

Competitive Time and Traffic Analysis of Position-based Routing using a Cell Structure

Stefan Rührup* Christian Schindelhauer†
Heinz Nixdorf Institute
University of Paderborn, Germany

Abstract

We present a strategy for organizing the communication in wireless ad hoc networks based on a cell structure. We use the unit disk graph model and assume positioning capabilities for all nodes. The cell structure is an abstract view on the network and represents regions where nodes reside (node cells), regions that can be used for the communication flow (link cells) and regions that cannot be bridged due to the restricted transmission range (barrier cells). The cell structure helps to determine local minima for greedy forwarding and improves recovery from such minima, because for recovery all edges can be used in contrast to other topology-based rules that work only on a planar subgraph. For the analysis of position-based routing algorithms the measures time and traffic are based on the cell structure. The difficulty of exploring the network is expressed by the size of the barriers (number of cells in the perimeters). Exploration can be done in parallel, but with increasing traffic. We propose a comparative measure to assess both time and traffic, the combined comparative ratio, which is the maximum of the ratio of routing time and optimal time and the ratio of the traffic and the minimum exploration costs. While flooding and common single-path strategies have a linear ratio, we present a simple algorithm that has a sub-linear combined comparative ratio of $\mathcal{O}(\sqrt{h})$, where h is the minimal hop distance between source and target.

1 Introduction

For position-based routing nodes are identified by their unique geographical positions. The task is to deliver a message from a source node to a target node identified by its position in an unknown wireless ad hoc network. We try to optimize the number of messages and the time to perform this task in a worst case setting. As communication model we use the unit disk graph model.

One obstacle for efficient position based routing is the lack of knowledge about the network structure available at the beginning. In particular, reactive routing protocols that do not know any network structure in advance fail to solve this problem efficiently. We show that in the unit disk graph model even for a constant hop distance between source and target all nodes of the network need to send at least one message. Yet, if some local proactive communication is allowed to structure the basic network, the situation becomes much easier.

It is a common technique to build clusters and use special subgraphs as a backbone to simplify communication [6]. Other approaches rely on planarized graphs where no communication links may cross each other [10, 1]. These graph topologies can be built by some local communication protocol without much overhead. In this paper we strive for the simplest topology possible and establish a grid cell structure. We show that for an appropriate choice of the cell size routing is as efficient as in the unit disk graph with only a constant factor loss in performance.

Such topologies are not built on the fly during the routing, yet built in advance by providing some local information to neighboring nodes. We formalize this pre-structuring of the network. If no communication is

*DFG Graduiertenkolleg 776 (postgraduate programme) “Automatic Configuration in Open Systems”, sr@uni-paderborn.de

†Institute of Computer Science, schindel@uni-paderborn.de. Partially supported by the DFG Sonderforschungsbereich 376 and by the EU within 6th Framework Programme under contract 001907 “Dynamically Evolving, Large Scale Information Systems” (DELIS).

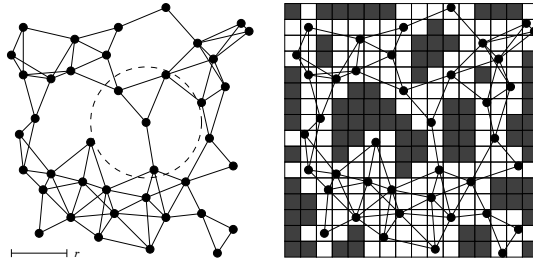


Figure 1: Unit disk graph with transmission radius r (left) and grid subdivision (right) of a node set.

done without a demand, i.e. a message to be routed through the network, we call the corresponding routing **reactive**. Then, a topology discovery has to be done during the routing. Here, we show that in a worst case setting this additional overhead cannot be neglected. If in advance the complete network structure can be exchanged between all reachable participants of the network in advance, we call this communication pattern **proactive**.

All these routing approaches have their benefits. In this article we try to keep proactive communication at a minimum. For this, we measure the impact of information by the number of hops k it has to be transmitted, denoted as k -hop-proactive protocols. We establish our basic communication network by a 2-hop-proactive communication pattern providing the coordinates of the neighbors and information about the coordinates of the neighbors' neighbors at each node.

One of the main ideas in this paper is to use the position information to define a global grid. The length of a grid square ℓ is a constant fraction of the transmission range r . We will call such grid squares *cells*. Each mobile host uses its positioning capabilities to determine to which cell it belongs. It can also assign its neighbors to the cells within its transmission range, based on their position coordinates. Thus, we can abstract from the view of a mobile host's environment by classifying the cells. In Section 4 we will see how this classification leads to a formal description of barriers.

This cell structure is a tool for describing efficient algorithms, not merely an end in itself. We can show that one loses only (small) constant factors in resources if one uses this cell structure. Furthermore, we show how this cell structure can be established using local information. Basically we have to distinguish only between three types of cells: node cells, link cells and barrier cells. This cell structure gives an abstract view on the network and represents regions where nodes reside (node cells), regions that can be used for the communication flow (link cells) and regions that cannot be bridged due to the restricted transmission range (barrier cells). The cell sizes are chosen such that locality, i.e. two nodes are located in vertically or horizontally neighboring cells, ensures communication. Under this assumption we can show that the cell path using node cells and link cells is equivalent to the optimum path connecting two nodes up to a constant factor.

This leads to our second main contribution. The complexity of the network scenario can be described by the sum of perimeters of all barriers. If this sum is small then routing is easy, which means it can be achieved using greedy algorithms with a recovery strategy in the case a small barrier is hit. If this barrier size is large, then flooding is the best possible algorithm.

We formulate this problem as an online routing problem. As complexity measures we consider time and traffic for delivering the message from source to target cell. Time is the number of rounds (measured in cell hops; a message is passed from one cell to another in one round) until the message reaches the destination if the node is accessible. Traffic is the total number of messages sent between cells. We investigate the time under a competitive measure, i.e. we compare the time needed by the message delivery algorithm with the length h of the shortest cell path using only node and link cells between source and target. This so-called *competitive analysis* is well known in the field of online algorithms [3]. For the traffic we define a comparative measure, which is the number of messages divided by the perimeter size p of all barriers formed by barrier cells plus the minimum hop-distance h . This perimeter size denotes the length of all right-hand traversal paths surrounding all sets of barrier nodes. For some instances this perimeter size gives a lower bound on the traffic.

There are single-path strategies¹ that can route a message in such a network with asymptotically optimum comparative traffic ratio of $\mathcal{O}(1)$. However, for such strategies the competitive time ratio is up to $\Omega(h)$. The best time behavior is provided by broadcasting a message, aka. as flooding. This method has optimum competitive time ratio 1, yet an arbitrarily bad comparative traffic ratio. One can improve this method by the expanded ring search scheme, giving a competitive time ratio of $\mathcal{O}(1)$, yet the competitive traffic ratio of $\Omega(h)$ for some instances remains rather high. A detailed description of these ratios is given in Subsection 5.2.

At a first look it seems that for routing in an unknown environment there is a trade-off between time and traffic with respect to the comparative ratio. In this paper we focus on this question and introduce the combined comparative ratio which is the maximum of the competitive time ratio and the comparative traffic ratio. Note that all position based routing algorithms for ad hoc networks considered so far have a combined comparative ratio of $\Omega(h)$. In this paper we present as our third main contribution a simple online routing algorithm with a sub-linear combined comparative ratio of $\mathcal{O}(\sqrt{h})$.

2 Related Work

There are many algorithms for routing in mobile ad hoc networks. Some rely on the maintenance of routing tables, others do the routing completely reactive which leads to flooding algorithms. If the nodes know their position, then single-path routing can be done without establishing routing tables.

Position-based routing (also called geometric or geographic routing, see [15, 7] for a survey) uses positioning capabilities of the nodes for routing packets in the network. It is based on the assumptions that (1) nodes can determine their own position, (2) know the position of their direct neighbors and (3) that the source node knows the position of the destination. Besides this common model, there is a routing scheme, called beacon-less routing [8], that is not reliant on the second assumption and uses the position data for implicit addressing.

Karp and Kung [11] and Bose et al. [4] have proposed the idea to combine a greedy forwarding strategy with a recovery strategy that is used when the greedy forwarding rule has guided the message into a local minimum. *Greedy forwarding* is based on a local decision of a node to pass a packet to a neighboring node that is nearer to the destination. Local decision strategies are: MFR (most forward within the transmission radius) [16] tries to minimize the number of hops, NFP (nearest with forwarding progress) [9] tries to minimize energy consumption, and compass routing [13] tries to minimize the Euclidean path length by choosing the node with the minimal angular distance to the direction of the destination. Since a packet could get stuck in a local minimum, greedy forwarding needs a *recovery strategy* like perimeter routing [11] (cf. “face-2” in [4]), where a packet is passed along the edges of the face incident to the dead end node until a forwarding progress according to the greedy strategy can be made. Another strategy, where the incident face is traversed completely in case of a local minimum (“face routing”), has been proposed by Kranakis et al. (see [13], “compass routing II”; cf. “face-1” in [4]) as a routing strategy (not only for recovery). They consider a line segment between the source and the destination and then traverse the faces of the network graph that are intersected by the line segment. Using position information the nodes determine the intersection points and know where and when to switch to the next face. For perimeter routing and face routing the so-called *right-hand rule*² is used for traversing a face. To prevent messages from getting into a loop while applying this rule the network topology must be a planar embedding of a graph. There are various planar graphs that can be used for constructing a planar subgraph of the unit disk graph, e.g. the Delaunay triangulation or relative neighborhood graphs [10]. The construction can be done locally by position information and communication in the local neighborhood. These subgraph constructions and variants have been proposed by Gao et al. [6] and Alzoubi et al. [1] as topologies for routing in mobile ad hoc networks. The corresponding planarization strategies eliminate advantageous long edges and preserve short ones. This leads to an increased hop distance of the shortest path in the planarized graph compared to the original communication graph. In our construction we allow all edges to be used in recovery mode.

¹An algorithm that follows a straight line connecting source and target and traverses all barriers intersecting this line needs $\mathcal{O}(h + p)$ messages.

²The right-hand rule is a well-known rule for traversing a maze: By keeping one hand (either the left or the right hand) in touch of the wall, it is guaranteed to find a way out of a maze.

The proposed topologies for planarization have been analyzed with respect to their spanning properties³. Often the desired properties (e.g. constant stretch factor) are only valid if a constant minimum distance between the nodes is assumed (“civilized graphs” [17], $\Omega(1)$ -model [14]). To take advantage of such properties in scenarios with unbounded node density, where the minimum distance assumption may be violated, a clustering approach can be applied. Clustering is used in [6] or [1] to form a backbone network connecting the clusterheads which has the desired graph properties.

We use an implicit clustering based on a cell structure, which is based on the positions of the nodes in the neighborhood. We need no additional communication for leader election etc. The advantage of the cell structure over the planarization strategies is that communication to all the neighbors is allowed. Recovery becomes easy because the right-hand rule used for traversing a barrier can be applied easily in the cell structure. Topology-based rules (e.g. for applying the right-hand rule on the edges on the graph), that need the planarization are no longer necessary.

Furthermore we use the cell structure to identify barriers, which are local minima to a greedy forwarding strategy. Fang, Gao and Guibas [5] proposed an algorithm for identifying local minima and the regions behind, which they call “routing holes” and which are similar to our notion of barriers. The algorithm uses local rules, but the nodes on the border of the barriers need to communicate so that traffic used for traversing the barriers is shifted from the routing algorithm to the pro-active part of the communication.

3 Learning the Neighborhood

We start our discussion with the question how much communication is reasonable to learn enough about the neighborhood to establish a grid structure. For some applications, this step is not necessary. In particular, if the nodes are already placed on a grid like structure and the position is given by two integer coordinates. Now, if the grid size is small enough we can skip some parts of the following protocol and continue with routing in the grid structure.

Here, we discuss the issue how far location information must be proactively distributed until a routing protocol can deliver a message. As a simplification we consider the unit disk graph model in this section combined with a round model. Every node has its exact, unique location information. In one round (or time step) at most one message can be sent by a node. In this paper we use the terms message and data packet interchangeably. Note that in such a round model, the beacon-less routing protocol [8] cannot be applied, because it requires a continuous time space for distinguishing between the positions of the neighboring nodes in order to choose the node with the most suitable location. We motivate the round model by the fact, that the response time of a node in a wireless network is limited, and therefore in practice a continuous medium access protocol relying on arbitrarily small time differences does not work.

We model the network as a graph $G = (V, E)$ with $|V| = n$, where the n nodes V are the participants of the network and the edges E are the communication links. The nodes are placed in the Euclidean plane and have a fixed transmission radius of r . Because of the bounded transmission range, we refer to *unit disk graphs* which correspond to a normalized transmission radius ($r = 1$).

Proactive routing protocols are often referred to as table-driven protocols, whereas reactive routing is also called on-demand routing. In position-based routing proactive communication is often used to exchange position information among neighboring nodes. We call a routing algorithm **proactive** if a communication pattern includes a periodic exchange of control messages even if there is no communication demand. If communication is triggered only by demand, we call this routing algorithm **reactive**.

Here, we define a parameterized notion of **k -hop-proactive routing**. A communication protocol is k -hop-proactive, if proactively transmitted information of a node are available only in nodes in a hop-distance of k . Such a constant-hop routing algorithm can adapt quickly to network changes, while reducing the proactive traffic. According to this notion 0-hop-proactive routings refers to reactive routing, while n -hop-proactive routing denotes the classical exchange of routing tables.

In [5] a similar approach is used. In their routing protocol a message is sent around the routing holes in the network (here called barriers). If p is the maximum hop distance such a message travels around a barrier, then this routing protocol is p -hop-proactive. In this paper we reduce the proactive hop distance

³A subgraph of a graph is a spanner, if the length (Euclidean length, hop distance or power distance) of a path between two nodes is only a constant factor larger than the distance of the nodes in the original graph.

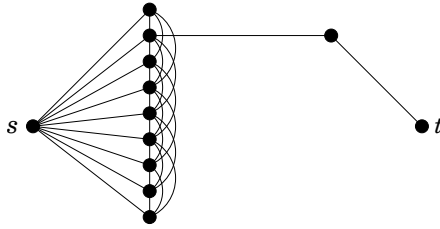


Figure 2: Worst-case example for reactive routing

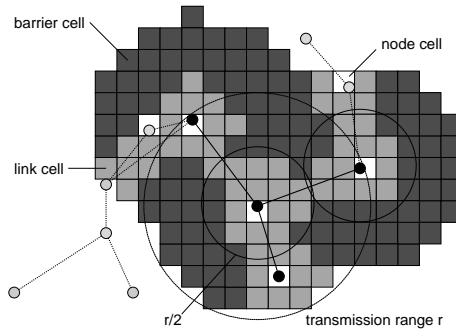


Figure 3: Classification of cells.

even further and present a 2-hop-proactive communication pattern establishing a grid structure that supports barrier traversal in position-based routing.

Before we present this scheme, we show that completely reactive protocols cannot cope with the reactive communication part of 2-hop-proactive routing protocols. The main advantage of reactive routing is the lack of communication overhead during demandless time phases. The following theorem shows the high price to be payed for this.

Theorem 1 *Every reactive position-based routing protocol needs $\Omega(n)$ message transmissions or $\omega(1)$ rounds for delivering a single packet in the worst case even if the hop distance between source and target is constant.*

The proof of this theorem follows the ideas presented in [2] using the situation depicted in Figure 2 and will appear in the full paper. The source s can reach $n - 3$ nodes directly. But only one of them is on the shortest path of length 3 from s to the destination node t , which one is not clear at the beginning of the routing.

Note that we will present a 2-hop-proactive protocol that uses constant number of rounds and messages to deliver packets over a constant hop distance. So, this theorem shows that some minimum proactivity is helpful. In the next subsection we show how to cluster nodes in a grid structure using a 2-hop-proactive communication pattern: *1st information hop*: Every node knows the coordinates of its neighbors. *2nd information hop*: Every node knows all neighbors of the neighbors and their coordinates.

Standard protocols can be used to provide this information. In the following we assume that every node has access to this information.

4 Establishing the Cell Structure

In this section we describe how we establish the grid structure from the proactively distributed information. We choose a cell structure based on a grid subdivision with $\ell \times \ell$ squares, called cells, where r is the transmission radius in the unit disk graph model and $\ell \leq \frac{1}{2}(\sqrt{3} - 1)r$.

The basic questions to be answered for establishing the cell structure are: Which cells are directly connected to my cell (two cells are directly connected, if at least a pair of nodes from each cell is connected)? How can I deliver a message to such a cell?

The first question can be solved by using the 2-hop information about the coordinates of the neighbors' neighbors. This follows from the fact that all nodes in a cell are within reach (since $\ell \leq r/\sqrt{2}$). Now, all directly connected cells have at least one node which has a neighbor in the neighboring cell. Then the second question can be answered from the same information: We can route a message to each directly connected cell within two hops.

Note that for providing this information it suffices to transmit even less information according to the following scheme: *1st information hop*: Every node knows the coordinates of its neighbors. *2nd information hop*: Every node knows the cells containing all neighbors of the neighbors.

This communication pattern reduces the information size in the second hop by replacing continuous coordinations of arbitrary many nodes with a constant number of discrete cell addresses.

Now, we extend this scheme by a classification of cells which allows to use edges that span over more than two neighboring cells. We introduce three classifications for a cell, see Figure 3: A *node cell* contains at least one node. A *link cell* does not contain any node. It satisfies one of the following geometric properties: It is intersected by the line between two nodes with maximum (transmission) distance r . All points of this cell have distance at most $r/2$ to a node. Every other cell is called a *barrier cell*.

Barrier cells can obstruct the position-based routing, especially if they come in a larger group.

Definition 1 A barrier is a set of barrier cells, which are connected orthogonally or diagonally. Barrier cells that have node or link cells as orthogonal or diagonal neighbors are called border cells. The perimeter of a barrier is the number of border cells.

Note that all border cells of a barrier are orthogonally adjacent. One of the main contributions of this paper is that paths in the cell structure using only node and link cells have equivalent paths in the communication network. We define such cell-based paths formally as follows.

Definition 2 A cell path (C_1, \dots, C_m) consists of node or link cells, such that C_i and C_{i+1} are orthogonally neighboring cells.

We call an edge the *implicant* for a link cell if it intersects with the link cell. Similarly we call a node the implicant of a link cell, if all points in the link cell have maximum distance $r/2$ to the node. We show that paths in the network and cell routes are essentially equivalent. The equivalence is based on the *connectivity property* of a link cell: A link cell C fulfills the connectivity property, if there is a direct connection (i.e. the distance is less or equal to r) between all the implicants for C . In case of an edge being the implicant this is required for at least one of the incident nodes.

Lemma 1 If $\ell \leq \sqrt{\frac{3}{8}} r$ then all link cells fulfill the connectivity property.

Proof: In the case of two nodes causing the link cell, this follows by the triangle inequality, since all points of the link cell have maximum distance $r/2$ to the nodes. In the case of a node and an edge as implicants, there is a point in the link cell, which has at most distance $r/2$ to one of the nodes of the edge and the connectivity property follows.

If two edges are implicants for the link cell then the question is: How must the edges be placed such that the minimum distance between two nodes incident to different edges is maximal? In a worst case a node v incident to one edge e_1 is equally distant to both nodes of the other edge e_2 , i.e. these three end points form an isosceles triangle. Since there exist two points p_1 on e_1 and p_2 on e_2 with $\|p_1 - p_2\| \leq \sqrt{2} \ell$ (causing e_1 and e_2 being implicants of the same cell), the distance between v and one point on e_2 (the mid point) is at most $\|p_1 - p_2\| \leq \sqrt{2} \ell$. Then, the maximum distance between v and $u \in e_2$, which should not exceed r , is $\sqrt{2\ell^2 + r^2/4}$. This leads to the bound $\ell \leq \sqrt{3/8} r$. ■

Theorem 2 If $\ell \leq \frac{1}{2}(\sqrt{3}-1) r$, then paths in the network and cell routes in the cell structure are equivalent up to a constant factor:

- 1.) Every path P of the network can be replaced with a cell path P' of length at most $2\lceil\frac{r}{\ell}\rceil|P| + 1$ containing all nodes of the path P , where $|P|$ is the number of nodes on the path.
- 2.) For every cell path P' there exists a path P of length at most $2|P'|$, where $|P'|$ is the number of cells on the path.

Proof: 1.) We substitute the path $P = (u_1, \dots, u_n)$ as follows. For each edge $e_i := (u_i, u_{i+1})$ we replace the nodes u_i and u_{i+1} by their node cells and add the link cells that are intersected by e_i . So we obtain node cells which are connected via link cells. If we measure the length of each edge e_i by using the Manhattan distance between u_i and u_{i+1} based on the cell size, we obtain $2\lceil\frac{r}{\ell}\rceil|P| + 1$ cells for the whole path.

2.) For two orthogonally neighboring cells on a cell path the connectivity of the implicants has to be guaranteed. There are three possible transitions between such adjacent cells:

- Node cell to node cell: Connectivity between the nodes is guaranteed, if $\ell \leq r/\sqrt{5}$.
- Node cell to link cell (or vice versa): Consider a node v in the node cell. There exists a point p on the edge e that is implicant of the link cell with $\|v - p\| \leq \sqrt{5}\ell$. The distance between v and the nearer node of e is reached, if both nodes of e are equally distant to v . If $r \geq \sqrt{r^2/4 + 5\ell^2} \Rightarrow \ell \leq \sqrt{3/20}r$, then v is connected to at least one node of e .
- Link cell to link cell: The following considerations refer to the connectivity of the implicant nodes and edges:
 - a) Implicant edges: Consider two edges e_1 and e_2 , each of them implicant of one of the two link cells. Then, there are two points p_1 on e_1 and p_2 on e_2 with $\|p_1 - p_2\| \leq \sqrt{5}\ell$. A similar argument as for node cell and link cell leads to $\ell \leq \sqrt{3/20}r \approx 0.387r$ to guarantee connectivity, i.e. $\exists u \in e_1, v \in e_2 : \|u - v\| \leq r$.
 - b) Implicant edge and implicant node: Consider a node v that is implicant of the cell C and an edge e that is implicant of the neighboring cell. The distance between v and any point p_1 in C is at most $r/2$. Thus, there is a point p_2 on e with $\|p_1 - p_2\| \leq \ell$. The connection of v and one of the nodes incident to e is guaranteed, if $r^2 \geq (\frac{r}{2} + \ell)^2 + (\frac{r}{2})^2 \Rightarrow \ell \leq \frac{1}{2}(\sqrt{3} - 1)r \approx 0.366r$.
 - c) Implicant nodes: Since the distance of each point inside the cell to the implicant node is at most $r/2$, the two implicant nodes of neighboring cells are always connected.

The transitions between the cells on the cell path P' are transformed as follows: We start with $P = \{u_1\}$, where u_1 is one node in the first node cell of P' . A node u is responsible for a cell C , if u is inside C or implicant of C or incident to an edge that is implicant of C . We substitute a transition from a cell C_i to C_{i+1} as follows: If C_i is a node cell containing the last node of P , then we add a node to P , that is responsible for C_{i+1} . If C_i is a link cell for which the last node u_i of P is an implicant, then u cannot always reach a node u_{i+1} that is responsible for C_{i+1} . From the considerations above and the connectivity property we know that there is another node v responsible for C_i that is in reach of u_i and u_{i+1} . We add v and u_{i+1} to P . So we add at most two nodes to P for one cell of P' . ■

For routing messages we abstract from the actual network topology and consider only the cells. Passing messages from one cell to another is implemented on a lower layer using the responsible nodes. If these responsibilities are optimized, most of the time the routing between cells does not imply actual message transmission. A responsible node can forward a message (virtually) from cell to cell until a barrier prevents further forwarding or his visibility range is exceeded.

Remember that barrier cells are empty cells which obstruct routing. One may extend this notation also to barriers in the original meaning like walls obstructing radio signals. Our concept of barrier cells is compatible to this notion as long as the connectivity property of link and node cells is satisfied. In Figure 4 an example is given where this property is violated by barriers obstructing radio signals. In such a case the communication graph is not planar anymore and position-based traversal or face routing algorithms fail. A greedy strategy would run into a local minimum (node w in the figure), then a right-hand traversal gets into a cycle.

<ol style="list-style-type: none"> 1. $i \leftarrow 1$; 2. $d \leftarrow 2^i$; 3. start greedy/recovery strategy with time-to-live = $d^{3/2}$; 4. if the target is not reached then start flooding with time-to-live = d; 5. if the target is not reached then $i \leftarrow 2i$; goto line 2;

Figure 5: The Alternating Algorithm

Regarding traffic, the optimal offline solution needs h messages, because the structure of the complete network is known to the offline algorithm. In this case the costs for the exploration of the network are neglected. But any online algorithm can be forced to pay these exploration costs in a worst case scenario by a proper placement of barriers. Therefore, we consider the quotient of the traffic produced by the online algorithm and the online lower bound for traffic. We call this measure *comparative ratio*. This term was introduced by Koutsoupias and Papadimitriou [12] in a more general context.

Definition 4 Let h be the length of the shortest barrier-free path between source and target and p the sum of the perimeters of all barriers in the network. A routing algorithm has comparative traffic ratio $\mathcal{R}_{Tr} := M/(h + p)$ if the algorithm needs M messages to deliver a message to the target.

The efficiency of a routing algorithm regarding both time and traffic is expressed by the *combined comparative ratio*:

Definition 5 The combined comparative ratio is the maximum of the competitive time ratio and the comparative traffic ratio: $\mathcal{R}_c := \max\{\mathcal{R}_t, \mathcal{R}_{Tr}\}$

5.3 A Routing Algorithm with Sub-Linear Comparative Ratio

There are two basic routing strategies: single-path or multi-path. The best single-path strategy needs $\mathcal{O}(h + p)$ time and traffic, where p is bounded by an area of size h^2 . One can easily see that in a worst case scenario any algorithm can be forced to traverse a labyrinth of barriers which contributes with a time overhead of $\mathcal{O}(p)$. Though the single-path strategy has optimal comparative traffic ratio, the competitive time ratio is $\mathcal{O}(h)$. The best multi-path strategy is flooding. Assumed we know the right flooding depth, this strategy is time-optimal. But we use always traffic h^2 , even if there are small barriers ($p \approx h$), which gives a comparative traffic ratio of $\mathcal{O}(h)$.

In the single-path case, we allow the algorithm to spend time for exploration, in the multi-path case we do the exploration in parallel. Both strategies have a combined comparative ratio of $\mathcal{O}(h)$.

We will see that we can achieve a sub-linear combined comparative ratio with a strategy that combines a single-path strategy with flooding. For the single-path strategy we use a combination of greedy forwarding and recovery for traversing barriers (cf. Section 2): The greedy rule is to follow a guide line connecting source and target. If this is not possible because of a barrier, then the barrier is traversed by applying the right-hand rule until the guide line is reached again at another position which is nearer to the target (progress condition). For the multi-path strategy we use flooding. Both strategies are applied with a search depth restriction (time-to-live). We start with a small search depth and repeat the greedy strategy and flooding alternately with increased search depth until the target can be reached (see Figure 5).

It seems that we combine the disadvantages of both flooding and the greedy strategy. We will see that this algorithm has a better combined competitive ratio.

Theorem 3 The Alternating Algorithm needs time $\mathcal{O}(h^{3/2})$ and produces traffic $\mathcal{O}(\min\{h^2, h^{3/2} + p\})$.

Proof: We distinguish between two cases:

Case 1: The target is reached while applying the greedy strategy (Fig. 5, line 3). The greedy strategy uses

greedy paths (segments on the guide line) and recovery paths (along the border of a barrier). As the greedy rule is only applied if a progress condition is fulfilled, the length of the greedy paths is bounded by h . In recovery mode at most all the barriers have to be traversed, so the length of the recovery paths is bounded by p . The target is reached in $h + p$ steps, i.e. within $\frac{2}{3} \log(h + p)$ iterations (because $2^{3i/2} \leq h + p$). The search depth for flooding in the previous iteration has been $2^{2/3 \cdot \log(h+p)-1} = \frac{1}{2}(h+p)^{2/3}$. The target could not be reached while flooding, but flooding yields optimal paths. Hence, h is larger than the flooding depth: $\frac{1}{2}(h+p)^{2/3} \leq h \Rightarrow h+p = \mathcal{O}(h^{3/2})$. The traffic is bounded by $\sum_{i=1}^{\frac{2}{3} \log(h+p)} 2^{3i/2} + 2^{2i} = \mathcal{O}((h+p)^{4/3})$

Case 2: The target is reached while flooding (Fig. 5, line 4). It takes $\log h$ iterations to approach the target through incremental flooding. In each iteration there is an additional delay caused by the greedy strategy. So, the time is bounded by $\sum_{i=1}^{\log h} 2^{3i/2} + 2^i = \mathcal{O}(h^{3/2})$. The target is only reached by flooding (with depth h), if the greedy strategy is forced to make a detour. But any barrier with perimeter smaller than $h^{1/2}$ cannot prevent a message in recovery mode from returning to the guide line and then reaching the target. That means reaching the target by flooding occurs only if barriers of size $p \geq h^{1/2}$ are present. If there are smaller barriers ($p < h^{1/2}$) then the target can be reached in greedy mode with traffic $\mathcal{O}(h + p)$ for the greedy strategy and $\mathcal{O}(h^{3/2})$ for the flooding strategy. Otherwise ($p \geq h^{1/2}$) the traffic is bounded by $\sum_{i=1}^{\log h} 2^{3i/2} + 2^{2i} = \mathcal{O}(h^2)$. ■

Corollary 1 *The Alternating Algorithm has a combined comparative ratio of $\mathcal{O}(\sqrt{h})$.*

Proof: Follows by the definition of competitive time ratio and comparative traffic ratio and Theorem 3. ■

References

- [1] K. Alzoubi, X.-Y. Li, Y. Wang, P.-J. Wan, and O. Frieder. Geometric spanners for wireless ad hoc networks. *IEEE Transactions on Parallel and Distributed Systems*, 14(4):408–421, 2003.
- [2] R. Bar-Yehuda, O. Goldreich, and A. Itai. On the time-complexity of broadcast in multi-hop radio networks: An exponential gap between determinism and randomization. *Journal of Computer and System Sciences*, 45(1):104–126, Aug. 1992.
- [3] A. Borodin and R. El-Yaniv. *Online Computation and Competitive Analysis*. Cambridge University Press, 1998.
- [4] P. Bose, P. Morin, I. Stojmenovic, and J. Urrutia. Routing with guaranteed delivery in ad hoc wireless networks. *Wireless Networks*, 7(6):609–616, 2001.
- [5] Q. Fang, J. Gao, and L. J. Guibas. Locating and bypassing routing holes in sensor networks. In *Proc. of the 23rd Conference of the IEEE Communications Society (INFOCOM'04)*, Mar. 2004.
- [6] J. Gao, L. J. Guibas, J. E. Hershberger, L. Zhang, and A. Zhu. Geometric spanner for routing in mobile networks. In *Proc. of the 2nd Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'01)*, pages 45–55. ACM, Oct 2001.
- [7] S. Giordano and I. Stojmenovic. Position based routing algorithms for ad hoc networks: A taxonomy. In X. Cheng, X. Huang, and D. Du, editors, *Ad Hoc Wireless Networking*, pages 103–136. Kluwer, 2004.
- [8] M. Heissenbüttel and T. Braun. A novel position-based and beacon-less routing algorithm for mobile ad-hoc networks. In *Proc. of the 3rd IEEE Workshop on Applications and Services in Wireless Networks (ASWN' 03)*, pages 197–209, Bern, Switzerland, July 2003.
- [9] T.-C. Hou and V. Li. Transmission range control in multihop packet radio networks. *IEEE Transactions on Communications*, 34(1):38–44, January 1986.
- [10] J. W. Jaromczyk and G. T. Toussaint. Relative neighborhood graphs and their relatives. *Proceedings of the IEEE*, 80:1502–1517, 1992.
- [11] B. Karp and H. T. Kung. GPSR: greedy perimeter stateless routing for wireless networks. In *Mobile Computing and Networking*, pages 243–254, 2000.
- [12] E. Koutsoupias and C. H. Papadimitriou. Beyond competitive analysis. *SIAM Journal on Computing*, 30(1):300–317, 2000.
- [13] E. Kranakis, H. Singh, and J. Urrutia. Compass routing on geometric networks. In *Proc. of the 11th Canadian Conference on Computational Geometry*, pages 51–54, Vancouver, August 1999.
- [14] F. Kuhn, R. Wattenhofer, and A. Zollinger. Asymptotically optimal geometric mobile ad-hoc routing. In *Proc. of the 6th Int. Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, pages 24–33, 2002.
- [15] M. Mauve, J. Widmer, and H. Hartenstein. A survey on position-based routing in mobile ad hoc networks. *IEEE Network Magazine*, 15(6):30–39, November 2001.

- [16] H. Takagi and L. Kleinrock. Optimal transmission ranges for randomly distributed packet radio terminals. *IEEE Transactions on Communications*, 32(3):246–257, March 1984.
- [17] Y. Wang and X.-Y. Li. Distributed spanner with bounded degree for wireless ad hoc networks. In *Proc. of the 16th Int. Parallel and Distributed Processing Symposium (IPDPS'02)*, page 120. IEEE Computer Society, 2002.