

# Experimental Analysis of Adjustable Sectorized Topologies for Static Ad Hoc Networks

[Extended Abstract]

Klaus Volbert\*

Heinz Nixdorf Institute, Paderborn University

## ABSTRACT

We consider static wireless networks where the area around each device is subdivided into a fixed number of equal sectors or cones. Each node has one transmitter per sector and can adjust its transmitting power to send out data in each sector separately. In general, it is assumed that the sectors of the nodes underly a fixed orientation or that they are oriented as a result of the movement of nodes. We assume an extended model and allow each node to adjust the orientation of its senders not in dependence from any other conditions. We introduce new optimization tasks and present algorithms which improve the stretch factors of known sectorized topologies only by adjusting the orientation of some sectors of the nodes. Further, we present experimental results on random vertex sets to investigate the characteristics of these topologies under the extended model with regard to energy consumption, given by the so-called power spanner property, and to congestion, given by the (weak) spanner property. We take also interferences into account and measure the degree of the nodes and other statistical data.

In addition, we present new algorithms to calculate exactly the power stretch factor and the weak stretch factor of a graph  $G = (V, E)$  in time  $O(|V|^2 \log |V| + |V||E|)$ . Hence, we can determine these factors for sparse graphs, considered in this work, in time  $O(|V|^2 \log |V|)$ . Applying this yields a lower bound for an optimal orientation of all senders, e.g., with regard to energy consumption, that can be computed in polynomial time. We use this bound to analyze the simulation results.

Our extensive experimental evaluation of three sectorized topologies on random vertex sets under different sector alignments show that we can improve the known stretch factors and that the considered topologies perform well on “normal” vertex sets where the nodes are placed uniformly at random.

---

\*Institute of Computer Science, kvolbert@upb.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).

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

DIALM-POMC'04, October 1, 2004, Philadelphia, Pennsylvania, USA.  
Copyright 2004 ACM 1-58113-921-7/04/0010 ...\$5.00.

## Categories and Subject Descriptors

C.2.1 [Computer-Communication Networks]: Network Architecture and Design; G.2.2 [Discrete Mathematics]: Graph Theory

## General Terms

Algorithms, Experimentation, Performance, Theory

## Keywords

Ad hoc Networks, Topology Control, Algorithms, Spanner

## 1. INTRODUCTION

In this work we consider a static ad hoc network consisting of  $n$  nodes distributed in the Euclidean plane. We model this network by a graph  $G = (V, E)$  where  $V \subseteq \mathbb{R}^2$ . Each node has  $k \in \mathbb{N}$  transmitting devices that allow sending data in *sectors* (sectorized model). We divide the area around a node in  $k$  equal sectors of angle  $2\pi/k$  and use one transmitting device per sector. We assume that there is given only one transmitting frequency and that each node can adjust the transmitting power of its devices. The power needed to support a link between two nodes  $u$  and  $v$  is  $\|u, v\|_2^\delta$  where  $\|u, v\|_2$  is the Euclidean distance between  $u$  and  $v$  and  $\delta$  is a real constant between 2 and 6 or even 8 dependant on the wireless transmission environment. The sectorized model is not new and a lot of network topologies has been investigated, e.g., in [33, 18, 32, 15, 28, 31]. Excellent surveys are presented by R. Rajaraman [23] and X.-Y. Li [17, 16]. In [7, 30, 25, 8] we showed besides theoretical results and simulations that the sectorized model is also suitable in practical situations. We have developed a communication module for a mini robot that can transmit and receive data in eight sectors using infrared light with variable transmission distances up to one meter. Nowadays communication techniques make it also possible to build up such a wireless network using directional antennas, adaptive antennas or beam-forming antennas. There exists a lot of (practical) workings concerning wireless networks using those antennas, e.g., [11, 2, 22, 24, 29, 3, 9, 14].

Note, that in all existing theoretical workings concerning topology control in wireless networks it is assumed that the orientation of the senders is fixed or is affected by the movement of a node. In this work, we introduce a new dimension, since we allow each node to control the orientation of its senders/sectors to better transmissions concerning energy consumption or to decrease the number of interferences. To the best of our knowledge, this paper is the first paper concerning topology control in wireless networks in which this additional freedom for the devices is studied.

Our model also diverges from the standard model coming from computational geometry where the sectors on all nodes are fixed

and equal oriented. Under the new communication model we perform an extensive experimental evaluation of three known sectorized graphs. The first is the so-called Yao-graph (or  $\theta$ -graph or sector graph or YG) introduced to computational geometry by Yao [34]. Here, in each sector a node establishes a communication link to the nearest node in this sector. Note, that the definition of nearest neighbor differs in some approaches because of additional side conditions. We use the definition where the nearest node in a sector is given by the node with the shortest Euclidean distance. In the SparsY-graph (or sparsified Yao-graph or YY) each node allows only the shortest incoming communication link [7, 18]. Finally, in the SymmY-graph (or symmetric Yao-graph or YS) only links are established that are symmetric [7, 18]. A fixed orientation of the sectors has the advantage that the construction of the Yao-graph and its variants is possible centrally in time  $O(n \log n)$  where  $n$  is the number of nodes. In our model, if the orientation can be arbitrary, there is only the naive algorithm which needs time  $O(n^2)$ .

A challenge in wireless networks is to spare energy and to avoid congested communication links having the number of interferences so low as possible. From a graph theoretic point of view this means that we want to build up a graph that gives at least one path between two arbitrary nodes with low energy-cost, low interferences and low congestion. For this purpose we consider spanners, weak spanners and power spanners. One aim of this work is to measure and compare the stretch factors given by the definitions of these structures. On the other hand we use the new communication model to improve the stretch factors and show the improvement experimentally. We developed a testbed [13] which can also be used via World Wide Web and we present new algorithms we used to perform our simulations.

The aim of this work is not to go into technical details, but more on presenting first algorithms, ideas and experimental results for the extended communication model. It is more a theoretical study on the question: Assumed each node can adjust the orientation of its senders, what is possible concerning energy consumption? In contrast to the workings investigating wireless networks using directional antennas we consider three known topologies under the extension that each node can adjust its sectors/senders.

The rest of this paper is organized as follows. In Section 2 we give the notations used in this work and review the formal definitions of the Yao-graph, the SparsY-graph and the SymmY-graph as well as the definitions of a spanner, a weak spanner and a power spanner. Further, we present the theoretical background and sum up known results. In Section 3 we present algorithms to determine exactly the power stretch factor and the weak stretch factor of a given graph  $G = (V, E)$ . We show that this can be done in general in time  $O(|V|^2 \log |V| + |V||E|)$  and in time  $O(|V|^2 \log |V|)$  if we consider the Yao-graph and its variants. In Section 4 we introduce the extended communication model and present first algorithms to orientate sectors. We can improve the length stretch factor of the Yao-graph by a factor of 2 and the power stretch factor by a factor of  $2^\delta$ , where  $\delta$  is a constant that depends on transmitting characteristics, if it is allowed to adjust the sectors for all nodes on a complete path from a source node to a destination node. In Section 5 we present a lower bound for an optimal orientation of the sectors. We construct a graph in which we can calculate lower bounds for the stretch factors of the Yao-graph under any orientation of the sectors in polynomial time. In Section 6 we present the results of our extensive experimental evaluation. Finally, we conclude our work in Section 7 and discuss open problems and research directions.

## 2. PRELIMINARIES

Let  $V$  be a set of  $n$  points or nodes in  $\mathbb{R}^2$ . We denote by  $\|u, v\|_2$

the Euclidean distance between two points  $u, v \in \mathbb{R}^2$ . We consider geometric graphs where the nodes are given by points in  $\mathbb{R}^2$  and the cost of an edge between two nodes is given by its Euclidean length or its Euclidean length to a  $\delta \geq 2$ . Taking the Euclidean length to a  $\delta \geq 2$  is motivated by the fact that wireless transmitting over a distance  $d$  needs energy  $d^\delta$  for a  $\delta \geq 2$ . Let  $G = (V, E)$  be a directed geometric graph. A path  $P_G(u, v)$  from  $u$  to  $v$  in  $G$  is a finite sequence  $P_G(u, v) := (u = u_1, \dots, u_l = v)$  of vertices  $u_i \in V$  such that  $(u_i, u_{i+1}) \in E$  for all  $i \in \{1, \dots, l-1\}$ . The hop-length of this path is given by the number of edges lying on this path. We denote it by  $|P_G(u, v)| := l - 1$ . The Euclidean length of  $P_G(u, v)$  is given by  $\|P_G(u, v)\| := \|P_G(u, v)\|_2 := \sum_{i=1}^{l-1} \|u_i, u_{i+1}\|_2$ . For  $\delta \geq 2$  we denote by  $\|u, v\|^\delta := \|u, v\|_2^\delta$  the  $\delta$ -cost or energy-cost of an edge  $(u, v)$  and by  $\|P_G(u, v)\|^\delta := \|P_G(u, v)\|_2^\delta := \sum_{i=1}^{l-1} \|u_i, u_{i+1}\|_2^\delta$  the  $\delta$ -cost or energy-cost of a path  $P_G(u, v)$ . We can also have a path  $Q(u, v)$  from  $u$  to  $v$  that contains edges which are not necessarily in  $G$ . Then we leave out the index  $G$ . Now, we give the formal definition of the Yao-graph, the SparsY-graph and the SymmY-graph. For further details see, e.g., [7, 33].

**DEFINITION 1.** *Let  $V \subseteq \mathbb{R}^2, k \in \mathbb{N}$  and  $G = (V, E)$  be a geometric graph. The area around a node  $u \in V$  is divided into  $k$  non-overlapping sectors or cones of angle  $\theta = 2\pi/k$ . We denote the sector of  $u$  in which a node  $v \in V$  lies by  $\triangleleft(u, v)$ .*

- *$G$  is the Yao-graph of  $V$ , if  $E := \{(u, v) \mid \forall w \neq u : \triangleleft(u, v) = \triangleleft(u, w) \Rightarrow \|u, v\|_2 \leq \|u, w\|_2\}$ .*
- *$G$  is the SparsY-graph of  $V$ , if  $E := \{(u, v) \in E(\overline{G}) \mid \forall w \neq u : ((w, v) \in E(\overline{G}) \text{ and } \triangleleft(v, w) = \triangleleft(v, u)) \Rightarrow \|u, v\|_2 \leq \|u, w\|_2\}$  where  $\overline{G}$  denotes the Yao-graph of  $V$ .*
- *$G$  is the SymmY-graph of  $V$ , if  $E := \{(u, v) \in E(\overline{G}) \mid (v, u) \in E(\overline{G})\}$  where  $\overline{G}$  denotes the SparsY-graph of  $V$ .*

It follows directly from the definition that the SymmY-graph is a subgraph of the SparsY-graph, and that the SparsY-graph is a subgraph of the Yao-graph. As already mentioned it is known that all three graphs can be constructed locally, e.g., using a sweepline algorithm, in time  $O(n \log n)$  if the orientation of the sectors is fixed and equal on all nodes (see [5, 6]). Now, we review the definition of a spanner, a weak spanner [5, 6] and a power spanner [7, 33].

**DEFINITION 2.** *Let  $G = (V, E)$  be a geometric graph consisting of  $n$  nodes in  $\mathbb{R}^2$  and  $c \geq 1$ .*

- *$G$  is a  $c$ -spanner, if for all  $u, v \in V$  there is a path  $P_G(u, v)$  in  $G$  with  $\|P_G(u, v)\| \leq c \cdot \|u, v\|_2$ .*
- *$G$  is a weak  $c$ -spanner, if for all  $u, v \in V$  there is a path  $P_G(u, v)$  in  $G$  with  $\|u, u_i\|_2 \leq c \cdot \|u, v\|_2$  for all  $i \in \{1, \dots, |P_G(u, v)|\}, u_i \in P_G(u, v)$ .*
- *$G$  is a  $(c, \delta)$ -power spanner for a  $\delta \geq 2$ , if for all  $u, v \in V$  there is a path  $P_G(u, v)$  in  $G$  with  $\|P_G(u, v)\|^\delta \leq c \cdot \min_{Q(u, v)} \|Q(u, v)\|^\delta$ .*
- *$G$  is a  $c$ -power spanner, if  $G$  is a  $(c, \delta)$ -power spanner for all  $\delta \geq 2$ .*

*The factor  $c$  is called length stretch factor, weak stretch factor or power stretch factor, respectively.*

Sometimes we shorten the notions and omit constant factors. So, if we say that a graph is a good spanner then it is meant that there exists a constant  $c$  such that this graph is a  $c$ -spanner. A graph is a bad spanner if there exists such a  $c$  only depending on the number of nodes in the graph. Recently, we investigated the relation between spanners, weak spanners and power spanners, see [27].

Topology	Connectivity	Weak Spanner	Power Spanner	Spanner
Yao	$k \geq 4$	$k = 4, c = \sqrt{3 + \sqrt{5}}$ $k \geq 6, c = \max\{\sqrt{1 + 48 \sin^4(\pi/k)}, \sqrt{5 - 4 \cos(2\pi/k)}\}$	$k > 6, c = \left(\frac{1}{1 - 2 \sin \frac{\pi}{k}}\right)^\delta$ $k > 6, c = \frac{1}{1 - (2 \sin \frac{\pi}{k})^\delta}$	$k > 6, c = \frac{1}{1 - 2 \sin \frac{\pi}{k}}$
SparsY	$k > 6$	$k > 6, c = \frac{1}{1 - 2 \sin \frac{\pi}{k}}$	$c$ exists for $k > 6$	OPEN
SymmY	$k > 6$	—	—	—

**Table 1: Elementary Graph Properties**

## 2.1 Known Results

There exists a lot of workings dealing with topology control for wireless networks, for surveys see [23, 17, 16]. In the following we try to summarize the most important results with regard to our investigations. The Yao-graph was first introduced to computational geometry by Yao [34]. In Table 1 we give an overview about some elementary graph properties. The weak stretch factor of the Yao-graph for  $k = 4$  was proven in [5], for  $k \geq 6$  in [6]. The proofs of the power spanner factors of the Yao-graph can be found in [7, 18]. The spanner property of the Yao-graph comes from [26]. In [7] it was shown that the SparsY-graph is a good weak spanner and in [12, 27] it was proven that it is also a good power spanner for  $k > 6$ . It is still unknown whether the SparsY-graph is a good spanner. About the SymmY-graph it is known that it fulfills none of the three spanner definitions, but anyway, it is connected [7].

It is easy to construct examples for  $k \leq 3$  such that the Yao-graph with  $k$  sectors is not connected. We can also show that there exist placings such that the SparsY-graph and the SymmY-graph are not connected using less than 6 sectors.

In different workings it has been suggested to use the Yao-graph and its variants as an underlying network structure for a wireless network. Hence, there are some results about the suitability of these graphs as communication networks. The SparsY-graph and the SymmY-graph were introduced because of the high non-avoidable in-degree of the Yao-graph. The Yao-graph can have  $n - 1$  incoming edges and in wireless networks this is equivalent to a high interference number of  $n - 1$  [7]. The extensions of the Yao-graph have constant out- and in-degree of at most  $k$ . Hence, these structures are free of interferences theoretically. In [25] it was shown that the influence of interferences in these graphs in reality is not so hard as assumed in most of the papers. Therefore, we can also assume that the SparsY-graph and the SymmY-graph are nearly optimal topologies with regard to interferences practically. As shown in Table 1 the Yao-graph is a good spanner and it is known that good spanners are good choices for topologies minimizing energy consumption [7]. It was first mentioned in [33] that the SparsY-graph could be a good spanner but to the best of our knowledge there exists no proof for this or a counter-example in literature. The problem is still open and it seemed to be very hard to solve it. On the other hand it has been shown that the SparsY-graph is a power spanner and so it is suitable for a topology that should conserve energy [12, 7, 27]. As already mentioned the SymmY-graph is not a good topology conserving energy in general.

Another critical issue in designing a topology for a wireless network is routing time. In [1, 19] routing time has been defined in terms of congestion and dilation. For some placings the Yao-graph and its variants have dilation of  $\Omega(n)$ . But there exist workings that suggest hierarchical definitions (see [10]) which can be applied here to get a dilation of  $O(\log n)$ . Note, that we have a trade-off between congestion and dilation [20]. The main focus in this work

lies on congestion. It was shown in [19] that a good weak spanner allows a topology with low congestion. More formally, a weak  $c$ -spanner with interference number  $I$  approximates a congestion-optimal network by a factor of  $O(I \log n)$ . A corollary is that the congestion approximation factor of the SparsY-graph is given by  $O(\log n)$ .

In this work we want to evaluate the length stretch factor, the power stretch factor and the weak stretch factor experimentally. In literature, there exist already experimental results concerning these graphs on random vertex sets, e.g. [33], but they are not comprehensive and there it is not allowed to adjust the orientation of the sectors.

## 3. CALCULATING STRETCH FACTORS

In this section we present new algorithms we need to efficiently compute the length stretch factor, the power stretch factor and the weak stretch factor. In [21] algorithms are presented for approximating the length stretch factor, but in this work we want to compute the exact values. Let again  $G = (V, E)$  be a geometric graph and  $n := |V|$  and  $m := |E|$ . The length stretch factor can be calculated by solving the all-pairs-shortest-path problem (APSP). We check all  $n(n - 1) = O(n^2)$  directed shortest paths and determine the maximum length stretch factor. Hence, running Dijkstra's algorithm using Fibonacci heaps gives the length stretch factor of  $G$  in time  $O(n^2 \log n + nm)$  (see e.g., [4] for further details). The power stretch factor can be calculated in the same way if we use  $\|e\|_2^\delta$  as the cost of an edge  $e$  instead of  $\|e\|_2$ . The attentive reader might have observed that this computation would not exactly match the definition of a power spanner, but Lemma 1 shows that both are in fact equivalent.

**LEMMA 1.** *Let  $G = (V, E)$  be a geometric graph in which for all  $u, v \in V$  there exists a path  $P_G(u, v)$  with  $\|P_G(u, v)\|^\delta \leq c \cdot \|u, v\|_2^\delta$ . Then  $G$  is a  $(c, \delta)$ -power spanner.*

**PROOF.** We consider two arbitrary nodes  $u, v \in V$ . Let  $Q_{\text{OPT}} := \min_{Q(u, v)} \|Q(u, v)\|^\delta$  be an optimal path for the  $\delta$ -cost from  $u$  to  $v$  of length  $l$ . Each edge  $(u_i, u_{i+1})$  on this path can be replaced by a path in  $G$  with cost at most  $c \cdot \|u_i, u_{i+1}\|_2^\delta$ . The concatenation of all these new paths in  $G$  yields to a path from  $u$  to  $v$  in  $G$  with cost at most  $\sum_{i=1}^{l-1} c \cdot \|u_i, u_{i+1}\|_2^\delta = c \cdot \|Q_{\text{OPT}}\|^\delta$ .  $\square$

The reverse direction is obvious. In Figure 1 you see the main part of our algorithms based on Dijkstra's algorithm and on the equivalent power spanner definition. An exact calculation of the weak stretch factor seems to be more complicated since we know nothing about the length of a path in a weak spanner. Nevertheless, we can prove that we can also compute the weak stretch factor in the time needed by Dijkstra's algorithm for solving the APSP problem. The main idea of the algorithm is presented in Figure 1.

<pre> POWERSTRETCH(<math>G = (V, E), V \subseteq \mathbb{R}^2, s \in V</math>) 1  for all <math>v</math> in <math>V</math> 2  do <math>\text{dist}(v) \leftarrow \infty, \text{pre}(v) \leftarrow \emptyset</math> 3  <math>\text{dist}(s) \leftarrow 0, \text{pre}(s) \leftarrow s, Q \leftarrow V</math> 4  while <math>Q \neq \emptyset</math> 5  do <math>u \leftarrow \text{EXTRACTMIN}(Q)</math> 6  for each <math>v \in \text{Adj}[u]</math> 7  do if <math>\text{dist}(u) + \ u, v\ _2^2 &lt; \text{dist}(v)</math> 8     then <math>\text{dist}(v) \leftarrow \text{dist}(u) + \ u, v\ _2^2, \text{pre}(v) \leftarrow u</math> 9  power-stretch <math>\leftarrow 0</math> 10 for all <math>v</math> in <math>V \setminus \{s\}</math> 11 do if <math>\text{dist}(v)/\ s, v\ _2^2 &gt; \text{power-stretch}</math> 12    then power-stretch <math>\leftarrow \text{dist}(v)/\ s, v\ _2^2</math> </pre>	<pre> WEAKSTRETCH(<math>G = (V, E), V \subseteq \mathbb{R}^2, s \in V</math>) 1  for all <math>v</math> in <math>V</math> 2  do <math>\text{dist}(v) \leftarrow \infty, \text{pre}(v) \leftarrow \emptyset</math> 3  <math>\text{dist}(s) \leftarrow 0, \text{pre}(s) \leftarrow s, Q \leftarrow V</math> 4  while <math>Q \neq \emptyset</math> 5  do <math>u \leftarrow \text{EXTRACTMIN}(Q)</math> 6  for each <math>v \in \text{Adj}[u]</math> 7  do if <math>\max\{\text{dist}(u), \ s, v\ _2\} &lt; \text{dist}(v)</math> 8     then <math>\text{dist}(v) \leftarrow \max\{\text{dist}(u), \ s, v\ _2\}, \text{pre}(v) \leftarrow u</math> 9  weak-stretch <math>\leftarrow 0</math> 10 for all <math>v</math> in <math>V \setminus \{s\}</math> 11 do if <math>\text{dist}(v)/\ s, v\ _2 &gt; \text{weak-stretch}</math> 12    then weak-stretch <math>\leftarrow \text{dist}(v)/\ s, v\ _2</math> </pre>
--	--

Figure 1: Calculating a) power stretch factor and b) weak stretch factor

We run the algorithms for all  $n$  nodes and then calculate the maximum over these  $n$  stretch values. In the following we show that both algorithms terminate and calculate the required values correctly. We denote by  $\text{dist}(u)$  the  $\delta$ -cost of a cheapest path found so far from  $s$  to  $u$  (power stretch) or the smallest radius found so far of a disk in which a path from  $s$  to  $u$  lies (weak stretch). Let  $d(u)$  be the value of  $\text{dist}(u)$  at the time when  $u$  is dequeued. In the following we consider a) the power stretch factor and b) the weak stretch factor at the same time.

LEMMA 2. *If  $u$  is dequeued before  $v$  we have  $d(u) \leq d(v)$ .*

PROOF. We assume that there exist two nodes  $u$  and  $v$  with  $u$  is dequeued before  $v$  and  $d(u) > d(v)$ . W.l.o.g. let  $v$  be the first such node. At the time  $u$  is dequeued, it holds that  $d(u) = \text{dist}(u) \leq \text{dist}(v)$ . By the definition  $d(u)$  is never changed again and hence  $\text{dist}(v)$  has to be decreased. This happens only if a node  $w$  is dequeued and a)  $\text{dist}(w) = \text{dist}(w) + \|w, v\|_2^2$  or b)  $\text{dist}(w) = \max\{\text{dist}(w), \|s, v\|_2\}$  is set. W.l.o.g. let  $w$  be the last such node. Then we have a)  $d(v) = d(w) + \|w, v\|_2^2$  or b)  $d(v) = \max\{d(w), \|s, v\|_2\}$ . By the choice of  $v$  it holds that  $d(u) \leq d(w)$ . Further, a)  $\|w, v\|_2^2 \geq 0$  or b)  $\max\{d(w), \|s, v\|_2\} \geq d(w)$ . In both cases, we get a contradiction:  $d(v) \geq d(u)$ .  $\square$

LEMMA 3. *Let  $Q$  be empty. Then a)  $\text{dist}(v) \leq \text{dist}(u) + \|u, v\|_2^2$  or b)  $\text{dist}(v) \leq \max\{\text{dist}(u), \|s, v\|_2\}$  for all  $(u, v) \in E$ .*

PROOF. It is easy to see that in both cases the claim holds at the time  $u$  is dequeued. After this point of time the claim breaks only, if  $\text{dist}(u)$  is decreased. Since  $\text{dist}(v)$  is never increased, this can only happen when a node  $w$  is dequeued and a)  $\text{dist}(u) = d(w) + \|w, u\|_2^2$  or b)  $\text{dist}(u) = \max\{d(w), \|s, u\|_2\}$  is set. From Lemma 2 it follows that  $d(w) \geq d(u)$ . With a)  $\|w, u\|_2^2 \geq 0$  or b)  $\max\{d(w), \|s, u\|_2\} \geq d(w)$  we get that  $\text{dist}(u) \geq d(u)$  and this is a contradiction.  $\square$

THEOREM 1. a)  $\text{POWERSTRETCH}(G, s)$  calculates the maximum power stretch factor over all paths starting at  $s$ .

b)  $\text{WEAKSTRETCH}(G, s)$  calculates the maximum weak stretch factor over all paths starting at  $s$ .

PROOF. We prove the claim by induction. Let  $v \in V$  and  $u := \text{pre}(v)$  be a predecessor of  $v$  on an optimal path from  $s$  to  $v$  concerning the a)  $\delta$ -cost or b) radius of a disk. By induction we can assume that  $\text{dist}(u)$  is the minimum a)  $\delta$ -cost of a path from  $s$  to  $v$  or b) radius of a disk which contains the whole path from  $s$  to  $v$ . Since a)  $\text{dist}(v) \leq \text{dist}(u) + \|u, v\|_2^2$  or b)  $\text{dist}(v) \leq \max\{\text{dist}(u), \|s, v\|_2\}$  (Lemma 3) we get that  $\text{dist}(v)$  is the a)  $\delta$ -cost of a cheapest path from  $s$  to  $v$  or b) minimum required radius of a disk such that the whole path from  $s$  to  $v$  lies in this disk. The last operation gives the maximum over all stretch values.  $\square$

## 4. ADJUSTING SECTOR ORIENTATIONS

In this section we present the extended communication model and present first heuristics to improve the length stretch factor, the power stretch factor and the weak stretch factor of the Yao-graph and its variants. We consider a static ad hoc network consisting of  $n$  nodes given by  $V \subset \mathbb{R}^2$ . Further, let  $k \in \mathbb{N}$  be the number of senders with that a node is equipped. In general, it is assumed that the orientation of the senders on a node is fixed or that the orientation is depending on node movements. We allow each node to adjust not only the transmission power of a sender but also the orientation. Therefore, the area around a node is subdivided into  $k$  sectors of equal size and each node can set an angle where the first sector begins. The orientation can be changed at any time and is given by a rotation function  $r : V \rightarrow [0, 2\pi/k)$ . Hence, e.g., sector  $i$  of a node  $u \in V$  goes from  $r(u) + (i-1) \cdot 2\pi/k$  to  $r(u) + i \cdot 2\pi/k$ . In a fixed model, assumed in computational geometry where e.g., the Yao-graph is used as a data structure to calculate properties like nearest neighbors efficiently, we have  $r(u) = 0$  for all  $u \in V$ . In [7, 25]  $r(u)$  is defined by the forefront of the used robot and so depends on the movement of a robot. Since we allow each node  $u$  to adjust  $r(u)$ , we got some interesting questions, e.g.:

- What is the best orientation of the sectors on the nodes? Which measure do we want to optimize?
- How can we compute an optimal orientation with regard to a specific stretch factor?
- What is the complexity of this problem?
- Is it possible to improve the stretch factors using simple algorithms?
- Is it possible to adjust the orientations such that the SparsY-graph will be a good spanner? (for fixed orientations it is still an open problem whether the SparsY-graph is a spanner or not)
- Is it possible to adjust some orientations such that the SymmY-graph will be a good spanner, a good weak spanner, or at least a good power spanner?

The aim of this work is not to answer all these questions, but to discuss some of them and to present the results of an extensive experimental evaluation of some new algorithms for the new communication model. We do not know how to find a best orientation for a node set with regard to a given measure in a time faster than the time needed by a naive approach trying all possible combinations which is possible in exponential time. Optimizing measures are, e.g., the stretch factors, the energy needed to keep up all network links, the number of links, or the in-degree of the nodes to reduce the number of interferences.

Now, we show that it is possible to improve the stretch factors by adjusting some orientations. We present first algorithms that use these results to improve all three stretch factors.

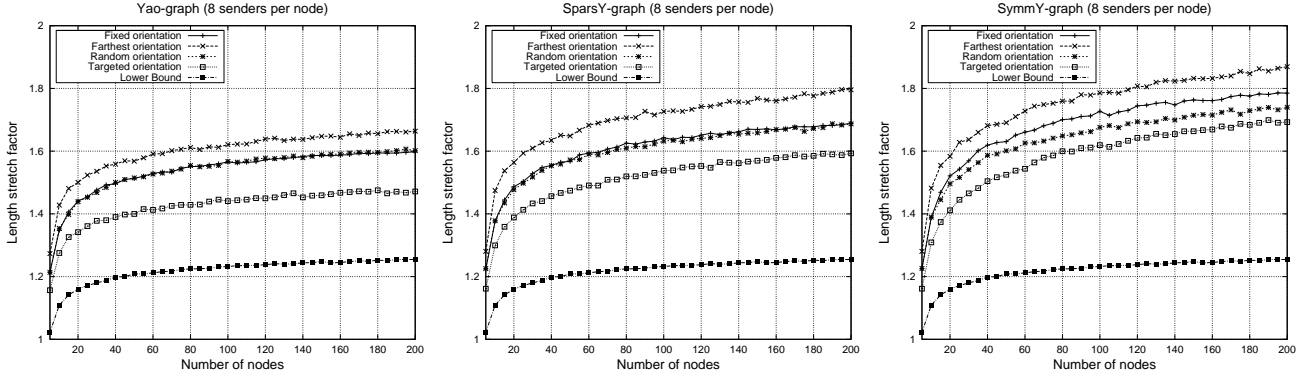


Figure 2: Maximum length stretch factors of Yao-graph, SparsY-graph and SymmY-graph

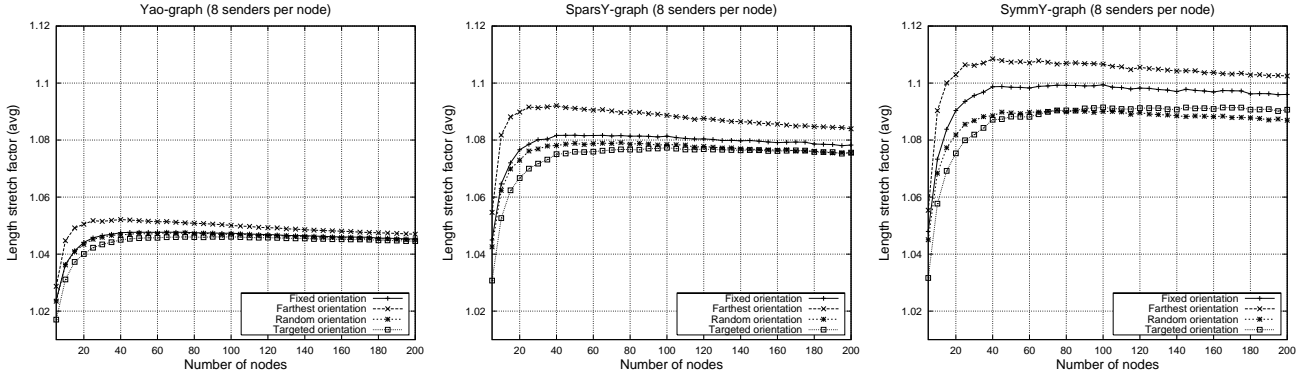


Figure 3: Average length stretch factors of Yao-graph, SparsY-graph and SymmY-graph

LEMMA 4. Let  $G = (V, E)$  be the Yao-graph and  $k \in \mathbb{N}$  the number of sectors per node. If it is allowed to adjust the orientation of the sectors on the nodes at any time and to establish a communication link between two nodes  $u, v \in V$ , then  $G$  is a  $c$ -spanner with length stretch factor  $c = \frac{1}{1 - \sin \pi/k}$  for  $k > 3$ .

PROOF. We consider two nodes  $u$  and  $v$  and give a path construction for a path  $P_G(u, v)$  from  $u$  to  $v$ . We rotate the sector of  $u$  in which  $v$  lies such that  $v$  is placed on the bisector. In worst-case there exist two further nodes  $u_1$  and  $u_2$  on the opposite boundaries of this sector and both are nearer to  $u$  than  $v$  (see Figure 4).

Now, it is not possible to get an edge directly to  $v$ . To see that

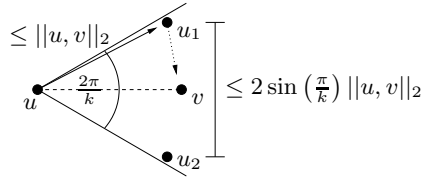


Figure 4: Adjusting a sector

this placement yields to a longest path from  $u$  to  $v$  see [26]. If we assume that the same construction works for the path from  $u_1$  to  $v$  and for  $u_2$  to  $v$  and so on recursively, we get the following bound for the maximum path length between  $u$  and  $v$ :  $\|P_G(u, v)\| \leq \sum_{i=0}^{\infty} (\sin(\pi/k))^i \|u, v\|_2 \leq \frac{1}{1 - \sin(\pi/k)} \|u, v\|_2$   $\square$

As you can see, in best case, we can improve the length stretch factor of the Yao-graph by a factor of 2. Applying this result to Lemma 4 of [7] yields to a better power stretch factor of the Yao-

graph of  $\left(\frac{1}{1 - \sin \pi/k}\right)^\delta$ . It is possible to improve this factor to  $\frac{1}{1 - (\sin \pi/k)^\delta}$  using the analysis of Li et al. [18]. Hence, we can get an energy improvement by a factor of at most  $2^\delta$ .

Another interesting and helpful observation for the search for an orientation which improves stretch factors is that there is always a symmetric communication link to the nearest neighbor of a node (the nearest neighbor over all nearest neighbors in its sectors). This link exists in all possible orientations.

LEMMA 5. Let  $G = (V, E)$  be the Yao-graph and  $u, v \in V$  with  $\forall w \in V : \|u, w\|_2 \leq \|u, v\|_2 \Rightarrow w = v$ . Then  $(u, v) \in E$  and  $(v, u) \in E$  for  $k \geq 6$ .

PROOF. Let  $u, v \in V$  with  $\forall w \in V : \|u, w\|_2 \leq \|u, v\|_2 \Rightarrow w = v$ . It follows directly that  $(u, v) \in E$ . Now, we consider the sector of  $v$  in that  $u$  lies. For  $k \geq 6$  the disk around  $u$  with radius  $\|u, v\|_2$  contains the complete area of this sector of  $v$  up to the distance of  $\|u, v\|_2$ . If there would be a nearer neighbor of  $v$  then this node would also be a nearer neighbor of  $u$ . Hence,  $(v, u) \in E$  and the claim follows.  $\square$

Now, we are ready to present algorithms that adjust the sectors of the nodes. Our main focus in the following is on the improvement of the maximum stretch factors. Lemma 5 shows that it makes no sense to adjust the sectors to nearest node neighbors because they are connected by a symmetric link in all possible sector orientations. Mainly, we differ between four sector orientations:

**Fixed Sector Orientation.** This is the assumption that comes with the definition of the Yao-graph from computational geometry. The orientation of the sectors is fixed and uniformly arranged on all nodes. Hence, in our definition we have  $r(u) = 0$  for all  $u \in V$ .

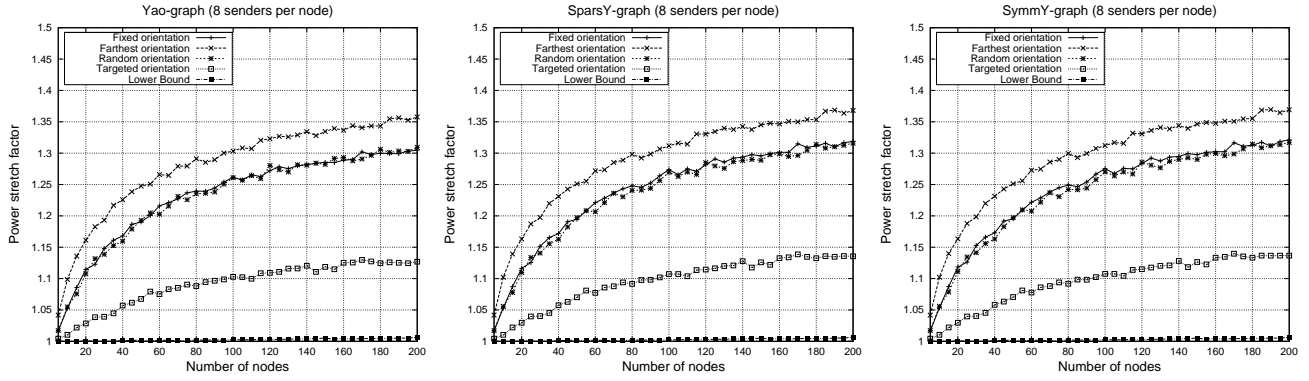


Figure 5: Maximum power stretch factors of Yao-graph, SparsY-graph and SymmY-graph

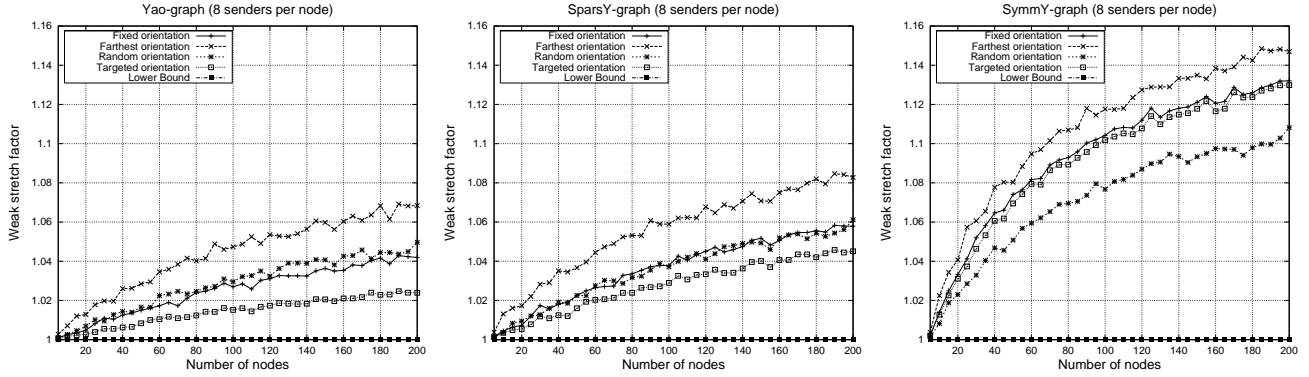


Figure 6: Maximum weak stretch factors of Yao-graph, SparsY-graph and SymmY-graph

**Farthest Sector Orientation.** Here, we adjust the orientation of the sectors on a node to the farthest possible neighbor. In Section 5 we present an algorithm to compute all possible neighbors of all nodes in time  $O(n^2 \log n)$ . We apply this algorithm to determine the farthest possible neighbors of all nodes in the same time (farthest possible neighbor of one node in time  $O(n \log n)$ ).

**Random Sector Orientation.** Every node  $u \in V$  chooses  $r(u)$  uniformly at random from  $[0, 2\pi/k)$ .

**Targeted Sector Orientation.** We begin with a fixed sector orientation and calculate a specific stretch factor. Then, we get the maximum stretch factor and a source node  $s$  and a target node  $t$  which are responsible for this bad stretch factor. Now, we use the result of Lemma 4 and adjust the sector of  $s$  in which  $t$  lies to  $t$  such that  $t$  is exactly on the bisector. Again, we calculate the corresponding stretch factor and repeat the process until no improvement of the maximum stretch factor is achieved.

## 5. LOWER BOUND FOR AN OPTIMAL ADJUSTMENT

In this section we present algorithms to compute lower bounds for the stretch factors of the Yao-graph in the extended model where each node can adjust the orientation of its senders. Let  $G = (V, E)$  be the Yao-graph of  $V$  consisting of  $k$  sectors and  $n := |V|$ . Now, we construct the extended graph  $G^*$  which is defined by  $V$  and all anyhow possible Yao-edges denoted by  $E^*$ . Hence,  $(u, v) \in E^*$  if there exists an adjustment of the sectors of  $u$  such that  $(u, v)$  is an edge in the Yao-graph of  $V$  under this sector orientation. It is obvious that the stretch factors of this extended graph yield to lower bounds for the stretch factors in a Yao-graph in general and hence for an optimal adjustment with regard to a specific stretch factor.

Our algorithm works as follows. Each node  $u \in V$  determines all its possible neighbors. Therefore,  $u$  sorts all other  $n - 1$  nodes according to their Euclidean distance. This needs time  $O(n \log n)$  per node. The nearest node  $v_1$  and the second nearest node  $v_2$  are always possible neighbors of  $u$ . It is possible to adjust the sectors such that  $v_1$  and  $v_2$  are nearest neighbors in two different sectors. We insert them into a sequence sorted by the angle, e.g., for  $v_1$  the angle is given by the three points  $(u(x) + 1, u(y))$ ,  $u$ ,  $v_1$ . For a following node  $v_i, i > 2$ , we determine again the angle  $(u(x) + 1, u(y))$ ,  $u$ ,  $v_i$  and look up the sequence for the node which builds the next smaller angle and for the node which builds the next greater angle. If the difference between these two angles is greater than  $2\pi/k$  it is possible to build up an edge from  $u$  to  $v_i$  and we insert  $v_i$  with its angle into the sequence. In the other case there exists no adjustment such that the link between  $u$  and  $v_i$  is also an edge of the Yao-graph. As you can see each node is inserted at most once into the sequence and so we need time  $O(n \log n)$ . Altogether over all nodes we need time  $O(n^2 \log n)$  to calculate  $G^*$  centrally.

**COROLLARY 1.** Let  $G = (V, E)$  be the Yao-graph of  $G$  under any sector orientation with  $k$  sectors,  $c$  be its length (weak, power) stretch factor and  $c^*$  be the length (weak, power) stretch factor of  $G^*$  as defined before. Then  $c^* \leq c$ .

## 6. EXPERIMENTAL RESULTS

We implemented an interactive testbed in Java where we could evaluate all variants of the three sectorized topologies under different sector orientations [13]. W.l.o.g. in the following we concentrate on the case where the energy needed to transmit some data over a distance  $d$  is given by  $d^\delta$  with  $\delta = 2$ .

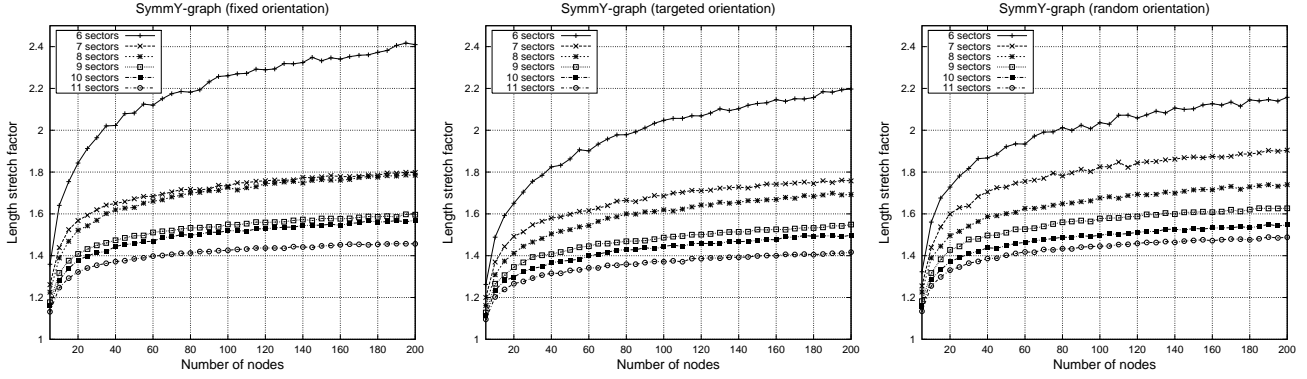


Figure 7: Maximum length stretch factors of SymmY-graph varying the number of sectors

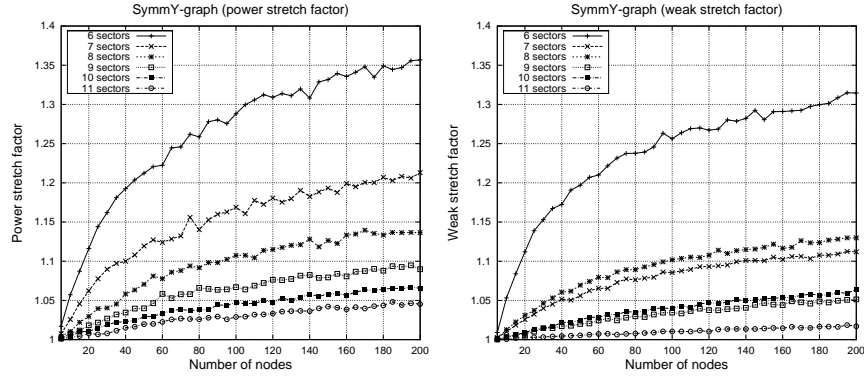


Figure 8: Maximum stretch factors of SymmY under targeted orientation varying the number of sectors

Mainly, we compare stretch factors, number of edges, node degrees, edge lengths and network energy consumption of the Yao-graph and its variants under fixed, farthest, random and targeted sector orientation. We assume that there is given a static set of nodes and the number of nodes varies from 5 to 200. The number of senders per node goes from 6 to 11. All nodes are placed uniformly at random over the given area of size  $652 \times 476$ . For each measurement we performed 500 passes and computed the average and/or the maximum over all these 500 values to get significant simulation results. Our main focus is on improving the maximum stretch factors. We want to reduce energy consumption of the topology on which other tasks like selecting routing paths and forwarding packets take place. Further, we want to give a complete overview about the properties of the Yao-graph and its variants under random vertex sets. In the following, we present our simulation results. Sometimes we placed three small graphics side by side in one figure because the reader should see the difference between the curves on one graphic as well as the difference between three different measurements. The y-axes are arranged in all illustrations.

**Stretch Factors.** Energy conservation is a critical issue for network lifetime, so the stretch factors, especially the length and the power stretch factor, of a wireless network should be minimized.

In Figure 2 the maximum length stretch factors of the Yao-graph and its variants are presented. We compare them under different sector orientations with regard to the lower bound shown in Section 5. It is known that the length stretch factor of the Yao-graph and the weak stretch factor of the SparsY-graph can be upper-bounded by  $1/(1 - 2 \sin(\pi/k))$ . For 8 sectors this value is given by 4.262. The diagram shows that the maximum length stretch factor on random vertex sets is smaller and far away from this upper

bound. The average length stretch factors, given in Figure 3, point out this behavior.

As expected the length stretch factor of the Yao-graph is the best, followed by the factor of the SparsY-graph and finally the factor of the SymmY-graph. Further, it is shown that a farthest sector orientation yields to bad length stretch factors. A targeted sector orientation better the length stretch factor of a fixed sector orientation in all three graphs. Hence, on random vertex sets it is very useful to adjust some sectors of the nodes to get better results concerning energy consumption.

Another very interesting observation, comparing the three diagrams in Figure 2, is that the quality of a random orientation increases with decreasing the number of edges. In the Yao-graph a random orientation gives nearly the same results as a fixed orientation. In the SparsY-graph there is already a relatively small difference between a fixed and a random orientation. But in the SymmY-graph it makes more sense to orientate the sectors at random instead of using a fixed orientation. Randomness helps improving the length stretch factor.

We got similar results for the maximum power stretch factors and for the maximum weak stretch factors (see Figure 5 and Figure 6). The diagrams concerning the power stretch factors show that there is only a small difference between the maximum power stretch factor of the Yao-graph and its variants. On random vertex sets they have nearly the same maximum power stretch factors. The difference between a random sector orientation and a fixed one is not so high as before.

Comparing the maximum weak stretch factors it stands out that in the Yao-graph a random sector orientation is worse than a fixed orientation, but in the SymmY-graph it is suddenly better than a

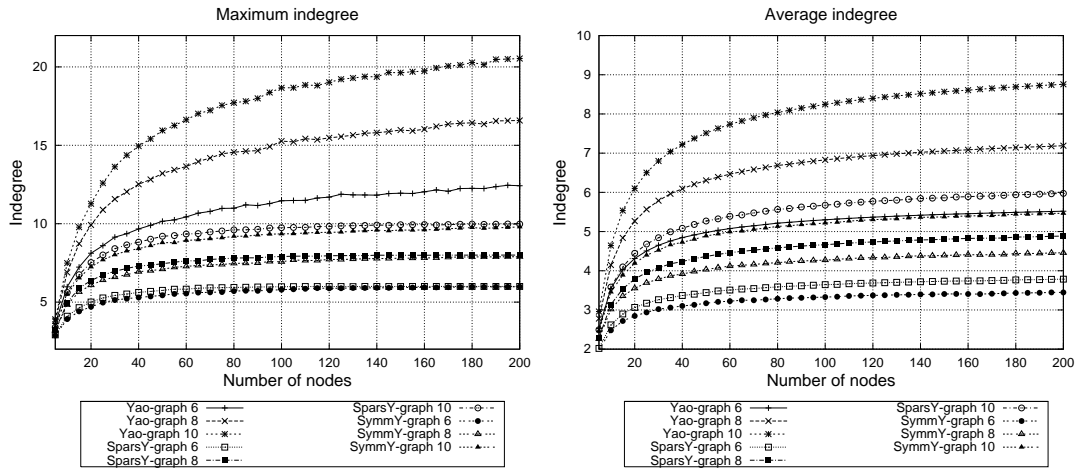


Figure 9: Maximum and average in-degree of the nodes under fixed sector orientation varying the number of sectors

fixed as well as a targeted orientation. All results show that for a given graph the weak stretch factors have the smallest values, followed by the power stretch factors and the highest values are given by the length stretch factors. But this behavior is not really amazing. We can argue that the Yao-graph and its variants on random vertex sets are good spanners, good weak spanners and good power spanners. Further, our new communication model allows to better their stretch factors.

Finally, we varied the number of sectors from 6 to 11 to investigate how this affects the behavior of the different sector orientations. The results of the SymmY-graph are given in Figure 7 and Figure 8. Again, best stretch factors are achieved using targeted orientation. Theoretically, it is known that the values of the stretch factors of the Yao-graph converge to 1 if the number of sectors goes to infinity. The figures point out this behavior for all stretch factors. The curve progression of the maximum length stretch factor of the SymmY-graph using 6 senders per node differs from the characteristic of the other curves. It is interesting that the curves of a fixed sector orientation form a double pack: The curves representing the results under 7 and 8 senders as well as under 9 and 10 senders are nearly identical but there is a small gap between these two progressions. A targeted orientation balances the curves and a random orientation gives homogeneous curves without nonlinear gaps.

**Number of Edges** Figure 12 shows the number of edges of the Yao-graph and its variants under fixed and targeted sector orientation. On the one hand the maximum number of edges of the Yao-graph is upper-bounded by  $kn$ . Hence, using 8 senders per node we can have at most  $8n$  edges. On the other hand all three graphs are connected such that we have at least  $n - 1$  edges. As you can see in the diagrams there is nearly no difference between the number of edges under different sector orientations. We could observe this behavior during all other simulations and we were not wondered about that. Both diagrams show expected values: The Yao-graph has the highest number of edges, followed by the SparsY-graph and three the SymmY-graph at position three.

**Node Degrees.** Here, we present the results concerning the maximum in-degree of a node. The in-degree is very important since there is only one transmitting frequency given. The in-degree gives the maximum number of interferences. In this work the focus was not on improving the in-degree but for completeness we want to present the maximum in-degree of the graphs and the average in-degree of a node, see Figure 9. Simulations showed that the type of orientation does not influence the in-degrees appreciably. The

maximum in-degree of the Yao-graph can only be upper-bounded by the trivial bound of  $n - 1$ . The maximum in-degree of the other both variants converges relatively fast to the number of senders per node. On random vertex sets we can assume that the average in-degree of the Yao-graph is also given by a constant. The Yao-graph consists of at most  $kn$  edges and hence, the average in-degree of a node should converge to  $kn/n = k$ . This observation can be seen in the diagrams.

**Edge Lengths.** In order to spare energy in a wireless network we consider the edge lengths that appear in the sectorized graphs under different sector orientations. Figure 10 gives an overview about the maximum edge lengths.

Note, that it is clear that a farthest orientation brings longest edges. Further, the longest edge of the Yao-graph should be longer than the longest edge of the SparsY-graph which should be longer than the longest edge of the SymmY-graph. The figures point out this behavior. An interesting observation is that a random orientation yields to best values concerning longest edges in the Yao-graph. You have to notice that we performed 500 passes per measurement and this diagram points out that the new communication model brings an improvement to energy consumption. In the sparser variants a fixed and a targeted orientation gives best results. Our simulations showed, looking at the average edge lengths, that there are only very small differences between the curves.

**Network Energy Consumption.** Finally, we compare the topologies with regard to the energy-cost of the whole network structure. We sum up over the energy-cost of all communication links.

As expected the SymmY-graph has lowest energy consumption maintaining all edges, followed by the SparsY-graph and then by the Yao-graph (see Figure 11). We observe the same behavior as for the edge lengths: a random orientation is wise in case of the Yao-graph. Otherwise a fixed or a targeted orientation is adequate.

## 7. CONCLUSIONS

In this work we performed an extensive experimental analysis of adjustable sectorized topologies for static ad hoc networks. We considered three different known topologies and investigated extensively their behavior and their suitability on randomly generated vertex sets. Our experiments show that the Yao-graph, the SparsY-graph and the SymmY-graph are very good choices for topologies on random networks with regard to congestion, energy and interferences. Further, we investigated an extended communication model where we allow each node to adjust the orientation of its senders.

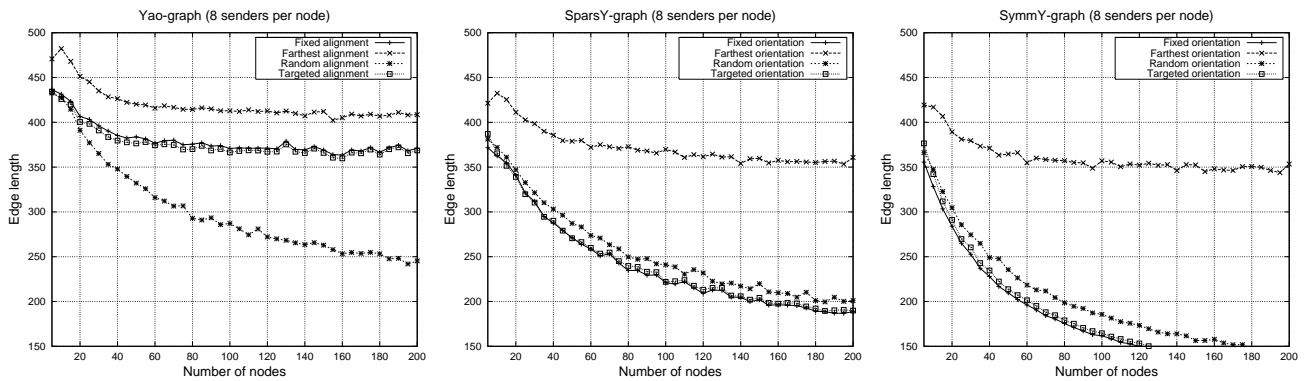


Figure 10: Maximum edge lengths

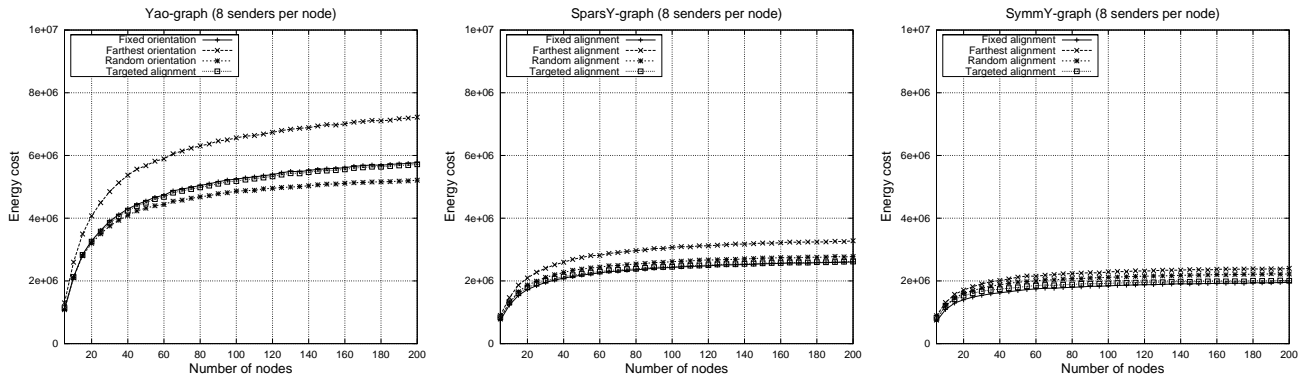


Figure 11: Network energy consumption: summation over all communication links

This is an interesting addition to variable and directional transmitting. We showed by simulations that the extended model allows to improve the stretch factors of the Yao-graph and its variants.

For our experimental evaluation we presented new algorithms for calculating stretch factors efficiently. We can exactly determine the length, the power and the weak stretch factor of a graph consisting of  $n$  nodes and  $m$  edges in time  $O(n^2 \log n + nm)$ . Since the Yao-graph and its variants are sparse graphs of at most  $O(n)$  edges we need only time  $O(n^2 \log n)$  to calculate the stretch factors for these graphs. We used these results to give lower bounds for the stretch factors and applied this in our simulations.

For the new communication model we introduced new algorithms motivated theoretically and showed experimentally that a targeted orientation betters the results of the Yao-graph, the SparsY-graph and the SymmY-graph with regard to energy consumption.

Finally, we want to point out some further research directions. It is still an open problem whether the SparsY-graph is a good spanner in general. We extend this problem and ask whether it is possible to adjust the sectors of a node set or only of some nodes such that the SparsY-graph is a good spanner depending on an specific orientation. Is it even possible to adjust some sectors of nodes of the SymmY-graph such that it gives a good spanner, weak spanner or power spanner? Our simulations showed that this holds on random vertex sets.

Further research can also be done extending our communication model, e.g., consider the case in which the size of the angle of each transmitter can also be controlled or the case where the number of sectors per device is variable. The first model brings new problems since the area around a node is not necessarily divided into non-overlapping regions. In practical settings smart antennas could

be used to form one or more beams in arbitrary directions, so the sectors are completely independent from each other.

Another open question is what are problems and costs of antenna rotation in reality. In this work, e.g., we did not account for the energy consumed by orienting the sectors mechanically.

## 8. REFERENCES

- [1] M. Adler and C. Scheideler. Efficient Communication Strategies for Ad-Hoc Wireless Networks (extended Abstract). In *ACM SPAA'98*, pages 259–268, 1998.
- [2] L. Bao and J.J. Garcia-Luna-Aceves. Transmission scheduling in ad hoc networks with directional antennas. In *MobiCom'02*, pages 48–58. ACM Press, 2002.
- [3] R. R. Choudhury, X. Yang, N. H. Vaidya, and R. Ramanathan. Using directional antennas for medium access control in ad hoc networks. In *MobiCom'02*, pages 59–70. ACM Press, 2002.
- [4] T. H. Cormen, C. Stein, R. L. Rivest, and C. E. Leiserson. *Introduction to Algorithms*. 2001.
- [5] M. Fischer, T. Lukovszki, and M. Ziegler. Geometric searching in walkthrough animations with weak spanners in real time. In *ESA '98*, pages 163–174, 1998.
- [6] M. Fischer, F. Meyer auf der Heide, and W.-B. Strohmann. Dynamic data structures for realtime management of large geometric scenes. In *ESA '97*, pages 157–170, 1997.
- [7] M. Grünewald, T. Lukovszki, C. Schindelhauer, and K. Volbert. Distributed Maintenance of Resource Efficient Wireless Network Topologies (Ext. Abstract). In *EURO-PAR'02*, pages 935–946, 2002.

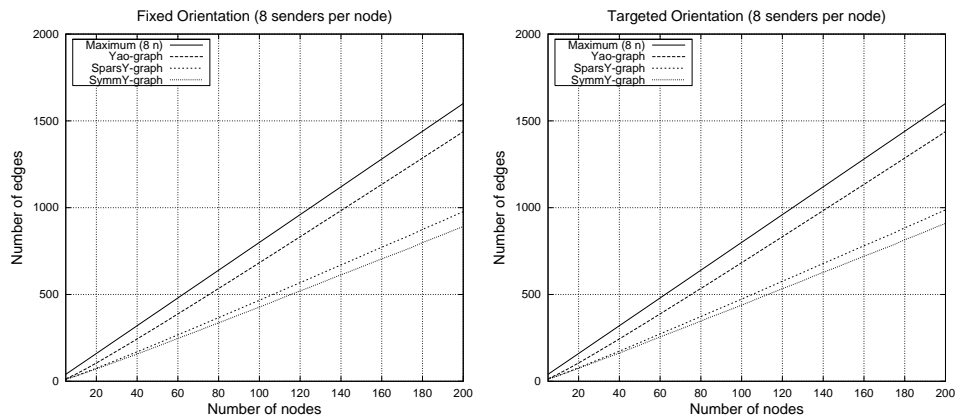


Figure 12: Number of edges in the Yao-graph and its variants

- [8] M. Grünewald, U. Rückert, C. Schindelbauer, and K. Volbert. Directed power-variable infrared communications for the mini robot Khepera. In *AMiRE'03*, pages 113–122, 2003.
- [9] S. Guo and O. W. Yang. Antenna orientation optimization for minimum-energy multicast tree construction in wireless ad hoc networks with directional antennas. In *MobiHoc'04*, pages 234–243, 2004.
- [10] Y. Hassin and D. Peleg. Sparse communication networks and efficient routing in the plane (ext. abstract). In *PODC '00*, pages 41–50, 2000.
- [11] M. Horneffer and D. Plassmann. Directed antennas in mobile broadband systems. In *INFOCOM'96*, pages 704–712, 1996.
- [12] L. Jia, R. Rajaraman, and C. Scheideler. On local algorithms for topology control and routing in ad hoc networks. In *ACM SPAA '03*, pages 220–229, 2003.
- [13] K. Volbert. *Investigating sectorized topologies*, 2004. <http://wwwcs.upb.de/cs/ag-madh/WWW/english/kvolbert/yao.html>.
- [14] Y.-B. Ko, V. Shankarkumar, and N. H. Vaidya. Medium Access Control Protocols Using Directional Antennas in Ad Hoc Networks. In *INFOCOM'00*, pages 13–21, March 2000.
- [15] L. Li, J.-Y. Halpern, P. Bahl, Y.-M. Wang, and R. Wattenhofer. Analysis of a cone-based distributed topology control algorithm for wireless multi-hop networks. In *PODC'01*, pages 264–273. ACM Press, 2001.
- [16] X.-Y. Li. Applications of Computational Geometry in Wireless Ad Hoc Networks, Book Chapter of Ad Hoc Wireless Networking, Kluwer, edited by X. Z. Cheng, X. Huang, and D.-Z. Du. 2003.
- [17] X.-Y. Li. Topology Control in Wireless Ad Hoc Networks, Book Chapter of Ad Hoc Networking, IEEE Press, edited by S. Basagni, M. Conti, S. Giordano, and I. Stojmenovic. 2003.
- [18] X.-Y. Li, P.-J. Wan, and Y. Wang. Power Efficient and Sparse Spanner for Wireless Ad Hoc Networks. In *ICCCN '01*, pages 564–567, 2001.
- [19] F. Meyer auf der Heide, C. Schindelbauer, K. Volbert, and M. Grünewald. Energy, Congestion and Dilation in Radio Networks. In *ACM SPAA'02*, pages 230–237, 2002.
- [20] F. Meyer auf der Heide, C. Schindelbauer, K. Volbert, and M. Grünewald. Congestion, Dilation, and Energy in Radio Networks. In *TOCS'04*, 2004.
- [21] G. Narasimhan and M. Smid. Approximating the stretch factor of euclidean graphs. *J. Comput.*, 30(3):978–989, 2000.
- [22] G. T. Okamoto. *Smart Antenna Systems and Wireless LANs*. Kluwer Academic Publishers, 1998.
- [23] R. Rajaraman. Topology Control and Routing in Ad hoc Networks: A Survey. In *SIGACT News, June 2002*, pages 60–73, 2002.
- [24] R. Ramanathan. On the performance of ad hoc networks with beamforming antennas. In *MobiCom'01*, pages 95–105. ACM Press, 2001.
- [25] S. Rührup, C. Schindelbauer, K. Volbert, and M. Grünewald. Performance of Distributed Algorithms for Topology Control in Wireless Networks. In *IPDPS'03*, page 28, 2003.
- [26] J. Ruppert and R. Seidel. Approximating the  $d$ -dimensional complete Euclidean graph. In *CCCG '91*, pages 207–210, 1991.
- [27] C. Schindelbauer, K. Volbert, and M. Ziegler. Spanners, Weak Spanners, and Power Spanners for Wireless Networks, accepted for ISAAC'04. 2004.
- [28] W.-Z. Song, Y. Wang, and X.-Y. Li. Localized algorithms for energy efficient topology in wireless ad hoc networks. In *MobiHoc'04*, pages 98–108. ACM Press, 2004.
- [29] M. Takai, J. Martin, R. Bagrodia, and A. Ren. Directional virtual carrier sensing for directional antennas in mobile ad hoc networks. In *MobiHoc'02*, pages 183–193, 2002.
- [30] K. Volbert. A Simulation Environment for Ad Hoc Networks Using Sector Subdivision. In *10th Euromicro Workshop on Parallel, Distributed and Network-based Processing (PDP'02)*, pages 419–426, 2002.
- [31] Y. Wang and X.-Y. Li. Localized construction of bounded degree and planar spanner for wireless ad hoc networks. In *Proceedings of the 2003 joint workshop on Foundations of mobile computing*, pages 59–68. ACM Press, 2003.
- [32] Y. Wang, X.-Y. Li, and O. Frieder. Distributed Spanner with Bounded Degree for Wireless Ad Hoc Networks. In *Int. Journal of Foundations of Computer Science (14)*, pages 183–200, 2003.
- [33] Y. Wang, X.-Y. Li, P.-J. Wan, and O. Frieder. Sparse power efficient topology for wireless networks. In *HICSS'02*, page 296, 2002.
- [34] A. C.-C. Yao. On constructing minimum spanning trees in  $k$ -dimensional spaces and related problems. *SIAM J. Comput.*, 11(4):721–736, 1982.