Abstract

We consider ad-hoc networks consisting of \( n \) wireless nodes that are located on \( \mathbb{R}^2 \). Any two given nodes are called neighbors if they are located within a certain distance from one another. A given node can be directly connected to any one of its neighbors and picks its connections according to a unique topology control algorithm that is available at every node. Given that each node knows only the indices of its one- and two-hop neighbors, we identify an algorithm that preserves connectivity and can operate without the need of any synchronization among nodes. Moreover, the algorithm results in a sparse graph with at most \( 5n \) edges and a maximum node degree of 10. Existing algorithms with the same promises further require neighbor distance and/or direction information at each node.

1 Introduction

We consider \( n \) wireless nodes indexed (and uniquely identified) by the natural numbers \( 1, \ldots, n \) with locations \( x_1, \ldots, x_n \in \mathbb{R}^2 \). Given \( 1 \leq i < j \leq n \), Node \( i \) may potentially be connected to any Node \( j \) with \( |x_i - x_j| \leq R \), where \( |\cdot| \) is the Euclidean distance, and \( R > 0 \) is the communication range. As an example, a network consisting of 7 nodes with no connections is as shown in Fig. 1(a). In this example, Node 7 can potentially be connected to any other node except Nodes 2 and 4.

A major goal of topology control is to have a connected network, i.e. a network where there exists a path between any two given nodes so that information from one node may be conveyed to another [1]. In the case of our disk-connectivity model, the network can be made connected only whenever the Gilbert graph \( (\mathcal{V}, \mathcal{E}(\mathcal{V})) \) with \( \mathcal{V} \triangleq \{1, \ldots, n\} \) and \( \mathcal{E}(\mathcal{W}) \triangleq \{(i, j) : i, j \in \mathcal{W}, i < j, |x_i - x_j| \leq R\} \), \( \mathcal{W} \subset \mathcal{V} \) is connected. As an example, Fig. 1(b) shows the Gilbert graph corresponding to the setup in Fig. 1(a). This particular graph has 13 edges with a maximum (node) degree of 6, while a general Gilbert graph may have \( \frac{1}{2}n(n-1) \) edges with a maximum degree of \( n-1 \).

The existence of many edges and nodes with high degrees is not desirable in wireless networks due to several practical issues such as radio interference [2, 3]. One thus wishes to obtain sparse connected topologies with a constant maximum node degree. In practice, such a topology should be generated locally with every node picking its own connections according to a certain common algorithm that requires as little information as possible. In this context, given \( i \in \mathcal{V} \), let \( \mathcal{N}_i \triangleq \{ j : j \in \mathcal{V}, j < i, |x_i - x_j| \leq R \} \) represent the lower neighborhood of Node \( i \) (Lower in the sense that it only contains neighbors with smaller indices/identification numbers.). We assume that a given Node \( i \) only knows \( \mathcal{N}_i \) and \( \mathcal{N}_j, j \in \mathcal{N}_i \). In the following, we introduce a corresponding algorithm that provides a connected sparse network with constant maximum degree. We note that several existing
local algorithms such as XTC [4], NTC [5], LMST [6], CBTC [7] (also see [8-10] and [11] for a general survey on topology control and other algorithms) can all provide topologies with the same guarantees; however, they further require each node to know its exact distance and/or direction to its neighboring nodes as well as other extra side information. In fact, to the best of our knowledge, no previous algorithm can guarantee even a sparse connected topology (with no degree constraints) under the restrictions that we impose on node knowledge.

2 The Algorithm

Given \( i \in V \), consider the Gilbert graph \((N_i, E(N_i))\) generated by the lower neighborhood of Node \( i \). It is not difficult to show that \((N_i, E(N_i))\) can have at most 5 connected components, which we shall refer to as \((N_{ij}, E(N_{ij}))\), \( j = 1, \ldots, 5 \) (Of course, some or all of \( N_{ij} \) may be empty.). Our algorithm (at Node \( i \)) is then to “Connect to all nodes in the set \( \{\max N_{ij} : N_{ij} \neq \emptyset\} \).” Running the algorithm at each node exactly once results in a graph that we refer to as \((V, F)\). Nodes may run the algorithm in arbitrary order, or simultaneously in a completely asynchronous fashion.

As an example, for the setup in Fig. 1(a), the algorithm results in the topology in Fig. 1(c). In detail, for Node 6 in Fig. 1(a), the vertex sets of the two connected components induced by \( N_6 \) are \( N_{61} = \{1, 2, 5\} \) and \( N_{62} = \{3, 4\} \) so that Node 6 will establish connections to Nodes 5 and 4. Node 1, having no lower neighbors, will not attempt to connect to any other node. On the other hand, for Node 2, we have the single vertex set \( N_{21} = \{1\} \), so that Node 2 will connect to Node 1 (Hence, Node 1 in fact gets connected to Node 2, even though it is not Node 1 that initiates this connection.).

The following theorem summarizes some of the properties of the resulting topology \((V, F)\)

**Theorem 1.** The graph \((V, F)\) is connected if and only if the Gilbert graph \((V, E(V))\) is connected. Moreover, we have \(|F| \leq 5n\) and the degree of each node in \((V, F)\) is no more than 10.

**Proof.** For the statement regarding connectivity, we only need to prove the “if” part with the “only if” part being trivial. Suppose \((V, E(V))\) is connected. Then, for any given two nodes in \( V \), there is a (finite) path in \((V, E(V))\) that connects these two nodes with each edge in the path consisting of two neighboring nodes. To show that \((V, F)\) is connected, it is thus sufficient to show that any two Nodes \( i \) and \( j \) within range and (without loss of generality) \( i < j \) are path-connected in \((V, F)\). To prove this, first note that if \( i = j - 1 \), then, by design, Node \( j \) initiates a connection to Node \( i \) and we are done. Otherwise, \( \exists k \in V \) with \( i < k < j \) such that (i) Node \( j \) initiates a connection to Node \( k \), and (ii) there is a path \( P \) in \((V, E(V))\) connecting Node \( k \) to Node \( i \) such that the index of each node in \( P \) is no more than \( k \leq j - 1 \). It is then sufficient to show that any two distinct neighboring nodes that appear in \( P \) are path-connected in \((V, F)\). On the other hand, to prove this latter claim, it is sufficient to show that any two neighboring Nodes \( i' \) and \( j' \) with indices \( i' < j' \leq j - 1 \) are path-connected in \((V, F)\). Hence, any two neighboring Nodes \( i \) and \( j \) with \( i < j \) are path-connected in \((V, F)\) if either \( i = j - 1 \) or any two neighboring Nodes \( i' \) and \( j' \) with \( i' < j' \leq j - 1 \) are path-connected in \((V, F)\). This last statement describes a finite descent that immediately leads to the path-connectedness of Nodes \( i \) and \( j \). This concludes the proof of the claim on connectivity.

We now prove the rest of the claims. The inequality \(|F| \leq 5n\) follows immediately as each node initiates at most 5 connections. We now prove the degree bound. Let \( i \in V \). By design, a node with a lower index \( < i \) cannot initiate a connection to Node \( i \). On the other hand, Node \( i \) itself initiates at most 5 connections. It is thus sufficient to show that there are at most 5 nodes with a higher index \( > i \) initiating a connection to Node \( i \). Assume the contrary and suppose there are \( 6 \) or more such nodes. Two of these nodes, say Nodes \( j \) and \( k \) (with \( j < k \) without loss of generality) should then be within communication range as well as being within range of Node \( i \). This implies \( \{i, j\} \subset N_{kl} \) for some \( \ell \in \{1, \ldots, 5\} \) with \( i \notin N_{k\ell} \) and \( j \notin N_{k\ell} \) and \( \ell' \neq \ell \). Since \( \max C_{k\ell} \geq \max \{i, j\} = j > i \), and \( i \notin N_{k\ell} \) for \( \ell' \neq \ell \), we have, in fact, \( \max N_{k\ell} \neq i \) for every \( \ell \). This contradicts the fact that Node \( k \) initiates a connection to Node \( i \) and thus proves the degree bound.

There is more to say about the algorithm that generates \((V, F)\). For example, it can be shown that for a uniform random network [12] on \([0, 1]^2\), the algorithm provides asymptotically almost-sure connectivity with \( n(1 + o(1)) \) edges if \( R^2 \in \Omega(1/\log n) \) is just above the connectivity threshold. There are also applications to interference networks in the spirit of [3]. Also, the stretch factors associated with \((V, F)\) may be evaluated. Due to lack of space, we shall discuss these issues elsewhere.

**Acknowledgments**

This work was supported in part by the DARPA GRAPHS program Award N66001-14-1-4061.
References


