Topology Control of Multihop Wireless Networks using Transmit Power Adjustment

Ram Ramanathan, Regina Rosales-Hain
Internetwork Research Department
BBN Technologies (A Division of GTE)
Cambridge, Massachusetts, USA
{ramanath,rrosales}@bbn.com

Abstract— **We consider the problem of adjusting the transmit powers of nodes in a multihop wireless network (also called an ad hoc network) to create a desired topology. We formulate it as a constrained optimization problem with two constraints - connectivity and biconnectivity, and one optimization objective - maximum power used. We present two centralized algorithms for use in static networks, and prove their optimality. For mobile networks, we present two distributed heuristics that adaptively adjust node transmit powers in response to topological changes and attempt to maintain a connected topology using minimum power. We analyze the throughput, delay, and power consumption of our algorithms using a prototype software implementation, an emulation of a power-controllable radio, and a detailed channel model. Our results show that the performance of multihop wireless networks in practice can be substantially increased with topology control.**

I. Introduction

A *multihop wireless network* is one in which a packet may have to traverse multiple consecutive wireless links in order to reach its destination. Over the years, this general concept has manifested itself in numerous forms under numerous names. These include *packet radio* networks, developed several decades ago for tactical military communications, and more recently, *ad hoc* networks, used to refer to a collection of hosts communicating over a wireless channel. Other terms include *mobile* networks, *multihop radio* networks, and *dynamic* networks. Metricom Inc.'s Ricochet [1] network and the Army Near-Term Digital Radio (NTDR) [2] network are examples, respectively, of fully operational commercial and military multihop wireless networks.

The *topology* of a multihop wireless network is the set of communication links between node pairs used explicitly or implicitly by a routing mechanism. The topology depends on "uncontrollable" factors such as node mobility, weather, interference, noise, as well as on "controllable" parameters such as transmit power and antenna direction. While considerable research has been done on *routing* [3] - mechanisms that efficiently react to changes in the topology due to uncontrollable factors, the area of adjusting the *controllable* parameters in order to create the desired topology has received little attention. This paper addresses the problem of controlling the topology of the network by changing the transmit powers of the nodes.

Partially supported by the DARPA Global Mobile Information Systems (GloMo) contract DAAB07-97-C-D606

Why do we need to control the topology? Simply because the wrong topology can considerably reduce the capacity, increase the end-to-end packet delay, and decrease the robustness to node failures. For instance, if the topology is too sparse, there is a danger of network partitioning and high end-to-end delays. On the other hand, if the topology is too dense, the limited spatial reuse reduces network capacity. Networks that do not employ topology control are likely to be in one of these modes for a significant fraction of their operational time, resulting in degraded performance, or even disrupted connectivity. Furthermore, transmit power control results in extending battery life of the nodes - a crucial factor for many multihop wireless networks.

The specific problem we consider has not been studied previously. There has been some work in the general area of topology control and network design. In [4], an algorithm based on Delaunay triangulation is given to choose logical links. The objectives and constraints used there are different from ours, and adaptive control of transmit powers is not addressed. The selection of optimal transmission range to maximize throughput is studied in [5], [6]. However, they do not describe any techniques for actually controlling the power, nor do they concern themselves with connectivity. Topology design in *wired* networks, both in terms of physical links and virtual links to satisfy a given traffic matrix has been fairly well studied [7], [8] and are of some relevance as a source of adaptable ideas[1]. In summary, no research has considered the assignment of different transmit powers to different nodes to meet a *global topological property*, such as a connected network and studied an implementation in the context of a prototype multihop wireless network.

This paper makes several unique contributions to multihop wireless networking. We formulate topology control as a constrained optimization problem of practical importance, in particular as minimizing transmit power subject

[1]We note that the use of power control for solving the near-far problem in cellular networks - a topic of much current research - is completely unrelated to our work. Similarly, power-aware routing, which biases routes towards nodes with higher battery power, is also not relevant.

to the network being connected or biconnected. In doing so, we introduce a new analytical representation of multihop wireless networks that is more general and realistic than the conventional one. We then consider both static and mobile versions of the problem and give two sets of solutions, each respecting and exploiting the different characteristics of these domains. Specifically, we present provably optimal algorithms for static networks and present distributed heuristics for mobile networks. These heuristics involve techniques for global coordination with local information that might be of help in other distributed control problems.

II. Problem Statement

In this section, we develop a new representation for multihop wireless networks and define terms used in this paper. Conventionally, multihop wireless networks are represented as a graph where two vertices have an edge if and only if the corresponding nodes can communicate. We develop a new framework chiefly because the conventional representation hides the radio parameters and propagation properties that are critical to a realistic analysis. In our representation, the entities that contribute to the ability to communicate, namely the geographical locations, the propagation characteristics, and the node transmission parameters are kept separate.

Definition II.1: A *multihop wireless network* is represented as M = (N, L), where N is a set of nodes and L : N $\rightarrow$ (Z_0^+, Z_0^+) is a set of coordinates on the plane denoting the locations of the nodes.

Definition II.2: A *parameter vector* for a given node is represented as $P = \{f_1, f_2, .. , f_n\}$, where f_i : N $\rightarrow$ R, is a real valued adjustable parameter.

Examples of adjustable parameters include transmit power, antenna direction, spreading code length (chip length), etc. In this paper, we restrict our attention to transmit power. Thus, $P = \{p\}$, and the transmit power of a node u is given by $p(u)$. Following convention, we work in units of dB for power levels and signal strengths.

Definition II.3: The *propagation function* is represented as $\gamma : L \times L \rightarrow Z$, where L is a set of location coordinates on the plane. $\gamma(l_i, l_j)$ gives the loss in dB due to propagation at location $l_j \in L$, when a packet is originated from location $l_i \in L$.

The propagation function captures the environmental characteristics determining the formation of a link. It could be measured as described in [10] or approximately modelled with a function.

The successful reception of a transmitted signal depends, along with the propagation function γ, on the transmit power p, and the *receiver sensitivity* S. The receiver sensitivity is the threshold signal strength needed for reception and is assumed to be an apriori known constant, same for all nodes. In particular, for successful reception,

$$p - \gamma(l_i, l_j) \ \geq \ S \qquad (1)$$

We assume that γ is a monotonically increasing function of the geographical distance $d(l_i, l_j)$ between l_i and l_j. This is generally true for free space propagation or when environmental clutter causes the same amount of signal degradation in all directions [10]. We can then combine S and γ into one function as follows.

$$\lambda(d) = \gamma(d(l_i, l_j)) + S \qquad (2)$$

Clearly, p must be at least $\lambda(d)$ for successful reception. This leads to the following defintion, of significant importance in this paper.

Definition II.4: The *least-power* function $\lambda(d)$ gives the minimum power needed to communicate a distance of d.

The representation of the communication capability as a graph is useful when considering graph-theoretic concepts.

Definition II.5: Given a multihop wireless network M = (N, L), a transmit power function p, and a least-power function λ, the *induced graph* is represented as G = (V, E), where V is a set of vertices corresponding to nodes in N, and E is a set of undirected[2] edges such that $(u, v) \in$ E if and only if $p(u) \geq \lambda(d(u, v))$, and $p(v) \geq \lambda(d(u, v))$.

We use standard graph-theoretic terminology from [11]. In particular, a graph is said to be *k-vertex/edge-connected* if and only if there are k vertex/edge-disjoint paths between every pair of vertices. Note that if a graph is k-vertex connected, then it is also k-edge connected, but the converse is not true. For this reason, and because vertex connectivity is important for resilience to node failures and hotspots, we shall consider only vertex connectivity. We shall omit the word "vertex" for brevity. Thus, if k is 1, the graph is *connected*, and if k is 2, it is *biconnected*. The *degree* of a vertex is the number of edges incident on that vertex. We only consider *undirected* graphs, that is, all edge-relations on vertex pairs are symmetric.

In general, we can look at the topology control problem as one of optimizing a set of cost metrics under a given set of constraints. Examples of constraints include degree boundedness, k-connectivity for a particular value of k, bounded diameter, etc. Examples of cost metrics include total transmit power, maximum transmit power, maximum spreading length etc.

In this paper, we consider a single cost metric, namely the *maximum transmit power* used, and two constraints - *connectivity* and *biconnectivity*. Specifically, we consider the following constrained optimization problems.

Definition II.6: Problem *Connected MinMax Power (CMP)*. Given an M = (N, L), and a least-power function λ, find a per-node minimal assignment of transmit powers p : N $\rightarrow Z^+$, such that the induced graph of (M, λ, p) is connected, and $\mathrm{MAX}_{u \in N}(p(u))$ is a minimum.

[2] An alternate, and arguably superior representation would use directed edges to include unidirectional communication links. Note that we do not *assume* bidirectionality, we simply ignore unidirectional links. Using unidirectional links in an efficient manner requires sophisticated control protocols at several layers and is a subject of current research

Definition II.7: Problem *Biconnectivity Augmentation with MinMax Power (BAMP)*. Given a multihop wireless net M = (N, L), a least-power function λ, and an initial assignment of transmit powers p : N $\rightarrow Z^{+}$, such that the induced graph of (M, λ, p) is connected, find a per-node minimal set of power increases $\delta(u)$ such that the induced graph of (M, λ, $p(u) + \delta(u)$) is biconnected, and $\text{MAX}_{u \in N}(p(u) + \delta(u))$ is a minimum.

We define an assignment to be *per-node-minimal* for connectivity/biconnectivity if and only if it is not possible to lower the assigned power of any single node and still keep the representative graph connected/biconnected. Thus, an assignment in which every node transmits at the solution's maximum power is still inviolate of the MinMax property but may not be per-node-minimal.

Why these particular problems? The most important property of a network is connectivity. A biconnected network, unlike a merely connected one, has the desirable property that the loss of any single node or link will not partition the network. Furthermore, it affords multiple-path redundancy between every pair of nodes enabling fault-tolerance, load balancing or both. The objective of minimizing the *maximum* transmit power rather than the *total* over all nodes is because battery life is a local resource and so collective minimization has little practical value. While the total transmit power used has some bearing on the total interference in the system, the effect of that should be (and is in our work) studied directly using the throughput metric. Finally, per-node minimality ensures a certain "tightness" of the power assignments.

III. Static Networks: Optimum Centralized Algorithms

A static network such as the Metricom Ricochet [1] network affords the luxury of using a centralized or even an offline algorithm to compute the transmit power levels. The node locations, as well as the least-power function are available as input to the algorithm. We present two polynomial-time algorithms, one that results in a connected network, and the other in a biconnected network.

Algorithm CONNECT is given formally in the box below. It is a simple "greedy" algorithm, similar to the minimum cost spanning tree algorithm. It works by iteratively merging connected components until there is just one. Initially, each node is its own component. Node pairs are selected in non-decreasing order of their mutual distance. If the nodes are in different components, then the transmit power of each is increased to be able to just reach the other. This is done until the network is connected. The description assumes for simplicity that network connectivity can be achieved without exceeding the maximum possible transmission powers. However, the algorithm can be easily modified to return a failure indication if this is not true.

While this results, as proven in theorem III.1 below, in a minimum maximum transmit power, it may not be per-node-minimal. This is because a power increase may add

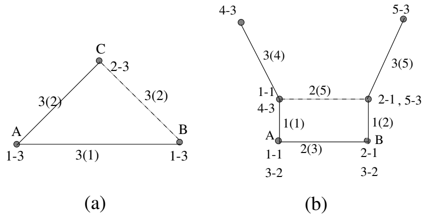

Fig. 1. Illustrating side effect edges. Side effect edges are shown with dashed lines. Legend for nodes is $s-p$, where s is the step number, and p is the power assigned during that step. Legend for edges is $d(s)$, where d is the distance between corresponding nodes, and s is the step during which this edge was formed. Figure (a) is per-node minimal, but in figure (b), the powers of A and B can be reduced back to 1 and still keep the graph connected.

more than one edge to the induced graph. Such additional edges, other than the one between the selected node pairs are called *side-effect* edges. An example of side-effect edge is illustrated in figure 1(a). A side-effect edge may form a loop with other edges and may allow the lowering of some power levels and the elimination of some edges added previously. An example is shown in figure 1(b).

A post-processing phase as given in the procedure *perNodeMinimalize* is the simplest way to exploit side-effect edges and make the assignment per-node minimal. The idea is to consider nodes one at a time and ramp down their powers to the maximum possible extent that does not disconnect the induced graph. But there are theoretically infinite power levels between two power levels, and practically there may be a large number of power levels depending on the granularity of power adjustment that the radio provides.

Our solution uses a binary search over the only set of power levels that "matter", a set that does not depend on the granularity of the power level adjustment. This set is determined by the set of nodes that are within range of the considered node with the current power level. In theorem III.1, we show that this suffices for per-node minimality.

Algorithm CONNECT
Input: (1) Multihop wireless network M = (N, L) (2) Least-power function λ
Output: Power levels p for each node that induces a connected graph

begin
1. sort node pairs in non-decreasing order of mutual distance
2. initialize $|N|$ clusters, one per node
3. **for** each (u,v) in sorted order **do**
4. **if** cluster(u) $\neq$ cluster(v)
5. $p(u) = p(v) = distance(u, v)$
6. merge cluster(u) with cluster(v)
7. **if** number of clusters is 1 **then end**
8. perNodeMinimalize(M, λ, p, 1)
end

Algorithm BICONN-AUGMENT
Input: (1) Multihop wireless network M = (N, L) (2) Least-power function λ (3) Initial power assignment inducing connected network
Output: Power levels p for each node that induces a biconnected graph.

begin
1. sort node pairs in non-decreasing order of distance
2. G = graph induced by (A, λ, p)
3. **for** each (u, v) in sorted order **do**
4. **if** biconn-comp(G,u) $\neq$ biconn-comp(G,v)
5. $q = \lambda$(distance(u,v))
6. $p(u) = \max(q, p(u))$
7. $p(v) = \max(q, p(v))$
8. add (u, v) to G
9. perNodeMinimalize(M, λ, p, 2)

end

Procedure perNodeMinimalize(M, λ, p, k)
begin
1. let S = sorted node pair list
2. **for** each node u do
3. T = $\{ (n_1, n_2) \in$ S : $u = n_1$ or $u = n_2 \}$
4. sort T in non-increasing order of distance
5. discard from T all (x, y) such that $\lambda(d(x, y)) > p(u)$
6. **for** $(x, y) \in$ T using binary search **do**
7. **if** graph with $p(u) = \lambda(d(x, y))$ is not k-connected, **stop**
8. **else** $p(u) = \lambda(d(x, y))$

end

The augmentation of a connected network to a biconnected network is done using Algorithm BICONN-AUGMENT. Once again, it is a greedy technique. We first identify the biconnected components in the graph induced by the power assignment from algorithm CONNECT. This is done using a standard method based on depth-first search given in [12]. Then, node pairs are selected in non-decreasing order of their mutual distance and joined only if they are in different biconnected components. This is continued until the network is biconnected.

A post-processing phase similar to that of Algorithm CONNECT ensures per-node minimality. In this case, the solution may not be per-node minimal even in the absence of side-effect edges. Nonetheless, the same "fix" works, whatever the cause.

We note that, in practice, the per-node-minimality post-processing phases for both CONNECT and BICONN-AUGMENT may be ignored. The few extra edges it introduces may be seen as an advantage. Indeed, if one were to build a biconnected network from scratch (that is, execute BICONN-AUGMENT immediately after CONNECT), there is no reason to make the connected graph per-node minimal. In our implementation, we have omitted per-node minimalization. The topology resulting from the execution of our implementation of CONNECT and BICONNECT on 40 nodes spread out with a density of 2 nodes/sq mile is shown (using the visualization tool "netviz") in figures 3 and 4, respectively. For comparison, figure 5 shows the topology without any topology control (power levels fixed at 30 dBm).

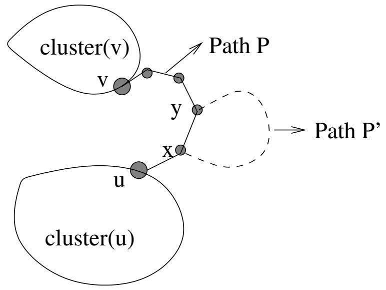

Fig. 2. Illustration for theorem III.1

We now prove the correctness and optimality of the algorithms.

Theorem III.1: Algorithm CONNECT is an optimum solution to the CMP problem.

Proof: Lines 4, 5 create an edge between two nodes if they are in different clusters. Line 7 ensures that if we end then the graph is connected and line 3 ensures that if we end then all node pairs have been considered. Thus, the algorithm is correct.

We first show that the assignment minimizes the maximum power. Following that, we will show that the assignment is per-node-minimal.

Suppose to the contrary that the maximum power used is not the optimum. Consider a node u that is assigned the maximum power. By line 4, this must have happened in order to connect to another node v in a different cluster. Further, since we consider node pairs in non-decreasing order of separation (lines 1,3), there can be no path between u and v such that all nodes along that path are separated by less than $d(u, v)$. That is,

$$\nexists\, path(u, v) : \forall (x, y) \in path(u, v),\ d(x, y) \leq d(u, v) \quad (3)$$

This is because, had there been such a path, algorithm CONNECT would have found it prior to joining u and v, thereby putting u and v in the same cluster and contradicting our assumption. To see this, consider any such path P and consider a pair of consecutive nodes x and y in that path. Either x and y have been joined by algorithm CONNECT in line 5 or they have not. If they have, we are done. If not, it was surely ignored only because there is some other path $P^{'}$ already connecting them, in which case a substitution of $P^{'}$ in place of (x, y) in P results in a path between u and v. See figure 2 for illustration.

By line 5 and distance monotonicity of λ, equation 3 can be rewritten as

$$\nexists\, path(u, v) : \forall x \in path(u, v),\ p(x) \leq p(u) \quad (4)$$

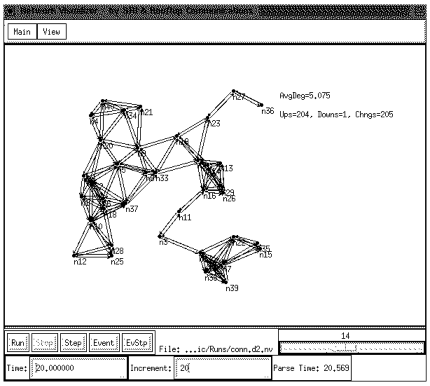

Fig. 3. CONNECTed network.

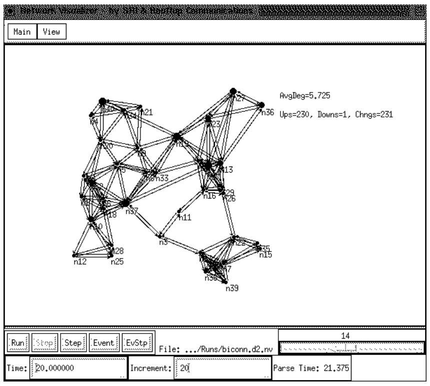

Fig. 4. BICONNECTed network.

Let $p_{opt}(i)$ denote the power of node i under the optimum algorithm. Let OPT be the optimum solution value, that is, OPT is the maximum power in the network under the optimum algorithm. Since by supposition, OPT $< p(u)$, and by definition,

$$\forall_i (p_{opt}(i) \leq OPT < p(u)) \tag{5}$$

we have in particular

$$p_{opt}(u) < p(u) \tag{6}$$

But if this is the case, then, u and v cannot be connected directly in the optimum solution since $p(u)$ is the minimum power required to connect u and v (by definition of λ and line 5). Thus, there must be a path that connects u and v and furthermore by Eq. 5, all such nodes must have powers less than $p(u)$. Formally,

$$\exists\, path(u,v) : \forall(x) \in path(u,v),\ p(x) \leq p(u) \tag{7}$$

This contradicts equation 4. Hence our supposition is false and algorithm CONNECT is optimum. ■

We now show that algorithm CONNECT produces a per node minimal solution. Consider lines 6,7 in procedure perNodeMinimalize. Let $\mathrm{T}_i = (u, v_i)$ be the element for which the graph is disconnected. Thus, $p(u)$ is $\lambda(\mathrm{d}(\mathrm{T}_{i-1}))$. Say $\mathrm{T}_{i-1} = (u, v_{i-1})$.

Since T contains *all* nodes within range of u sorted in non-increasing order of distance,

$$\nexists x : d(u, v_i) \ < \ d(u,x) \ < \ d(u, v_{i-1}) \tag{8}$$

Now, suppose to the contrary that p is not minimal for node u. Clearly, there is another power setting, say $p^{'}(u)$ $< p(u)$ and $p^{'}(u) > \lambda(\mathrm{d}(\mathrm{T}_i))$ such that the network is connected. Since we know that the network is not connected with $p(u)$ at $\lambda(\mathrm{d}(\mathrm{T}_i))$, there must be another node, say x, such that $p^{'}(u)$ reaches x.

Since $p(u) > p^{'}(u)$, we have

$$d(u, v_{i-1}) = \lambda^{-1}(p(u)) > \lambda^{-1}(p^{'}(u)) = d(u,x) \tag{9}$$

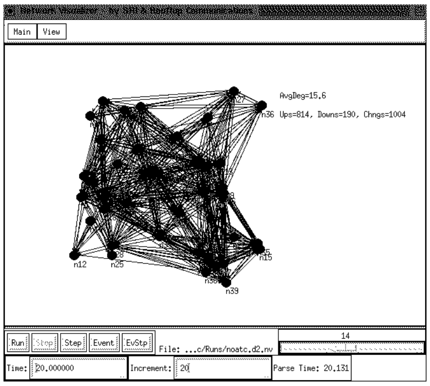

Fig. 5. Without topology control.

Since $p^{'}(u) > \lambda(\mathrm{d}(\mathrm{T}_i))$, we have, taking λ^{-1} of both sides,

$$d(u,x) > d(u, v_i) \tag{10}$$

But equations 9 and 10 contradict equation 8. Thus our supposition must be false, and hence the assignment must be per node minimal.

Theorem III.2: Algorithm BICONN-AUGMENT produces an optimum solution to the BAMP problem.

Proof: The correctness of BICONN-AUGMENT follows from lines 3 and 4 which force nodes to be in the same biconnected component. The proofs for optimality and per-node-minimality are similar to that for theorem III.1, and will be significantly abbreviated for lack of space.

Consider a node u that is assigned the maximum power to form an edge with a node v. Since we consider node pairs in non-decreasing order, following an argument analogous to that of theorem III.1, there cannot exist two disjoint paths between u and v such that all edges in these paths are at most $d(u,v)$. On the other hand, if the optimum is less than $p(u)$, u must have two disjoint paths to v in which each node pair separation is no more than $d(u,v)$, forcing a contradiction. ■

The proof for per node minimality is identical modulo substitution of "connect" for "biconnect" in the corresponding proof for theorem III.1

Theorem III.3: The worst case running time of algorithem CONNECT and BICONN-AUGMENT is $O(n^2 \cdot log(n))$, where n is the number of nodes.

Proof: By straightforward inspection of the algorithms. ■

IV. Mobile Networks : Distributed Heuristics

In a mobile multihop wireless network, the topology is constantly changing. The solution must, therefore, continually re-adjust the transmit powers of the nodes to maintain the desired topology. Further, the solution must use only local or already-available information since updating global information such as positions of all nodes requires prohibitive control overhead. Thus, the centralized solutions of section III are not viable in a mobile context.

In this section, we present two distributed heuristics for topology control, namely Local Information No Topology (LINT) and Local Information Link-State Topology (LILT). Both are zero-overhead protocols, that is, they do not use any special control messages for their operation. The main difference between them is the nature of feedback information used and the network property sought to be maintained. LINT uses locally available neighbor information collected by a routing protocol, and attempts to keep the degree (number of neighbors) of each node bounded. All routing protocols have a neighbor discovery or link determination mechanism that keeps track of the status of links to neighboring nodes. We assume that the neighbor discovery protocol returns only bidirectional neighbors. LILT also uses the freely available neighbor information, but additionally exploits the global topology information that is available with some routing protocols such as link-state protocols.

We note that while LINT and LILT do not explicitly introduce control overhead, the adjustment of transmit power may cause link up/downs. In many routing protocols, this causes routing updates. An excessive number of such topology control induced updates may actually eat up network bandwidth and *decrease* the effective throughput. In order to minimize this, LINT and LILT are *incremental*, in that they calculate the new transmit powers not from scratch, but based on the currently used values.

Due to these constraints, the mechanisms presented in this section are necessarily *heuristics* and offer no guarantees on the worst-case performance. In particular, the power minimization is done in an indirect manner by limiting the number of neighbors, and is at best a poor approximation to an optimal solution.

A. LINT Description

A node is configured with three parameters – the "desired" node degree d_d, a high threshold on the node degree d_h, and a low threshold d_l. Periodically, the node checks the number of active neighbors (degree) in its neighbor table (built by the routing mechanism). If the degree is greater than d_h, the node reduces its operational power. If the degree is less than d_l, the node increases its operational power. If neither is true, no action is taken. The increase or decrease in power is bounded by the maximum and minimum possible power settings of the radio.

The magnitude of the power change is a function of desired degree d_d and current degree d. In particular, the further apart d and d_d are, the more is the magnitude of the change.

The power changes are done in a *shuffle periodic* mode, that is, the time between power changes is randomized around a mean. This is done in order to eliminate lockstep execution and interference between packets.

We now derive the formula used in LINT to reduce the power. It is based on the well-known generic model for propagation [10] by which the propagation loss function varies as some $\mathcal{E}$ power of distance. The value of $\mathcal{E}$ is usually between 2 and 5, depending on the environment. Specifically, if γ is the loss in dB, then,

$\gamma(r) = \gamma(r_{thr})$, if $r < r_{thr}$
$\gamma(r) = \gamma(r_{thr}) + 10{\cdot}\mathcal{E}\cdot\log_{10}(\frac{r}{r_{thr}})$, if $r \geq r_{thr}$

where r is the distance, r_{thr} is a threshold distance below which the propagation loss is a constant $\gamma(r_{thr})$. All logarithms in the remainder of this section are base 10.

Let d_c and p_c denote, respectively, the current degree and current transmit power of a node in a network of density D. We need an expression for the new transmit power p_d so that the node has the desired degree d_d.

Let r_c denote the range of a node with power p_c, and r_d denote range of the node at the targeted power d_d. Assuming a uniformly random distribution of the nodes in the plane,

$$d_c = D{\cdot}\pi{\cdot}r_c^2 \tag{11}$$
$$d_d = D{\cdot}\pi{\cdot}r_d^2 \tag{12}$$

Let T denote the receiver sensitivity of the radio. Then, the following must hold

$$p_c - (\gamma(r_{thr}) + 10{\cdot}\mathcal{E}\cdot log(\frac{r_c}{r_{thr}})) = T \tag{13}$$
$$p_d - (\gamma(r_{thr}) + 10{\cdot}\mathcal{E}\cdot log(\frac{r_d}{r_{thr}})) = T \tag{14}$$

Equating (13) and (14), and substituting for r_c and r_d from (11) and (12) respectively, and simplifying, we get

$$p_d = p_c - 5{\cdot}\mathcal{E}\cdot log(\frac{d_d}{d_c}) \tag{15}$$

A node knows the current used power p_c and the current degree d_c. As mentioned before, d_d is a configured value. In our system, we use $\mathcal{E} = 4$, but $\mathcal{E}$ can also be configured depending upon the environment. Equation (15) can thus be used to calculate the new power periodically. We note that the formula applies for both power increase and decrease to bring the degree close to d_d.

B. *LILT Description*

A significant shortcoming of LINT is its incognizance of network connectivity and the consequent danger of a network partition. For instance, there may be situations, such as when two squads of soldiers move away from each other, in which the degree bound enforced by LINT prevents the connection between the squads.

In multihop wireless routing protocols based on the link-state approach (such as [13], [2], [14]), some amount of global connectivity information is available locally at every node. This is available at no additional overhead to the topology control mechanism. The idea in LILT is to exploit such information for recognizing and repairing network partitions.

There are two main parts to LILT – the neighbor reduction protocol (NRP) and the neighbor addition protocol (NAP). The NRP is essentially the LINT mechanism that tries to maintain the node degree around a certain configured value. The NAP is triggered whenever an event driven or periodic link-state update arrives. Its purpose is to override the high threshold bounds and increase the power if the topology change indicated by the routing update results in undesirable connectivity. The main challenge here is to coordinate such power changes with other nodes, since we do not want all nodes to react to the topology change.

Initially, all nodes start with the maximum possible power. This results in a maximally connected network, enables successful propagation of updates and the initalization of a network topology database at each node. After this initialization, the NRP and NAP are activated, as follows.

A node receiving a routing update first determines which of three states the updated topology is in – disconnected, connected but not biconnected, or biconnected. If it is biconnected, no action is taken. If it is disconnected, the node increases its transmit power to the maximum possible value.

If it is connected, but not biconnected, the node attempts to do biconnectivity augmentation, as follows. The node first finds its distance from the closest *articulation point*. An articulation point is a node whose removal will partition the network. Note that if the network is connected but not biconnected, it must have at least one articulation point. Articulation points are automatically found by the biconnectivity checking procedure. The node then sets a timer for a value t that is randomized around an exponential function of the distance from the articulation point. If after time t the network is still not biconnected, the node increases its power to the maximum possible.

Thus, a limited form of global coordination is achieved with zero overhead. Nodes closer to an articulation point are more likely to remove the articulation and therefore given priority using timers. The coordination is not perfect in that it is possible that the network over-reacts by having two or more nodes increase their power. However, the NRP reduces the powers to an appropriate level in time, and in any case the error is on the conservative (connectivity) side. The NRP and NAP intervals are kept sufficiently large to damp any oscillations. None have been observed in our experiments.

Note that nodes increase their power immediately to the maximum value rather than step by step. This reflects the need for expediency in fixing the connectivity. The goal of biconnectivity is to ensure that even in the transient period, the network will at least be connected.

V. Experimental Results

We have implemented the static and mobile algorithms within an existing prototype multihop wireless network. This system uses a flat *link-state routing* mechanism. Links are determined by a k-out-of-n neighbor discovery scheme. Event driven and periodic updates are flooded throughout the network and routes are generated using Dijkstra's shortest path algorithm. For details on this system, please refer [13].

Our implementation uses the C++ Toolkit (CPT) framework developed by Rooftop Communications. Within this framework, emulations of hardware and simulation models of the channel can be easily swapped for easy transition between the real network and its simulation. Thus, the level of detail is very high at all layers of the stack and the topology control and routing software for the simulation is identical to the code that runs in the embedded system. Consequently, the fidelity of our simulation and the confidence in our results is very high.

In the remainder of this section, we describe the radio and its emulation, the mobility and propagation models used, and discuss the observations from our experiments.

A. *Radio and its emulation*

Each node in our testbed consists of a radio modem and a router module connected using a serial interface. The radio we use is Utilicom Longranger 2020 [9]. It is a direct sequence spread spectrum radio in the 900 MHz ISM band capable of a raw data rate of about 300 Kbps. Compared to the popular WaveLAN radios, this has a lower data rate but a much higher transmission range (about 6 miles). Also, unlike the WaveLAN, the Utilicom has transmit power control, and is the basic reason for our choice. It uses the CSMA protocol for channel access. The router module is supplied by Rooftop Communications, and has an embedded Motorola 68360 processor that runs networking software that resides in Flash memory.

In order to experimentally study our topology control and routing algorithms for a relatively large network, we have a software emulation of the radio and its MAC-layer protocol. The emulation is very detailed and models radio features such as receiver sensitivity, transmit/receive turnaround time, framing and preamble bits, capture, carrier and interference thresholds, processing gain etc. In our experiments, the radio is replaced by its emulation and the channel is modelled as described in the next section.

The CSMA protocol used by the radio is modelled in full in our simulation. A node backs off randomly when sensing the channel to be busy. After every packet transmission, the node waits a small amount of time for fairness. This protocol is susceptible to the hidden and exposed terminal problems.

B. Mobility and propagation models

We use a psuedo-random mobility model. Nodes are initially randomly placed in a square area determined according to the *density* parameter. They then move in a random direction for a while, change their directions randomly, move in that direction for a while and so on. The speed of each node in all experiments reported here is 72 miles/hour.

The propagation model used is taken from [15]

$$\gamma(d) = 156 + 40 \cdot \log(d) - 15 \cdot \log(h1/h2) - (g1 + g2)$$

where d is the distance in one or miles, $h1$ and $h2$ are the heights of the antennas in feet (set to 20), and g1 and g2 are the antenna gains (set to 3 dB). If d is less than a mile, γ is the distance independent factor above, and equals 111 dB.

In order to determine whether or not a transmission is successfully received at a node n, the signal strength from all other simultaneous transmissions at node n is taken as interference. The result depends upon the relative strengths of the good and interfering signals, capture effect, and the receiver parameters.

C. Performance metrics

Traffic is offered to the mobile multihop wireless network using traffic generators at each node. For all of the results presented in the network, 12 streams were used, between randomly chosen source-destination pairs. Each stream consisted of 256-byte packets and the inter-arrival time was uniformly distributed around a mean rate of 4 Kbps per stream. Given the channel access protocol, CSMA, the aggregate 48 Kbps represents an aggressive load on the network.

Each packet has a sequence number and a timestamp which enables packet loss and delay to be tracked. We study four performance metrics.

1. *Throughput.* The fraction of packets sent by any source that was successfully received at the intended destination.
2. *Delay.* The average time elapsed, for all successful packets, between the packet being sent by a source and it being received.
3. *Maximum transmit power.* The maximum over all nodes, of the transmit power used by a node (relevant only for static networks).
4. *Average transmit power.* The average over all nodes, of the transmit power used by a node (relevant only for static networks).

We study the dependence of these metrics on network density, defined as the number of nodes per unit area. We chose this, rather than, say the network size because topological properties (e.g, how well a network is connected) are more critically dependent on density changes than on size increases with constant density.

D. Observations

All of our experiments were based on a 40 node network that was run for about 3 minutes of simulation time.

D.1 Static Networks

We compared algorithms CONNECT and BICONN-AUGMENT to no topology control (curves marked NONE in the plots).

Figure 6 shows the dependence of throughput on the density. When no topology control is employed (the transmit power was fixed at 30 dBm for all nodes), the throughput is acceptable only for a small range of density values. For greater than 1.5 nodes/sq mile, interference reduces spatial reuse and hence capacity. For less than 0.5 nodes/sq mile, the network is poorly connected. Algorithm BICONN gives the best throughput and adapts very well to changing density to maintain consistency. At densities above 1 node/sq mile, it improves the throughput significantly – as much as 227 % at 4 nodes/sq mile. Algorithm CONNECT suffers from congestion "hotspots" at low densities whereas at higher densities, side-effect edges make the graph look more and more like a biconnected one and hence the throughput approaches that of algorithm BICONN.

Figure 7 shows the dependence of the average and maximum transmit power used by the network on the density. We have shown the NONE curve as reference (power used by all nodes corresponding to the NONE curve of figure 6). Algorithm BICONN uses significantly more power than CONNECT at lower densities (at density 1 node/sq mile, it uses about 50 % more average power and 100 % more maximum power). At low densities, BICONN has to increase the powers of some "isolated" nodes so that there are no articulation points. As can be seen from the difference between maximum and average, only a few nodes have close-to-maximum powers.

Our observations show clearly that the effect of even a simple topology control algorithm on throughput is significant. We also infer that at high densities, it is better to use BICONNECT rather than CONNECT, whereas at low densities, the choice depends on which is more important - battery power or throughput. In practice, instant infrastructure networks may be deployed at different densities, and commercial networks such as the Metricom Ricochet will be made denser to accommodate more users. Our results help maximize the throughput for all of these scenarios.

D.2 Mobile Networks

There are two important differences between LINT/LILT on mobile networks and CONNECT/BICONNECT on static networks that predictably mitigate the gains that

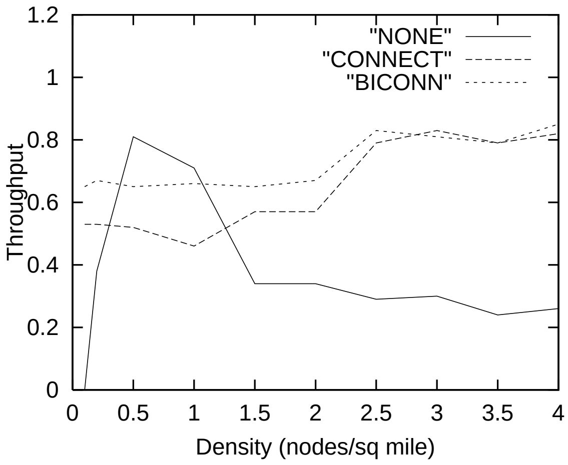

Fig. 6. Throughput vs density : Static networks

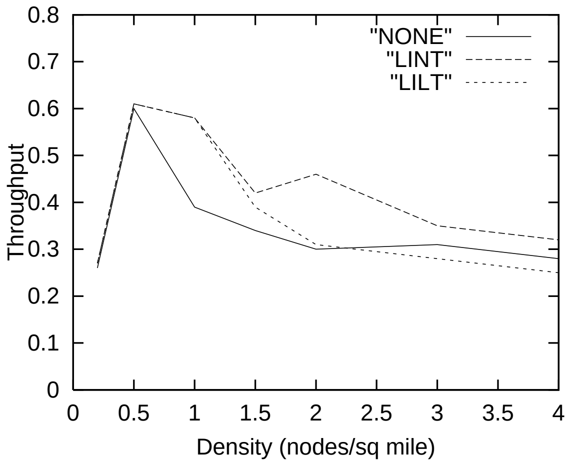

Fig. 8. Throughput vs density: Mobile networks

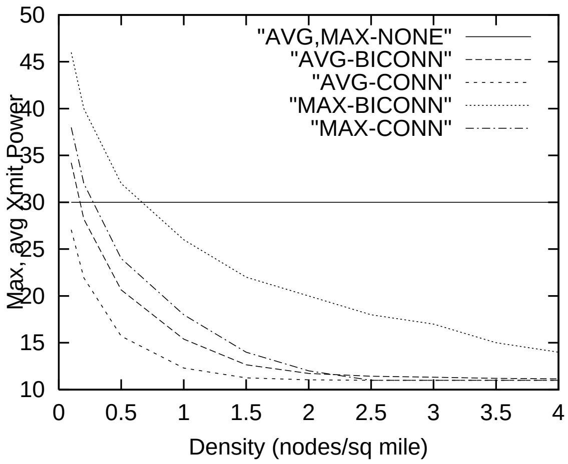

Fig. 7. Power vs density: Static Networks

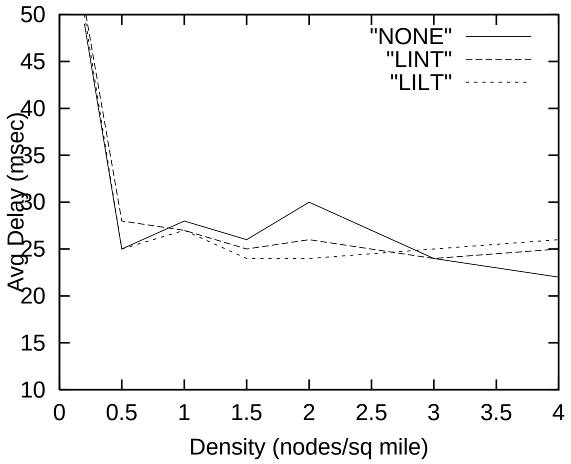

Fig. 9. Delay vs density: Mobile networks

LINT/LILT provide for mobile networks. First, since both LINT and LILT are reactive schemes, there are periods (between adjustments) in which the topology may be very undesirable. Second, as mentioned in section IV the repeated changes in transmit powers in turn introduce link-state updates and increase the routing overhead. In low capacity multihop networks such as the one studied, this is a non-trivial factor.

From figure 8, it can be seen that above a density of 1 node/ sq mile (after which point the network is connected), increased density causes a decrease in throughput in all cases. Both LINT and LILT cause nodes to lower their powers appropriately to reduce interference and improve throughput. At a density of 2, the throughput gain for LINT is approximately 53 %. LINT does better than LILT. One reason for this is that the radios use CSMA - a protocol that is not only poor at high loads but also suffers from the hidden terminal problem[3]. Because of this, the link-state database is often not up-to-date causing false alarms and an increase in the transmit power. The situation becomes worse at higher densities as there are more links to deal with. Further, our mobility model did not really bring out the drawback of LINT (mentioned in section IV-B) that LILT overcomes. Work with such mobility models is in progress.

The delay dependence is given in figure 9. There are only slight gains in delay performance, and that too only at the middle ranges in densities. One reason that not having topology control appears competitive is because we only take into account the delay for *successful* packets and if a packet is successful, it has typically gone over lesser number of hops when there is no topology control. Since, as discussed above, LILT generally operates at higher power levels than LINT, the delay is lesser for LILT. Note however that all of these delays are considerably less than 200 msec, which is generally regarded as the tolerable latency for real-time (voice) communications.

Although not reported here due to lack of space, we also measured the number of link state updates generated with-

[3]Note that the channel access is not our choice - we used what came with the COTS radio. A different channel access protocol may well improve LILT in comparison to LINT.

out topology control, with LINT and with LILT. We found that having LINT or LILT reduced the number of link state updates by a factor of 2 to 3, for densities above 1 node/sq mile.

VI. Concluding Remarks

Unlike wireline networks whose physical topology is inflexible, multihop wireless network topology can be controlled using node parameters such as transmit power, antenna direction, etc. With the advent of commerical radios that offer more sophisticated controls on transmission parameters, this aspect of wireless networks can be and must be exploited. This is a very challenging problem, especially in mobile networks, and one that we believe could be a source of exciting new research in multihop wireless networks.

Using (bi)connectivity as our objective, we have described optimal centralized algorithms and distributed heuristics for transmit power control. Of equal importance, however, is the new representational framework for studying this problem that we presented in section II. As new problems with new objectives appear, they could be formulated within this new framework. We have also experimentally studied our topology control mechanisms in the context of a prototype multihop wireless system and improved its throughput and power consumption significantly.

Our work is relevant to commercial as well as military networks. Our static network algorithms can significantly improve the throughput and battery life of infrequently mobile (or portable) instant-infrastructure networks. They are also well suited to commercial data service providers such as Metricom's Ricochet network. The Ricochet network [1] uses wireless repeaters mounted on poletops to form a multihop wireless mesh, and provides Internet service for mobile end users. Our distributed heuristics are widely applicable to civilian and military mobile multihop wireless networks, to increase their capacity, battery life and connectivity. Our experimental results are particularly useful in selecting the right solution for a given operating environment, and also for deciding an optimum operating density.

References

[1] http://www.metricom.com
[2] http://www.gordon.army.mil/tsmtr/ntdr.htm
[3] S. Ramanathan, M. Steenstrup, "A survey of routing techniques for mobile communications networks", *ACM/Baltzer Mobile Networks and Applications* 1 (1996) 89-104.
[4] L. Hu, "Topology Control For Multihop Packet Radio Networks", Proc. IEEE Infocom, 1991.
[5] H. Takagi, L. Kleinrock, "Optimal transmission ranges for randomly distributed packet radio terminals", *IEEE Transactions on Communications*, COM-32, No. 3, March 1984.
[6] T. Hou, V.O.K. Li, "Transmission range control in multihop radio networks", *IEEE Transactions on Communications*, COM-34, No. 1: 38-44, Jan. 1986.
[7] M. Gerla, J.A.S. Monteiro, R. Pazos, "Topology Design and Bandwidth Allocation in ATM Nets", *IEEE JSAC*, Vol. 7, No. 8, Oct. 1989.
[8] A. Farago, I. Chlamtac, S. Basagni, "Virtual Path Topology Optimization Using Random Graphs", Proc. IEEE Infocom '99, New York.
[9] http://www.utilicom.com
[10] T. S. Rappaport, *Wireless Communications, Principles and Practice*, Prentice-Hall, 1996.
[11] F. Harary, *Graph Theory*, Addison-Wesley, 1972.
[12] R. Sedgewick, *Algorithms*, Addison-Wesley, 1983.
[13] S. Ramanathan, M. Steenstrup, "Hierarchically-organized, Multihop Mobile Networks for Multimedia Support", *ACM/Baltzer Mobile Networks and Applications*, Vol. 3, No. 1, pp 101-119.
[14] J.J. Garcia-Luna-Aceves, J. Behrens, "Distributed, Scalable Routing Based on Vectors of Link States", *IEEE Journal on Selected Areas in Communications*, October 1995.
[15] W.C.Y. Lee, "A Computer Simulation Model for the Evaluation of Mobile Radio Systems in a Tactical Environment"