Physical layer impact on the design and performance of routing and broadcasting protocols in ad hoc and sensor networks

I. Stojmenovic, A. Nayak, J. Kuruvila, F. Ovalle-Martinez, E. Villanueva-Pena

School of Information Technology & Engineering, University of Ottawa, Ottawa, Ontario K1N 6N5, Canada.
Email:{ivan, anayak, kuruvila, fovalle, pvillanu}@site.uottawa.ca

Abstract

Existing routing and broadcasting protocols for ad hoc networks assume an ideal physical layer model, where two nodes communicate if and only if they are at distance at most R, where R is the transmission radius. This article surveys our efforts to consider a more realistic physical layer model, and its impact on routing and broadcasting. We apply the log normal shadow fading model to represent a realistic physical layer to derive (accurately and approximately) the probability $p(x)$ for receiving a packet successfully as a function of distance x between two nodes. We define the transmission radius R as the distance at which $p(R) = 0.5$. We consider routing algorithms with and without acknowledgements, with messages consisting of one or more packets. For single packet routing with hop by hop acknowledgements, we propose a MAC layer protocol where receiver node acknowledges packet to sender node u times, where $u * p(x) \approx 1$. The expected hop count (EHC) between two nodes under this protocol is $\frac{1}{p(x)^2} + \frac{1}{p(x)}$ (for $u = 1$). We show that forwarding to neighbor closest to destination is suboptimal, and that optimal forwarding distance is one that minimizes EHC per progress made, that is, minimizes $h(x) = (\frac{1}{p(x)^2} + \frac{1}{p(x)})\frac{1}{x}$. Depending on the power attenuation factor β (between 2 and 6), the optimal forwarding radius is between $0.7R$ and $0.8R$. The function $h(x)$ can be used to derive hop count optimal route discovery based routing. During route discovery, each node, receiving message from a neighbor at distance x (distance can be estimated from received signal strength), will set timeout proportional to $h(x)$, before retransmitting. The destination will respond to the first message received. Neighbor discovery is not straightforward since hello messages are not received by all neighbors. We proposed several localized routing schemes for the case when position of destination is known, optimizing expected hop count (for hop by hop acknowledgement), or maximizing the probability of delivery (when no acknowledgements are sent). We then considered localized power aware routing schemes under realistic physical layer, when nodes can adjust their transmission powers. Finally, we discuss broadcasting in ad hoc network with realistic physical layer, and propose new concept of dominating sets to be used in broadcasting process.

Preprint submitted to Elsevier Science *31 May 2004*

Key words: Ad hoc wireless networks, sensor networks, routing, broadcasting, physical layer.
PACS:

1 Introduction

Wireless as hoc networks emerged recently as a 'hot' research topic because of their potential applications in various situations such as battlefield, emergency relief, and conference environments. Sensor networks are currently one of prime research topics, and are being designed for monitoring environment for chemicals, temperature, movement, fire and other events. Ad hoc and sensor networks consist of hosts that communicate without a fixed infrastructure. Communications take place over a wireless channel, where each host has the ability to communicate with others in the neighborhood, determined by the transmission range. Since there is no infrastructure, every host has to determine its environment when the network is formed.

We assume that each node has a low-power Global Position System (GPS) receiver, which provides the position information of the node itself. If GPS is not available, the distance between neighboring nodes can be estimated on the basis of incoming signal strengths or time delays. Relative co-ordinates of neighboring nodes can be obtained by exchanging such information between neighbors [3].

In the routing task, a message is to be sent from a source node to the destination node. The nodes in the network may be static or mobile. The task of finding and maintaining routes in ad hoc networks is nontrivial since host mobility can result in unpredictable topology changes. Location updates schemes for efficient routing are reviewed in [4]. In a broadcasting task, a node wishes to send the same message to all the other nodes in the network.

We consider two routing scenarios. In one scenario, sender node is not aware of destination location, and initiates route discovery, which is a broadcasting task. In other scenario, source node is aware of geographic position of destination. Many routing algorithms proposed are non-local and require the complete knowledge and maintenance of the network topology. Recently, many *localized* routing algorithms have been proposed (a brief survey of them is given in [5]), where nodes do not require the complete network topological information to perform the routing task. More precisely, nodes only require the position of itself and its 1-hop neighbors (in some cases also position of its 2-hop neighbors), and position of destination. Consequently, neighboring nodes are aware of distances between them.

There are two assumptions about transmission power: fixed and adjustable. In the first case, all nodes have a fixed and equal transmission radius R. Existing network layer protocols (with few exceptions) for ad hoc networks assume an ideal physical layer model, where two nodes communicate if and only if the distance between them is at most R. In this model, known as the unit graph model, two nodes within transmission radius can exchange correctly bits, packets and messages (we assume that messages are composed of few fixed length packets, and packets are composed of fixed length bit-strings). In the unit graph model there exists therefore the unique transmission radius at all layers of communication. We apply, however, log normal shadow fading model to represent a realistic physical layer. By applying a realistic physical layer, the notion of transmission radius needs to be carefully defined and properly used in algorithms. The packet reception probability $p(x)$ depends on the probability of receiving a bit successfully $b(x)$ and the length of the the packet. There are different ways of determining R. Our approach is to divide message into fixed size packets, and transmit each packet individually. In this case, R can be determined so that packet error rate at distance R is 0.5. It obviously depends on packet length. The error rate for acknowledgements is then also 0.5 at distance R, since acknowledgements are assumed to be single packets with equal packet length, therefore the same probability for their reception is used. There are variety of ways to define medium access layer for acknowledging the packets. This interpretation for R appears to be the most convenient for deriving protocols and various acknowledgement schemes. When nodes can adjust their transmission radii, the goal is to minimize the total power for a route, with all retransmissions counted.

In this survey article, we consider routing with and without acknowledgements. In the HHR (Hop-by-hop retransmissions) model, a packet is retransmitted between two nodes until it is received and acknowledged correctly. We consider the *separate HHR* variant, where acknowledgements to the previous node and forwarding message to the next node are always done by separate messages. The variant where retransmissions to the next node can serve as acknowledgement to the previous node is left for future research.

For algorithms without acknowledgements, the probability of successful delivery of the packet (which is the product of all the probabilities of successful delivery of hops along the found route) from source to destination is used as a performance measure for routing algorithms.

2 Related Work

There exists a vast amount of literature devoted to position based routing in ad hoc networks. Finn [6] proposed localized greedy scheme, where node, currently holding the message, will forward it to the neighbor that is closest to destination. Only nodes closer to destination than the current node are considered. Another milestone achievement is localized greedy-face-greedy (GFG) algorithm, proposed in [7], which guarantees delivery under ideal MAC layer and correct position information. It applies greedy algorithm whenever possible, and restores to face routing in recovery mode. Face routing uses a planar graph to route from face to face between source and destination nodes. A survey of position based routing schemes is given in [5]. Power and cost aware routing was also intensively studied. The first localized algorithms which considered power and cost when making routing decisions, and having reasonable performance when compared to globalized schemes, were proposed in [12].

Our work has been inspired by recent observations made in [8–11]. Qin and Kunz [8] concentrate on the impact of a realistic physical layer (shadowing propagation model) on simulating the performance of well known $AODV$ and DSR on-demand wireless routing protocols. $AODV$ and DSR are non-position based routing schemes, where source issues route discovery via blind flooding (each node receiving route request message will retransmit it once), and destination replies to source using memorized path. Qin and Kunz [8] proposed new signal power thresholds for route discovery to enable the selection of links with strong enough signal strength and reduce some protocol control messages. They report significant increase in the packet delivery ratio and decrease in packet latency, and suggest that link status is a better metric than hop count for selecting routes in shadowing models.

MIT group [9] proposed to use the *expected transmission count metric* (ETX) for finding high throughput paths on multi-hop wireless networks. The ETX metric takes into account the effects of link loss ratios, asymmetry in the loss ratios between the two directions of each link and interference among links of a path. Then they apply ETX metric to $DSDV$ and DSR routing protocols and show that ETX metric improves performance. The protocols are tested on an 29 node 802.11 test-bed. Their observations are based on a real implementation, without giving any theoretical results or analysis in support of observations.

Banerjee and Misra [10,11] considered the cost of retransmitting messages due to link errors, and derive some optimal formulas and protocols for minimum energy routing. They considered separately end-to-end retransmissions EER (no acknowledgement or error recovery between any two links on a path) and hop-by-hop retransmissions HHR (where message is retransmitted between two nodes until it is received and acknowledged correctly). They first observed that the bit error rate associated with a particular link is a function of the ratio of received signal power to the ambient noise. In the variable-power transmission, they conclude that it is optimal if a transmitter adjusts transmitting power to ensure that the signal strength received by the receiver is independent of the distance d between two nodes. It is not clear what is the optimality measure selected to make this conclusion. It is used, however, as basis to make other conclusions. One immediate consequence of this approach is that, since reception power is fixed, the link error rate between any two nodes is fixed; therefore, probability p_{link} used in expressions is a fixed number. It also follows that transmission power, to achieve that, is proportional to d^β, where d is the distance between two nodes S and D. The authors then derive optimal minimum energy paths in EER case. The optimal number of hops N to minimize energy for transmission, assuming that retransmissions from S and D are done until message is received, is computed. The cost of acknowledging back from destination to source is not considered. In ideal case, additional nodes are placed so that the distance between them is $\frac{d}{N}$. The energy for N such transmissions is $N(\frac{d}{N})^\beta$. The probability of receiving it correctly after N hops is $(1 - p_{link})^N$. Thus, it will be sent expected $\frac{1}{(1-p_{link})^N}$ times before receiving correctly. Thus, the expected energy after retransmissions if necessary, is $\frac{N(\frac{d}{N})^\beta}{(1-p_{link})^N}$. This is then considered as a function of N, with d and p_{link} being fixed, to derive the optimal value for N, which depends only on β and p_{link}. However, since the 'optimal' value for N is independent of the distance d between source and destination, the same derivation can be recursively repeated between any two nodes in N intervals between S and D (thus producing N^2 intervals instead of N intervals), to arrive at even more power efficiency with the same arguments. This is clearly contradictory. In fact, with the energy model used, the optimal value for N is indeed the infinitely large one. However, if the computing cost and minimal reception power are taken into account, the optimal N is a finite number. The authors also considered HHR case, using similar arguments. The problems of finding minimal energy routes appears more difficult than assumed in this article, and we will address it in our future work. In this article, we consider a simpler case of expected hop count optimal routes in HHR case and create basis for later study of power and cost efficient routes.

Broadcasting in ad hoc networks has been extensively studied recently. The reader is referred to [13] for a survey of various methods proposed. Physical layer impact on broadcasting task has not been studied so far.

3 The Log-Normal Shadowing Model

Most of the published results in ad hoc wireless routing and broadcasting are based on free-space or two-ray ground propagation models, which are simplistic and idealistic physical layer models. However in real scenarios, the received signal strength is not only dependent on the distance between the transmitter and the receiver but also on the environment. Moreover, subsequent transmissions with the same transmission power, between same nodes in the same environment are not received with the same signal power. This means that, depending on threshold for correct reception, that message may or may not be received based on some random events. Following [8], we model a realistic physical layer using the shadow propagation model, where the noise element is modelled by a gaussian distribution. A brief description of the shadow fading model is given below.

The shadowing model is a statistical model. The mean received power at a distance d is computed relative to $P_r(d_0)$ (since the free space and two-ray ground models do not hold for $d = 0$) as

$$[\frac{P_r(d)}{P_r(d_0)}]_{db} = -10\beta \log(\frac{d}{d_0}) + X_\sigma.$$

This is also called the log-normal shadowing model (in dBs). The shadowing model consists of two components, the path loss model, where d_0 is the reference distance, β is the loss exponent. X_σ is a Gaussian random variable with zero mean and standard deviation σ. Both β and σ are obtained by measurements.

The average path loss for an arbitrary distance d can be expressed as a function of distance, using the path loss exponent β, as follows:

$$\overline{PL}(d) \propto [\frac{d}{d_0}]^\beta$$

$$\overline{PL}[dB] = \overline{PL}(d_0) + 10\beta \log[\frac{d}{d_0}].$$

With Log-normal shadowing model,

$$\begin{aligned} PL(d)[dB] &= \overline{PL}(d) + X\sigma \\ &= \overline{PL}(d_0) + 10\beta \log[\frac{d}{d_0}] + X_\sigma \end{aligned}$$

$$P_r(d)[dBm] = P_t[dBm] - PL(d)[dB]$$

$$P_r(d)[dBm] = P_t[dBm] - \overline{PL}(d_0) - 10\beta \log[\frac{d}{d_0}] - X_\sigma,$$

where X_σ is the 0 mean random variable (in dB) with standard deviation of σ (in dB). Free Space propagation model is used to calculate details at the reference distance d_0.

3.0.0.1 Probability of Reception: Log-normal shadowing can be used for area coverage calculations. The probability that the received power at a location d exceeds γ can be given as:

$$Pr[P_r(d) > \gamma] = Q[\frac{\gamma - \overline{P_r(d)}}{\sigma}]$$

where

$$Q[z] = 0.5[1 - erf[\frac{z}{\sqrt{2}}]]$$

$$Pr[P_r(d) > \gamma] = 0.5(1 - erf[\frac{\gamma - \overline{P_r(d)}}{\sqrt{2}\sigma}]).$$

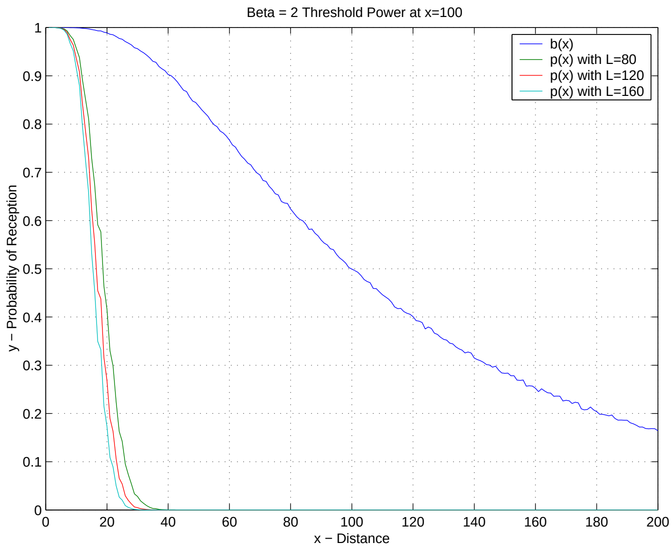

Fig. 1. $b(x)$, and $p(x)$ with $L = 80, 120, 160$ for $\beta = 2$

Therefore signals fade with distance and noise and there exists a probability for proper signal reception at the destination. We have derived the physical layer model in our first article [14], as follows. The probability of receiving packet is $p(x) = b(x)^L$, where L is the length of the packet, and $b(x)$ is the probability of receiving a bit successfully. Note that here we do not assume existence of any error correcting scheme, to recover some incorrectly received bits. Figure 1 plots the probabilities of bit and packet reception, with $\beta = 2$ and $L = 80, 120, 160$, using the shadowing propagation model. The bit transmission radius B is defined as the distance for which $b(B) = 0.5$ and the packet transmission radius R is defined as the distance for which $p(R) = 0.5$ is satisfied. We can observe that for $L = 120$ and $\beta = 2$, $B \approx 5.98R$.

The exact computation of $p(x)$, for use in routing decisions, is a time consuming process, and is based on several measurements (e.g. signal strengths, time delays, GPS) which are already causing some errors. It is therefore advisable to consider a reasonably accurate approximation that will be fast for use. Having in mind packet length $L = 120$ and an error within 4%, we designed the following approximation for $p(x)$. We approximated it by $P(x) = (1 - \frac{(\frac{x}{R})^{2\beta}}{2})$ for $x < R$, and $\frac{(\frac{(2R-x)}{R})^{2\beta}}{2}$ for all other x, where β is the power attenuation factor, with fixed value between 2 and 6. We received satisfactory precision with this approximation for $\beta = 2$ and $\beta = 4$ values. One can observe that the power attenuation factor in the approximation is 2β rather than β. This is due to approximating packet probability rate rather than bit probability rate, and the greater impact of packet length on packet reception at larger distances. Our best approximation for bit probability rate is, in fact, the same expression except that power attenuation factor is β instead of 2β. We anticipate that, in general, power attenuation factor $q\beta$ can be used, where q depends on L. Note that in the sequel we still use the notation $p(x)$ although the results were in fact derived using its approximation $P(x)$.

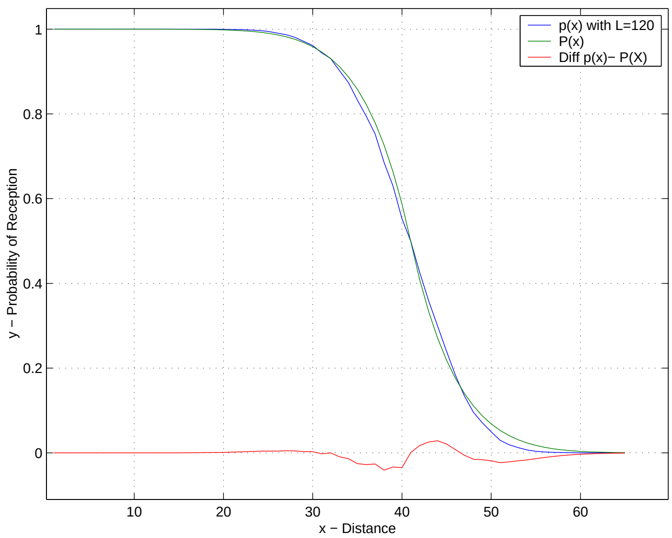

Fig. 2. $p(x)$, $P(x)$, and $p(x) - P(x)$ graphs for $\beta = 4, L = 120$, $B = 100$, $R = 41$.

Figure 2 shows the difference between $p(x)$ and the selected approximation $P(x)$ for $\beta = 4, L = 120$. The observed relative error of the approximation is below 4% for $x \leq 2R$.

4 MAC Layer Protocol Between Two Nodes

In this section, we consider HHR (hop-by-hop retransmission) routing protocol, where the sender of a packet requires the acknowledgement from receiver. To simplify our protocols and analysis, we assume that receiving node needs to send separate acknowledgement and forwarding packets to the previous and the next nodes on the route. We describe a simple MAC layer communication protocol between two nodes and present related analysis. After receiving any packet from sender, the receiver sends u acknowledgements. If the sender does not receive any acknowledgement, it retransmits the packet. We then derive the expected number of messages in this protocol, which is our proposed measure of hop count between two nodes. The count includes transmissions by sender and acknowledgments by receiver. Both the acknowledgement and data packets are of same length. This hop count is then used as weight in the shortest hop count path algorithm, for performance comparisons.

Let S and A be the sender and receiver nodes, respectively, and let $|SA| = x$ be the distance between them. Probability that A receives the packet from S is $p(x)$. Probability that S receives one particular packet from A is $p(x)$ and the probability that it does not receive the packet is $1-p(x)$. Therefore, the probability that S does not receive any of the u acknowledgements is $(1-p(x))^u$. Thus, the probability that S receives at least one of u acknowledgements from A is $1-(1-p(x))^u$. Therefore, $p(x)(1-(1-p(x))^u)$ is the probability that S receives acknowledgement after sending a packet and therefore stops transmitting further packets. Thus the total expected hop count between two nodes at distance x is

$$\frac{1}{[p(x)(1-(1-p(x))^u)]} + \frac{u}{[(1-(1-p(x))^u)]},$$

where the first term represent message count while the second term represents acknowledgement count. It is an interesting question to determine the value of u which will minimize the expected hop count, for given distance x. It appears that the best u value is one that satisfies $u * p(x) \approx 1$. The choice follows since the expected number of acknowledgements is 1. Thus the best choice of u for a given $p(x)$ is $round(1/p(x))$. The experiments show that this choice can be further optimized by using delayed rounding-off $((round((1/p(x)) - .1))$ to reduce the hop count variations between u transitions. Figure 3 shows the expected hop count for $u = 1, 2, 3, 4$ and confirms that dynamically calculated u values using the above method are optimal choices for different probability values.

Thus the choice of u does not need to be fixed in MAC protocol. It can be dynamically calculated using the $p(x)$ value for optimal hop count performance. This expected hop count can be used as a weight in the Dijkstra's shortest path algorithm to derive hop count optimal paths between any two nodes.

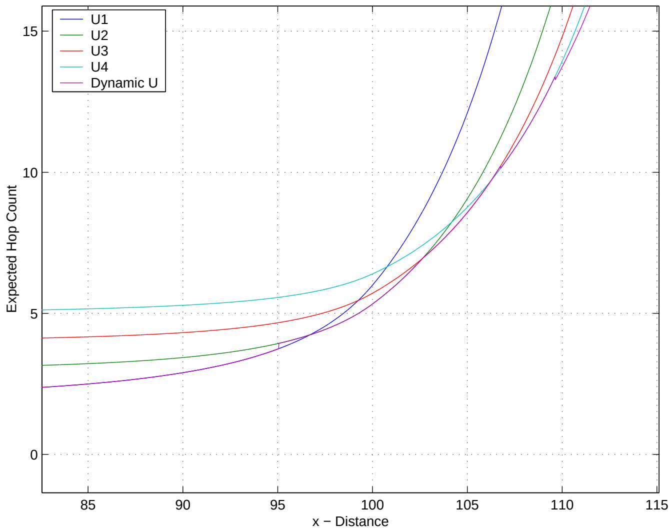

Fig. 3. Dynamic calculation of u value with $\beta = 4$, $R = 100$.

5 Optimal Packet Forwarding Distance is Less Than Transmission Radius

In [14], we further show that the optimal packet forwarding distance to minimize the hop count is less than the transmission radius R. To derive this result, we place $(n-1)$ equally spaced additional nodes, if needed and desired, between source S and destination D, along the straight line joining S and D. Let $x = d/n$ be the distance between two consecutive nodes (see Fig. 4). We now derive the optimal values for n and x, by finding the expected hop count of such placement, and finding its minimum analytically. We then show that such an ideal placement is achieved for $x < R$.

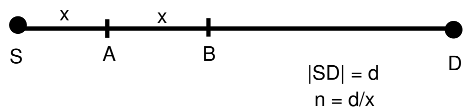

Fig. 4. Placing n nodes.

By applying the earlier analysis in Section 4, the total expected hop count from source to destination is

$$\frac{d}{x}\left[\frac{1}{[p(x)(1-(1-p(x))^u)]} + \frac{u}{[(1-(1-p(x))^u)]}\right].$$

In order to discuss optimizing a function independently on particular distance d, and particular transmission radius R, we consider then optimizing instead the function

$$h(x,u,\beta,R) = \frac{R}{x}\left[\frac{1}{p(x)(1-(1-p(x))^u)} + \frac{u}{(1-(1-p(x))^u)}\right].$$

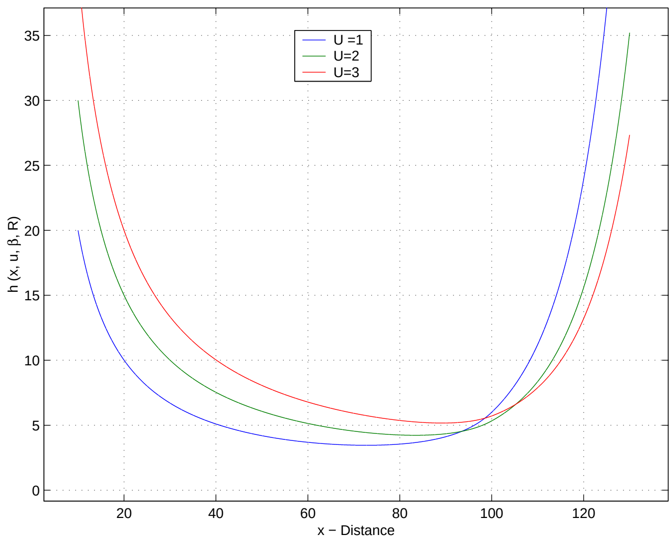

Fig. 5. Hop Count as a function of distance x, for different u ($\beta = 2$).

Figure 5 shows $h(x,u,\beta,R)$ as a function of x, for $\beta = 2$. The expected hop count is obtained when these normalized values are multiplied by $\frac{d}{R}$. We can observe that the expected hop count values are low in the range approximately $0.60R$ to $0.90R$ for $u = 1$, about 50% higher at $x = R$ and very high for $x > R$. For small x, the expected hop count is very high (and is not even shown in the figure, where x starts from $0.1R$). Let $IHC(u,\beta)$ be the minimum value for $h(x,u,\beta,R)$ over all values of x. Using our approximation for $p(x)$, for $\beta = 2$, we derived the minimum $IHC(1,2) = 3.4572$ for $u = 1$, at $x = 0.7272R$, and the ideal expected hop count is $3.4572\frac{d}{R}$. For $\beta = 4$, the minimum is $IHC(1,4) = 2.8519$ for $u = 1$ and $x = 0.7902R$. The expected hop count is minimal for $u = 1$.

6 Route discovery

We shall first consider the case when source node is not aware of the location of destination node, and not even aware of the location of its neighboring nodes. In coming sections, these assumptions will change. Qin and Kunz [8] already observed that, by applying reception signal thresholds, the delivery ratio and latency of selected route can decrease significantly. Position information was not used. In [15], we described several position based route discovery schemes. Based on distance from the sender node, the receiving node makes forwarding decision, to be explained below. We first observe that the position information can be replaced by signal strength or time delays to obtain a similar type of schemes. Based on signal strength received (the last packet alone, or average over several last packets), the receiver may estimate the distance and then follow approaches based on distance. The alternative approach is to estimate packet reception probability based on signal strength received, and use this as the function $p(x)$ needed in our derivations.

The signal threshold based scheme [8] can be replaced by distance threshold scheme as follows [15]. Each node, receiving the route discovery message for the first time, will retransmit the message if and only if its distance from the sender node is at most tR, where t is selected threshold. In routing without acknowledgements, smaller threshold is likely to increase the probability of delivery. However, in routing with hop by hop acknowledgements, threshold based technique reduces the success rate, and can increase or decrease expected hop count if threshold is selected far from optimal value.

The alternative methods designed in [15] is based on the optimality results described in the previous section. While in traditional route discovery process the waiting function is a random function (e.g. as specified by *IEEE 802.11 protocol*), the idea [15] is to apply a timeout at receiving node that depends on its suitability as forwarding node with respect to expected hop count. In general, nodes closer to optimal forwarding radius should be given smaller timeout, and chance to retransmit first. One algorithm in [15] is to set the timeout proportional to the difference between distance from receiving node and sender node, and optimal forwarding distance. However, this approach does not correspond to the optimality function described in the previous function, as some nodes may be penalized more and some nodes penalized less than fair value. However, on the other hand, such function has 'sharp' slope around optimal forwarding distance, which means that smaller constant of proportionality can be used, and destination being then discovered faster. The accumulated expected hop count can be forwarded with the message, so that the destination may wait for several incoming route discovery messages before selecting one of them as being best.

Another algorithm [15] is based on the timeout value that is proportional to

$$\frac{1}{x}\left[\frac{1}{p(x)(1-(1-p(x))^u)}+\frac{u}{(1-(1-p(x))^u)}\right] \tag{1}$$

where u is selected as discussed ($up(x) \approx 1$). The receiving node therefore assumes that all remaining hops will be made with the same distance x as the distance from sender node, and makes then fair estimate of the expected hop count it will generate if it contributes to the selected route. These timeout values can be accumulated and forwarded with the message, so that the destination receives also the expected hop count information for the first route received. The first route discovery message will also have the optimal expected hop count, since it is proportional to the accumulated delay in receiving route discovery message. Note that, to avoid simultaneous retransmission, the factor of proportionality could be large, since the function around optimal value has zero slope. To reduce the delay in finding the destination, and still find close to optimal route, a random value, selected in desired interval, can be added to the timeout. This modification leads to consideration of several incoming route discovery messages at destination before selecting best one.

7 Neighbor discovery

In the traditional unit graph model, the neighbor discovery is a trivial problem. Each node sends a *hello* message, which is received by all nodes at distance at most R. When physical layer model is considered, the problem becomes non-trivial, and more sophisticated protocols are needed.

Suppose that all nodes use equal power for transmission. We proposed *s-hello* protocol [16] as follows. Each node sends s hello messages to all its neighbors. The probability that a neighbor at distance x receives at least one of s hello messages is $q(s,x) = 1-(1-p(x))^s$. We can consider *s-unit* graph as the graph with n nodes with weights equal to $q(s,x)$. Theoretically, with s going to infinity, all nodes will get at least one hello message, but this is obviously not realistic. According to the derived function $p(x)$, only nodes at distance up to $2R$ may receive one of hello messages for 'reasonable' values of s. For $x > 2R$ we therefore may assume *s-unit* graph values $q(s,x) = 0$. With this assumption, we can then conclude that, when s goes to infinity, the unit graph with transmission radius $2R$ is obtained.

For finite values of s, *s-unit* graph, obtained with *s-hello* protocol, may be directional. That is, a node A may hear a hello message from B, but node B may not hear any hello message from A.

For dense networks, *1-hello* protocol provides sufficient number of neighbors for good performance of a localized routing protocol. In case of sparse networks, however, each node may need to transmit several hello messages before all of them collect sufficient number of neighbors. In the *variable-hello* protocol [16], each node sends hello messages until sufficient number of such messages is heard from other nodes. In this way, routing performance has a trade-off with hello messages.

8 Routing Algorithms with Acknowledgements

We now assume that each node is aware of its location, location of its neighbors, and position of destination node. Localized routing algorithms require only this information. In this section, we will consider routing with hop by hop acknowledgements (moreover, assuming that acknowledgement to previous node is made independently from forwarding message to the next neighbor on the route). We describe several localized routing protocols [14].

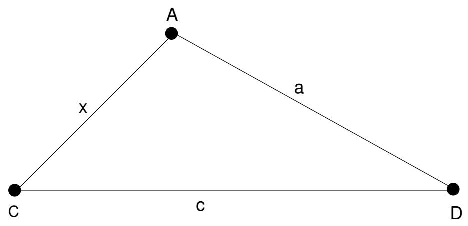

Fig. 6. Selecting the *best* neighbor A in localized routing schemes.

The *Ideal Hop Count Routing* ($IHCR$) [14] is based on the ideal packet forwarding. Let C be the node currently holding the packet destined for D. Node C will forward it to a neighbor A (closer to destination than itself) that minimizes the sum of the expected hop count measure from C to A and the ideal hop count between A and destination D. More precisely, the neighbor A that minimizes $(\frac{1}{[p(x)(1-(1-p(x))^u)]} + \frac{u}{[(1-(1-p(x))^u)]}) + \frac{a}{R} IHC(1, \beta)$ is selected, where $x = |CA|$ and $a = |AD|$ (see fig 6). The value of u is dynamically calculated based on distance $x = |AC|$, as described in Section 4. Only neighbors closer to the destination than C are considered. In the last term, however, the value for u is fixed at $u = 1$, since that choice gives the best expected performance in ideal conditions. The process continues until the destination is reached, or a node is reached that has no neighbor closer to the destination.

Let $|CD| = c$, $|AD| = a$ and $|CA| = x$ (see fig 6). The progress made by forwarding from C to A is $c - a$. Regular greedy scheme [6] maximizes $c - a$, by sending to a neighbor closest to the destination (minimizes a).

The progress that can be made by sending a packet to A is probabilistic. In $aEPR$ algorithm [14], a node C currently holding the packet will forward it to a neighbor A (closer to destination than itself) that maximizes the expected progress, which is the product of the probability of successful delivery of the packet from C to A and the progress made ($|CD| - |AD|$) by forwarding to A. Thus in $aEPR$, the neighbor A that maximizes $p^2(x)(c - a)$ is selected.

The progress that can be made by sending a packet to A can also be considered with respect to the cost measure for making such progress. The cost measure considered is the expected hop count. The expected hop count depends on distance and selected number u of acknowledgements. The progress made could be measured in different ways. In *aEPR-1* algorithm, a node C currently holding the packet will forward it to a neighbor A (closer to destination than itself) that maximizes the ratio of expected progress and cost for the progress made. Since the considered cost, expected hop count, is $1/p(x)^2 + 1/p(x)$, *aEPR-1* will select the neighbor A that maximizes $(c - a)/(1/p(x)^2 + 1/p(x))$.

Now consider the variant of the algorithm where best value of u is selected. The best value of u is approximated as $u = round((1/p(x)) - .1)$. The expected hop count is then $f(u, x) = \frac{1}{[p(x)(1-(1-p(x))^u)]} + \frac{u}{[(1-(1-p(x))^u)]}$. This variant, called *aEPR-u* , will select neighbor that maximizes $(c - a)/f(u, x)$.

Projection Progress based algorithms [14] differ from $aEPR$ schemes in the progress measure only. Instead of $c - a$, it is measured by dot product $(|CD| \cdot |CA|)$. In the $ProjectionProgress$ scheme, a node C, currently holding a packet, will forward it to a neighbor A (closer to destination than itself) that maximizes $p^2(|CA|)(|CD| \cdot |CA|)$. By substituting this new progress measure in *aEPR-1* and *aEPR-u*, we obtain two new routing schemes called *1-Projection* and *u-Projection* progress, respectively.

The well known greedy routing scheme, proposed by [6], works as follows. Node C, currently holding the packet, will forward it to the neighbor (among neighbors closer to destination than itself) that is closest to the destination. This algorithm is unambiguous with the existing definition of transmission radius in ideal unit graph model. However, with a realistic physical layer, it can receive different interpretations. We therefore modified its definition to accommodate the log normal shadowing model as follows. Consider as neighbors all nodes that are located at distance at most tR from C. Among these nodes, select one that is closest to destination (among those that are closer to destination than C) [14]. This algorithm is called *tR-greedy* routing scheme.

It was observed that the packet probability rate drops to near 0 at distance $2R$. Therefore the value $t = 2$ may be interpreted as sufficient to include all neighbors with sufficient packet probability rate to establish communication with C via some repeated hello messages. For example, if packet probability rate is 0.2, it is expected that one out of five transmitted hello messages can reach the neighboring nodes, so that node might be used for forwarding messages. However, such choices do not necessarily lead to optimal values for expected hop counts. As will be seen in experimental results, a neighbor at distance close to $2R$ may have extremely high expected hop count. We therefore believe that a better performance will be achieved if $t < 2R$.

We now present the results of our simulation study [14]. Each of n nodes ($n = 250$) is selected uniformly at random inside the square area. Dijkstra's shortest path scheme was used to test network connectivity, and only connected graph were used in measurements. The network density d is defined as the average number of neighbors per each node using the unit graph model. Two nodes are considered neighbors in this graph if and only if the distance between them is at most hR, where $p(R) = 0.5$ and hR is the distance such that $p(hR) = w$, for suitably selected threshold value w. Based on our approximation function, and value $w = 0.05$, the obtained $h = 1.4377$. We select d as the independent variable, and then find the appropriate value for R, which depends on network area size. Then this value of R is used in the approximation $P(x)$ for $p(x)$. The proposed experimental design allows for flexibility in the neighbor definition by selecting appropriate density. For example, if two nodes are considered as neighbors only when their distance is at most tR, then the corresponding density d' of a graph is approximately $d' = (t/1.4377)^2 d$, where d is the density that corresponds to $1.4377R$ neighbors. All the density values reported in tables are with respect to $1.4377R$ neighbors. We tested for $d = 6, 8, 10, 20, 24, 32, 40$ and 80. For acknowledgement based algorithms, the value of u is dynamically calculated based on the $p(x)$ value. We have used $\beta = 2$. We tested some other parameter settings, but the relative comparison remained the same.

We compared the success rates and expected hop count performance of or local algorithms $IHCR$, $aEPR$, $ProjectionProgress$ and *tR-greedy* for $t = 1$, and $t = 1.4377$ and shortest path algorithm(where link weights are computed as explained in Section 4). For averaging the hop count measure, only source-destination pairs where all of competing methods successfully found their routes to destination were considered (for low densities, the success rate of $R-greedy$ method is near zero, and it was then exceptionally ignored while averaging expected hop counts). We define the *hop count dilation* as the ratio of the expected hop count performance of the specific algorithm to that of the shortest path. The hop count dilation ratios and success rates are given in Figures 7 & 8.

It can be observed from tables that $IHCR$, $aEPR$ and Projection progress based localized algorithms have very similar performances.

Most importantly, at higher densities, $aEPR$, $IHCR$ and Projection Progress protocols had only relatively small additional hop counts with respect to the shortest weighted path algorithms, which requires global information. This is a very important achievement for localized routing schemes. Therefore, all the schemes remain candidates for future extensions (e.g. to routing scheme with guaranteed delivery).

The performance of *tR-greedy* routing algorithm was very dependant on the selected t value. For higher t, both success rates and hop count measures increase. The success rates for $t = 1$ and $t = 1.25$ are low, while hop count for $t = 1.4377$ is high. We tested more values of t in *tR-greedy* algorithm

Algorithm	Number of Nodes : 250							
	Density (with 1.4377R neighbors)							
	6	8	10	20	24	32	40	80
SP	1	1	1	1	1	1	1	1
aEPR	1.335	1.356	1.355	1.123	1.069	1.065	1.049	1.038
IHCR	1.348	1.356	1.356	1.107	1.067	1.060	1.047	1.035
Proj Progress	1.343	1.344	1.347	1.123	1.071	1.075	1.060	1.062
1.4377R Greedy	3.576	3.701	4.140	5.477	5.827	6.250	6.715	7.316
R Greedy	1.034	1.058	1.091	1.160	1.163	1.201	1.224	1.276

Fig. 7. Hop count performance of the algorithms for different densities ($\beta = 2$)

Algorithm	Number of Nodes : 250							
	Density (with 1.4377R neighbors)							
	6	8	10	20	24	32	40	80
SP	100%	100%	100%	100%	100%	100%	100%	100%
aEPR	36%	50.4%	74.4%	100%	100%	100%	100%	100%
IHCR	33.2%	47.6%	70.8%	100%	100%	100%	100%	100%
Proj Progress	34.4%	49.2%	73.2%	100%	100%	100%	100%	100%
1.4377R Greedy	45.2%	68.8%	81.2%	100%	100%	100%	100%	100%
R Greedy	0.4%	1.2%	6%	81.6%	89.6%	99%	100%	100%

Fig. 8. Success rate of the algorithms for different densities ($\beta = 2$).

($t = 0.9, 1.3, 1.6, 1.7$) but received either very high hop count or low success rates, and value $t = 1.25$ appears near best possible. Therefore, we concluded that *tR-greedy* scheme is inferior to other localized routing schemes proposed in this article for all values of t.

9 Routing Algorithms Without Acknowledgements

In the EER model, there are no hop-by-hop acknowledgements. When (and if) a message arrives at the destination, there may or may not be acknowledgments sent from the destination to the source node, as a routing task. We shall describe several localized routing protocols for EER model [17].

Consider the routing task of sending a message from source C to destination D. Consider intermediate nodes at distances $x_1, x_2, \ldots, x_n$. The probability that D receives the full message from C is $p(x_1)f(x_2), \ldots, p(x_n)$, which needs to be maximized. This is equivalent to maximizing $log(p(x_1)p(x_2), \ldots, p(x_n)) = log(p(x_1)) + log(p(x_2)) + \ldots + log(p(x_n))$. The shortest weighted path algorithm can be applied to find the route with maximal probability of delivering packet by assigning $-log(p(x_1)), -log(p(x_2)), \ldots, -log(p(x_n))$ as the respective weights for the links.

In order to derive an ideal EER algorithm, consider placing $n-1$ equally spaced nodes between the source and destination nodes, S and D. Let $x = \frac{d}{n}$ be the distance between the two consecutive nodes, where $|SD| = d$. The probability of receiving a message at the destination D is $p(x_1)p(x_2), \ldots, p(x_n) = (p(x))^n$. By taking the logarithm, this can be written as $nlog(p(x)) = \frac{d}{x}log(p(x))$, which needs to be optimized for the optimal placement distance x and for calculating ideal probability.

By applying l'Hôpital's rule, and the approximation for $p(x)$, we can show that, for the function $\frac{d}{x}log(p(x))$, the optimal value of x is 0 and the ideal probability is 1.

Following this observation, a localized EER algorithm can be described as follows [17]. The node C currently holding a message will forward it to a neighbor A (closer to destination that itself) that maximizes the sum of logarithmic probability to deliver to A and ideal logarithmic probability of delivering from A to D. However, the later ideal probability is 1 (that is, the logarithmic probability is 0). Therefore, the algorithm simply forwards the packet to neighbor A that maximizes $p(x)$, where $x = |CA|$, which is the closest neighbor to C among nodes which are closer to D than C. The process continues until the destination is reached or a node is reached that has no neighbor closer to the destination.

The described localized algorithm will also be referred to as the NC (nearest closer) routing scheme. This localized routing scheme was already proposed in [12].

The progress made by forwarding from C to A is $c - a$. This progress is probabilistic. In non-acknowledged progress routing ($nEPR$) algorithm [17], a node C currently holding message will forward it to a neighbor A (closer to destination than itself) that maximizes the expected progress, which is the product of the probability of successful delivery $p(x)$ of the message from C to A and the progress made ($|CD| - |AD|$) by forwarding to A. Therefore, the neighbor A that maximizes $p(x)(c - a)$ is chosen to forward the message.

The Iterative Expected Progress Algorithm ($InEPR$) [17] is an improved variant of $nEPR$. The algorithm can be described as follows. As in $nEPR$, a node C currently holding message will first find a neighbor A that maximizes $p(|CA|)(|CD| - |AD|)$. Then, an intermediate node B (closer to destination than C, if exists) is found (that is neighbor to both C and A) which satisfies $p(|CB|)p(|BA|) > p(|CA|)$ and has the maximum $p(|CB|)p(|BA|)$ measure. If found, B becomes new forwarding neighbor, taking the role of A. This process is iteratively repeated until no improvement is possible. Node C will forward the message to the selected neighbor A which than applies again the same scheme for its own forwarding.

Projection Progress based algorithms [17] differ from $nEPR$ schemes in the progress measure only. Instead of $c - a$, it is measured by dot product ($CD \cdot CA$). In the $ProjectionProgress$ scheme, a node C, currently holding a packet, will forward it to a neighbor A (closer to destination than itself) that maximizes $p(|CA|)(CD \cdot CA)$, where $(|CD| \cdot |CA|)$ is the dot product of two vectors. The *Iterative projection progress* scheme [17] is very similar to the $InEPR$ scheme, except that the first candidate node A is found using the projection progress method, (maximizes $p(|CA|)(CD \cdot CA)$), instead of $nEPR$ scheme.

The two performance measures we have used to compare the algorithms are the success rate and the probability of successful delivery. Success rate measures the rate of success in finding a route from the source to the destination in the attempted cases. The probability of successful delivery is the product of all probabilities of successful delivery along all hops of the computed route. The success rates in finding a route and probability of successful delivery along the found route to the destination are computed for the $InEPR$, Iterative Projection Progress and EER (NC) algorithms. Their performance is compared with that of *tR-greedy* (with t values 1, and 1.4377), and globalized shortest (weighted) path algorithm. The weight assignment scheme is as described in Section 9. The probabilities of delivering along found routes are measured only for source-destination pairs for which respective routes were successfully found by all the considered schemes (for low densities, the success rate of $R-greedy$ method is near zero, and it was then exceptionally ignored while averaging probabilities).

We define the *probability dilation* as the ratio of the probability of successful reception at the destination of the particular algorithm to that of the shortest path algorithm. The probability dilations are given in Figure 9 for $n = 250$. The results we obtained give low probability even for the shortest weighted path algorithm (for example 0.00334884, 0.00684035, 0.0302591, 0.361463, 0.467599, 0.64934, 0.761852 and 0.933141 for densities 6, 8, 10, 20, 24, 32, 40 and 80 respectively).

Figure 10 gives the success rates of these algorithms. For higher densities,

Algorithm	Number of Nodes : 250							
	Density (with 1.4377R Neighbors)							
	6	8	10	20	24	32	40	80
Shortest Path	1	1	1	1	1	1	1	1
EER (NC)	0.082	0.123	0.213	0.470	0.559	0.741	0.829	0.949
I nEPR	0.189	0.202	0.183	0.467	0.552	0.721	0.806	0.939
I Proj Progress	0.186	0.200	0.205	0.474	0.567	0.722	0.805	0.938
1.4377R Greedy	1.372*	1.381*	1.439*	26.51*	39.5*	114.9*	351.1*	5618.2*
R Greedy	0	0.023	0.074	0.0438	0.044	0.051	0.062	0.103

* To be multiplied by E-7

Fig. 9. Probability dilations at destination for different algorithms, $n = 250$.

our localized algorithms $InEPR$, *Iterative projection progress* and $EER(NC)$ nearly match the performance of the shortest path algorithm (on both measures), which is a significant success. However, $tR - greedy$ methods were inferior. For sparse networks, $InEPR$ and *Iterative projection progress* schemes performed best among localized schemes, better than $EER(NC)$, while $tR - greedy$ methods appear extremely inferior. Therefore $InEPR$ and *Iterative projection progress* schemes are good candidates for use with a localized recovery routing scheme to guarantee finding a route.

Algorithm	Number of Nodes : 250							
	Density (with 1.4377 Neighbors)							
	6	8	10	20	24	32	40	80
Shortest Path	100%	100%	100%	100%	100%	100%	100%	100%
EER (NC)	35%	49.6%	73.6%	97.6%	99.6%	100%	100%	100%
I nEPR	38%	52.8%	76.4%	98.4%	100%	100%	100%	100%
I Proj progress	38%	52.2%	76%	98.4%	100%	100%	100%	100%
1.4377R Greedy	46%	68%	85.2%	99.6%	100%	100%	100%	100%
R Greedy	0%	1.2%	8.4%	79.6%	87.2%	97.2%	100%	100%

Fig. 10. Success Rates

10 Power Aware Localized Routing

The transmission power so far was assumed to be fixed. If nodes can adjust transmission power, a natural criterion is then to minimize the power needed for routing, instead of minimizing expected hop counts. Localized power aware routing schemes, for the unit graph model, were described in [12], and recently in [18]. Physical layer impact on power aware routing is considered in our work currently in progress [19], as follows.

So far we have used packet reception probability function $p(x)$ for two nodes at distance x, assuming fixed transmission power at each node, and we defined transmission radius R so that $p(R) = 0.5$. In order to simplify further discussion, we shall use a normalized function $g(y)$ defined by $g(y) = p(Ry)$. That is, $g(0) = p(0) = 1$, but $g(1) = p(R) = 0.5$. In this way, we reduce number of variables in derivations.

Suppose that message is short, the size of one packet, and consider energy needed to transmit the message, with acknowledgement (HHR case). Assume that two nodes A and B are at fixed distance d. What is the minimal expected energy to send a message between them (retransmissions are counted)? If the impact of realistic physical layer is ignored (which is the case in almost all existing algorithms) then there exist a threshold for received signal power for correct packet reception. In order to receive signal with minimal necessary signal strengths, the power for sending message must be proportional to d^β. The exact transmission power is then d^β multiplied by a constant, which is assumed here to be 1 for simplicity. Let E is energy for processing signals at both transmitter and receiver nodes. Therefore the energy needed by sending node is $E + d^\beta$, while energy at receiving node is E, for a combined energy $2E + d^\beta$. When realistic physical layer is considered, the probability of packet reception depends on transmitted energy. There exists again a fixed reception power needed so that the packet reception probability is 0.5. The physical constant multiplier for d^β is then selected with respect to such probability, and we again assume that it is 1 for simplicity. With such choice of physical constants, if two other nodes are at distance r then the transmission power needed to achieve reception probability 0.5 is r^β.

Assume now that two nodes are at distance d, but packet is sent with transmission power r^β. What is probability of reception? The function $p(x)$ was derived assuming transmission power is fixed, but distance to sender varies. Here we have opposite behavior, distance between nodes is fixed, but transmission power varies. These two scenarios reflect the same physical aspects, therefore the reception probability is $p(Rd/r) = g(d/r)$.

We assume same MAC protocol used in article [14]. Receiver node sends u

acknowledgements for each received packet. Transmissions and acknowledgements in general do not need to be done with the same transmission powers. However, since they use the same probability function, we can argue that the optimal power is achieved when they both use the same power. Receiver node acknowledges the message only after receiving it correctly. Therefore sender node would like to minimize power needed for expected one reception of message. The same logic applies for acknowledgements; the expected number of them received should also be 1. Thus the problem decomposes into one of minimizing energy needed for receiving expected number of one packet between two nodes. Further, the optimal value for u is 1, because of the shape of function $p(x)$.

The sender node needs on average $1/g(d/r)$ messages to send for one correct reception. The energy needed for each transmission and reception is $2E + r^\beta$. The expected number of transmitted packets (for $u = 1$) is $1/g^2(d/r)$, while the expected number of acknowledgements is $1/g(d/r)$. Thus the overall required energy is $(2E+r^\beta)/g^2(d/r))+(2E+r^\beta)/g(d/r)) = (2E+d^\beta(r/d)^\beta)/g^2(d/r))+(2E + d^\beta(r/d)^\beta)/g(d/r))$. This is a function of one variable that needs to be optimized for r as function of d.

Suppose that two nodes A and B are at distance D, and we want to minimize the transmitting power by dividing interval into N hops. The length of each hop is $d = D/N$. Each of N transmissions can be assumed to be taken at same power, thus each transmission takes power $(2E + (zd)^\beta) = (2E + (zD/N)^\beta)$. Thus the total power is $(D/d)((2E + r^\beta)/g^2(d/r)) + (2E + r^\beta)/g(d/r)))$. This function needs to be optimized for d. When $E = 0$, optimal value would be $d = 0$.

Optimal hop count and power aware routing for EER and HHR cases can be derived in several ways. One is based on paradigm [12] for minimizing the sum of power to a neighbor and ideal power consumption from that neighbor to destination, which is obtained by minimizing above expression.

Other localized routing algorithms [19] are based on progress based routing schemes described in [18]. For example, progress $(c - a)$ is made with the cost equal to energy spent in forwarding to a neighbor, which is derived above. Then neighbor that maximizes $(c - a)/(2E + d^\beta)$ is selected. Several other schemes can also be derived [19].

11 Routing multi-packet messages

In [20] we considered routing of messages that are composed of several packets. Assume that a message is divided into M packets, each packet sent separately.

We consider here only routing with acknowledgements in hop by hop fashion (HHR case). To simplify, we assume that with one acknowledgement packet, receiver may acknowledge several packets at once. Obviously this may not be possible if the message is very long (for instance, the number of packets is more than the packet length). We assume that such long messages are subdivided into manageable numbers of sub-messages, each of them being handled as follows.

In general, acknowledging each packet separately is not a hop count efficient solution. To simplify, we assume that sender and receiver are aware of each other, and that the value of M, the number of unacknowledged packets m (initially $m = M$), and packet ID number, are in each packet, so that the receiver knows where packets belong and how many of them are expected. If acknowledgement packet is not received correctly by sender node, no information about receiving any of packets can be derived. We proposed the following efficient MAC layer protocol [20].

Sender node A marks each of its $m = M$ packets as unacknowledged. The following loop repeats until all M packets are acknowledged (or $m = 0$). Sender node A sends all m unacknowledged packets to the receiver node B. If any new packet is received correctly, receiver B sends back to A one packet, containing information on each packet whether or not it is received, u times. If any of u acknowledgement packets is received correctly, A will read the report and mark acknowledged packets which will not be sent again (and update m).

Note that, if none of packets is received correctly by B in a round, no acknowledgement is sent then. It is a challenge to find the best value of u. For very good channel, more than one acknowledgement merely increases packet count, while for a noisy channel, few packets may be beneficial to acknowledge several received packets and prevent retransmitting all of them. Additionally, u may depend on M, and also on the percentage of correctly received packets (as alternative or corrective measure of $p(x)$). Further, as the number of unacknowledged packets decreases, the value of u may be dynamically adjusted (reduced). The goal is to minimize expected hop count.

First we need to decide what is the best value for u. The number of packets sent by A is M. Assume that at least one of them is received by B. Then u acknowledgements are sent, so total number of packets sent is $M + u$. Each of them is received by B with probability $p(x)$, and acknowledged correctly with probability $p(x)[1 - (1 - p(x))^u] = g(u, x)$ (to shorten notation). Therefore the expected number of received packet acknowledgements is $Mg(u, x)$. This is the benefit, whose cost is u packets to send to get the benefit. Thus the expected profit is $Mg(u, x) - u$. This needs to be optimized for u, which can be done with straightforward calculus means.

Transmission between two nodes consists of several rounds. In the first round, the number of packets sent is $M+u$. The number of packets that remains unacknowledged, for the next round is then $m = M - Mg(u,x) = M(1 - g(u,x))$ where u is taking best determined value. The best u value for it may change. It appears that it is challenging to design a formula for the overall expected hop count (EHC) (summation of $m+u(m)$ over all rounds). However, EHC can be found by a local computation at each node, and the obtained result can be used in a shortest weighted path routing scheme. These values can be also used to define greedy progress based routing algorithm, as follows. Let $EHC(AB)$ denote the weight on edge AB, that is expected hop count between them. Let the message be at node A, let C be one of neighbors of A, and let D be destination. The progress made is $|AD| - |CD|$. Choose neighbor C which minimizes cost per progress, that is, minimizes $EHC(AC)/(|AD| - |CD|)$. Similarly, projection progress based routing can be considered [20].

12 Broadcasting

We shall now consider the task of broadcasting, assuming that each node is aware of the position of itself and its 1-hop (and possibly 2-hop) neighbors. The proposed solution [21] is based on the notion of physical layer based dominating sets, as follows.

Let $A_1, \cdots, A_k$ be active neighbors of given node B, and let $x_1, \cdots, x_k$ be their distances to B. Then $p(x_1), \cdots, p(x_k)$ are packet probability rates. The probability q that at least one of packets from active nodes is received by B is then $q = 1 - (1 - p(x_1))(1 - p(x_2)) \cdots (1 - p(x_k))$.

Node B is *m-covered* by active nodes A_1, A_k if $q \geq m$ [21]. A set of nodes is *m-dominating set* if each node is either in the set or is *m-covered* by nodes from the set [21].

These new simple definitions can then be used to efficiently design some dominating sets based on realistic physical layer. We shall describe here physical layer based generalized dominating sets, following the corresponding notion for the unit graph model, described in [13] (note that the original definition is due to Dai and Wu, improved then in [13] to eliminate communication overhead in defining the set, and further simplified by D. Simplot).

Let $A_1, \cdots, A_k$ be the set of higher ID neighbors of B. If the set is empty or disconnected then B is in dominating set. If the set is connected and each neighbor of B is *m-covered* by them then B is not in dominating set. Finally, if any neighbor of B is not *m-covered* by the set then B is in dominating set.

Note that, when we place $m = 1$ in all mentioned definitions and protocol, that we get the corresponding definitions and protocols for the unit graph model. Also, note that the connectivity needs to be defined as well, to make the definition complete.

We shall now describe how to broadcast by using this notion of dominating sets. After receiving a broadcast message, node A will set a timeout, short if it is in dominating set, and long if not. It calculates probabilities of each neighbor receiving the same message, and eliminates *m-covered* neighbors from the list. These probabilities are updated for any further copy of received message. Updates may include timeout that can be extended with more incoming messages. At end of timeout, if all neighbors are *m-covered*, retransmission is cancelled. Otherwise, node retransmits.

Further research is needed to relate dominating set definitions and the corresponding broadcast protocol with the probability of delivering message to each node in the network. This is not straightforward, and is related to the considerations of connectivity in given ad hoc network. What it means that a set of nodes is connected? When message is broadcast from a source, there exists probability of reaching any of other nodes. Connectivity can also be considered via the routing task between any two nodes. Routing can further proceed with and without hop by hop acknowledgements. A single route, several routes, or all routes between two nodes can be considered. For instance, one can propose the following definition for connectivity, which leaves details to be clarified in further research.

Definition: A set of nodes is *q-connected* if the probability of a packet to be routed/broadcasted successfully between any two of nodes is $\geq q$.

13 Conclusion

To the best of our knowledge, this is the first study of position based routing and broadcasting in ad hoc network with a realistic physical layer. We investigated routing with and without acknowledgements, and presented several greedy routing algorithms for ad hoc wireless networks. We show that realistic physical layer does have impact on the choice of best localized schemes.

The localized nature of the protocols avoids the energy expenditure and communication overhead needed to build and maintain the global topological information. Our simulation results show that, for higher densities, the performance of our localized algorithms is close to the performance of the shortest (weighted) path algorithms, which require global knowledge.

Our group plans to address, in our future research, several other problems, including physical layer based GFG routing with guaranteed delivery [7], location updates for efficient routing, greedy beaconless routing, depth-first search based routing, power aware beaconless routing, beaconless GFG routing, and area coverage in sensor networks. A number of other extensions to presented work remain as open problems for future research. For instance, we considered only separate HHR model, while one could study also model where forwarding messages may be used also as acknowledgement messages. We considered only a simple packet reception model, bit by bit. If some error correcting codes are applied, the packet probability rate will also change. New approximation of $p(x)$ is then needed, which may impact the performance of algorithms. Appropriate MAC layer protocols may be required to accommodate considered coding schemes. Next, we considered log normal shadowing model. It is possible to consider other models for physical layer, such as Raleigh fading.

We anticipate that this direction of research will soon receive more attention in the ad hoc networks research community.

References

[1] S. Basagni, M. Conti, S. Giordano, I. Stojmenovic (eds.), "Mobile Ad Hoc Networking," IEEE Press/Wiley, 2004, to appear.

[2] G. Giordano, "Mobile Ad Hoc Networks," in *Handbook of Wireless Networks and Mobile Computing*, I. Stojmenovic (ed.), Wiley, 2002, pp. 325-346.

[3] D. Niculescu, "Positioning in ad hoc sensor networks," *IEEE Networks*, to appear.

[4] I. Stojmenovic, "Location updates for efficient routing in ad hoc networks," in *Handbook of Wireless Networks and Mobile Computing*, I. Stojmenovic (ed.), Wiley, 2002, pp. 451-472.

[5] I. Stojmenovic, "Position based routing in ad hoc networks," *IEEE Commmunications Magazine*, Vol. 40, No. 7, July 2002, pp. 128-134.

[6] G.G. Finn, "Routing and addressing in large metropolitan-scale internetworks," *ISI Research Report*, ISU/RR-87-180, March 1987.

[7] P. Bose, P. Morin, I. Stojmenovic, J. Urrutia, "Routing with guaranteed delivery in ad hoc wireless networks," *ACM DIAL M Workshop*, Seattle, August 1999.

[8] L. Quin, T. Kunz, "On-demand Rounting in MANETs: The Impact of a Realistic Physical Layer Model," *Proceedings of the Second International Conference ADHOC-NOW*, 2003, pp. 37-48.

[9] D. De Couto, D. Aguayo, J. Bicket, R. Morris, A high-throughput path metric for multi-hop wireless routing," *Proc. ACM Mobicom*, San Diego, 2003.

[10] S. Banerjee, A. Misra, "Energy Efficient Reliable Communication for Multi-hop Wireless Networks," *Wireless Networks (WINET)*, to appear.

[11] S. Banerjee, A. Misra, "Adapting Transmission Power for Optimal Energy Reliable Multi-hop Wireless Communication," *Wireless Optimization Workshop (WiOpt'03)*, Sophia-Antipolis, France, March 2003.

[12] I. Stojmenovic, Xu Lin, Power aware localized routing in ad hoc networks," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 12, No. 10, October 2001, pp. 1023-1032.

[13] I. Stojmenovic, Xu Lin, Broadcasting and activity scheduling in ad hoc networks, in: S. Basagni, M. Conti, S. Giordano, I. Stojmenovic (eds.), "Mobile Ad Hoc Networking," IEEE Press/Wiley, 2004, to appear.

[14] J. Kuruvila, A. Nayak, I. Stojmenovic, Hop count optimal position based packet routing algorithms in ad hoc wireless networks with a realistic physical layer, submitted for publication.

[15] E. Villanueva-Pena, I. Stojmenovic, J. Kuruvila, Route discovery for minimizing hop count in position based routing in ad hoc networks with a realistic physical layer, in preparation.

[16] E. Villanueva-Pena, I. Stojmenovic, A. Nayak, Physical layer impact on finding local knowledge in ad hoc networks, in preparation.

[17] J. Kuruvila, A. Nayak, I. Stojmenovic, Greedy localized routing for maximizing probability of delivery in wireless ad hoc networks with a realistic physical layer, submitted for publication.

[18] J. Kuruvila, A. Nayak, I. Stojmenovic, Progress based localized power and cost aware routing algorithms for ad hoc and sensor wireless networks, submitted for publication.

[19] J. Kuruvila, A. Nayak, I. Stojmenovic, Power aware localized packet routing in ad hoc networks with a realistic physical layer, in preparation.

[20] J. Kuruvila, A. Nayak, I. Stojmenovic, Hop-count efficient greedy localized routing of multi-packet message with a realistic physical layer, in preparation.

[21] F. Ovalle-Martinez, I. Stojmenovic, Broadcasting in ad hoc networks with a realistic physical layer, in preparation.

Ivan Stojmenovic received Ph.D. degree in mathematics in 1985. He held regular or visiting positions in Serbia, Japan, USA, Canada, France and Mexico. He published over 170 different papers in journals and conferences, edited Handbook of Wireless Networks and Mobile Computing (Wiley, 2002), and co-edited Mobile Ad Hoc Networking (IEEE Press, 2004). His current research interests include wireless ad hoc, sensor and cellular networks. He is currently editor of several journals including Journal of Multiple-Valued Logic and Soft Computing, IEEE Transactions on Parallel and Distributed Systems, Parallel Processing Letters, and Parallel Algorithms and Applications. He guest edited recently special issues in several journals including IEEE Computer Magazine (February 2004), IEEE Networks, and Wireless Communications and Mobile Computing.

Amiya Nayak is an associate professor at the School of Information Technology and Engineering (SITE) at the University of Ottawa. He received his B.Math. degree in Computer Science from University of Waterloo and Ph.D. in Systems & Computer Engineering from Carleton University, Ottawa in 1982 and 1991 respectively. He has over 15 years of industrial experience in software engineering in avionics and telecommunication applications, working at CMC Electronics and Nortel Networks prior to joining University of Ottawa in 2002. He has been an adjunct research professor at Carleton University since 1994 and the Canadian Editor and the Book Review Editor of VLSI Design: An International Journal of Custom-Chip Design, Simulation, and Testing since 1998. His research interests are in the areas of fault-tolerant computing, ad hoc networks, and distributed computing.

Johnson Kuruvila has 15 years of industry experience in Research, S/W development, Embedded/Distributed System Design and IP Networking/Wireless protocols. Earlier he was Lead Engineer at AcceLight Networks and Director of Routing/Signaling Platforms at AcceLight Research. He was Manager of IP Routing system development at the Optera Packet Core project in Nortel Networks. He has been involved with the development of many leading edge products and is a recipient of many performance awards and has given lectures to many product groups in IP Routing Protocols, MPLS/GMPLS signalling and Parallel/Distributed Algorithms. He has given more than 10 invited lectures at various research labs and supervised many graduate and undergraduate research works at various universities. He holds degrees in Electronics/Communication Engineering and Computer Science. He has published many papers and has filed two patents. Currently, he is doing his research in Wireless ad hoc networks at University of Ottawa.

Francisco Ovalle-Martinez was born in Mexico City in 1977. He earned his Bachelor of Science degree in Telecommunications Engineering from the Autonomous National University of Mexico (UNAM), in 2001. Francisco has had the opportunity to develop and enhance, his professional skills through various software and hardware firms in Mexico, such as Red de Control Corporativo and Yum International Restaurants Mexico. His tenacity, determination and ever present desire for success, have taken him through different paths; today, he is on his way to achieve a master degree in Computer Science at the University of Ottawa, Canada. He has been awarded and holds a CONACyT grant. His interest areas mainly include mobile computing, digital communications, data networks and security.

Eduardo Villanueva-Pena graduated (with honours) from the National Autonomous University of Mexico in 2001, with the BCS degree. In 2002, he received the MSc degree in Engineering from the University of Sheffield, UK. He was an associated researcher at the Complex Systems Laboratory (Edo. Mex., Mexico) from 1998 to 2000, at the University of the Mexican Valley. He won a third prize at Mexico City Informatics Olympiad in 1998. He got the third best grade point average at the end of his bachelor studies. Currently, he is doing his PhD in Engineering as a full time student at the University of Ottawa. Mainly, his research interest resides on wireless networks.