Public Review for

A Shortest-Path-Based Topology Control Algorithm in Wireless Multihop Networks

Yao Shen, Yunze Cai, & Xiaoming Xu

This paper proposes a new topology control algorithm for multihop wireless networks. The authors propose the use of shortest path algorithm for topology control, where the weight of each edge in the network reflects the energy consumption required for transmissions along that link. The topology is then constructed in a way that neighbors are those nodes that participate in the minimum cost paths of each node in the network. The advantage of the proposed algorithm is that it is very simple to implement, relies on local knowledge (energy required for transmission to neighbors), and that it does not rely on specific energy models, thus being able to incorporate heterogeneity in energy consumption across the network. Having studied the state of the art in the area of topology control in ad hoc networks, I find that this work is interesting in that it proposes a simple solution that does not require prior knowledge of the node coordinates and can guarantee the spanner property. The authors further prove analytically that the proposed simple algorithm maintains network connectivity. The simulations included offer enough evidence on the benefits of the proposed algorithm. An actual implementation would definitely make the claims even stronger. I would greatly encourage the authors to pursue such a future direction.

Public review written by

Konstantina Papagiannaki

INTEL Research Pittsburgh

acm sigcomm

A Shortest-Path-Based Topology Control Algorithm in Wireless Multihop Networks

Yao Shen
Dept. of Computer Science & Engineering, Shanghai Jiaotong Univ., Shanghai 200240, China
shen_yao@sjtu.edu.cn

Yunze Cai
Dept. of Automation, Shanghai Jiaotong Univ., Shanghai 200240, China
yzcai@sjtu.edu.cn

Xiaoming Xu
Dept. of Automation, Shanghai Jiaotong Univ., Shanghai 200240, China
xmxu@mail.sjtu.edu.cn

ABSTRACT

In this paper, we present a shortest-path-based algorithm, called local shortest path(LSP), for topology control in wireless multihop networks. In this algorithm, each node locally computes the shortest paths connecting itself to nearby nodes based on some link weight function, and then it selects all the second nodes on the shortest paths as its logical neighbors in the final topology. Any energy model can be employed in LSP to design the link weight function whose value represents the power consumption required in the transmission along a link. We analytically prove that such a simple algorithm maintains network connectivity and guarantees that the minimal energy path between any two nodes is preserved in the final topology. Simulation results show that LSP can reduce the energy consumption, especially in heterogenous networks.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: Wireless Communication, Network Topology

General Terms

Algorithms

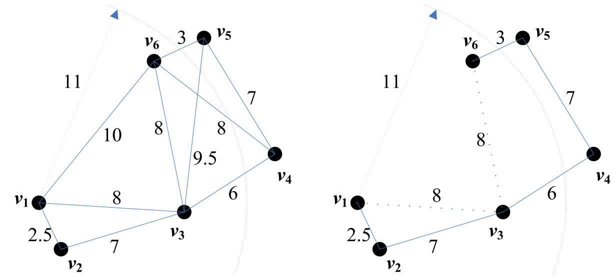
Keywords

Wireless multihop network, topology control, minimal energy path, connectivity.

1. INTRODUCTION

Energy conservation may be one of the most important issues for wireless networks. Topology control algorithms are used to reduce node power consumption and extend network lifetime while maintaining network connectivity. The objective of the topology control algorithms is actually to select appropriate logical neighbors(i.e. neighbors in the topology derived under a topology control algorithm) from physical neighbors(i.e. neighbors in the original network without any topology control algorithm employed) for each node in a network according to some special rules. Thus, each node can adjust its transmission power to just cover all of its logical neighbors.

Since the power required to transmit a message increases at least quadratically with distance, the strategy of selecting the nearest neighbors as the logical neighbors to maintain network connectivity is broadly employed by most of the existing topology control algorithms [1–10]. However, it needs more investigations to determine whether such a strategy is a practical way to construct the power-efficient topology. Actually, the power efficiency of a topology control algorithm is mainly determined by two components. One is the algorithm rules by which the logical neighbors are selected, and the other is the energy model that the rules depend on. We simply discuss them as follows.



(a) (b)

Figure 1: Necessity of the minimal energy path preservation in topology control algorithms.(The number on each link represents the link length.) (a)Original network topology with the common maximal transmission range of 11 units. (b)LMST$^-$(formed by solid line links) and LMST$^+$(formed by solid and dot line links).

First, the algorithm rules are pivotal since they directly affects the selection of logical neighbors. Most of the past algorithms select the nearest neighbors as their logical neighbors, whereas this rule usually leads to energy-inefficient transmissions. Figure 1 shows an original network and its topologies derived under LMST [8]. Suppose the minimal power required by a link is the square of the link length, then since v_6 is not selected as the logical neighbor of v_1, the transmission from v_1 to v_5 at least consumes power of 140.25 units in LMST$^-$(LMST with link removal) and 128.25 units in LMST$^+$(LMST with link addition), while in the original network, the most power-efficient path (v_1, v_6, v_5) only consumes 109 units. The same case exists in the transmission between v_1 and v_6. Note that the problem does not depend on the energy model employed in an algorithm. As a matter of fact, *topology control algorithms should select the appropriate logical neighbors to ensure that the minimal energy paths in the original network preserved in the final topology.*

However, most of the past algorithms have not explicitly considered this problem. The only two exceptions may be the algorithms R&M [1] and SMECN [2]. We will introduce them in the next section.

Secondly, for the energy model, it has been widely accepted that the assumption made in most of the past algorithms that the transmission power can represent the total power consumption of a transmission is unrealistic. Meanwhile, different networks in different environments may have different energy consumption model. Therefore, a good topology control algorithm suitable for different networks should not assume any special energy model.

Aiming at a topology control algorithm with the consideration of preserving the minimal energy paths, in this paper we propose the local shortest-path-based topology control algorithm named LSP. Our major contribution is that LSP is a framework which can be instantiated with any energy model, while ensuring that the minimal energy path between any two nodes is preserved in the final topology.

2. RELATED WORK

There are two algorithms closely related to ours. The first algorithm(denoted as R&M) is provided by Rodoplu and Meng [1], and the second algorithm SMECN is proposed by Li and Halpern [2]. Both algorithms run based on the information of relay region. The *relay region* of the transmit-relay node pair (v_i, v_j) is the physical region such that relaying through v_j to any point in the region takes less power than direct transmission. In R&M, each node first executes local search to find the enclosure graph. The *enclosure* of the transmit node v_i is defined as the union of the complement of relay regions of all nodes v_i can reach. Then each node runs the distributed Bellman-Ford shortest path algorithm upon the enclosure graph using power consumption as the cost metric. Lastly, the minimum cost neighbors are selected as the logical neighbors in the final topology. SMECN determines the minimal transmission power in a different manner. Each node v_i first broadcasts "Hello" message with an initial power, getting Acks from all nodes in its transmission range, and checking if the current range covers the region of the maximal transmission range minus the union of the complement of relay regions of all nodes v_i can reach. If not, it transmits with more power. This process is guaranteed to terminate as long as the power increases to the maximal transmission power.

Although both R&M and SMECN provide the solutions to preserve minimal energy paths in the final topology, several weaknesses may prevent their applications. The most important one is that both algorithms implicitly assume that a long link consumes more power than a short link. However, as we have discussed above, this assumption is unpractical in heterogeneous networks. Next, SMECN assumes that each node knows its transmission region, which is actually difficult for a wireless device. In R&M, the authors imply that the region is a circle. However, obstacles can not always be avoided in networks. Both algorithms spend a lot of time computing relay region and judging node position relations. Moreover, all the nodes need to know their exact positions in both algorithms. Our algorithm proposed in this paper does not require the above assumptions or preconditions. Furthermore, any realistic energy model can be used in LSP.

There are a number of other papers in the literature on power-efficient topology control. We approximately classify them into two families. The first family of algorithms, such as [3–7], is based on geometry and graph theory. The common ground of this family of algorithms is that all of them select the shortest links to preserve network connectivity while maintaining special geometric characteristics of the final topology. Unfortunately, those algorithms are usually not power-efficient under the realistic energy models where short links may consume more power than long links. The second family of algorithms, such as [8–10], is based on link weight. These algorithms are usually not efficient in heterogenous networks since the path preservation was not considered.

3. PROPOSED ALGORITHM

In this section, we propose the LSP algorithm to build power-efficient topology for wireless multihop networks. To facilitate discussion of the algorithm, we first define the following terms. We model the topology of a wireless network with each node using its maximal transmission power as an undirected graph $G = (V, E)$ in the two dimensional plane, where $V = \{v_1, v_2, ..., v_n\}$ is the set of nodes in the network and E is the set of bidirectional links. The network discussed in this paper may be heterogeneous, and hence each node v_i may have its own transmission power p_t^i which can be adjusted by itself. However, since asymmetric communication is unpractical in wireless multihop networks [11], in this paper, only bidirectional links are concerned. Therefore, the bidirectional link $(v_i, v_j) \in E$ implies that both v_i and v_j are covered by each other.

We define the physical neighbor set of each node v_i as

$$NS_p^i = \{v_k | (v_i, v_k) \in E(G) \text{ or } (v_k, v_i) \in E(G)\}. \quad (1)$$

The logical neighbor set NS_l^i is a subset of NS_p^i, i.e. $NS_l^i \subseteq NS_p^i$.

In LSP, each bidirectional link is assigned a weight which can be derived from the weight function w. Thus the weight of a link (v_i, v_j) can be expressed by $w(i, j)$. We use link weight to represent the power consumption required in the transmission along a link and use path weight to represent the sum of all link weights of a path. Therefore, we define the minimal energy path as the path with the minimal path weight among all the paths connecting two given nodes. The computation of $w(i, j)$ usually relates only to v_i and v_j, at most to their neighbors. This makes it possible that each node runs LSP only according to the locally collected information. However, the algorithm will not miss any logical neighbor which is placed on some minimal energy path in the original network. Therefore, LSP preserves minimal energy paths in the final topology.

The algorithm mainly consists of following steps:

1) Link weight calculation

2) Link weight information exchange

3) Local topology construction

4) Transmission power adjusting

3.1 Link weight calculation

Each node in this step locally collects the information needed in the weight calculations for all links associated with it, and calculates link weights. Note that different energy model may require different information in the calculations.

For the design of the weight function w, we suggest that it should have the same value for both directions of a link. This makes sense since communication always consists of transmissions(including data transmissions and control message transmissions) in both directions, and furthermore this ensures that for any two nodes, they have the same minimal energy path connecting them. However, in some energy model the two endpoints of a link (v_i, v_j) may derive different weights(called unidirectional link weight, and denoted as $\overrightarrow{w}(i,j)$ and $\overrightarrow{w}(j,i)$ respectively). In this case, v_i can know $\overrightarrow{w}(j,i)$ and v_j also can know $\overrightarrow{w}(i,j)$ in the following second step by information exchanging. Thus $w(i,j)$ can be designed as

$$w(i,j) = (\overrightarrow{w}(i,j) + \overrightarrow{w}(j,i))/2. \qquad (2)$$

We do not assume any form of $\overrightarrow{w}$ since here we only consider the LSP framework. Therefore, in this subsection, we simply suppose that each node derives the unidirectional link weights.

3.2 Link Weight Information Exchange

In this step, each node v_i exchanges the unidirectional link weight information derived in the first step with its 1-hop neighbors by broadcasting using its maximal transmission power p^i_{max}. After that, v_i knows all the weights calculated at the nodes in the set of $\{v_i\} \cup NS^i_p$. Therefore, for any two nodes $v_x, v_y \in \{v_i\} \cup NS^i_p$, if there is a bidirectional link (v_x, v_y), v_i can derive $w(x,y)$ according to (2). Two points should be mentioned. First, node v_i may receive $\overrightarrow{w}(j,i)$ from node v_j while v_j is out of its range. In this case, link (v_j, v_i) is not considered by v_i. Secondly, v_i only receives $\overrightarrow{w}(x,y)$ and does not receive $\overrightarrow{w}(y,x)$, which is because $v_y \notin \{v_i\} \cup NS^i_p$. Thus, link (v_x, v_y) is still not considered.

The information exchange in this step is avoidable if each node v_i can obtain the information required to calculate the unidirectional link weights for all the nodes in NS^i_p. In this case, each node does all the weight calculations for all links. Although this approach needs more computation time and stricter preconditions, it reduces the number of information exchanges to one.

3.3 Local Topology Construction

In this step, each node v_i computes the local shortest path connecting it to every node $v_j \in NS^i_p$ according to the derived link weights. The Dijkstra's algorithm can be employed if there is no negative link weight, and the time complexity varies from $O(|NS^i_p|^2)$ to $O(|E_{NS^i_p}| log|NS^i_p|)$ depending on the implementation of the algorithm where $E_{NS^i_p}$ is the set of all bidirectional links whose endpoints are in NS^i_p. If negative link weights exist, then the Bellman-Ford algorithm can be used and the time complexity is $O(|NS^i_p| \cdot |E_{NS^i_p}|)$. Note that in some networks, transmissions along a link may reduce the power consumptions of the nearby nodes by using some strategy such as turning their radios off. In this case, the link may have a negative weight.

Denote the local shortest path connecting v_i to $v_j \in NS^i_p$ as

$$path^{i,j} = (v_{p^{i,j}_0} = v_i, v_{p^{i,j}_1}, ...v_{p^{i,j}_{n-1}}, v_{p^{i,j}_n} = v_j) \qquad (3)$$

where $v_{p^{i,j}_m} \in NS^i_p, m = 1, 2, ..., n$ and $n \geq 1$. Then the logical neighbor set NS^i_l can be represented by

$$NS^i_l = \{v_{p^{i,x}_1} | v_x \in NS^i_p\}. \qquad (4)$$

That is, all the second nodes on the paths compose the logical neighbor set.

Note that the path is bidirectional since every link on the path is bidirectional. However, the path $path^{j,i}$ is not the reverse of $path^{i,j}$ since $NS^i_p \neq NS^j_p$. Another point we should mention is that not every neighbor close to v_i is its logical neighbor and meanwhile not every logical neighbor of v_i is close to it since the short link is not always power-efficient according to some realistic energy model. Through the construction of the local shortest paths, each node can derive a local route table which is described as

Destination	Next hop	Link weight

Each physical neighbor(the destination) has an entry in the table. The link weight represents the weight of the link connecting the current node and the next hop. It can be used by upper level routing algorithm to find a least weighted path.

The network topology under LSP is all the nodes in V and their individually perceived logical neighbor relations.

Definition 1 (Topology G_0): The topology G_0, derived under LSP is an undirected graph $G_0 = (V(G_0), E(G_0))$, where $V(G_0) = V$, $E(G_0) = \{(v_i, v_j) | v_i \in NS^j_l$ or $v_j \in NS^i_l\}$.

3.4 Transmission Power Adjusting

Topology control algorithms usually compute the minimal transmission power lastly. However, it is an optional step. In some networks a node can adjust its transmission power to just cover the next hop on the routing path when transmitting. In that case, it is possible to transmit messages with minimal energy. However, frequently adjusting transmission power for each different routing path is not suitable for nodes in most networks. Therefore, we simply discuss the transmission power adjusting in LSP as follows.

After each node v_i derives its logical neighbor set NS^i_l, it computes the minimal transmission power p^i_{adj} required to cover all of its logical neighbors, and then adjusts its transmission power to be p^i_{adj}. The computation of p^i_{adj} relates to the energy model, and may require information such as the distances from v_i to its logical neighbors and their power consumption properties. Different information can be derived by different means which will not be discussed in this paper.

4. THEORETICAL ANALYSIS

For any two nodes $v_i, v_j \in V(G)$, node v_i is said to be connected to node v_j (denoted as $v_i \Leftrightarrow v_j$) if there exists at least one bidirectional path $(v_{p_0} = v_i, v_{p_1}, ...v_{p_{n-1}}, v_{p_n} = v_j)$ in G_0 where $v_{p_x} \in V(G), x = 1, 2, ..., n-1$. It follows that $v_i \Leftrightarrow v_k$ if $v_i \Leftrightarrow v_j$ and $v_j \Leftrightarrow v_k$.

LEMMA 1. *For any two nodes $v_i, v_j \in V(G)$, if $v_j \in NS^i_p$, then $v_i \Leftrightarrow v_j$.*

PROOF. Note that according to the definition of topology G_0, $v_x \Leftrightarrow v_y$ if $v_y \in NS^x_l$. For any nodes v_i and v_j satisfying that $v_i, v_j \in V(G)$ and $v_j \in NS^i_p$, there must be the local shortest path $path^{i,j} = (v_{p_0} = v_i, v_{p_1}, ...v_{p_{n-1}}, v_{p_n} = v_j)$ connecting them where $n \geq 1$. Therefore, $v_{p_1} \in NS^i_l$, and

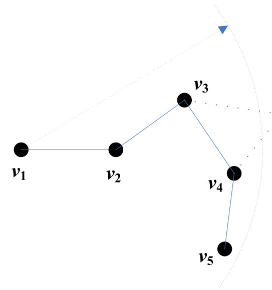


Figure 2: The local shortest path on which $v_4 \notin NS_l^3$.

hence $v_i \Leftrightarrow v_{p_1}$. If $n = 1$, lemma 1 is proved. Otherwise, for any nodes v_{p_x} and $v_{p_{x+1}}$, $x = 1, 2, ..., n-1$, if $v_{p_{x+1}} \in NS_l^{p_x}$, then $v_{p_x} \Leftrightarrow v_{p_{x+1}}$. If $v_{p_{x+1}} \notin NS_l^{p_x}$, since $v_{p_{x+1}} \in NS_p^{p_x}$, there exists the local shortest path $path^{p_x,p_{x+1}} = (v_{p'_0} = v_{p_x}, v_{p'_1}, ...v_{p'_{m-1}}, v_{p'_m} = v_{p_{x+1}})$, $m > 1$. Therefore, we have $v_{p_x} \Leftrightarrow v_{p'_1}$. Continue this iterative process, we finally have $v_i \Leftrightarrow v_j$. $\square$

Theorem 1. *G_0 preserves the connectivity of G, i.e., G_0 is connected if G is connected.*

Proof. Suppose G is connected. For any two nodes $v_i, v_j \in V(G)$, there is at least one path $(v_{p_0} = v_i, v_{p_1}, ... v_{p_{n-1}}, v_{p_n} = v_j)$ connecting them, where $(v_{p_i}, v_{p_{i+1}}) \in E(G)$, $i = 0, 1, ...n-1$. Since $v_{p_{i+1}} \in NS_p^{p_i}$, according to lemma 1, we have $v_{p_i} \Leftrightarrow v_{p_{i+1}}$. Therefore, $v_{p_0} = v_i \Leftrightarrow v_{p_1} \Leftrightarrow ... \Leftrightarrow v_{p_{n-1}} \Leftrightarrow v_{p_n} = v_j$. $\square$

In our algorithm, each node v_i computes its local shortest paths only depending on the locally collected information. However, for any two connected nodes in the topology G, there actually exists the global shortest path(i.e. the minimal energy path) connecting them. Denote the global shortest path connecting v_i to v_j as $path_g^{i,j}$, where $v_i, v_j \in V(G)$ and $v_i \neq v_j$. For node v_i, suppose all the second nodes on the paths $path_g^{i,x}(v_x \in V(G), v_i \neq v_x)$ compose the global logical neighbor set $NS_{l,g}^i$, then we have the following lemma.

Lemma 2. *For any node $v_i \in V(G)$, $NS_{l,g}^i \subseteq NS_l^i$.*

Proof. For each node v_i, we prove that for any node $v_x \in NS_{l,g}^i$, $v_x \in NS_l^i$. Otherwise, if $v_x \notin NS_l^i$, then since $v_x \in NS_{l,g}^i$, $v_x \in NS_p^i$. That means there must be the local shortest path $path^{i,x}$ connecting v_i to v_x with a less weight than the one of the link (v_i, v_x). Suppose that v_x is selected as a member of $NS_{l,g}^i$ because of the global shortest path $path_g^{i,j}$, then replace the link (v_i, v_x) of $path_g^{i,j}$ with $path^{i,x}$, we have a less weighted path, which contradicts that $path_g^{i,j}$ is the global shortest path in the network. $\square$

Lemma 3. *Given any node v_{p_x} on the global shortest path $path_g^{i,j}$ satisfying $x \geq 1$, then $v_{p_x} \in NS_l^{p_{x-1}}$.*

Proof. For any global shortest path $path_g^{i,j} = (v_{p_0} = v_i, v_{p_1}, ...v_{p_{n-1}}, v_{p_n} = v_j)$, the path $(v_{p_{x-1}}, v_{p_x}, ...v_{p_{n-1}}, v_{p_n} = v_j)$ is also the global shortest path where $1 \leq x \leq n$. Therefore, any node v_{p_x} satisfies $v_{p_x} \in NS_{l,g}^{p_{x-1}}$ where $1 \leq x \leq n$. According to lemma 2, we have $v_{p_x} \in NS_l^{p_{x-1}}$. $\square$

Lemma 3 does not hold for the local shortest path, say $path^{i,j}$. It is because for any node v_{p_x} on $path^{i,j}$ satisfying $x \geq 1$, $path^{p_x,j}$ may not be a part of $path^{i,j}$ since $NS_p^{p_x} \nsubseteq NS_p^i$. Figure 2 shows an example where $path^{1,5} = (v_1, v_2, v_3, v_4, v_5)$ and $path^{3,5} = (v_3, v_6, v_4, v_5)$. Therefore, $v_4 \notin NS_l^3$.

Theorem 2. *For any two connected nodes $v_i, v_j \in V(G)$, the global shortest path $path_g^{i,j}$ exists in G_0.*

Proof. First, since v_i and v_j are connected in G, there must be the global shortest path $path_g^{i,j} = (v_{p_0} = v_i, v_{p_1}, ... v_{p_{n-1}}, v_{p_n} = v_j)$ connecting them. Next, according to lemma 3, $v_{p_x} \in NS_l^{p_{x-1}}$, $x = 1, 2, ..., n$. Lastly, according to the definition of topology G_0, $(v_{p_{x-1}}, v_{p_x}) \in E(G_0)$. $\square$

Since in LSP, link weight represents the power consumption required in the transmission along a link, according to theorem 2, *LSP preserves the minimal energy paths connecting any two nodes in V(G)*, which is a main motivation of LSP.

Theorem 3. *For the two topologies G_0 and G_s(derived under SMECN [2]) of the same homogeneous network, each node in G_0 has a transmission power not larger than that of the corresponding node in G_s.*

Proof. According to the definition of the logical neighbor set(refer to (4)), for each logical neighbor v_j of any node v_i in G_0, v_j must be in the region(called *direct-transmission region*) in which v_i transmitting directly to v_j takes less power than relaying through any node in the transmission range of v_i. Therefore, according to the algorithm SMECN, v_j must be a neighbor of v_i in G_s. That means v_i in G_s uses a transmission power at least as large as that of v_i in G_0. $\square$

Note that in SMECN v_i increases its transmission power when it finds that the current transmission range can not cover its direct-transmission region. However, the power can not be increased continuously(in [2], the authors assumed that the transmission power was doubled in each step). Therefore, actually the final transmission power in G_s is usually larger than necessary. Meanwhile, in G_0, v_i can adjust its transmission power to just cover all its logical neighbors. Consequently, v_i usually has a transmission power in G_s larger than in G_0, which means transmission in G_s usually consumes more power than in G_0. Moreover, theorem 3 implies that G_0 is a subgraph of G_s. Meanwhile, it has been proved in [2] that G_s is a subgraph of the topology constructed by R&M. Therefore, LSP is actually more power-efficient than both SMECN and R&M in homogeneous networks.

5. PERFORMANCE EVALUATION

In this section, we present several sets of simulation results to evaluate the effectiveness of LSP. We mainly compare the performance of LSP with SMECN [2] since both algorithms preserve the minimal energy paths. R&M [1] is not selected since unlike LSP and SMECN, it requires the entire network information to compute the shortest paths in its Phase 2, and hence, actually it is not a localized algorithm. We also select LMST [8] and CBTC [7] since both of them are remarkable power-efficient algorithms with good performances. We compare LSP with them to show that LSP is a more power-efficient algorithm, while in other several aspects, LSP has the performances close to theirs.

Since SMECN is only applicable to the homogeneous networks where the minimal power required in supporting a link is a strictly increasing function of the link length, we first assume a homogeneous network energy model, and compare the performance of LSP with SMECN, LMST, CBTC and MaxPower. Then we assume a heterogenous network energy model and compare LSP with LMST, CBTC and MaxPower. In our simulations, we compare only to the optimized CBTC algorithm with parameter $\alpha = 2\pi/3$.

The performance metrics used in the study include the number of logical neighbors, the adjusted transmission range and the path power which is consumed by all the nodes on the minimal energy paths when transmitting. In our simulations, the minimal energy path is the path which has the least average of the path powers in both directions of a path among all the paths connecting two given nodes. We use *real path power* to denote the total power actually consumed by all the nodes on the minimal energy path when transmitting, and use *minimal path power* to denote the minimal total power required by those nodes for a successful transmission. Note that after adjusting transmission power, each node uses fixed power to send message. Therefore, for a minimal energy path, the real path power is usually larger than the minimal path power required by the path.

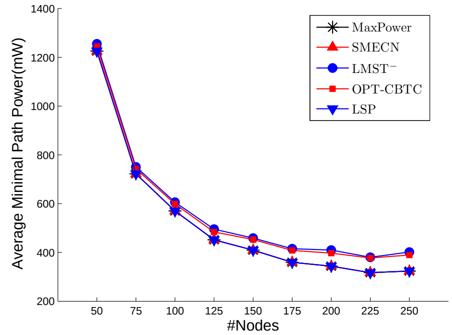
5.1 Simulation Results in Homogeneous Networks

We assume that a transmission between node v_i and v_j takes power $p_t^i(j) = td(i,j)^\alpha$ and a reception at the receiver takes power c, where t is the predetection threshold and α is the path loss exponent. Therefore, the link weight w can be defined as $w(i,j) = td(i,j)^\alpha + c$.

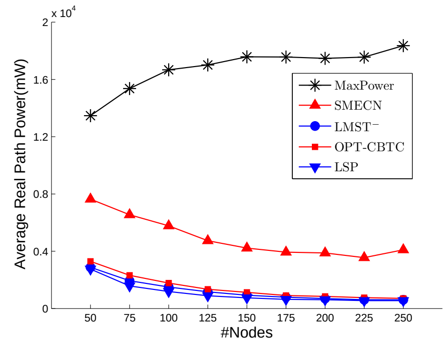
In all simulations in this subsection, we assume $t = 5 \times 10^{-7}$ and $\alpha = 4$. Nodes are randomly distributed in a 1000×1000 m^2 region. Each node has an omnidirectional antenna. The common maximal transmission range is 250 m, and the common receiver power is 20 mW. In SMECN, each node doubles its transmission power when the transmission range can not cover its direct-transmission region. Node v_i is said to be a neighbor of node v_j if both v_i and v_j cover each other.

For a network of 200 nodes, the topologies derived using the maximal transmission power, LMST$^-$ [8], OPT-CBTC [7], SMECN [2] and LSP are shown in Figure 3.

Next we vary the number of nodes in the region from 50 to 250. Each data point is the average of 50 simulation runs. Figure 4 and Figure 5 show the simulation results. As shown in Figure 4(a), both SMECN and LSP have the same average minimal path power as MaxPower, which means SMECN and LSP do preserve the minimal energy paths in the original networks. However, LMST$^-$ and OPT-CBTC do not have this property. From Figure 4(b) we can see that LSP has the slightly smaller average real path power than LMST$^-$ and OPT-CBTC. Meanwhile, although both LSP and SMECN preserve the minimal energy paths, SMECN surprisingly has a large average real path power. Figure 5 also shows that LSP has the performance close to those of LMST$^-$ and OPT-CBTC, and meanwhile significantly outperforms SMECN. Notice that the correctness of theorem 3 is illustrated by the fact that SMECN has larger average transmission ranges than LSP.



(a)



(b)

Figure 4: Performance comparisons in homogeneous networks (w.r.t minimal path power and real path power).(a)Average minimal path power. (b)Average real path power.

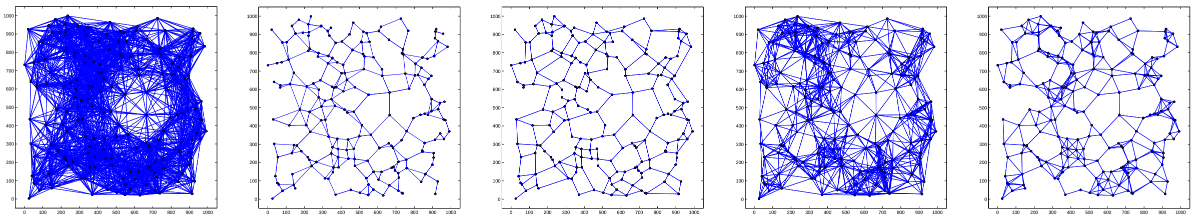
Although LMST$^-$ and OPT-CBTC do not preserve the minimal energy paths, the average real path powers of LMST$^-$ and OPT-CBTC approximate to that of LSP. However, the situation is not the same when running the algorithms in heterogeneous networks. This is because both LMST and OPT-CBTC are mainly based on link length while in heterogeneous networks, short links may need more power than long links. It may be worth considering that LMST uses a different weight function which is suitable for heterogeneous networks. However, this needs several modifications for LMST, and may make LMST lose some properties such as degree bound.

5.2 Simulation Results in Heterogeneous Networks

We assume that each node in the networks operates in one of the following states: *Idle*, *Transmit* and *Receive*. Denote the powers consumed at v_i in three states as p_{idle}^i, p_{tx}^i and p_{rx}^i respectively. The p_{tx}^i can be broken down into two parts, the power consumed by RF amplifier p_{amp}^i and the one by non-amplifier p_{elec}^i. Denote the power efficiency of RF amplifier as η^i, then $p_{amp}^i = p_t^i/\eta^i$ where p_t^i is the transmission power. Thus the power consumed at the sender v_i is

$$p_{tx}^i = p_{elec}^i + p_t^i/\eta^i \qquad (5)$$

The minimal transmission power $p_t^i(j)$ for supporting a link



(a) (b) (c) (d) (e)

Figure 3: Network topologies derived under different algorithms.(a)MaxPower. (b)LMST$^-$. (c)OPT-CBTC. (d)SMECN. (e)LSP.

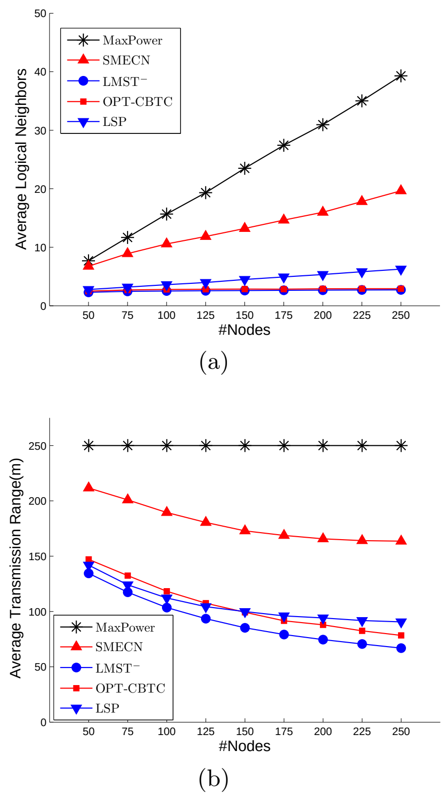


Figure 5: Performance comparisons in homogeneous networks (w.r.t logical neighbor and transmission range).(a)Average logical neighbor. (b)Average transmission range.

(v_i, v_j) is assumed to be

$$p_t^i(j) = p_s^j \times |v_i, v_j|^\alpha \tag{6}$$

where p_s^j is the receiver sensitivity of v_j.

In this model, a transmission also consumes energy of the nodes within the transmission range of the sender. We call the node within the transmission range of the sender an auditor. Denote the power consumed at the auditor v_k as p_a^k, then the power consumed at all auditors in a transmission from v_i is given by

$$p_{aud}^i = \sum_{\substack{v_k \in R^i(d), \\ v_k \neq v_i}} p_a^k \tag{7}$$

Table 1: Parameter Settings

Parameter	Value	Parameter	Value
p_{idle}	20-100mW	p_s	10^{-6}-10^{-8}
p_{rx} (ratio)	1.2-1.8	α	4
p_{elec} (ratio)	1.2-2.0	η	0.2-0.8
p_a (ratio)	1.05-1.15	r_{max}	250m

where $R^i(d)$ denotes the transmission range of the sender v_i with the transmission radius d.

We then define the unidirectional link weight $\overrightarrow{w}(x, y)$ as the minimum additional power consumption induced by the transmission from v_x to v_y. That is,

$$\begin{aligned} \overrightarrow{w}(x,y) &= p_{elec}^x + p_t^x(y)/\eta^x - p_{idle}^x \\ &+ p_{rx}^y - p_{idle}^y \\ &+ \sum_{\substack{v_k \in R^x(|v_x, v_y|), \\ v_k \neq v_x, v_y}} (p_a^k - p_{idle}^k). \end{aligned} \tag{8}$$

Note that $|v_x, v_y|$ is the minimum transmission radius for node v_x to cover v_y. Therefore, all the nodes other than v_x and v_y within $R^x(|v_x, v_y|)$ are considered to be the auditors. Thus, the link weight $w(x, y) = (\overrightarrow{w}(x, y) + \overrightarrow{w}(y, x))/2$ represents the average of the minimum additional power induced by the communication along the link (v_x, v_y) in both directions. As discussed in Section 3, all the information required in calculating w can be obtained through local information collection.

In the simulations, different nodes are randomly distributed in a 1000 × 1000 m^2 region. To simulate different node devices, we set several parameters to be randomly distributed in given intervals. As shown in Table 1, each node has p_{idle} randomly distributed between 20 and 100 mW. Its p_{rx}, p_{elec} and p_a are set to be the ratios of the corresponding values to p_{idle}. In the simulation network, we assume that each auditor can turn off its receiving circuitry if it detects that the signal on the air is not for it. Thus, each auditor has a small power consumption.

Figure 6 and Figure 7 show the performance comparisons. Each data point is also the average of 50 simulation runs. Unlike the simulation results in homogeneous networks, LSP has much smaller average path powers than both LMST$^-$ [8] and OPT-CBTC [7]. This is especially true for the average real path power. Therefore, in heterogeneous networks, LSP usually outperforms LMST$^-$ and OPT-CBTC with respect to the path power, and hence, LSP is more power-efficient. This superiority will be especially obvious in the networks where node devices differ from each other greatly.

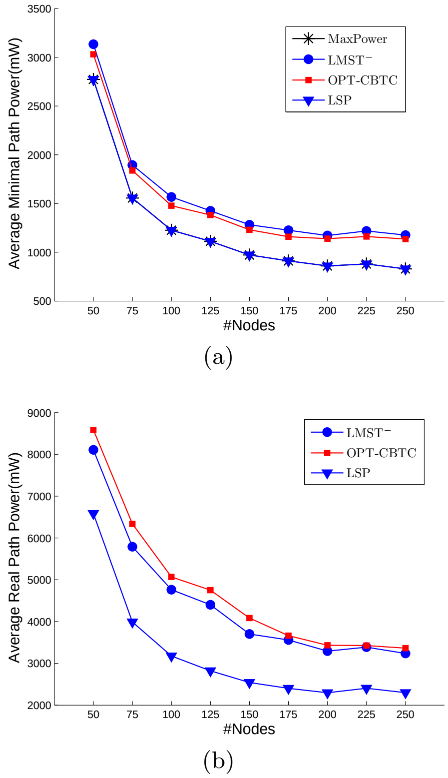


Figure 6: Performance comparisons in heterogeneous networks (w.r.t minimal path power and real path power).(a)Average minimal path power. (b)Average real path power.

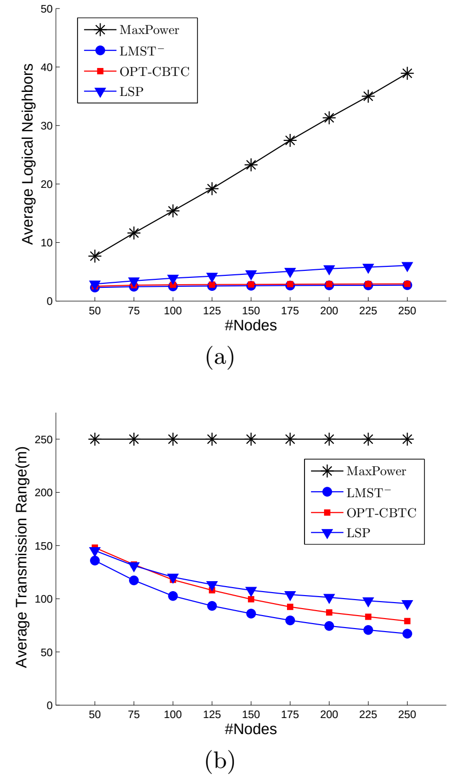


Figure 7: Performance comparisons in heterogeneous networks (w.r.t logical neighbor and transmission range).(a)Average logical neighbor. (b)Average transmission range.

6. CONCLUSIONS

We have proposed an algorithm LSP that computes a topology preserving the minimal energy paths. Differently from other topology control algorithms, LSP is a framework which can be instantiated with different energy models. Theoretical analysis proved that the topology derived under LSP maintains the network connectivity and preserves the minimal energy paths. Simulation results show that in homogenous networks, LSP significantly outperforms SMECN while having good performances close to those of LMST$^-$ and OPT-CBTC. Furthermore, in heterogenous networks, LSP is much more power-efficient than LMST$^-$ and OPT-CBTC while keeping a low number of average logical neighbors and a low average transmission range.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grant 60575036.

7. REFERENCES

[1] V. Rodoplu and T.H. Meng. Minimum energy mobile wireless networks. *IEEE Journal on Selected Areas in Communications*, 17(8):1333–1344, 1999.

[2] L. Li and J.Y. Halpern. A minimum-energy path-preserving topology-control algorithm. *IEEE Transactions on Wireless Communications*, 3(3):910–921, 2004.

[3] S. A. Borbash and E.H. Jennings. Distributed topology control algorithm for multihop wireless networks. In *Proceedings of the 2002 International Joint Conference on Neural Networks (IJCNN'02), (Honolulu, HI, USA, May 2002)*, pages 335–360, 2002.

[4] L. Hu. Topology control for multihop packet radio networks. *IEEE Transactions on Communications*, 41(10):1474–1481, 1993.

[5] X.-Y. Li, G. Calinescu, P.-J. Wan, and Y. Wang. Localized delaunay triangulation with application in ad hoc wireless networks. *IEEE Transactions on Parallel and Distributed Systems*, 14(10):1035–1047, 2003.

[6] W.-Z. Song, Y. Wang, X.-Y. Li, and O. Frieder. Localized algorithms for energy efficient topology in wireless ad hoc networks. In *Proc. ACM Mobihoc'04*, May 2004.

[7] L. Li, J.Y. Halpern, P. Bahl, Y.-M. Wang, and R. Wattenhofer. Analysis of a cone-based distributed topology control algorithm for wireless multihop networks. In *Proceedings of the ACM Symposium on Principles of Distributed Computing (PODC), (Newport, RI, Aug. 2001)*, pages 264–273, 2001.

[8] N. Li, J. C. Hou, and L. Sha. Design and analysis of an mst-based topology control algorithm. In *Proc. IEEE INFOCOM'03, (San Francisco, CA, Apr. 2003)*, pages 1702–1712, 2003.

[9] N. Li and J. C. Hou. Localized topology control algorithms for heterogeneous wireless networks. *IEEE/ACM Transactions on Networking*, 13(6):1313–1324, 2005.

[10] R. Wattenhofer and A. Zollinger. Xtc: a practical topology control algorithm for ad-hoc networks. *Technical Report 407, Department of Computer Science, ETH Xurich*, 2003.

[11] R. Prakash. Unidirectional links prove costly in wireless ad-hoc networks. In *Proceedings of the 3^{rd} international workshop on discrete algorithms and methods for mobile computing and communications (DIAL-M)*, 1999.