Energy Efficient Adaptation of Multicast Protocols in Power Controlled Wireless Ad Hoc Networks*

Caimu Tang
Computer Science Department
University of Southern California
Los Angeles, CA 90089

Cauligi S. Raghavendra and Viktor Prasanna
Department of EE-Systems
University of Southern California
Los Angeles, CA 90089

Abstract

In this paper a methodology for adapting existing wireless ad hoc network protocols to power controlled networks is presented. Wireless nodes are assumed to have transmission power control with m discrete levels, and a clustering scheme is used in the adaptation to be energy efficient. Clusters are formed among the nodes in a distributed self-organized manner so that each node is a member of some cluster and within a cluster, each node can reach any other node with appropriate power level. Each cluster has a designated head node which acts as forwarding agent for its members and these head nodes form a supernode topology. We adapt an ad hoc network multicast protocol by executing it on this supernode topology. Multicast data will move from the sender to its cluster head, then along the supernode topology according to the results of the chosen multicast protocol, and finally from cluster head to receivers within their clusters. At every step, nodes use appropriate power level, $1 \leq j \leq m$, to reach all the intended recipients. For more balanced way of depleting energy in the network, nodes take turn to become cluster heads. We applied our technique to ODMRP and the experiement shows siginifcant energy reduction.

1. Introduction

Recently, attention has been given to designing protocols that are energy efficient. In wireless communications between two nodes, there is a fixed energy cost in transceiver electronics and power amplifier and a variable energy cost that depends on the distance of transmission. Therefore, nodes having power control can transmit with just the right amount of power to reach the intended recipient [1]. In the 70s, work was done on packet radio networks with such power-controlled networks, and an interesting problem solved then was the optimization for throughput when all nodes adjust and use the same level of power. Later, similar optimization problems were solved when different nodes could use different power. More recently, there have been some works which assumed continuous power control to perform communication tasks, including initialization [6], topology design and routing [8]. These latter works focused on energy efficiency in ad hoc networks.

In this paper, we assume a more practical model for power control in nodes, where a node can transmit with m discrete power levels. There exist several radios including Sincgars [11], Rockwell Wins [10], that exhibit this characteristic. Radios with several power levels are more practical to build and easier to operate than radios with continuous adjustable power. We assume that the highest level of power in the power controlled nodes (i.e., m-th level) corresponds to the amount of power used by fixed range nodes. Since a node with power control can communicate with some other nodes by adjusting its power, there is no well-defined topology for the network unless nodes fix the amount of power they will use. In such networks, one may redesign power efficient protocols by taking into account this capability.

The idea is to take advantage of the existing protocols for ad hoc networks with fixed range nodes and to adapt them using a clustering technique so as to be energy efficient. In multicast and broadcast operations, it is natural to use clusters so that nearby nodes can be reached with a single transmission. Our greedy algorithms operates in a self-organized manner by successively increasing the power level until the successive incremental grows with an increase in power level. Here, each node in a cluster can transmit with a power level that is just enough to reach all other nodes within its cluster. Each cluster will have a designated cluster head which is chosen in a distributed manner as explained later. The power level needed in different clusters to reach all nodes may be different. Each cluster head can be thought of as a supernode, and these cluster heads form a supernode topology on which we run any known ad hoc net-

---

*This research is supported by the DARPA contract F33615-C-00-1633 in the Power Aware Computing and Communications Program

work protocol. With local broadcasting within clusters and communication in supernode topology according to the selected protocols, we accomplish energy efficient versions of those protocols. In this paper, for brevity, we assume that the nodes are not mobile. The clustering naturally allows easy updates when nodes are non-stationary. We have developed a scheme to handle the node mobility to update the clusters and supernode topology to remain energy efficient, however, a detailed discussion is beyond the scope of this paper.

This adaptation of existing ad hoc network protocols using clustering is general and quite powerful and can be used for developing energy efficient routing, multicasting, and broadcasting in dense wireless networks. We demonstrate this adaptation using ODMRP (On-Demand Multicast Routing Protocol) [4]. In order to evaluate the effectiveness of our adaptation, we enhanced GloMoSim 2.0 package with energy calculations with multiple levels of power at nodes. We developed a front end software HMMA 1.0 that performs clustering and interfaces to GloMoSim. We conducted extensive simulations for different network sizes ranging from 75 nodes to 300 nodes. Results show that using our clustering and adaptation, there is 100% or more improvement in energy costs compared to multicasting without clusters.

This paper is organized as follows. In Section 2, we present the cluster formation algorithm for power-controlled networks where nodes can use m discrete power levels. In Section 3, we discuss in detail the adaptation methodology using ODMRP protocol and our clustering technique. In this section we also discuss how multicast data forwarding derived from ODMRP is accomplished in an energy efficient manner in both intra-cluster and inter-cluster transmissions. In Section 4, we present our simulation setup and experimental results for different size networks. Finally, we give some concluding remarks in Section 5.

2. Clustering Technique

Clustering is a well known technique for grouping nodes that are close to one another in a network [3], [5], [8]. In this paper, we restrict our discussion of clustering to support multicast routing. The members of a multicast group will be distributed among the nodes in the network, and our goal is to be energy efficient in sending data to all members. Clearly, reaching more multicast receivers in a single transmission is advantageous. This is possible only when there are multiple receivers that are close to one another. This suggests using clusters as all receivers in a cluster can possibly be reached by a single transmission from a designated node within each cluster. Our approach is to partition the network into a set of clusters with a cluster head in each cluster. The cluster heads can be thought of as supernodes and they form a supernode network. Packets flow from sender to its cluster head, then along the supernode topology, and finally get disseminated within clusters. We can see that having few clusters is advantageous as cluster heads can reach more nodes in a single hop within its cluster. However, it wastes energy as cluster heads must use enough power to reach the farthest node in their cluster. On the other head, having more clusters will increase the energy cost along the supernode network as we need to reach more cluster heads. Therefore, a balance between these two extreme cases needs to be achieved by forming clusters such that any cluster head uses a power level that is energy efficient. The approach we use to form clusters is based on a greedy algorithm to incrementally grow clusters and stop when the growth in a successive step drops or we reach the highest power level.

2.1. Energy Cost Analysis

We will further justify the use of greedy heuristic to form clusters from the energy cost analysis for packets traveling a network when there is power control. When a packet is sent between two nodes, there is a fixed energy cost in transceiver circuitry and a variable cost in the propagation of the signal to reach the receiver. The energy cost in receiving a packet is not negligible. For example, in WaveLAN [7] card with omni-directional antenna, it requires 185 mA for reception and 235mA for transmission at 3.3V. For a single hop the energy consumed is:

$$W = E_{Tx} + E_{Rx} + E'$$

where E_{Tx} and E_{Rx} are the energy consumption for transmission and reception, and E' is the overhead for MAC (Media Access Control) in both sender and receiver which may involve a few short packet exchanges as RTS/CTS (request to send, clear to send) in IEEE 802.11b, or CSMA-like scheme, involving channel sensing.

We use four parameters, α, γ, d, β to model the energy cost in a transmission as $\alpha * d^{\gamma} + \beta$. d is the radio apart distance between two nodes, β is the cost in the transceiver electronics, γ ranges from 2 to 4 [9], α is the radio energy cost coefficient depending on antenna gain, channel modulation etc.

In general, having bigger clusters is useful because a cluster head can send multicast data to all the receivers in its cluster in one transmission. However, we will now show when a cluster size should be limited. In order to analyze the energy consumption, we use the following notation: m_i: number of nodes between energy level i and $i + 1$. N_i: total number of receivers in the first i levels. R_i: area diameter covered in first i levels. r: incremental distance between two consecutive levels, it is a constant.

Consider the scenario shown in Figure 1. Assume the current cluster size requires power level $k - 1$ and it has

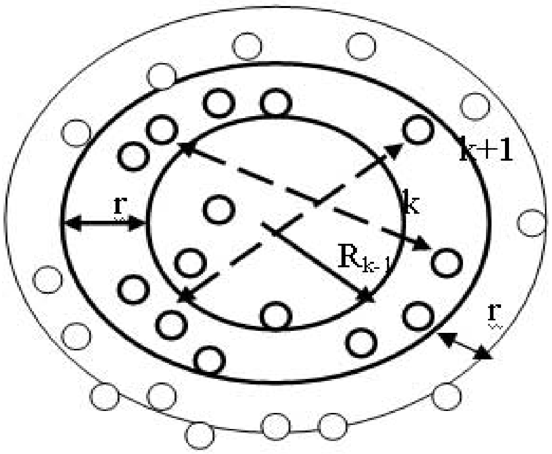

Figure 1. A multi-level cluster

a total of $\frac{N_{k+1}\sum_{i=1}^{i=k-1} m_i}{\sum_{i=1}^{i=k+1} m_i}$ multicast receivers in this cluster. When we increment the power level by 1 to k, we will be adding m_k nodes to the cluster. We can either (i) choose power level $k+1$ to reach all nodes in the increased area shown in Figure 1 or (ii) use power level k to reach nodes in the area up to level k followed by some forwarding in the incremental area. The following Eq. (1) and Eq. (2) show the energy cost for case (i) and (ii) respectively.

$$E_1 = \alpha(R_{k-1} + 2r)^\gamma + \beta + N_{k+1}\beta \qquad (1)$$

The first part is the transmission cost with level $k+1$. The second part is the total receiving cost for all multicast receivers in the bigger cluster.

$$E_2 = E' + \bar{E} \qquad (2)$$

where $E' = \alpha(R_{k-1} + r)^\gamma + \beta$ and

$$\bar{E} = \frac{N_{k+1} m_{k+1}}{\sum_{i=1}^{k+1} m_i}(\alpha r^\gamma + 3\beta) + N_{k+1}\beta$$

Here, the first part is the transmission cost with level k. The coefficient in the second part is the fraction of nodes that are receivers between levels k and $k+1$. Therefore, it is reasonable to expect that many additional cluster heads receive and retransmit data to cover the nodes in this strip. And the energy cost difference between the two scenarios is:

$$E_1 - E_2 = c_k(\alpha(R_{k-1} + r)^\gamma + 3\beta) - \Delta$$

where $\Delta = (R_{k-1} + 2r)^\gamma - (R_{k-1} + r)^\gamma$ and $c_k = N_{k+1}\frac{m_{k+1}}{3(m_k + m_{k+1})}$ is the k-level adjustment coefficient. Note that $(m_k + m_{k+1}) > \frac{1}{3}\sum_{i=1}^{k-1} m_i$, when the number of levels is much smaller than the total number of nodes. When m_{k+1} is greater them m_k, the coefficient is between $\frac{1}{3}$ and $\frac{1}{6}$ in the first term, and since N_{k+1} is usually much greater than the number of levels, it is advantageous to use scenario (i). When m_{k+1} is less than m_k, we lose the benefits of using larger cluster when γ is larger than 2 or the number of multicat receivers is not so many. This justifies using the our heuristic for clustering presented next.

2.2. Greedy Heuristic for Clustering

Given N nodes with some distribution in the field, we form clusters using a simple greedy heuristic algorithm starting from a single node. We assume that each node has some neighbors within its reach. Since the distribution of multicast receivers is unknown, it is not possible to formulate a clustering problem to optimize for energy cost. Instead, we should try to reduce energy cost to reach a group of receivers in a neighborhood. Thus, within a cluster, we would like any node to reach any other node with a single transmission. Forming very large clusters will waste energy as there may be few multicast receivers within a cluster. Forming large number of clusters requires many hops on the supernode network which increases energy cost end-to-end. With discrete power levels in nodes, we can strike a balance by growing the clusters as long as with each incremental level, additional nodes found in the neighborhood are higher than in the previous step. In other words, we grow the cluster as long as the delta change with each incremental level is positive. Every node will be part of some cluster, and it is perfectly possible that some nodes may be singleton cluster.

Since the number of power levels m is finite and a small number, this greedy heuristic can be very efficient. We will use a distributed self-organized approach to grow clusters with successively increasing the power levels and counting the additional nodes seen in the neighborhood. Informally, this process works as follows. In the beginning, all nodes are cluster heads for its own singleton cluster. To recruit members into its cluster, a cluster head will increment its power level to reach the next larger area and counts the number of additional nodes seen. If the delta change is positive the cluster size grows. The procedure stops whenever the delta change compared to previous step is positive or power level m is reached. All the decisions are made locally in each station.

Greedy Cluster Forming Algorithm

1. BEACON PHASE
   - Initially all nodes are cluster heads. Each node sends a beacon signal with the highest power level either using TDMA or some MAC protocol.
   - All nodes listen to these beacons. It is assumed that each node determines the distance of all its

neighbors via the signal strength received.

2. CLUSTER BUILDING PHASE
Each node uses a greedy method by increasing its cluster power level one step at a time until the incremental change in number of nodes seen reduces or the power level reaches a specified maximum. This is accomplished completely locally using the cached distance information of all neighbors heard in the Beacon Phase. The maximum level is a parameter chosen to ensure inter-cluster and network connectivity.

3. RECRUITING PHASE
Each node sends a recruiting message containing all the Ids of the neighbors with its power level computed in the Cluster Building Phase. A node joins the first cluster that it hears and whose Id is greater than its own. Once a node has joined a cluster, it leaves the recruiting phase.

4. CLUSTER FORMING PHASE
    - All remaining cluster candidates send a cluster forming message to its member(s) with the cluster level and all the Ids of the neighbors within its cluster. Any node detecting that it cannot reach one or more nodes in this list will notify this information to its cluster head. A Cluster collects all such information, if any, makes decisions on which nodes to include, sends a finalizing message with a list of accepted and not accepted members
    - At the end of this phase, any node that has not received a cluster forming or finalizing message will form a singleton cluster and its level is the power level to reach its nearest neighbor.

In terms of complexity of messages and energy costs, in this greedy algorithm Phase 1 has O(N) messages of fixed length, Phase 2 has only local computations, Phase 3 has O(N) messages of relatively short length, and Phase 4 has O(no. of clusters) messages. The energy costs depend on the number of bits in different messages and the number of operations in computation. The cluster formation is a one time initial cost with small incremental cost for maintenance.

3 Adaptation of ODMRP

In this section, we will show how a multicast protocol can be adapted using our clustering in power controlled networks. On Demand Multicast Routing Protocol (ODMRP) is an ad hoc network multicast protocol which builds the mesh structure on demand to provide multiple paths to multicast members [4]. ODMRP is designed for ad-hoc wireless networks with mobile hosts where bandwidth is limited, topology changes frequently and rapidly, and power is constrained. In any multicast protocol, there should be procedures for data forwarding, members joining, and leaving. In ODMRP group membership and multicast routes are established on demand.

In our adaptation scheme, there are three major steps: (1) form clusters using the greedy heuristic and generate the supernode topology using the cluster heads; (2) execute ODMRP on this supernode topology; (3) forward multicast from a sender to multicast receivers through their respective cluster heads. All the major steps of ODMRP will be performed on the supernode topology, and therefore, there is no need to make any modification in the functional steps of the protocol for its use in this topology. The head node of the cluster in which a sender node resides will act as the sender for that multicast in the supernode topology. One or more multicast receivers in a cluster will have their head node as the receiver supernode. With multicasting in dense ad hoc networks, there can be multiple receivers in a neighborhood, the clusters facilitate efficient local broadcasting and significantly reduces the overhead during ODMRP member advertising and route discovery phase. ODMRP relies on frequent network-wide packet flooding and performing this on the supernode topology with fewer nodes helps improve the performance compared to using ODMRP on the entire topology. Another benefit with our approach is improved scalability of multicast protocols due to clustering.

The other part of adaptation in power controlled networks is the power levels used by various nodes in performing multicast. A cluster head node will use the power level just enough to reach all nodes within its cluster. Nodes within a cluster will use power level sufficient to reach all other nodes within their cluster as discussed in the clustering section. Internal nodes not participating in multicast will be turned off to conserve energy. Unlike in fixed range ad hoc network nodes, here the power level needed between different cluster heads can be different. The edge weight on inter-cluster links will specify the level needed for communication. For data transmissions on the supernode topology, each cluster will use the appropriate level to reach one or more of its neighbors as required by the steps of ODMRP. In other words, a cluster head will use the maximum of the edge weights of its inter-cluster links to broadcast a packet to all its neighbors. Since the supernode topology will be used for all multicast data forwarding, the nodes on this topology will do more work than other nodes. For balancing energy depletion, nodes in a cluster take turn to become cluster heads using some round robin schedule or with some energy threshold.

Channel Frequency	2.4GHz
Bandwidth	2Mbits/s
Radio Type	Accumulating Noise
Propagation Model	Free Space
Propagation Pathloss Model	Two-Ray Ground Reflection
Max. Radio Tx Power	15 dBm
Radio Rx Sensitivity	-91.0 dBm
Radio Rx SNR Threshold	10.0dBm
Maximum Power Levels	10
Field Dimension	500 x 500 square feet
Simulation Duration (unicast)	300 simulation secs
Simulation Duration (multicast)	120 simulation secs
MAC Protocol	IEEE 802.11

Table 1. Simulation Parameters

4. Simulation Experiments

To demonstrate the benefits of our adaptation techniques, we enhanced the GloMoSim package with wireless nodes having multiple power levels for transmission. We added code to determine the appropriate power level to use for transmission from a node to its intended recipients in a hop. We also enhanced GloMoSim to calculate the energy costs with power levels for various communication activities. For cluster forming, we developed a program, HMMA 1.0, and interfaced it to GloMoSim 2.0 package. The program is written in C++, runs on Microsoft Windows and provides GUI to tune various parameters for clustering and generating the supernode topology. This front-end software can be easily modified to generate cluster data for other simulation packages as well. We conducted extensive simulations using the enhanced GloMoSim to evaluate collision, delay, and power consumption performances for ODMRP with our clustering and without clustering. The simulation parameters used are given in Table 1. For the non-cluster simulation of ODMRP, we assumed that all nodes use the same power level, as that is considered to give good perfomance [12]. To be fair in comparison, we conducted non-cluster simulations with each of the possible power levels and took the best results. In all the simulation results shown in this section we are comparing the cluster results with the best power level results of non-cluster runs.

Our first simulation runs investigated the improvements in network life and connectivity with 200 nodes in a 500x500 sq. ft. terrain. We used some FTP and HTTP applications and a few generic CBR sources for generating the traffic. FTP file sizes are chosen randomly by GloMoSim and the packet sizes are also chosen at random. For the HTTP traffic, we have one client requesting pages of varying sizes from five httpd servers. For the CBR traffic, we used five sources that were alive during the entire simulation time of 300 ticks. We simulated several rounds and changed the set of senders and receivers for each round. As the simulation proceeds, nodes consume energy when packets are transmitted and received. We used our enhanced GloMoSim to calculate the energy consumptions and also included the costs in collisions to access the shared wireless media. Nodes die if they deplete all their available energy. Figure 2 (top-left) and (top-right) show the distribution of nodes that are still alive after 10 and 20 rounds respectively without clustering and Figure 2 (bottom-left) and (bottom-right) show these results in the form of energy remaining in clusters. From the two figures, with using clusters, more nodes are still alive after 20 rounds of runs although with less energy. We can clearly see from these figures that our clustering significantly improves the network life time and connectivity.

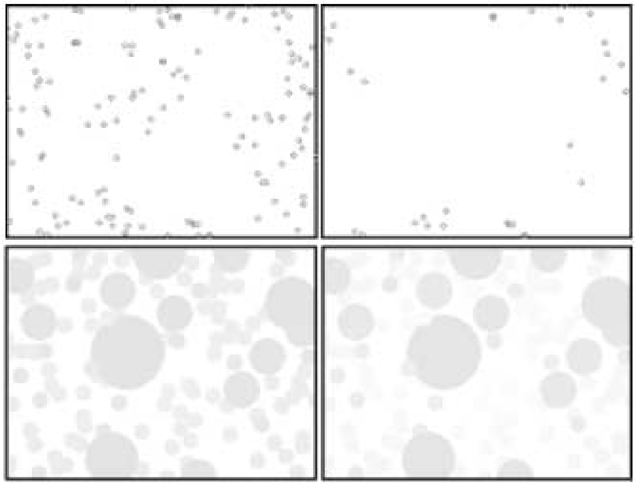

Figure 2. Node Energy Distribution after 10 and 20 rounds of runs.

Our next simulation runs compare the performance of ODMRP with our adaptation using clustering and ODMRP without clustering. We conducted extensive simulations with 75 to 300 nodes in a 500x500 sq. ft. terrain. For all the simulation runs, we used 13 multicast groups, a total of 57 members in these groups, and 29 sender nodes. For both clustering and non-clustering simulation runs, we used identical initial node distribution, multicast groups, and senders/receivers. For multicast data traffic, we used only CBR sources with packet start times varying from 0 to 50 simulation seconds and inter-arrival time of 500 msec simulation time. We collected various statistics including average packet delay, collision and energy cost. For the non-clustering simulation runs, we simulated ODMRP with all nodes having the same power levels and for all the m possible power levels and took the set of best results. Figure 3 (top plot) shows the comparison of average end-to-end delay with and without clustering. When the number of nodes increases, the delay in a non-clustering run increases. With clustering, it does not change much and is at least 200% on the average compared to non clustering runs. Figure 3 (middle plot) shows the collision comparison. In the non clustering runs, the number of collisions increases with number of nodes. With clustering, the number of collision incidents stays relatively the same and it is at least 100% better on the average. Figure 3 (bottom plot) shows the energy consumption in the clustering runs and non clustering runs. With our adaptation using clustering, there is up to 300% improvement over non clustering runs on the average. In all the plots, we show the two best results of non clustering runs, e.g. 13dBm and 15dBm. For the clustering runs, the radio power corresponding to the maximum power level is 15dBm.

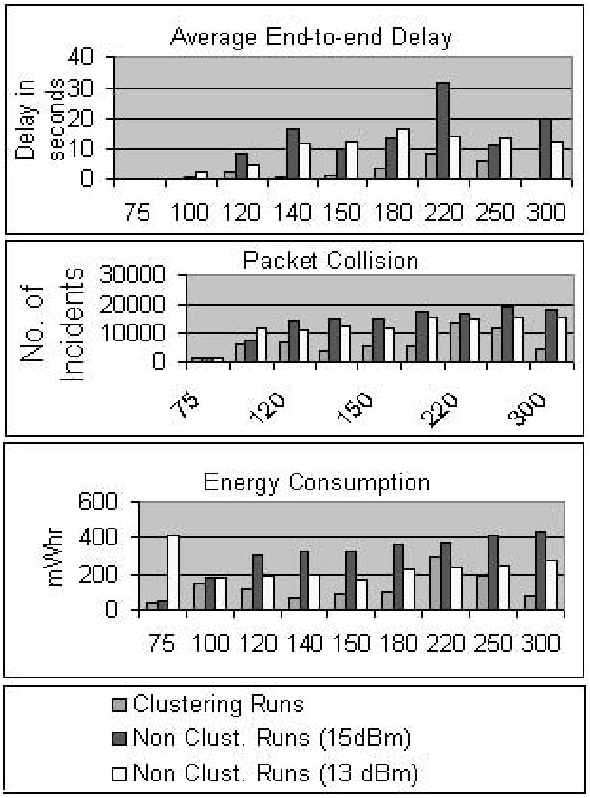

Figure 3. End-to-End Delay, Collision and Energy Dissipation Comparison.

One noticeable feature of clustering is that the gains do not degrade when the number of nodes changes. It is especially important for protocol scalability.

5. Conclusion and Future Work

In this paper, we presented a general methodology for adapting any existing ad hoc network protocol to power controlled networks. When nodes have multiple power levels for transmission, our approach is to use clustering to generate a supernode topology and execute a given protocol on this supernode topology. We developed a front-end software for clustering and interfaced it to GloMoSim 2.0 simulation package. We also enhanced the GloMoSim package with the ability to simulate nodes with multiple discrete power levels for transmission. We demonstrated our methodology and results through extensive simulations with ODMRP. Results show up to 100% improvement in energy consumption for multicasting compared to running ODMRP without clustering. In future work, we will evaluate our adaptation scheme with mobility of nodes, adapt other protocols, and evaluate the possibility of premature network partition.

References

[1] N. Bambos. Toward power-sensitive network architectures in wireless communications: Concepts, issues, and design aspects. *IEEE Personal Communications*, pages 50–59, June 1998.

[2] J. H. Chang and L. Tassiulas. Energy conserving routing in wireless ad hoc networks. *Proceedings of IEEE Infocom 2000*, March 2000.

[3] T. Kwon and M. Gerla. Clustering with power control. *Proceedings of IEEE MILCOM99*, November 1999.

[4] S. Lee, W. Su, and M. Gerla. On demand multicast routing protocol in multihop mobile wireless networks. *ACM/Baltzer Mobile Networks and Applications*, 2000.

[5] R. Nagpal and D. Coore. An algorithm for group formation and maximal independent set in an amorphous computer. *MIT Artificial Intelligence Laboratory Technical Report 1626*, 1998.

[6] K. Nakano and S. Olariu. Randomized initialization protocols for ad hoc networks. *IEEE Transactions on Parallel and Distributed Systems*, 11(7), July 2000.

[7] ORiNOCO. Orinoco wavelan radio. *http://www.orinoco.com*, 2000.

[8] R. Ramanathan and R. Hain. Topology control of multihop wireless networks using transmit power adjustment. *Proceedings IEEE Infocom 2000*, March 2000.

[9] T. S. Rappaport. *Wireless Communications: Principles and Practice*. Prentice Hall, 1995.

[10] Rockwell. Electronic controls and communications, wins radio. *http://www.rockwell.com*, 2000.

[11] Sincgars. An/arc-222 sincgars radio. *http://www.fas.org/man/dod-101/sys/ac/equip/an-arc-222.htm*, 2000.

[12] H. Takagi and L. Kleinrock. Optimal transmission ranges for randomly distributed packet radio terminals. *IEEE Transactions on Communications*, COM-32(3), March 1984.