LOW-OVERHEAD OPPORTUNISTIC ROUTING FOR WIRELESS AD HOC AND SENSOR NETWORKS IN A FADING ENVIRONMENT

A Thesis
Presented to
The Academic Faculty

by

Benjamin R. Hamilton

In Partial Fulfillment
of the Requirements for the Degree
Masters of Science in the
School of Electrical and Computer Engineering

Georgia Institute of Technology
December 2007

LOW-OVERHEAD OPPORTUNISTIC ROUTING FOR WIRELESS AD HOC AND SENSOR NETWORKS IN A FADING ENVIRONMENT

Approved by:

Professor Xiaoli Ma, Advisor
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor Guotong Zhou
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor David V. Anderson
School of Electrical and Computer Engineering
Georgia Institute of Technology

Date Approved: November 8, 2007

ACKNOWLEDGEMENTS

I would like express my sincere appreciation to the people that made this thesis possible. First I wish to thank my advisor Dr. Xiaoli Ma for her inspiration and constant prodding. I would also like to thank my committee for their patience and guidance. In addition I would like to thank Wei Zhang for his assistance with the MIMO variation of the protocol. Finally, I also would like to thank my wife, Lei, for her encouragement and support.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS iii

LIST OF TABLES v

LIST OF FIGURES vi

SUMMARY vii

I INTRODUCTION 1

1.1 Routing 2

1.2 Harnessing the Wireless Link 4

1.3 Opportunistic Routing 5

1.4 Channel Model 6

1.5 Organization of the Thesis 7

II PROTOCOL FRAMEWORK 8

2.1 Framework Description 8

2.2 Performance 10

2.3 Implementation Issues 18

III NON-COOPERATIVE ROUTING WITH COOPERATIVE DIVERSITY 21

3.1 Cooperative Diversity 21

3.2 Choosing the Timer Function 22

3.3 Performance 26

IV OTHER VARIATIONS BASED ON THIS FRAMEWORK 30

4.1 Multi-Antenna Networks 30

4.2 Including Distance 33

V CONCLUSION 36

REFERENCES 37

LIST OF TABLES

1 Timer functions and timer duration distributions for Rayleigh fading 23

LIST OF FIGURES

1 Example of path selection on a random network 9
2 Example of path on a random network 10
3 Illustration of effect of timer discretization 15
4 CDF of a collision-optimized timer distribution for c=0.02 16
5 Performance Comparison for Max-Delay normalized functions 24
6 Performance Comparison for Average Delay normalized functions . . 24
7 Performance for timer functions under a Rayleigh channel 25
8 Performance for timer functions under a realistic channel 25
9 PER vs. SNR ($c = 0.002$) for Two-Hop Transmissions 27
10 Multi-hop routing simulation: ATAN timers, $c = 0.002$ 29
11 MIMO Performance for two-hop transmissions 31
12 Performance of multihop system with varying numbers of antennas . 33
13 PER vs SNR for iPRRd, PRRd and NCRP protocols 35

SUMMARY

The development of miniaturized radio and sensing technologies have enabled the deployment of large quantities of wireless sensors capable of forming multi-hop networks. Emerging applications of this technology such as surveillance and disaster monitoring have throughput and efficiency requirements not met by current routing algorithms. These requirements are also shared by ad-hoc networks.

Early routing protocols for these wireless networks were based on algorithms designed for wired networks. Geographic routing (routing based on position), was proposed. These algorithms perform poorly since they do not account for the fading and interference effects of wireless channels. Recent protocols that have attempted to account for the wireless channel focus on single-hop situations and are not readily extensible to multi-hop networks.

In this thesis we present a framework for routing based on a distributed routing decision and provide several example protocols. This framework provides a cross-layer design where the routing decision is decided through silent negotiation between candidate relays. We investigate the performance and parameters of this framework. We then present an example protocol using this framework which provides low-overhead opportunistic routing using cooperative diversity. This protocol uses the intrinsic characteristics of the wireless channel to achieve diversity while still maintaining relatively low overhead. An adaptation of the protocol for heterogeneous networks equipped with multiple antennas has also been discussed and evaluated through simulations. We also investigate another protocol based on this framework using the product of the instantaneous packet reception rate and the marginal progress towards the destination as a routing metric, offering enhanced throughput.

CHAPTER I

INTRODUCTION

The advances in miniaturized radio and sensing technologies have enabled the construction of small battery powered sensors which can report results wirelessly to a base station. These cheap, disposable sensors can be rapidly deployed anywhere they are needed. Networks with such sensors have a number of military applications such as rapidly deployable communication infrastructure, surveillance, and targeting systems. They are also useful for weather or seismic monitoring or even monitoring disaster areas [2]. Due to the limited power available to nodes from their small batteries, transmission power is often limited, requiring nodes to transmit to the base station over multiple hops.

The situation is similar for ad hoc networks. More and more devices are acquiring wireless networking capabilities. User's expectations of the connectivity and capability of wireless networks are increasing, moving towards ubiquitous networking [25]. Current hierarchical architectures are extremely inefficient for short range node-to-node communication common in this paradigm, and thus do not have sufficient capacity to meet demand. In order to meet these requirements, a true ad hoc network where each node functions as both a wireless router and a client is required [1].

Both these problems require efficient routing algorithms. In computer networks routing is simply the problem of finding a path from one point to another [1]. Though so trivially stated, once the problem is distributed and replicated thousands of times it becomes difficult. When this problem is posed for wireless ad hoc networks and sensor networks which lack the hierarchy imposed on wired networks, finding a practical solution becomes truly challenging. Power limitations due to battery powered nodes

limit transmission range. The broadcast nature of wireless links causes inexorable interference. Mobility and the mercurial wireless channel collude to create a transient network topology that confounds traditional routing protocols. In spite of this, multi-hop routing in wireless networks has clear benefits: increased connectivity, increased capacity, and power efficiency. Traditional approaches to solve the wireless routing have split the problem into two distinct parts and addressed them separately. The first was a routing problem similar to that of wired networks. The second focused on the development of techniques to take advantage of the wireless link. More recent works have shown increased performance through addressing the combined problem as a whole.

1.1 Routing

The first component deals with designing a protocol to perform routing in the network. These designs are mostly adaptations of routing protocols designed for use in wired networks such as the Internet. Traditionally routing protocols were classified according to three primary types: proactive, reactive and hybrid. Recent works have added geographic routing.

Proactive routing protocols actively find routes to all destinations in the network regardless of whether they are being used. This makes them more efficient in networks with high utilization, but limits their scalability. In wired networks the scalability problem has been addressed by combining the use of a hierarchical design with route summarization. The most common approach uses the distributed Bellman-Ford algorithm to calculate path costs and choose the best cost. Nodes periodically transmit information on all the destinations they know about. After receiving, nodes update their routing tables with the current known best routes based on the information received in the updates. This type of protocol can cause routing loops in highly variable networks. Destination Sequenced Distance Vector (DSDV) [18] implements this type

of routing protocol. It uses sequence numbers for route updates in order to reduce the likelihood of forming a routing loop.

In reactive protocols nodes only calculate routes as they are needed. Reactive protocols have reduced storage overhead in networks with low utilization, but exhibit scaling problems in large or busy networks. Nodes can generate routes as needed by flooding a route discovery packet from the source to the destination and choosing the best route. In Dynamic Source Routing (DSR) [9], the intermediate hops are recorded in the route discovery packet and then the best route is sent backwards to the source. The source then adds the route list to each packet and routing is performed by simply sending to the next hop listed in the routing header. In Ad-hoc On-Demand Distance Vector Routing (AODV) [17], this is modified so the source does not need to put the hop list in the packet. The intermediate nodes maintain a table of active routes through them and know the next hop. Reactive routing protocols suffer from an increase in communication overhead in rapidly varying networks. Routes that are no longer valid need to be rebuilt. A few techniques have been proposed to allow partial route discovery, but still incur significant overhead.

Hybrid protocols attempt to combine the characteristics of both proactive and reactive protocols: nodes proactively maintain routes for all destinations nearby, but reactively generate routes for destinations that are unknown. An example of this is Zone Routing Protocol (ZRP) [7]. In ZRP, a global zone radius is set and nodes receive proactive route updates for destinations within this radius. When a source node needs to send a packet to destination outside the zone radius, a flooding algorithm is used to find a path and routing is performed similar to DSDV.

Geographic routing is able to respond more rapidly to network changes due to the deceased state and use of purely local information. The key behind geographic routing is the assumption that nodes are aware both of their own location and the locations of their neighbors. To send a packet a node must also know the location

of the destination. This is reasonable in sensor networks since the positions of nodes already need to be calculated to properly analyze the reported data. There have been a number of geographic protocols proposed. Some use the location information to form a planar graph and walk along the edges towards the destination [11, 12]. Another approach is to use a greedy algorithm to determine the next-hop node. This means a node chooses the neighbor closest to the destination as the next hop. In arbitrary networks it is possible that there is no node closer to the destination than the current node. For this reason, most protocols specify an alternate routing method for the case when greedy routing fails. This was originally proposed, as Cartesian Routing, as a solution in wired networks in [5]. It used flooding as the alternate routing method. Greedy Perimeter Stateless Routing (GPSR) [10] was proposed as a solution for wireless networks based on greedy geographic forwarding and uses perimeter routing as a fallback. These protocols perform reasonably in the rapidly varying network environments, but tend to prefer high-loss rate links [26], which results in reduced throughput and power efficiency. In [20] a modification to the greedy algorithm was proposed to incorporate the average packet reception rate (PRR) into the metric used by greedy forwarding, maximizing the product of the PRR with the distance instead of simply using the distance traveled towards the destination for each hop. This provides some improvement but increases overhead by requiring local dissemination of link loss rates and does not consider the effects of fading on the wireless link.

1.2 Harnessing the Wireless Link

The second component deals with harnessing the wireless link to reduce the effect of the disruptive wireless link characteristics and, if possible, achieve increased performance. The most common way to increase performance is through diversity. Diversity techniques improve performance in a manner similar to the increase in reliability

achieved in a system by having redundant backups. There are innumerable ways to obtain diversity in wireless systems. The most common ones are multipath and spatial diversity used in co-located multi-antenna systems. In these systems there is a channel between each transmitter and each receiver. Since the received signal is at least as strong as the strongest signal, this achieves diversity. In practice, coding techniques are used to trade diversity for extra throughput. Many papers have tried to extend these techniques to networks of discrete cooperating nodes. In [6, 15, 24] various cooperative coding schemes have been proposed. In [13] Laneman et al. developed an architecture for multi-node cooperative schemes based on the classical relay channel model. In this architecture a source would send a packet to the destination aided by one or more relays. This architecture was used in [4] where a protocol using selection diversity was employed. In this protocol, each relay generated a metric based on both the channel state between the source and relay, and the channel state between the relay and the destination. Each relay sets a timer based on this metric, and the relay whose timer expires first is used. These techniques are theoretically sound and effective in their assumed system models, but do not consider routing and are not easily extended to multi-hop networks

1.3 Opportunistic Routing

Recently some work in cross-layer design has resulted in a few protocols that take both aspects into account. Opportunistic routing is similar to geographic routing except that the routing algorithm is allowed to choose multiple candidate next-hop nodes which are considered equivalent for routing [3]. The actual next-hop node is decided based on the channel from the current hop to the next hop. This makes it apparent that opportunistic routing is simply an adaptation of cooperative diversity techniques. In opportunistic routing, the relays are chosen by the routing layer

and then relay selection is used to achieve diversity. In Selection Diversity Forwarding [14], the transmitter generates a list of candidate relays and transmits it along with the packet. The transmitter chooses the next hop as the candidate that claims to have correctly received the packet that makes the most progress towards the destination. Extremely Opportunistic Routing (ExOR) [3] takes a similar approach, except proposes a recursive multi-stage approach where the candidates consist of all nodes between the current position and destination. Since this method had significant overhead, numerous packets were grouped together in batches and sent simultaneously to amortize the overhead. In [22], opportunistic routing was posed as an optimization problem and the power efficiency was optimized. As shown in [21], these protocols suffer increased overhead that significantly degrades their performance, reducing power efficiency to less than greedy geographic routing for some classes of networks.

1.4 Channel Model

There are two channel models used in this work.

The first is the Rayleigh channel model. We characterize the wireless link $h_m(k)$ between nodes n_m and n_{m+1} in the kth time-slot by only considering the small-scale random fading effect $\eta_m(k)$:

$$h_m(k) = \eta_m(k), \tag{1}$$

where the $\eta_m(k)$'s are complex Gaussian distributed with zero mean and unit variance.

The other is a more realistic channel model. For this model, we characterize the wireless link by taking into account three effects [19]: the shadowing effect ζ_m, the attenuation due to the distance d_m and the small-scale random fading effect $\eta_m(k)$ as

$$h_m(k) = \zeta_m^{\frac{1}{2}}\, \eta_m(k)\, d_m^{-\frac{\alpha}{2}}, \tag{2}$$

where d_m is the distance between n_m and n_{m+1} and α is the power loss exponent with a value between 2 (free space) and 4. The shadowing component ζ_m is assumed having

a log-normal distribution [19] whose probability density function can be described as

$$f_\zeta(x) = \frac{1}{x\sigma_\zeta\sqrt{2\pi}} e^{-(\ln x - \mu_\zeta)^2/2\sigma_\zeta^2}, \tag{3}$$

with μ_ζ and σ_ζ^2 being the mean and the variance of $\ln x$. The large-scale shadowing effect ζ_m and the attenuation term $d_m^{-\alpha}$ do not change during the time period of interest and therefore they do not depend on k. For small-scale fading, we assume that there is non-line-of-sight (NLOS) and $\eta_m(k)$'s are complex Gaussian distributed with zero mean and unit variance. Within one time-slot, the link does not change. For different time-slots, i.e., different k, $\eta_m(k)$'s are independent.

1.5 Organization of the Thesis

The rest of this thesis is organized as follows. In Chapter 2, a framework for distributed routing in a wireless network is introduced and its characteristics are analyzed. In Chapter 3, a protocol based on the framework is examined and results are simulated. In Chapter 4, extensions of this protocol within the framework are presented and examined. In Chapter 5, we conclude and outline future research directions.

CHAPTER II

PROTOCOL FRAMEWORK

We propose a novel opportunistic protocol framework that can be used to implement a distributed selective combining scheme. Like a few of the opportunistic protocols mentioned in [21], this protocol achieves cooperative diversity using the routing decision. This protocol builds upon a new framework which enables distributed forwarding decisions based only on a node's own locally available information, requiring no negotiation among next-hop candidates and no neighbor location information.

2.1 Framework Description

Under this framework nodes use the following information to route a packet: the node's location, the previous hop location, the destination location, the angle spread parameter α, and some timer value generated based on only local information. In the case of the protocol we describe in Chapter 3, this timer value would be a monotonically decreasing function of the channel state information (CSI) between the previous hop and the current node. The routing process proceeds as follows:

1. Packet is transmitted (broadcast);
2. Nodes that receive the packet check if they are within an angle $\alpha/2$ from the direction specified in the packet:
   (a) The nodes within the region become candidates;
   (b) The nodes outside the region ignore the packet;
3. Candidates set a timer and listen while waiting for the timer to expire:

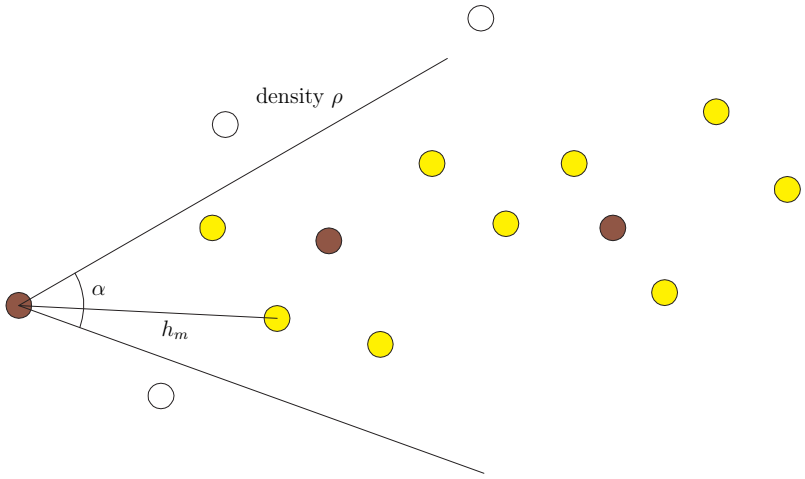


Figure 1: Example of path selection on a random network

(a) Nodes that overhear the same packet being forwarded by another node cancel their timer;

(b) The node forwards the packet when the timer expires;

Note that step 3a ensures that, in the ideal case (i.e., the transition time and propagation delay are ignored), only one node will forward the packet. In a realistic situation timers have only finite precision and the propagation delay and time needed for a node to switch from listening to transmitting are nonzero, so multiple nodes may try to forward the packet. This will generally cause a collision, resulting in the loss of the packet. This issue will be discussed further in Section 2.3.

Figure 1 shows an example: dark circles denote selected senders for next hops; light circles denote the involved nodes; and the hollow ones denote those not involved ones. Suppose node A transmits a packet towards some destination. All the colored nodes in range of node A are within the angle spread, so they will set a timer. If B is the node whose timer expires first, then B will forward the packet and the other candidates will cancel their timers. The nodes in B's angle spread (possibly including some that were also in A's angle spread) will set a timer and the process will repeat.

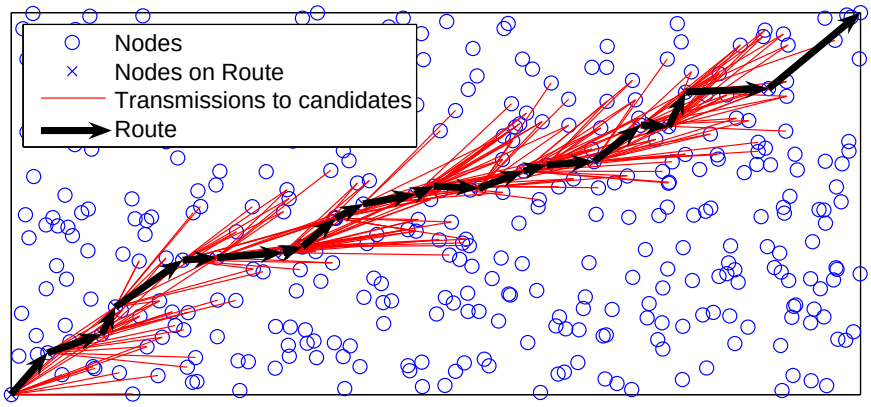


Figure 2: Example of path on a random network

A more detailed diagram of a simulated path chosen by this protocol is shown in Figure 2. The thin light lines link the broadcast source with the candidate relay nodes for a hop. The thick dark arrows link the broadcast source with the chosen relay, showing the actual packet routing path from source to destination. The direction was chosen to be towards the destination, and the angle spread was always set to 60 degrees.

The primary adjustable parameters for our protocol framework are the spreading angle α and the timer function. The spreading angle controls the number of candidate nodes and will be shown to control the tradeoff between diversity and collision probability. The same timer function is adopted by all nodes. It controls the extra transmission delay introduced by the protocol.

2.2 Performance

In this section, the impact of the timer generation function on the performance of the protocol is analyzed. We define two primary metrics for the purpose of this analysis:

- **Average Delay**: Since a node may transmit only after its timer expires, there is an extra delay introduced by our protocol. Average delay is a measure of this extra time. In our analysis it is measured as a proportion of the maximum

timer value.

- **Collision Probability**: This is the probability that at least one node has a timer that expires within some interval c from the time instant when the first timer expires. In this case, it is assumed that some portion c of the maximum timer value is required for signal propagation and for a node to disable the receiver prior to transmission.

2.2.1 Average Delay

Since the input to the timer function (i.e. the CSI) is random, the delay caused by the timer is also random. This means the average delay can be calculated as the expected value of the minimum of the M different random delay variables x_i as:

$$\overline{D} = E\{\min(x_i, i \in [1, M])\}. \tag{4}$$

If we assume the random delay variables are all independent and identically distributed (i.i.d.) with a cdf given by $F(x)$ and pdf given by $f(x)$, this simplifies to:

$$\overline{D} = N \int_0^\infty x f(x) \left[1 - F(x)\right]^{N-1} \quad dx. \tag{5}$$

Since in many cases it is desirable to place an upper bound to the delay incurred, we create $D_{\max}$ as the maximum acceptable timer value. This means for nodes calculating their timer starting values based on the CSI, if the initial value is less than $D_{\max}$, the node will start the timer. There are two advantages to define $D_{\max}$: (i) If a candidate node calculates a timer length above this limit, i.e., its channel response is weaker than a threshold, there is no need to start a timer; Therefore, nodes save some energy; and (ii) If all nodes experience weak channels, i.e., even the strongest channel is weak (the timer initial value is greater than $D_{\max}$), no node should forward this packet because most likely the packet has an error.

The probability that at least one timer is less than $D_{\max}$ is $1 - [1 - F(D_{\max})]^M$.

Thus, we obtain the average delay as

$$\overline{D} = \frac{M \int_0^{D_{\max}} x f(x)[1 - F(x)]^{M-1}\, dx}{1 - [1 - F(D_{\max})]^M}. \tag{6}$$

Let us clarify the effect that changing $D_{\max}$ has on the average delay. Taking the derivative of $\overline{D}$ in Eq. (6) with respect to $D_{\max}$, we obtain

$$\frac{\partial \overline{D}}{\partial D_{\max}} = \frac{M f(D_{\max})[1 - F(D_{\max})]^{M-1}}{1 - [1 - F(D_{\max})]^M}(D_{\max} - \overline{D}).$$

Note that this derivative is always positive (if $F(D_{\max}) < 1$) or zero (if $F(D_{\max}) = 1$). This means that as $D_{\max}$ increases, average delay increases. This is because when $D_{\max}$ increases, the timer range is more spread-out and thus the average delay increases.

2.2.2 Probability of Collision

In an earlier work [4], a similar solution on the collision probability for two-hop relay networks was analyzed. It is shown that the collision probability based on the timer distribution is

$$1 - P_{\text{coll}} = M(M-1) \int_0^{\infty} \int_{y_1+c}^{\infty} f(y_1) f(y_2)[1 - F(y_2)]^{M-2}\, dy_2 dy_1.$$

If we pull $f(y_1)$ out of the inner integral and substitute $u = F(y_2)$ and $du = f(y_2)dy_2$, we can evaluate the inner integral and simplify it to

$$1 - P_{\text{coll}} = M \int_0^{\infty} f(y_1)[1 - F(y_1 + c)]^{M-1}\, dy_1. \tag{7}$$

However, Eq. (7) is based on unbounded delay. If we have finite $D_{\max}$, a few modifications are needed. First we look at the integral in piece. For a minimum delay between 0 and $D_{\max} - c$ the integral is the same, but for delays between $D_{\max} - c$ and $D_{\max}$ a collision only occurs if one of the other nodes has delay less than $D_{\max}$. Then, since we only consider the collision probability for the case there is at least one node with a timer below $D_{\max}$, we normalize it by $1 - [1 - F(D_{\max})]^M$. Therefore, the probability of collision is given as

$$
\begin{aligned}
1 - P_{\text{coll}} = M \Bigg(& \int_0^{D_{\max}-c} \frac{f(x)[1-F(x+c)]^{M-1}}{1-[1-F(D_{\max})]^M} dx + \\
& \int_{D_{\max}-c}^{D_{\max}} \frac{f(x)[1-F(D_{\max})]^{M-1}}{1-[1-F(D_{\max})]^M}\, dx \Bigg) \\
= M \Bigg(& \int_c^{D_{\max}} \frac{f(x-c)[1-F(x)]^{M-1}}{1-[1-F(D_{\max})]^M} dx + \\
& \frac{[1-F(D_{\max})]^{M-1}[F(D_{\max})-F(D_{\max}-c)]}{1-[1-F(D_{\max})]^M} \Bigg).
\end{aligned} \tag{8}
$$

This can be simplified further to:

$$
\begin{aligned}
1 - P_{\text{coll}} = & \frac{M[1-F(D_{\max})]^{M-1}[F(D_{\max})-F(D_{\max}-c)]}{1-[1-F(D_{\max})]^M} \\
& + \frac{M\int_0^{D_{\max}-c} f(x)[1-F(x+c)]^{M-1}\, dx}{1-[1-F(D_{\max})]^M}.
\end{aligned}
$$

Similar to the average delay case, we can determine the relationship between $D_{\max}$ and collision probability. The derivative of the collision probability with respect to $D_{\max}$ is

$$
\begin{aligned}
\frac{\partial(1-P_{\text{coll}})}{\partial D_{\max}} = & \big(-M(M-1)f(D_{\max})[1-F(D_{\max})]^{M-2}[F(D_{\max})-F(D_{\max}-c)] \\
& + M[1-F(D_{\max})]^{M-1}[f(D_{\max})-f(D_{\max}-c)] \\
& + Mf(D_{\max}-c)[1-F(D_{\max})]^{M-1} \\
& -(1-P_{\text{coll}})[1-F(D_{\max})]^{M-1}f(D_{\max})\big) / (1-[1-F(D_{\max})]^M).
\end{aligned}
$$

This can be simplified further:

$$
\begin{aligned}
\frac{\partial(1-P_{\text{coll}})}{\partial D_{\max}} = & \frac{Mf(D_{\max})[1-F(D_{\max})]^{M-1}}{1-[1-F(D_{\max})]^M} \\
& \cdot \left[1 - \frac{1-P_{\text{coll}}}{M} - \frac{(M-1)[F(D_{\max})-F(D_{\max}-c)]}{1-F(D_{\max})}\right].
\end{aligned}
$$

When $c \ll D_{\max}$, this derivative is greater than zero. That means as $D_{\max}$ increases, P_{coll} decreases. Similarly, it can be verified that when c increases, P_{coll} also increases simply because a wider window allows more nodes to have a chance to transmit.

Suppose that we choose $D_{\max}$ such that the timer function $F(D_{\max}) \lessapprox 1$. When c is small, $f(x) \approx f(x+c)$. The probability of collision in Eq. (7) is approximated by

$$P_{\text{coll}} \approx 1 - [1 - F(c)]^M. \tag{9}$$

Eq. (9) shows that the probability of collision depends on parameters c, M, and $F(c)$ which indirectly relate to $D_{\max}$. When c increases, P_{coll} also increases, because nodes have more chance to transmit through window c and thus higher probability to cause collision. When M increases, P_{coll} also increases since more nodes are involved.

2.2.3 Timer Distribution Minimizing Collision Probability

While finding a timer distribution offering minimum delay (a delta function at 0) is relatively simple, finding a timer distribution that minimizes the collision probability is nontrivial.

A MAC technique is described in [23] for networks with event-driven traffic. Event driven traffic exists when several nodes produce data to transmit simultaneously in response to external stimuli affecting them all. Using this MAC technique, nodes select a slot to contend for according to some probability distribution p. The node transmitting in the earliest slot is allowed to transmit during this period. They derive the optimal distribution p as a function of the number of slots K and the number of nodes N. Although there is no analytical solution it was shown that p_r, the probability of selecting slot r, could be found according to this recursion:

$$p_r = \frac{1 - f_{K-r}(N)}{N - f_{K-r}(N)}(1 - \sum_{1}^{r-1} p_i) \tag{10}$$

for $r \in [1, K-1]$ and $f_r(N)$ is defined recursively as

$$f_r(N) = \left(\frac{N-1}{N - f_{r-1}(N)}\right)^{N-1}.$$

Recognize that this MAC problem is very similar to the problem at hand. If we ignore, for the moment, the desire to maintain some ordering between different nodes,

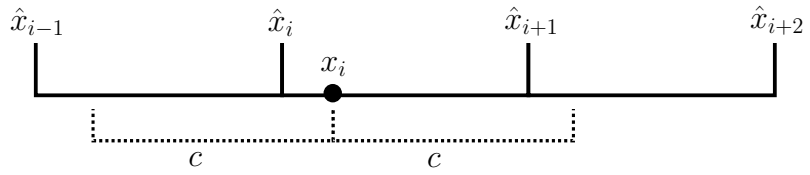


Figure 3: Illustration of effect of timer discretization

and simply focus on the relays, they simply are all responding to an event (the original transmission), the technique from [23] applies. One difference is that in the problem at hand we are not limited to a discrete choice of the set of slots, but instead to a more general selection of continuous timer values. We can show that the distribution from Eq. (10) is both applicable and optimal for this more general case.

Given a set of random continuous timer values x_i from the continuous distribution $F(x)$ and a set of discrete equivalent timer values $\hat{x_i} = c \left\lfloor \frac{x_i}{c} \right\rfloor$, the collision probability $P_{\text{coll}}(x_i) \geq P_{\text{coll}}(\hat{x_i})$. In Figure 3 the reason for this is apparent: For every x_i that would be mapped into $\hat{x}_i$, x_i collides with every other x_j that is also mapped into $\hat{x}_i$. Additionally x_i collides with some x_j that are mapped into $\hat{x}_{i-1}$ and $\hat{x}_{i+1}$. This additional collision range covers a length of c, so that for the continuous case the area around x_i which causes collisions has a width of $2c$, compared to c for the discrete case. The implications of this are:

1. For all the cases where there is a collision among $\hat{x_i}$ there will also be a collision among x_i, while the reverse is not always true;

2. Since the collision range of $\hat{x_i}$ is half the size of x_i the discrete case will have almost 50% fewer collisions;

This shows that the optimal distribution will be discrete. The optimal slot spacing will be the minimum spacing large enough to avoid inter-slot collisions. Since the optimal distribution is discrete, the optimal distribution will be the optimal discrete distribution from 10 with $K = \left\lfloor \frac{D_{\max}}{c} \right\rfloor$ slots with an interval of c between slots.

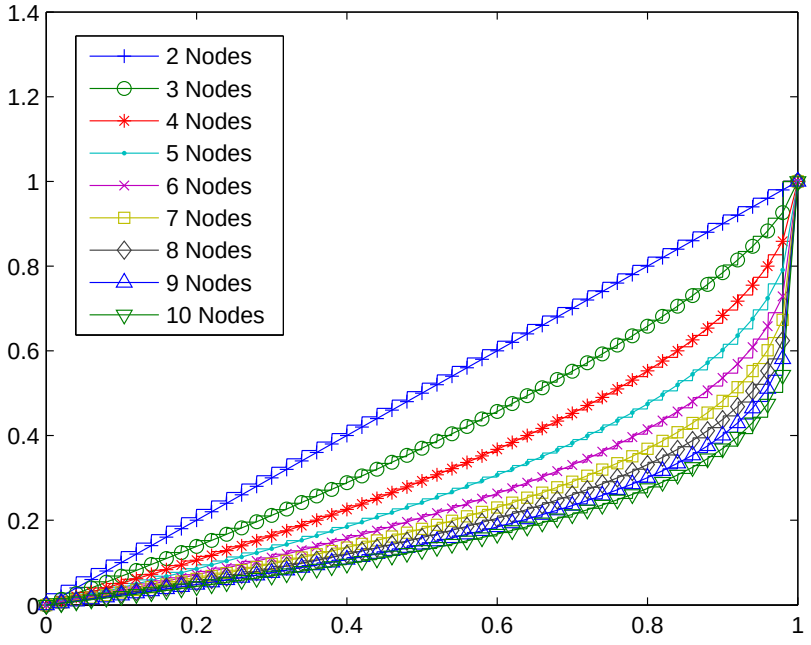


Figure 4: CDF of a collision-optimized timer distribution for c=0.02

We have verified this analysis by using a hill climbing algorithm, starting from a continuous uniform distribution, to numerically compute the optimum distribution. The results can be seen in Figure 4. In this figure the optimal cdf is plotted along with some approximate fit lines based on the beta distribution with $\alpha = 1$ and $\beta = 2/N$. Note that while the optimal distribution is not beta distributed, for this value of c the shape is very similar. In fact this shape varies very slowly with c, so excepting large values of c ($c > 0.1$), we can approximate the optimal distribution as a discretized beta distribution.

This discretized beta distribution has some interesting properties:

1. The average delay is fixed at 1/3 regardless of the value of N]:

   For this distribution:

$$D_{\max} = 1 \qquad f(x) = 2/N(1-x)^{2/N-1} \qquad F(x) = 1-(1-x)^{2/n}$$

So the average delay will be:

$$\begin{aligned}
\overline{D} &= \frac{N\int_0^{D_{\max}} xf(x)[1-F(x)]^{N-1}\,dx}{1-[1-F(D_{\max})]^N} \\
&= N\int_0^1 xf(x)[1-F(x)]^{N-1}\,dx \\
&= 2\int_0^1 x(1-x)^{2/N-1}[1-x]^{2(N-1)/N}\,dx \\
&= 2\int_0^1 x[1-x]\,dx \\
&= 1/3
\end{aligned}$$

2. The collision probability is increasing with N and relatively constant for large values of N:

$$\begin{aligned}
1-P_{\text{coll}} &= N\int_0^{1-c} f(x)[1-F(x+c)]^{N-1}\,dx \\
&= 2\int_0^{1-c}(1-x)^{2/N-1}[1-x-c]^{2-2/N}\,dx
\end{aligned}$$

$$\frac{\partial(1-P_{\text{coll}})}{\partial N} = \frac{4}{N^2}\int_0^{1-c}(1-x)^{2/N-1}(1-x-c)^{2-2/N}\left(\ln|1-x-c| - \ln|1-x|\right)$$

Note that $\ln|1-x-c| < \ln|1-x|$ so $1-P_{\text{coll}}$ decreases with increasing N, implying P_{coll} is increasing with N. Also, as N becomes large this approaches zero as $\mathrm{O}(1/N^2)$, so $1-P_{\text{coll}}$ decreases like $1/N$ and becomes nearly constant for large N.

3. The collision probability approaches $4c-3c^2+2c^2\ln c$ for large N:

$$\begin{aligned}
\lim_{N\to\infty} 1-P_{\text{coll}} &= \lim_{N\to\infty} 2\int_0^{1-c}(1-x)^{2/N-1}[1-x-c]^{2-2/N}\,dx \\
&= 2\int_0^{1-c}\frac{[1-x-c]^2}{1-x}\,dx \\
&= 1-4c+3c^2-2c^2\ln c
\end{aligned}$$

These properties allow for the framework to behave consistently regardless of the number of relays used.

This analysis only specifies the distribution the timer values should attempt to follow. Since, unlike in [23] the ordering of timer values should be based on some specific metric, we need a timer function to map the metric into a timer value. In practice it is unlikely to find a function that maps the metric into a timer with this desired distribution. For the case where CSI is the metric of choice, this will be discussed further in Section 3.2.

2.3 Implementation Issues

In actual implementations there are a few concerns with protocols based on this framework.

2.3.1 Effect of Clock Skew and Timing Offsets

This framework is dependent on the relationship between multiple timers located at discrete nodes, so it will be affected by timing offsets between nodes. These timing offsets can be approximated with the following model: $t = s_i t_0 + \delta_i$. Under this model each node's timer has some frequency offset and time offset relative to the others. The time offset may be due to propagation delay and other factors, while the frequency offset could be due to imperfections in the local oscillator.

This problem will result in two effects: a probability of reordering, and increased collision probability.

The probability of reordering for a set of nodes with timers s percent faster and offset δ from the 'true' shortest timer is given as:

$$p_{\text{reorder}} = 1 - N \int_0^{\frac{D_{\max}-\delta}{1+s}} f(x) \left[1 - F(x + sx + \delta)\right]^{N-1} \, dx$$

Skew and timing offsets will also affect the collision probability. The primary effect will be to "smooth out" the timer distribution, reducing the strict c-spacing

present in discrete distributions, causing them to suffer the increased collision probability of continuous distributions. This smoothing will have very little impact to continuous distributions since the small shifts in the distribution shape have little effect. Under the slightest suboptimal conditions, discrete distributions will behave as badly as similar continuous distributions. This means that networks implementing this framework that rely on the decreased collision probability provided by discrete distributions will need to provide extra guard intervals between slots and use clock synchronization techniques to reduce the effects of skew.

2.3.2 Hidden Nodes

Since forwarding nodes may transmit to separate directions or "hidden nodes" may exist, there is a possibility that two nodes may transmit at the same time without causing a collision at any of their destinations. In this case, the same packet may be transmitted through more than one path. Due to the nature of this protocol, the split paths will converge before or at the final destination. Nodes can either maintain a list of recent packets, or the destination can enjoy the extra packets to achieve multi-route diversity.

In realistic networks it is possible that some of the candidate next-hop nodes will be unable to hear each other's transmissions. If a node does not hear the node with the lowest timer, they both will transmit, potentially resulting in packet duplication. This is analogous to the hidden node problem discussed previously, and can be dealt with similarly. Other possible solutions include reducing the likelihood of this occurring by transmitting a low-rate flag before the message, or by sending a message to the previous hop and letting that node broadcast who the transmitter this hop should be.

2.3.3 Scalability

Scalability of the network is an important issue. If the timer function is unbounded the worst case of maximum delay will be infinite. In this case, for multi-hop networks, the average delay and maximum delay are not scalable with the number of hops and no matter what the tolerable $D_{\max}$ is, the outage probability to exceed $D_{\max}$ is always greater than zero. If the timer function is bounded so that it has some maximum value $D_{\max}$ then the maximum delay (worst case) will scale linearly with the number of hops.

Another way to reduce the average delay without incurring increased collision probability is to allow the candidate nodes some amount of extra communication overhead. These nodes could wait a small random interval after receiving the packet and then broadcast the most significant bits of their calculated timer. This would allow the candidates with timers whose most significant bits are greater than the received broadcast to cancel the timer and drop the packet. Also, after all candidates have communicated these bits, this also allows the candidates to agree on a lower bound for the minimum timer value and subtract that from their timers, resulting in a maximum delay significantly smaller than the timer maximum.

CHAPTER III

NON-COOPERATIVE ROUTING WITH COOPERATIVE DIVERSITY

Our protocol based on this framework achieves selection diversity by adopting a timer function based on the CSI, h_m. The timer function $T(h_m)$ should be a monotonically decreasing function of the channel state information. This is an effective timer function since a large value of $|h_m|$ results in a lower error probability. The channel model described in Eq. (2) shows that although h_m depends on the distance it is also random. By using CSI as a metric our protocol opportunistically utilizes links that are strong, reducing the probability of error due to noise. The CSI can be estimated by having the transmitting nodes send training symbols at the beginning of the timeslot. A 2-hop cooperative protocol similar to ours was investigated in [4], but our analysis differs since we provide a framework for an arbitrary number of hops and investigate the choice of timer functions.

Also note that as shown in Section 2.2.1 timer functions with bounded delay can be used. With this protocol bounding the timer value also serves another purpose. If a node calculates a large timer value, that implies the channel is very weak. By setting the bound appropriately a minimum channel strength cutoff can be enforced. If this is combined with a MAC-layer ARQ, when all the channels are below this cutoff, the source will automatically resend the packet.

3.1 Cooperative Diversity

This protocol achieves cooperative diversity. Consider single hop transmission with M candidate nodes under Rayleigh fading. If s is the transmitted packet, w_m is

complex Gaussian noise, and y_m is the received signal at the mth node then,

$$y_m = h_m s + w_m, \forall m \in [1, M].$$

Then since h_m is the channel state information and the timer function ensures the strongest channel is used, the model becomes:

$$\tilde{y} = \tilde{h} s + w,$$

where $\tilde{h}$ corresponds to the one with maximum $|h_m|$.

Since the h_m's are independent and complex Gaussian distributed, the error probability for this hop is [16]:

$$P_e^{(s)} \leq (c \cdot SNR)^{-M},$$

where c is a constant and SNR is the signal-to-noise ratio at the transmitter ($E[s^2]/\sigma_w^2$). The diversity order is defined as the negative slope of the log error probability at high SNR, which in this case is M. This shows that our protocol achieves the maximum diversity order possible in this system, M. This diversity is in fact quite similar to the one collected by selective combining techniques for multi-antenna or multi-path systems [4]. Compared with maximum ratio combining (MRC) and equal gain combining (EGC), the major advantage of our protocol is that nodes do not need to exchange channel state information, resulting in improved bandwidth and energy efficiencies and reduced communication and computational overheads.

3.2 Choosing the Timer Function

As shown in Section 2.2 the timer distribution has an enormous impact on the performance of protocols under this framework.

In Table 1 a variety of possible timer functions are considered and their distributions for purely Rayleigh faded channels are given. If you recall in Section 2.2.3 we characterized the distribution which optimizes collision probability. In Figure 4 we have shown that this distribution's shape was approximated by the beta distribution

Table 1: Timer functions and timer duration distributions for Rayleigh fading

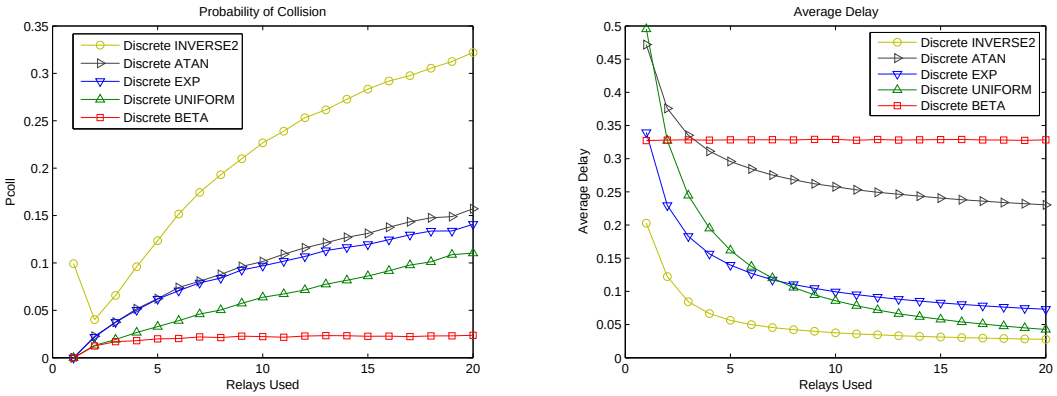
Name	Function of h	PDF	CDF
inverse2	$\frac{1}{\lvert h\rvert^2 D_{max}}$	$f(x) = D_{max}\frac{1}{2x^2 D_{max}^2} e^{-\frac{1}{2xD_{max}}}$	$F(x) = e^{-\frac{1}{2xD_{max}}}$
atan	$\frac{2}{\pi}\arctan(\frac{1}{\lvert h\rvert})$	$f(x) = \frac{\pi\cos\left(\frac{x\pi}{2}\right)}{2\sin^3\left(\frac{x\pi}{2}\right)} e^{\frac{-1}{2\tan^2\left(\frac{x\pi}{2}\right)}}$	$F(x) = e^{-\frac{1}{2\tan^2\left(\frac{x\pi}{2}\right)}}$
exp	$e^{-\lvert h\rvert}$	$f(x) = -\frac{\ln x}{x} e^{\frac{-(\ln x)^2}{2}}$	$F(x) = e^{-\frac{(\ln x)^2}{2}}$
uniform	$e^{-\frac{\lvert h\rvert^2}{2}}$	$f(x) = 1$	$F(x) = x$
beta	$1 - \left[1 - e^{\frac{-\lvert h\rvert^2}{\sigma^2}}\right]^{\frac{N}{2}}$	$f(x) = \frac{2}{N}(1-x)^{\frac{2}{N}-1}$	$F(x) = 1 - (1-x)^{\frac{2}{N}}$

with parameters $a = 1$ and $b = 2/N$. The last entry in Table 1 shows the timer function which provides this beta distribution for a Rayleigh distributed channel. Note that for best performance these timer functions should be discretized using a method such as $\hat{T}(h_m) = c\left\lfloor \frac{T(h_m)}{c} \right\rfloor$.

3.2.1 Normalization

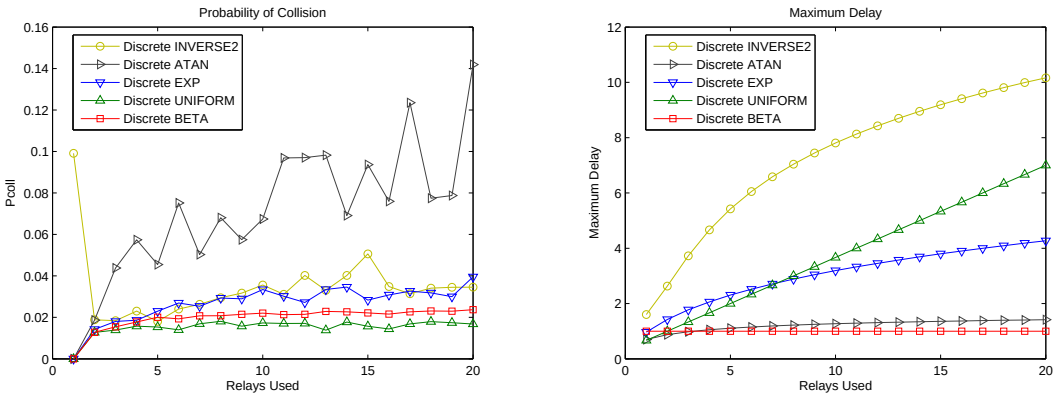
In order to compare these timer functions, they must be normalized. There are two potential methods of normalizing different distributions for comparison. They could be normalized in terms of maximum allowable delay, or in terms of average delay. The timer functions listed in Table 1 have been normalized so that the maximum delays are equal. Note that function INVERSE2, which was also accepted in [4], has no bound under a Rayleigh channel. However, we also set up a bound such that if the channel is complex Gaussian with zero mean and unit variance, the probability that a timer would set outside the bound is less than 0.1.

In Figure 5 and Figure 6 the delay and collision probability for the functions in Table 1 are compared with the functions normalized by holding maximum delay constant (Figure 5) or by making average delay constant (Figure 6). From these figures the plots do indeed show distinctive differences. The only collision probabilities that are mostly unchanged are the ATAN and BETA timer functions, due to their more relatively constant average delay. UNIFORM and INVERSE2, due to their



(a) Collision Probability (b) Average Delay

Figure 5: Performance Comparison for Max-Delay normalized functions



(a) Collision Probability (b) Maximum Delay

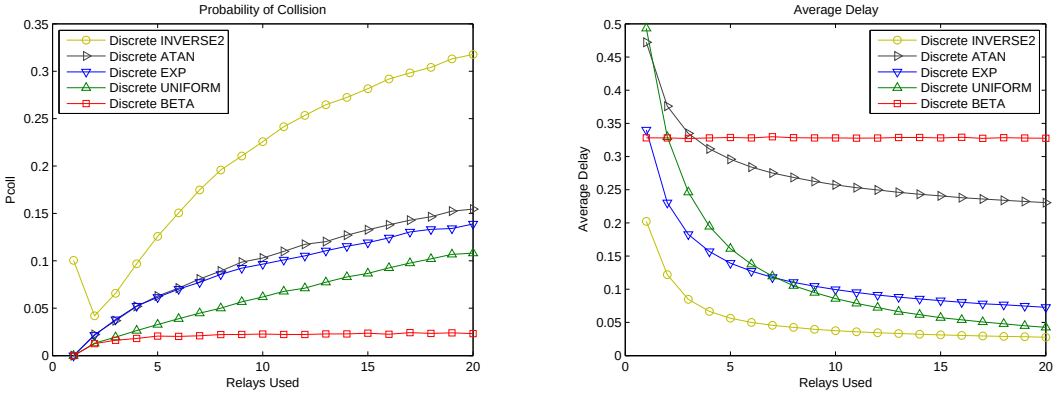
Figure 6: Performance Comparison for Average Delay normalized functions

constantly decreasing average delay achieve the most profound changes.

3.2.2 Channel Model

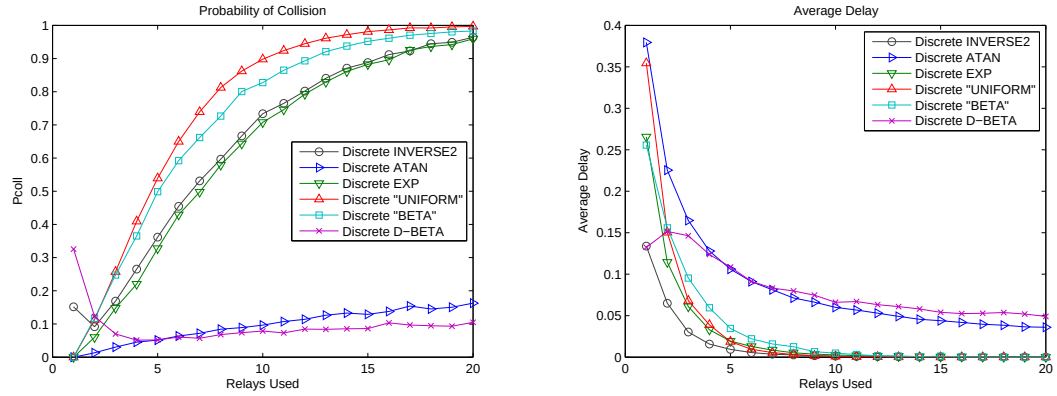
Another aspect of comparing timer functions is the effect of the channel model. The channel model has a significant impact on the distribution of the CSI (h) which in turn determines the distribution of the timer ($T(h)$). The probability of collision and average delay under a Rayleigh channel model is shown in Figure 7. These results under a more complex channel model such as that of Eq. (2) are shown in Figure 8.

Under the radically altered channel state distribution of Figure 8, the 'BETA' and 'UNIFORM' timer functions lose their special qualities, and no longer cause the



(a) Collision Probability (b) Average Delay

Figure 7: Performance for timer functions under a Rayleigh channel



(a) Collision Probability (b) Average Delay

Figure 8: Performance for timer functions under a realistic channel

timer distributions to be beta or uniformly distributed. The difference is significant enough to transform the 'BETA' timer function, nearly optimal in a pure Rayleigh channel, to the second worst when distance attenuation and shadowing effects are considered. In fact the Rayleigh component of the channel plays a less significant part of the eventual timer distribution than the distance component. The ATAN function performs better in this case because it provides better collision performance for a channel function based purely on attenuation.

If we consider a channel distributed only based on distance attenuation ($h = d^{\frac{-\alpha}{2}}$), a beta distributed channel can be created using the timer function

$$T_{\text{D-BETA}}(h) = 1 - \left(1 - \frac{|h|^{\frac{-4}{\alpha}}}{R_{\max}^2}\right)^{\frac{N}{2}},$$

where $R_{\max}$ is the maximum distance a node can be and still be considered a neighbor. This function requires $|h| \geq R_{\max}^{\frac{-\alpha}{2}}$. Since, due to fading and shadowing, this is not generally the case for the more realistic channel model, this function cannot be used unless the timer function is adjusted to eliminate this constraint. The function cannot simply be bounded as before because there is a significant probability ($> 25\%$) that the channel will exceed this range. Instead we use $T(R_{\max}^{\frac{-\alpha}{2}}) = D_{\max}$ as the timer value for the cases where the timer would otherwise be outside this range. The performance of this D-BETA function is also shown on Figure 8. It has slightly better performance on the realistic channel than the ATAN function, but due to its dependence on $R_{\max}$ and α it is not nearly as practical as the ATAN function.

3.3 Performance

We will evaluate the protocol performance through simulation.

3.3.1 Two-Hop Exapmple

First we will evaluate the performance of the protocol over a two-hop transmission. The network consists of a single node transmitting to a number of candidate relays.

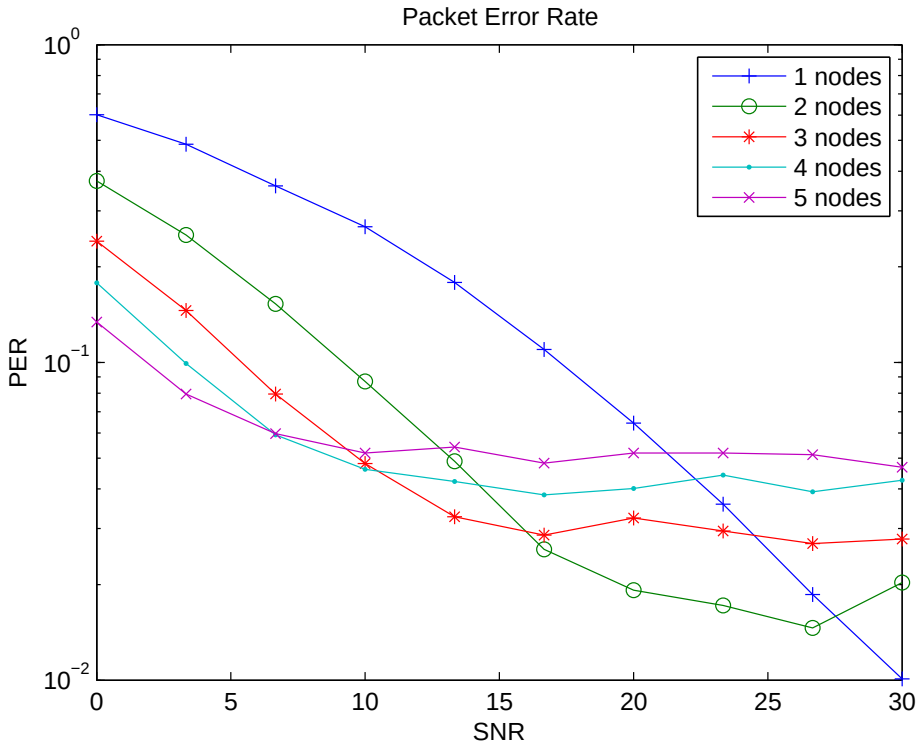


Figure 9: PER vs. SNR ($c = 0.002$) for Two-Hop Transmissions

The number of relays and the SNR are varied. The necessary spacing c is set to 0.01, and a collision is said to occur if the there exists a timer less than c greater than the minimum timer: $T(h_2) - T(h_1) < c$. Since a discretized timer is used, this implies that a collision will occur when the minimum timer value is shared by two nodes. The channel model in Eq. (2) is used with $\alpha = 2$, $\mu_\zeta = 0$ and $\sigma_\zeta = 1$. Since this is the more complex fading environment, the ATAN timer function is used. The relays are assumed to be uniformly distributed so distances are selected randomly from a $F(x) = x^2$ distribution with a maximum distance of 1.

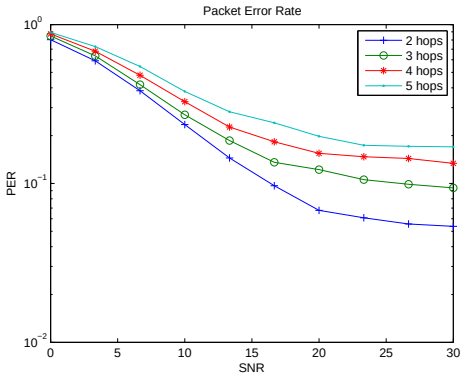
The simulation results are presented in Figure 9 where "m nodes" means there are m relays between source and destination. Each different curve shows the packet-error rate (PER) vs. SNR by applying our protocol with a different number of relays. The "1 node" curve will be identical to that of any protocol that unicasts the packet to a selected next-hop node and thus does not exploit diversity. At low SNR, PER is

dominated by the errors caused by noise and fading effects. In this case, when we increase the number of involved nodes (M), the PER will be smaller thanks to the diversity. When SNR is high, PER is mainly due to the probability of collision P_{coll}. Since P_{coll} does not depend on SNR, it shows up as an error-floor in the figure. As we have shown in Sec. 2.2, as the number of nodes increases, P_{coll} also increases. This is consistent with the observation from the figure. From this figure we can see that with a reasonable SNR range $[0, 30]$dB, our protocol outperforms the unicast protocols over wireless fading environment.

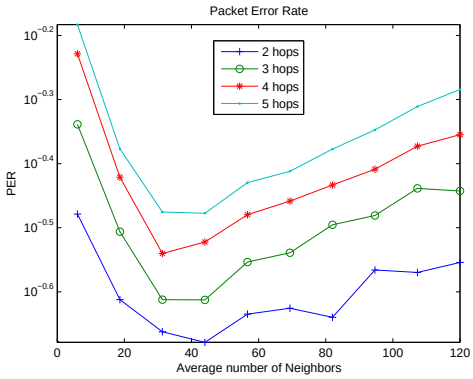
3.3.2 Multiple Hops

Next we consider the end-to-end performance of this protocol on a more realistic network. For this simulation nodes are uniformly randomly placed in a 3×10 rectangle and a random source-destination pair is chosen. The link between two nodes follows the link model in Eq. (2). The number of nodes is varied such that the average number of nodes per unit disk ranged from 6 to 120, corresponding to an average of 1 to 20 relays within the 60-degree spread similar to the single-hop case. Note that since this simulates a complete network, the actual number of involved nodes used each hop is random, but with a mean determined by the density. Also, since the source and destination are random, the number of hops for the entire trip also varies. For this simulation we adopt the ATAN timer function. The channel is identical to the single hop case in Section 3.3.1. If there is no greedy path from source to destination, the network realization is discarded. Collisions are assumed to always occur when two nodes transmit within an interval c of each other, and are handled as in the same way as the one-hop case.

The results are shown in Figure 10 for $c = 0.002$. In Figure 10(a), the PER vs. SNR curves for routes with 2 to 5 hops are plotted. For this figure, the average number of nodes per unit disk is set to $\rho = 30$ and the ATAN timer function is used.



(a) PER vs. SNR (30 nodes per unit disk)



(b) PER vs. Number of Nodes (10 dB SNR)

Figure 10: Multi-hop routing simulation: ATAN timers, $c = 0.002$

As expected routes with more hops have higher error rates. In Figure 10(b) the PER is plotted against the average number of relays available.

CHAPTER IV

OTHER VARIATIONS BASED ON THIS FRAMEWORK

4.1 Multi-Antenna Networks

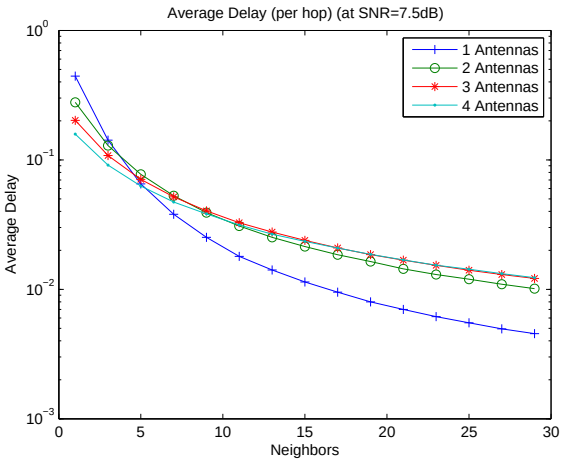
In modern wireless systems the use of spatial diversity derived from collated multi-antenna systems is becoming increasingly common place (e.g., IEEE802.11n, IEEE802.16e). We show our scheme can be readily adapted to a MIMO architecture and demonstrate the results for a V-BLAST (Vertical Bell Labs Layered Space-Time) system.

The block system model is

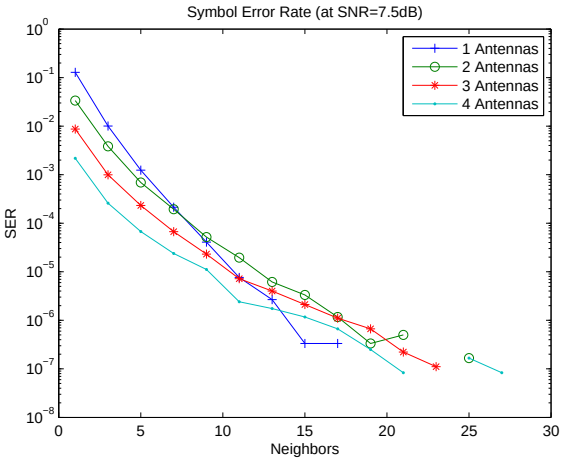
$$\mathbf{y}_j = \mathbf{H}_{ij}\mathbf{s}_i + \mathbf{W}_j, \tag{11}$$

where $\mathbf{H}_{ij}$ is an $M_j \times M_i$ channel state matrix from node-i (with M_i antennas) to node-j (with M_j antennas). Note that since different nodes may equip different numbers of antennas, we need a metric to compare different sizes of matrices. An appropriate link metric to base the timer value on is the value of the smallest eigenvalue $\lambda_{\min}$ of $\mathbf{H}_{ij}^H\mathbf{H}_{ij}$. Maximizing this value is equivalent to minimizing the BER of the weakest subchannel in V-BLAST systems [8]. Since the weakest link has the greatest contribution to the BER in a V-BLAST MIMO system, this will also minimize the effective BER of the entire system [8]. Other metrics such as maximum orthogonal degree could also be used. Using $\lambda_{\min}$ as a metric has the benefit of allowing comparison between nodes in heterogeneous networks that have different numbers of antennas.

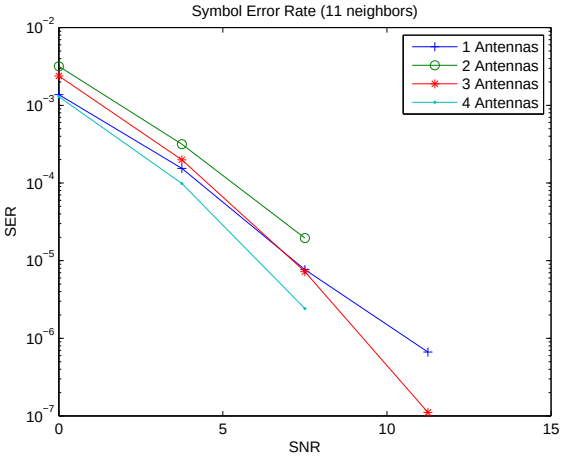
Using the $\lambda_{\min}$ metric we have simulated the performance of this protocol. In this simulation we simulated a single hop VBLAST system and varied the number of relays, number of antennas per relay, and the SNR. SNR was defined as the symbol energy over noise power. The channel was assumed to follow the channel model in Eq. (2). QPSK (Quadrature Phase Shift Keying) signalling with spherical decoding



(a) Delay



(b) Symbol Error Rate vs Number of Relays



(c) Symbol Error Rate vs SNR

Figure 11: MIMO Performance for two-hop transmissions

was used. The ATAN timer function was used to convert $\lambda_{\min}$ into a timer value. The error rate and delay were measured, assuming no collision probability. To facilitate comparison, the delay was measured as the delay per symbol assuming each of the transmitter's antenna transmitted 100 symbols per packet.

In Figure 11(a), the average delay is shown. Note that since this is the per-symbol delay, if delay were independent of the number of antennas the average delay in a system would be given by $\overline{D_{\text{M antennas}}} = \overline{D_{\text{1 antenna}}}/M$. When the network density is low, as the number of antennas increases, the average delay is smaller because multiple antennas enhance the transmission rate. However when the network density is high, this trend does not hold. This is because taking the minimum of the λ values results in the distribution of $\lambda_{\min}$ to be biased toward lower values, which result in larger timer values. As the number of antennas increase, this biasing increases, resulting in increased timer values. As the number of neighbors increases, this reduces the rate the delay decreases, causing the results in the figure.

In Figure 11(b), the symbol error rate (SER) is plotted against the number of relays used. If we compare the slope of the plots in Figure 11(b), it becomes apparent that systems with fewer antennas are able to take better advantage of the increased selective diversity. This occurs for similar reason as the delay case. Since the distribution of $\lambda_{\min}$ has been biased, increasing the number of nodes has a smaller effect because of the bias. Note that this does not affect the diversity of the system: systems with more antennas have better performance at high SNR, as shown in Figure 11(c).

This was also simulated in a multi-hop network. The parameters were similar to the single hop case. In Figure 12, the multihop performance of the system with an average of 5 relays is shown for a varying number of antennas. Here we observe that multihop routing benefits from multiple antennas.

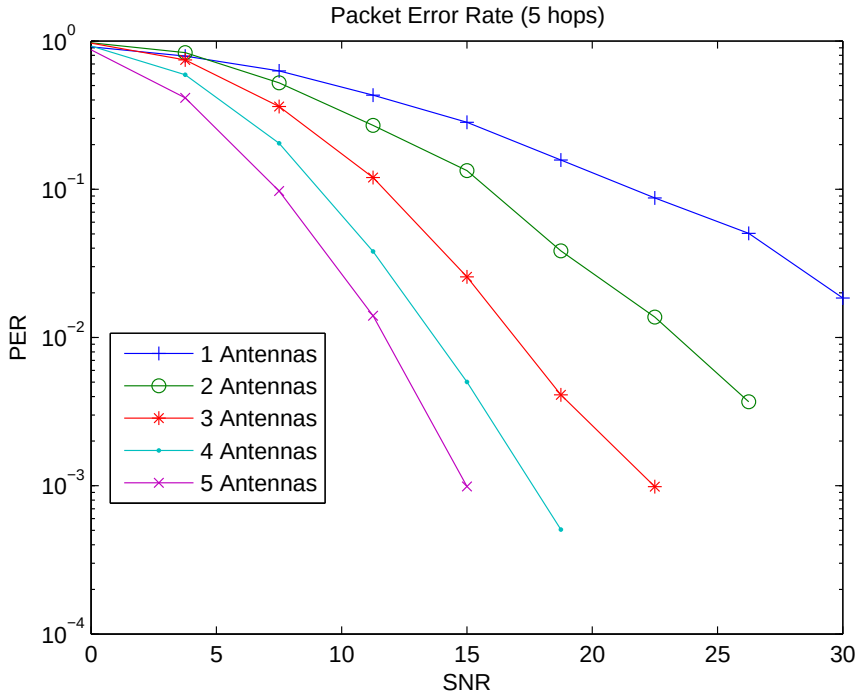


Figure 12: Performance of multihop system with varying numbers of antennas

4.2 Including Distance

It is apparent that any routing protocol that does not consider the progress toward the destination will be suboptimal. In [20] a modification to the greedy routing protocol to include packet reception rate was proposed. Nodes would use a greedy algorithm to forward the packet to the node with the highest score. This score was calculated as the product of the packet reception rate (PRR) and the marginal distance to the destination. For example, if node N_0 at position P_0 wishes to forward a packet to node N_D at position P_D it would calculate the marginal distance $d_i = |P_D - P_i| - |P_D - P_0|$ for each of the neighbors N_i. Nodes maintain time average statistics on the PRR to each neighbor and then use $Score_i = PRR_i \times d_i$ to calculate the score. This scheme was shown to be optimal for chain topologies with infinite ARQ retries, but as the channel model was not considered, and since the packet reception rate was calculated based on a time average it is incapable of dealing with fading. We denote this scheme

as the "PRRd" protocol.

We can adapt the protocol mentioned in [20] into the framework from Chapter 2. The timer could use the score as the metric. Since the timer is calculated at each relay, nodes can estimate the PRR based on the current channel state information, allowing it to take advantage of fading. For example in systems with binary phase shift keying (BPSK) modulation the PRR can be calculated as $PRR(h) = \left(1 - Q(\frac{|h|}{\sigma_w})\right)^{\text{Packet Length}}$. From this the score can be calculated and then a timer function such as the ATAN function can be applied to this metric to produce the final timer value. Since this protocol uses the instantaneous PRR we denote this protocol as the "iPRRd" protocol to differentiate from the original.

We simulated this protocol and compared it both to the original PRRd protocol as well as the protocol we described in Chapter 3 (denoted as 'NCRP') on a multi-hop network. The simulation was setup similar to that described in Section 3.3, except the collision probability was not considered for this simulation. The results appear in Figure 13. In this figure it is apparent that the iPRRd protocol has the best performance as it takes advantage of both the marginal progress toward the destination and the instantaneous PRR. Interestingly the NCRP protocol still performs well despite not considering the marginal progress toward the destination. As the SNR (symbol energy vs. noise power) becomes higher, NCRP's performance becomes less impressive. Since it does not consider the progress towards the destination it tends to take more hops than necessary at high SNR, resulting in degraded performance.

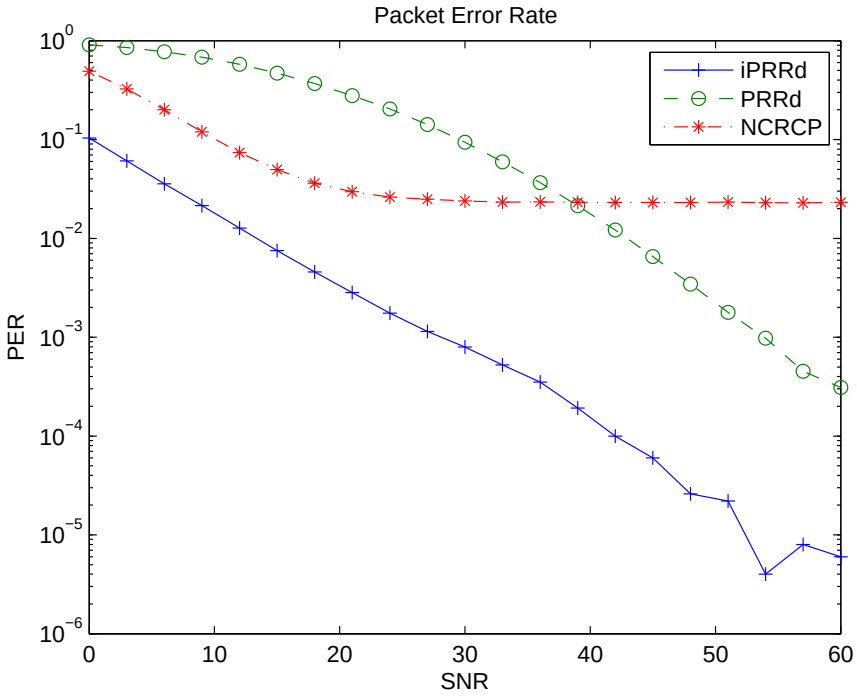


Figure 13: PER vs SNR for iPRRd, PRRd and NCRP protocols

CHAPTER V

CONCLUSION

In this thesis we have described the problem of routing in wireless sensor and ad hoc networks. We have discussed current techniques and their limitations. We developed and described a simple framework enabling low-overhead distributed routing in wireless networks. The performance characteristics of this framework such as the delay and collision probability were then theoretically analyzed. A protocol built based on this framework which achieves cooperative diversity through the routing decision was described and its parameters explored. This protocol was simulated on both single-hop and multi-hop networks. A variation on this protocol for networks of nodes with multiple antennas was discussed and evaluated through simulation. An adaptation of this protocol to include the marginal progress toward the destination was discussed and compared against both to the cooperative diversity protocol and the protocol proposed in [20].

In the future, we will continue our investigations in both wireless sensor networks and ad hoc networks through investigating the application of this framework to network efficient protocol design. We will also focus further on network synchronization issues such as those only tangentially addressed in this work.

REFERENCES

[1] Aggelou, G., *Mobile Ad Hoc Networks*. Mc Graw Hill, first ed., 2005.

[2] Akyildiz, I. F., Weilian, S., Sankarasubramaniam, Y., and Cayirci, R., "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, pp. 102–114, Aug. 2002.

[3] Biswas, S. and Morris, R., "ExOR: Opportunistic multi-hop routing for wireless networks," in *Proc. ACM SIGCOMM*, pp. 133–144, ACM, Aug. 2005.

[4] Bletsas, A., Khisti, A., Reed, D. P., and Lippman, A., "A simple cooperative diversity method based on network path selection," in *IEEE Journal on Selected Areas in Comm.*, vol. 24, pp. 659–672, Mar. 2006.

[5] Finn, G. G., "Routing and addressing problems in large metropolitan-scale internetworks." Technical Report ISI/RR-87-180, USC/Information Sciences Institute, Mar. 1987.

[6] Gamal, H. E. and Aktas, D., "Distributed space-time filtering for cooperative wireless networks," in *GLOBECOM*, 2003.

[7] Haas, Z., "A new routing protocol for the reconfigurable wireless networks," in *Proc. IEEE International Conference on Universal Personal Communications*, vol. 2, pp. 562–566, Oct. 1997.

[8] Heath, Jr., R. W., Sandhu, S., and Paulraj, A., "Antenna selection for spatial multiplexing systems with linear receivers," in *IEEE Communications Letters*, vol. 5, pp. 142–144, Apr. 2001.

[9] Johnson, D. and Maltz, D., "Dynamic source routing in ad-hoc wireless networks," in *Proceedings of SIGCOMM '96*, Aug. 1996.

[10] Karp and Kung, H. T., "GPSR: greedy perimeter stateless routing for wireless networks," in *Proc. IEEE/ACM MobiCom*, pp. 243–254, Aug. 2000.

[11] Kranakis, Singh, H., and Urrutia, J., "Compass routing on geometric networks," in *Proc. 11 th Canadian Conference on Computational Geometry*, pp. 51–54, 1999.

[12] Kuhn, Wattenhofer, R., and Zollinger, A., "Asymptotically optimal geometric mobile ad-hoc routing," in *Proc. 6 th Int. Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications (Dial-M)*, pp. 24–33, 2002.

[13] Laneman, J. N., Tse, D. N. C., and Wornell, G. W., "Cooperative diversity in wireless networks: Efficient protocols and outage behavior," in *IEEE Trans. Information Theory*, pp. 3062–3080, Dec. 2004.

[14] Larsson, P., "Selection diversity forwarding in a multihop packet radio network with fading channel and capture," in *Proc. IEEE/ACM Mobicom*, vol. 5, pp. 47–54, Aug. 2001.

[15] Mergen, B. S. and Scaglione, A., "Randomized space-time coding for distributed cooperative communication," in *Proc. International Conference on Communications*, 2006.

[16] Papoulis, A. and Pillai, U. S., *Probability, Random Variables and Stochastic Processes*. McGraw-Hill, 4th ed., 2001.

[17] Perkins, C. E. and Royer, E. M., "Ad-hoc on-demand distance vector routing," in *Proc. IEEE WMCSA'99*, vol. 3, pp. 90–100, Aug. 1999.

[18] Perkins, E. and Bhagwat, P., "Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers," in *Proc. ACM SIGCOMM*, pp. 234–244, Aug. 1993.

[19] Rappaport, T. S., *Wireless Communications: Principles and Practice*. Prentice Hall, second ed., Dec. 2001.

[20] Seada, K., Zuniga, M., Helmy, A., and Krishnamachari, B., "Energy-efficient forwarding strategies for geographic routing in lossy wireless sensor networks," in *Proc. ACM SenSys*, Nov. 2004.

[21] Shah, R., Wietholter, S., Wolisz, A., and Rabaey, J., "When does opportunistic routing make sense," in *Proc. IEEE PerCom*, pp. 350–356, Mar. 2005.

[22] Shah, R. C., Bonivento, A., Petrovic, D., E. Lin, J. v. G., and Rabaey, J., "Joint optimization of a protocol stack for sensor networks," in *IEEE MILCOM*, vol. 1, pp. 480–486, Nov. 2004.

[23] Tay, Y. C., Jamieson, K., and Balakrishnan, H., "Collision-minimizing CSMA and its applications to wireless sensor networks," in *IEEE Journal on selected areas in Comm.*, vol. 22, pp. 1048–1057, Aug. 2004.

[24] Wei, S., Goeckel, D. L., and Valenti, M. C., "Asynchronous cooperative diversity," in *IEEE Trans. On Wireless Comm.*, vol. 5, June 2006.

[25] Wiser, M., "Hot topis– ubiquitous computing," *Computer*, vol. 26, pp. 71–72, Oct. 1993.

[26] Zuniga, M. and Krishnamachari, B., "Analyzing the transitional region in low power wireless links," in *IEEE Secon*, 2004.