Fault Tolerant and Energy Efficient Routing for Sensor Networks

Sinem Coleri and Pravin Varaiya
Department of Electrical Engineering and Computer Science
University of California, Berkeley, CA 94720
Email: {csinem, varaiya}@eecs.berkeley.edu

September 1, 2004

Abstract

The paper presents a fault tolerant routing algorithm that maximizes the lifetime of a sensor network by adjusting the number of packets traversing each node over multiple routes. An LP formulation gives the optimal single route. A distributed, iterative algorithm based on least cost path routing approximates the LP solution. Multiple path routing extends the iterative solution to increase resilience to link failures. Simulations show significant increase in network lifetime, and the tradeoff between the number of successful packet transfers and network lifetime for different multipath routing mechanisms.

1 Introduction

A wireless sensor network comprises a group of nodes, each with one or more sensors, a processor, a radio and a battery. Data packets from all nodes are destined for the same collection node, called the access point. Shortest hop routing is the most common routing protocol for ad hoc wireless networks [1], including table-driven protocols DSDV (Destination-Sequenced Distance-Vector), WRP (Wireless Routing Protocol), and source-initiated protocols AODV (Ad Hoc On-Demand Distance Vector) and DSR (Dynamic Source Routing). But shortest hop routing is not suitable for sensor networks with many flows towards one access point, as the elimination of a node may disconnect a large number of nodes from the access point. Various power-aware metrics are discussed in [2] to find the traffic distribution that balances energy consumption among nodes. These metrics include the time to network partition [3, 4] and the cost per packet [2, 3].

The first part of this paper gives an LP formulation of the lifetime maximization and approximates the LP solution by a sequence of least cost path calculation problems, in which the path cost is either the sum or the maximum of the cost of the nodes on that path, and the cost of each node is a function of its initial and remaining battery energy.

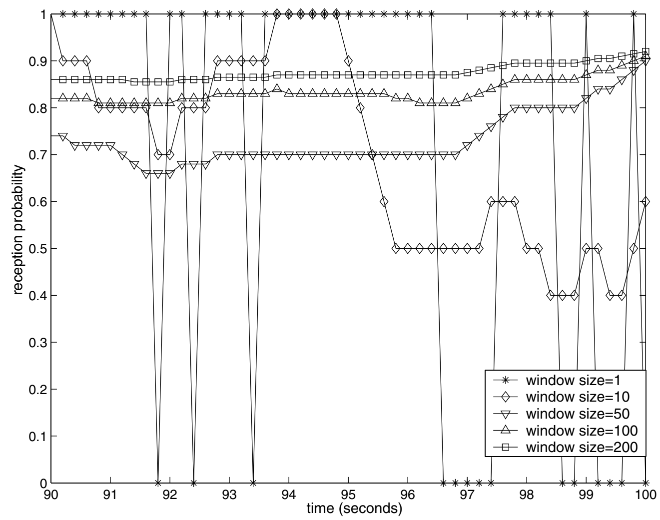

Figure 1: Reception probability of a 36 feet long wireless link for different moving window sizes.

The second part of the paper extends energy-efficient routing over multiple paths to increase network reliability under link failures. Figure 1 shows how the link quality between two Berkeley mica2dot [5] sensor nodes varies over time. The link quality at a distance greater than a certain value, which is $30ft$ in this case, shows similar behavior. Multipath routing can be used to provide robustness of the system against this kind of link failures. Multiple paths are discussed in [6] to rapidly find alternative paths between source and sink. The first to combine multiple path and minimum energy routing is [7]. The goal is to minimize the total energy consumed in forwarding each packet between a source and a destination. However, the impact of this routing on network lifetime is not considered. We introduce an algorithm to find multiple paths in order to maximize the network lifetime while increasing network resilience.

Section 2 presents the background assumptions. Section 3 formulates the LP problem and gives the iterative algorithm. Section 4 discusses multiple path routing. Section 5 provides some concluding remarks.

2 Assumptions

The following assumptions underly this study.

1. The network comprises one access point (AP) and several sensor nodes that generate data for transfer to the AP without aggregation.
2. All nodes have the same transmission power, so links are bidirectional, as required by protocols such as distributed Bellman-Ford algorithms [8].
3. Network topology is represented by a graph $G = (V, E)$. V is the set of nodes; the AP is node 1; $N = |V|$ is the number of nodes. Edge $(i, j) \in E \subset V \times V$ if nodes i and j can communicate.
4. Sensor nodes do not fail. Links fail independently with the same probability. Resilience to link failure is achieved by link-disjoint multiple paths in Section 4.
5. Nodes generate data at a specific rate, which may be different for each node.
6. Power is consumed by the radio, sensor and microprocessor. The MAC protocol keeps radios in sleep mode when they are not transmitting or receiving.
7. The lifetime of the sensor network is the time when the first node "dies" (consumes all its energy).

3 Energy Efficient Single-Path Routing

The routing algorithm assigns optimal packet flow rate f_{ij} from node i to node j. If two outgoing flows from node i are non-zero, choosing the same next hop from i is not optimal; the next hop from node $i \in V$ should alternate among nodes $j \in V$ with $f_{ij} > 0$.

Section 3.1 gives the LP formulation for optimal flow rates. Section 3.2 gives iterative distributed algorithms that approximate the centralized LP solution. Simulations in section 3.3 show the advantage of energy efficient routing over minimum hop routing.

3.1 Linear Programming Formulation

The variables in the optimization problem of Figure 2 are the packet flow rates f_{ij}, the average packet transmission time from i to j per unit time, and the network lifetime t. The objective is

Maximize t
Subject to: $f_{ij} >= 0$ for $i, j \in [1, N]$
$f_{ij} = 0$ for $(i, j) \notin E$
$\Sigma_j f_{ij} - \Sigma_j f_{ji} = g_i\, tt$ for $i \in [2, N]$
$t(\Sigma_j p_{tx} f_{ij} + \Sigma_j p_{rx} f_{ji} + p_s g_i + (1 - \Sigma_j f_{ij} - \Sigma_j f_{ji}) p_l) <= e_i$ for $i \in [2, N]$

Figure 2: LP problem to determine optimal link flows

to maximize network lifetime.

The first and second constraints are obvious. The third says flow is conserved at all nodes except AP. The fourth says that the initial energy e_i bounds the total energy consumed. Energy spent in transmission (reception) is $\Sigma_j p_{tx} f_{ij}$ ($\Sigma_j p_{rx} f_{ji}$), where p_{tx} (p_{rx}) is the energy per unit time to transmit (receive). Energy spent in listening is then $(1 - \Sigma_j f_{ij} - \Sigma_j f_{ji}) p_l$, where p_l is the energy spent by the radio in sleep mode per unit time. Finally, the energy spent in sensing is $p_s g_i$, where p_s is the energy spent in obtaining the samples in one packet.

This is an LP problem because all constraints are linear after we replace variables f_{ij} by $t f_{ij}$. A centralized solution to the LP problem can be used to generate a single path from any sensor node to the AP at each time to attain the optimal flow rates at each link at the end.

3.2 Iterative Approach

Unlike the centralized LP solution, we now present a periodic, iterative, distributed *least max-cost* path algorithm. We compare its performance with the distributed *least sum-cost* path algorithm, which includes previous work on minimizing per packet cost for different cost functions in [2, 3]. (Each iteration is based on the energy remaining in the nodes.) The *least max-cost* path algorithm converges to the optimal LP lifetime.

The *directed* graph G is $G_d = (V, E_d)$, with $< i, j > \in E_d$ and $< j, i > \in E_d$ for each $(i, j) \in E$. The iterative algorithm assigns a cost C_{ij} to every link $< i, j > \in E_d$ and then finds shortest path tree from AP to all the nodes at the beginning of each time period to be used for routing until the end of that period.

Since the cost of the links are used in finding the shortest path from AP to the sensor nodes, the cost of the directed link $< j, i >$, C_{ji}, is equal to the cost of including the node i on the path. The cost of node i at the p-th iteration is the ratio of the to-

operation	power consumption
transmitting one packet	$0.92mJ$
receiving one packet	$0.69mJ$
listening to channel	$29.71mJ/sec$
operating radio in sleep mode	$15\mu J/sec$
sampling sensor	$1.5\mu J$/sample

Table 1: Power consumption in Berkeley mica nodes.

tal energy consumed up to period p over the total battery energy:

$$C_i = \frac{\text{total energy consumed up to now}}{\text{total battery energy}} \quad (1)$$

$$= p * \frac{\Sigma_j p_{tx} f_{ij} + \Sigma_j p_{rx} f_{ji} + p_s g_i + (1 - \Sigma_j f_{ij} - \Sigma_j f_{ji}) p_l}{e_i}. \quad (2)$$

Here f_{ij} is the average resulting flow rate on link $< i, j >$. C_i increases from 0 to 1 as network evolves. The link cost function can be any increasing function d, $C_{ji} = d(C_i)$. For instance, the cost in [3] is the ratio of total battery energy to the remaining energy, given by $d(x) = \frac{1}{1-x}$.

The next step is to calculate the shortest path tree from node 1 (AP) to all nodes. The cost of a path is either the sum of the costs of the links in the path for the least sum-cost path algorithm, or the maximum of the link costs for the least max-cost path algorithm. A Bellman-Ford procedure can be used to obtain the optimum tree for both algorithms.

3.3 Simulations

In the simulations, nodes are randomly distributed in a circular area of radius 100 units. The transmission range is slightly larger than that necessary for network connectivity [9]. There is an edge between the nodes closer than this transmission range. The results below are averages of the performance of ten different random configurations unless otherwise stated.

The power consumption figures are in Table 1. The transmission rate is 50 kbps. The packet generation rate g_i at each node i is $1/30$ per second, which is a typical value for traffic monitoring [10]. The energy e_i for all nodes except AP is for a pair of AA batteries, or 2200mAh at 3V. The sampling rate is 128Hz at each node.

Figure 3 compares the lifetime of the network using the optimal LP solution with minimum hop routing. The LP solution increases network lifetime by 50-150 days. If energy spent in sampling were ignored in the lifetime calculations as in [3], the LP solution would show further improvements.

The distributed, iterative algorithms approximate the solution of LP formulation by iteratively solving minimum cost path problems. The critical parameters are *the type of least cost path algorithm*, *step interval* and *cost function*. *Step interval* is the length of each iteration during which the same routing paths are used. *Cost function* emphasizes different battery levels at different intensities in least sum-cost path algorithm through the function $d(x)$ in Section 3.2.

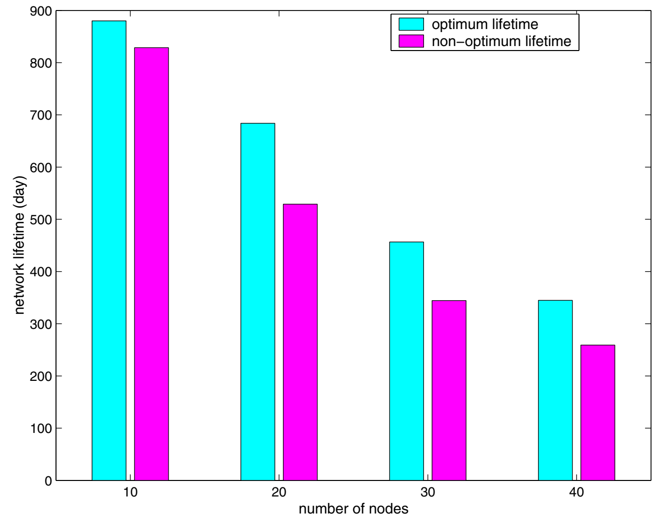

Figure 3: Comparison of the lifetime for optimum and minimum hop routing.

Figure 4 shows the network lifetime for iterative algorithms. As the step interval increases, the lifetime of the network decreases because there are fewer iterations. For small step intervals, the lifetime is almost equal to the optimal for least max-cost path algorithm, and close to optimal for least sum-cost path algorithm with cost functions $d(x) = x^{50}$ and $d(x) = 1/(1-x)^{50}$. This suggests that for least sum-cost path, with $d(x) = x^n$, n should not be too small since the cost becomes almost linear, making it hard to differentiate between a path having one node with small residual energy and many other nodes with high residual energy, and a path with many nodes with medium residual energy. Also, n should not so large that it is hard to differentiate between $d(x) = x^n$ values unless x is very close to 1. On the other hand, as n gets larger in $d(x) = 1/(1-x)^n$ it gets closer to the optimal value since the path cost is close to the maximum of the link costs on the path.

4 Energy Efficient Multipath Routing

The goal of multipath routing is to increase the resilience of the network against the failure of the links by trading off the energy consumption. The cost of a path p is the maximum of the costs of the links in p whereas the cost of a set of paths

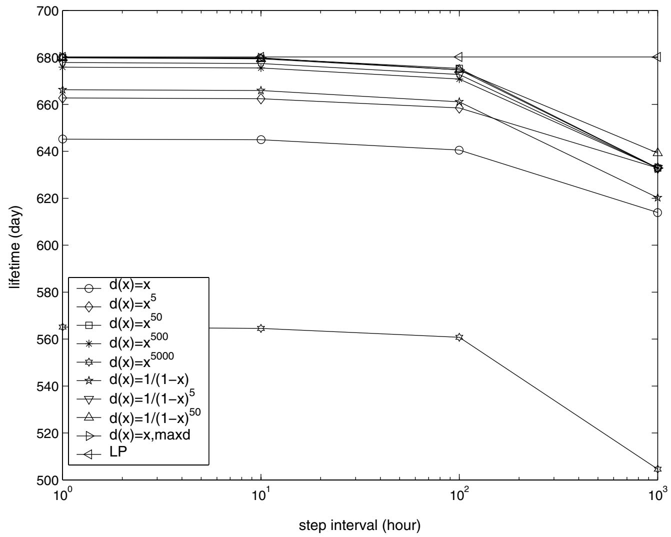

Figure 4: Average battery lifetime of the random configurations of 20 nodes for different cost functions and step intervals.

$P = \{p_1, p_2, ..., p_k\}$ is the maximum of the cost of each path in P. The minimum cost k link-disjoint path problem is: Given $G_d = (V, E_d)$, find a set of k link-disjoint paths P from node AP to a sensor node such that the cost of the set is minimized.

4.1 Multiple Path Routing Algorithm

The algorithm in [11] gives a minimum cost edge-disjoint path algorithm in an undirected graph where the cost of a path is the sum of its link costs. We modify this algorithm to find minimum cost edge-disjoint paths in G_d with the cost of a path being the maximum of its link costs based on the assumption that $< i, j >$ and $< j, i >$ are not disjoint links. Since nodes i and j transmit at the same power, links $< i, j >$ and $< j, i >$ both fail at the same time. Therefore, if one of them is used in a path, the other cannot be used in another path to provide edge disjoint multiple paths.

This is the algorithm for minimum cost two edge-disjoint paths from AP to a sensor node i:

1. Run Bellman-Ford algorithm to find the shortest path from AP to the sensor node i in G_d.
2. For each $< i, j >$ on the shortest path, remove the edge $< i, j >$ and set the cost of the link $< j, i >$ to $C_{ji} = -C_{ij}$.
3. Run Bellman-Ford algorithm to find the shortest path from the AP to node i in the modified graph.
4. Remove the overlapping edges of the two paths. The desired pair of paths results.

Removing the arcs belonging to the shortest path in step 2 ensures that this path is not reproduced when the Bellman-Ford algorithm is run in the modified graph. Moreover, modifying the cost of the link in reverse direction permits the interlacing of the shortest path to be found in Step 4 with the one found in the original graph while eliminating the possibility of including the reverse link before removing the forward link. At the end of the algorithm, two disjoint paths are obtained by erasing the interlacing part of the two paths. Allowing for interlacing and the negativity of the links leads to optimality.

Minimum cost k edge-disjoint paths are generated iteratively from minimum cost $k - 1$ edge disjoint paths. The algorithm is similar to the 2 edge-disjoint path case:

1. For each $< i, j >$ on the minimum cost $k - 1$ edge disjoint paths, remove the edge $< i, j >$ and set the cost of the link $< j, i >$ to $C_{ji} = -C_{ij}$.
2. Run Bellman-Ford algorithm to find the shortest path from the AP to node i in the resulting graph.
3. Remove the overlapping edges of the different paths. The desired k paths results.

We omit the proof of optimality of the algorithm.

If node i has infinite cost in Step 2, then no more edge disjoint paths can be found. Then the algorithm stops before reaching k. An example of generating two edge-disjoint paths from AP to node 4 is given in Figure 5. The original network is given in Figure 5(a). The first path found is $\{AP, 2, 3, 4\}$, with the modified graph given in Figure 5(b). The next path found is $\{AP, 6, 7, 3, 2, 5, 4\}$. The two edge-disjoint paths resulting from the removal of the overlapping edges are then $\{AP, 2, 5, 4\}$ and $\{AP, 6, 7, 3, 4\}$. Figure 5 (c) gives the modified graph as a result of these two paths. Since there is no path from AP to node 4 in this modified graph, no more edge-disjoint paths can be found.

The algorithm is executed k times for each node. The running time of each iteration is dominated by Bellman Ford algorithm. The resulting running time of the algorithm is therefore $O(k|E||V|^2)$.

The multipath routing can be implemented in a centralized manner at the AP by running the above algorithm for each node after collecting the topology information and then disseminating this information to the nodes. The algorithm can be made distributed by running a distributed Bellman-Ford algorithm for each path updating the link cost according to the previous paths found. Bellman-Ford algorithm takes $|V|$ pulses for each path so $k|V|$ pulses to generate k edge-disjoint paths to each node.

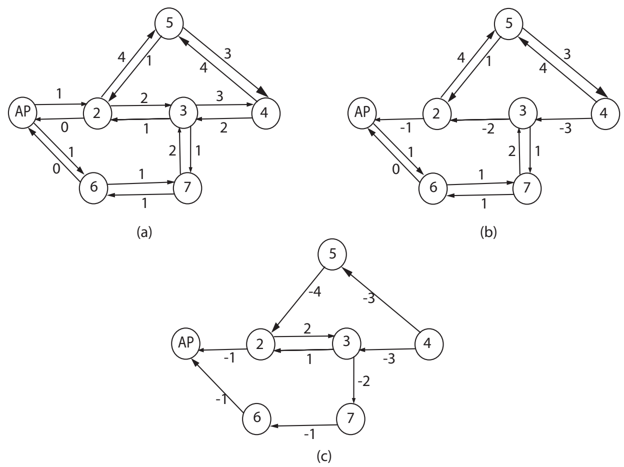

Figure 5: (a) Original graph $G_d = (V, E_d)$. (b) Modified graph after finding the path $\{AP, 2, 3, 4\}$. (c) Modified graph after finding the paths $\{AP, 2, 5, 4\}$ and $\{AP, 6, 7, 3, 4\}$.

4.2 Simulations

The simulations help to understand the tradeoff between the resilience of the network to link failures and the extra energy consumption.

Figure 6 shows the average percentage of the nodes that can send their packets successfully to the AP for different link failure probabilities, p, for a random 20-node network. The cost $d(x) = x$ corresponds to minimizing the maximum cost on the k disjoint paths for the cost defined in Section 3.2 whereas $d(x) = 1$ corresponds to choosing any k disjoint paths without considering the real costs. The percentage of the successful nodes increases by $15 - 30\%$ for both cases. The rate of increase decreases with the number of paths since no more paths can be found for the nodes (The average number of paths found for the nodes are $1, 1.8421, 2.6316, 3.2632, 3.8947, 4.4211$ for $1, 2, 3, 4, 5, 6$ paths respectively). The $d(x) = 1$ case gives slightly better result since the total number of edges in the least cost paths for $d(x) = x$ may be larger than that for $d(x) = 1$.

Figure 7 shows the average percentage of the lifetime that can be achieved at different multiple paths for a random 20-node network. Since the edge-disjoint multiple paths between a sensor node and the AP can contain common nodes, we consider two cases: *join* and *no join*. The *join* case assumes that a node common to multiple paths for the same source-destination pairs waits until it receives the same packet over multiple paths and then sends it only once whereas *no join* case assumes that they are transmitted separately. The $d(x) = x$ cost increases the network lifetime by 20% over the $d(x) = 1$ cost. Moreover, the lifetime only decreases by 50% for 6 multiple paths, where an average of 4.4211 paths are found for each node. This slow decrease is

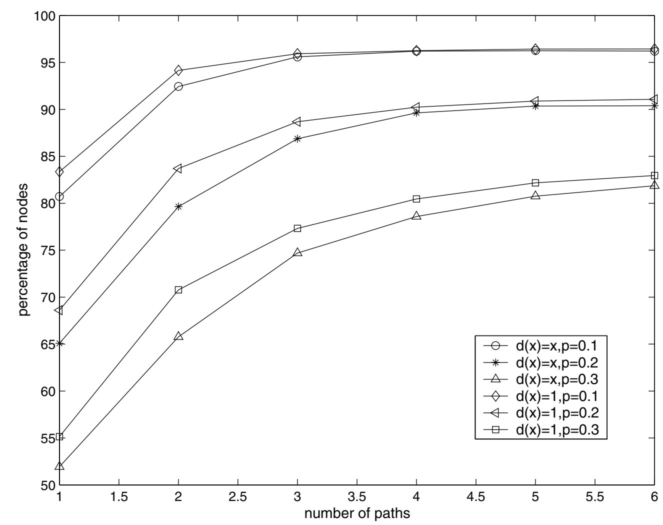

Figure 6: Average percentage of the nodes that can send their packets successfully to the AP for a random 20-node network.

because the nodes may not find another edge-disjoint path after choosing minimum cost paths, thereby preventing the load of critical nodes from increasing linearly with the number of multiple paths. The same kind of behavior is observed for the simulations of different number of nodes in the number of successfully received packets and the lifetime as a result of multipath routing. The only difference is that the percentage of the successful nodes decreases as the number of nodes increases since the number of hops to reach AP increases.

5 Conclusion

In this paper, we present a routing protocol for sensor networks, in which all data packets are destined for a single collection node, with the goal of maximizing the time duration until the first node dies.

We first focus on single path routing. We formulate the path optimization as a Linear Programming (LP) problem in which the objective is to maximize the lifetime of the network. This optimal routing is shown to increase the network lifetime by 50-150 days over the minimum hop routing. We then describe a distributed algorithm based on iterative least cost path routing where the cost of each path is either the sum or the maximum of the cost of the nodes on that path at each step. The distributed algorithm is extended to include multiple paths between each source and destination in the network to provide the resiliency of the network to link failures. We introduce a multipath routing algorithm that is optimal in terms of minimizing the maximum cost on the multiple paths from each sensor node to the AP. We show that the

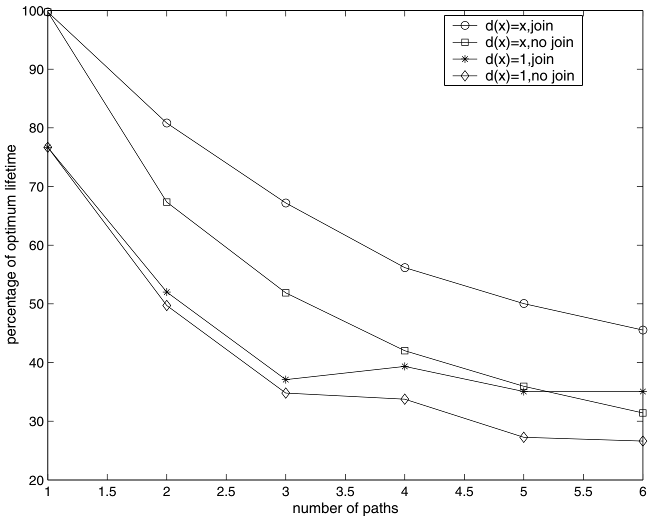

Figure 7: Percentage of optimal lifetime that can be achieved at different multiple paths for a random 20-node network.

number of successfully received packets at the AP increases by 20% for a decrease of 50% in the resulting lifetime of the network.

References

[1] E. M. Royer, and C. Toh, *A review of current routing protocols for ad hoc mobile wireless networks*, IEEE Personal Communications, pp. 46-55, April 1999.

[2] S. Singh, M. Woo, and C. Raghavendra, *Power-aware routing in mobile ad hoc networks*, MOBICOM 1998, pp.181-190.

[3] J-H. Chang and L. Tassiulas, *Energy conserving routing in wireless ad hoc networks*, IEEE INFOCOM 2000, vol. 1, pp. 22-31.

[4] M. Bhardwaj and A. P. Chandrakasan, *Bounding the lifetime of sensor networks via optimal role assignments*, IEEE INFOCOM 2002, pp.1587-1596.

[5] *www.tinyos.net*

[6] D. Ganesan, R. Govindan, S. Shenker and D. Estrin, *Highly-resilient, energy efficient multipath routing in wireless sensor networks*, Mobile Computing and Communications Review, vol. 4, no. 5, October 2001.

[7] A. Srinivas and E. Modiano, *Minimum energy disjoint path routing in wireless ad hoc networks*, ACM MOBICOM 2003.

[8] Swetha Narayanaswamy, Vikas Kawadia, R. S. Sreenivas, and P. R. Kumar, *Power control in Ad hoc networks: theory, architecture, algorithm and implementation of the COMPOW protocol*, Proceedings of European Wireless 2002, February 2002, Italy.

[9] B. Krishnamachari, S. B. Wicker, and B. Bejar, *Phase transition phenomena in wireless ad hoc networks*, Symposium on Ad-Hoc Wireless Networks, GlobeCom 2001, San Antonio, Texas, November 2001.

[10] S. Coleri, *PEDAMACS: Power efficient and delay aware medium access protocol for sensor networks*, MS Thesis, Department of Electrical Engineering and Computer Science, University of California, Berkeley, December 2002.

[11] R. Bhandari, *Optimal physical diversity algorithms and survivable networks*, IEEE Symposium on Computers and Communications, July 1997, pp.433-441