Distributed and Compact Routing Using Spatial Distributions in Wireless Sensor Networks

RIK SARKAR, Free University of Berlin
XIANJIN ZHU, Microsoft Inc
JIE GAO, Stony Brook University

In traditional routing, the routing tables store shortest paths to all other destinations and have size linear in the size of the network, which is not scalable for resource constrained networks such as wireless sensor networks. In this paper we show that by storing selectively a much smaller set of routing paths in the routing tables one can get low-stretch, compact routing schemes.

Our routing scheme includes an approximate distance oracle with which one can obtain approximate shortest path length estimates to destinations. This distance oracle can be obtained, for example, by a landmark based scheme, or in case of sensor networks, from the geographic distance between node locations. With an approximate distance oracle one can attempt *greedy routing* by forwarding to the neighbor whose estimate is closer to the destination. But there is no guarantee of delivery nor of the routing path length. We augment the distance oracle by storing, for each node u, routing paths to $O(\log^2 n)$ strategically selected nodes that serve as intermediate destinations. These nodes are selected with probability proportional to $1/r^\rho$ where r is the distance to u and ρ is a suitable constant for the network. Then we derive a set of sufficient conditions to select the next step at each stage of routing, such that these conditions can be verified locally and guarantee $1+\varepsilon$ stretch routing on any metric. These conditions serve as the 'greedy routing' or local decision rule.

On graphs of bounded growth, our scheme guarantees $1+\varepsilon$ stretch routing with high probability, with an average routing table size of $O(\sqrt{n}\log^2 n)$. This scheme is favorable for its simplicity, generality and blindness to any global state. It demonstrates that global routing properties could emerge from purely distributed and uncoordinated routing table design.

Categories and Subject Descriptors: C.2.2 [**Computer-Communication Networks**]: Network protocols—*Routing protocols*; F.2.2 [**Analysis of Algorithms and Problem Complexity**]: Nonnumerical Algorithms and Problems—*Geometrical problems and computations*

General Terms: Algorithms, Design, Theory

Additional Key Words and Phrases: Sensor Networks, Small Stretch Routing, Spatial Distribution

1. INTRODUCTION

Scalable routing is one of the most challenging problems in distributed network design — considerations include compact storage, efficient propagation of topology update, and most importantly, distributed and uncoordinate decisions to enable globally close to optimal routing properties.

This work is partially supported by the National Science Foundation, under grant CNS-0643587, grant CNS-1016829 and grant CNS-1116881, and by the German Research Foundation (DFG) through the research training group Methods for Discrete Structures (GRK 1408).
Author's address: R. Sarkar is with the Institut Für Informatik, Freie Universität Berlin, Germany. Email: sarkar@inf.fu-berlin.de. X. Zhu is with Microsoft Corporation, One Microsoft Way, Redmond, WA 98052. Email: xianjin@gmail.com. J. Gao is with Department of Computer Science, Stony Brook University, Stony Brook, NY, 11794. Email: jgao@cs.sunysb.edu.
A 5-page preliminary version appeared in the Proceedings of the IEEE International Conference on Computer Communications (INFOCOM 2009) mini-conference. April 2009.

DOI 10.1145/0000000.0000000 http://doi.acm.org/10.1145/0000000.0000000

Consider the most basic routing table approach. Each node keeps a routing table entry for each possible destination. Following the routing table one can get shortest path routing. But the size of the routing table is linear in the size of the network and the scheme is not scalable to large networks.

Internet routing adopts the basic routing table approach but achieves scalability through sub-network partitioning hierarchy and address aggregation, with one routing table entry representing routing information to many IP addresses in a subnetwork, and powerful switches to quickly classify and deliver packets. For resource constrained wireless nodes used in sensor networks, scalable routing requires even more aggressive methods to produce compact routing information, and innovative ways to exploit the special properties of such networks.

Large-scale wireless sensor networks have a lot of spatial structures — they are closely related with the underlying geometric domain in which they are embedded, in terms of node distribution and the strong correlation of graph connectivity and node proximity. Various properties of the geographical embedding of the nodes have been exploited for compact routing in a sensor network — mostly in an explicit manner, as the geographical locations used in geographical routing families [Bose et al. 2001; Karp and Kung 2000; Kuhn et al. 2003], or as in many virtual coordinate system design [Fang et al. 2005; Bruck et al. 2007] that abstracts the global geometric/topological properties of the embedding.

1.1. Overview

In this paper we use some implicit geometric properties of a wireless sensor network for routing, and store selective routing paths in the routing tables, such that the average routing table size is small, the path stretch is close to optimal (the ratio to the shortest path is $1+\varepsilon$ for any given $\varepsilon > 0$), and both the preprocessing and the routing can be achieved by the nodes making decisions on their own, blind to any global state.

Our solution has two components: an *approximate distance oracle* and a set of *augmented routing paths*. We describe the two components separately.

Approximate distance oracle. An approximate distance oracle gives an estimation of the shortest path length (i.e., minimum hop count) between any two nodes. That is, for nodes p and q at a true distance of $\sigma(p,q)$ in the metric, the estimate $d(p,q)$ supplied by the oracle satisfies the relation $\delta_1 \cdot d(p,q) \leq \sigma(p,q) \leq \delta_2 \cdot d(p,q)$ for some constants δ_1 and δ_2.

First we remark that if we are given an *accurate* distance oracle that returns the hop distance of any two nodes in the network, then greedily selecting the next hop as the neighbor with smallest distance to the destination will guarantee delivery along a shortest path. Of course, the construction, maintenance and compact representation of an accurate distance oracle is not easy in a distributed setting. As shown in [Thorup and Zwick 2001a], accurate distance oracle would require about $\Omega(n)$ storage per node. An approximate distance oracle is easier to obtain. In many cases, some approximate distance estimation is readily available. For example, the geographical distance is often a good approximation to the minimum hop count distance in the network. We will show later that for a sensor network with near uniform node distribution and when the network holes are 'fat'[1], the Euclidean distance is a good approximate distance oracle. Note that this assumption makes no unit disk graph requirement on the wireless radio communication model and allows localization errors as well. When node location information is not available, one can show that with randomly selected nodes as landmarks and using merely triangle inequality on the hop count distances from source and destination to these landmarks (as shown in [Kleinberg et al. 2004]) one can get a fairly good approximate distance oracle as well. Both implementations require only information of the source node and the destination.

With only an approximate oracle we can still try a greedy routing method — forward the message to the neighbor whose approximate distance to the destination from the oracle is smaller — but there is no guarantee that such a neighbor can be found. Indeed, this is the major problem that geographi-

[1]We define a hole to be *fat* if any two nodes on the boundary of a hole has its hop count distance to be at most a constant factor of the Euclidean distance. For example, a square is fat, but an arbitrarily thin rectangle is not

cal greedy forwarding can get stuck at a local minimum. Thus we need our second component in the routing scheme that complements the approximate distance oracle, which is our main contribution.

Augmented routing paths. With an approximate distance oracle, the solution we propose is to store routing paths between some pairs of nodes that are not immediate neighbors. We call these paths *long links*. In particular, for some selected pair u, v a route $P(u, v)$ between u, v, is recorded implicitly in the routing table entries of nodes on the route. When a node p wants to send a message to a node q, it also considers the nodes to which p has long links – these act as a generalization of the neighbor set in the network. The routing information stored on the path $P(p, x)$ is used to deliver the message to x. Node x then repeats an identical procedure to advance the message. Now the question is, what long links should each node build and what are the forwarding regions, such that the routing table size is small, the path stretch is low, and delivery rate is high?

Our main theoretical results are the following. Given an approximate oracle for a general metric space, we come up with a simple local rule such that the source node p can decide which long link neighbor is good for forwarding the message. All such nodes are conveniently characterized by the *forwarding region* (see Fig. 1), from which p selects the long link neighbor in the path to q. Next we show a randomized method for building the long links and the resultant routing tables. This will guarantee that long links satisfying the required conditions, for any potential destination, exist with a high probability. In particular, each node selects its long links with a spatial distribution. A node p would select a long link partner q with probability proportional to $1/d(p, q)^\rho$, where $d(p, q)$ is the approximate distance between p, q returned by the oracle. The number of long links for each node is $O(\log^2 n)$ with the constant depending on the stretch requirement $1 + \varepsilon$ and the distance oracle error bounds δ_1, δ_2.

This distribution guarantees that on a graph of bounded growth rate, we will have long links satisfying the required conditions to perform low stretch routing with high probability. A graph has bounded growth rate ρ if the number of nodes within r hops from any node p in the network is bounded by $c_1 r^\rho$ and $c_2 r^\rho$ from below and above respectively, with two constants $c_1 \leq c_2$. This model has been used to capture any physical constraints that disallow too many nodes 'packed' within certain distance and the graph has a bounded polynomial growth pattern instead of an exponential growth pattern (e.g., a balanced binary tree). This kind of geometric growth has been observed in many different scenarios such as VLSI design, the delay metric on the Internet overlay networks, and in our setting, wireless sensor networks. When sensor nodes are roughly uniformly deployed in a geometric region with bounded density per unit area[2] and when the network is not too much fragmented by deployment holes, the graph growth rate is typically 2. It is this packing property that allows us to aggressively compress the routing table entries by a simple routing table neighbor selection rule dominated by a spatial distribution.

We also report simulation evaluations of this approach in a sensor network setting, to complement the theoretical analysis. For a connectivity network in which geographical greedy routing only achieves a delivery rate of 50% or so, with about 7 long links per node, we are able to achieve a delivery rate of 99% or higher. The routing table construction can be implemented in a completely distributed manner. Each node simply chooses its respective long links by sampling geographical locations under the spatial distribution, rounded to the nodes closest to the sampled locations, as in [Sarkar et al. 2007]. The routing table information for these long links is constructed in a bootstrapping manner, with the routes for nearby pairs constructed first and the routes for far away pairs constructed by using the routing tables already constructed so far, in the same manner as regular routing requests.

We have a second implementation using landmark-based routing to show the power of the spatial distribution in routing table design. In particular, we select $O(\log^2 n)$ landmarks that flood the entire network and each node records the landmark distance vector. The approximate distance oracle

[2]If the density in a region becomes too high, it is easy to cluster neighboring nodes and operate on clusterheads so that the density of clusterheads is bounded by a constant.

is implemented by the centered virtual distance as proposed in [Fang et al. 2005], which only requires the landmark distance vector of two nodes. We select on the paths to the landmarks long link neighbors to help improve the delivery rate. This implementation will involve some preprocessing of flooding the network from the landmarks but the routing paths of the long links are implicitly contained in the landmark distances. Thus the routing table size is improved to $O(\log^4 n)$, compared with $O(n^{1/\rho} \log^2 n)$ when the routes have to be explicitly stored on the nodes of the paths (ρ is the growth rate – a constant similar to dimension of the network).

In summary, the augmentation of long links with spatial distribution to get $1 + \varepsilon$ stretch routing on an approximate distance oracle is favorable for its simplicity, generality and 'blindness' to any global state. Global routing properties emerge from purely distributed and uncoordinated routing table design.

2. RELATED WORK

In this section we survey related work in compact routing and establish their connection to our results.

Spatial distribution in routing. The spatial distribution in selecting the long links in our paper coincides with the small-world model and decentralized search proposed by Kleinberg [Kleinberg 2000b; 2000a] to model Stanley Milgram's famous experiment [Milgram 1967; Travers and Milgram 1969] on the small-world phenomena in social networks. The setup in the small world model is the following. Given a 2D grid (possibly of infinite size), each node chooses a long link with probability $1/r^2$ where r is the length of the long link. Together with the four neighbors per node on the grid, a greedy routing with the location of the destination can be achieved with $O(\log^2 n)$ jumps (on either short links between neighbors on the grid or the long links constructed) with high probability. Notice that in this setting an accurate distance oracle is actually available and greedy routing on the original grid suffices to deliver the message along the shortest paths on the grid. In the small world literature people care most about adding extra long links to create short paths between any two nodes. In our setting the long links are realized as paths in the original network. Nevertheless, our results show that if each node chooses $O(\log n)$ long links, a slightly more sophisticated but distributed routing scheme with long links has $O(\log n)$ jumps, and also a total travel distance at most $1 + \varepsilon$ of the distance between source and destination on the grid.

The spatial distribution has been explored in a number of other data delivery and information dissemination scenarios in sensor networks, e.g., for adding long communication wires to reduce power consumption [Sharma and Mazumdar 2005], for gossip and locality-sensitive information exchange [Kempe et al. 2001; Sarkar et al. 2007], for data delivery using mobile nodes [Wu and Yang 2008].

Small state routing in sensor networks. To deal with the problem of local minimum in geographical forwarding, various techniques have been proposed to solve the problem of 'routing around holes'. Earlier proposals assume unit disk graph model on the communication network and propose to planarize the network and apply face routing [Bose et al. 2001; Karp and Kung 2000; Kuhn et al. 2003]. Such planarization unfortunately fails badly in practice due to complex radio characteristics [Kim et al. 2005]. Improvement of the planarization process may selectively remove crossing edges [Govindan et al. 2006], or use a generalized face routing on graphs with crossing edges [Zhang et al. 2007], or planarize an abstracted graph to filter out the local connectivity irregularity [Funke and Milosavljević 2007]. Alternatively, one may also develop virtual coordinates to support greedy routing [Rao et al. 2003; Newsome and Song 2003; Fonesca et al. 2005; Nguyen et al. 2007; Fang et al. 2005; Bruck et al. 2007]. Most of them do not guarantee small stretch routing and often require preprocessing to first discover and understand the network topologies.

We explain two protocols in more details as they are more relevant and compare with our scheme. In virtual ring routing (VRR) [Caesar et al. 2006], proposed by Caesar *et al.*, the nodes are ordered by their node IDs (or any other identifiers) on a ring and the paths for nearby nodes on the ring are stored in the routing tables of the nodes on these paths. Notice that nearby nodes on the ring may be far away in the communication network. When a packet is routed to a destination, it is delivered by using the local routing table to the next hop on the pre-constructed path leading to a node closest to the destination in the ID space. VRR can be understood as building long links connecting nodes with adjacent IDs, which can be arbitrarily far apart in the network. The routing table size is roughly in the order of $O(\sqrt{n})$ in a uniform and dense network. And there is no guarantee on the path stretch.

The small state and small stretch (S4) routing by Mao *et al.* [Mao et al. 2007] adopted the idea of compact routing schemes by Thorup and Zwick [Thorup and Zwick 2001a; 2001b]. The basic idea is to select about $O(\sqrt{n})$ landmarks. These landmarks flood the network and other nodes record the hop count distance to these landmarks. In addition, a node p also maintains routing table entries to the nodes that are closer to p than their closest landmarks. The routing table size is about $O(\sqrt{n})$ and a greedy routing scheme is guaranteed to deliver the message to the destination with maximum stretch of 3. By exploiting the geometric properties of the sensor network deployment, we are able to get $1+\varepsilon$ stretch and reduce both the number of landmarks and the routing table size to polylogarithmic in the network size.

Compact routing in general. From a theoretical aspect, compact routing that minimizes the routing table size while achieving low stretch routing has been studied extensively [Peleg 2000]. There are two popular models in the literature, the *labeled routing model* and *name-independent routing*. In the labeled routing model [Cowen 1999; Eilam et al. 2003; Thorup and Zwick 2001b], one is allowed to produce for each node a label (typically of polylogarithmic size) such that routing is done with the labels of the source and destination. In the name-independent model [Abraham and Malkhi 2005; Konjevod et al. 2006], the nodes are given generic IDs that are independent of the routing scheme. Thus routing is inherently more difficult as the routing scheme needs to also find out where the node is. To understand this in the case of sensor network routing, name-independent routing works directly on the node IDs (such as in the virtual ring routing scheme). If we use geographical locations or any other virtual coordinates, such coordinates are the 'labels' and to complete the solution one needs to also employ a location service (as in [Li et al. 2000]) that maps node IDs to their geographical locations or virtual coordinates. Put in this perspective, our scheme stays in between the labeled model and the name-independent routing model. We have a label of the nodes (such as the geographical locations) naturally, but the labels only give imperfect distance information and do not guarantee delivery.

Generally speaking, the theoretical results in compact routing in a graph whose shortest path metric has a constant doubling dimension are able to obtain, with polylogarithmic routing table size, $1+\varepsilon$ stretch routing in the labeled routing scheme (see [Chan et al. 2005] and many others in the reference therein), and constant stretch factor routing in the name-independent routing scheme [Konjevod et al. 2006; Abraham et al. 2006] (getting a stretch factor of $3-\varepsilon$ will require linear routing table size [Abraham et al. 2006]). The results here are all centralized constructions and aim to get the best asymptotic bounds. Our focus in this paper is on a principle for distributed implementation at each node and its practical implementation in the scenario of ad-hoc sensor network routing.

There has been a lot of work on constructing overlay graphs on nodes staying in a metric space (to name a few, as in [Plaxton et al. 1997; Abraham et al. 2004]). We do not survey those work in detail as in our case we are not given the perfect knowledge of the metric and we can not construct communication links between any nodes.

3. SMALL STRETCH ROUTING WITH APPROXIMATE DISTANCES

In this section we describe the idea of routing with $1+\varepsilon$ stretch in a suitable metric space $\mathcal{M}$. We use $d(p,q)$ to represent the estimate of distance between p and q supplied by the approximate oracle $\mathcal{O}$, and $\sigma(p,q)$ to denote the true but possibly unknown graph distance (hop count distance) in $\mathcal{M}$. We assume that a node is able to get the approximate distance $d(p,q)$ from just the names of p,q. The implementation of this distance oracle is elaborated in a later section. Here we show that when the long links are carefully chosen the routing stretch is low.

Routing with accurate distance oracle. To demonstrate the basic concept, we first consider the case in which the oracle is in fact accurate, that is, $d = \sigma$. The objective is to recursively build a route from s to t with the help of the long links. Suppose s takes a long link to node p, then we want $\sigma(s,p) + \sigma(p,t)$ to be not very large compared to $\sigma(s,t)$:

$$\sigma(s,p) + \sigma(p,t) \leq \gamma \cdot \sigma(s,t), \tag{1}$$

Where $\gamma \geq 1$ is a parameter depending on ε. Observe that inequality (1) defines an ellipse in $\mathbb{R}^2$ with s and t at foci. Now we impose an additional restriction that moving from s to p implies a certain progress in direction of t. In particular, p is closer to t by a factor of at least $0 \leq \beta \leq 1$:

$$\sigma(p,t) \leq \beta \cdot \sigma(s,t). \tag{2}$$

This describes a disk centered at t.

Next, we select γ and β such that the selection procedure enforced by inequalities (1) and (2) when applied recursively, produces a path of stretch at most $1+\varepsilon$:

$$R(s,t) \leq (1+\varepsilon) \cdot \sigma(s,t), \tag{3}$$

where R gives the length of the path created recursively.

A *forwarding region* $F_\varepsilon(s,t)$ is a set of points p in $\mathcal{M}$ from which s can select p satisfying the above relations. The following lemma gives a detailed description:

LEMMA 3.1. *Values of γ and β satisfying $\gamma + \varepsilon\beta \leq 1 + \varepsilon$ constitute the forwarding region, with the equality corresponding to the region boundary.*

PROOF. Given that in the route from q to t, the first long link is to a node p, the total length of the recursive path $R(q,t) = \sigma(q,p) + R(p,t)$. Let us assume that routes have already been built such that $R(p,t) \leq (1+\varepsilon)\sigma(p,t)$. Then we have:

$$\begin{aligned} R(q,t) &\leq \sigma(q,p) + R(p,t) \\ &\leq \sigma(q,p) + (1+\varepsilon) \cdot \sigma(p,t) \\ &\leq \sigma(q,p) + \sigma(p,t) + \varepsilon\sigma(p,t) \\ &\leq \gamma\sigma(q,t) + \varepsilon\beta\sigma(q,t). \end{aligned}$$

When $\gamma+\varepsilon\beta \leq 1+\varepsilon$, the right hand side is no greater than $(1+\varepsilon)\cdot\sigma(q,t)$. An inductive application of this argument shows a $(1+\varepsilon)$ stretch for any route $R(s,t)$. □

It is easy to see that γ must lie in the interval $[1, \frac{2+3\varepsilon}{2+\varepsilon}]$ for a given ε. For each value of γ, we have a region $H_{\gamma,\varepsilon}(s,t) \subseteq \mathcal{M}$ which is the intersection of the ellipse bounded region and the disk. Thus, formally, the forwarding region is the union: $F_\varepsilon(s,t) = \cup_\gamma H_{\gamma,\varepsilon}(s,t)$. See Figure 1.

Routing with approximate distance oracle. Now we look at the case in which the oracle supplies an approximate measure of the distance, with δ_1 and δ_2 as the lower and upper bounds: $\forall p,q \in \mathcal{M}$, $\delta_1 d(p,q) \leq \sigma(p,q) \leq \delta_2 d(p,q)$. Then, allowing for approximation error, it would be sufficient to guarantee the following inequalities (corresponding to relations (1)-(2) respectively,):

$$\begin{aligned} &\delta_2 d(s,p) + \delta_2 d(p,t) \leq \gamma\delta_1 d(s,t) \\ &\delta_2 d(p,t) \leq \beta\delta_1 d(s,t) \end{aligned} \tag{4}$$

It can be verified that a sufficient relation between γ, β and ε is again given by the same inequality as lemma 3.1. And we can obtain again that $R(s,t) \leq (1+\varepsilon)\sigma(s,t)$.

As long as a node has a long link p in the forwarding region, the routing idea described above guarantees low stretch for any metric.

Routing Mechanism. The analysis above suggests a natural routing scheme. Each node selects long links such that it has either an immediate neighbor or a long link to the forwarding region of any destination, and keeps corresponding routing table entries. The routes to the long link neighbors are stored on the routing table of the nodes on the path. When a node s has a message to be delivered to a destination t, s will check its routing table to find a node p (either s's 1-hop neighbor or s's long link neighbor,) such that p lies in the forwarding region $F_\varepsilon(s,t)$. Node p on receiving the message will execute an identical procedure to forward the message into $F_\varepsilon(p,t)$ and so on. Efficient randomized construction of the routing table is shown in next section.

3.1. $(1+\varepsilon)$-stretch forwarding region

Geometric setting. We first discuss the case of the Euclidean plane $\mathbb{R}^2$, which provides intuition about the metric properties of the method. W.l.o.g. the coordinates of s and t, separated by a distance r, are $(-r/2,0)$ and $(r/2,0)$ respectively. We examine the forwarding region to select the long link neighbor p to realize a $1+\varepsilon$ stretch path to t.

With an accurate distance oracle, the relation (1) defines in $\mathbb{R}^2$ a region whose boundary is given by an ellipse:

$$\frac{4x^2}{\gamma^2 r^2} + \frac{4y^2}{r^2(\gamma^2-1)} = 1.$$

And (2) defines a disk whose boundary is given by a circle:

$$\left(x-\frac{r}{2}\right)^2 + y^2 = \frac{(1+\varepsilon-\gamma)^2}{\varepsilon^2} r^2.$$

As gamma is varied, the locus of intersection of these two curves traces out the boundary of the forwarding region $F_\varepsilon(s,t)$ (see Fig. 1 (i)).

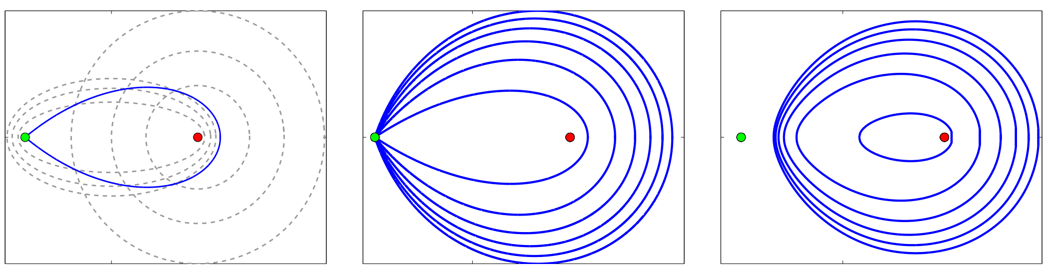

Fig. 1. (i) Boundary of F_ε as intersection of ellipses and circles. (ii) Forwarding regions for different values of ε from 0.2 to 2. (iii) Forwarding regions for different values of ε from 0.2 to 2 for approximate oracle.

For any point q on the boundary of $F_\varepsilon(s,t)$, the angles $\angle qst$ and $\angle qts$ are functions of γ and ε only, and are independent of r. This implies that the shape of the forwarding region is scale invariant, i.e., it does not depend on the distance between source and destination. Figure 1 (ii) shows the shapes of forwarding regions for different values of ε. Smaller values of ε create smaller and narrower forwarding regions.

With an approximate distance oracle, the corresponding ellipse and circle equations are given by:

$$\frac{\delta_2^2}{\delta_1^2}\cdot\frac{4x^2}{\gamma^2 r^2} \;+\; \frac{4y^2}{r^2\left(\frac{\delta_1^2}{\delta_2^2}\gamma^2-1\right)} = 1$$

$$\left(x-\delta_2\frac{r}{2}\right)^2 \;+\; y^2 = \left(\frac{\delta_1}{\delta_2}\cdot\frac{1+\varepsilon-\gamma}{\varepsilon}\cdot r\right)^2$$

The corresponding forwarding regions are shown in Fig. 1 (iii). Observe that in this case the forwarding regions are smaller and source s is not in the forwarding region. This is due inaccurate distance estimates and necessitates the use of *long links* - without which s cannot access the forwarding region.

The graph setting. The geometric intuition needs to be realized in an ad hoc sensor network setting. In the literature, there have been a number of models for graphs that have some geometric growth features. In the following description we focus on an undirected unweighted graph G and we denote by $N_r(p)$ the set of nodes within r hops from p. A graph is said to have Δ-*expansion rate* if $|N_{2r}(p)| \leq \Delta |N_r(p)|$, for any p, r [Karger and Ruhl 2002; Abraham and Malkhi 2005]. A graph is said to have *doubling dimension* Δ if any ball of radius $2r$ can be covered by at most 2^Δ balls of radius r [Gupta et al. 2003]. A graph is said to have *bounded growth rate* Δ if $|N_r(p)| = O(r^\Delta)$ [Linial et al. 1995]. All three models try to capture that the metric growth is restrictive. For example, a binary tree does not satisfy any of the definitions above.

In this section, we use the concept of a finite graph and a continuous metric space interchangeably for ease of description, but the results hold for any metric space that fits the model. A graph metric refers to the shortest path metric.

In a sensor network setting, we use the (upper and lower) bounded growth rate model, as it follows immediately from a bounded density deployment. For example, if we place at most a constant number of sensor nodes inside any unit disk and the holes in the sensor networks are not very fragmenting, the number of nodes at k hops from a node p will be around $\Theta(k)$.

Formally, we consider a graph such that the number of nodes at a distance exactly r from p, represented by $|\partial N_r(p)|$ is bounded by $|\partial N_r(p)| = \Theta(\rho r^{\rho-1})$. This is equivalent to $|N_r(p)| = \Theta(r^\rho)$. Note that the diameter D of such a graph is bounded by $\Theta(n^{1/\rho})$. We have the following quick observation.

LEMMA 3.2. *Given an unweighted graph G with $|N_r(p)| = \Theta(r^\rho)$, the graph has a doubling dimension $\eta = O(\rho)$.*

PROOF. Consider a ball $B_{2r}(p)$, we use a greedy algorithm to select balls of radius r to cover it. In particular, we select a node q in $B_{2r}(p)$ that is not yet covered, and cover all nodes in $B_r(q)$. Iterate until all nodes are covered. Now we bound how many balls are selected (denote this set as Q). To see that, we take the selected nodes $q \in Q$ and the balls $B_{r/2}(q)$. First they do not overlap as any two nodes in Q are of distance at least r away. Thus by a volume argument we have $|Q| \leq |N_{2r}(p)| / \min(|N_{r/2}(q)|) = O(\frac{(2r)^\rho}{(r/2)^\rho}) = O(4^\rho)$. $\square$

LEMMA 3.3. *In a metric space with doubling dimension η, a ball of radius R can be covered with $O(c^\eta)$ balls of radius R/c.*

PROOF. A ball of radius R can be covered with 2^η balls of radius $R/2$. We recursively cover each such ball with balls of half the radius, until the size of balls used falls below R/c. The resultant number of balls is $2^{\eta k}$, where $k = \lceil \log c \rceil$. This is equivalent to $O(c^\eta)$. $\square$

We now show the presence of a sizeable forwarding region for such a graph, when one routes from s to t:

LEMMA 3.4. *There is a ball of radius $\frac{\delta_1}{\delta_2}\left(\frac{\gamma-1}{2}\right) r$ that lies inside $F_\varepsilon(s,t)$.*

PROOF. Consider a point q on the shortest path between s and t separated by $d(s,t) = r$. Now, we take a ball of radius $h = \frac{\delta_1}{\delta_2}\left(\frac{\gamma-1}{2}\right) r$ centered at q. One can verify that all the points inside the ball $N_h(q)$ are inside $F_\varepsilon(s,t)$, as they satisfy the inequalities (4). In particular, for any point $p \in N_h(q)$, $d(s,p) \leq d(s,q) + h$, $d(p,t) \leq d(q,t) + h$. Now we can verify that $\delta_2(d(s,p) + d(p,t)) \leq \delta_2(r + 2h) \leq \delta_1 \gamma r$. Also $\delta_2 d(p,t) \leq \delta_2(d(q,t) + h) \leq \delta_1 \beta r \leq \delta_1 \left(\frac{1+\varepsilon-\gamma}{\varepsilon}\right) r$.

This ball is inside a neighborhood of $\delta_2 r - \frac{\delta_1}{\delta_2}\left(\frac{1+\varepsilon-\gamma}{\varepsilon} - (\gamma - 1)\right) r$ from s. The number of nodes inside this ball is at least $\Omega\left(\left(\frac{\delta_1}{\delta_2}\left(\frac{\gamma-1}{2}\right) r\right)^\rho\right)$. $\square$

This lower bound on the size of forwarding region suggests that among long links chosen randomly according to a spatial distribution, at least one is likely to lie in the forwarding region with high probability. The next subsection shows that this is indeed the case.

4. ROUTING TABLE CONSTRUCTION BY SPATIAL DISTRIBUTION

To build the routing table, we use a spatial distribution of directed links. In particular, for nodes p and q separated by a distance r, the probability of a directed link pq being built is proportional to $1/r^\rho$. The rest of this section analyzes random selection of long links to make sure there is a long link in the forwarding region for every possible destination. Combined with the recursive routing in the beginning of this section, the existence of such links guarantee $1+\varepsilon$ stretch routing.

The analysis below uses essentially balls and bins probabilistic analysis. When a long link is picked randomly with the spatial distribution, we have the following lemma.

LEMMA 4.1. *For any $\mu > 1$, a link from p lies in the annulus $N_r(p) - N_{r/\mu}(p)$ with probability $\Theta\left(\frac{\ln \mu}{\ln n}\right)$.*

PROOF. Suppose C is the normalizing factor of the probability distribution for the given network. This means: $C \int_1^D \frac{1}{r^\rho} |\partial N_r(p)|\, dr = 1$. Integrating, $C = \Theta\left(\frac{1}{\rho \ln n}\right)$.

The probability that a given link lies in an annulus $N_r(p) - N_{r/\mu}(p)$ is given by

$$\Pr(r/\mu, r) = C \int_{r/\mu}^{r} \frac{1}{\xi^\rho} |\partial N_\xi(p)|\, d\xi = \Theta\left(\frac{\ln \mu}{\ln n}\right).$$

Note that this probability is independent of r. □

THEOREM 4.2. *From each node it is sufficient to select $k = O\left(\left(\frac{2}{\varepsilon}\right)^{O(\rho)} \ln^2 n\right)$ links, to guarantee a link in the forwarding region for any destination with probability at least $1 - 1/n^2$.*

PROOF. Consider the forwarding region $F_\varepsilon(s,t)$, with $d(s,t) = \ell$. We choose a valid value γ. By lemma 3.4, there is a ball B_h of radius $h' = \frac{\delta_1}{\delta_2} \frac{\gamma - 1}{2} \ell$ within a distance of $r = \delta_2 \ell - \frac{\delta_1}{\delta_2}\left(\frac{1+\varepsilon-\gamma}{\varepsilon} - (\gamma - 1)\right)\ell$ from s.

Choose μ' such that $B_{h'}$ lies in the annulus $N_r(s) - N_{r/\mu'}(s)$. This implies that $\mu' = \frac{r}{r - 2h'}$. Substituting, and simplifying, we have $\mu' = \Omega(1+\varepsilon)$. To show that a link lies in $B_{h'}$, it is sufficient to show that it lies in a smaller ball $B_h \subseteq B_{h'}$, which is defined below. If $h \geq r/4$ we assign $B_h = B_{r/4}$, and $\mu = 2$, where $B_{r/4} \subseteq B_h$, and $B_{r/4} \subseteq N_r(s) - N_{r/2}(s)$. If $h < r/4$, we assign: $B_h = B_{h'}$ and $\mu = \mu'$. Thus, the width of the annulus $N_r(s) - N_{r/\mu}(s)$ is at most $r/2$, and $\mu \leq 2$.

Now we show that with $k = O\left(\left(\frac{2}{\varepsilon}\right)^{O(\rho)} \ln^2 n\right)$ links, there is a link to B_h (and hence to $B_{h'}$) with high probability. The basic idea is the following. The annulus $N_r(s) - N_{r/\mu}(s)$ can be covered by a small number of balls, by the constant doubling dimension property. Thus with randomly selected links, at least one will fall inside B_h.

By Lemma 3.3, the ball $N_r(s)$ can be covered by at most $A = a\left(\frac{2\mu}{\mu - 1}\right)^\eta$ balls of radius h for some constant a. Restricting attention only to links from s to inside $N_r(s) - N_{r/\mu}(s)$, consider a covering of the annulus with balls of radius h. The ball B_h belongs to this set, and each node in B_h is selected by s with probability at least $C\frac{1}{r^\rho}$, where $C = \Theta(1/(\rho \ln n))$ is the normalizing factor. Similarly, every node in the other $A - 1$ balls is selected with a probability at most $C\frac{\mu^\rho}{r^\rho}$.

Thus, given that a link is in the annulus $N_r(s) - N_{r/\mu}(s)$ the probability that it is in B_h is:

$$\Pr(B_h | (N_r(s) - N_{r/\mu}(s))) \geq \frac{(\mu - 1)^\eta}{a(2\mu)^\eta \mu^\rho + (\mu - 1)^\eta}.$$

Combining with the result of lemma 4.1 of the link being in the annulus, we get that the probability of a random link to B_h is $\Pr(B_h) \geq \left(\frac{1}{K \ln n}\right)$, where $K = O\left(\left(\frac{2}{\varepsilon}\right)^{O(\rho)}\right)$.

If $2K \ln^2 n$ links are chosen from s, then the probability that none of them lie in B_h is $\left(1 - \frac{1}{K \ln n}\right)^{(K \ln n) 2 \ln n}$. Therefore, the probability that at least one link lies in B_h is $\left(1 - 1/n^2\right)$. Therefore, $O(\left(\frac{2}{\varepsilon}\right)^{O(\rho)} \ln^2 n)$ links per node suffice to obtain the given probability. □

The theorem above describes a guarantee for a suitable link to a forwarding region to exist. In fact, the detailed proof says that a link exists to a ball $B_{h'}$ of a radius h' inside the forwarding region. However, we still need to prove the existence of a path of $(1 + \varepsilon)$ stretch for a given routing request, that will take us to within a small constant distance of the destination. This is done by showing the existence of a short sequence of forwarding links. First we show, that if the path exists, it only involves a few long links.

LEMMA 4.3. *If a path obtained by appending the long links in the balls $B_{h'}$ exists then it consists of $O(\log n)$ long links and has a stretch of $(1 + \varepsilon)$.*

PROOF. As in the proof of theorem 4.2, there is a ball $B_{h'}$ of radius $h' = \frac{\delta_1}{\delta_2} \frac{\gamma' - 1}{2} l$ which by lemma 3.4 lies within a distance $\frac{\delta_1}{\delta_2} \frac{1 + \varepsilon - \gamma'}{\varepsilon} l = \frac{\delta_1}{\delta_2} \beta' l$ of t.

Thus, by selecting the long link to the ball $B_{h'}$, we take the message to be within a constant fraction β' of the remaining distance to the destination at every step. Since the diameter of the network is $n^{1/\rho}$, this recursive forwarding will reach a constant neighborhood of t using $O(\log n)$ hops. Given that $B_{h'}$ is selected to be inside the forwarding region for each step, this path will have a stretch $1 + \varepsilon$. □

Now we combine the number of links with the probability of each link to get the final result:

THEOREM 4.4. *Given a source-destination pair, a path of stretch $1 + \varepsilon$ exists with probability at least $1 - 1/n$ if $O\left(\left(\frac{2}{\varepsilon}\right)^{O(\rho)} \ln^2 n\right)$ long links have been created per node.*

PROOF. Observe that by lemma 4.3 the path consists of $O(\log n)$ long links, each of which exists with probability at least $1 - 1/n^2$, by theorem 4.2. Combining the two, we get that the path exists with probability $\left(1 - 1/n^2\right)^{O(\log n)}$, which is at least $1 - 1/n$. □

And the routing table size is not too large.

THEOREM 4.5. *The average routing table size of the scheme is bounded by $O\left(\left(\frac{2}{\varepsilon}\right)^{O(\rho)} n^{1/\rho} \ln^2 n\right)$.*

PROOF. The length of a long link is at most the diameter of the network, which is $O(n^{1/\rho})$. Thus a link can contribute at most $O(n^{1/\rho})$ number of routing tables entries. By theorem 4.2, each node of n nodes can add $O\left(\left(\frac{2}{\varepsilon}\right)^{O(\rho)} \ln^2 n\right)$ such links to the network. Thus, the average number of entries, when divided among n nodes, is $O\left(\left(\frac{2}{\varepsilon}\right)^{O(\rho)} n^{1/\rho} \ln^2 n\right)$. □

In the case of sensor networks in a plane ($\rho \approx 2$), for a given stretch ε, this amounts to a table size of $O\left(\sqrt{n} \ln^2 n\right)$ per node. In the next section we describe an implementation that implicitly stores the long links with substantially smaller routing table sizes of $O(\ln^4 n)$.

5. IMPLEMENTATION IN SENSOR NETWORKS

Here we describe the implementation of the routing table design in a distributed setting. In particular, how to implement the approximate distance oracle, how to choose the long links with the spatial distribution and how to build routes representing the long links. We give two different approaches to implement the distributed routing table, one with the geographical locations, one with landmarks and landmark-based distances.

Note that the implementation of approximate distance oracle is really independent of our routing table design and the implementations can be entirely decoupled. Any method that provides reasonably good distance estimate can be used as a distance oracle.

5.1. Geographic routing table design

In this part we describe using the spatial distribution principle to augment standard geographical forwarding with additional routing information to increase the delivery rate.

Approximate distance oracles. As mentioned in the introduction, the geographical locations often serve as a good approximate distance oracle to the minimum hop count distance metric on the communication network. To formulate this notion rigorously, we assume that the sensor field is deployed in an environment with *fat* (not necessarily convex) obstacles. That is, for any two points p, q on the boundary of a hole, the geodesic distance[3] $g(p, q)$ is at most τ times the Euclidean distance $d(p, q)$ for a constant $\tau > 1$, as shown in Figure 2. Given this, we can show

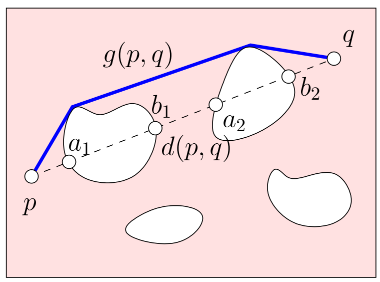

Fig. 2. The geodesic distance $g(p, q)$ is at most $\tau \cdot d(p, q)$ with fat holes.

that for any two points p, q in the underlying geometric domain, we have $g(p, q) \leq \tau d(p, q)$. In addition, we assume that the sensor nodes are deployed in the environment approximately uniformly such that the minimum hop count distance is at most τ' the geodesic distance. Thus we have $d(p, q) \leq \sigma(p, q) \leq \delta \cdot d(p, q)$, for a constant $\delta = \tau \cdot \tau' > 1$.

Geographic spatial sampling. We include the routing paths between pairs of nodes chosen with a spatial distribution. With geographical locations, we will implement the spatial sampling of a partner q of p by choosing with probability proportional to $1/r^2$ a *geographical location* q^* and round it to the nearest node q. That is, the node q whose Voronoi cell contains the sampled location q^* is taken as the long link partner of p. If the nodes are not uniformly distributed, the Voronoi cells have different areas and the nodes are selected with a biased probability. Thus we use von Neumann's rejection sampling to 'smooth out' the non-uniformity introduced by the variation of Voronoi cell area. This idea is originally proposed and used in taking a uniform random sampling of sensor nodes [Bash et al. 2004; Dimakis et al. 2006] and later adapted to get a similar spatial sampling [Sarkar et al. 2007].

Incremental routing table construction. The last implementation problem is to discover and store the routes of the long links selected by the spatial distribution for each node. Notice that here we have a seemingly chicken-and-egg problem, as route discovery requires a routing algorithm, while

[3]The geodesic distance between two points in a geometric domain is defined as the Euclidean length of the shortest path connecting the two points in the domain, avoiding obstacles.

the routing table construction is to supply such a routing scheme. Here we suggest a heuristic that finds the routes with bootstrapping and incrementally construct the routes for the long links with increasing lengths.

Every node first selects their long link partners (in fact, the geographical locations). The routes for the pairs with shorter distances are constructed first, and the routes for the pairs with length k are discovered with the current routing table information, that is, with the help of the long links with lengths smaller than k.

The route for a long link pq is stored on the routing table of the nodes on this path. Specifically, each routing table entry is a tuple (p, q, N_q), where N_q is the next hop neighbor leading to q. Thus a node maintains the routes to its long link partners as well as the routes that pass through it.

The simplicity of this scheme also suggests an 'on-demand' implementation to improve the basic routing. That is, when a packet is stuck at a local minimum we will select long links according to the spatial distribution. Thus routing delivery rate might be low or the delay can be long initially but as the routes for the long links are constructed and recorded the network gradually 'learns' and 'repairs' the imperfect distance oracle. This heuristic can be used to circumvent the issue of finding paths without a routing table, but unfortunately in this case, no proofs are known that would guarantee the stretch bounds shown in the previous sections.

5.2. Landmark-based routing table design

When the location information is not available or when the sensor field is deployed in an environment so that the Euclidean distance does not provide a good approximate distance oracle, we propose a second scheme with landmark-based distances. Specifically, we select $m = O(\log^2 n)$ landmarks ℓ_i uniformly randomly in the sensor network. For example, each node proposes to be a landmark with probability $\log^2 n/n$. The landmarks then flood the network and every other node records the hop count distance to these landmarks. The communication cost for the preprocessing is $O(n \log^2 n)$.

Landmark-based distance oracles. Each node p is given a landmark-based distance vector, represented by the vector of minimum hop count distance to all m landmarks, $(\sigma(p, \ell_1), \sigma(p, \ell_2), \cdots, \sigma(p, \ell_m))$. We would like to use the landmark distances to estimate the hop count distance of any two nodes. In the simulations we used the centered distance measure proposed in [Fang et al. 2005], which is a ℓ_2 norm of the centered landmark-based distance vector $(\sigma(p, \ell_1)^2 - M, \sigma(p, \ell_2)^2 - M, \cdots, \sigma(p, \ell_m)^2 - M)$, where $M = \sum_i \sigma(p, \ell_i)^2/m$.

Landmark-based sampling. To build the long links for a node p, we will use the landmarks to help with sampling. In particular, we select first randomly k out of the m landmarks. For each landmark ℓ_i, we select from the distribution $1/(r \ln D)$ (D is the network diameter) a distance ξ. If $\xi \leq \sigma(p, \ell_i)$, we take the node q along the path from p to ℓ_i with distance ξ from p as the long link partner. Otherwise we drop landmark ℓ_i. Intuitively, we select along the path from p to ℓ_i a node q with the spatial distribution restricted on this path. Since the landmarks are randomly selected, the probability that a landmark ℓ_i is at distance r from p is proportional to r. Now the probability that for each landmark ℓ_i we can obtain a valid long link is

$$\text{Prob}\{\xi < \sigma(p, \ell_i)\} = \int_0^D \int_1^\zeta \frac{1}{\xi \ln D} d\xi \frac{2\zeta}{D^2} d\zeta = 1 - \frac{1}{2 \ln D}.$$

Thus in expectation we obtain $k(1 - \frac{1}{2 \ln D})$ long links for each node. This means that choosing $m = O(\log^2 n)$ landmarks suffices to get enough long links for each node. At last we remark that although different nodes use the same set of landmarks to create their long links, the theoretical analysis in the previous section still holds – as the only requirement is that we have a sufficient number of independent long links for each individual node.

Landmark-based routing tables. With the long links constructed by the landmarks, the routing table size can be further reduced. In fact, a node p remembers in its routing table the long link partners and their landmark-based addresses. Different from the geographical case, the routes for the long links are implicitly implied by the landmark distances. The size of the routing table is therefore $O(\log^4 n)$, for $O(\log^2 n)$ landmarks/long link neighbors, and a storage of $O(\log^2 n)$ for storing the address of each long link neighbor.

5.3. Routing Scheme Implementation

We implemented our routing algorithm for simulations, using both the Euclidean distance oracle and the landmark based oracle. Each node keeps the routing table entries for its immediate neighbors, as well as the long link neighbors it has selected. The routes to the long link neighbors are stored on the routing tables of the nodes on the path. When a node s has a message to be delivered to a destination t, s will check its routing table to find a next hop node p. The node p was selected randomly from the set of feasible nodes in the forwarding region. Other than this stretch guaranteed strategy, we also simulated the effects of selecting a long link greedily from the routing table, where the p is the node in the routing table that is nearest to t according to the oracle. The message may not travel the entire long link if on a node in the the middle the message finds a closer neighbor to the destination.

The simulations (Figure 4) show that the greedy heuristic performs well in practice. Both schemes achieve high delivery rate and low stretch. The greedy routing may sometimes have lower delivery rate, but has better stretch. These results are understandable in the light of the fact that the forwarding region contains the destination, and a large region in between the source and the destination. Thus, the link in routing table that reaches closest to the destination is likely to be one in the forwarding region. Which means, in many cases, this heuristic satisfies the conditions of the algorithm, and because greedy choice is more likely to be nearer the destination than a random choice, it results in a low stretch. Thus, in simulations, we consider the greedy strategy to be comparable to the theoretical strategy. This also suggests further study and analysis of the spatial distribution and routing table constructions along these lines.

In a real network, there exists the additional problem of how to decide the correct number of long links or landmarks to create. In a situation where no prior knowledge of the network is available, this can be done adaptively. For example we start with a small number of landmarks, and monitor the failure rates of routing requests over a period again. If a certain fraction (say 1% or more) of requests have failed, we add a few more random landmarks. We do this check after every time period, and add landmarks unless the failure rate is lower than desired.

6. SIMULATIONS

In this section, we present simulation results to show the performance of the proposed schemes in practice. We mainly focus on geographic routing table to show the tradeoff of the routing table size v.s. routing stretch. We also evaluate the performance of landmark-based scheme on a network of complex topology, for which landmark-based approximate distance oracle captures the underlying network connectivity more accurately. We compare our approach with two recently proposed routing protocols, S4 [Mao et al. 2007] and VRR [Caesar et al. 2006], on three important criteria, i.e., delivery rate, the size of routing table and routing stretch. We also discuss the preprocessing cost of each scheme. In summary, our approach achieves high delivery rate (above 99%) and small stretch (about 1.03) with only a small number of long links, and a small routing table with modest preprocessing.

Simulation setup. We focus on evaluating the performance of all approaches at the routing layer, and assume the underlying details (i.e., packet loss and interference) have been handled at MAC and link layers. This is sufficient for our purpose of verifying the validity of the proposed ideas. Respecting reality, we adopt a lossy radio model used in the standard simulator TOSSIM [Levis et al. 2003] to determine direct communication links between nodes. The lossy radio model is generated based on empirical data and specifies the loss rate on the link between a pair of nodes. We only consider links with sufficient low loss rate and the resulted network is not necessarily unit disk graph, and could have directional links. We run simulations on three topologies. The first is a sparse network with 1000 random distributed nodes – this is representative of a large region monitored by a few inexpensive sensors. Second, a network with a large hole in the center and third network with multiple holes (see Figure 3) – these two are representative of certain urban or sensing environments that are closely monitored, but contain regions where sensors cannot be deployed. Each simulation run is repeated 10 times. In each round, we randomly selected 10000 pairs of source and destination. All results are averaged on all pairs.

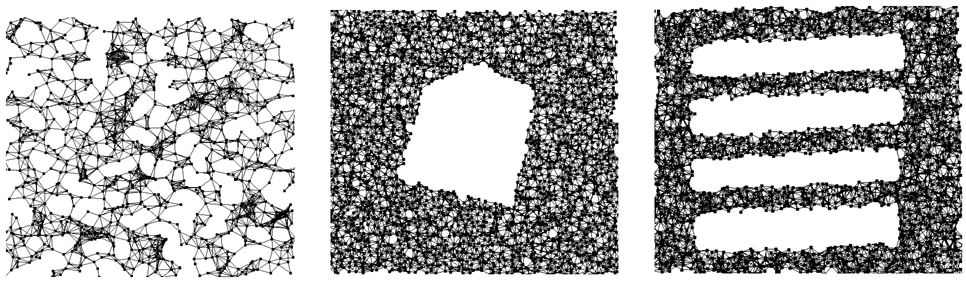

Fig. 3. Network topologies used in simulations. (i) Topology 1. Random network: 1000 nodes, avg. degree 7.2; (ii) Topology 2. Network with one hole: 2400 nodes, avg. degree 9.5; (iii) Topology 3. Network with multiple holes: 2000 nodes, avg. degree 10.6.

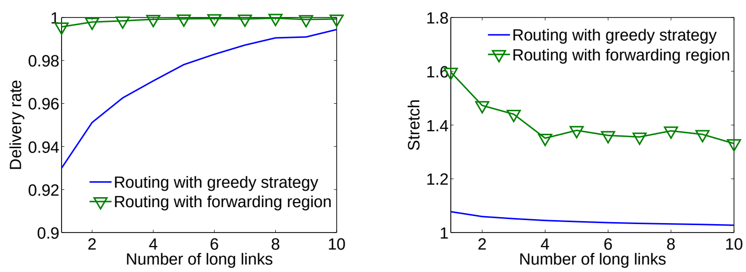

Fig. 4. (i) Delivery rate for Topology 2. (ii) Stretch for Topology 2.

6.1. Geographic routing table

We evaluate the performance of our approach with geographic routing table, as explained in Section 5.1.

Delivery rate. To show the effect of long links on the delivery rate, we vary the number of long links each node maintains from 0 to 16. When the number of long links is set to 0, the routing protocol is essentially the geographical greedy routing based on the location information within one-hop neighborhood. Figure 7 (i) shows that greedy routing performs very poorly without long links. The delivery rate is only around 50%, 65% and 44% in Topology 1, 2 and 3 respectively. When the number of long links increases, the delivery rate reaches 99% with $6, 8, 7$ long links per node in three different topologies, respectively. The results confirm that a small number of long links are sufficient and can significantly improve the delivery rate in some typical network topologies. The delivery rates of S4 and VRR are both 100%. The 1% failed message rate is the cost we pay for the substantially smaller routing table. Since our scheme behaves similarly in various topologies, in the rest of this subsection, unless mentioned otherwise, we only present results on Topology 2 due to space limitation.

We show the preprocessing cost of our scheme with varying number of long links in Figure 7 (iv). More long links results in higher preprocessing cost and increased delivery rate.

Routing table size. The size of routing table is measured by the number of entries in the table. We compare the average routing table size of our scheme with VRR and S4. For VRR, each node maintains routes to a set of virtual neighbors on the ID ring. Those virtual neighbors can be viewed as "long links". Thus, we show the change of routing table size as the number of long links changes for both our scheme and VRR in Figure 7(ii). It is easy to see that the size of routing table is proportional to the number of long links. But our scheme uses much smaller routing table than VRR when maintaining the same number of long links. Our scheme saves routing table size by taking long links with probability $1/r^2$ rather than the uniform distribution used in VRR. Thus, our scheme favors relatively shorter links. Figure 6 shows the distribution of the lengths of the long links in terms of hop counts. In our scheme there are fewer long links, while the distribution in VRR is more uniform.

Size of routing table	Our scheme	S4	VRR
Topology 1	26.08	68.83	41.52
Topology 2	39.02	105.85	62.48
Topology 3	37.28	90.62	63.82

Fig. 5. Average size of routing table.

The table in Figure 5 shows the routing table size of three schemes with a set of fixed parameters. For comparisons, we use 50 landmarks for S4 and each node maintains routes to 4 virtual neighbors in VRR. We select those parameters since they give the best performance of S4 and VRR in terms of both routing table size and stretch. For our scheme, we use $6, 8, 7$ long links in three topologies respectively to get above 99% delivery rate. We use the same set of parameters in other Tables. From Table 5, S4 requires the largest routing table, since each node needs to maintain routes to roughly $O(\sqrt{n})$ landmarks and $O(\sqrt{n})$ nodes within its local cluster. Our scheme has the smallest routing table size, but achieves comparable delivery rate.

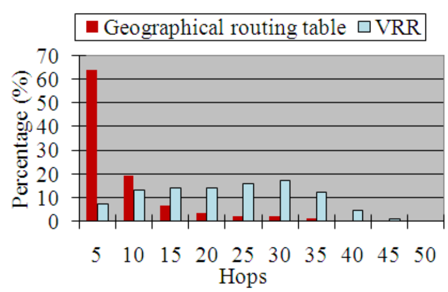

Fig. 6. The distribution of long links w.r.t their lengths in hops.

Stretch. Figure 7(iii) shows the average stretch of our scheme and VRR with varying number of long links. The stretch of our scheme is always below 1.1 and decreases when the number of long links increases. With 6 long links, the stretch is only about 1.03. With more long links, each node has more choices when choosing the next hop and can switch to the best direction as soon as it finds a neighbor or long link closer to the destination. Figure 9 compares the average stretch of three schemes. It shows that our scheme achieves similar stretch as S4 (but with smaller routing table) and is much better than VRR.

Diversity of inaccuracy. The inaccuracy of distance oracle is due to diverse disturbances of the network, like low density of node distribution or holes and obstacles. Here, we study the impact of

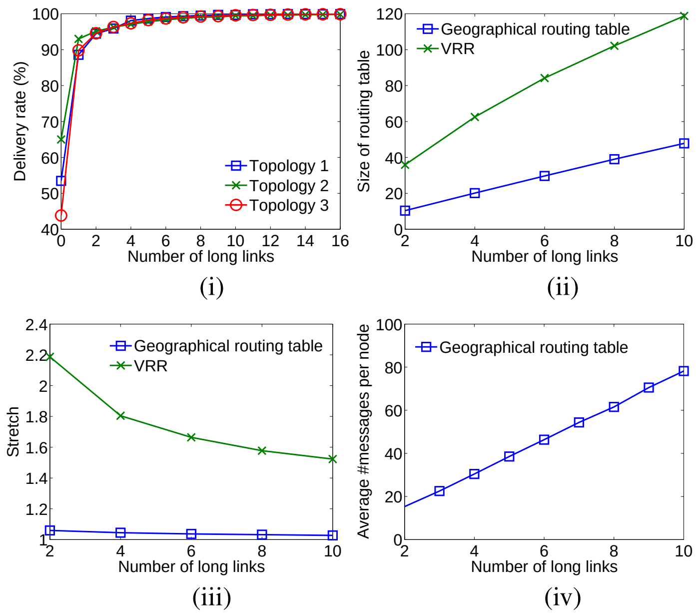

Fig. 7. (i) Delivery rate of geographical routing table with varying number of long links in different network topologies. (ii)-(iv) Performance of our scheme and VRR in Topology 2. (ii) The average size of routing table. (iii) Average stretch. (iv) Communication cost in preprocessing stage.

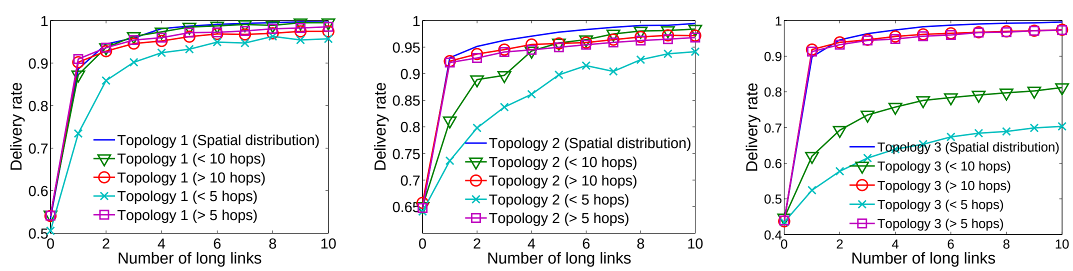

Fig. 8. Delivery rate for different topologies. (i) Topology 1. (ii) Topology 2. (iii) Topology 3.

Average stretch	Our scheme	S4	VRR
Topology 1	1.03	1.03	1.73
Topology 2	1.03	1.03	1.80
Topology 3	1.04	1.02	1.75

Fig. 9. Average stretch.

different types of links (relatively short links and long links) on different types of network topologies. We compare spatial-distribution link selection scheme with other four schemes, i.e., schemes that only select nodes within 5 hops (< 5), within 10 hops (< 10), at least 5 hops apart (> 5) and at least 10 hops apart (> 10). From all three figures (Figure 8), we can see that spatial distribution with a mixture of short and long links (blue line) achieves the highest delivery rate for all topologies. Relatively short links (< 5 hops) works best for Topology 1 compared to the other two topologies, and the scheme with only links shorter than 10 hops even performs better than other schemes with relatively longer links, because the local disturbance due to sparsity can be resolved by short links to close nodes. Longer links (> 10 hops) performs significantly better than pure short links in Topology 3, since global disturbance (big holes) requires longer links to compensate the inaccurate distance measure. Different network topologies may require different types of links, but the spatial distribution with a mixed set of short and long links gives a generic solution and hides the diversity of distance inaccuracy, with high delivery rate, small routing table size, low stretch and cost.

6.2. Landmark-based routing table

We evaluate the performance of the landmark-based routing table (in 5.2) on three topologies, compared with S4, as both use a set of landmarks. The benefits of our scheme are that it incurs much cheaper preprocessing cost with smaller routing table size than S4. Our scheme needs fewer landmarks ($O(\log^2 n)$ rather than $O(\sqrt{n})$ landmarks). Each node only needs to remember the next hop to each landmark and the sample along the path to that landmark, and does not construct any additional local routing tables. So the size of the routing table is exactly the number of landmarks. The total preprocessing cost is just the message flooding from the landmarks. After that, routes to all long links are built up automatically.

Simulation results show that 30 landmarks are sufficient to achieve good delivery rate (above 94%) and small stretch (about 1.04) in our scheme. In S4, we use 50 landmarks with an average routing table size of 90.62 to achieve the best stretch and routing table size tradeoff. The routing table size in our scheme is 30, with the total preprocessing cost only about 1/3 that of S4 on Topology 3.

7. CONCLUSION

We presented in this paper a theory to build a small number of routing links in very general domains. The method is distributed and uncoordinated, but guarantees global properties such as routing with low stretch and compact routing tables. The use of spatial distribution ensures that the routing works well at all scales and distances.

We have presented here implementation details and simulation results for sensor networks, but we expect the core results to be useful in a wide variety of graphs such as overlays networks.

REFERENCES

ABRAHAM, I., GAVOILLE, C., GOLDBERG, A. V., AND MALKHI, D. 2006. Routing in networks with low doubling dimension. In *Proc. of the 26th International Conference on Distributed Computing Systems (ICDCS)*.

ABRAHAM, I. AND MALKHI, D. 2005. Name independent routing for growth bounded networks. In *SPAA '05: Proceedings of the seventeenth annual ACM symposium on Parallelism in algorithms and architectures*. 49–55.

ABRAHAM, I., MALKHI, D., AND DOBZINSKI, O. 2004. LAND: stretch $(1 + \varepsilon)$ locality-aware networks for dhts. In *SODA '04: Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms*. 550–559.

BASH, B. A., BYERS, J. W., AND CONSIDINE, J. 2004. Approximately uniform random sampling in sensor networks. In *DMSN '04: Proceeedings of the 1st international workshop on Data management for sensor networks*. 32–39.

BOSE, P., MORIN, P., STOJMENOVIC, I., AND URRUTIA, J. 2001. Routing with guaranteed delivery in ad hoc wireless networks. *Wireless Networks 7,* 6, 609–616.

BRUCK, J., GAO, J., AND JIANG, A. 2007. MAP: Medial axis based geometric routing in sensor networks. *Wireless Networks 13,* 6, 835–853.

CAESAR, M., CASTRO, M., NIGHTINGALE, E. B., O'SHEA, G., AND ROWSTRON, A. 2006. Virtual ring routing: network routing inspired by DHTs. In *SIGCOMM '06: Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications*. 351–362.

CHAN, H. T.-H., GUPTA, A., MAGGS, B. M., AND ZHOU, S. 2005. On hierarchical routing in doubling metrics. In *SODA '05: Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*. 762–771.

COWEN, L. J. 1999. Compact routing with minimum stretch. In *SODA '99: Proceedings of the tenth annual ACM-SIAM symposium on Discrete algorithms*. 255–260.

DIMAKIS, A. G., SARWATE, A. D., AND WAINWRIGHT, M. J. 2006. Geographic gossip: efficient aggregation for sensor networks. In *IPSN '06: Proceedings of the fifth international conference on Information processing in sensor networks*. 69–76.

EILAM, T., GAVOILLE, C., AND PELEG, D. 2003. Compact routing schemes with low stretch factor. *J. Algorithms 46,* 2, 97–114.

FANG, Q., GAO, J., GUIBAS, L., DE SILVA, V., AND ZHANG, L. 2005. GLIDER: Gradient landmark-based distributed routing for sensor networks. In *Proc. of the 24th Conference of the IEEE Communication Society (INFOCOM)*. Vol. 1. 339–350.

FONESCA, R., RATNASAMY, S., ZHAO, J., EE, C. T., CULLER, D., SHENKER, S., AND STOICA, I. 2005. Beacon vector routing: Scalable point-to-point routing in wireless sensornets. In *Proc. of the 2nd Symposium on Networked Systems Design and Implementation (NSDI)*. 329–342.

FUNKE, S. AND MILOSAVLJEVIĆ, N. 2007. Guaranteed-delivery geographic routing under uncertain node locations. In *Proceedings of the 26th Conference of the IEEE Communications Society (INFOCOM'07)*. 1244–1252.

GOVINDAN, Y.-J. K. R., KARP, B., AND SHENKER, S. 2006. Lazy cross-link removal for geographic routing. In *SenSys '06: Proceedings of the 4th international conference on Embedded networked sensor systems*. 112–124.

GUPTA, A., KRAUTHGAMER, R., AND LEE, J. R. 2003. Bounded geometries, fractals, and low-distortion embeddings. In *FOCS '03: Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science*. 534–543.

KARGER, D. AND RUHL, M. 2002. Find nearest neighbors in growth-restricted metrics. In *Proc. ACM Symposium on Theory of Computing*. 741–750.

KARP, B. AND KUNG, H. 2000. GPSR: Greedy perimeter stateless routing for wireless networks. In *Proc. of the ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*. 243–254.

KEMPE, D., KLEINBERG, J., AND DEMERS, A. 2001. Spatial gossip and resource location protocols. In *STOC '01: Proceedings of the thirty-third annual ACM symposium on Theory of computing*. 163–172.

KIM, Y.-J., GOVINDAN, R., KARP, B., AND SHENKER, S. 2005. On the pitfalls of geographic face routing. In *DIALM-POMC '05: Proceedings of the 2005 joint workshop on Foundations of mobile computing*. 34–43.

KLEINBERG, J. 2000a. Navigation in a small world. *Nature* 406, 845.

KLEINBERG, J. 2000b. The small-world phenomenon: an algorithm perspective. In *STOC '00: Proceedings of the thirty-second annual ACM symposium on Theory of computing*. 163–170.

KLEINBERG, J., SLIVKINS, A., AND WEXLER, T. 2004. Triangulation and embedding using small sets of beacons. In *Proc. 45th IEEE Symposium on Foundations of Computer Science*. 444–453.

KONJEVOD, G., RICHA, A. W., AND XIA, D. 2006. Optimal-stretch name-independent compact routing in doubling metrics. In *PODC '06: Proceedings of the twenty-fifth annual ACM symposium on Principles of distributed computing*. 198–207.

KUHN, F., WATTENHOFER, R., ZHANG, Y., AND ZOLLINGER, A. 2003. Geometric ad-hoc routing: of theory and practice. In *PODC '03: Proceedings of the twenty-second annual symposium on Principles of distributed computing*. 63–72.

LEVIS, P., LEE, N., WELSH, M., AND CULLER, D. 2003. TOSSIM: accurate and scalable simulation of entire tinyos applications. In *SenSys '03: Proceedings of the 1st international conference on Embedded networked sensor systems*. 126–137.

LI, J., JANNOTTI, J., DECOUTO, D., KARGER, D., AND MORRIS, R. 2000. A scalable location service for geographic ad-hoc routing. In *Proceedings of 6th ACM/IEEE International Conference on Mobile Computing and Networking*. 120–130.

LINIAL, N., LONDON, E., AND RABINOVICH, Y. 1995. The geometry of graphs and some of its algorithmic applications. *Combinatorica 15*, 215–245.

MAO, Y., WANG, F., QIU, L., LAM, S. S., AND SMITH, J. M. 2007. S4: Small state and small stretch routing protocol for large wireless sensor networks. In *Proceedings of the 4th USENIX Symposium on Networked System Design and Implementation (NSDI 2007)*.

MILGRAM, S. 1967. The small world problem. *Psychology Today* 1.

NEWSOME, J. AND SONG, D. 2003. GEM: graph embedding for routing and data-centric storage in sensor networks without geographic information. In *SenSys '03: Proceedings of the 1st international conference on Embedded networked sensor systems*. 76–88.

NGUYEN, A., MILOSAVLJEVIC, N., FANG, Q., GAO, J., AND GUIBAS, L. J. 2007. Landmark selection and greedy landmark-descent routing for sensor networks. In *Proceedings of the 26th Conference of the IEEE Communications Society (INFOCOM'07)*. 661–669.

PELEG, D. 2000. *Distributed Computing: A Locality-Sensitive Approach*. SIAM Monographs on Discrete Mathematics and Applications.

PLAXTON, C. G., RAJARAMAN, R., AND RICHA, A. W. 1997. Accessing nearby copies of replicated objects in a distributed environment. In *Proc. ACM Symposium on Parallel Algorithms and Architectures*. 311–320.

RAO, A., PAPADIMITRIOU, C., SHENKER, S., AND STOICA, I. 2003. Geographic routing without location information. In *Proceedings of the 9th annual international conference on Mobile computing and networking*. 96–108.

SARKAR, R., ZHU, X., AND GAO, J. 2007. Hierarchical spatial gossip for multi-resolution representations in sensor networks. In *Proc. of the International Conference on Information Processing in Sensor Networks (IPSN'07)*. 420–429.

SHARMA, G. AND MAZUMDAR, R. 2005. Hybrid sensor networks: a small world. In *MobiHoc '05: Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*. 366–377.

THORUP, M. AND ZWICK, U. 2001a. Approximate distance oracles. In *Proc. ACM Symposium on Theory of Computing*. 183–192.

THORUP, M. AND ZWICK, U. 2001b. Compact routing schemes. In *SPAA '01: Proceedings of the thirteenth annual ACM symposium on Parallel algorithms and architectures*. 1–10.

TRAVERS, J. AND MILGRAM, S. 1969. An experimental study of the small world problem. *Sociometry* 32.

WU, J. AND YANG, S.-H. 2008. Small world model-based polylogarithmic routing using mobile nodes. *Journal of Computer Science and Technology 23,* 2, 327–342.

ZHANG, F., LI, H., JIANG, A. A., CHEN, J., AND LUO, P. 2007. Face tracing based geographic routing in nonplanar wireless networks. In *Proceedings of the 26th Conference of the IEEE Communications Society (INFOCOM'07)*. 2243–2251.