Minimum Energy Mobile Wireless Networks

Volkan Rodoplu, *Student Member, IEEE,* and Teresa H. Meng, *Fellow, IEEE*

***Abstract*—We describe a distributed position-based network protocol optimized for minimum energy consumption in mobile wireless networks that support peer-to-peer communications. Given any number of randomly deployed nodes over an area, we illustrate that a simple local optimization scheme executed at each node guarantees strong connectivity of the entire network and attains the global minimum energy solution for stationary networks. Due to its localized nature, this protocol proves to be self-reconfiguring and stays close to the minimum energy solution when applied to mobile networks. Simulation results are used to verify the performance of the protocol.**

***Index Terms*— Distributed algorithms, energy management, graph theory, mobile communication, network fault tolerance, networks, packet radio, portable radio communication, power measurement, protocols, radio repeaters.**

I. Introduction

THIS paper describes a distributed network protocol optimized for achieving the minimum energy for randomly deployed ad hoc networks. The network protocol not only maintains a globally connected network in spite of possible module failure, but also defines the major power management strategy based on low-power RF transceiver design. Minimum energy consumption in portable communication devices has been one of the major design goals, if not the most important one, in recent IC designs [9], [10]. In wireless communication systems, the need for low power becomes even more pronounced when designing RF transceivers for small-sized portable user sets [3], [20].

For wireless network designers, on the other hand, the emphasis has traditionally been on increasing system capacity (e.g., the number of users a base station can support), maximizing point-to-point throughput in packet-switching networks, and minimizing network delay [7], [19].

Our thesis is that significant reductions in energy consumption can be achieved if wireless networks are designed specifically for minimum energy. In order to maximize the total battery life of a wireless network, we must minimize the energy consumption of the entire network.

Applications where minimum energy networking can effect significant benefits include the digital battlefield, where soldiers are deployed over an unfamiliar terrain, and multisensor networks, where sensors communicate with each other with no base station nearby. Even in the presence of base stations, such as in cellular phone systems, minimum energy network design can allow longer battery life and mitigate interference.

In this paper, we present a position-based algorithm to set up and maintain a minimum energy network between users that are randomly deployed over an area and are allowed to move with random velocities. We denote these mobile users by "nodes" over the two-dimensional plane. Our network protocol reconfigures the links dynamically as nodes move around, and its operation does not depend on the number of nodes in the system.

Each mobile node is assumed to have a portable set with transmission, reception, and processing capabilities. In addition, each has a low-power global positioning system (GPS) receiver on board, which provides position information within at least 5 m of accuracy [12]. The recent low-power implementation of a GPS receiver [17] makes its presence a viable option in minimum energy network design.

There have been only a few works in this area so far, most notably the work by Scott and Bambos. In their recent paper [16], they proposed a routing and channel assignment scheme for low power transmission in personal communication systems (PCS). Our work differs in the following respects.

1) We do not assume a fixed and connected network topology. Instead, we introduce a local optimization procedure that finds the minimum energy links and dynamically updates them.
2) We show that our protocol is self-reconfiguring in mobile scenarios.

The GeoCast scheme proposed by Navas and Imielinski [11] for geographic addressing and routing is also based on the availability of GPS position information. There are three major differences between their work and ours: 1) GeoCast assumes an existing wired infrastructure while our scheme assumes no underlying infrastructure or protocols; 2) GeoCast assumes fixed routers with stationary distribution areas (polygons) while our protocol, instead, is designed for mobile nodes; 3) GeoCast does not address energy considerations—in our work, energy consumption is the key metric.

The rest of the paper is organized as follows. Section II discusses the network layer requirements that the network protocol must satisfy. Section III gives preliminaries on outdoor radio propagation and describes the intuition that underlies this work. In Section IV, we develop a theory of minimum energy for stationary networks and prove this notion in a rigorous mathematical setting. In Section V, we present our distributed network protocol. Section VI gives an example of a point-to-point connection formed by applying this theory. In Section VII, we set up a stationary network simulator and measure energy consumption as a function of the number of

Manuscript received June 10, 1998; revised January 10, 1999. This work was supported in part by DARPA.

The authors are with the Department of Electrical Engineering, Stanford University, Stanford, CA 94305 USA (e-mail: vrodoplu@stanford.edu; meng@mojave.stanford.edu).

Publisher Item Identifier S 0733-8716(99)04800-3.

nodes in a distributed network. In Section VIII, we apply this distributed protocol to mobile networks and show that it is self-reconfiguring. Section IX demonstrates the low energy performance of the protocol for mobile networks.

II. Network Layer Requirements

In peer-to-peer communications, each node is both an information source and an information sink. This means that each node wishes to both send messages to and receive messages from any other node. An important requirement of such communications is strong connectivity of the network. A network graph is said to be "strongly connected" if there exists a path from any node to any other node in the graph [8]. A peer-to-peer communications protocol must guarantee strong connectivity.

For mobile networks, since the position of each node changes over time, the protocol must be able to dynamically update its links in order to maintain strong connectivity. A network protocol that achieves this is said to be "self-reconfiguring." A major focus of this paper is the design of a self-reconfiguring network protocol that consumes the least amount of energy possible.

In order to simplify the discussion of our protocol, we take one of the nodes to be the information sink for all nodes in the network. We call this node the "master-site." The master-site can be thought of as the headquarters located at the edge of the digital battlefield, the supervisory station in a multisensor network, or the base station in a cellular phone system. All of these scenarios are special cases of peer-to-peer communications networks.

Each node knows its own instantaneous position via GPS, but not the position of any other node in the network, and its aim is to send its messages to the master-site whenever necessary.

A protocol that solves the minimum energy problem with a single master-site simultaneously solves the general peer-to-peer communications problem because each node can independently be taken as a master-site, and the optimal topologies can be superimposed. We take advantage of this simplification and concentrate on the problem with a single master-site without loss of generality.

III. The Power Consumption Model

Modeling the radio channel has always been one of the most difficult parts of the design of terrestrial wireless communication systems. Typically, the channel variations are characterized statistically and are grouped into two broad categories: large-scale and small-scale variations. Large-scale propagation models are used to predict the mean signal power for any transmitter-receiver separation. Small-scale signal models characterize the rapid fluctuations of the received signal strength over very short travel distances [14].

We will consider the most common channel model used for RF systems. In practice, many channels have been found to fit this model well with appropriate parameters from field measurements [1], [2], [13], [14]. This model has the following components.

1) Path loss: the received signal power averaged over large-scale variations has been found to have a distance dependence which is well modeled by $1/d^n$, where d denotes the distance between the transmitter and receiver antennas, and the exponent n is determined from field measurements for the particular system at hand [14].
2) Large-scale variations: these are modeled by the log-normal shadowing model. In this model, the received signal power averaged over small-scale variations is statistically described by a lognormal distribution with the distance-dependent mean obtained from the path loss calculation [14].
3) Small-scale variations: these are modeled by a Rayleigh distribution. In the Rayleigh model, the received signal is a wide-sense stationary stochastic process whose amplitude at each point in time is a Rayleigh random variable [5], [13], [14].

Typically, a wireless communication receiver is designed with diversity reception to combat small-scale variations. Diversity reception means that the receiver can collect streams of the same data that have traveled through independent paths. A widely used diversity technique is the Rake receiver in spread-spectrum communication systems, which collects multipath components at intervals of the chip period [13], [14].

A technique called maximum ratio combining (MRC) is used to optimally combine these independent streams. In a full Rake receiver, all multipath components are collected and combined optimally.

In well-designed multiuser communication systems, small-scale variations are therefore handled by diversity techniques and combiners at the physical layer. The only parameter of consequence to designing power-efficient network topologies at the upper layers is the power of the received signal after MRC, which is determined only by path loss and large-scale variations but not small-scale variations.

Typically, in multiuser system designs, a tolerable outage probability is specified for large-scale variations [14]. For instance, in a cellular phone system, it may be required that the received signal power after MRC stay above a certain detection threshold 99% of the time (or with outage probability 0.01). If there is only a single transmitter to transmit the signal (e.g., no base station diversity), this transmitter can adjust its transmit power to satisfy the outage probability specification.

We show in Appendix A that a minimum-power network design that addresses the increase in transmit power to handle large-scale variations is fundamentally the same as a design that considers only the path loss. In order not to obscure the mathematics with outage probabilities, we have chosen to place this part in Appendix A.

In the path-loss model, the path loss may normally depend on the heights of the transmit antennas as well as the transmitter–receiver separation [14]. In this paper, we assume that the mobile devices have similar antenna heights so that this variation in the third dimension can be ignored. For example, in an ad hoc network made up entirely of users carrying hand-held devices, this assumption is justified.

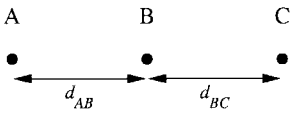


Fig. 1. Three colinear nodes A, B, C.

Therefore, we will concentrate only on path loss that is distance-dependent in our network configuration algorithm. Our algorithm does not depend on the particular value of the path loss exponent n ($n \geq 2$ for outdoor propagation models [14]) and thus offers the flexibility to be applied in various propagation environments.

Our main observation is this: since the transmit power falls as $1/d^n, \quad n \geq 2$, as given by the path loss model, relaying information between nodes may result in lower power transmission than communicating over large distances.

As a simple illustration, consider three nodes A, B, and C on a line, as in Fig. 1. Assume that all three nodes use identical transmitters and receivers. Node A wants to send a message to C. Let t denote the predetection threshold (in mW) at each receiver. In other words, the minimum power that a transmitter must radiate in order to allow detection at distance d meters away is td^n, where n is the exponent in the path loss model. Assuming that node A knows the positions of B and C, it has two options: it can transmit the signal directly to C, which entails a power consumption of td_{AC}^n at node A, or it can relay the message through node B and have it retransmit it with the minimum power needed for B to reach C. In this second case, the total transmit power consumption is $td_{\mathrm{AB}}^n + td_{\mathrm{BC}}^n$. In the case of three colinear nodes, it is easily seen that relaying the message through the middle node always comes at a lower total transmit power consumption than transmitting directly.

When the three nodes are allowed to lie on a two-dimensional plane, which is denoted by $\Re^2$, the option that costs less total power becomes a function of where the receive node is positioned. In the next section, we find the positions for the receive node, where relaying will always consume less total power than transmitting directly.

There is another source of power consumption that must be considered in addition to path loss. In the previous example, when node A relays through B, node B has to devote part of its receiver to receive and store node A's message. This additional power will be referred to as the receiver power at the relay node and will be denoted by c. Each relay induces an additional receiver power to be consumed at the relay node. For the previous example, the total power consumption, including transmit and receiver power consumption in the transmission, is thus $td_{\mathrm{AB}}^n + td_{\mathrm{BC}}^n + c$ when node B is used as a relay.

A third component in power consumption is the power required to process the signal. In this case, the relayed signal is simply buffered. Additional power will also be consumed when running the algorithm that we propose. In the design of modern processors, however, the power consumption required for such processing and computation can be made negligible compared to transmit and receiver powers [6], [18]. Therefore, our power consumption model will ignore this third component.

Based on the observation concerning relays, we will first tackle the problem of finding the minimum power topology in a network where the nodes are stationary. Our main goal is to arrive at an algorithm that requires only local computation for updates and requires as little global information as possible. A protocol requiring only local information is extremely advantageous for networks with mobile nodes since delays associated with disseminating global information would be intolerable. From the perspective of power consumption, a distributed protocol running almost exclusively on local information requires transmission only over small distances. This in turn conserves the total power required for transmitting that information. A third advantage of the use of only local information is that it reduces the interference levels dramatically, since a user's communication with only nodes in its immediate surroundings causes little interference to nodes further away.

IV. Minimum Power Networks

In this section, we develop a general mathematical theory that will eventually lead to the design of a minimum power topology on a stationary network. First, we investigate the implications of our power consumption model. We show that power-efficient transmission can be achieved by each node when it considers only its immediate locality, which is called its enclosure. One of the key results is that if every node maintains communication links with the nodes in its enclosure, the resulting network is strongly connected. Then, we introduce definitions that will help us describe a protocol in the next section which is based entirely on the key results of this section. The proofs of all the lemmas and theorems of this section are given in Appendix B.

In order to investigate the implications of local information on power-efficient transmission, we consider three nodes in $\Re^2$, denoted by i, r, and j. Node i is a node that wishes to transmit information to node j. Accordingly, node i is called the "transmit node" and node j the "receive node." Node i considers the third node r to be used as a relay for transmission from i to j. Node r is called the "relay node." Our aim is to transmit information from i to j with minimum total power incurred by i, j, and r. By varying the position of j, we investigate under which conditions it consumes less power to relay through r. Below, the position of j is denoted by (x, y).

Definition 1—Relay Region: The relay region $R_{i\to r}$ of the transmit-relay node pair (i, r) is defined to be

$$R_{i\to r} \equiv \left\{(x, y) | P_{i\to r\to(x,y)} < P_{i\to(x,y)}\right\}$$

where $P_{i\to r\to(x,y)}$ denotes the power required to transmit information from node i to (x, y) through the relay node r, whereas $P_{i\to(x,y)}$ denotes the power required to transmit information from i to (x, y) directly.

Fig. 2 illustrates a typical relay region in a propagation environment with $1/d^4$ transmit power rolloff.

Lemma 1—Asymptotic Behavior of Relay Regions: Let $B(i, r)$ be the boundary of $R_{i\to r}$. Let the relay node r be located at the origin and the transmit node i be located on the $+x$ axis. Let (x_B, y_B) represent a point on $B(i, r)$. For propagation laws with $1/d^n$, $n > 2$ transmit power rolloff,

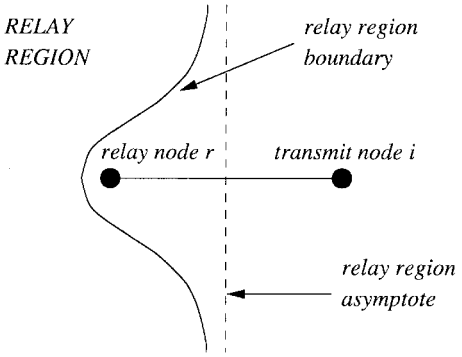


Fig. 2. Relay region of the transmit-relay node pair (i, r).

as $y_B \to \pm\infty$, $x_B \to d_{ir}/2$ where d_{ir} denotes the distance between i and r. For $n = 2$, $B(i, r)$ is given by $x_B = -K$, where K is a positive constant.

We now introduce an ease of notation. Let (x_j, y_j) denote the position of node j on $\Re^2$. For a particular transmit node i that we will specify, we will denote the relation $(x_j, y_j) \in R_{i\to k}$ by $j \in R(k)$. We use this new notation in the following lemmas.

Lemma 2—Distance Properties of Relay Nodes: Let i be the transmit node, r the relay node, and j the receive node. If $j \in R(r)$, then:

2.1) $d_{jr} < d_{ij}$;
2.2) $d_{ij} > d_{ir}$.

Lemma 3—Properties of Relay Regions: Let the transmit node be denoted by i. Relay regions of i have the following properties.

1) $k \notin R(k)$ for any $k \neq i$.
2) If $k \in R(j)$, then $j \notin R(k)$.
3) If $k \in R(j)$ and $l \in R(k)$, then $j \notin R(l)$.

We now consider a finite set $\aleph$ of randomly deployed stationary nodes over $\Re^2$. In the development below, i denotes any node that wishes to transmit information. In a real application, the nodes will be distributed over some finite area. We could designate a rectangular area that includes all the nodes as the deployment region. However, specifying the exact shape of the deployment region would unnecessarily restrict us. In order to keep the theory applicable in general, it suffices to define the deployment region as any bounded region that includes the nodes in it. This definition includes the special case of a rectangular area on which the nodes can be considered being deployed.

Definition 2—Deployment Region: Any bounded set in $\Re^2$ that has the positions of the nodes in $\aleph$ as a subset is said to be a deployment region for the node set $\aleph$.

The reason for explicitly introducing a deployment region in the discussion is that in practice, there is a finite area beyond which no nodes should be looking for neighbors with which to communicate. The boundaries of deployment regions can also be taken as known and impenetrable obstacles to communication. Then, the nodes near the edges can use this fact not to search unnecessarily beyond the boundary of the deployment region.

We next introduce two important definitions: enclosure and neighbor. The main idea behind enclosure is illustrated in Fig. 3. The figure is drawn from the perspective of node i which has found three other nodes $\{k_1, k_2, k_3\}$ in its surroundings. Node i can compute the relay region with each of the three nodes it has found. The three relay regions computed this way are illustrated in Fig. 3. This in turn specifies a region around i, beyond which it is not power efficient for i to search for more neighbors. This follows directly from the definition of relay regions. This bounded region around i is the region of enclosure or simply the enclosure of i. The formal definition below includes the deployment region $D_{\aleph}$ to limit the enclosure to within the deployment region since the deployment region is the only region of interest.

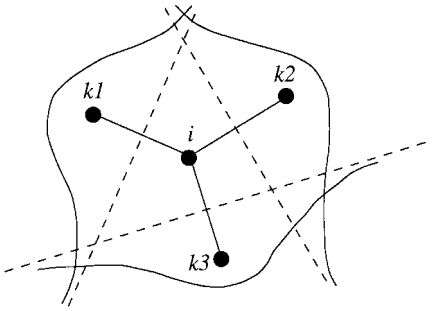


Fig. 3. Enclosure of node i.

Some new nodes that i finds may lie in the relay regions of previously found nodes. Then, it is not power-efficient to transmit directly to these new nodes, and thus i can simply eliminate them from consideration. Thus, i keeps only those nodes that are in its enclosure. The nodes that lie in the enclosure of i will be called the neighbors of i, and these will be the only nodes to which i will maintain communication links for power-efficient transmission. The following definition formalizes these concepts in a more general setting.

Definition 3—Enclosure and Neighbor: The enclosure of a transmit node i is defined as the nonempty solution ε_i to the set of the equations

$$\varepsilon_i = \bigcap_{k \in N(i)} R^c_{i\to k} \bigcap D_{\aleph}$$

and

$$N(i) = \{n \in \aleph | (x_n, y_n) \in \varepsilon_i, n \neq i\}.$$

Above, A^c denotes the complement of any set A, and $D_{\aleph}$ denotes the deployment region for the node set $\aleph$. Each element of $N(i)$ is said to be a "neighbor" of i and $N(i)$ is called the "neighbor set" of i.

Notice that the enclosure of i is bounded since $D_{\aleph}$ is bounded. We will show in the next section that the pair $(\varepsilon_i, N(i))$ exists and is unique, by presenting an algorithm that computes this unique solution.

Definition 4—Enclosed Node: A node i is said to be enclosed if it has communication links to each of its neighbors and to no other node.

The main goal of the next section will be to compute a sparse and strongly connected graph of communication links between all the nodes. This graph will be computed from only

local information, and the existing links will be only between nodes that are close enough to be neighbors (as the term is used in Definition 3). This sparse graph of communication links between neighbor nodes is called the enclosure graph.

Definition 5—Enclosure Graph: The enclosure graph of a set of nodes $\aleph$ is the graph whose vertex set is $\aleph$ and whose edge set is

$$\bigcup_{i\in\aleph}\bigcup_{k\in N(i)} l_{i\to k}$$

where $l_{i\to k}$ is the directed communications link from i to k.

As mentioned in Section II, an important requirement for peer-to-peer communications on an ad hoc network is that the network be strongly connected. The following theorem shows that the enclosure graph satisfies this requirement.

Theorem 1—Strong Connectivity: Fix the deployment region $D_{\aleph}$ for a set of nodes $\aleph$. The enclosure graph of $\aleph$ is strongly connected.

Finally, we would like to find a graph that not only is strongly connected, but also contains only the minimum-power paths from each node to the master-site. This optimal spanning tree that has the master-site as its root will be called the minimum power topology. In Theorem 4, we will show that the minimum power topology is necessarily contained in the enclosure graph and can thus be found by dropping the nonoptimal links of the enclosure graph.

Definition 6—Minimum Power Topology: A graph on the stationary node set $\aleph$ is said to be a minimum power topology on $\aleph$ if:

1) every node has a directed path to the master-site;
2) the graph consumes the least total power over all possible graphs on $\aleph$ for which 1) holds.

V. Distrubuted Network Protocol

In this section, we describe a distributed network protocol that finds the minimum power topology for a stationary set of nodes with a master-site. The main idea in this protocol is that a node does not need to consider all the nodes in the network to find the global minimum power path to the master-site. By using a very localized search, it can eliminate any nodes in its relay regions from consideration and pick only those few links in its immediate neighborhood to be the only potential candidates.

We divide the protocol into two parts: first, a local search executed by each node to find the enclosure graph, and second, a cost distribution from the master-site to every node. The cost metric is the total power required for a node to reach the master-site along a directed path.

A. Phase 1—Search for Enclosure

In order for the protocol to find the enclosure graph, each node must find its enclosure and its neighbor set. Since computing enclosure requires knowledge of the positions of nearby nodes, each node broadcasts its position to its search region. The search region is defined to be the region where a node's transmitted signal (and hence its position) can be correctly detected by any node in that region.

We first introduce a conceptual tool that makes the description of the search algorithm easier. When searching for neighbors, a node must keep track of whether a node found is in the relay region of previously found nodes in the search. The relay graph defined below is in effect a data structure which stores this information.

Definition 7—Relay Graph of a Node: Let A denote the set of all nodes that transmit node i has found thus far in its search. Let j and k be two nodes in A. Whenever $k \in R(j)$, we form a directed edge from j to k and denote it by $e_{j\to k}$. The relay graph of a transmit node i is defined to be the directed graph whose vertex set is A and whose edge set is

$$\bigcup_{j\in A}\bigcup_{k\in R(j)} e_{j\to k}.$$

The relay graph of i is denoted by $G(i)$.

It is important to note that $e_{j\to k}$ represents a relation between j and k based on their positions. It indicates that k lies in the relay region $R_{i\to j}$. It does not represent a communication link between j and k.

Lemma 4—No Cycles on the Relay Graph: The relay graph of a transmit node i has no cycles.

We now describe a localized search algorithm executed by each node, which finds $N(i)$, namely the neighbor set of i.

We will give the intuition behind the search algorithm before we state it precisely. Each node in the algorithm starts a search by sending out a beacon search signal that includes the position information for that node. Since every node runs exactly the same algorithm, we will concentrate on a particular node and call it the transmit node. The transmit node also listens for signals from nearby nodes. When it receives and decodes these signals, it finds out the positions of the nearby nodes and calculates the relay regions for them. As we described in the discussion preceding the definitions of enclosure (Definition 3) and the relay graph (Definition 7), the transmit node must keep only those nodes that do not lie in the relay regions of previously found nodes. Therefore, each time new nodes are found, the transmit node must update its relay graph.

The nodes that have been found thus far in the neighbor search fall into two categories: if a node found (call it node k) falls in the relay region of some other found node (call it j), then we mark k "dead." We say that j "blocks" k. This is simply terminology we introduce to keep track of the state of the nodes on the relay graph. If a node is not blocked by any other node found in the search, then we mark that node "alive." The set of alive nodes when the search terminates constitutes the set of neighbors for transmit node i. In effect, when the search terminates, the transmit node is enclosed, and the nodes that enclose the transmit node are not in the relay region of any node found. Therefore, this satisfies the definition of neighbor (Definition 3).

We will need an auxiliary function called *FlipAllStates-DownChain* in order to update the relay graph. This function is necessary to handle the following situation: at some point in the algorithm, assume that a node denoted by k was blocked only by one node called j. Then, in the next iteration, assume that a new node p blocks j but not k. In this case, k should be revived since it is no longer blocked by any node. In fact, there

```
F = ∅;
A = ∅;
AliveNodes = ∅;
DeadNodes = ∅;
while(TRUE){
    S = SetSearchRegion();
    F = F ∪ S;
    M = {n | (x_n, y_n) ∈ S, n ∉ A, n ≠ i};
    A = A ∪ M;
    foreach(m ∈ M){
        MarkDead(m);
        foreach(n ∈ A){
            if(n ∈ R(m))DrawEdge(m → n);
            elseif(m ∈ R(n))DrawEdge(n → m);
        }
    }
    foreach(m ∈ M){
        FlipAllStatesDownChain(m);
    }
    η = ⋂_{k∈AliveNodes} R^c_{i→k} ∩ D_R;
    if(F ⊃ η)break;
}
N(i) = AliveNodes;
ε_i = η;
```

Fig. 4. Search algorithm for Phase 1.

may be a whole chain of nodes (i.e., a path on the relay graph) where one node blocks the next one down the chain. When a new node found blocks the first alive node in this chain, the states of all the nodes down the chain need to be flipped. The function *FlipAllStatesDownChain* handles this situation.

We now begin the formal description of the algorithm. Below, $D_{\mathcal{R}}$ denotes the fixed deployment region, A denotes all the nodes that i has found thus far in its search, M denotes the new nodes found in the current iteration, S denotes the current search region, and F denotes all the area that has been searched so far.

We associate a state variable with every node in A. The state of a node in A is allowed to be only one of three possibilities: alive, dead, or it may be yet "unmarked" by the algorithm. The sets *AliveNodes* and *DeadNodes* denote the set of alive nodes in A and the set of dead nodes in A, respectively. The function *MarkAlive*(m) marks the state of node m alive. The function *MarkDead*(m) marks the state of node m dead.

The function *DrawEdge*$(m \rightarrow n)$ forms $e_{m \rightarrow n}$ on the relay graph of i. The algorithm for computing $(\varepsilon_i, N(i))$ appears in Fig. 4. The auxiliary function *FlipAllStatesDownChain* appears in Fig. 5.

In this algorithm, the function *SetSearchRegion* sets the search region in each iteration depending on the nodes that have been found thus far and the remaining area to be searched. The termination of the algorithm depends on the choice of the search regions. It is always possible to terminate the "while" loop by setting $S = D_{\mathcal{R}}$. For mobile networks, the challenge is to find the function *SetSearchRegion* such that the energy consumption until the algorithm terminates is minimized.

```
void FlipAllStatesDownChain(m){
    if(m ∈ AliveNodes){
        MarkDead(m);
        foreach(k ∈ R(m)) FlipAllStatesDownChain(k);
    }
    elseif(m ∉ R(k)  ∀k ∈ AliveNodes){
        MarkAlive(m);
        foreach(k ∈ R(m)) FlipAllStatesDownChain(k);
    }
}
```

Fig. 5. Auxiliary function *FlipAllStatesDownChain*.

In Appendix C, we discuss some subtle features of the search algorithm. The next two theorems assert the correctness of the search algorithm and the uniqueness of the solution that this algorithm finds.

Theorem 2—Correctness of Search Algorithm: When the search for enclosure algorithm terminates, it terminates with $(\varepsilon_i, N(i))$ as the solution to the two equations in Definition 3.

Theorem 3—Uniqueness of Enclosure and Neighbor Set: The solution $(\varepsilon_i, N(i))$ found by the search algorithm is unique.

B. Phase 2—Cost Distribution

In Phase 1 of the algorithm, we took a geometric problem described only by the positions of the nodes on a two-dimensional plane and specified how to construct a sparse graph (called the enclosure graph) of communication links between these nodes. Therefore, Phase 1 constitutes a link setup and configuration phase. The key point is that the globally optimal links (for the minimum power consumption for communication to the master-site) are all contained in the enclosure graph.

Phase 2 of the algorithm finds the optimal links on the enclosure graph. Therefore, after the enclosure graph has been found in Phase 1, we apply the distributed Bellman–Ford shortest path algorithm [8] on the enclosure graph using power consumption as the cost metric. In Phase 2, each node broadcasts its cost to its neighbors. The cost of a node i is defined as the minimum power necessary for i to establish a path to the master-site.

Each node calculates the minimum cost it can attain given the costs of its neighbors. Let $n \in N(i)$. When i receives the information Cost(n), it computes

$$C_{i,n} = \text{Cost}(n) + P_{\text{transmit}}(i, n) + P_{\text{receiver}}(n)$$

where $P_{\text{transmit}}(i, n)$ is the power required to transmit from i to n, and $P_{\text{receiver}}(n)$ is the additional receiver power that i's connection to n would induce at n. $P_{\text{receiver}}(n)$ is either known to i, if for instance every user carries an identical receiver, or can be transmitted to i as a separate piece of information along with Cost(n). Then, i computes

$$\text{Cost}(i) = \min_{n \in N(i)} C_{i,n}$$

and picks the link corresponding to the minimum cost neighbor. This computation is repeated, and the minimum cost

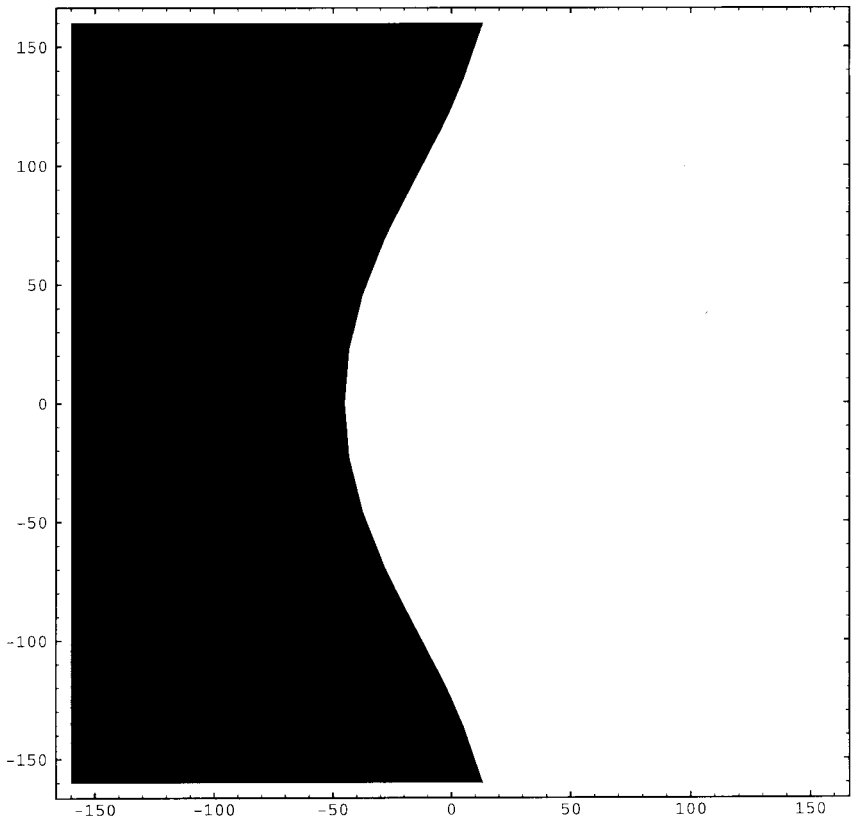


Fig. 6. Relay region for 80 m internodal distance.

neighbor is updated each time. The convergence of the algorithm to a set of links after a finite number of iterations is guaranteed, as discussed in [8]. The data transmission from i to the master-site can then start on the minimum cost neighbor link, which is the global minimum power link, as the next theorem shows.

Theorem 4—Minimum Power: The distributed protocol described above finds the minimum power topology on $\aleph$.

VI. Computation of the Relay Region

In the following example, we illustrate the relay region of a single node, assuming the two-ray propagation model for terrestrial communications, which implies a $1/d^4$ transmit power rolloff [14]. The close-in reference distance is taken as 1 m. The carrier frequency is 1 GHz, and the transmission bandwidth 10 kHz. We assume omnidirectional antennas with 0 dB gain, -160 dBm/Hz thermal noise, 10 dB noise figure in the receiver, and a predetection signal-to-noise ratio (SNR) of 10 dB. Using the Friis free-space formula gives -67.5 dBm as the minimum transmit power required for detection at 1 m. We take this to be roughly -70 dBm for our simulations. This can be treated as an effective predetection threshold to be used with the $1/d^4$ rolloff formula to compute the minimum required transmit power for any distance.

We assume the following model for receiver power at any relay node: a fixed receiver power of 80 mW is consumed at each node, with 20 mW increase for each additional node from which transmission is received. This model can be easily modified according to actual receiver design [4], [15].

With the previous assumptions, the relay region is obtained by solving the following two equations simultaneously:

$$d_{ij}^4 \geq d_{ir}^4 + d_{rj}^4 + c/t$$

and

$$d_{ij}^2 = d_{ir} + d_{rj}^2 - 2d_{ir}\, d_{rj} \cos\theta$$

and θ where is the angle between position vectors $\boldsymbol{r}_{r\to i}$ and $\boldsymbol{r}_{r\to j}$. These equations are obtained by the same method as in the proof of Lemma 1. Above, c denotes the additional receiver power cost of 20 mW for relays and t the predetection threshold of 10^{-7} mW.

Fig. 6 displays the relay region in the case where the relay node is at (0,0), and the transmit node is at (80,0). The relay region has been shaded. The units are meters.

VII. Stationary Network Simulation

We now simulate a stationary network with nodes deployed over a square region of 1 km on each side. The (x, y) coordinates of the nodes are generated as independent, identically distributed (i.i.d.) uniform random variables over this region. Since the nodes are stationary, once each node is enclosed and obtains a valid cost, the network remains in the minimum power topology.

The transmit and receiver powers for providing point-to-point connections are as described in Section VI. In this simulation, we investigate how the total power consumption of the minimum power topology varies with the number of

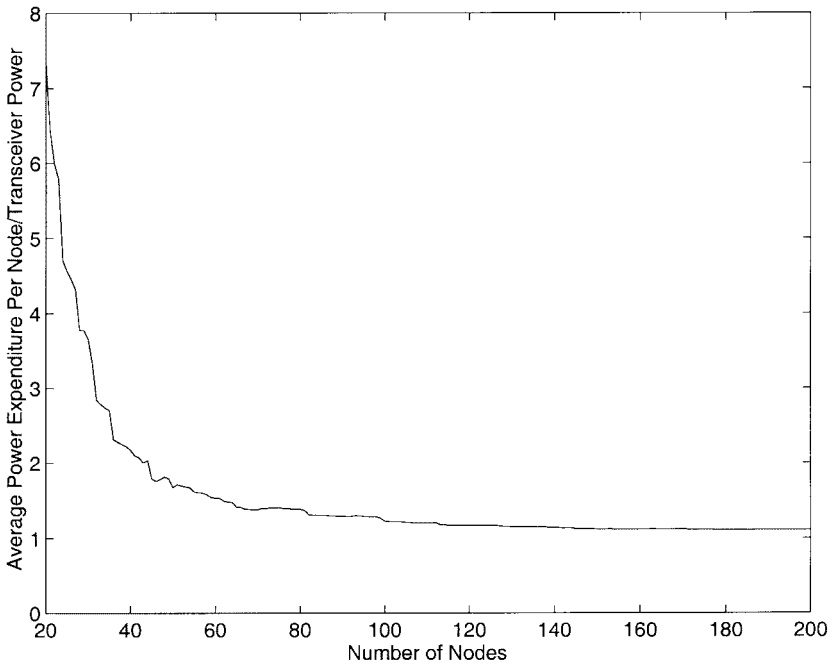


Fig. 7. Average power expenditure per node.

nodes. Fig. 7 illustrates this relationship. As the number of nodes grows larger, the average power decreases toward its asymptote of 100 mW receiver power/node. The plot has been normalized to the receiver power.

VIII. Distributed Mobile Networks

The protocol developed so far has been for stationary networks. However, due to the localized nature of its search algorithm, it proves to be an effective energy-conserving protocol for the mobile case as well.

Synchronization in a mobile network can be achieved by use of the absolute time information provided by GPS up to 100 ns resolution [12]. In a synchronous network, each node wakes up regularly to "listen" for change and goes back to the sleep mode to conserve power. We call the time between successive wakeups the cycle period of the network. If the cycle period is too long, the power costs to the master-site can change significantly from one wakeup to the next. In this case, the network cannot track the correct costs. If the cycle period is too short, then the network consumes unnecessary energy to compute costs that change only slowly. The choice of the cycle period for energy-efficient operation of a wireless network must address this tradeoff. In our simulation, we assume that the cycle period has been chosen to meet these two constraints.

After wakeup, each node executes Phase 1 of the protocol, as described in Section V. When a node completes Phase 2, it either starts data transmission on the optimal link, or it goes to the sleep mode to conserve power.

The protocol is self-reconfiguring since strong connectivity is ensured within each cycle period, and the minimum power links are dynamically updated. It can be seen that this protocol is also fault tolerant. A network protocol is "fault tolerant" if it is self-reconfiguring when nodes leave or new nodes join the network. Under such a scenario, each node employing our protocol would compute its new enclosure and find the minimum power topology.

IX. Mobile Network Simulation

In this section, we simulate a mobile set of nodes and measure the energy consumption. The initial positions of 100 nodes are generated as i.i.d. uniform random variables over a square field, 1 km on each side. The velocity in each coordinate direction is uniformly distributed on the interval $(-v_{\max}, v_{\max})$. The velocity is the vector sum of the velocities in each coordinate direction. We vary $v_{\max}$ to observe how the energy consumption changes.

The choice of the *SetSearchRegion* function in the search algorithm, which is optimized to perform the minimum energy neighbor search, is a topic of our current research. Therefore, in this simulation, we assume omnidirectional antennas and use a heuristic strategy for the choice of the search radius. The results indicate that even with a heuristic, the energy consumption is very low.

Let T be the cycle period of the network. Assume that node i is enclosed in the nth iteration, and let e_n be the distance of i to its furthest neighbor in the nth iteration. In the next iteration, if i sets its search radius to

$$r_{n+1} = e_n + 2\sqrt{2}\, v_{\max} T$$

then its neighbors in the nth iteration must fall within this radius. Because the cycle period is small enough to allow positions to vary only slightly from one iteration to the next, in most cases the node will have its previous neighbors in its new enclosure as well. Nodes employing this strategy are enclosed within one iteration of the search algorithm presented in Section V.

From a system perspective, the measure of mobility is not the velocities, but rather the displacements of nodes in a cycle

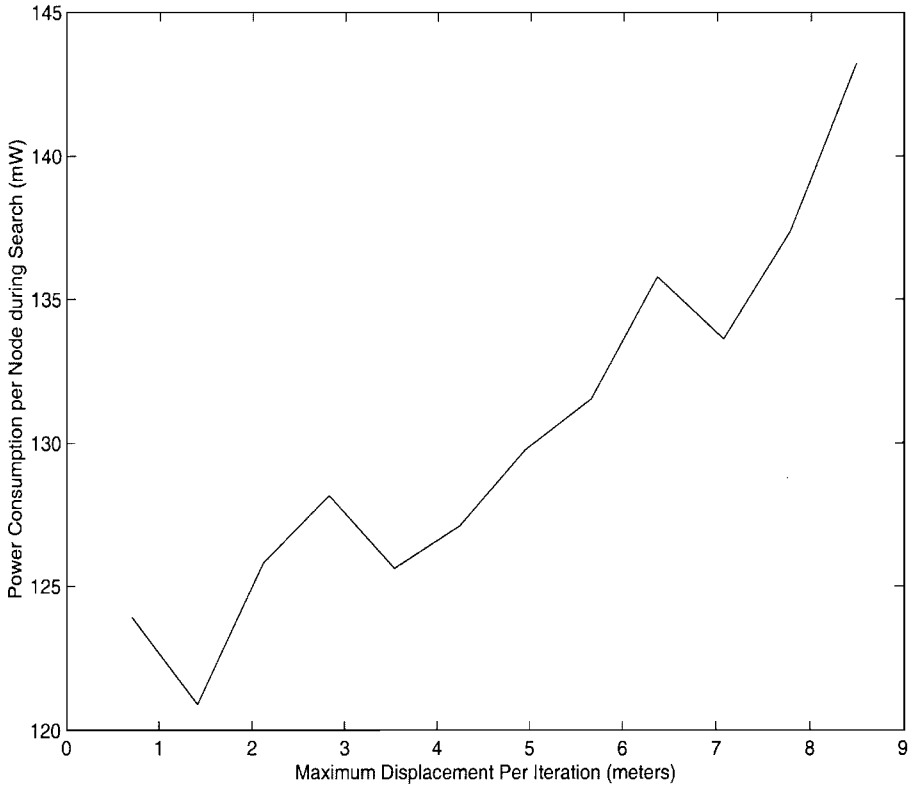


Fig. 8. Power consumption per node during search period.

period of the network. The maximum displacement of a node in a cycle period is $\sqrt{2}\,v_{\max}T$ from the previous analysis. Fig. 8 displays the search period power level per node averaged over 10 000 iterations and averaged over all the nodes. The horizontal axis on this graph is the maximum displacement in meters. Since the average distance between nodes is about 100 m in this particular simulation, we estimated that the network cannot track correct costs for maximum displacements greater than 8 m, and we graphed power consumption over only this range.

Fig. 9 displays the search period power consumption per meter of maximum displacement. The graph indicates that the power consumption per node scales better than linearly with maximum displacement for the range of displacements for which the network can track the correct costs.

The energy expenditure during the search depends on the search duration. For the particular network in this simulation, a two-way propagation delay between a node and its neighbors is estimated to be on the order of 1 μs. The time that it takes for the transceiver circuits to ramp up and transmit at full power is estimated to be on the order of 1 ms, which is much larger, and hence is the determining factor for the length of the search period. The energy expenditure per node during a search can then be found by multiplying the search-period power consumption by this delay.

The energy consumption of a mobile network that uses this protocol is very low. As an example, for $v_{\max} = 10$ m/s and for a cycle period of $T = 210$ ms, the maximum displacement is about 3 m. Then, the power consumption during the search period of a node is about 127 mW from Fig. 8. If the node goes to the sleep mode after the search, the search period is simply the "on" period of 1 ms/cycle, which is the time required for transceiver circuits to operate. Then, the average power that the protocol consumes over a cycle period is only 0.6 mW/node.

X. Conclusion

We have described a distributed protocol to find the minimum power topology for a stationary ad hoc network. Because the topology is found via a local search in each node's surrounding, we argued that this is applicable to a mobile ad hoc network. We simulated the performance of the protocol for a mobile network and found that the average power consumption per node is significantly low.

Appendix A

In this appendix, we show that if the lognormal shadowing model is included in addition to path loss, the shape of the relay regions does not change. In fact, an effective detection threshold can be defined as a function of the tolerable outage probability and the variance of the lognormal distribution. Then, this effective detection threshold can be used in the place of the detection threshold in the rest of the analysis in this paper.

Let α denote the target probability that the received power level after MRC stays above the power threshold for detection (denoted by P_{thr}). Let d denote the distance between the transmit and receive antennas. Let σ denote the standard deviation of the Gaussian random variable underlying the

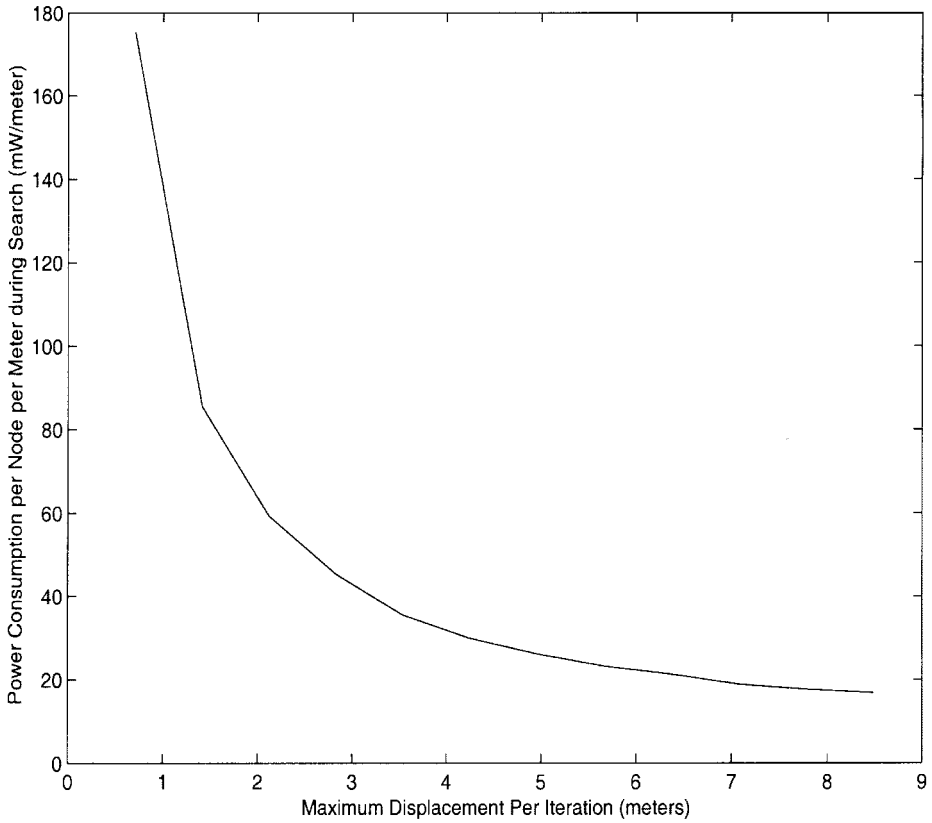


Fig. 9. Power consumption per node per meter of maximum displacement during a search.

lognormal distribution. Let $Q(.)$ denote the Q function. Then, we would like to have

$$Q\left(\frac{P_{\text{thr}} - P + 10\log_{10}(d)}{\sigma}\right) = \alpha$$

where P and P_{thr} are measured in dBW. Writing this equality with the transmit power P on the left-hand side gives

$$P = \left(10^{\sigma \times \arg\, Q(-\alpha)/10} P_{\text{thr}}\right) d^n$$

where P and P_{thr} are measured in W. Clearly, by defining the effective predetection threshold to be the coefficient of d^n in the second equation, we arrive at an expression for transmit power identical in form to the one obtained using only path loss.

The conclusion of this discussion is that even when the lognormal shadowing effects are considered, the asymptotic properties of the relay region stay the same. Compared to the relay region obtained using only path loss, the boundary for the relay region adjusted for lognormal shadowing is shifted outwards; hence, the enclosure for each node would be slightly larger, depending on the measured σ for the environment and the target probability α.

APPENDIX B

In this Appendix, we give the proofs of the lemmas and theorems proved in the paper.

Lemma 1—Asymptotic Behavior of Relay Regions:

Proof: At the boundary $B(i, r)$, we have $P_{i\to r\to(x_B, y_B)} = P_{i\to(x_B, y_B)}$. Let c denote the receiver power consumption at each of the three nodes. The left-hand side of this equation is comprised of the transmit power to reach from i to r, the transmit power to reach from r to the boundary, and the additional receiver power for the relay node. The righthand side contains only the transmit power to reach from the transmit node to the boundary. Hence, at the boundary

$$td_{ir}^n + td_{r,(x_B, y_B)}^n + c = td_{i,(x_B, y_B)}^n.$$

In addition to this relationship, by the law of cosines, we have

$$d_{ir}^2 + d_{r,(x_B, y_B)}^2 - 2d_{ir}\, d_{r,(x_B y_B)} \cos\theta = d_{i,(x_B, y_B)}^2$$

where θ is the angle between the position vectors $\boldsymbol{r}_{r\to i}$ and $\boldsymbol{r}_{r\to(x_B, y_B)}$). Solving for x_B as $y_B \to \pm\infty$, we obtain $x = d_{ir}/2$. The proof for $n = 2$ is similar.

Lemma 2—Distance Properties of Relay Nodes:

Proof:

1) We will use the coordinate system of Lemma 1. For the case with $n > 2$, the asymptote of $R_{i\to r}$ is the set of equidistant points from i and r. Since j lies to the left of the asymptote, $d_{jr} < d_{ji}$. For the case with $n = 2$, j lies on the $-x$ axis, and hence $d_{jr} < d_{ji}$.
2) If $j \in R(r)$, then $P_{i\to r\to j} < P_{i\to j}$. Writing this in terms of transmit and receiver power terms gives $c + td_{ir}^n + td_{rj}^n < td_{ij}^n$. Then, $d_{ij}^n > d_{ir}^n + d_{rj}^n + c/t > d_{ir}^n$ where the last inequality follows from the nonnegativity of distance and power. Since $n \geq 2$, this establishes the result.

Lemma 3—Properties of Relay Regions:

Proof:

1) Since $P_{i\to k\to k} = P_{i\to k}$, the inequality in the definition of the relay region, with k taken as both the relay and receive node, is not satisfied. The result follows immediately from this fact.
2) If $k \in R(j)$, then $P_{i\to j\to k} < P_{i\to k}$. Expressing this in terms of the transmit and receiver power terms gives $c + td_{ij}^n + td_{jk}^n < td_{ik}^n$. Then, $d_{ij}^n < d_{ik}^n - (d_{jk}^n + c/t)$. By the nonnegativity of distance and power, this implies that $d_{ij}^n < d_{ik}^n + (d_{jk}^n + c/t)$. Then, $P_{i\to k\to j} > P_{i\to j}$, which shows $j \notin R(k)$.
3) If $k \in R(j)$, then $d_{ij}^n < d_{ik}^n - (d_{jk}^n + c/t)$, which implies $d_{ij}^n < d_{ik}^n + (d_{lj}^n + c/t)$ by the nonnegativity of distance and power. This implies that $d_{ij}^n < d_{il}^n + (d_{lj}^n + c/t)$ by Lemma 2.2 and the fact that $l \in R(k)$. This shows that $P_{i\to l\to j} > P_{i\to j}$. In other words, $j \notin R(l)$.

Theorem 1—Strong Connectivity:

Proof: We prove this theorem by setting up an iterative algorithm that terminates with the desired result. Let (i, j) be any pair of distinct nodes in the node set. Our aim is to show that there always exists a directed path from i to j. In the algorithm that follows, p denotes the current node, and the variable *Path* denotes the ordered collection of nodes on the path formed thus far in the algorithm. The algorithm is as follows:

$$
\begin{aligned}
&p = i;\\
&Path = Concat\{\varnothing, i\};\\
&while(j \notin N(p))\ \{\\
&\qquad p = Neighbor_j(p);\\
&\qquad Path = Concat\{Path, p\};\\
&\}\\
&Path = Concat\{Path, j\};
\end{aligned}
$$

In this algorithm, $\varnothing$ denotes the null set. The function *Concat* appends the node in its second argument to the path in its first argument. The function $Neighbor_j(p)$ returns a node $k \in N(p)$ such that $(x_j, y_j) \in R_{p\to k}$. Such a node k always exists for the following reason: $j \notin N(p)$ and $(x_j, y_j) \in D_{\aleph}$ imply $(x_j, y_j) \in R_{p\to k}$ for some $k \in N(p)$. Now, by the strict inequality in Lemma 2.1, no node can appear in the *Path* more than once. Because the number of nodes $|\aleph|$ is finite, the loop terminates after at most $|\aleph| - 1$ iterations with a path between i and j.

Lemma 4—No Cycles on the Relay Graph:

Proof: Assume that there is a cycle of length n on $G(i)$. Then, the n distinct nodes in the cycle can be labeled as $1 \to 2 \to 3 \to \cdots \to n \to 1$. This implies that $P_{1\to 2\to 3\to \ldots \to n} < P_{i\to n}$ and $P_{i\to n\to 1} < P_{i\to 1}$. But $P_{i\to 1} < P_{i\to 1\to 2\to 3\to \ldots \to n}$ since the nodes are distinct, and the power consumption is always nonnegative. Therefore, $P_{i\to n\to 1} < P_{i\to n}$, which is a contradiction.

Theorem 2—Correctness of Search Algorithm:

Proof: The expression for ε_i in the first part of Definition 3 is satisfied due to the definition of the variable η in the algorithm. We must show that the second part of Definition 3 holds, i.e., that $N(i) = \{n \in \aleph | (x_n, y_n) \in \varepsilon_i,\ n \neq i\}$. Equivalently, we must show that $n \in N(i)$ if and only if $(x_n, y_n) \in \varepsilon_i,\ n \neq i$. Let $n \in N(i)$. Then, in the last iteration before the algorithm terminates, $n \in$ *AliveNodes*. Then, there can be no directed edges from any node in *AliveNodes* to n by the condition of the second *foreach* loop in the *FlipAllStatesDownChain* function. This implies that $(x_n, y_n) \notin R_{i\to k}$ for any $k \in$ *AliveNodes*. To prove the result in the other direction, let n be a node such that $(x_n, y_n) \in \varepsilon_i,\ n \neq i$. The fact that the "while" loop terminates implies that $F \supset \varepsilon_i$, where F is the total search area in the final iteration of the "while" loop. Hence $(x_n, y_n) \in F$, which implies $n \in A$ where A is the set of all nodes in the final iteration. Now, since $(x_n, y_n) \notin R_{i\to k}$ for any $k \in$ *AliveNodes* by assumption and the algorithm marks a node in A as dead only if there is a directed edge to it from an alive node, this shows that $n \in$ *AliveNodes* in the last iteration. Therefore, $n \in N(i)$.

We will use the following lemma in the proof of Theorem 3.

Lemma 5—Pairing of Enclosure and Neighbor Set: Let $(\varepsilon^{(1)}, N^{(1)})$ and $(\varepsilon^{(2)}, N^{(2)})$ be two solutions to the set of equations in Definition 3. Then $\varepsilon^{(1)} = \varepsilon^{(2)}$ if and only if $N^{(1)} = N^{(2)}$.

Proof: Let $\varepsilon^{(1)} = \varepsilon^{(2)}$. Then, $N^{(1)} \backslash N^{(2)} = \{n | (x_n, y_n) \in \varepsilon^{(1)}, (x_n, y_n) \notin \varepsilon^{(2)}, n \neq i\}$. Reversing the roles of $N^{(1)}$ and $N^{(2)}$ in the last argument shows that $N^{(1)} = N^{(2)}$. The proof in the other direction follows trivially.

Theorem 3—Uniqueness of Enclosure and Neighbor Set:

Proof: Let $(\varepsilon^{(1)}, N^{(1)})$ and $(\varepsilon^{(2)}, N^{(2)})$ be two solutions to the set of equations in Definition 3. By way of contradiction, assume that $N^{(1)} \neq N^{(2)}$. Let $n \in N^{(1)}$ and $n \notin N^{(2)}$ without loss of generality. Then, there exists a node $k \in N^{(2)}$ such that $n \in R(k)$. We construct a path starting from n and append the node k to this path. Now, $k \notin N^{(1)}$ since otherwise we could not have $n \in N^{(1)}$. We repeat this argument for k, find a node p in $N^{(1)}$ such that $k \in R(p)$, and append p to the path. By Lemma 4, the path constructed this way cannot have any cycles. Since $|\aleph|$ is finite, at some finite iteration, there are no nodes left outside the path to satisfy the condition of the argument. This contradiction and Lemma 5 establish the uniqueness of $(\varepsilon_i, N(i))$.

Theorem 4—Minimum Power:

Proof: We divide the proof into two parts called 1) and 2). These two parts correspond to the two defining properties of the minimum power topology. 1) Every node has a directed path to the master-site by Theorem 1 at the end of Phase 1 of the protocol. Phase 2 of the protocol eliminates the link $l_{i\to j}$ only if node i has a valid cost, i.e., only if i has a path to the master-site. Hence, every node has a path to the master-site at the end of Phase 2. 2) Form a fully connected graph on $\aleph$ by connecting every node to every other node directly. The distributed Bellman–Ford algorithm finds the optimal links (using power as the cost metric) for this graph as shown in [8]. We need to prove that these optimal links are necessarily contained in the enclosure graph of $\aleph$. Let $l_{i\to j}$ be an optimal link. If $l_{i\to j}$ is not in the enclosure graph of $\aleph$, then there exists a relay node r such that $P_{i\to r\to j} < P_{i\to j}$. But this contradicts that $l_{i\to j}$ is an optimal link, which proves the result.

APPENDIX C

A few remarks are in place to describe some subtle features of the search algorithm. First, the recursive function *FlipAll-*

StatesDownChain terminates at most at depth $|A|-1$ because the relay graph has no cycles by Lemma 4, and $|A|$ is finite since $|A| \leq |\aleph| - 1 < \infty$. Second, examine the statements in the algorithm and in the auxiliary function that are instances of the generic statement

$$\textit{foreach}(k \in H)\textit{FlipAllStatesDownChain}(k);$$

where H is any set with $H \subset A$. We must show that whenever this statement is executed, the order in which k is chosen out of H has no effect on the final values of the variables when the *foreach* loop terminates. We prove this result as follows: Let O_1 be an ordering of the elements of H, and let O_2 be another ordering of the elements of H, which is distinct from O_1. By way of contradiction, assume that there exists a node called m with the following two properties.

$P1$ Under O_1, when the *foreach* loop terminates, it leaves m marked "dead."

$P2$ Under O_2, when the *foreach* loop terminates, it leaves m marked "alive."

By $P1$, there exists a node, call it n, such that n is alive under O_1, and n has an edge to m on the relay graph. By $P2$, all nodes that have directed edges to m must be dead under O_2. In particular, n must be dead under O_2. Then n satisfies the following properties.

$P1_n$ Under O_1, when the *foreach* loop terminates, it leaves n marked "alive."

$P2_n$ Under O_2, when the *foreach* loop terminates, it leaves n marked "dead."

We replace m by n, repeat the argument, and construct a path on the relay graph to which we append the new node each time the argument is repeated. Each iteration leaves the last node marked dead under one of the orderings. However, since there are no cycles on the relay graph, no node can be repeated in this path. After $|\aleph| - 1$ iterations, the last node that was added to the path is marked dead under one of the orderings, but there can be no alive node that has an edge to it on the relay graph since all the other nodes have already been added to the path. This contradiction establishes the result.

REFERENCES

[1] J. B. Andersen, T. S. Rappaport, and S. Yoshida, "Propagation measurements and models for wireless communication channels," *IEEE Commun. Mag.*, vol. 33, pp. 42–49, Jan. 1995.

[2] D. C. Cox, "910 MHz urban mobile radio propagation: Multipath characteristics in New York City," *IEEE Trans. Commun.*, vol. COM-21, pp. 1188–1194, Nov. 1973.

[3] P. Davis, P. Smith, E. Campbell, J. Lin, K. Gross, G. Bath, Y. Low, M. Lau, Y. Degani, J. Gregus, R. Frye, and K. Tai, "Silicon-on-silicon integration of a GSM transceiver with VCO resonator," in *Proc. IEEE Int. Solid-State Circuits Conf.*, Feb. 1998, pp. 248–249.

[4] P. Gray and R. Meyer, "Future directions in silicon IC's for RF personal communications," in *Proc. CICC'95*, May 1995, pp. 83–90.

[5] W. C. Jakes, *Microwave Mobile Communications.* Piscataway, NJ: IEEE Press, 1994.

[6] W. Lee, P. Landman, B. Barton, S. Abiko, H. Takahashi, H. Mizuno, S. Muramatsu, K. Tashiro, M. Fusumada, L. Pham, F. Boutaud, E. Ego, G. Gallo, H. Tran, C. Lemonds, A. Shih, M. Nandakumar, B. Eklund, and I.-C. Chen, "A 1V DSP for wireless communications," in *Proc. IEEE Int. Solid-State Circuits Conf.*, Feb. 1997, pp. 92–93.

[7] B. M. Leiner, D. Nielson, and F. A. Tobagi, "Issues in packet radio network design," *Proc. IEEE*, vol. 75, pp. 6–20, Jan. 1987.

[8] N. A. Lynch, *Distributed Algorithms.* San Mateo, CA: Morgan Kaufmann, 1996, pp. 51–80.

[9] S. Mattisson, "Minimizing power dissipation of cellular phones," in *Proc. 1997 Int. Symp. Low Power Electronics and Design*, Aug. 1998, pp. 42–45.

[10] T. Meng, B. Gordon, E. Tsern, and A. Hung, "Portable video-on-demand in wireless communication," *Proc. IEEE*, vol. 83, pp. 659–680, Apr. 1995.

[11] J. C. Navas and T. Imielinski, "GeoCast—Geographic addressing and routing," in *Proc. Third ACM/IEEE Int. Conf. Mobile Computing and Networking 1997 (MobiCom'97)*, pp. 66–76.

[12] B. W. Parkingson and J. J. Spilker, *Global Positioning System: Theory and Applications Vol. I.* Washington, DC: American Institute of Aeronautics and Astronautics, Inc., 1996.

[13] Proakis, *Digital Communications*, 3rd ed. New York: McGraw-Hill, 1995, pp. 758–806.

[14] T. S. Rappaport, *Wireless Communications: Principles and Practice.* Englewood Cliffs, NJ: Prentice-Hall, 1996, pp. 69–122, 139–196.

[15] J. C. Rudell *et al.*, "A 1.9-GHz wide-band IF double conversion CMOS receiver for cordless telephone applications," *IEEE J. Solid-State Circuits*, vol. 32, pp. 2071–2088, Dec. 1997.

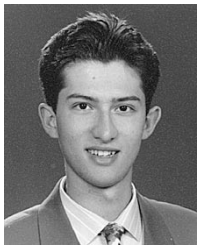
[16] K. Scott and N. Bambos, "Routing and channel assignment for low power transmission in PCS," in *Proc. ICUPC, Fifth Int. Conf. Universal Personal Communications*, Oct. 1996, vol. 2, pp. 498–502.

[17] A. R. Shahani, D. K. Schaeffer, and T. H. Lee, "A 12 mW wide dynamic range CMOS front-end for a portable GPS receiver," in *Proc. IEEE Int. Solid-State Circuits Conf.*, vol. 40, Feb. 1997, pp. 368–369.

[18] R. Stephany, K. Anne, J. Bell, G. Cheney, J. Eno, G. Hoeppner, G. Joe, R. Kaye, J. Lear, T. Litch, J. Meyer, J. Montanaro, K. Patton, T. Pham, R. Reis, M. Silla, J. Slaton, K. Snyder, R. Witek, "A 200MHz 32b 0.5W CMOS RISC microprocessor," in *Proc. 1998 IEEE Int. Solid-State Circuits Conf.*, Feb. 1998, pp. 238–239.

[19] F. A. Tobagi, "Modeling and performance analysis of multihop packet radio networks," *Proc. IEEE*, vol. 75, pp. 135–155, Jan. 1987.

[20] A. Wagemans, P. Ballus, R. Dekker, A. Hoogstraate, H. Maas, A. Tombeur, J. Van Sinedren, "A 3.5 mW 2.5GHz diversity receiver and a 1.2 mW 3.6GHz VCO in silicon-on-anything," in *Proc. 1998 IEEE Int. Solid-State Circuits Conf.*, Feb. 1998, pp. 250–251.



Volkan Rodoplu (S'94) received the B.S.E.E. degree (with highest honors) from Princeton University, Princeton, NJ, in 1996 and the M.S.E.E. degree from Stanford University, Stanford, CA, in 1998. He is currently a Ph.D. candidate at Stanford University and a Stanford graduate fellow.

His current research projects include power-efficient distributed network design, multiuser detection architectures, and transmitter optimization in wireless multiuser systems.

Teresa H. Meng (S'82–M'83–SM'93–F'99) received the B.S. degree from National Taiwan University in 1983 and the M.S. and Ph.D. degrees from the University of California, Berkeley, in 1984 and 1988, respectively.

She joined the faculty of the Electrical Engineering Department at Stanford University, Stanford, CA, in 1988, where she is an Associate Professor. Her current research activities include low-power design, portable DSP systems, and wireless communications.

Dr. Meng received the IEEE Signal Processing Society's Paper Award in 1989, the 1989 NSF Presidential Young Investigator Award, the 1989 ONR Young Investigator Award, a 1989 IBM Faculty Development Award, and the 1988 Eli Jury Award from the University of California at Berkeley for recognition of excellence in systems research. She also served as General Chair of the 1996 IEEE Workshop on VLSI Signal Processing. She was Program Cochair of the 1992 Application Specific Array Processor Conference and the 1993 HOTCHIP Symposium.