

Delta-Graphs for Wireless Ad Hoc Networks

Ashikur Rahman and Carey Williamson

Department of Computer Science, University of Calgary
2500 University Drive, NW, Calgary, AB, Canada, T2N 1N4
Email: arahma@ucalgary.ca, carey@cpsc.ucalgary.ca

Abstract—Most topology control algorithms for wireless ad hoc networks strive to reduce energy consumption by creating a sparse topology with few long-distance links. However, in a sparse topology, the average path length is relatively large (increasing end-to-end delay), and the number of vertex-disjoint paths between source-destination pairs is relatively small (reducing fault-tolerance). Unlike traditional topology control algorithms that generate a single topology with a certain property, we propose a distributed algorithm that generates a family of topologies with a range of characteristics. The network designer can choose a suitable topology by simply tuning a single parameter Δ (power savings threshold), trading off energy savings for other features such as low latency and fault-tolerance. For the topologies generated by the proposed algorithm, we also provide an analytical model to estimate their structural density. The accuracy of the analytical model is validated with extensive simulation results.

I. INTRODUCTION

Wireless ad hoc networks are known as “networks without networking”, since they do not rely on any pre-configured fixed infrastructure. Rather, the nodes can be deployed spontaneously, perhaps at random, and then be dynamically reconfigured into a manageable and controllable network topology. Such dynamic topology control is widely used in wireless sensor networks, vehicular networks, and wireless mesh network environments, to name a few.

Many of these wireless ad hoc networks consist of mobile nodes equipped with limited power sources, such as a battery or a solar cell. Thus energy consumption is a prime consideration in designing such networks. The transmission power between a pair of communicating nodes is one of the dominant factors in the overall energy consumption. Since the transmission power requirement grows (at least) quadratically with the distance between the communicating parties, using one or more nearby neighboring nodes as intermediate *relays* can dramatically reduce energy consumption. Thus any energy-aware topology control algorithm gives preference to short-distance links rather than long-distance links.

In general, a topology control algorithm works as follows. Each node in the network classifies all of its directly reachable neighbors as either “close” or “far”. The “closeness” of a neighbor is based on a suitably chosen *objective function*. If the goal is to conserve energy, then direct communication is used for the close neighbors, while indirect communication (via a relay node) is used for the far neighbors. After the classification step, each node reduces its transmission power to the minimum required to communicate with all of its close neighbors. Thereby a node uses one or more close neighbors to relay information to the far neighbors.

One of the major challenges is how to define the energy efficiency rules for determining the closeness of a neighbor. There are many solutions proposed in the literature [4], [7], [16], [20], [21], most of which produce very sparse topologies. However, energy conservation is not the only important goal. When other metrics are considered, the strict rules used to enforce sparse topologies may introduce some new problems. For example, the tendency toward numerous short-distance links increases the end-to-end store-and-forward delay for messages. Moreover, these strict rules may eliminate too many neighbors, reducing the multi-path redundancy between node pairs. With fewer neighbors, the sender has limited choice for forwarding nodes, which can lead to network congestion along the minimum-energy paths in the network.

In this paper, we argue that slightly relaxing the energy-efficiency rules for topology formation can provide flexibility for augmenting other desirable topological properties (e.g., latency, fault-tolerance, load balancing). For example, it may not be worth using a relay node if it saves only 5% transmission power but adds 30% more latency. Specifically, we show that by relaxing the power control rule, it is easy to create a family of topologies with enhanced features like low latency and fault tolerance. We propose a simple, flexible distributed topology control algorithm, with a single tunable parameter Δ to control network density and energy efficiency. We also provide an analytical model to estimate the expected fraction of links eliminated from a node’s neighborhood, and present simulation results to validate the analytical model.

Figure 1 illustrates the central idea in our paper. Figure 1(a) shows the initial topology for a random network of 100 nodes, with the link connectivity induced by a wireless transmission range of 100 m. Figure 1(f) shows the well-known *Gabriel graph*, which prunes all network links for which the use of an intermediate relay node provides a lower-energy path for communication. In between these two extremes, our Δ -graph approach defines a family of network topologies, with the level of link pruning controlled by the parameter Δ . Specifically, we prune direct links for which the use of an intermediate relay node saves at least a fraction $\Delta \geq 0$ in terms of energy consumption. For $\Delta = \infty$ (actually, $\Delta = 0.5$ in this example), the Δ -graph corresponds to the initial graph. For $\Delta = 0$, the Δ -graph corresponds to the Gabriel graph. For other values $0 \leq \Delta \leq 0.5$, the topologies differ in the density of their connectivity, providing interesting tradeoffs between path length, delay, fault-tolerance, and energy consumption. If the initial graph is connected, so are the Δ -graphs, for all

TABLE I
SUMMARY STATISTICS FOR Δ -GRAPH EXAMPLE IN FIGURE 1

Graph Example				Shortest-Hop Path			Shortest-Distance Path			Lowest-Energy Path		
Figure	Δ	Nodes	Links	Hops	Dist	Energy	Hops	Dist	Energy	Hops	Dist	Energy
1(a)	0.5	100	718	5.06	401.0	32886	5.22	377.5	29064	8.94	409.5	22893
1(b)	0.4	100	590	5.38	407.3	32297	5.69	378.4	27733	8.94	409.5	22893
1(c)	0.3	100	502	5.88	413.1	31101	6.23	381.8	26610	8.94	409.5	22893
1(d)	0.2	100	422	6.94	426.1	29623	7.37	391.6	24858	8.94	409.5	22893
1(e)	0.1	100	360	7.72	432.5	27978	8.06	398.5	23923	8.94	409.5	22893
1(f)	0.0	100	320	8.02	434.6	27920	8.27	402.2	23802	8.94	409.5	22893

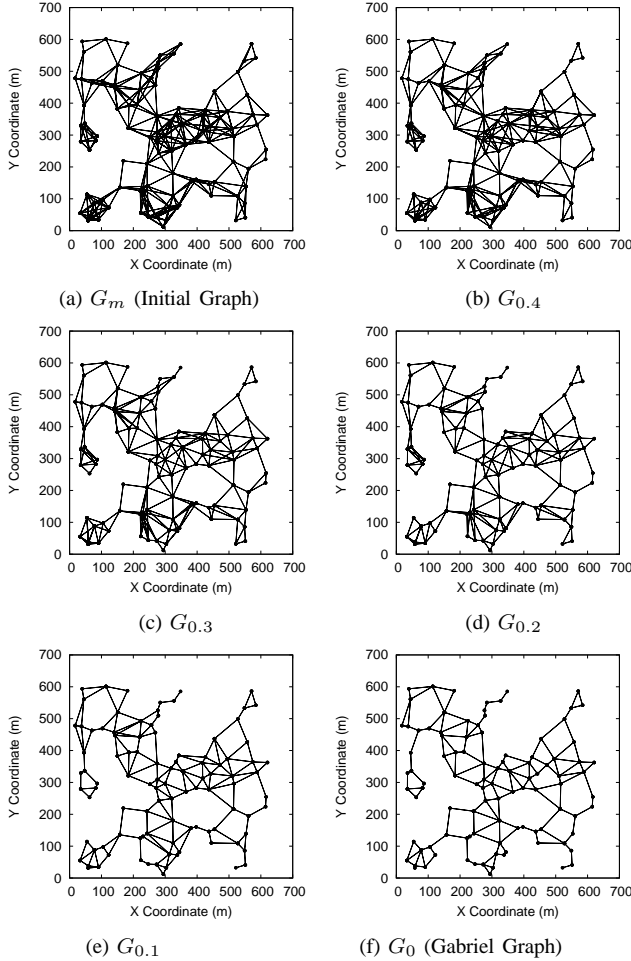


Fig. 1. Δ -graph example for a random network of 100 nodes

values of Δ . Furthermore, all Δ -graph topologies preserve the minimum-energy routing path between every pair of nodes.

Table I provides a detailed statistical summary of the example topologies from Figure 1. For unit link costs, the results show that the sparsest topology $G_{0.0}$ has 60% longer paths (8.02 versus 5.06) than $G_{0.5}$ in terms of hop count, but they are only about 8% longer in distance. Importantly, they save about 15% in energy consumption. For shortest-distance routing on these two topologies, there is again about a 60% difference in hop counts, a small (6%) difference in distance, and slightly larger savings (20%) in energy. For minimum-energy path routing, all of the topologies are the

same. However, the average hop count is 80% higher (8.94 versus 5.06) compared to shortest-hop-path routing on the initial topology $G_{0.5}$, while the energy savings are only 30%.

In the rest of the paper, we present the formal definition of Δ -graphs, analyze their structural properties, and discuss their performance tradeoffs. Section II provides a brief discussion of prior related work. Section III provides a formal definition of the problem, while Section IV presents our main ideas and provides some fundamental bounds. Section V proposes a distributed topology control algorithm, showing that our approach is practical to implement. Section VI develops our analytical model to characterize the structure of Δ -graph topologies, while Section VII presents simulation results to validate our model. Finally, Section VIII concludes the paper. The appendix contains proofs for our main theoretical results.

II. RELATED WORK

Rodoplu and Meng [16] first conceived the idea of minimum energy mobile wireless networks. They allow nodes to set their transmission range much lower than the maximum, while maintaining minimum energy paths between each pair of nodes. Li and Halpern [6] improved their result by developing a low-overhead protocol where nodes start with a very small transmit power and incrementally search for a suitable value until all minimum energy paths are preserved. Li and Wang [8] addressed the algorithmic complexity of the work in [16] and provided an algorithm with lower time complexity.

Cone-Based Topology Control (CBTC), proposed by Wattenhofer *et al.* [20], generates a graph structure similar to a Yao graph. In CBTC, each node determines a power level to reach at least one node in every cone of degree α within its surrounding area. However, the connectivity of the produced subgraph is only guaranteed when $\alpha \leq 5\pi/6$.

Toussaint [19] proposed Relative Neighborhood Graph (RNG) as a sparse connected topology for wireless networks. In RNG, a link $\langle s, t \rangle$ is eliminated if there is a suitably-located *witness* node w satisfying the following condition:

$$\exists w \neq s, t : \max(d(s, w), d(t, w)) < d(s, t)$$

Li *et al.* [7] propose a distributed algorithm (called LMST) based on constructing minimum spanning trees (MST). Instead of building a global MST, which requires global knowledge of the network topology, each node creates its own local MST among its neighbor set. While constructing such LMST, the weight of an edge is set to the transmission power needed between its endpoints. Once the local tree has been

TABLE II
SUMMARY OF MODEL NOTATION

Notation	Description
G_{max}	Graph induced by max transmit power P_{max}
R	Transmission radius
r, s, t, u, v, w	Nodes in a wireless ad hoc network
(s, t)	A node pair
$\langle s, t \rangle$	An edge between s and t
$d(s, t)$	Distance between a node pair (s, t)
α	Path loss factor
$P_{s \rightarrow t}$	Transmission power required from s to t
$\mathcal{R}_{s \rightarrow r}$	Relay region of (s, r) node pair
$C_{(s,t)}$	Cover region for transmission from s to t
$\mathcal{L}^{\mathcal{R}}_{s \rightarrow r}(\Delta)$	Contour line within relay region
$\mathcal{L}^{\mathcal{C}}_{s \rightarrow t}(\Delta)$	Contour line within cover region
$\mathcal{S}^{\mathcal{R}}_{s \rightarrow r}(\Delta_1, \Delta_2)$	Contour strip within relay region
$\mathcal{S}^{\mathcal{C}}_{s \rightarrow t}(\Delta_1, \Delta_2)$	Contour strip within cover region
$C_{(s,t,\Delta)}$	Δ -cover region for transmission from s to t
$\xi_{(s,t,\Delta)}$	Δ -cover set for transmission from s to t
G_{Δ}	Δ -graph
\mathcal{F}_{elim}	Fraction of neighbors eliminated

constructed, each node contributes to the final topology those nodes that are its neighbors in the local MST. However the resulting graph does not preserve the minimum-energy paths.

There is recent research interest in topology control as a means of interference reduction. The natural assumption that a sparse topology implies low interference was first refuted by Bukhart *et al.* [2]. They provide an intuitive definition of interference and propose algorithms to construct connected subgraphs and spanners with minimum interference. Unfortunately, their solution does not preserve the minimum-energy paths between node pairs. Another algorithm, with the same drawback, is presented in [10] and aims at minimizing the average or maximum interference (per link or node). Tang *et al.* [18] propose an interference-aware algorithm for multi-channel mesh networks based on IEEE 802.11. These issues are beyond the scope of our paper. Although an algorithm that adjusts node degree to tradeoff energy is proposed in [5], there is no guarantee that the derived structure would preserve minimum energy paths. Moreover, no mathematical expression for estimating topology size is provided.

III. PROBLEM STATEMENT AND ASSUMPTIONS

In this section, we formulate our problem definition, and lay the groundwork for our analysis. Table II provides a summary of the notational conventions used in our analysis.

A. Problem statement

Any n -node, multi-hop, wireless network can be modeled as a graph $G_{max} = (V, E_{max})$ with the vertex set V representing the nodes, and the edge set E_{max} defined as follows:

$$E_{max} = \{\langle s, t \rangle \mid \langle s, t \rangle \in V \times V \wedge s \neq t \wedge d(s, t) \leq R_{max}\}$$

where R_{max} is the distance beyond which the maximum transmission power P_{max} decays too much to be correctly decoded. The graph G_{max} defined this way is called the *maximum powered network*. In this paper, we solve the following topology control problem:

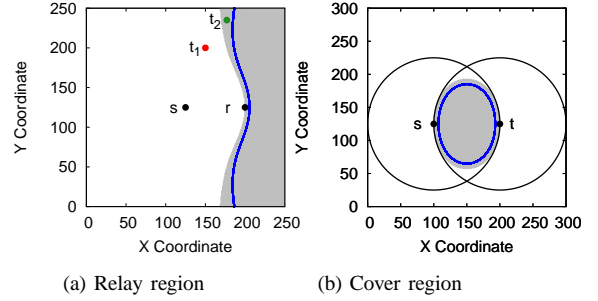


Fig. 2. Relay region, cover region and contour lines ($\alpha = 4$)

Problem 3.1: Given an initial topology G_{max} , provide a distributed algorithm based on local information only to construct a *connected* subgraph $G_{\Delta} = (V, E_{\Delta})$ parameterized by Δ , where an edge $\langle s, t \rangle \in E_{\Delta}$ iff there does not exist any relay node r such that relaying through r saves at least Δ fraction of the power required to send directly from s to t .

B. Power model and assumptions

To send a message from node s to node t at distance $d(s, t)$, the minimum transmission power required is $P_{s \rightarrow t} = Kd^{\alpha}(s, t)$ where $\alpha \in [2, 4]$ is the path loss factor, and K is a global constant [15]. We assume that network links are symmetric, and that the power required for processing and receiving the signal are negligible. We also assume that each node is aware of its own position with reasonable accuracy, perhaps via GPS. The minimum power required using a node r as relay is: $P_{s \rightarrow r \rightarrow t} = P_{s \rightarrow r} + P_{r \rightarrow t}$. The benefit of using a relay r can easily be quantified by measuring the *fraction of power saved* (which is denoted by Δ throughout the paper):

$$\Delta = \frac{P_{s \rightarrow t} - P_{s \rightarrow r \rightarrow t}}{P_{s \rightarrow t}} = 1 - \frac{P_{s \rightarrow r \rightarrow t}}{P_{s \rightarrow t}}$$

C. Relay region

Rodoplu and Meng [16] devised a topology control protocol based on the notion of a *relay region*. Given a node s and another node r within s 's communication range, the relay region $\mathcal{R}_{s \rightarrow r}$ of node r is the set of points such that relaying through r to any point in $\mathcal{R}_{s \rightarrow r}$ takes less energy than a direct transmission to that point. The shaded area of Figure 2(a) shows $\mathcal{R}_{s \rightarrow r}$ for $\alpha = 4$. For example, s can save power by relaying through r to send messages to t_2 (the green node). For t_1 (the red node), however, it is more efficient for s to send directly, since t_1 is outside of the relay region $\mathcal{R}_{s \rightarrow r}$.

D. Cover region

The algorithm presented by Rahman *et al.* [14] is based on a notion of *cover region*. Consider a pair of nodes (s, t) such that the target t lies within the communication range of the source s . Envision the set of all points that can possibly act as relays between s and t such that it would be more power efficient for s to use an intermediate relay node r instead of sending directly to t . This set of points collectively forms the cover region of (s, t) pair. Mathematically:

$$C_{(s,t)} = \{\langle x, y \rangle \mid P_{s \rightarrow \langle x, y \rangle \rightarrow t} \leq P_{s \rightarrow t}\}$$

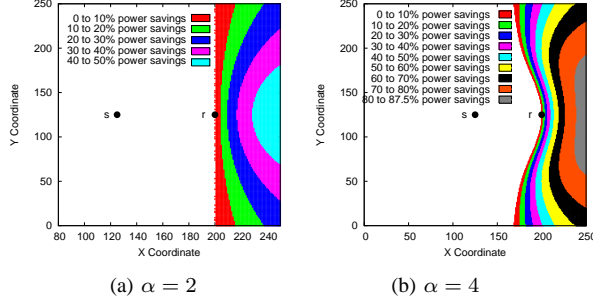


Fig. 3. Contour strips inside relay regions

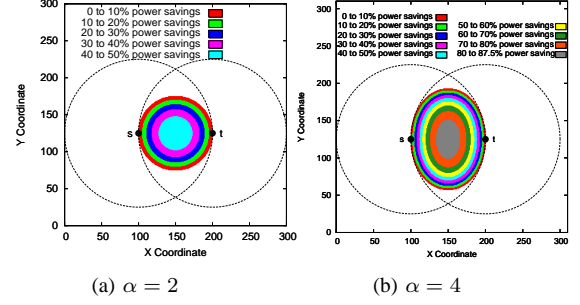


Fig. 4. Contour strips inside cover regions

Here we use $\langle x, y \rangle$ to denote a hypothetical node located at position $\langle x, y \rangle$. The shaded region of Figure 2(b) shows $C_{(s,t)}$ for $\alpha = 4$. Using any node within the shaded area as a relay node saves energy when sending from s to t (or t to s).

IV. Δ -GRAPH ANALYSIS

In this section, we develop and formalize the main idea of the paper. We start with outlining some definitions required for the development of the algorithm, followed by some useful properties of the topology generated by the algorithm.

A. Contour lines and Contour strips

Consider the set of hypothetical points within the relay region for which a Δ fraction of the transmission power is saved when r is used as a relay instead of sending directly to any of those points from s . Such points collectively define a *contour line* inside the relay region, denoted by $\mathcal{L}_{s \rightarrow r}^{\mathcal{R}}(\Delta)$. For example, the blue curve in Figure 2(a) is a contour line for $\Delta = 0.25$ (i.e., 25% power is saved when r is used for relaying from s to any points on the curve). Formally:

$$\mathcal{L}_{s \rightarrow r}^{\mathcal{R}}(\Delta) = \{ \langle x, y \rangle \mid \frac{P_{s \rightarrow r \rightarrow \langle x, y \rangle}}{P_{s \rightarrow \langle x, y \rangle}} = (1 - \Delta) \}$$

A set of consecutive contour lines collectively forms a *contour strip* within a relay region. Any contour strip is parameterized by the interval $[\Delta_1, \Delta_2]$ where Δ_1 and Δ_2 are possible values of Δ . The beginning and the end of the strip are defined by the contour lines $\mathcal{L}_{s \rightarrow r}^{\mathcal{R}}(\Delta_1)$ and $\mathcal{L}_{s \rightarrow r}^{\mathcal{R}}(\Delta_2)$, respectively. Thus a contour strip shows a power saving between $[\Delta_1, \Delta_2]$ fraction of the direct transmission power. Mathematically:

$$\mathcal{S}_{s \rightarrow r}^{\mathcal{R}}(\Delta_1, \Delta_2) = \{ \langle x, y \rangle \mid (1 - \Delta_2) < \frac{P_{s \rightarrow r \rightarrow \langle x, y \rangle}}{P_{s \rightarrow \langle x, y \rangle}} \leq (1 - \Delta_1) \}$$

Figure 3 shows the contour strips within the relay region for $\alpha = 2$ and $\alpha = 4$ respectively. If a node t is located anywhere in the green strip of the relay region, for example, s can save somewhere between 10 – 20% of power by using r as a relay instead of sending directly to t .

Contour lines and strips can be defined similarly for cover regions. Consider two nodes s and t as shown in Figure 4. A contour line within their cover region is:

$$\mathcal{L}_{s \rightarrow t}^{\mathcal{C}}(\Delta) = \{ \langle x, y \rangle \mid \frac{P_{s \rightarrow \langle x, y \rangle \rightarrow t}}{P_{s \rightarrow t}} = (1 - \Delta) \}$$

and the contour strip for the interval $[\Delta_1, \Delta_2]$ is:

$$\mathcal{S}_{s \rightarrow t}^{\mathcal{C}}(\Delta_1, \Delta_2) = \{ \langle x, y \rangle \mid (1 - \Delta_2) < \frac{P_{s \rightarrow \langle x, y \rangle \rightarrow t}}{P_{s \rightarrow t}} \leq (1 - \Delta_1) \}$$

The blue line within the cover region of Figure 2(b) is a contour line with a 25% power savings. Figure 4 shows contour strips within cover regions.

Let us examine the maximum attainable value of Δ , which we call Δ_{max} . It is easy to show that Δ_{max} depends on α .

Lemma 4.1: Bounded Power Savings: The maximum possible power savings using a single relay is $\Delta_{max} = 1 - \frac{1}{2^{\alpha-1}}$.

Corollary 4.1: For $\alpha = 2$, $\Delta_{max} = 0.5$, and for $\alpha = 4$, $\Delta_{max} = 0.875$.

B. Δ -cover regions and Δ -cover sets

When the second parameter Δ_2 of a contour strip $\mathcal{S}_{s \rightarrow t}^{\mathcal{C}}(\Delta_1, \Delta_2)$ is set to Δ_{max} , then the strip provides all possible locations between s and t that can save *at least* Δ_1 fraction of power compared to direct communication between s and t . Such a contour strip now depends on a single parameter Δ (replacing Δ_1) and will be called *Δ -cover regions*. Formally Δ -cover regions and Δ -cover sets are described by the following definitions:

Definition 4.1: Consider the set of all points that can possibly act as relays between s and t such that it would save at least Δ fraction of power for s to use a node located at one of those points instead of sending directly to t . This set of points collectively forms the Δ -cover region of (s, t) pair:

$$C_{(s,t,\Delta)} = \{ \langle x, y \rangle \mid \frac{P_{s \rightarrow \langle x, y \rangle \rightarrow t}}{P_{s \rightarrow t}} \leq 1 - \Delta \}$$

The collection of all nodes falling within the Δ -cover region of s and t is called the Δ -cover set of s and t :

$$\xi_{(s,t,\Delta)} = \{ v \mid v \in V \wedge Location(v) \in C_{(s,t,\Delta)} \}$$

Figure 5 shows the Δ -cover regions for different values of Δ . For $\alpha = 2$, all Δ -cover regions are circular regions centered at the midpoint connecting s and t . The circular region shrinks to a point as we increase the value of Δ to Δ_{max} .

The next three lemmas establish some basic properties of Δ -cover sets and Δ -cover regions:

Lemma 4.2: Properties of Δ -Cover Sets: (a) For any $r \in \xi_{(s,t,\Delta)}$, $d(s, r) < d(s, t)$. (b) If $r \in \xi_{(s,t,\Delta)}$, then $t \notin \xi_{(s,r,\Delta)}$.

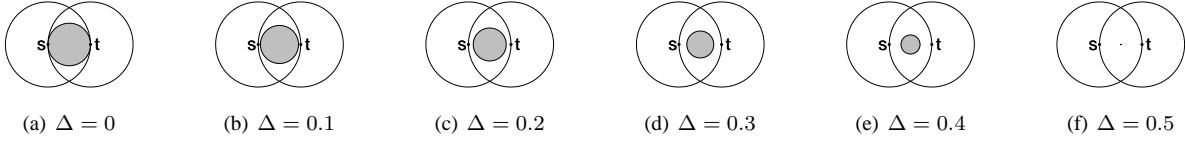


Fig. 5. Δ -cover regions for $\alpha = 2$

Lemma 4.3: Circle Property of Δ -Cover Regions: Suppose the distance between s and t is d . For $\alpha = 2$, the Δ -cover regions are circular regions centered at the midpoint on the straight line connecting s and t , and have radius $\sqrt{(1-2\Delta)}d/2$.

Lemma 4.4: Relationship among Δ -Cover Regions: For any α and a pair of nodes (s, t) , if $\Delta_1 > \Delta_2$, then $C_{(s,t,\Delta_1)} \subset C_{(s,t,\Delta_2)}$ and $\xi_{(s,t,\Delta_1)} \subset \xi_{(s,t,\Delta_2)}$, where $0 \leq \Delta_2 < \Delta_1 \leq \Delta_{max}$.

Note that, according to Lemma 4.3, for $\alpha = 2$ both $C_{(s,t,\Delta_1)}$ and $C_{(s,t,\Delta_2)}$ are regions confined within the circles centered at $(d/2, 0)$ with radius $R_1 = \sqrt{(1-2\Delta_1)}d/2$ and $R_2 = \sqrt{(1-2\Delta_2)}d/2$. If $\Delta_1 > \Delta_2$ then $R_1 < R_2$ and $C_{(s,t,\Delta_1)} \subset C_{(s,t,\Delta_2)}$.

C. Δ -graphs

The Δ -graph, denoted by $G_\Delta = (V, E_\Delta)$, is a subgraph of G_{max} where $V(G_\Delta) = V(G_{max})$ and $E_\Delta \subseteq E_{max}$. Formally, an edge $\langle s, t \rangle$ in G_{max} also exists in G_Δ if there is no node r in the Δ -cover region of (s, t) pair: $E_\Delta = \{\langle s, t \rangle | s \neq t \wedge \xi_{(s,t,\Delta)} = \emptyset\}$. Once the Δ -cover regions are known it is very easy to construct G_Δ graphs. For example, during construction of the graph $G_{0.3}$, the edge $\langle s, t \rangle$ will be eliminated if there exists at least one node r in the shaded circle of Figure 5(d).

The following lemma shows an interesting result about G_0 .

Lemma 4.5: Relationship with Gabriel Graph: For $\alpha = 2$, the graph G_0 is a Gabriel Graph.

Connectivity is an important property for any topology control algorithm. With the help of the subsequent lemmas, we show that all G_Δ graphs are connected.

Lemma 4.6: Connectivity of G_0 : For any α , if G_{max} is connected, then the subgraph G_0 is also connected.

Lemma 4.7: Δ -Graph Relationship: For any α , if $0 \leq \Delta_2 < \Delta_1 \leq \Delta_{max}$, then $G_{\Delta_2} \subset G_{\Delta_1}$.

Combining Lemma 4.6 and Lemma 4.7 yields:

Theorem 4.1: Δ -Graph Connectivity: If G_{max} is connected, then G_Δ is connected, for any $0 \leq \Delta \leq \Delta_{max}$ and any α .

V. DISTRIBUTED ALGORITHM FOR CONSTRUCTING G_Δ

In this section, we present a distributed algorithm for constructing G_Δ . The operation is described from the viewpoint of a single node s . For static networks, the construction starts by broadcasting a single neighbor discovery message (NDM) at the maximum power P_{max} . When mobility is considered, NDM needs to be sent periodically at an interval suitably chosen based on the mobility dynamics. All the nodes receiving the NDM from s send back a reply. While s collects the replies

of its neighbors, it learns their identities and locations. It also constructs the Δ -cover sets with those neighbors. Initially, all those sets are empty (s does not even know what neighbors there are). The set A_s , which also starts empty, keeps track of all the nodes discovered in the neighborhood.

Whenever s receives a reply to its NDM from a node v , it performs the steps listed in Algorithm 1. Its purpose is to update the configuration of the Δ -cover sets. At the end, when s has received all the replies, the configuration of Δ -cover sets is complete. The goal of node s is to determine its neighbor set, i.e., the collection of neighbors to which transmission should be direct. Having determined the Δ -cover regions of all its neighbors, s is in position to identify the members of its neighbor set. If $\xi_{(s,t,\Delta)} \neq \emptyset$ for some t , it means that sending directly to node t is not Δ -power efficient, i.e., there is at least one node $r \in \xi_{(s,t,\Delta)}$ that can act as a relay between s and t and save at least Δ fraction of power versus direct transmission. A node t that has an empty Δ -cover set with s has no Δ -power-efficient relays and thus belongs to the neighbor set of s . Consequently, the loop listed in Algorithm 2 completes the algorithm by generating the neighbor set \mathcal{N}_s of s . The transmission power $p(s)$ for the node s is determined as follows. For every unicast packet, node s transmits with the minimum power required to reach the destination. For every neighbor broadcast message, node s uses the power required to reach the furthest neighbor in \mathcal{N}_s . Specifically, $p(s)$ is determined as follows: $p(s) \leftarrow \max \{Kd^\alpha(s, v) : v \in \mathcal{N}_s\}$.

VI. ANALYTICAL MODEL

In this section, we develop a mathematical expression for estimating the expected fraction of neighbors that are eliminated in G_Δ graphs for $\alpha = 2$. Expressions for general values of α are left for future work. The model is developed by augmenting the approach provided in [9] with necessary modifications to accommodate Δ -cover regions.

Consider a multi-hop wireless network with n nodes uniformly distributed over a rectangular region with area A . The average node density is $D = \frac{n}{A}$. The maximum transmission radius of each node is R , which is assumed to be the same for all nodes. Let us observe an arbitrary node s within the deployment area. The average number of nodes located in the communication region of node s is $N_R = \pi R^2 \times D = \frac{\pi n R^2}{A}$.

Let $P_N(x)$ be the probability that there exists a neighbor t at distance x from s . Clearly, $P_N(x) = 0$ for $x > R$. For $x \leq R$, at first consider a small area strip defined by dx at the perimeter of the circle with radius x and centered at s as shown in Figure 6. Also consider a small angle $d\theta$ measured from an arbitrary but fixed axis. The length of the arc $\ell = x d\theta$

Algorithm 1 UpdateCoverSets (s, v, Δ)

for each $w \in A_s$ **do**
 if Location(v) $\in C_{(s,w,\Delta)}$ **then**
 $\xi_{(s,w,\Delta)} = \xi_{(s,w,\Delta)} \cup \{v\}$
 else if Location(w) $\in C_{(s,v,\Delta)}$ **then**
 $\xi_{(s,v,\Delta)} = \xi_{(s,v,\Delta)} \cup \{w\}$
 end if
end for

Algorithm 2 NeighborSet (s, Δ)

$\aleph_s = \emptyset$
for each $t \in A_s$ **do**
 if $\xi_{(s,t,\Delta)} = \emptyset$ **then**
 $\aleph_s = \aleph_s \cup \{t\}$
 end if
end for

and the area of the small region dA within this small strip can be approximated as $dA = \ell dx = x dx d\theta$. Therefore the area of the entire small strip denoted by A_{strip} becomes,

$$A_{strip} = \int_0^{2\pi} dA = \int_0^{2\pi} \ell dx = \int_0^{2\pi} x dx d\theta = 2\pi x dx$$

Thus $P_N(x)$ becomes:

$$P_N(x) = A_{strip} \times D = 2\pi x dx \times D = \frac{2\pi n x dx}{A} \quad (1)$$

Let $P_C(x)$ be the probability that there exists a node in the cover region $C_{(s,t,\Delta)}$ between node pair (s, t) . The distance x between s and t plays an important role in determining the value of $P_C(x)$. For large x , the size of the cover region is also large and the probability of a node's existence within this cover region also becomes large.

The probability $P_E(x)$ of eliminating any node t from the neighbor set of s is the probability that there exists a neighbor t at distance x from s , and there is a node in the Δ -cover region between (s, t) . So $P_E(x)$ is:

$$P_E(x) = P_N(x) \times P_C(x) = \left(\frac{2\pi n x dx}{A} \right) \times P_C(x) \quad (2)$$

The expected number of neighbors eliminated by s from its neighbor set is found by integrating $P_E(x)$ over the transmission radius R within which s possibly can communicate:

$$\mathcal{T}_{elim} = \int_0^R \frac{2\pi n x \times P_C(x) dx}{A} \quad (3)$$

If we divide the expected number of eliminated neighbors by the expected number of nodes in a node's communication area, we get the expected fraction of neighbors that are eliminated from a node's neighbor set:

$$\mathcal{F}_{elim} = \frac{\mathcal{T}_{elim}}{N_R} = \frac{\int_0^R \frac{2\pi n x P_C(x) dx}{A}}{\frac{\pi n R^2}{A}} = \frac{\int_0^R 2\pi n x P_C(x) dx}{\pi n R^2} \quad (4)$$

Let's find the probability that at least one node is located within the Δ -cover region of (s, t) , i.e., $P_C(x)$. Recall from

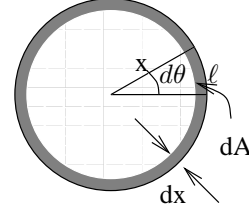


Fig. 6. Illustrating circular strip at distance x

Lemma 4.3 that for $\alpha = 2$, Δ -cover regions (i.e., $C_{(s,t,\Delta)}$) are circular regions with radius $R = \sqrt{(1-2\Delta)}x/2$. Let $\beta = \sqrt{(1-2\Delta)}$ so that $R = \beta x/2$. The probability that a node is placed in the circular area $\frac{\pi\beta^2 x^2}{4}$ within the area A is $P_\Delta = \frac{\frac{\pi\beta^2 x^2}{4}}{A} = \frac{\pi\beta^2 x^2}{4A}$. The probability $P_k(C_{(s,t,\Delta)})$ that exactly k nodes are in $C_{(s,t,\Delta)}$ is:

$$P_k(C_{(s,t,\Delta)}) = \binom{n-2}{k} P_\Delta^k \times (1-P_\Delta)^{n-2-k} \quad (5)$$

Note that $n-2$ is used rather than n because we exclude s and t . For large n and small P_Δ , the binomial distribution can be approximated using Poisson distribution [1] with mean nP_Δ :

$$P_k(C_{(s,t,\Delta)}) = \frac{(nP_\Delta)^k \times e^{-nP_\Delta}}{k!} \quad (6)$$

The probability that there exists at least one node in $C_{(s,t,\Delta)}$:

$$\begin{aligned} P_C(x) &= \sum_{k=1}^n P_k(C_{(s,t,\Delta)}) = \sum_{k=1}^{\infty} \frac{(nP_\Delta)^k \times e^{-nP_\Delta}}{k!} \\ &= e^{-nP_\Delta} \sum_{k=1}^{\infty} \frac{(nP_\Delta)^k}{k!} = e^{-nP_\Delta} \left(\sum_{k=0}^{\infty} \frac{(nP_\Delta)^k}{k!} - 1 \right) \\ &= e^{-nP_\Delta} (e^{nP_\Delta} - 1) = 1 - e^{-nP_\Delta} \\ &= 1 - e^{-\frac{\pi n \beta^2 x^2}{4A}} \quad [\text{substituting the value of } P_\Delta] \end{aligned} \quad (7)$$

By replacing $P_C(x)$ in Equation 4 using Equation 7, we get:

$$\begin{aligned} \mathcal{F}_{elim} &= \frac{\int_0^R 2\pi n x \times P_C(x) dx}{\pi n R^2} \\ &= \frac{\int_0^R 2\pi n x \times \left(1 - e^{-\frac{\pi n \beta^2 x^2}{4A}} \right) dx}{\pi n R^2} \\ &= \frac{\pi n \beta^2 R^2 + 4A e^{-\frac{\pi n \beta^2 R^2}{4A}} - 4A}{\pi n \beta^2 R^2} \end{aligned} \quad (8)$$

Finally, restoring $\beta = \sqrt{(1-2\Delta)}$, we obtain:

$$\mathcal{F}_{elim} = \frac{\pi n (1-2\Delta) R^2 + 4A e^{-\frac{\pi n (1-2\Delta) R^2}{4A}} - 4A}{\pi n (1-2\Delta) R^2} \quad (9)$$

According to Equation 9, the expected fraction of neighbors that are eliminated from a node's neighbor set is a function of four network parameters, namely power savings threshold, number of nodes, deployment area, and maximum transmission radius (i.e., $\mathcal{F}_{elim} = f(\Delta, n, A, R)$). The general effects of these parameters (while changing one and keeping others fixed) are as follows: \mathcal{F}_{elim} increases when either n or R increase, and decreases when either Δ or A is increased.

VII. SIMULATION RESULTS

In this section, we present simulation results to verify the accuracy of our analytical model, and we explore the performance tradeoffs offered by Δ -graphs.

A. Simulation environment and performance metrics

To evaluate the performance, we simulate randomly deployed networks of 100, 300, and 500 nodes uniformly distributed over a $625\text{ m} \times 625\text{ m}$ square region. The maximum transmission range is limited between 100 m to 250 m . We only consider connected networks, since it is not possible to generate a connected subnetwork unless the initial network G_{max} is connected. We use $\alpha = 2$ for all scenarios.

We have generated 10 instances of networks for each network size. Performance measures are reported as an average of these 10 random scenarios (unless explicitly stated otherwise). We analyze the performance of the different Δ -graphs using two performance criteria: sparseness, and fault tolerance.

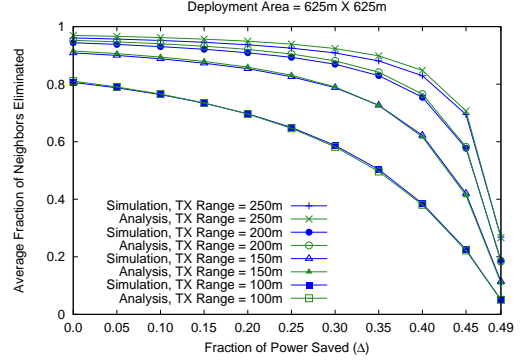
Sparseness. The number of links remaining in a subgraph G_Δ determines its sparseness. The degree of sparseness of a network affects routing performance of the routing layer. For example, if flooding is used for route discovery, it may cause serious broadcast storm problems [11] in a dense graph, such as G_{max} . By reducing the neighbor set of each node, this problem can be mitigated to some extent. We determine the sparseness of a network by measuring the average fraction of neighbors that are eliminated, i.e., \mathcal{F}_{elim} . Larger values of \mathcal{F}_{elim} indicate more sparseness. We run the proposed algorithm on a given G_{max} by setting different values of Δ and measure \mathcal{F}_{elim} for each subnetwork G_Δ .

Fault tolerance. For each of the communication subnetworks G_Δ , we measure the degree of connectivity to assess fault tolerance. A (sub)network G_Δ is called k -connected if there exists at least k vertex-disjoint paths between all pairs of nodes. A large value of k generally indicates a high level of fault resilience.

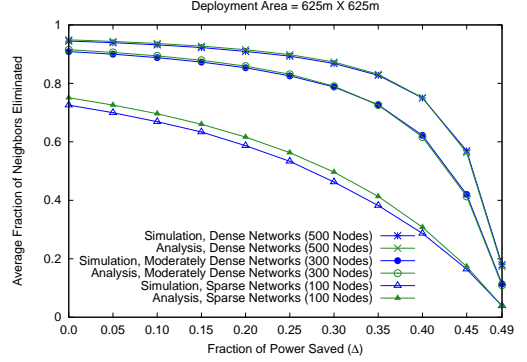
B. Performance evaluation

1) *Sparseness of G_Δ graphs:* To see the effect of Δ on sparseness, we return to the example in Figure 1(a), which shows a typical random deployment of 100 nodes, with each node having a maximum communication range of 100 m . This is the initial graph G_{max} for our algorithm. The remaining parts of Figure 1 show G_Δ for $\Delta = \{0, 0.1, 0.2, 0.3, 0.4\}$. G_0 is the Gabriel graph (see Lemma 4.5). All five subgraphs have considerably fewer links and lower average node degree compared to G_{max} (see Table I). For larger values of Δ , more links are retained, since the condition for eliminating neighbors with long-distance links is more stringent. For example, to eliminate a link $\langle s, t \rangle$ in $G_{0.1}$, there must exist a relay between s and t that can save at least 10% power, whereas this requirement is 30% savings for $G_{0.3}$. Therefore, the long-distance links may be pruned in $G_{0.1}$, but survive in $G_{0.3}$.

The sparseness of different G_Δ graphs is compared under different parameter settings in Figure 7. Two parameters were varied: maximum transmission range in Figure 7(a), and



(a) TX range was varied (300 nodes)



(b) Node density was varied (TX range = 150 m)

Fig. 7. Expected Fraction of Neighbors Eliminated

number of nodes in Figure 7(b). In each of the plots, we show the values of \mathcal{F}_{elim} found during the simulation and calculated from the analytical model.

In Figure 7(a), we deployed 300 nodes and varied the maximum transmission range (R). Every node has the same R . \mathcal{F}_{elim} slowly decreases at the beginning of each curve and drastically drops at the end. For large values of Δ , only a few longer links were eliminated, since the rule for elimination becomes harder to satisfy with large Δ . Moreover, more neighbors are pruned in networks with higher R , since the chance of elimination for a neighbor is also higher.

Figure 7(b) shows the effect of node density on neighbor elimination by varying the number of nodes (100, 300, and 500) within the same deployment region. R is kept fixed at 150 m for all scenarios. The result is as expected: a higher fraction of neighbors is eliminated in more dense networks for all values of Δ .

Figure 7(a) and 7(b) also show that for all scenarios, the results of analysis are very close to the simulation results, with a difference of at most 3%. The small inaccuracy arises from nodes located close to the boundaries of the simulation region, for which the communication area is restricted, and thus they have fewer neighbors. We ignored this “boundary effect” to simplify our analytical model.

2) *Fault tolerance:* We next focus on the benefit of having more neighbors. To see the benefit, we determine the number of vertex-disjoint (VD) paths between all pairs of nodes in

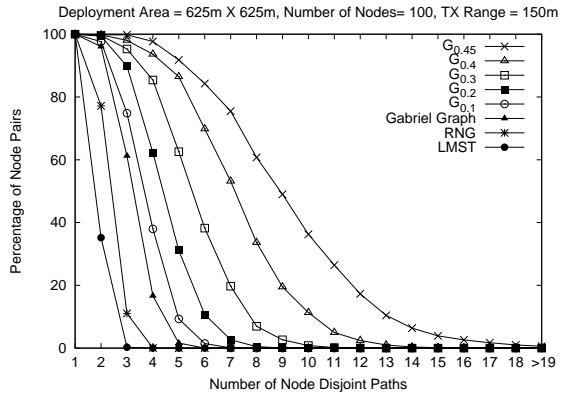


Fig. 8. Connectivity on a network of 100 nodes, TX Range = 150m

each subnetwork G_{Δ} . R is set to 150m (the trends are similar for other values of R). Figure 8 shows the percentage of node pairs versus the number of VD paths between them. All G_{Δ} subgraphs are connected because the plots for all Δ -graphs show 100% of node pairs having at least one VD path. While there is a small (albeit visible) difference in the percentage of node pairs having at least two or three VD paths in different G_{Δ} subnetworks, the difference is much larger for even more VD paths. Only 18% of node pairs have four VD paths in the Gabriel graph, but this percentage grows to 38%, 62%, 83%, 90%, and 98% for $\Delta = 0.1, 0.2, 0.3, 0.4$, and 0.45, respectively. Clearly, larger values of Δ add more fault-resilience to the network by providing auxiliary paths.

C. Comparison with traditional topologies

Sparseness. Δ -graphs can be compared to some traditional topologies such as Minimum Spanning Tree (MST), Local Minimum Spanning Tree (LMST), Relative Neighborhood Graphs (RNG) and Gabriel Graphs (GG) in terms of sparsity. Prior work [12], [19] has shown that $MST \subseteq RNG \subseteq GG$. Cartigny *et al.* [3] show that $LMST \subseteq RNG$. It is also known that LMST contains MST (see [13]). By combining Lemma 4.5 and 4.7, we can deduce that $GG \subseteq G_{\Delta}$, for $\Delta \geq 0$. Combining all these results we can infer that $MST \subseteq LMST \subseteq RNG \subseteq GG \subseteq G_{\Delta} \subseteq G_{max}$ and consequently,

$$n - 1 \leq |LMST| \leq |RNG| \leq |GG| \leq |G_{\Delta}| \leq |G_{max}|$$

(the lower bound is the size of MST; n is the number of nodes). Moreover, only GG and G_{Δ} graphs in the above series preserve minimum energy paths between all pairs of nodes.

Fault Tolerance. Figure 8 shows how the network connectivity varies among the foregoing topologies. Although all of the topologies guarantee 1-connectivity (i.e., 100% node pairs have at least one VD path in the plots), a majority of node pairs have 2-connectivity ($> 96\%$) in Δ -graphs. On the other hand, only about 36% of node pairs are 2-connected in LMST or 76% in RNG. LMST and RNG have less than 10% of node pairs that are 3-connected, but Δ -graphs have a rich proportion of such nodes ($> 60\%$). Clearly, sparsity and fault tolerance create an interesting tradeoff; sparser networks have

lower fault-tolerance, and all Δ -graphs provide a higher level of fault-tolerance compared to other traditional topologies.

VIII. CONCLUSIONS

Topology control in wireless ad hoc networks introduces many challenges due to multiple conflicting performance optimization criteria. Solutions optimizing one performance criteria are detrimental to others. This paper tackles this issue by providing an algorithm that generates a class of topologies and provides the flexibility to network designers to choose the appropriate topology with desired performance characteristics.

Our Δ -graph approach has three main advantages. First, the algorithm is simple, practically implementable, distributed, and strictly local. Second, unlike many other traditional topology control algorithms, our algorithm is extremely flexible, while still preserving minimum-energy paths. Third, the structural properties of the topologies generated by the algorithm can be calculated analytically, perhaps prior to network deployment. These advantages make our proposed algorithm appealing for deploying flexible ad hoc networks. Ongoing work is extending our approach to mobile ad hoc networks. At the same time, we are evaluating Δ -graphs for other performance characteristics, such as interference, packet delivery ratio (PDR), and throughput, with the help of packet-level simulations.

REFERENCES

- [1] Poisson distribution. http://en.wikipedia.org/wiki/Poisson_distribution.
- [2] M. Burkhart, P. Rickenbach, R. Wattenhofer, and A. Zollinger. Does topology control reduce interference? In *MobiHoc*, 2004.
- [3] J. Cartigny, F. Ingelrest, D. Simplot-Ryl, and I. Stojmenovic. Localized LMST and RNG based minimum-energy broadcast protocols in ad hoc networks. *Ad Hoc Networks*, pages 1–16, 2005.
- [4] F. Che, E. Lloyd, and L. Zhao. Improved topology control algorithms for simple mobile networks. *Ad hoc Networks*, 28(1), 2010.
- [5] A. Jeng and R. Jan. The r-neighborhood graph: An adjustable structure for topology control in wireless ad hoc networks. *IEEE Trans. Parallel Distrib. Syst.*, 18(4):536–549, April 2007.
- [6] L. Li and J. Halpern. Minimum energy mobile wireless networks revisited. In *Proceedings of IEEE ICC*, 2001.
- [7] N. Li, J. Hou, and L. Sha. Design and analysis of an MST-based topology control algorithm. In *Proceedings of IEEE INFOCOM*, 2003.
- [8] X. Li and P. Wan. Constructing minimum energy mobile wireless networks. In *Proceedings of ACM MobiHoc*, pages 55–67, 2001.
- [9] B. Milic and M. Malek. Dropped edges and faces' size in gabriel and relative neighborhood graphs. *IEEE International Conference on Mobile Adhoc and Sensor Systems Conference*, 0:407–416, 2006.
- [10] K. Moaveninejad and X. Li. Low-interference topology control for wireless ad hoc networks. *AHSN*, 1(1-2), 2005.
- [11] S. Ni, Y. Tseng, Y. Chen, and J. Sheu. The broadcast storm problem in a mobile ad hoc network. In *Proceedings of ACM MobiCom*, 1999.
- [12] A. Okabe, B. Boots, and K. Sugihara. Spatial tessellations: Concepts, and applications of voronoi diagrams. *John Wiley & Sons*, 1992.
- [13] F. Ovalle-Martinez, I. Stojmenovic, F. Nocetti, and J. Solano-Gonzalez. Finding minimum transmission radii for preserving connectivity and constructing minimal spanning trees in ad hoc and sensor networks. *Journal of Parallel and Distributed Computing*, 65:132–141, 2005.
- [14] A. Rahman and P. Gburzynski. MAC-assisted topology control for ad-hoc wireless networks. *Int. Journal of Comm. Sys.*, 19(9):955–976, 2006.
- [15] T. Rappaport. Wireless communications: principles and practice, 1996.
- [16] V. Rodoplu and T. Meng. Minimum energy mobile wireless networks. *IEEE Journal of Sel. Areas in Comm. (JSAC)*, 17(8):1333–1344, 1999.
- [17] I. Stojmenovic and X. Lin. Power-aware localized routing in wireless networks. *IEEE Trans. Parallel Distrib. Syst.*, 12:1122–1133, 2001.

- [18] J. Tang, G. Xue, and W. Zhang. Interference-aware topology control and QoS routing in multi-channel wireless mesh networks. In *Proceedings of ACM MobiHoc*, pages 68–77, 2005.
- [19] G. Toussaint. The relative neighborhood graph of a finite planar set. *Pattern Recognition*, 12:261–268, 1980.
- [20] R. Wattenhofer, L. Li, P. Bahl, and Y. Wang. Distributed topology control for wireless multihop ad-hoc networks. In *IEEE INFOCOM*, 2001.
- [21] R. Wattenhofer and A. Zollinger. XTC: A practical topology control algorithm for ad-hoc networks. In *4th WMAN*, April 2004.

APPENDIX

This appendix contains the formal proofs for the theoretical results claimed in the paper.

Lemma 4.1. Bounded Power Savings.

Proof: Consider a relay node r between a node pair (s, t) . Assume that s and t are within their mutual transmission ranges. Although r can be placed at arbitrary positions, placing r at the midpoint on the straight line connecting s and t always produces the best power savings [17]. Now let us consider the case where s , r and t are placed on a straight line. Let $d(s, t) = d$, $d(s, r) = x$, and $d(r, t) = d - x$. Therefore, $P_{s \rightarrow t} = Kd^\alpha$, and $P_{s \rightarrow r \rightarrow t} = Kx^\alpha + K(d - x)^\alpha$. Energy is saved if $P_{s \rightarrow r \rightarrow t} < P_{s \rightarrow t}$ and the fraction of power savings becomes a function of distance x :

$$\begin{aligned} \Delta = f(x) &= \frac{P_{s \rightarrow t} - P_{s \rightarrow r \rightarrow t}}{P_{s \rightarrow t}} = 1 - \frac{P_{s \rightarrow r \rightarrow t}}{P_{s \rightarrow t}} \\ \Leftrightarrow \Delta = f(x) &= 1 - \frac{x^\alpha + (d - x)^\alpha}{d^\alpha} \end{aligned}$$

The maximum of $f(x)$ is obtained for $f'(x) = 0$, i.e.,

$$f'(x) = -\frac{1}{d^\alpha} [\alpha x^{\alpha-1} - \alpha(d - x)^{\alpha-1}] = 0$$

Thus $x^{\alpha-1} = (d - x)^{\alpha-1}$, or $x = d - x$ or $x = d/2$. So the maximum power savings is achieved when the relay node is placed at the midpoint of the straight line connecting s and t . The power savings there are:

$$\Delta_{max} = f\left(\frac{d}{2}\right) = 1 - \frac{\left(\frac{d}{2}\right)^\alpha + \left(d - \frac{d}{2}\right)^\alpha}{d^\alpha} = 1 - \frac{1}{2^{\alpha-1}}$$

Lemma 4.2. Properties of Δ -Cover Sets.

Proof: (a) If $r \in \xi_{(s,t,\Delta)}$, then from Definition 4.1 we get, $P_{s \rightarrow r \rightarrow t} \leq (1 - \Delta)P_{s \rightarrow t}$. Since $\Delta \geq 0$, it follows that $P_{s \rightarrow r \rightarrow t} \leq P_{s \rightarrow t}$. Replacing $P_{s \rightarrow r \rightarrow t}$ and $P_{s \rightarrow t}$ by distances (using the path loss model) we get, $d^\alpha(s, r) + d^\alpha(r, t) \leq d^\alpha(s, t)$. Now for $\alpha \geq 1$, $d(s, t) > d(s, r)$.

(b) If $r \in \xi_{(s,t,\Delta)}$ then from (a), $d(s, t) > d(s, r)$. Now suppose $t \in \xi_{(s,r,\Delta)}$ then again from (a), $d(s, r) > d(s, t)$ which is a contradiction. ■

Lemma 4.3. Circle Property of Δ -Cover Regions.

Proof: Without loss of generality, let us assume that s and t are located at $(0, 0)$ and $(d, 0)$. Consider a hypothetical relay node r within the Δ -cover region of node pair (s, t) positioned at $\langle x, y \rangle$. Here Δ is a real number satisfying $0 \leq \Delta \leq 0.5$. From Definition 4.1, the fraction of power saved when r is used as a relay must be at least Δ , therefore:

$$\begin{aligned} \Delta &\leq \frac{P_{s \rightarrow t} - P_{s \rightarrow r \rightarrow t}}{P_{s \rightarrow t}} = 1 - \frac{P_{s \rightarrow r \rightarrow t}}{P_{s \rightarrow t}} \\ \Leftrightarrow \Delta &\leq 1 - \frac{x^2 + y^2 + (d - x)^2 + y^2}{d^2} \end{aligned}$$

After some easy manipulations, the expression becomes:

$$(x - \frac{d}{2})^2 + y^2 \leq (1 - 2\Delta)\left(\frac{d}{2}\right)^2 = (\sqrt{(1 - 2\Delta)}d/2)^2$$

which is an equation of a circle centered at $(d/2, 0)$, and with radius $\sqrt{(1 - 2\Delta)}d/2$. Thus any relay node providing at least Δ fraction power savings must fall within this circle. ■

Lemma 4.4. Relationship among Δ -Cover Regions.

Proof: Consider a point $\langle x, y \rangle \in C_{(s,t,\Delta_1)}$. From the definition of Δ -cover regions we get,

$$P_{s \rightarrow \langle x, y \rangle \rightarrow t} \leq (1 - \Delta)P_{s \rightarrow t} \quad \text{for } \alpha \geq 2$$

Since $\Delta_1 > \Delta_2$, we get, $(1 - \Delta_1) < (1 - \Delta_2)$ and the above equation becomes:

$$P_{s \rightarrow \langle x, y \rangle \rightarrow t} \leq (1 - \Delta_1)P_{s \rightarrow t} \leq (1 - \Delta_2)P_{s \rightarrow t}$$

which implies $\langle x, y \rangle \in C_{(s,t,\Delta_2)}$.

The nodes actually present within the smaller region $C_{(s,t,\Delta_1)}$ must also be contained within the larger region defined by $C_{(s,t,\Delta_2)}$. Therefore, $\xi_{(s,t,\Delta_1)} \subset \xi_{(s,t,\Delta_2)}$. ■

Lemma 4.5. Relationship with Gabriel Graph.

Proof: In G_Δ , an edge $\langle s, t \rangle$ is eliminated if for any triplet $\langle s, r, t \rangle$ it holds that $P_{s \rightarrow r \rightarrow t} \leq (1 - \Delta)P_{s \rightarrow t}$. Putting $\Delta = 0$ the condition becomes, $P_{s \rightarrow r \rightarrow t} \leq P_{s \rightarrow t}$, or $d^2(s, r) + d^2(r, t) \leq d^2(s, t)$, which is the condition for elimination of an edge in Gabriel graph [9]. Thus G_0 is a Gabriel graph. ■

Lemma 4.6. Connectivity of G_0 .

Proof: First, we show that $G_0 = (V, E_0)$ preserves all minimum energy paths in $G_{max} = (V, E_{max})$ between every pair of nodes. Suppose that the cost of an edge $\langle s, t \rangle$ is $\gamma(\langle s, t \rangle) = P_{s \rightarrow t}$. Since a path is defined by consecutive edges, the cost of a path $\pi = \langle s_0, s_1, \dots, s_{n+1} \rangle$ becomes $\gamma(\pi) = \sum_{i=0}^n \gamma(\langle s_i, s_{i+1} \rangle)$. Suppose, by way of contradiction, there exist nodes s, t and a simple path δ in G_{max} such that $\gamma(\delta) < \gamma(\delta')$ for any simple path δ' from s to t in G_0 . Let $\delta = \langle s_0, s_1, \dots, s_{n+1} \rangle$ with $s_0 = s$ and $s_{n+1} = t$. Then for all $i = 0, 1, \dots, n$, it must be true that $\langle s_i, s_{i+1} \rangle \in E_0$. Otherwise, there exists a node r in the Δ -cover region of node pair (s_i, s_{i+1}) . But then $\gamma(\langle s_i, r, s_{i+1} \rangle) < \gamma(\langle s_i, s_{i+1} \rangle)$ and the path δ can be replaced by a least cost path by replacing the edge $\langle s_i, s_{i+1} \rangle$ with $\langle s_i, r, s_{i+1} \rangle$, which is a contradiction. So G_0 preserves all minimum energy paths in G_{max} between all pairs of nodes. Thus it immediately follows that G_0 is connected providing at least the minimum energy paths between all pairs of nodes. ■

Lemma 4.7. Δ -Graph Relationship.

Proof: Suppose the edge sets of G_{Δ_1} and G_{Δ_2} are E_{Δ_1} and E_{Δ_2} respectively. Then it is sufficient to show that if an edge $\langle s, t \rangle \in E_{\Delta_2}$ then it must be the case that $\langle s, t \rangle \in E_{\Delta_1}$. Since $\langle s, t \rangle \in E_{\Delta_2}$, there exists no node r inside $C_{(s,t,\Delta_2)}$. Since $\Delta_1 > \Delta_2$, according to Lemma 4.4, $C_{(s,t,\Delta_1)} \subset C_{(s,t,\Delta_2)}$. Therefore there is no node r inside the $C_{(s,t,\Delta_1)}$ and $\langle s, t \rangle \in E_{\Delta_1}$. ■