \rho\|uv\|^\beta.$$

This finishes the proof. ◻

3.2 Novel Space Partition

Partitioning the space surrounding a node into k equal-sized cones enables us to bound the node out-degree using the Yao structure. Using the same space partition, Yao-Yao structure [9, 10] produces a topology with bounded in-degree when the networks are modeled by UDG. They also showed that another structure YaoSink [9, 10] has not only the bounded node degree but also constant bounded stretch factors. The network topology with bounded degree can increase the communication efficiency. These methods [9, 10] may fail when the networks are modeled by MG: they cannot even guarantee the connectivity.

Let $I(v) = \{w \mid wv \in \overrightarrow{EYG}_k(G)\}$. In other words, $I(v)$ is the set of nodes that have directed links to v in $\overrightarrow{EYG}_k(G)$. Let $I_i(v)$ be the nodes in $I(v)$ located inside the ith cone. Remember that Yao-Yao and YaoSink structures will pick the closest node in $I_i(v)$. In addition, YaoSink structure will recursively build a sink tree to connect all nodes $I(v)$. See [9, 10] for more detail. Figure 6 illustrates an example such that a node v has $p+1$ incoming neighbors w_i, $0 \leq i \leq p$, in the Yao structure. Node v will only select the closest neighbor w_0 in the Yao-Yao structure. The connectivity of the final Yao-Yao structure is proved by induction on the length of links [10]. Here the existence of links $w_j w_0$, $1 \leq j \leq p$, is essential and this is trivially satisfied in UDG. However, this is not the case in MG. Assume that each node w_i has a transmission radius $r_{w_i} = 3^i a \leq r_v$ and $\|vw_i\| = r_{w_i}$. Here a is a positive real number satisfying $3^p a \leq r_v$. Obviously, $\|w_i w_j\| > \min(r_{w_i}, r_{w_j})$, i.e., any two nodes w_i, w_j are not directly connected in MG. Clearly, the Yao-Yao structure will only have one edge uw_0 left in this configuration of nodes, thus disconnecting the network.

Selecting the closest incoming neighbor in each cone is too aggressive to guarantee the connectivity. Observe that, to guarantee the connectivity, when we delete a directed link $\overrightarrow{w_i v}$, we need to keep *some* link, say $w_j v$, such that $w_i w_j$ is a link in MG. Thus, we want to further partition the cone into a limited number of smaller *regions* and

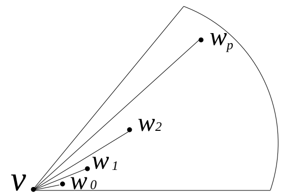


Figure 6: All nodes w_i has a directed edge to node v in the Yao structure.

we will keep the closest node in each region. Clearly, we have to make sure that any two nodes w_i, $w_j \in I(v)$ that co-exist in a same small region are directly connected in MG. Consequently, if the number of regions is bounded by a constant, a degree-bounded structure could be generated. In the remainder of this subsection, we will introduce a novel space partition strategy based on pigeon-hole principal.

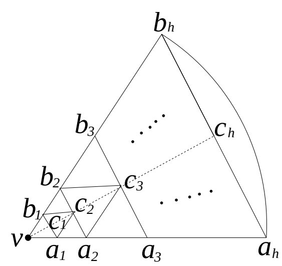


Figure 7: The space partition for each cone.

Method 1 *Partition-EYG*

For each node v, let $\gamma_v = \max_{w \in I(v)} \frac{r_v}{r_w}$. Remember that all nodes in $I(v)$ have transmission radius at most r_v. Let h be the positive integer satisfying $2^{h-2} < \gamma_v \leq 2^{h-1}$. We then discuss in detail our partition strategy of the cones, which is illustrated by Figure 7. Each node v divides each cone centered at v into limited number of triangles and caps, where $\|va_i\| = \|vb_i\| = \frac{1}{2^{h-i}} r_v$ and c_i is the mid-point of the segment $a_i b_i$, for $1 \leq i \leq h$. Notice that this partition can be conducted by node v locally since it can collect the transmission radius information of nodes in $I(v)$. The triangles $\triangle v a_1 b_1$, $\triangle a_i b_i c_{i+1}$, $\triangle a_i a_{i+1} c_{i+1}$, $\triangle b_i b_{i+1} c_{i+1}$, for $1 \leq i \leq h-1$, and the cap $\widehat{a_n b_n}$ form the final space partition of each cone. For simplicity, we call such a triangle or the cap as a *region*. We then prove that this partition indeed guarantees that any two nodes in any same region are connected in MG.

Lemma 4 *Assume that $k \geq 6$. Any two nodes $u, w \in I(v)$ that co-exist in any one of the generated regions are directly connected in MG, i.e., $\|uw\| < \min(r_u, r_w)$.*

Proof. There are four different cases.

1. Two nodes are in triangle $\triangle va_1b_1$, as shown in Figure 8.

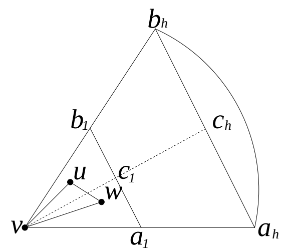


Figure 8: Two nodes are in triangle $\triangle va_1b_1$.

Remember that all nodes in $I(v)$ have transmission radius at least $\|va_1\| = \frac{1}{2^{(n-1)}}r_v$. We have $\min(r_u, r_w) \geq \|va_1\| = \|vb_1\|$ and $\|a_1b_1\| \leq \|va_1\|$. In addition, since uw is a segment inside $\triangle va_1b_1$, we have $\|uw\| \leq \max(\|a_1b_1\|, \|va_1\|, \|vb_1\|)$. Consequently, $\|uw\| < \min(r_u, r_w)$, i.e. $uw \in MG$.

2. Two nodes are in triangle $\triangle a_ib_ic_{i+1}$, as shown in Figure 9.

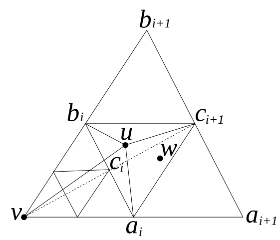


Figure 9: Two nodes are in triangle $\triangle a_ib_ic_{i+1}$.

In this case, we have

(a) $\|vu\| > \|uc_{i+1}\|$, because a_ib_i is the perpendicular bisector of vc_{i+1} and u is at the same side of line a_ib_i as c_{i+1}.

(b) $\|vu\| > \|ua_i\|$, because $\angle va_iu > \frac{\pi}{3} > \angle uva_i$.

(c) $\|vu\| > \|ub_i\|$, because $\angle vb_iu > \frac{\pi}{3} > \angle uvb_i$.

(d) $\|uw\| < \max(\|uc_{i+1}\|, \|ua_i\|, \|ub_i\|)$, because node w must be inside one of the triangles $\triangle a_ib_iu$, $\triangle a_ic_{i+1}u$ and $\triangle b_ic_{i+1}u$.

Thus, $\|uw\| < \|uv\|$. Similarly, $\|uw\| < \|wv\|$. Consequently, $uw \in MG$ from

$$\|uw\| < \min(\|uv\|, \|wv\|) < \min(r_u, r_w).$$

3. Two nodes are in triangle $\triangle a_i a_{i+1} c_{i+1}$, as shown in Figure 10.

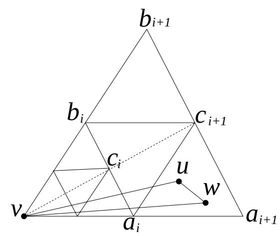


Figure 10: Two nodes are in triangle $\triangle a_i a_{i+1} c_{i+1}$.

We have

$$\min(r_u, r_w) \geq \|va_i\| = \|a_i a_{i+1}\| = \|a_i c_{i+1}\| > \|a_{i+1} c_{i+1}\|.$$

Since uw is a segment inside $\triangle a_i a_{i+1} c_{i+1}$, $\|uw\| < \max(\|a_i a_{i+1}\|, \|a_i c_{i+1}\|, \|a_{i+1} c_{i+1}\|) < \min(r_u, r_w)$, i.e. $uw \in MG$.

4. Triangle $\triangle b_i b_{i+1} c_{i+1}$ is the symmetric case with triangle $\triangle a_i a_{i+1} c_{i+1}$, so the claim holds similarly.

5. Two nodes are inside the cap $\widehat{a_h b_h}$, as shown in Figure 11, where $a_h z$ and $b_h z$ is the tangent of arc $\widehat{a_h b_h}$ at point a_h and b_h respectively.

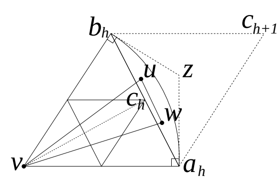


Figure 11: Two nodes are inside cap $\widehat{a_h b_h}$.

Since $\angle a_h v b_h < \frac{2\pi}{k}$, $k \geq 6$, we have

$$\angle v b_h z = \frac{\pi}{2} < \pi - \angle a_h v b_h = \angle v b_h c_{h+1}.$$

Similarly, $\angle v a_h z < \angle v a_h c_{h+1}$. This means $\widehat{a_h b_h}$ is inside $\triangle a_h b_h c_{h+1}$. The remaining of the proof directly follows from the proof for the case of $\triangle a_i b_i c_{i+1}$.

This finishes the proof. $\square$

3.3 Extended Yao-Yao Graph

In this section, using the space partition discussed in Section 3.2, we extend the Yao-Yao structure to build distributed sparse ad-hoc network topology with bounded degree if γ is bounded.

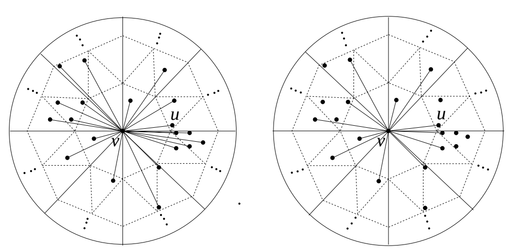
Algorithm 2 *Constructing-EYY*

1. Each node finds the incident edges in the Extended Yao graph $\overrightarrow{EYG}_k(G)$, as described in Algorithm 1.

2. Each node v partitions the k cones centered at v using the partitioning method described in Method 1. Notice that for partitioning, node v uses parameter γ_v in Method 1, which can be easily calculated from local information. Figure 12 (a) illustrates such a partition.

3. Each node v chooses a node u from each generated region so that the link uv has the smallest $ID(uv)$ among all links computed in the first step in the partition. Figure 12 (b) illustrates such a selection of incoming links.

4. Finally, for each link uv selected by v, node v informs node u of keeping link uv.



(a) (b)

Figure 12: (a) Star formed by links toward to u. (b) Node v chooses the shortest link in $EYG_k(G)$ toward itself from each region to produce $EYY_k(G)$.

The union of all chosen links is the final network topology, denoted by $\overrightarrow{EYY}_k(G)$. We call it *extended Yao-Yao* graph. Let $EYY_k(G)$ be the undirected graph by ignoring the direction of each link in $\overrightarrow{EYY}_k(G)$.

Theorem 5 *The out-degree of each node v in $\overrightarrow{EYY}_k(G)$, $k \geq 6$, is bounded by k and the in-degree is bounded by $(3\lceil \log_2 \gamma_v \rceil + 2)k$, where $\gamma_v = \max_{w \in I(v)}(\frac{r_v}{r_w})$.*

PROOF. It is obvious that the out-degree of node v is bounded by k because the out-degree bound of $\overrightarrow{EYG}_k(G)$ is k and this algorithm does not add any directed link.

For the in-degree bound, as shown in Figure 7, obviously, the number of triangle regions in each cone is $3h-2$. Remember that $2^{h-2} < \gamma_v \leq 2^{h-1}$, which implies $h = 1 + \lceil \log_2 \gamma_v \rceil$. Thus, considering the cap region also, the in-degree of node v is at most $(3\lceil \log_2 \gamma_v \rceil + 2)k$. ◻

Notice that $\gamma \geq \max_v \gamma_v$. Obviously, the maximum node degree in graph $EYY_k(G)$ is bounded by $(3\lceil \log_2 \gamma \rceil + 3)k$.

We then show that, in the worst case, any connected graph has degree at least $O(\log_2 \gamma)$, i.e., the structure generated by our Algorithm 2 has degree within a constant factor of the smallest possible. Consider the example illustrated by Figure 6. Edges vw_i, $0 \leq i \leq p$, are all possible communication links. Thus, node v in any connected spanning graph has degree $p+1$. Assume that $3^p a = r_v$. It is easy to compute γ_v as $r_v/a = 3^p$. Thus, v has degree $\log_3 \gamma_v + 1 = O(\log_2 \gamma)$.

Notice that the extended Yao-Yao graph $EYY_k(G)$ is a subgraph of the extended Yao graph $EYG_k(G)$, there are at most $k \cdot n$ edges in $EYY_k(G)$. Thus, the total communications of Algorithm 2 is at most $O(k \cdot n)$, where each message has $O(\log n)$ bits. It is interesting to see that the communication complexity does not depend on γ at all.

Theorem 6 *The graph $EYY_k(G)$, $k \geq 6$, is connected if MG is connected .*

PROOF. Notice that it is sufficient to show that there is a path from u to v for any two nodes with $uv \in MG$. Remember the graph $EYG_k(G)$ is connected, therefore, we only have to show that $\forall uv \in EYG_k(G)$, there is a path connecting u and v in $EYY_k(G)$. We prove this claim by induction on the ranks of all links in $EYG_k(G)$.

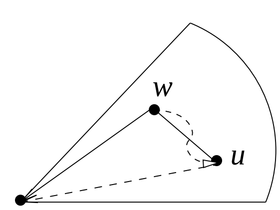


Figure 13: $EYY_k(G)$ is connected if $EYG_k(G)$ is connected.

If the link uv has the smallest rank among all links of $EYG_k(G)$, then uv will obviously survive after the second step. So the claim is true for the smallest rank.

Assume that the claim is true for all links in $EYG_k(G)$ with rank at most r. Then consider a link uv in $EYG_k(V)$ with $rank(uv) = r + 1$ in $EYG_k(G)$. If uv survives in Algorithm 2, then the claim holds. Otherwise, assume that $r_u < r_v$. Then directed edge vu cannot belong to $\overrightarrow{EYG}_k(G)$ from Algorithm 1. Thus, directed edge uv is in $\overrightarrow{EYG}_k(G)$. In Algorithm 2, directed edge uv can only be removed by node v due to the existence of another directed link wv with a smaller identity and w is in the same region as u. In addition, the angle $\angle wvu$ is less than $\theta = \frac{2\pi}{k}(k \geq 6)$. Therefore we have $\|wu\| < \|uv\|$. Notice that here wu is guaranteed to be a link in MG, but it is not guaranteed to be in $EYG_k(G)$. We then prove by induction that there is a path connecting w and u in $EYY_k(G)$. Assume $r_w \leq r_u$. The scenario $r_w > r_u$ can be proved similarly. There are two cases here.

Case 1: the link wu is in $EYG_k(G)$. Notice that rank of wu is less than the rank of uv. Then by induction, there is a path $w \leftrightarrow u$ connecting w and u in $EYY_k(G)$. Consequently, there is a path (concatenation of the undirected path $w \leftrightarrow u$ and the link wv) between u and v.

Case 2: the link wu is not in $EYG_k(G)$. Then, from proof of Theorem 2, we know that there is a path $\Pi_{EYG_k}(w, u) = q_1 q_2 \cdots q_m$ from w to u in $EYG_k(G)$, where $q_1 = w$ and $q_m = u$. Additionally, we can show that each link $q_i q_{i+1}$, $1 \leq i < m$, has a smaller rank than wu, which is at most r. Here $rank(q_1 q_2 = wq_2) < rank(w, u)$ because the selection method in Algorithm 1. And $rank(q_i q_{i+1}) < rank(w, u)$, $1 < i < m$, because

$$\|q_i q_{i+1}\| \leq \|q_i u\| < \|q_{i-1} u\| < \cdots < \|q_1 u\| = \|wu\|.$$

Then, by induction, for each link $q_i q_{i+1}$, there is a path $q_i \leftrightarrow q_{i+1}$ survived in $EYY_k(G)$ after Algorithm 2. The concatenation of all such paths $q_i \leftrightarrow q_{i+1}$, $1 \leq i < m$, and the link wv forms a path from u to v in $EYY_k(G)$.

This finishes the proof. □

Although $EYY_k(G)$ is a connected structure, it is unknown whether it is a power or length spanner. We leave it as a future work.

3.4 Extended Yao-Sink Graph

In [9, 10], the authors applied the technique in [1] to construct a sparse network topology in UDG, *Yao and sink graph*, which has a bounded degree and a bounded stretch factor. The technique is to replace the directed star consisting of all links toward a node v by a directed tree $T(v)$ with v as the sink. Tree $T(v)$ is constructed recursively. To apply this technique into MG, we need extend it by a more sophisticated way.

Algorithm 3 *Constructing-EYG**

1. Each node finds the incident edges in the Extended Yao graph $\overrightarrow{EYG}_k(G)$, as described in Algorithm 1. Each node v will have a set of incoming nodes $I(v) = \{u \mid uv \in \overrightarrow{EYG}_k(G)\}$.

2. Each node v partitions the k cones centered at v using the partitioning method described in Method 1. Notice that for partitioning, node v uses parameter γ_v in Method 1, which can be easily calculated from local information. Figure 14 (a) illustrates such a partition.

3. Each node v chooses a node u from each region Ω, let $\Omega_u(v)$ be the region Ω partitioned by node v with node u inside, so that the link uv has the smallest $ID(uv)$ among all links computed in the first step in the region $\Omega_u(v)$. In other words, in this step, it constructs $\overrightarrow{EYY_k(G)}$.

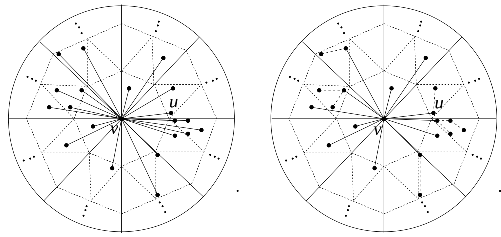
4. For each region $\Omega_u(v)$ and the selected node u, let $S_\Omega(u) = \{w \mid w \neq u,\ w \in \Omega_u(v) \cap I(v)\}$. For each node u, node v uses the following function Tree(u,$S_\Omega(u)$) to build a tree $T(u)$ rooted at u. We call $T(u)$ a *sink tree* and call the union of all links chosen by node v the *sink structure* at v. Figure 14 (b) illustrates a sink structure at v, which is composed of all trees $T(u)$ for u selected in the previous step.

5. Finally, node v informs nodes x and y for each selected link xy in the sink structure rooted at v.

Algorithm 4 *Constructing-Tree Tree(u,$S_\Omega(u)$)*

1. Partition the disk centered at u by k equal-sized cones: $\mathbb{C}_1(u)$, $\mathbb{C}_2(u)$, $\cdots$, $\mathbb{C}_k(u)$.

2. Find the node $w_i \in S_\Omega(u)$ in $\mathbb{C}_i(u)$, $1 \leq i \leq k$, with the smallest $ID(w_iu)$, if there is any. Link w_iu is added to $T(u)$ and node w_i is removed from $S_\Omega(u)$.



(a) (b)

Figure 14: (a) Star formed by links toward to v. (b) The sink structure at v.

3. For each node w_i, call $\mathsf{Tree}(w_i,\, S_\Omega(u) \cap \mathbb{C}_i(u))$ and add the created edges to $T(u)$.

The union of all chosen links is the final network topology, denoted by $EYG_k^*(G)$. We call such structure as the *Extended Yao-Sink* graph. Notice that, sink node v, not u, constructs the tree $T(u)$ and then informs the end-nodes of the selected links to keep such links if already exist or add such links otherwise.

Theorem 7 *The maximum node degree of the graph* $\overrightarrow{EYG}_k^*(G)$ *is at most* $k^2 + 3k + 3k \cdot \lceil \log_2 \gamma \rceil$.

PROOF. Initially, each node has at most k out-degrees after constructing graph $EYG_k(G)$. In the algorithm, each node v initiates only one sink structure, which will introduce at most $(3\lceil \log_2 \gamma \rceil + 2) \cdot k$ in-degrees. Additionally, each node x will be involved in Algorithm 4 for at most k sink trees (once for each directed link $xy \in EYG_k(G)$). For each sink tree involvement, Algorithm 4 assigns at most k links incident on x. Thus, at most k^2 new degrees could be introduced here. Then the theorem follows. ◻

Since the total number of edges is at most $(k^2 + 3k + 3k \cdot \lceil \log_2 \gamma \rceil) \cdot n$, the total communication cost of our method is $O(\log_2 \gamma \cdot n)$. Here each message has $O(\log n)$ bits.

Theorem 8 *The length stretch factor of the graph* $EYG_k^*(G)$*,* $k > 6$*, is at most* $(\frac{1}{1-2\sin(\frac{\pi}{k})})^2$.

PROOF. We have proved that $EYG_k(G)$ has length stretch factor at most $\frac{1}{1-2\sin(\frac{\pi}{k})}$. We thus have only to prove that, for each link $vw \in EYG_k(G)$, there is a path connecting them in $EYG_k^*(G)$ with length at most $\frac{1}{1-2\sin(\frac{\pi}{k})}\|vw\|$. If link vw is kept in $EYG_k^*(G)$, then this is obvious. Otherwise, assume $r_w \le r_v$, then directed link wv belongs to $\overrightarrow{EYG}_k(G)$. Then, there must have a node u in the same region (partitioned by node v) as node w. Using the same argument as Theorem 2, we can prove that there is a path connecting v and w in $T(u)$ with length at most $\frac{1}{1-2\sin(\frac{\pi}{k})}\|vw\|$. It implies that the length stretch factor of $EYG_k^*(G)$ is at most $(\frac{1}{1-2\sin(\frac{\pi}{k})})^2$. ◻

Similarly, we have:

Theorem 9 *The power stretch factor of the graph $EYG_k^*(G)$, $k > 6$, is at most* $(\frac{1}{1-(2\sin\frac{\pi}{k})^\beta})^2$.

4 Conclusion

In this paper, we extended the Yao graph to MG model, which is more practical and useful in real communication environment, especially our algorithms could be used in application of ad hoc networks with directional antennas.

We presented several efficient localized algorithms to construct network topologies with bounded node degrees for wireless ad hoc networks. We showed that $EYG_k(G)$, $EYY_k(G)$, and $EYG_k^*(G)$ are connected if MG is connected, while $EYG_k(G)$ and $EYG_k^*(G)$ have constant bounded power and length stretch factors. Additionally, we showed that $EYY_k(G)$ and $EYG_k^*(G)$ have bounded node degrees $O(\log_2 \gamma)$. We show by example that in the worst cast any connected graph will have degree at least $O(\log_2 \gamma)$. In other words, the structures constructed by our method is almost optimum. Our algorithms are all localized and have communication cost at most $O(\log_2 \gamma \cdot n)$, where each message has $O(\log n)$ bits.

These structures can be constructed efficiently even when the wireless nodes are not static. For mobile wireless network, there are three events that will possibly trigger the change of the Yao structure, namely, a node leaving the transmission range, a node entering the transmission range, and a node switching the cone region. Updating the Yao structure is fast in all these three scenarios.

Notice that, it is an open problem whether graph $EYY_k(G)$ is a length or power spanner. Some other future works could be, what are the conditions that we can build a structure with some other properties for MG, such as planar or low weight.

References

[1] S. Arya, G. Das, D. Mount, J. Salowe, and M. Smid. Euclidean spanners: short, thin, and lanky. In *Proc. 27th ACM STOC*, pages 489–498, 1995.

[2] Mohsen Bahramgiri, Mohammad Taghi Hajiaghayi, and Vahab S. Mirrokni. Fault-tolerant and 3-dimensional distributed topology control algorithms in wireless multi-hop networks. In *Proceedings of the 11th Annual IEEE Internation Conference on Computer Communications and Networks (ICCCN)*, pages 392–397, 2002.

[3] P. Bose, L. Devroye, W. Evans, and D. Kirkpatrick. On the spanning ratio of gabriel graphs and beta-skeletons. In *Proceedings of the Latin American Theoretical Infocomatics (LATIN)*, 2002.

[4] J.A. Cobb. Forward-only uni-directional routing. In *Computer Communications and Networks, 2002. Proceedings. Eleventh International Conference*, pages 370–375, 2002.

[5] L. Hu. Topology control for multihop packet radio networks. *IEEE Trans. Communications*, 41(10), 1993.

[6] J.W. Jaromczyk and G.T. Toussaint. Relative neighborhood graphs and their relatives. *Proceedings of IEEE*, 80(9):1502–1517, 1992.

[7] J. M. Keil and C. A. Gutwin. Classes of graphs which approximate the complete euclidean graph. *Discr. Comp. Geom.*, 7:13–28, 1992.

[8] Li Li, Joseph Y. Halpern, Paramvir Bahl, Yi-Min Wang, and Roger Wattenhofer. Analysis of a cone-based distributed topology control algorithms for wireless multi-hop networks. In *ACM Symposium on Principle of Distributed Computing (PODC)*, 2001.

[9] Xiang-Yang Li, Peng-Jun Wan, and Yu Wang. Power efficient and sparse spanner for wireless ad hoc networks. In *IEEE Int. Conf. on Computer Communications and Networks (ICCCN01)*, pages 564–567, 2001.

[10] Xiang-Yang Li, Peng-Jun Wan, Yu Wang, and Ophir Frieder. Sparse power efficient topology for wireless networks. In *IEEE Hawaii Int. Conf. on System Sciences (HICSS)*, 2002.

[11] Tamas Lukovszki. *New Results on Geometric Spanners and Their Applications.* PhD thesis, University of Paderborn, 1999.

[12] Asis Nasipuri, Kai Li, and Uma Reddy Sappidi. Power consumption and throughput in mobile ad hoc networks using directional antennas. In *Computer Communications and Networks, 2002. Proceedings. Eleventh International Conference*, pages 620– 626, 2002.

[13] Rajmohan Rajaraman. Topology control and routing in ad hoc networks: A survey. *SIGACT News*, 33:60–73, 2002.

[14] R. Ramanathan and R. Rosales-Hain. Topology control of multihop wireless networks using transmit power adjustment. In *IEEE INFOCOM*, 2000.

[15] Chien-Chung Shen, C. Jaikaeo, C. Srisathapornphat, and Zhuochuan Huang. Topology control for ad hoc networks with directional antennas. In *Computer Communications and Networks, 2002. Proceedings. Eleventh International Conference*, pages 16– 21, 2002.

[16] Roger Wattenhofer, Li Li, Paramvir Bahl, and Yi-Min Wang. Distributed topology control for wireless multihop ad-hoc networks. In *IEEE INFOCOM'01*, 2001.

[17] A. C.-C. Yao. On constructing minimum spanning trees in k-dimensional spaces and related problems. *SIAM J. Computing*, 11:721–736, 1982.