

Fault-tolerant Power-aware Topology Control for Ad-hoc Wireless Networks

Harichandan Roy, Shuvo Kumar De, Md.Maniruzzaman, and Ashikur Rahman

Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology, Dhaka, Bangladesh
{hari252bd, shuvo.buet, monir085}@gmail.com, ashikur@cse.buet.ac.bd

Abstract. Minimizing energy consumption and ensuring fault tolerance are two important issues in ad-hoc wireless networks. In this paper, we describe a distributed topology control algorithm which minimizes the amount of power needed to maintain bi-connectivity. The algorithm selects optimum power level at each node based on local information only. The resultant topology has two properties: (1) it preserves the minimum energy path between any pair of nodes and (2) it ensures fault tolerance by maintaining bi-connectivity. By presenting experimental results, we show the effectiveness of our proposed algorithm.

keywords: Ad-hoc Network, Topology control, Minimum-energy, Fault tolerance.

1 Introduction

Ad-hoc wireless networks are getting widespread with the recent development of wireless communication system. Since the basic components of multi-hop wireless networks are mostly battery-operated devices, power conservation is one of the key issues of such networks. It is not energy efficient to use the communication networks where each node transmits with its maximum power. So, power control is needed which deals with the problem of choosing the minimum power level by each node to minimize the energy consumption for the whole network. To ensure minimum power level, the topology has to preserve minimum-energy paths between the nodes. Besides preserving minimum-energy path, the topology control automatically maintains some properties such as reduced average node degree, smaller average transmission power etc. The network built in this way has profound effect on the performance of routing layer. Power control also results in extending battery life of the nodes.

On the other hand, by reducing the number of links in the network, topology control algorithms actually decrease the degree of routing redundancy. As a result, the topology thus derived is more susceptible to node failures/departures. Besides this, failure of nodes is a common phenomena in ad-hoc wireless network. This problem can be mitigated if an adequate level of fault tolerance can be properly ported into topology control. Fault tolerance increases the robustness of the network maintaining connectivity in case of any breakdown or an increase in load at any vicinity of the network.

In this paper, we propose a *distributed* topology construction algorithm based on *local information* only. By the term *local information* we mean that a node only has information about the position of one hop or two hops neighbors. Requiring more than two hops neighbor information a much overhead is incurred which will subdue the benefit of the topology control. Finally, the proposed algorithm preserves all minimum-energy paths between every pair of nodes and ensures fault tolerance by maintaining global bi-connectivity.

1.1 The Problem Statement

We use the same model as [7] which considers, a n -node, multi-hop, ad-hoc wireless network deployed on a two-dimensional plane. Suppose that each node is capable of adjusting its transmission power up to a maximum denoted by P_{max} . Such a network can be modeled as a graph $G = (V, E)$, with the vertex set V representing the nodes, and the edge set E defined as follows:

$$E = \{(x, y) | (x, y) \in V \times V \wedge d(x, y) \leq R_{max}\} \quad (1)$$

where $d(x, y)$ is the distance between nodes x and y and R_{max} is the maximum distance reachable by a transmission at the maximum power P_{max} . The graph G defined this way is called the maximum powered network. Note that the graph constructed this way is a visual representation of the inherent topology of the network. That is why we use the term *topology* and *graph* interchangeably throughout this paper.

Formally, the aim of this paper is to construct a graph $G' \subseteq G$ in a distributed fashion based on local information, where for any node pair u and v the minimum-energy path between u and v in G is also preserved in G' and moreover it provides fault tolerance as G' has at least two vertex-disjoint paths between any two nodes. Controlling topology in this way has a benefit to maintain connectivity through another backup path and hence make the topology more resilient to any node failures or departures.

2 Related work

A significant amount of research has been directed at power control algorithms for wireless mobile networks but a very few consider the problem of minimizing energy consumption and providing fault tolerance simultaneously.

Ramanathan et al. [8] considered the problem of adjusting the transmission powers of nodes and presented two *centralized algorithms* CONNECT and BICONN-AUGMENT. They introduced two heuristics to deal with the dynamics of the mobile environment. But neither heuristic absolutely preserves connectivity, even if it is achievable in principle. Cone-Based Topology Control (CBTC), proposed by Li et al. [5], generates a graph structure. A serious drawback of the algorithm is the need to decide on the suitable initial power level and the increment at each step. Bahramgiri et al. [1] augmented the CBTC algorithm

[5] to provide fault tolerance. However there is no guarantee that the proposed modification preserves minimum-energy paths.

Rodoplu and Meng [9] addressed a work targeting significant reductions in energy consumption. They introduced the notion of relay region based on a specific power model. Their work provides a distributed position-based network protocol optimized for minimum-energy consumption in mobile wireless networks. However the algorithm does not consider fault tolerance. Li [4] modified the algorithm of Rodoplu and Meng [9]. The sub-network constructed by their algorithm is provably smaller than that constructed by Rodoplu and Meng [9]. But the algorithm has the problem of partially-enclosed nodes. If a node is partially-enclosed, it has to use its maximum transmission power, which will soon drain out its battery power. Also this algorithm does not consider fault tolerance.

Shen et al. [10] proposed a distributed topology control algorithm, which preserves minimum-energy property and bi-connectivity. But the algorithm uses another algorithm presented in [3] to identify cut vertices which can not be done using *local information*. Acquiring global topology information is expensive and wastes more energy. Hajiaghayi et al. [2] addressed minimizing power while maintaining k -fault tolerance. However their distributed algorithm uses MST for which no locally computable algorithm is available. So, their distributed algorithm is not a local algorithm. Ning Li and Jennifer C. Hou [6] considered k -connectivity of wireless network to meet the requirement of fault tolerance. They presented a centralized greedy algorithm, called Fault-tolerant Global Spanning Subgraph ($FGSS_k$) and based on $FGSS_k$, proposed a localized algorithm, called Fault-tolerant Local Spanning Subgraph ($FLSS_k$). But their work, not necessarily, contains minimum-energy paths between any two node.

3 Definitions

3.1 Power Model

We use the same power model as [7]. Here, we assume the well known, generic, two-way, channel path loss model where the minimum transmission power is a function of distance. To send a packet from node x to node y , separated by distance $d(x, y)$, the minimum necessary transmission power is approximated by

$$P_{trans}(x, y) = t \times d^\alpha(x, y) \quad (2)$$

where $\alpha \geq 2$ is the path loss factor and t is a constant. Signal reception is assumed to cost a fixed amount of power denoted by r . Thus, the total power required for one-hop transmission between x and y becomes

$$P_{trans}(x, y) = t \times d^\alpha(x, y) + r \quad (3)$$

The model assumes that each node is aware of its own position with a reasonable accuracy, e.g., via a GPS device.

3.2 Smallest minimum-energy path-preserving subgraph of G

We say that a graph $G' \subseteq G$ is a minimum-energy path-preserving graph or, alternatively, that it has the minimum-energy property, if for any pair of nodes (u, v) that are connected in G , at least one of the (possibly multiple) minimum-energy paths between u and v in G also belongs to G' . Minimum-energy path-preserving graphs were first defined in [4]. Typically, many minimum-energy path-preserving graphs can be formed from the original graph G . The smallest of such subgraphs of G is $G_{min} = (V, E_{min})$, where $(u, v) \in E_{min}$ iff there is no path of length greater than 1 from u to v that costs less energy than the energy required for a direct transmission between u and v . Let $G_i = (V, E_i)$ be a subgraph of $G = (V, E)$ such that $(u, v) \in E_i$ iff $(u, v) \in E$ and there is no path of length i that requires less energy than the direct one-hop transmission between u and v . Then G_{min} can be formally defined as follows:

$$G_{min} = G_2 \cap G_3 \cap G_4 \cap \dots \cap G_{n-1} \quad (4)$$

Any subgraph G' of G has the minimum-energy property iff $G' \supseteq G_{min}$. Thereby, each of $G_i \supseteq G_{min}$, for any $i = 2, 3, \dots, n-1$ is a minimum-energy path preserving graph.

3.3 Cover region and Cover set

Consider a pair of nodes (s, f) , such that f lies within the transmission range of s , i.e., is reachable by s at P_{max} . Consider the set of all points that can possibly act as relays between s and f , such that it would be more power efficient for s to use an intermediate node located at one of those points instead of sending directly to f . We will call it the cover region of s and f and denote by $C(s, f)$. The collection of all nodes falling into the cover region of s and f will be called the cover set of s and f . Formally the cover region and cover set, are described by the following definition. The Cover Region $C(s, f)$ (Figure 1) of a pair of nodes (s, f) in G , where f is reachable from s is defined as:

$$C(s, f) = \{ \langle x, y \rangle \mid td^\alpha(s, \langle x, y \rangle) + td^\alpha(\langle x, y \rangle, f) + r \leq td^\alpha(s, f) \} \quad (5)$$

where $a \geq 2$. In the above equation, $d(s, \langle x, y \rangle)$ denotes the distance between node s and a hypothetical node located at $\langle x, y \rangle$ and r is the fixed receiving power. The Cover Set of the same pair (s, f) in G is defined as:

$$\xi_G(s, f) = \{ v \mid v \in V \wedge Loc(v) \in C(s, f) \} \quad (6)$$

where $Loc(v)$ is the location of the node positioned at $\langle x, y \rangle$ in the network.

3.4 Articulation point

A vertex v in a connected graph G is an articulation point iff the deletion of vertex v together with all edges incident to v disconnects the graph into two or more nonempty components. A graph G is bi-connected iff it contains no articulation points.

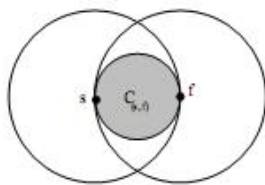


Fig. 1. The cover region between s and f

3.5 Minimum-energy path-preserving bi-connected graph

A graph G' is a minimum-energy path-preserving bi-connected graph iff it contains no articulation points and it keeps pairwise minimum-energy paths. A graph $G' = (V, E')$ is a minimum-energy path-preserving bi-connected subgraph of a bi-connected graph $G = (V, E)$ iff $E' \subseteq E$ and G' is bi-connected and for each pair of $(u, v) \in V$, minimum-energy paths between u and v in G are also in G' .

A variety of minimum-energy path-preserving graphs can be created if one or two hops' neighbor information is available. For example, let us consider a graph $G_2^1 = (V, E_2^1)$ which is a subgraph of $G = (V, E)$ such that $(u, v) \in E_2^1$ iff $(u, v) \in E$ and there is no path of length two that requires less energy than the direct path between u and v . Shortly, $(u, v) \in E_2^1$ iff $\xi_G(u, v)$ is empty. Note that such graph can be locally constructed if only one hop neighbors' position information is available to a node because a path of length two between u and v can only be created by using a neighbor z of u which is also a neighbor of v .

Another minimum-energy path-preserving graph is denoted as $G_2^2 = (V, E_2^2)$, which is another subgraph of $G = (V, E)$ such that $(u, v) \in E_2^2$ iff $(u, v) \in E$ and there are no two or more vertex-disjoint paths of length two requiring less energy than the direct path between u and v . Shortly, $(u, v) \in E_2^2$ iff $|\xi_G(u, v)| < 2$ where $|\xi_G(u, v)|$ represents number of elements in $\xi_G(u, v)$. Such a graph can be constructed if a node knows the position of all single hop neighbors.

Now let us define another graph $G_3^2 = (V, E_3^2)$, a subgraph of $G = (V, E)$ such that $(u, v) \in E_3^2$ iff $(u, v) \in E$ and there are no two or more vertex-disjoint paths of length three built with the neighbors of either u or v that require less energy than the direct path between u and v . However, to construct such a graph a node must know all its one hop and two hop neighbors exact location.

Mixing G_2^2 and G_3^2 we define $G_{2|3}^2 = (V, E_{2|3}^2)$ as the subgraph of $G = (V, E)$ such that $(u, v) \in E_{2|3}^2$ iff $(u, v) \in E$ and there are no two or more vertex-disjoint paths of length *two or three* requiring less energy than the direct path between u and v .

In this paper, we propose distributed algorithms to construct G_2^1 , G_2^2 , and $G_{2|3}^2$. We also prove that G_2^2 and $G_{2|3}^2$ are *minimum-energy path-preserving bi-connected* graphs.

4 Algorithms

In this section we describe distributed algorithms for constructing G_2^1 , G_2^2 , and $G_{2|3}^2$ topologies for a given topology G . The notations used in this section indicate the same meaning as in definition section. As we are considering distributed algorithm, the following algorithms run at each node as per necessity.

Consider a node s is constructing either of G_2^1 , G_2^2 , and $G_{2|3}^2$. At first s broadcasts a single *neighbor discovery message* (NDM) at the maximum power P_{max} . All 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 cover sets with those neighbors. Initially, all those sets are empty. The set $N_G(s)$, which also starts with empty set, keeps track of all the nodes discovered in the neighborhood of s in G . Whenever s receives a reply to its NDM from a node v , it executes the algorithm *updateCoverRegion*(s, v) described in Algorithm 1. After running *updateCoverRegion*(s, v), node s executes one of the following

Algorithm 1 *updateCoverRegion*(s, v)

```

1: for each  $w \in N_G(s)$  do
2:   if  $Loc(v) \in C(s, w)$  then
3:      $\xi_G(s, w) = \xi_G(s, w) \cup \{v\}$ 
4:   else if  $Loc(w) \in C(s, v)$  then
5:      $\xi_G(s, v) = \xi_G(s, v) \cup \{w\}$ 
6:   end if
7:    $N_G(s) = N_G(s) \cup \{v\}$ 
8: end for

```

algorithms presented in 4.1, 4.2 and 4.3 depending on the subgraph it likes to construct.

4.1 G_2^1 Topology construction

Assume the algorithm (Algorithm 2) is running on node s . For each neighbor v of s in G this algorithm checks whether the cover set between s and v is empty or not. If it is empty then v is included into the neighbor set of s in G_2^1 since it indicates that there is no node that can be used as relay to transmit message using lower energy than the direct path between s and v . Otherwise, v is not included into the neighbor set of s in G_2^1 .

4.2 G_2^2 Topology construction

Assume the algorithm (Algorithm 3) is running on node s . For each neighbor v of s in G this algorithm checks whether number of nodes in the cover set is less than 2 or not. If the number of nodes in the cover set is less than 2 then v is included into the neighbor set of s in G_2^2 since it indicates that there are no

Algorithm 2 G_2^1 TOPOLOGY CONSTRUCTION

```
1: for each  $v \in N_G(s)$  do
2:   if  $\xi_G(s, v)$  is empty then
3:      $N_{G_2^1}(s) = N_{G_2^1}(s) \cup \{v\}$ 
4:   end if
5: end for
```

two or more nodes that can be used as relay to transmit message using lower energy than the direct path between s and v . Otherwise, v is not included into the neighbor set of s in G_2^2 .

Algorithm 3 G_2^2 TOPOLOGY CONSTRUCTION

```
1: for each  $v \in N_G(s)$  do
2:   if  $|\xi_G(s, v)| < 2$  then
3:      $N_{G_2^2}(s) = N_{G_2^2}(s) \cup v$ 
4:   end if
5: end for
```

4.3 $G_{2|3}^2$ Topology construction

Assume the algorithm (Algorithm 4) is running on node s . Initially it is assumed that all the links in G are also in $G_{2|3}^2$. Then for each neighbor v of s in G this algorithm checks whether number of nodes in the cover set is greater than or equal to 2 or not.

If the number of nodes in the cover set $\xi_G(s, v)$ is greater than or equal to 2 then the vertex v is removed from $N_{G_{2|3}^2}(s)$ as it indicates that there are two or more lower energy paths than the direct path between s and v . If the cover set consists of only one node z , then the algorithm checks the presence of any path of length 3 built with neighbors of s excluding z . This path of length 3 is denoted by P_{-z} in the algorithm where $-z$ indicates that z is excluded and P_{-z} is boolean. If P_{-z} is true then there exists such path of length 3 otherwise not. So, if P_{-z} is true then v is removed from $N_{G_{2|3}^2}(s)$.

If the cover set is empty, v is removed from $N_{G_{2|3}^2}(s)$ if there exists at least two vertex-disjoint paths of length 3. These two vertex-disjoint paths must be built with neighbors of s .

5 Theorems

Lemma 1. For a given topology $G = (V, E)$ each pair of $(u, v) \in V$, $\xi_G(u, v)$ is empty iff $(u, v) \in E$ is a minimum-energy path.

Algorithm 4 $G_{2|3}^2$ TOPOLOGY CONSTRUCTION

```

1:  $N_{G_{2|3}^2}(s) = N_G(s)$ 
2: for each  $v \in N_G(s)$  do
3:   if  $|\xi_G(s, v)| \geq 2$  then
4:      $N_{G_{2|3}^2}(s) = N_{G_{2|3}^2}(s) - \{v\}$ 
5:   else if  $\xi_G(s, v)$  is  $\{z\}$  and  $P_{-z}$  is true then
6:      $N_{G_{2|3}^2}(s) = N_{G_{2|3}^2}(s) - \{v\}$ 
7:   else if  $\xi_G(s, v)$  is empty and at least two vertex-disjoint paths of length 3 consisting of the neighbors of  $s$  exist then
8:      $N_{G_{2|3}^2}(s) = N_{G_{2|3}^2}(s) - \{v\}$ 
9:   end if
10: end for

```

Proof. Since $(u, v) \in E$ is a minimum-energy path it indicates that there is no node in their cover region $C(u, v)$ which can be used as relay to transmit information between u and v with less energy than the direct transmission between u and v . So, $\xi_G(u, v)$ is empty.

On the other hand, if $\xi_G(u, v)$ is empty then there is no node in their cover region which implies no node exists that can be used as relay to transfer information between u and v with less energy than the direct transfer between u and v . So, $(u, v) \in E$ is a minimum-energy path.

Lemma 2. G_2^2 keeps all edges of G_2^1 .

Proof. Each edge $(u, v) \in E(G)$ is preserved in G_2^1 , iff the cover set $\xi_G(u, v)$ is empty. In case of G_2^2 two cases arise:

Case 1: $(u, v) \in E(G)$ is preserved when the cover set $\xi_G(u, v)$ is empty.

Case 2: $(u, v) \in E(G)$ is also preserved when the cover set $\xi_G(u, v)$ contains only one node. So, G_2^2 keeps all edges of G_2^1 by case 1.

Lemma 3. $G_{2|3}^2$ keeps all edges of G_2^1 .

Proof. Each edge $(u, v) \in E(G)$ is preserved in G_2^1 , iff the cover set $\xi_G(u, v)$ is empty. In $G_{2|3}^2$, three types of cases arise:

Case 1: The edge $(u, v) \in E(G)$ is preserved in $G_{2|3}^2$ when the cover set $\xi_G(u, v)$ is empty.

Case 2: The edge $(u, v) \in E(G)$ is preserved in $G_{2|3}^2$ iff there are no less costly two paths, one of length 2 and another of length 3.

Case 3: The edge $(u, v) \in E(G)$ is preserved in $G_{2|3}^2$ iff there are no less costly two paths of length 3. So, $G_{2|3}^2$ preserves all edges of G_2^1 by case 1.

Theorem 1. G_2^1 topology preserves minimum-energy paths.

Proof by Induction. Base Case. Let $(u, v) \in E(G)$ and is a minimum-energy path between u and v . Applying Lemma 1 $\xi_G(u, v)$ is empty. So, $(u, v) \in E(G_2^1)$.

Induction Hypothesis. Let $u \sim w$, a minimum-energy path in G , is also in

G_2^1 where $u \in V$ and $w \in V$ where V indicates vertex set of G and G_2^1 .

Induction Step. Let $x \in V$. If $u \sim x$ is a minimum-energy path in G consisting of $u \sim w$ and (w, x) , then (w, x) must be a minimum-energy path. Since $(w, x) \in E(G)$ and is a minimum-energy path, from base case we can say that $(w, x) \in E(G_2^1)$. And from Induction Hypothesis, $u \sim w$, a minimum-energy path in G , is also in G_2^1 . Merging these conditions, $u \sim x$ is a minimum-energy path and it is contained in G_2^1 . So, G_2^1 topology preserves minimum-energy path.

Theorem 2. G_2^2 and $G_{2|3}^2$ topologies preserve minimum-energy path.

Proof. It is clear that $E(G_2^2) \subseteq E(G_2^1)$ by applying Lemma 2. And $E(G_2^1) \subseteq E(G_{2|3}^2)$ by applying Lemma 3. Since G_2^1 preserves minimum-energy path from Theorem 1, G_2^2 and $G_{2|3}^2$ topologies must preserve minimum-energy path.

Theorem 3. G_2^1 topology may not ensure bi-connectivity.

Proof by Case. Let, $G = (V, E)$ is a bi-connected graph where $V = \{1, 2, 3\}$ and $E = \{a, b, c\}$ (Figure 2(a)). And two distinct paths between 1 and 3 are $\langle 1, 2, 3 \rangle$ and $\langle 1, 3 \rangle$.

If $\xi_G(1, 3) = \{2\}$ then according to G_2^1 topology construction algorithm $\langle 1, 3 \rangle \notin E(G_2^1)$. Then now, G_2^1 topology has a path $\langle 1, 2, 3 \rangle$ where 2 is an articulation point (Figure 2(b)). So, Bi-connectivity may not ensure.



Fig. 2. Proof by case: (a) Bi-connected (b) Not bi-connected

Theorem 4. G_2^2 topology ensures bi-connectivity.

Proof by contradiction. Let $G = (V, E)$ is a bi-connected graph. Consider any vertex u and v is a neighbor of u . There exists less costly two paths $\langle u, x, v \rangle$ and $\langle u, y, v \rangle$ where $\{x, y\} \subseteq \xi_G(u, v)$. So, according to G_2^2 topology construction algorithm $(u, v) \notin E(G_2^2)$. Now, we have to prove that $G - (u, v)$ is still bi-connected. Let $G' \equiv G - (u, v)$ is not bi-connected and there must be at least one articulation point w . So, $G' - \{w\}$ is not connected. Three types of cases arise in choosing w :

Case 1. If $w \notin \{x, y\}$ and $w \notin \{u, v\}$ then $G' - \{w\}$ has $\langle u, x, v \rangle$ and $\langle u, y, v \rangle$ paths.

Case 2. If $w \in \{x, y\}$ then $G' - \{w\}$ has at least one path between u and v . Such as if $w = x$ then $\langle u, y, v \rangle$ still exists and if $w = y$ then $\langle u, x, v \rangle$.

Case 3. If $w \in \{u, v\}$ then all vertices in $G' - \{w\}$ are in same component.

So, u and v are always connected. This implies that there is no possibility to find out such w as an articulation point which contradicts our assumption that w is an articulation point. So, $G - (u, v)$ is bi-connected and G_2^2 topology ensures bi-connectivity.

Theorem 5. $G_{2|3}^2$ topology ensures bi-connectivity.

Proof by contradiction. Let $G = (V, E)$ is a bi-connected graph. Consider any vertex u and v is a neighbor of u . There exists less costly two paths $\langle u, X, v \rangle$ and $\langle u, Y, v \rangle$ where X and Y are the set of relay nodes. So, according to $G_{2|3}^2$ topology construction algorithm $(u, v) \notin E(G_{2|3}^2)$. Now, we have to prove that $G - (u, v)$ is still bi-connected. Let $G' \equiv G - (u, v)$ is not bi-connected and there must be at least one articulation point w . So, $G' - \{w\}$ is not connected. Three types of cases arise in choosing w :

Case 1. If $w \notin \{X \cup Y\}$ and $w \notin \{u, v\}$ then $G' - \{w\}$ has $\langle u, X, v \rangle$ and $\langle u, Y, v \rangle$ paths.

Case 2. If $w \in \{X \cup Y\}$ then $G' - \{w\}$ has at least one path between u and v . Such as if $w \in X$ then $\langle u, Y, v \rangle$ still exists and if $w \in Y$ then $\langle u, X, v \rangle$.

Case 3. If $w \in \{u, v\}$ then all vertices in $G' - \{w\}$ are in same component.

So, u and v are always connected. This implies that there is no possibility to find out such w as an articulation point which contradicts our assumption that w is an articulation point. So, $G - (u, v)$ is bi-connected and $G_{2|3}^2$ topology ensures bi-connectivity.

6 Simulation output

To evaluate performance of our algorithm we have created some sample ad hoc networks where nodes are deployed under uniform distribution. Initially we deployed 50 – 100 nodes over a flat square area of $650m \times 650m$. Figure 3(a) shows a typical deployment of 50 nodes, each having a maximum communication range of 250m. This is the starting graph G for our algorithm. The later one in Figure 3(b) is our reference graph G_{REF} which is created for comparison purpose with G_2^1 , G_2^2 and $G_{2|3}^2$. G_{REF} is constructed in the following way. **First**, we find the minimum-energy path between a pair of vertices such as u, v by Dijkstra's algorithm. **Then** we find the second minimum-energy path between u, v by Dijkstra's algorithm which is exclusive of the vertices of the former one. This process continues for all pairs of the vertices. **Finally** we remove all the edges not included in the paths chosen by this process. The graphs found by G_2^1 , G_2^2 , and $G_{2|3}^2$ topology construction algorithms from graph G are in Figure 3(c), Figure 3(d) and Figure 3(e) respectively.

Similarly, we have run simulations for 75 nodes and 100 nodes to compare the outputs with G_{REF} . The result is presented in Table-1, 2. From Table-1 we see that the edge reduction in G_2^1 is the most. But it may not ensure bi-connectivity. Among two bi-connected graphs $G_{2|3}^2$, and G_2^2 , the former one reduces more edges than the later one. We now compare the issue of fault tolerance among various

topology construction algorithms. Fault tolerance means maintaining network connectivity in case of any breakdown of the network. In case of bi-connected graphs, there are two disjoint paths between any two nodes. When one path gets destroyed, another path can still sustain the connectivity. So if a topology is bi-connected then it can ascertain fault tolerance. Since G_2^1 topology may not

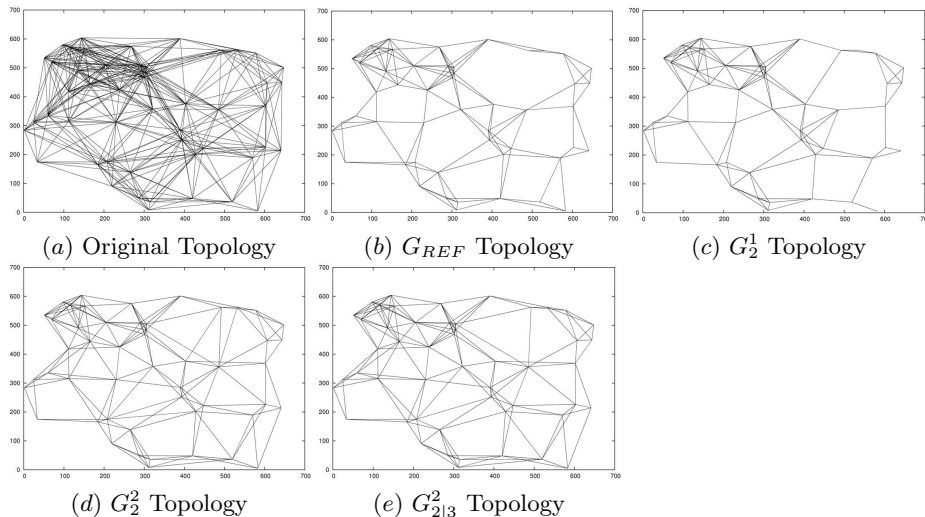


Fig. 3. Topologies for 50 nodes

ensure bi-connectivity, so it will not always ensure fault tolerance. But G_2^2 and $G_{2|3}^2$ are both bi-connected. So, they ensure fault tolerance.

Although G_2^2 and $G_{2|3}^2$ are bi-connected and they ensure two vertex-disjoint paths in any pair of vertices and hence fault-tolerant but none of them can be minimum-energy path. In Table-2 we show what percentage of node pairs have a vertex-disjoint backup path besides the minimum-energy path.

7 Conclusion

The algorithm we have presented to build minimum-energy path-preserving bi-connected graph is a distributed algorithm that maintains minimum-energy paths as well as provides fault tolerance in ad-hoc wireless networks. We have proposed G_2^1 , G_2^2 and $G_{2|3}^2$ topology construction algorithms of which G_2^1 seems less robust but G_2^2 and $G_{2|3}^2$ have higher fault tolerance and robustness. Simulation results show that $G_{2|3}^2$ has less number of edges than G_2^2 on the average. However G_2^2 preserves vertex-disjoint backup paths in addition to minimum-energy paths among more node pairs.

Table 1. Average number of edges in different topologies

Number of nodes	Original	G_{REF}	G_2^1	G_2^2	$G_{2 3}^2$
50	391	146	140	187	184
75	435	184	173	238	237
100	556	245	234	315	314

Table 2. Percentage of node pairs having vertex-disjoint backup path besides minimum-energy path

Number of nodes	Original	G_{REF}	$G_2^1(\%)$	$G_2^2(\%)$	$G_{2 3}^2(\%)$
50	100.0	100.0	93.75	99.30	98.30
75	94.68	94.68	87.75	93.92	93.92
100	100.0	100.0	92.79	97.99	97.98

References

1. M. Bahramgiri, M. Hajiaghayi, and V. S. Mirrokni. Fault-tolerant and 3-dimensional distributed topology control algorithms in wireless multi-hop networks. In *Proc. Eleventh International Conference on Computer Communications and Networks (ICCCN)*, pages 392–397, Oct. 2002.
2. M. Hajiaghayi, N. Immorlica, and V. S. Mirrokni. Power optimization in fault-tolerant topology control algorithms for wireless multi-hop networks. *IEEE/ACM Trans. Netw.*, 15(6):1345–1358, 2007.
3. A. Kazmierczak and S. Radhakrishnan. An optimal distributed ear decomposition algorithm with applications of biconnectivity and outerplanarity testing. *IEEE Trans. Parallel Distrib. Syst.*, 11(2):110–118, 2000.
4. L. Li and J. HalpernRahman. Minimum energy mobile wireless networks revisited. In *Proc. IEEE International Conference on Communications (ICC)*, june 2001.
5. L. Li, J. Y. Hlapern, P. Bahl, Y. Wang, and Watenhofer R. Analysis of a con-based topology control algorithm for wireless multi-hop networks. In *ACM Symposium on Principle of Distributed Computing (PODC)*, 2001.
6. Ning Li and Jennifer C. Hou. Flss: A fault tolerant topology control algorithm for wireless networks. In *Proc. of the 10th annual international conference on Mobile computing and networking*, pages 275–286, 2004.
7. A. Rahman and P.V. Gburzynski. Mac-assisted topology control for ad-hoc wireless network. *International Journal of communication Systems*, 19(9):995–976, 2006.
8. R. Ramanathan and R. Rosales-Hain. Topology control of multihop wireless networks using transmit power adjustment. In *Proc. of IEEE INFOCOM 2000*, pages 404–413, Tel Aviv, Israel, March 2000.
9. V. Rodoplu and T. Meng. Minimum energy mobile wireless networks. *IEEE Journal of Selected Areas in Communications*, 17:1333–1344, 1999.
10. Z. Shen, Y. Chang, C. Cui, and X. Zhang. A fault-tolerant and minimum-energy path-preserving topology control algorithm for wireless multi-hop networks. In *Proc. of the International Conference on Computational Intelligence and Security-I (CIS)*, pages 864–869, 2005.