On the Pitfalls of Directional Location–based Randomized Routing

Israat Tanzeena Haque   Ioanis Nikolaidis   Pawel Gburzynski
Computing Science Department
University of Alberta
Edmonton, Alberta T6G 2E8, Canada
{israat,yannis,pawel}@cs.ualberta.ca

Keywords: mobile ad–hoc networks, location–based routing, directional transmission

Abstract

Many location–based MANET routing algorithms rely on local decisions based on coordinates of neighboring nodes. In order to deal with the dead ends reached by the deterministic nature of location–based routing, *randomized* versions have been proposed. Randomization is seen also as helpful for dealing with highly dynamic topologies due to mobility. In addition, directional transmission is considered as one of the possible ingredients for lower energy consumption and higher capacity. We combine concepts of directional transmission with randomized location–based routing in a scheme called Directional Location-based Randomized Routing (DLR). Our simulation study demonstrates that while the packet delivery rate increases, one must be cautious when considering such combined schemes for the sake of energy efficiency.

1. INTRODUCTION

A mobile ad hoc network (MANET) is a collection of mobile wireless devices that can communicate with each other without resorting to centralized control or fixed infrastructure. Owing to the limited range of communication over a radio channel, nodes located farther than that range have to rely on intermediate nodes acting as routers. The mobile and infrastructure-less nature of the nodes imposes resource constraints (bandwidth, battery power), which make the routing problem in such networks challenging [4].

The standard taxonomy of routing protocols for MANETs identifies *proactive* protocols, i.e., ones that try to maintain up-to-date routing information at every node in anticipation of demand [2, 14, 17], and *reactive* ones, which collect the necessary routing information only when it becomes explicitly needed to sustain an actual session [6, 7, 11, 15, 16, 24]. With few exceptions (e.g., [18]), routing schemes assume point-to-point communication, whereby each node forwarding the packet on its way to the destination sends it to a *specific* neighbor. If the identity and/or location of the neighboring nodes is unknown or uncertain, flooding is used as a means of route discovery [4]. One way of limiting flooding is to use the geographic coordinates (position) of the nodes, if it is available, e.g., via GPS.

One obvious optimization criterion of the routing problem (in MANETs as well as other networks) is the number of hops from source to destination. The inclusion of power considerations in MANETs adds a new dimension to this problem, which can be viewed in three ways [22]. Namely, the objective can be *power aware* routing, which minimizes the total expense of energy required to deliver a packet from source to destination, or *cost aware* routing which strives to prolong the network lifetime by balancing the battery energy depletion over all nodes, and, finally, the two objectives can be combined in *power-cost aware* routing, which tries to minimize global energy expense while preferring nodes with a large amount of battery power still available.

Another possible option for the transceiver of an ad-hoc mobile node is to use a directional antenna. Owing to the technological advances that have rendered them inexpensive and flexible (electronically adjustable), such antennas are recently gaining ground in MANETs because of two main advantages. First, by narrowing down the transmission/reception angle, the transmission range (achievable within the same power budget) is significantly extended. Second, directionality greatly improves spatial reuse, which means that the total bandwidth (capacity) available to the network is increased. Even though these advantages have to be mitigated against some drawbacks (directional interference, lobe patterns, deafness, a special variant of the hidden terminal problem—[25]), the general consensus is to view directional antennas as an enhancing component with a high potential for improving the overall performance of a wireless ad-hoc network.

Furthermore, we distinguish between deterministic and randomized routing. A routing algorithm is said to be *deterministic* if the routing rules never rely on a "coin toss" to decide a packet's fate, i.e., every decision is arrived at via a deterministic set of rules that produce the same output under identical inputs. Routing algorithms that are not deterministic are called *randomized* [4]. Randomized schemes provide degrees of freedom in terms of selection of the next hop to use. In this sense, the randomized algorithms provide flexibility when either the location of the neighbors changes due to mobility (when e.g., a choice for next hop deemed as good at one point in time ceases to be a good choice a short time later) or because random choices "spread" the energy cost of forwarding over a larger set of nodes.

In short, the basis of the problem discussed in this paper is that of proper synthesis of directional transmission, on one end, with randomization, on the other. As we will see, the synthesis is not without problems. The intention of directional transmissions, at least in principle, is to purposefully reduce the choices for packet forwarding. Randomization, on the other hand, dilutes the selections across many possible options.

In this paper, we propose and study a new Directional Location–based Randomized routing protocol, or DLR for short. DLR uses directional antennas and its decisions are intentionally probabilistic. It attempts to push a packet towards the destination by restricting the angle for the lookup of a next-hop node. The selection of a specific node within that angle is based on its residual battery power. To alleviate the reliability problems haunting other solutions in the same class [19], DLR does not drop the packet if the constrained angle is devoid of neighbors, but instead forwards it to another neighbor, located outside the angle, in a manner that tries to balance energy expense and the extent to which the packet deviates from the target.

Adapting a location–aided routing scheme to the demands of a mobile environment is not a trivial matter. For example, the location of the destination may change over time. Even if we assume (as indeed the majority of the literature on location–based routing assumes) that we know the current location of the destination[1], determining the best neighboring nodes to route through has to continuously adapt to the changing topology. Location–based schemes that obsessively seek the most direct path to the destination can fail if they encounter “dead ends” and decide to drop the packet. The changing topology suggests that forwarding to apparently sub–optimal next hops gives a second chance to deliver along the direct path a short while later. Thus, DLR admits the possibility that seemingly non-optimal choices are in fact quite attractive in providing improved packet delivery ratio.

We study the performance of DLR in 2-dimensional and 3-dimensional deployment scenarios and compare it with *greedy* [5] and PGR (Probabilistic Geographic Routing) [19]. Simulation results show that DLR exhibits the highest packet delivery rate, while PGR ensures the longest network lifetime, and *greedy* offers the shortest path length.

2. THE FRAMEWORK

2.1. Network model

A MANET is, formally, described as a set V of n nodes placed in in 2- or 3-dimensional Euclidean space. In location–based schemes, it is assumed that each node is aware of its location, expressed as Cartesian coordinates (x,y) or (x,y,z). We forego any discussion of how locations are estimated properly, as it is a separate research issue. We assume the transmission range of all nodes is the same and equal to *R*. Two nodes can communicate with each other if and only if their Euclidean distance is at most *R*. The ability to communicate is represented by an edge between the corresponding nodes. The resulting graph, *G=(V,E)*, is the topology of the network. *G* varies over time due to the mobility of the nodes which results in *E* being dependent on time.

Given G and a pair of nodes (i,j), $i,j \in V$, the problem of energy-aware position-based routing is to find a path from i to j that minimizes energy expenditure and thus maximizes the network lifetime. The latter can be defined as the number of successful routing tasks that the network can perform until either, (a) a node (any node) loses all its energy, or, (b) a certain percentage of nodes lose their energy, or even, (c), the network becomes partitioned because of elimination, due to energy depletion, of one or more nodes.

We define *reliability* as the ratio of packets successfully received at their destinations within some time T to the total number of packets that would have been received within the same period, if every transmitted packet had always made it [19]. Generally, links can be asymmetric and then it makes sense to talk about the *forward reliability* from node i to j (denoted r_{ij}) and the *backward reliability* from node j to i (denoted r_{ji}) [19]. The inverse of reliability is the expected number of retransmissions required to deliver a packet.

2.2. Related work

One of the simplest position-based routing schemes is *greedy routing* [5], whereby the routing node selects the next-hop neighbor as the one with the shortest distance to the destination. Although the scheme works statistically well in dense networks, it suffers from the problem of *local maxima*, i.e., nodes with no neighbors in the transmission range located closer to the destination than themselves. GPSR (Greedy Perimeter Stateless Routing) and GFG (Greedy-Face-Greedy) were was proposed to address this issue [8, 1]. Upon hitting a local maximum, the protocol switches to a different mode of operation (traversing the so-called *faces*) until it can safely revert to the greedy mode.

Another position-based approach, called *compass*, was proposed in [9]. With compass, a routing node i forwards the packet to the neighbor j that minimizes the angle formed between j, i and the destination. The protocols, whose performance is similar to that of *greedy*, suffers from loops.

As these schemes do not take the energy budget into account, they do not perform very well in an energy constrained environment [19]. As pointed out in [21], such algorithms will generally perform poorly in such environments because of the tendency to exhibit hot spots.

[1]The assumption of knowing the destination location is more reasonable when seen in the context of applications where mobile nodes route via the mobile ad hoc network to static/fixed infrastructure access points.

The proper definition of local progress as the routing goal is a prerequisite for efficient routing in MANETs. Such definitions are proposed in [10], where the *power progress* objective is to minimize

$$(t \times dist^{\sigma}(c,x) + r)/(dist(c,d) - dist(x,d))$$

where c is the forwarding node, x is its neighbor, r is the constant component of a single forwarding step, and t and σ transform the transmission distance into the requisite transmission power. The denominator captures the proportion of the contribution of a single hop to the packet delivery task. To account for the remaining lifetime of a node, the *cost progress* is defined as minimizing

$$(f(x) \times (t \times dist^{\sigma}(c,x) + r))/(dist(c,d) - dist(x,d))$$

where $f(x)$ is the inverse of the node's remaining lifetime interpreted as its reluctance to forward. The two components can be combined into a single metric either by straightforward multiplication (*power progress* $\times$ *cost*) or taking the dot product over the vectors.

With PGR (Probabilistic Geographic Routing) [19], the routing is carried out in two phases. In the *discovery phase*, each node sends a `HELLO` message at T sec intervals, which contains the geographic location of that node, its remaining battery power, and a list of its neighbors with their reliability values. A link's reliability is estimated as an exponential moving average:

$$R_{t+1} = R_t \times \alpha + (1 - \alpha) \times m/(m+n)$$

where R_t is the previous estimation, $0 < \alpha < 1$, m and n are number of received and missed packets, respectively. The latter are reset to zero every w (window size) packets.

Once the discovery phase is over, each node select N of its neighbors with the highest (backward) reliabilities as the prospective next-hop nodes. An alternative way of selecting them is to use a threshold. The nodes continue updating the neighbor tables every $T_1 > T$ sec, to reduce the control overhead. A routing node c first defines an angular area (sector) centered at itself around the direction of the destination. The initial size of the sector is chosen arbitrarily and then increased up to at most π if it contains less than two neighbors. If the enlarged sector still has less than two neighbors the packet is dropped. Otherwise, c assigns every neighbor j falling into the sectors the rank $P(j) = E_{res}(j)/R(j)$, where $E_{res}(j)$ is the residual power of j, and $R(j)$ is the number of retransmission required over the link. The next-hop node is then chosen at random with the probability directly proportional to its rank. The performance study presented in [19] indicates that PGR outperforms GPSR in terms of throughput, delay, network lifetime, and the number of retransmissions per packet. However, the path length (in hops) of PGR is longer than that of GPSR.

SELAR [13] is intended for sensor networks with sink nodes. The sink node initiates a flooding-based identification of node locations and their available energy. Later on, only the energy information is updated (the network is assumed to be static). Each node defines the forwarding zone, i.e., an angular area of γ centered at the current node around the direction to the sink. The size of the zone is varied from 15 to 90 degrees to provide for a nonempty set of neighbors (empty zones trigger "gossiping," which may consume extra energy). The neighbor with the highest residual energy in the zone is selected for forwarding. While the algorithm performs better than flooding with respect to network lifetime, the nodes close to the sink may lose their energy quickly rendering the sink inaccessible. The authors suggest that the sink be automatically moved if the amount of energy available at its surrounding node drops beyond a threshold.

Regarding directional antennas, the problem of transmission energy is addressed in E-MAC [20]. When a node is in idle mode, it uses omni-directional channel sensing and then, having sensed a signal above a specific threshold, it switches to *rotational-sector* receive-mode, whereby it sweeps its directional antenna [3] over the complete 360^o circle to find the direction of the highest received signal strength. To compensate for the delay incurred by sweeping, the protocol employs a *tone signal* before sending a packet.

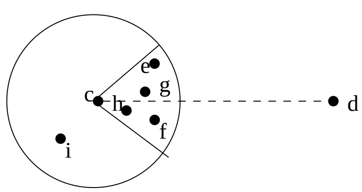
The key assumption in E-MAC is that for a given input power, the transmission range R^{direc} of a directional antenna is greater than the transmission range R^{omni} of an omni-directional antenna. Three control packets: beacon, RTS and CTS, are received over the omni-directional antenna, while ACK and data packets use the directional antenna. The purpose of the omni-directional reception of RTS and CTS is to learn about the activities in the neighborhood. With the availability of location information, a node may start transmitting (using a directional antenna) even though another transmission in the neighborhood appears to be in progress. The simulation results presented in [20] demonstrate that E-MAC improves the throughput and decreases the one-hop average end-to-end delay compared with IEEE 802.11, while bringing about considerable energy savings.

Another protocol along the same lines, dubbed DiMAC, has been proposed in [3]. In DiMAC, an idle node is in omni-receive mode, while RTS is sent directionally. As soon as the node receives an RTS, it determines the beam direction to send CTS (directionally). Once the RTS/CTS handshake has been established, data and ACK are also transmitted in directional mode.

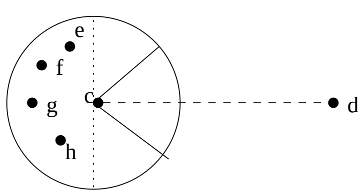
3. DIRECTIONAL LOCATION-BASED RANDOMIZED (DLR) ROUTING

In DLR, a routing node i starts by drawing a sector of size $\pi/3$ around the direction towards the destination to identify a subset of prospective next-hop neighbors. Each of those neighbors receives a rank of E_{res} (the remaining battery power). The actual node is chosen at random with the probability directly proportional to its rank. If the sector is empty, it is not enlarged. Instead, i ranks all the neighbors (located outside the sector) with weights proportional to $1/(E_{res} + c_0 \frac{n-1}{2\pi} \theta_x)$, where θ_x is the angle formed by i, the neighbor (x) and the destination and c_0 is a constant. The next node is then chosen according to the assigned weight such that chance is high that it is close to the direction of the destination (smaller angle). Assuming that the traffic load and the mobility of nodes are uniform across the network, then we expect that the energy depletion is approximately the same across all nodes. Therefore, if nodes start with the same energy reserves, they have approximately equal reserves at a later point. Hence, even though the weight biases in favor of nodes with less energy, a suitable choice of constant c_0 can amplify the impact of the angle to be dominant over the smaller differences we expect in terms of the energy E_{res} across nodes.

The following example illustrates the operation of DLR. In Figure 1 (a), the routing node c has four candidate nodes inside the sector of size $\pi/3$. The weight assigned to each node is E_{res} and the protocol will tend to pick a node with a high remaining energy. In Figure 1 (b), there is no candidate node inside the sector, and hence DLR will pick the next node from among e, f, g, h. We can expect that the protocol will exhibit a high packet delivery rate, as it tries to find an alternative path when the initial sector is empty. This path, however, may be long.



(a)



(b)

Figure 1. Example illustrating DLR.

4. EXPERIMENTS

4.1. The model

We assume the shadowing propagation model with the path loss at distance d being

$$PL(d)[dB] = PL(d_0) + 10\beta \log(d/d_0) + X_\sigma$$

where $PL(d_0)$ is the path loss at the reference distance d_0, β is the path loss exponent, and X_σ is a zero-mean Gaussian distributed random variable with standard deviation σ. The antenna gain is included in $PL(d)[dB]$,

$$PL(d_0) = 20 \log(4\pi d_0/\lambda)$$

where λ (the wavelength) corresponds to 2.4GHz. The received power is expressed as

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

where P_t is the transmission power. If the received power is less than the threshold power P_{th}, the signal is not correctly received [23]. In our experiments, β and σ are 3 and 8, respectively. The reference distance is 1m, the transmit power is 25dBm, and the threshold is -95dBm.

The energy model is the same as in [19], which coincides with the one used in ns-2 [23]. A node loses $P_t \times t_{transmit}$ amount of energy, where $t_{transmit}$ is the transmission time. Also, when receiving a packet, the energy loss is $P_r \times t_{receive}$.

The protocol uses a smart switched-beam antenna [3] with multiple predefined directional beams. There are two modes of operation: omni-directional and directional, with one mode being active at a time. The size of the main lobe in DLR is $\pi/3$; the side lobes (deemed insignificant) are approximated into a (single) sphere. The PGR protocol also starts with this main lobe, but increases it up to π. The remaining parameters of PGR are $w = 4$, $\alpha = 0.5$, and R_t initialized to zero.

For mobility, we use the *random waypoint* model, whereby each node chooses a uniformly distributed random location from a rectangular area and moves there at a constant speed selected at random from [0, V_{max}]. After reaching the new location, the node stays there for a *pause time*. Then, the node repeats the same process until the end of the simulation run. In this project, V_{max} is 10m/s and the pause time is 30 sec.

4.2. Simulation Environment

A set V of n nodes is generated at random and uniformly spread over a 1000m×1000m rectangle, with $n \in 50, 70, 90, 110, 130$. The nodes are moving using the random waypoint model with the maximum speed of 10m/s. The transmission range of each node is 200m and its initial energy is 600J. The transmit and receive power is 25dBm. In the three dimensional cases, the nodes are spread inside a 1000m×1000m×1000m cube.

In a 3D network, the transmission range of a node has been increased to 300m to make the results comparable. This is because the extra dimension reduces node density, and the network simply cannot perform (being too frequently disconnected) as its 2D counterpart (with the same number of nodes). Each experiment run simulates 500sec of operation. Each data point is the average result from 10 independent replications.

In order to maintain up–to–date information about the neighboring nodes, a neighbor discovery process is needed. Upon bootstrap, the initial neighbor discovery phase takes 40 sec, with each node emitting a beacon message every 4 sec. Neighboring nodes respond to the beacon. Following the bootstrap phase, beacons are sent once every 10sec. Routing starts after the neighbor discovery phase has been bootstrapped. Data traffic is generated by randomly selecting source–destination pairs. Each source–destination pair generates a packet to route once every 15 sec. The traffic load is not high and does not lead the network to saturation but is generated infrequently enough that mobility will almost certainly render useless the path used by the previous packet send from the same source to the same destination.

The following performance measures are collected:

- **Packet delivery rate**: the ratio of the total number of packets successfully received by the destination to the total number of packets originated at the source [19].
- **Path length**: the average number of hops taken by a packet to reach the destination in case of a successful packet delivery [19].
- **Network lifetime**: the average number of successful routing tasks before the first node in the network has lost all its energy.

5. SIMULATION RESULTS

The performance of greedy, PGR, and DLR is studied both in two and three dimensional deployments. The results are presented in Tables 1 through 6.

The packet delivery rate of the three protocols is shown in Table 1. Note that the packet delivery rate of DLR is higher than in the other protocols. This is because DLR increases the choice of alternative paths and thus reduces the packet dropping rate.

The performance of greedy and PGR is very close. PGR can reach the destination as long as it finds eligible forwarding nodes inside the sector. Greedy may drop packets due to local maxima, even though the chance of facing local maxima is less in dense networks.

As the number of nodes increases, all the protocols exhibit better performance due to high node density. For example, the chance of facing a local maximum by greedy is reduced, which tends to increase the packet delivery rate. In PGR, the chance of having more forwarding nodes inside the sector increases with the increasing node density, which also pushes up the delivery rate. Also in DLR, the delivery rate is slightly improved owing to the reduced probability of reaching the threshold.

The performance of the three routing strategies in 2D in terms of the network lifetime is shown in Table 2, with PGR being the winner. This is because PGR considers residual energy of nodes and link reliability to balance the energy utilization among the nodes. The next protocol is greedy, which does not confine routing to a narrow sector, which gives it more flexibility to distribute the energy utilization among the neighbors of a routing node. Finally, DLR has the worst performance in terms of network lifetime. In DLR, a packet may bounce back and forth, possibly several times, before arriving at the destination, which may increase overall energy usage by involving more nodes than necessary.

With fewer nodes in the network, greedy performs better than PGR. The probable explanation is that PGR has then fewer choices for next-hop nodes, due to its dependence on link reliability and the sector size. The greedy protocol retains a relatively large choice for balancing energy, even within a relatively sparse network. This advantage disappears with increased node density, as the choice for PGR becomes relevant and discriminating.

Similarly, the performance of DLR also improves with the increasing node density as the likelihood of “bouncing” a packet gets smaller.

Table 3 shows the performance of three routing schemes in terms of the average path length. Greedy is the winner here, followed by PGR and then DLR. This result was expected as minimizing the distance towards destination is greedy’s primary objective. Both PGR and DLR may traverse a longer route due to the (biased) randomization. Then, in DLR, packets may occasionally travel backwards, which can never happen in PGR.

As the number of nodes increases, the path length of greedy decreases slightly. In DLR, the likelihood of a backward “bounce” decreases, and so does the average path length. PGR, however, may need to traverse a few extra hops in such circumstances, as it always prefers shorter links with high reliability. Hence, the path length of PGR tends to increase as the network becomes denser.

The behavior of the three protocols in 3D is highly consistent with their planar performance. In particular, DLR still offers the highest packet delivery rate, the performance of greedy and PGR is almost the same and close to that of DLR. The network lifetime of greedy starts better than PGR, but, as the node density increases, PGR picks up. DLR still yields to the other protocols in terms of network lifetime, improving its performance with increased node density. In terms of the path

Table 1. The average packet delivery rate in 2D space.

	$n = 50$		$n = 70$		$n = 90$		$n = 110$		$n = 130$	
Algorithms	Aver.	Std. Dev.	Aver.	Std. Dev.	Aver.	Std. Dev.	Aver.	Std. Dev.	Aver.	Std. Dev.
GREEDY	69.50	19.86	91.90	11.23	96.80	4.89	98.30	2.71	99.40	1.90
DLR	82.90	16.26	91.90	12.78	99.30	2.21	99.50	0.85	99.70	0.95
PGR	68.30	19.97	90.10	13.09	97.50	3.87	98.00	2.21	99.40	1.58

Table 2. The average network lifetime in 2D space.

	$n = 50$		$n = 70$		$n = 90$		$n = 110$		$n = 130$	
Algorithms	Aver.	Std. Dev.	Aver.	Std. Dev.	Aver.	Std. Dev.	Aver.	Std. Dev.	Aver.	Std. Dev.
GREEDY	51.20	16.82	62.90	15.43	89.90	29.17	103.40	24.17	134.80	26.73
DLR	15.30	8.41	36.00	24.44	47.90	23.48	76.20	33.63	80.30	44.61
PGR	46.50	15.09	70.10	14.74	90.60	26.97	106.60	33.51	156.80	73.11

Table 3. The average number of hops in 2D space.

	$n = 50$		$n = 70$		$n = 90$		$n = 110$		$n = 130$	
Algorithms	Aver.	Std. Dev.	Aver.	Std. Dev.	Aver.	Std. Dev.	Aver.	Std. Dev.	Aver.	Std. Dev.
GREEDY	3.67	0.32	3.97	0.17	3.85	0.39	3.80	0.29	3.65	0.17
DLR	9.71	1.42	7.97	4.00	6.93	3.36	5.52	1.31	4.71	0.41
PGR	3.83	0.44	4.40	0.20	4.25	0.31	4.51	0.41	4.40	0.39

Table 4. The average packet delivery rate in 3D space.

	$n = 50$		$n = 70$		$n = 90$		$n = 110$		$n = 130$	
Algorithms	Aver.	Std. Dev.	Aver.	Std. Dev.	Aver.	Std. Dev.	Aver.	Std. Dev.	Aver.	Std. Dev.
GREEDY	58.70	8.90	81.10	7.95	87.50	5.08	94.70	5.93	98.10	1.60
DLR	71.40	13.05	91.70	10.12	94.60	4.95	99.60	1.26	99.90	0.32
PGR	65.90	7.19	82.80	6.92	89.40	7.85	94.90	5.45	98.60	1.17

Table 5. The average network lifetime in 3D space.

	$n = 50$		$n = 70$		$n = 90$		$n = 110$		$n = 130$	
Algorithms	Aver.	Std. Dev.	Aver.	Std. Dev.	Aver.	Std. Dev.	Aver.	Std. Dev.	Aver.	Std. Dev.
GREEDY	55.00	13.66	66.00	20.26	93.10	20.15	114.50	26.85	115.00	31.66
DLR	13.30	7.94	17.20	8.13	22.70	13.73	27.60	16.97	41.30	20.62
PGR	42.90	8.84	62.00	21.07	69.10	34.42	114.20	27.78	164.40	80.76

Table 6. The average number of hops in 3D space.

	$n = 50$		$n = 70$		$n = 90$		$n = 110$		$n = 130$	
Algorithms	Aver.	Std. Dev.	Aver.	Std. Dev.	Aver.	Std. Dev.	Aver.	Std. Dev.	Aver.	Std. Dev.
GREEDY	3.25	0.27	3.57	0.20	3.58	0.23	3.42	0.20	3.42	0.15
DLR	9.24	1.26	11.07	0.84	9.65	1.39	8.87	0.74	7.44	0.99
PGR	3.99	0.25	4.13	0.41	4.52	0.64	4.02	0.58	3.81	0.25

length, the performance of the three protocols is also similar to their 2D variants.

6. FUTURE WORK

The primary drawback of DLR is that it does not balance energy consumption well when the initial $\pi/3$ sector is empty. In that case, the protocol favors a neighbor with the smaller angle, without much consideration for its residual energy. For example, if two nodes have the same amount of residual energy, DLR will favor the node with the smaller angle. Then, if the nodes offer (almost) identical angle but different amounts of energy, DLR will still prefer the node offering the smaller angle. This strategy does not help increase network lifetime.

An alternative way of assigning the weight could be to multiply E_{res} by a factor α, which could be related to the angle. Similarly, it may be possible to multiply θ_x by another factor β that would be related to the energy. One more possibility is to consider formulas like $E_{res} \times \cos(\theta_x)$. We are planning to investigate the performance of DLR using different variations on these themes with the intention of improving the network lifetime and reducing the average path length without sacrificing the high delivery rate.

In [12], it is shown that energy consumption can be reduced by routing over network subgraphs, notably over Gabriel graphs. It will be interesting to measure the performance of all three protocols on Gabriel graphs.

We are also planning to verify the performance of the three protocols with a more realistic antenna model. Namely, with directional antennas it is desirable to avoid the dependence on the position information for measuring the angular position of the neighbors. It is suggested in [25] that relying on the position information may occasionally lead to confusion because of obstacles between nodes. Instead, the direction of arrival for the incoming signal may be used as a better indicator of the angle at which the sender can be reached. It might be interesting to try this idea in the context of the three protocols.

7. CONCLUSIONS

Due to the central importance of energy constraints in routing protocols for mobile ad hoc networks, it is desirable that any other routing objective be seen from an energy consumption perspective. In the particular case of location–based routing, its synthesis with energy considerations can be achieved by ensuring that the selection of the next–hop node is made on the basis of a combination of energy and direction considerations. We have presented DLR, a protocol that attempts a synthesis of the two factors. Moreover DLR does not easily “give up” forwarding packets, in that it is willing to divert packets away from the path to the destination and play the odds that at some later time the packet will be eventually pushed in the right direction.

On the other hand, location–based routing naturally begs the use of directional antennas and we have done so for DLR and PGR. While the study provided here may not be conclusive in terms of what is the best way to synthesize energy and direction costs it points at promising alternatives. For one thing, the results indicate that DLR, as well as the existing schemes, greedy and PGR, each achieve a distinctly different performance, improving network lifetime, improving packet delivery rate and decreasing path lengths respectively. Thus, a comprehensive study of the involved trade-offs is likely to bring about a reasonable compromise.

REFERENCES

[1] P. Bose, P. Morin, I. Stojmenovic, and J. Urrutia. Routing with guaranteed delivery in ad hoc wireless networks. In *3rd Workshop on Discrete Algorithms and Methods for Mobile Computing and Communication (DIALM)*, 1999.

[2] T-W. Chen and M. Gerla. Global state routing: a new routing scheme for ad-hoc wireless networks. In *Proceedings of ICC'98*, June 1998.

[3] R.R. Choudhury and N. Vaidya. Impact of directional antennas on ad hoc routing. In *Eighth International Conference on Personal Wireless Communication (PWC)*, Venice, September 2003.

[4] T. Fevens, I.T. Haque, and L. Narayanan. A class of randomized routing algorithms in mobile ad hoc networks. In *AlgorithmS for Wireless and mobile Networks (A_SWAN 2004)*, Boston, MA, August 2004.

[5] G.G. Finn. Routing and addressing problems in large metropolitan-scale internetworks. Technical Report ISU/RR-87-180, USC ISI, Marina del Ray, CA, March 1987.

[6] M. Gunes, U. Sorges, and I. Bouazizi. ARA—the ant-colony based routing algorithm for manets. In *Proceedings of International Workshop on Ad-hoc Networking (IWAHN)*, Vancouver, British Columbia, Canada, August 2002.

[7] D. B. Johnson and D. A. Maltz. Dynamic Source Routing in ad hoc wireless networks. In Imielinski and Korth, editors, *Mobile Computing*, volume 353. Kluwer Academic Publishers, 1996.

[8] B. Karp and H. Kung. GPSR: greedy perimeter stateless routing for wireless networks. In *Proc. of 6th ACM Conference on Mobile Computing and Networking (Mobicom '00)*, 2000.

[9] E. Kranakis, H. Singh, and J. Urrutia. Compass routing on geometric networks. In *Canadian Conference on Computational Geometry (CCCG '99)*, pages 51–54, 1999.

[10] J. Kuruvila, A. Nayak, and I. Stojmenovic. Progress based localized power and cost aware routing algorithms for ad hoc and sensor wireless networks. In *Third Int. Conf. on AD-HOC Networks and Wireless ADHOC-NOW*, pages 294–299, Vancouver, BC, July 2004.

[11] J. Li, J. Jannotti, D. De Couto, D. Karger, and R. Morris. A scalable location service for geographic ad hoc routing. In *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM' 00)*, pages 120–130, 2000.

[12] X.-Y. Li, P.-J. Wan, and Y. Wang. Power efficient and sparse spanner for wireless ad hoc networks. In *IEEE Int. Conf. on Computer Communications and Networks (ICCCN01)*, pages 564–567, 2001.

[13] G. Lukachan and M.A. Labrador. Selar: Scalable energy-efficient location aided routing protocol for wireless sensor networks. In *IEEE international Conference on Local Computer Networks (LCN'04)*, 2004.

[14] S. Murthy and J. J. Garcia-Luna-Aceves. An efficient routing protocol for wireless networks. *ACM Mobile Networks and Applications Journal*, pages 183–197, October 1996.

[15] V.D. Park and M.S. Cors, on. A performance comparison of TORA and ideal link state routing. In *Proceedings of IEEE Symposium on Computers and Communications '98*, June 1998.

[16] C. Perkins, E. Belding Royer, and S. Das. Ad-hoc On-demand Distance Vector Routing (AODV), February 2003. Internet Draft: draft-ietf-manet-aodv-13.txt.

[17] C.E. Perkins and P. Bhagwat. Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. In *ACM SIGCOMM '94 Conference on Communications Architectures, Protocols and Applications*, pages 234–244, August 1994.

[18] A. Rahman, W. Olesinski, and P. Gburzynski. Controlled flooding in wireless ad-hoc networks. In *Proceedings of IWWAN'04*, Oulu, Finland, June 2004.

[19] T. Roosta, M. Menzo, and S. Sastry. Probabilistic geographic routing in ad hoc and sensor networks. In *Int. Workshop on wireless Ad-Hoc Networks*, 2005.

[20] S. Roy, D. Saha, S. Bandyopadhyay, T. Ueda, and S. Tanaka. A power-efficient mac protocol with two-level transmit power control in ad hoc network using directional antenna. In *5th International Workshop on Distributed Computing (IWDC 2003)*, Calcutta, India, December 2003.

[21] S. Singh, M. Woo, and C.H. Raghabendra. Power aware routing in mobile ad hoc networks. In *Mobile Computing (MOBICOM)*, 1998.

[22] I. Stojmenovic and X. Lin. Power aware localized routing in ad hoc networks. *IEEE Transactions on Parallel and Distributed Systems*, 12(10):1023–1032, October 2001.

[23] The VINT project. The ucb/lbnl/vint network simulator ns (version 2). http://mash.cs.berkeley.edu/ns.

[24] C-K Toh. A novel distributed routing protocol to support ad-hoc mobile computing. In *Proceedings of IEEE 15th Annual International Phoenix Conf. on Comp. and Comm.*, pages 480–486, March 1996.

[25] R. Vilzmann and C. Bettstetter. A survey on mac protocols for ad hoc networks with directional antennas. In *EUNICE Open European Summer School*, Spain, July 2005.