An efficient implementation of the greedy forwarding strategy*

Hannes Stratil

Embedded Computing Systems Group E182/2
Technische Universität Wien
Treitlstraße 3
A-1040 Vienna
Email: hannes@ecs.tuwien.ac.at

Abstract: A wireless sensor network needs a suitable multihop routing protocol to facilitate the communication between arbitrary sensor nodes. Position-based routing protocols are attractive for large-scale sensor networks due to their location awareness and scalability. A large number of position-based routing protocols use the greedy forwarding strategy. This paper presents an efficient implementation of the greedy forwarding algorithm. The nodes of the sensor network are viewed as points in the plane. Each node uses a Voronoi Diagram to generate a subdivision of the plane, which is solely based upon its local neighborhood. Every node forwards incoming messages to that Voronoi Cell the destination node belongs to, according to its position information. The complexity of our solution is $O(n \log n)$ for computing the Voronoi Diagram, which is typically done only once, but only $O(\log n)$ for finding the next-hop node. Interestingly, n is not the total number of nodes in the network here, but only the number of single hop neighbors.

1 Introduction

Wireless sensor networks are currently an active field of research both in industry and academia. A sensor network is composed of a large number of nodes that are randomly dispersed over some area of interest. Not all nodes in a wireless network can communicate directly, so a multihop routing protocol is needed. Most routing protocols for wireless networks have been designed for networks of just a few hundreds of nodes and do not scale to networks with thousands of nodes. This is not the case for *position-based routing* protocols. They assume the existence of a position service which provides network participants with their location. The Global Position Service (GPS) is undoubtedly the most well known position service in use today. A survey of position services can be found in [HB01]. In sensor networks, the position is in fact more important than a specific node ID. For example, tracking applications are more interested in where the target is located than in the ID of the reporting node.

*This research has been supported by the Austrian START programme Y41-MAT

Many position-based routing protocols use variants of the greedy forwarding approach which is a localized approach: The routing decision at a node in the network is only based on its own position, the position of its single hop neighbor nodes and the position of the destination node. Greedy routing does not require the establishment or maintenance of routes: The nodes neither have to store routing tables nor do they transmit messages to keep the routing tables up-to-date, and no global information about the topology of the network is needed. When an intermediate node receives a message for a specific destination node, it forwards the message to the neighbor node which is closest to the destination node among all its neighbors. The question of how to find the closest neighbor node is usually not addressed, however.

Our implementation uses a *Voronoi Diagram* to determine the neighbor node that is closest to the destination node. For a set of sites in the plane, the Voronoi Diagram partitions the plane into convex polygon cells that contain all points which are closer to the site of this cell than to any other site. The closest site to a given query point can be found by a point location algorithm. The time complexity of computing a Voronoi Diagram for n sites in the plane is $O(n \log n)$, and the point location algorithm needs only $O(\log n)$ time, where the Voronoi Diagram has to be computed only once. Note that n is not the total number of nodes in the network but only the number of nodes in the local neighborhood, which is independent of the network size.

We concentrate in this paper only on an efficient method for determining the closest neighbor node with respect to a specific destination node. Additionally, we show that each message eventually reaches its destination if the spatial density of network nodes is high in relation to the transmission range. The problem of recovery in sparse networks (because of void regions or ineffective nodes) is not considered in our work.

The remainder of our paper is organized as follows: The following Section 2 gives a short overview of some publications that are related to our work. In Section 3 we provide some preliminaries and the basic principles of Voronoi Diagrams. A description of our algorithm is presented in Section 4. Finally, Section 5 concludes the paper.

2 Related Work

Routing in ad hoc wireless (sensor) networks has been studied extensively in the last years. A survey and comparison of position-based approaches can be found in [GSB03]. A large number of position-based routing protocols use the greedy forwarding strategy proposed by [Fi87], where only neighbors with a position closer to the destination are considered (e.g, [KK00, FGG04]).

3 Preliminaries and Assumptions

3.1 Network Model

We assume that the nodes have negligible difference in altitude, so they can be considered roughly in a plane.

A wireless network can be modeled as an undirected communication graph $G(S,E)$ in the plane, with a set of sites S and a set of edges E. Each site s_i of the set $S := \{s_0, \ldots, s_{N-1}\}$ represents a node of our wireless sensor network. The total number of sites is $N = |S|$. An edge $(s_i, s_j) \in E$, $s_i, s_j \in S$, represents a wireless link of the ad hoc network in $G(S,E)$. An edge (s_i, s_j) is present in $G(S,E)$ if and only if $\|(s_i, s_j)\|$ is less than the common communication range R, where $\|(s_i, s_j)\|$ denotes the Euclidean distance between s_i and s_j. The neighborhood of $s \in S$, denoted by $\mathcal{N}(s)$, is the set of nodes within s's communication range R and the node s itself. Any node $s_i \in \mathcal{N}(s)$ is called visible to s, and we assume $|\mathcal{N}(s)| \leq n$, $\forall s \in S$. Note that the number of nodes in the neighborhood is many times smaller than the overall number of nodes in the network: Sensor networks typically have some thousand+ nodes, but the number of neighbors of a specific node is only in the range of ten.

3.2 Basic Principles of Voronoi Diagram

Voronoi Diagrams belong to most important structures of computational geometry. Parts of the following definitions are adopted form the work of Aurenhammer [Au91].

Definition 1 (Voronoi Diagram) *Set $\mathcal{N}(s)$ includes the single-hop neighbors of s and s itself. Let $|\mathcal{N}(s)| = n'$, $n' \leq n$. The Voronoi Diagram of $\mathcal{N}(s)$ is the subdivision of the plane into n' Voronoi Cells, one for each site $s_i \in \mathcal{N}(s)$. A point p lies in the cell corresponding to a site s_i, if $\|(p, s_i)\| < \|(p, s_j)\|$, for all $s_i, s_j \in \mathcal{N}(s), j \neq i$.*

Properties of Voronoi Diagrams:

- **Voronoi Edge:** A point p lies on a Voronoi Edge between sites s_i and s_j *iff* the largest empty circle centered at p touches only s_i and s_j. A Voronoi Edge is a subset of locus of points equidistant from s_i and s_j.
- **Voronoi Vertex:** A point p is a Voronoi Vertex *iff* the largest empty circle centered at p touches at least 3 sites. A Voronoi Vertex is an intersection of three or more Voronoi Edges, each equidistant from a pair of sites.
- **Voronoi Cell:** A Voronoi Cell is a (possibly unbounded) convex polygon. The boundary of a Voronoi Cell consists of Voronoi Edges and Voronoi Vertices.

The worst-case complexity of computing a Voronoi Diagram for n sites in two dimensions is $O(n \log n)$ and the storage requirement is only $O(n)$ [Au91]. An optimal method for the computation of Voronoi Diagrams is *Fortune's plane sweep* algorithm [Fo87].
One of the most popular problems in the context of Voronoi Diagram is the *post-office problem*. Given are n sites in the plane (post offices), the problem is to find the site which is closest to a given query point p (the location of a person). Note that there exists a trivial $O(n)$-time solution by computing all n distances. Voronoi Diagrams offer a more efficient solution for point location: A site s_i is closest to p if and only if p falls into the Voronoi Cell of s_i. There exist worst-case optimal techniques to perform point location on a Voronoi Diagram, such as the *triangulation refinement method* due to Kirkpatrick [Ki83], and the *bridged chain method* due to Edelsbrunner et al. [EGS86]. Point location in a Voronoi Diagram with n regions is possible in $O(\log n)$ time and needs $O(n)$ storage.

Using any of the optimal Voronoi Diagram algorithms presented above we obtain the following result related to the point location problem.

Theorem 1 *Given a set of n sites in the plane, one can, within $O(n \log n)$ time and linear storage $O(n)$, construct a data structure that supports nearest neighbor queries: For an arbitrary query point p, its nearest neighbor can be found in time $O(\log n)$.*

4 Greedy forwarding algorithm with Voronoi Diagrams

Our algorithm uses Voronoi Diagrams to find the closest neighbor node with regard to a specific destination node. Each node s in our wireless sensor network only knows his own position and the positions of its immediate neighbors $s_1 \ldots s_{n-1} \in \mathcal{N}(s)$. These neighbors are the only nodes with which a node can communicate without using multi-hop connections.
The positions of the neighbors are used to create the local Voronoi Diagram. Figure 1 illustrates the local Voronoi Diagram of node s. The circle with radius R indicates the maximum communication range of s. The local Voronoi Diagram divides the plane into Voronoi Cells and each cell contains all points closer to the site of this cell than to all other sites. When a node wants to forward a message to a destination node, it determines the cell the destination node belongs to (using the point-location algorithm in Section 3.2), and forwards the message to that unique neighbor node that is the site of this cell. If two or more neighbor nodes are equidistant from the destination node, one of the neighbor nodes is either chosen arbitrarily or by some deterministic rule (e.g. node ID).

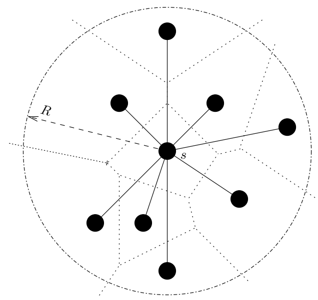

Figure 1: *Voronoi Diagram of node s_0*

In Lemma 1 below, we will show that the greedy routing approach delivers a message reliable to the destination if the network is dense.

Definition 2 (Dense Network) *A network is called dense if each node s in the network has, for each destination node (that is not a single hop neighbor node), a single hop neighbor which is closer to the destination than node s itself.*

Lemma 1 *In a Dense Network, the greedy forwarding algorithm eventually delivers each message to its destination.*

Proof: The greedy forwarding algorithm is loop-free in dense networks, because each forwarding step decreases the distance to the destination node: If a node would receive the same message twice, at least one forwarding step had increased the distance to the destination node, which is a contradiction. Hence, each message is eventually delivered to its destination. □

5 Conclusion

This paper presents a novel implementation of the well known greedy forwarding algorithm. It shows that conjunction of Voronoi Diagrams and the post office problem yield a very efficient solution for position based routing. Further details are provided in [St04], which also shows how to extend this approach for networks that are not dense.

References

[Au91] Aurenhammer, F.: Voronoi diagrams – a survey of a fundamental geometric data structure. *ACM Computing Surveys*. 23(3):345–405. September 1991.

[EGS86] Edelsbrunner, H., Guibas, L. J., und Stolfi, J.: Optimal point location in a monotone subdivision. *SIAM Journal on Computing*. 15(2):317–340. May 1986.

[FGG04] Fang, Q., Gao, J., und Guibas, L. J.: Locating and bypassing routing holes in sensor networks. In: *Proceeding of IEEE Infocom*. March 2004.

[Fi87] Finn, G. G.: Routing and addressing problems in large metropolitan-scale internetworks. Technical Report ISI Research Resport ISI/RR-87-180. Information Sciences Institute, University of Southern California. March 1987.

[Fo87] Fortune, S.: A sweepline algorithm for voronoi diagrams. *Algorithmica*. 2:153–174. 1987.

[GSB03] Giordano, S., Stojmenovic, I., und Blazevic, L.: Position based routing algorithms for ad hoc networks: A taxonomy. In: Cheng, X., Huang, X., und Du, D.-Z. (Hrsg.), *Ad Hoc Wireless Networking*. Kluwer Academic Publisher. 2003.

[HB01] Hightower, J. und Borriello, G.: Location systems for ubiquitous computing. *IEEE Computer Magazine*. 34(8):57–66. August 2001.

[Ki83] Kirkpatrick, D. G.: Optimal search in planar subdivisions. *SIAM Journal on Computing*. 12(1):28–35. February 1983.

[KK00] Karp, B. und Kung, H. T.: GPSR: greedy perimeter stateless routing for wireless networks. In: *Proceedings of the sixth annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 2000)*. S. 243–254. August 2000.

[St04] Stratil, H.: Design of a voronoi-aided routing (VAR) protocol for wireless sensor networks. Research Report 15/2004. Technische Universität Wien, Institut für Technische Informatik. Treitlstraße 3, A-1040 Vienna, Austria. 2004. (submitted).