On Sparse Spanners of Weighted Graphs*

Ingo Althöfer,[1] Gautam Das,[2] David Dobkin,[3] Deborah Joseph,[4] and José Soares[5]

[1] Fakultät für Mathematik, Universität Bielefeld, Postfach 8640,
4800 Bielefeld, Federal Republic of Germany

[2] Mathematical Sciences Department, Memphis State University,
Memphis, TN 38152, USA
dasg@next1.mcsi.memst.edu

[3] Department of Computer Science, Princeton University,
Princeton, NJ 08544, USA
dpd@princeton.edu

[4] Department of Computer Sciences, University of Wisconsin,
Madison, WI 53706, USA
joseph@cs.wisc.edu

[5] Universidade de São Paulo, IME–USP/MAC,
01498-970 Sao Paolo-SP, Brazil, and
Department of Computer Science, University of Chicago,
Chicago, IL 60637, USA
jose@cs.uchicago.edu

Abstract. Given a graph G, a subgraph G' is a t-spanner of G if, for every $u, v \in V$, the distance from u to v in G' is at most t times longer than the distance in G. In this paper we give a simple algorithm for constructing sparse spanners for arbitrary weighted graphs. We then apply this algorithm to obtain specific results for planar graphs and Euclidean graphs. We discuss the optimality of our results and present several nearly matching lower bounds.

1. Introduction

Let $G = (V, E)$ be a connected n-vertex graph with arbitrary positive edge weights. A subgraph $G' = (V, E)$ is a *t-spanner* if, between any pair of vertices, the distance in G' is at most t times longer than the distance in G. The value of t is the *stretch factor* associated with G'. We consider the problem of determining t-spanners for graphs where the spanners are sparse and t is a constant independent of the size of the graph. Sparsity is measured according to two criteria. Let *Weight(G)* denote the sum of all edge weights of graph G, and let *Size(G)* denote the number of edges. A graph is sparse in *size* if it has few edges. Similarly, a graph is sparse in *weight* if its total edge weight is small. Our results separate graphs into classes where spanners with linearly many edges achieve constant stretch factors, and classes where a nonlinear number of edges are necessary.

Problems of this type appear in numerous applications. Spanners appear to be the underlying graph structure in various constructions in distributed systems and communication networks [Aw], [PUl], [PUP]. They also appear in biology in the process of reconstructing phylogenetic trees from matrices, whose entries represent genetic distances among contemporary living species [BD]. Robotics researchers have studied spanners under the constraints of *Euclidean* geometry, where vertices of the graph are points in space, and edges are line segments joining pairs of points [C], [DFS], [DJ], [K], [KG], [LL].

In the above applications previous research has focused on graphs with specific constraints. In distributed computation the design of synchronizers [Aw], [PUl] and the design of succinct routing tables [PUP] implicitly generated spanners for graphs with *unit edge weights*. For example, in designing routing tables [PUP] the routes follow the edges of a sparse spanner. For any stretch factor $O(t)$, the size of these spanners are $O(n^{1 + 1/t})$. These designs are based upon a *clustering algorithm*, which is more complex than the algorithm in this paper. Moreover, it is not easy to generalize the clustering algorithm to graphs with arbitrary edge weights. Recently the problem of designing succinct routing tables has been considered for graphs with arbitrary edge weights [ABLP], [AP]. For any t, the scheme in [ABLP] routes along paths with are at most $O(t^2 9^t)$ longer than the shortest paths, while the total memory required for the tables is $O(tn^{1 + 1/t} \log n)$. In [AP] the routes are $O(t^2)$ longer, while the memory required is $O(n^{1 + 1/t} \log^2 n \log D)$, where D is the *diameter* of the graph. Spanners have been considered for special classes of graphs in [PS], however, these graphs have unit edge weights.

In all the above research, sparseness has been achieved in the size of spanners, but not in the weight. In robotics, graphs with varying edge weights have been examined, but the weights are Euclidean distances and not arbitrary. Because of this restriction, it has been possible to construct linearly sized spanners, unlike the above examples. For instance, the *Delaunay* and other triangulations approximate complete straight-edged graphs on the plane [C], [DFS], [DJ], [LL], [KG]. A few papers have considered weight sparseness of spanners for these graphs [DJ], [LL]. For any t, there exist spanners for the complete graph on the plane

with stretch factor $O(t)$, and weight within an $O(1 + 1/t)$ multiple of the weight of the minimum spanning tree. Weight sparseness has also been considered in [Al], [D], and [S] for the special case of 1-spanners under a general model where spanners may have auxiliary vertices.

In this paper we approach the problem from a very broad perspective. Our graphs have no special embeddings, and we allow arbitrary positive edge weights. For any t, we show that every such graph has a spanner with an $O(t)$ stretch factor, and $O(^{1 + 1/t})$ size. We also provide weight bounds for our spanners. The contributions of this paper are: a very simple polynomial-time algorithm for constructing sparse spanners (sparse *both* in size and weight) of arbitrary weighted graphs, using these ideas for constructing spanners of *planar* graphs, some lower bound results, and some results on spanners in Euclidean spaces of arbitrary dimensions and norms. Since any spanner with appropriate sparseness and stretch factor can be used for constructing synchronizers [PU1], our algorithm provides a simple alternative to the clustering algorithm previously used for constructing synchronizers. Similarly, we hope that our results will simplify the construction of succinct routing tables for arbitrary weighted graphs.

The next section describes how to construct sparse spanners for general as well as planar graphs with arbitrary edge weights. Sections 3 and 4 deal with several lower bound results. Section 5 discusses spanners for Euclidean graphs. We conclude with some open problems.

2. Construction of Sparse Spanners

Let G be an n-vertex, connected, weighted graph. Let us examine the minimum spanning tree, denoted as $MST(G)$, as a possible candidate for a spanner. It is obviously the sparsest spanner under both criteria of sparseness, because it is the subgraph with the least total edge weight that still connects all vertices. However, its stretch factor can be as bad as $\Omega(n)$. To see this, consider an n-vertex graph, with only n unit weight edges, arranged in a cycle. Clearly, the minimum spanning tree does not contain an edge, say $[u, v]$, and the distance from u to v in the tree is $n - 1$. Thus the tree is an $(n - 1)$ spanner.

Instead, we would like to look for spanners whose stretch factors are constants, independent of n. We use the minimum spanning tree as a yardstick for measuring sparseness, and try to generate spanners whose sparseness comes close to it. Our results are encouraging, because we show that any graph has spanners with constant stretch factors, whose sparseness can be made arbitrarily close to that of $MST(G)$. They are summarized by Theorems 1 and 2.

Theorem 1. *Given an n-vertex graph G and a t > 0, there is a polynomially constructible* $(2t + 1)$-*spanner G' such that,*

(1) $Size(G') < n\lceil n^{1/t}\rceil$,
(2) $Weight(G') < Weight(MST(G))(1 + n/2t)$.

Note that the stretch factor is independent of the number of vertices in the graph, and of the edge weights. Thus, even for a dense graph with $\Omega(n^2)$ edges, sparse spanners exist which have good stretch factors. Furthermore, the sparseness can be made arbitrarily close to that of $MST(G)$, at the expense of an increase in the stretch factor. Previous research [Aw], [PS] has produced spanners with the same size and stretch factors, but the graphs considered there were unweighted, and sparseness was measured by size only. Also, our algorithm is simpler than the clustering algorithm employed. These bounds should be compared with the following lower bounds, to be proved in Section 3:

Theorem 4. *For every pair of integers $t \geq 0$ and $n \geq 3$, there exists an n-vertex graph G for which every $(2t + 1)$-spanner G' is such that,*

(1) $Size(G') > \frac{1}{8}n^{1+4/3(2t+3)}$,

(2) $Weight(G') > \frac{1}{8} Weight(MST(G))n^{4/3(2t+3)}$.

Actually, a slightly weaker lower bound holds even for *Steiner* spanners, where auxiliary vertices are allowed to be added to the graph (see Section 4).

The next theorem concerns planar graphs.

Theorem 2. *Given an n-vertex planar graph H and a $t > 0$, there is a polynomially constructible $(2t + 1)$-spanner G' such that,*

(1) $Size(G') < (n - 1)(1 + 1/t)$,

(2) $Weight(G') < Weight(MST(G))(11/t)$.

Here we again observe that the stretch factor is independent of the number of vertices in the planar graph, and of the edge weights. Thus, arbitrarily sparse spanners exist which have good stretch factors. It is well known that the maximum size of a planar graph can be $3n - 6$, so in effect we demonstrate that even the multiplicative constant in the size can be reduced. Theorem 2 is stronger in flavor than Theorem 1, because our lower bound results will show that both size and weight bounds are tight. We later give an interesting application of this in connection with Euclidean graphs.

Before we prove our results, we introduce an algorithm for constructing spanners. It constructs a sparse subset of edges so that a required stretch factor is achieved. The algorithm, called $SPANNER(G, r)$, takes as input a weighted graph G, and a positive parameter r. The weights need not be unique. It produces as output a subgraph G'. Note that this algorithm is essentially a generalization of Kruskal's algorithm for computing minimum spanning trees [T]. The algorithm has also been independently discovered by Bern [Be] in a different context, in connection with the problem of *dilation of metric spaces* [RS]. While it is very simple and easy to implement in polynomial time, the subgraph it generates has many interesting properties.

Algorithm $SPANNER(G = (V, E), r)$
begin
  Sort E by nondecreasing weight;
  $G' := (V, \{\})$;
  for every edge $\varepsilon = [u, v]$ in E **do**
    begin
      Compute $P(u, v)$, the shortest path from u to v in G':
      if $(r \cdot Weight(e) < Weight(P(u, v)))$ **then**
        add e to G';
    end;
  Output G';
end;

The following lemmas describe the properties of the output graph G'.

Lemma 1. G' *is an r-spanner of G.*

Proof. Consider any edge $[u, v]$ in $G - G'$. Since it was rejected by the algorithm, the following is true. At the instant it is examined by the algorithm, there is a path $P(u, v)$ from u to v in the current graph of length $\leq r \cdot Weight([u, v])$. Thus, in the final output each deleted edge $[u, v]$ is associated with a short path $P(u, v)$. Now consider any shortest path between vertices a and b in G. Let the length of this path be l. For every edge $[u, v]$ along this path not in G', we replace the edge by the corresponding short path $P(u, v)$. The resulting path between a and b is clearly in G', and has length $\leq l \cdot r$, which proves the lemma. $\qquad\square$

Lemma 2. *Let C be any cycle in G'. Then* $Size(C) > r + 1$.

Proof. Assume that a cycle C remained with size $\leq r + 1$. Let $[u, v]$ be the last edge in C to be examined by the algorithm. Clearly, it is one of the (possibly many) edges in the cycle with the largest weight. When $[u, v]$ is being considered by the algorithm, the remaining portion of the cycle from u to v has already been included in G'. This portion has at most r edges, and each edge has weight not greater than $Weight([u, v])$. Thus, at that instant there is a path from u to v in G', whose length is $\leq r \cdot Weight([u, v])$. In that case $[u, v]$ should not have been added, which is a contradiction. $\qquad\square$

Lemma 3. *Let C be any cycle in G', and let e be any edge in C. Then* $Weight(C - \{e\}) > r \cdot Weight(e)$.

Proof. Assume that a cycle C remained which violates the above weight condition for some edge e in C. Clearly, it violates this condition for the last edge in C to be examined by the algorithm, say $[u, v]$, because the weight of $[u, v]$ is at least as large as the weight of e. As in the above lemma, $[u, v]$ is one of the (possibly many) edges in the cycle with the largest weight. Thus, the remaining portion of

the cycle has length $> r \cdot Weight([u, v])$. In this case $[u, v]$ should not have been added, which is a contradiction. $\square$

Lemma 4. *MST(G) is contained in G'.*

Proof. Before we prove this lemma, let us repeat that our algorithm is essentially a generalized minimum spanning tree algorithm. In fact, for an infinite r its behavior is exactly like Kruskal's minimum spanning tree algorithm [T].

Our proof is by induction on the order in which edges are examined. Let the sequence $\{\} = G'_0, G'_1, \ldots, G'_{Size(E)} = G'$ represent the growth of G', where G'_i represents the partially constructed subgraph after the ith edge has been examined. At any stage the subgraph will be a collection of connected components, which will finally become one connected component.

Now let us consider Kruskal's algorithm, which also examines edges by nondecreasing weight. Let the sequence $\{\} = M_0, M_1, \ldots, M_{Size(E)} = MST(G)$ represent the growth of the minimum spanning tree, where M_i represents the partially constructed tree after the ith edge has been examined. This edge is added only if it does not form a cycle in the partially constructed tree. Thus at any stage, the partially constructed tree will be a forest, which will finally become one connected tree.

We now prove by induction the following, which is a stronger hypothesis than the lemma. For all i, the number of connected components of M_i is the same as that of G'_i, and each component of M_i is contained in a corresponding component of G'_i.

To prove this, let us imagine that both algorithms are being run simultaneously. The basis of the induction is easily proved, because at the start both the tree and the spanner have n components each, consisting of isolated vertices. For the induction step, assume that the hypothesis is true for some i. Let the $(i + 1)$st edge to be considered be $e = [u, v]$. Two cases arise.

Case 1: u and v belong to the same component of M_i. Then e forms a cycle with M_i, and hence it is not included in M_{i+1}. Since u and v belong to the same component of M_i, by the hypothesis they also belong to the same corresponding component of G'_i. If e is not included in G'_{i+1}, nothing has happened, and the hypothesis remains true. If e is included in G'_{i+1}, it gets added to the same component of G'_i, and the hypothesis remains true.

Case 2: u and v belong to different components of M_i. Then e does not form a cycle with M_i, and hence it is included in H'_{i+1}. In this case two components of M_i merge to form one component of M_{i+1}. Since u and v belong to different components of M_i, by the hypothesis they also belong to different corresponding components of G'_i. In other words, the distance from u to v in G'_i is infinite. Thus e will clearly pass the test in the *SPANNER* algorithm and be included in G'_{i+1}. In this case the two components of G'_i merge to form one component of G'_{i+1}. Thus the hypothesis remains true, and the inductive proof in complete.

Finally, since $M_{Size(E)} = MST(G)$, and $G'_{Size(E)} = G'$, the lemma is proved. $\square$

In proving the two theorems, Lemma 2 is used in proving size sparseness, while Lemmas 3 and 4 are used in proving weight sparseness of the resulting subgraphs.

We still need two more lemmas. Let the *size* of a face of a planar graph be the number of edges encountered while traversing around the boundary of the face. We are allowed to repeat an edge along the boundary, in case the face contains several biconnected components, connected by edges. Lemma 5 bounds the size of a planar graph, given a minimum face size.

Lemma 5. *If all the faces of an n-vertex connected planar graph G have sizes* $\geq r$, *then* $Size(G) \leq (n - 2)(1 + 2/(r - 2))$.

Proof. Intuitively, the lemma shows that the minimum face size acts as a parameter for size sparseness of planar graphs. To prove this, let m be the size of the graph. Euler's formular for planar graphs states that $n - m + f = 2$, where f is the number of faces in the graph. Thus, $f = m + 2 - n$. If we traverse the boundary of each face and mark the edges encountered, every edge in the graph will eventually be marked twice. This is because edges that have different faces adjacent on both sides are counted twice, the traversal of each adjacent face contributing once. Edges with the same face on both sides are also counted twice during the traversal of the same face. Since the size of each face is $\geq r$, we have $f \cdot r \leq 2m$. Substituting for f from above, we get

$$(m + 2 - n)r \leq 2m,$$

$$m \leq (n - 1)\left(1 + \frac{2}{r - 2}\right).$$

Thus the lemma is proved. We observe that the lemma has meaning only for $r \leq 2n - 2$, because this is the maximum possible face size of a connected planar graph. This happens when the graph is a tree. $\qquad\square$

Our next lemma is from extremal graph theory, and is easily derivable from Theorem 3.7, Chapter III, in [Bo]. Define the *girth* of a graph as the size (number of edges) of its smallest simple cycle. This lemma provides an upper bound on the size of a (not necessarily planar) graph with a given girth.

Lemma 6. *Let G be an n-vertex graph with girth* $> r$. *Then* $Size(G) < n\lceil n^{2/(r-2)}\rceil$.

Thus the girth acts as a parameter for size sparseness of general graphs. At this stage we are ready to prove our two theorems. We first prove Theorem 2, then use some of the ideas involved to prove Theorem 1.

Proof of Theorem 2. We run the *SPANNER* algorithm on a given n-vertex planar graph G and a $t > 0$, after setting $r = 2t + 1$. By Lemma 1, the resulting graph has a stretch factor at most $2t + 1$. Also, by Lemma 2, the girth of the output is

$> 2t + 2$. Now if the spanner is a tree, its size is $n - 1$, and the size bound is trivially true. If the spanner is not a tree, then it has cycles. For a connected planar graph with cycles, every face has to contain a cycle. Hence the minimum face size is at least as large as the girth, that is, $> 2t + 2$. Thus, by Lemma 5, the size of the output graph is

$$Size(G') < (n - 2)\left(1 + \frac{2}{(2t + 2) - 2}\right),$$

or

$$Size(G') < (n - 1)\left(1 + \frac{1}{t}\right).$$

We now prove the weight bound, which requires a different approach (the proof is similar to the method in [LL]). By Lemma 4, the output subgraph has to contain $MST(G)$. If the subgraph is $MST(G)$, the weight bound is trivially true. If it has more edges than $MST(G)$, we prove the weight bound as follows. Consider a planar drawing of the subgraph, with $MST(G)$ embedded in the subgraph. If we walk around the tree (visiting each edge twice), our path will resemble a skinny polygon with perimeter $2 \cdot Weight(MST(G))$. Our accounting strategy will be to grow this polygon outward by absorbing neighboring edges of the subgraph, until it becomes the outer face of the graph. At any stage, an edge is selected which, along with a portion of the polygon's current boundary, circumscribes a face which is adjacent and exterior to the polygon.

Consider stage i. Let the length of the polygon be W_i. In this case $W_0 = 2 \cdot Weight(MST(G))$. Let T_i be the total length of all edges encountered so far that do not belong to the minimum spanning tree. Thus $T_0 = 0$. Let the edge selected to be absorbed be $e_i = [u, v]$, and let the remaining boundary of the adjacent face be $P(u, v)$, which is in fact the current polygon's boundary from u to v. Now this face contains a cycle, such that $[u, v]$ is an edge of the cycle, with the remaining portion of the cycle residing within $P(u, v)$. We can use Lemma 3, and conclude that $Weight(P(u, v)) > (2t + 1) \cdot Weight(e_i)$. Thus, we get two equations:

(1) $W_{i+1} = W_i - Weight(P(u, v)) + Weight(e_i) < W_i - Weight(e_i)(2t)$.
(2) $T_{i+1} = T_i + Weight(e_i)$.

We now prove that, in the limit when i goes to infinity T_i converges to at most $Weight(MST(G))(1/t)$. To see this, imagine that initially W is a bucket with $2 \cdot Weight(MST(G))$ units of water inside, and that T is an empty bucket. At every stage, if we decide to add an arbitrary k ($= Weight(e_i)$) units of water to T, we are required to remove at least $k \cdot 2t$ units of water from W. The process stops when W is empty. It is easy to see that, irrespective of whatever quantity k is chosen at any stage, T will finally contain at most $2 \cdot Weight(MST(G))(1/2t)$ units of water.

Since the limit is never quite reached in actual graphs, the weight of the

nonminimum spanning tree edges is less than $Weight(MST(G))(1/t)$. Thus $Weight(G') < Weight(MST(G))(1 + 1/t)$, and the theorem is proved. ☐

Proof of Theorem 1. We are given an n-vertex graph G and a $t > 0$ as input. We set $r = 2t + 1$ and run the *SPANNER* algorithm. By Lemma 1, the resulting graph has stretch factor $\leq 2t + 1$. Also, by Lemma 2, the girth of the output is $> 2t + 2$. Thus, by Lemma 6, the size of the output is $< n\lceil n^{1/t}\rceil$, which proves the size bound.

We now prove the weight bound. By Lemma 4, $MST(G)$ is contained in the subgraph. For each vertex v, consider the corresponding graph G_v, composed of $MST(G)$ and the edges of G' incident to v but not in $MST(G)$. Let the latter set of edges be denoted as E_v. We claim that these graphs are planar. To see this, we lay out G_v, for some v, on a plane as follows. First draw $MST(G)$ on the plane, without crossing edges. Then walk around the tree, forming a skinny polygon. Order all vertices (other than v), as they *first* appear along the boundary of the polygon. Now lay out the edges from v to these vertices, in the above order, with the edges drawn *outside* the polygon. Clearly, we can complete the layout without crossing edges.

For the planar graph G_v, using methods similar to the proof of Theorem 2, we see that $Weight(E_v) < Weight(MST(G))(1/t)$. Since there are n vertices, it is easy to see that $\sum_{v\in v} Weight(E_v) < Weight(MST(G))(n/t)$. However, in this summation, each nonminimum spanning tree edge of G' has been added twice. This is because any such edge $[v, v]$ belongs to both E_u and E_v. So the weight of the nonminimum spanning tree edges is at most $Weight(MST(G))(n/2t)$. Thus $Weight(G') < Weight(MST(G))(1 + n/2t)$.

This proves the weight bound, and the theorem. ☐

3. Lower Bounds

In this section we show various lower bounds for spanners of graphs.

Our first result concerns general graphs with arbitrary positive weights. Peleg and Shäffer gave the following bound.

Theorem 3 [PS]. *For every pair of integers $t \geq 0$ and $n \geq 3$, there exists an n-vertex graph for which every $(2t + 1)$-spanner has at least $\frac{1}{4}n^{1 + 1/(2t + 3)}$ edges.*

This result follows directly from the two lemmas below.

Lemma 7 [PS]. *The only t-spanner of a graph G with unit weight on the edges and with girth $g \geq t + 2$ is G itself.*

Lemma 8 [ES]. *For every pair of integers $r, n \geq 3$, there exists an n-vertex graph with girth $g \geq r$ and with at least $\frac{1}{4}n^{1 + 1/r}$ edges.*

Recently, the result of Erdös and Sachs [ES] has been improved by Margulis

[M] and Lubotzky *et al.* [LPS]. Although they use very similar constructions, the result stated in [M] is more general and provides the required improvement. We now briefly describe the graphs constructed in [M].

Let p be an odd prime. Let $m > 2\sqrt{p}$ be an even integer relatively prime to p. The graph $G^{p,m}$ obtained from p and m is a $(p + 1)$-regular graph on $n \le m^3/8$ vertices. The girth of $G^{p,m}$ is $g \ge r = 4 \log_p(m/2)$ [M]. Thus, $G^{p,m}$ is an n-vertex graph with more than $\frac{1}{2}n^{1+4/3r}$ edges.

Using these graphs, we improve the bound given in Lemma 8:

Lemma 9. *For every pair of integers $r, n \ge 3$, there exists an n-vertex graph G with girth $g \ge r$ and with more than $\frac{1}{8}n^{1+4/3r}$ edges.*

Proof. If $r = 3$ we can take the complete graph on n vertices as G.

If $r = 4$ we can take the complete bipartite graph with partition of sizes $\lfloor n/2 \rfloor$ and $\lceil n/2 \rceil$ as G.

From now on we consider $r \ge 5$.

Case 1: $n < (3^{r/4} + 2)^3$. Then

$$\frac{1}{8}n^{1+4/3r} < \frac{1}{8}n(3^{r/4} + 2)^{4/r} < \frac{1}{8}n \cdot 5 < n - 1.$$

Thus, we can take a tree on n vertices as G.

Case 2: $n \ge (3^{r/4} + 2)^3$. By Bertrand's postulate there exists a prime p such that $\frac{1}{2}(n^{1/3} - 2)^{4/r} \ge p \le (n^{1/3} - 2)^{4/r}$. Since $\frac{1}{2}(n^{1/3} - 2)^{4/r} \ge \frac{1}{2}((3^{r/4} + 2) - 2)^{4/r} > \frac{5}{2}$, p is an odd prime.

Let m be the least even integer such that $m \ge 2p^{r/4}$ and m is relatively prime to p. Clearly, $m \le 2p^{r/4} + 4$.

Consider the graph $G^{p,m}$ as described before. This graph is a $(p + 1)$-regular graph on $n' \le m^3/8$ vertices and it has girth $g \ge \log_p(m/2)$.

Let G be the graph consisting of the union of $\lfloor n/n' \rfloor$ disjoint copies of $G^{p,m}$ and $n \bmod n'$ new isolated vertices. Since $n' \ge m^3/8$ we have that

$$n' \le \frac{(2p^{r/4} + 4)^3}{8} \le \frac{(2(n^{1/3} - 2) + 4)^3}{8} = n.$$

This implies that $n \bmod n' < n/2$.

Thus, we obtain that

$$Size(G) \ge \frac{(n - (n \bmod n'))(p + 1)}{2} > \frac{n}{4}(p + 1)$$

$$\ge \frac{n}{4} \frac{(n^{1/3} - 2)^{4/r} + 2}{2} > \frac{n}{4} \frac{n^{4/3r}}{2} = \frac{1}{8}n^{1+4/3r},$$

which proves the bound on the size. It remains to show that G has girth as promised.

The girth g of G is equal to the girth of $G^{p,m}$. Thus

$$g \geq 4 \log_p\left(\frac{m}{2}\right) \geq 4 \log_p\left(\frac{2p^{r/4}}{2}\right) \geq r. \qquad \square$$

If we put unit weights on the edges of these graphs, then the weight of the minimum spanning tree becomes $n - 1$. Thus, the following is immediate from Lemmas 7 and 9.

Theorem 4. *For every pair of integers $t \geq 0$ and $n \geq 3$, there exists an n-vertex graph G for which every $(2t + 1)$-spanner G' is such that*

(1) *$Size(G') > \frac{1}{8}n^{1 + 4/3(2t + 3)}$,*
(2) *$Weight(G') > \frac{1}{8} Weight(MST(G))n^{4/3(2t + 3)}$.*

This theorem shows that the size result of Theorem 1 is tight up to a constant factor in the exponent of n, while, for the weight, the gap between lower and upper bounds is larger.

Our next result concerns planar graphs with arbitrary positive weights.

Theorem 5. *For every integer $t \geq 1$, there are infinitely many values of n, for which there exist n-vertex planar graphs G with unit edge weights, such that every $(2t + 1)$-spanner G' satisfies*

(1) *$Size(G') = \Omega(n(1 + 1/t))$,*
(2) *$Weight(G') = \Omega(Weight(MST(G))(1 + 1/t))$.*

Proof. Let n take the values $k(t + 1) + 2$ for every $k \geq 2$. For every admissible pair of t and n, we construct an n-vertex planar graph G as follows. Lay out two special vertices, u and v, on the plane. Connect them via k chains of $t + 1$ vertices each. Let all edges have unit weights.

Clearly, the graph is planar. Furthermore, the girth is $2t + 4$. Now consider any *proper* subgraph of G, say G'. Let $[a, b]$ be an edge in $G - G'$. Because the girth is $\geq 2t + 4$, any alternate path from a to b in G' has length at least $2t + 3$. This means that there can be no proper spanner of G with a stretch factor $2t + 1$. To achieve this stretch factor or less, we have to include the whole graph.

We now prove the size bound of G.

By construction, G consists of $(n - 2)/(t + 1)$ edge-disjoint chains. Since each chain contains $t + 2$ edges, the total number of edges in G is $(t + 2)((n - 2)/(t + 1))$.

Thus the size of the graph is $\Omega(n(1 + 1t))$. The weight bound is trivially true, since unit edge weights imply that the weight of the minimum spanning tree is $n - 1$. This theorem shows that the size and weight results of Theorem 2 are tight. $\qquad \square$

4. Steiner Spanners: A Lower Bound

In what follows we prove lower bound results for various generalizations of spanners. Consider graphs with arbitrary positive weights. Let $Distance(u, v, G)$ be the distance from vertex u to vertex v in graph G. Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two graphs with V_1 a subset of V_2. G_2 is called a *Steiner t*-spanner of G_1 if, for all $u, v \in V_1$, $Distance(u, v, G_1) \leq Distance(u, v, G_2) \leq t \cdot Distance(u, v, G_1)$. Thus, a Steiner spanner is not simply a subgraph of G_1, rather it may contain auxiliary vertices and edges. An important point that is the Steiner spanner is not allowed to "cheat," that is, paths in G_2 are never shorter than those in the original graph G_1, though they may use auxiliary vertices and edges.

For special graphs Steiner spanners can be substantially smaller than simple spanners. For example, consider the complete *n*-vertex graph with unit edge weights. Clearly, every simple 1-spanner requires all edges, but the *star graph* with one auxiliary vertex attached to all n original vertices via n additional edges of weight $\frac{1}{2}$ is a Steiner 1-spanner with only n edges. However, the following lower bound result shows that such constructions are not always possible for all graphs. In fact, there exist graphs such that Steiner spanners cannot be much smaller than simple spanners.

Theorem 6. *For every pair of integers* $t \geq 1$ *and* $n \geq 3$, *there exists an n-vertex graph* G *with unit edge weights for which every Steiner t-spanner requires* $\Omega(1/\log n) \cdot n^{1 + 4/3(t + 2)})$ *edges.*

Before we prove the theorem, we need some definitions and lemmas. Let $t \geq 1$ and let m be a positive integer. Let g be a function that maps each unweighted *n*-vertex graph to a string of m bits. We say g is an (n, t, m)-*compressor* if the following is true. For all pairs of graphs G_1, G_2, if $g(G_1) = g(G_2)$, then, for all pairs of vertices u, v, $Distance(u, v, G_1) \leq t \cdot Distance(u, v, G_2)$. Informally, the compressor is just like a hash function. It partitions all graphs into groups, such that in each group the graphs have approximately the same distances between any given pair of vertices. Clearly, as t increases, it should be possible to construct compressors with fewer groups. The following lemma provides a lower bound on the number of groups, given any t.

Lemma 10. *For every pair of integers* $t \geq 1$ *and* $n \geq 3$, (n, t, m)-*compressors exist only for* $m \geq \frac{1}{8} \cdot n^{1 + 4/3(t + 2)}$.

Proof. By Lemma 9, for every pair of integers $t \geq 1$ and $n \geq 3$, there exists an *n*-vertex graph with girth $\geq t + 2$ and at least $\frac{1}{8} \cdot n^{1 + 4/3(t + 2)}$ edges. Let $G = (V, E)$ be such a graph, and let G_1, G_2 be two different subgraphs of G. By the girth condition of G, both subgraphs have to belong to different groups of the compressor. Thus the total number of groups is $\geq 2^{|E|}$, thus $m \geq |E|$. $\qquad\square$

Two further observations are useful in proving the theorem. First, existence of spanners (even Steiner spanners) for all graphs implies the existence of a com-

pressor. For example, consider a collection of t-spanners, one for each n-vertex graph. Let the set of spanners (without repetitions) be $\{G'_1, \ldots, G'_k\}$. Then, setting $g(G)$ to be the binary representation of the integer i if G'_i is a spanner of G, yields an $(n, t, \lceil \log k \rceil)$-compressor.

Second, we encode an unweighted graph as a bit string in the following way. Every n-vertex graph with m edges can be encoded by $2 \cdot \lceil \log n \rceil \cdot m$ bits, by representing every edge as a pair of $\lceil \log n \rceil$ bit strings, each in turn representing a vertex.

Proof of Theorem 6. We first prove the theorem for unweighted spanners, then prove it for weighted spanners. To prove the first part, consider any collection of t-spanners, one for each n-vertex graph. Since we are interested in sparse Steiner spanners of n-vertex graphs, even with auxiliary vertices these spanners should have no more than $n + 2 \cdot n(n-1)/2 = n^2$ vertices. Let m be the size of the largest spanner in this set. Each spanner can be encoded in $\leq 4 \cdot \lceil \log n \rceil \cdot m$ bits. Thus by our previous observation, these spanners imply the existence of an $(n, t, 4 \cdot \lceil \log n \rceil \cdot m)$-compressor, but, by Lemma 10, $4 \cdot \lceil \log n \rceil \cdot m \geq \frac{1}{8} \cdot n^{1 + 4/3(t+2)}$. Solving for m proves the theorem.

To prove the second part, again consider a collection of t-spanners, one for each n-vertex graph with unit edge weights. As before these spanners cannot have more than n^2 vertices. However, since the edges have weights, we have to devise a different encoding scheme. Informally the idea is to encode the weights themselves as bit strings. The problem arises because weights may require an arbitrary number of bits. However, we show that by ignoring very large weights, and rounding off the remainder in a certain way, we can get suitably encodable spanners with slightly larger stretch factors.

Consider any t-spanner G' in the set. First notice that any edge with weight larger than $n - 1$ can be removed without affecting the stretch factor. Let G'' be another spanner derived from G' in the following way. For each edge e replace its weight by $Weight''(e) = \lceil n^3 \cdot Weight(e) \rceil / n^3$. Notice that $Weight''(e) < Weight(e) + 1/n^3$.

We now claim that G'' is a $(t + 1/n)$-spanner. This is easy to see because any path in G'' has at most $n^2 - 1$ edges, and each weight in G'' is at most $1/n^3$ greater than the corresponding weight in G'.

G'' is more convenient to code than G'. Let m be the size of the largest G'' in the set. Each edge weight is a rational, where the numerator is an integer between 1 and $n^4 - n^3$ and the denominator is n^3. Thus each weight can be encoded by less than $\lceil \log n^4 \rceil \leq 4 \lceil \log n \rceil$ bits. Thus G'' can be encoded by at most $8 \cdot \lceil \log n \rceil \cdot m$ bits. So these spanners imply the existence of an $(n, t, 8 \cdot \lceil \log n \rceil \cdot m)$-compressor. Applying Lemma 10 and solving for m, the theorem follows.   □

Note that we have no lower bounds on the weight sparseness of Steiner spanners. We now generalize spanners in another direction. Let $t \geq 1$. $G' = (V, E)$ is a *t-approximator* of $G = (V, E)$ if, for all $u, v \in V$, $1/t \cdot Distance(u, v, G') \leq Distance(u, v, G') \leq t \cdot Distance(u, v, G)$. The following theorem provides a lower bound on the size of approximators.

Theorem 7. *For every pair of integers $t \geq 1$ and $n \geq 3$, there exists an n-vertex graph G with unit edge weights such that every t-approximator has $\Omega(1/\log n) \cdot n^{1+4/3(t+2)}$ edges.*

The proof is very similar to Theorem 6 and we omit the straightforward elaboration. Basically a set of approximators implies the existence of a compressor, and an approximator with m edges can be encoded in $O(\log n \cdot m)$ bits.

We finish this section by proving a lower bound for *complete* graphs with weights that are *proper*. The latter mean that the weight of any edge is never more than the length of any path between its end vertices.

Theorem 8. *For every pair of integers $t \geq 1$ and $n \geq 3$, there exists an n-vertex complete graph K with proper edge weights, such that every $(2t - 1)$-spanner requires $\Omega(1/t \cdot n^{1+4/3(2t+1)})$ edges.*

Proof. Let $G = (V, E)$ by an n-vertex graph with at least $\frac{1}{8} \cdot n^{1+4/3(2t+1)}$ edges and girth $\geq 2t + 1$. Define a complete graph K over the same vertex set where the weight of $[u, v] = Distance(u, v, G)$. Clearly, all weights are proper.

Let $G' = (V, E')$ be a $(2t - 1)$-spanner of K. Every edge $[u', v']$ in E' with weight $\leq t$ corresponds to a unique path $u' = u_0, u_1, \ldots, u_{Weight([u',u'])} = v'$ in G, due to the girth of G. An edge $[u, v]$ in E is called *saturated* byn $[u', v']$ in E', if it is contained in this path. If $[u, v]$ in E would not be saturated by some edge of G', then $Distance(u, v, G') \geq 2t$ by the girth condition. This contradicts the fact that G' is a $(2t - 1)$-spanner of K. Thus every edge of G is saturated. Every edge of G' with weight $\leq t$ can saturate at most t edges in G. Thus $Size(G') \geq 1/t \cdot Size(G)$ and the theorem follows. $\square$

All the above lower bound results emphasize robustness, that is, they hold even after allowing more general spanners, or more restrictive graphs. In the next section we discuss spanners of Euclidean graphs.

5. Spanners in Euclidean Graphs

In Euclidean graphs the weights assigned to the edges are not arbitrary, hence the lower bounds of Section 3 do not apply. It has been possible to construct spanners for such graphs with a linear number of edges.

Our first result is on Euclidean graphs on the plane. The vertices of such a graph are a set of V of n points on the plane. The edges are line segments joining pairs of vertices and each edge weight is the Euclidean distance between the vertices. These graphs may be either planar or nonplanar. We let $K(V)$ be the complete Euclidean graph (which is clearly nonplanar). We pose the question: are there sparse *planar* spanners of $K(V)$? This problem has been extensively studied in the past. In [C], [DFS] it was shown that *Delaunay triangulations* in the $\| \cdot \|_1$ and $\| \cdot \|_2$ norms are spanners with constant stretch factors. Using a general framework, Das and Joseph [DJ] showed that other planar graphs such as *greedy*

triangulations and *minimum weight triangulations* also have constrant stretch factors (with different constant values). In [KG] the stretch factor for Delaunay triangulations in the $\| \cdot \|_2$ norm was improved to 2.42 (current best). In [DJ] and [LL] it was shown that there exist extremely short Euclidean planar graphs (that is, almost as short as the minimum spanning tree) that have constant stretch factors. The algorithm in [LL] produces arbitrarily short graphs based upon a parameter, though it is not obvious how small the *sizes* of these graphs are. Using Theorem 2 we can produce planar spanners that are both short, as well as small, in size.

Let $t > 0$, and consider the Delaunay triangulation over V in the $\| \cdot \|_2$ norm. It is known that this triangulation contains $MST(K(V))$. We now apply the *SPANNER* algorithm on the triangulation with $r = 2t + 1$. Because spanners are transitive, it is easy to see that the output (denoted as G') satisfies the following property.

Theorem 9.   *G' is a $(2.42) \cdot (2t + 1)$-spanner of $K(V)$, such that*

(1) $Size(G') \leq (n - 1)(1 + 1t)$,
(2) $Weight(G) < Weight(MST(K(V)))(1 + 1/t)$.

In higher dimensions planarity of spanners is not an issue. Our next result is on constructing linear-sized spanners for complete graphs of point sets V in $(R^d, \| \cdot \|)$, for all dimensions $d \geq 2$, and all norms $\| \cdot \|$. Fix some angle $\delta > 0$. The key idea is to cover R^d by finitely many open cones $C_1, \ldots, C_{s(\delta)}$, all with the same focus at the origin O, such that, for all points u, v in the same cone, the angle between u and v at O is less than δ. Such a covering exists for every d and every δ by the theorem of Heine–Borel [CS]. Similar ideas for constructing spanners for fixed norms have been considered in [K].

We construct the spanner G' as follows. For every $v \in V$, consider the covering of R^d by the cones $C_1 + v, \ldots, C_{s(\delta)} + v$, where $C + v$ represents a shifting of the cone C to a new origin v, in the spirit of Minkowski. For every cone $C_i + v$, let u be the vertex in the cone such that $\|u - v\|$ is minimized. We add $[u, v]$ to G', and u is known as the ith neighbor of v.

Clearly, the size of G' is $\leq s(\delta) \cdot n$, and is therefore linear. It remains to show that its stretch factor is small. Consider $u, v \in V$. A short path between them is constructed in the following way. Let u be inside the cone $C_i + v$. Go from v to its ith neighbor, and proceed from there in the same way via u. We show that this path is not too long with respect to $\|u - v\|$.

Lemma 11.   *For every $\varepsilon < 1/(1 + 2\sqrt{2})$, there is an angle $\delta(\varepsilon) > 0$ such that*

$$Distance(u, v, G') < \frac{\|u - v\|}{(1 - \varepsilon(1 + 2\sqrt{2}))} \quad \textit{for all } u, v \textit{ in } V.$$

The estimation of the angle $\delta(\varepsilon)$ is very technical and we prove it after stating the final theorem of the paper, which follows immediately from Lemma 11.

Theorem 10. *For every* $t > 1$, *dimension* d, *and norm* $\|\cdot\|$ *of* R^d, *there exists a constant* $c(t, d, \|\cdot\|)$ *such that every finite set* V *in* R^d *has a t-spanner with at most* $c \cdot n$ *edges.*

In the rest of this section we estimate the angle $\delta(\varepsilon)$ and thus prove Lemma 11. The following lemmas will be useful in the proof.

Lemma 12. *Let* $\alpha, \beta, \gamma > 0$, $\alpha + \beta + \gamma = \pi$, $\gamma \leq \beta$, *and* $\alpha \leq 2\pi/3$. *Then*

$$\frac{\sin \beta - \sin \alpha}{\sin \gamma} \geq 1 - \alpha.$$

Proof. Fix $\alpha > 0$ and define

$$g(\gamma) = \frac{\sin(\pi - \alpha - \gamma) - \sin \alpha}{\sin \gamma}.$$

We have to show that $g(\gamma) \geq 2\alpha$ for all γ in $(0, (\pi - \alpha)/2]$. By the rule of Hôpital,

$$\lim_{\gamma \to 0} g(\gamma) = \lim_{\gamma \to 0} \frac{-\cos(\pi - \alpha - \gamma)}{\cos \gamma} = -\cos(\pi - \alpha) = \cos \alpha = 1 - 2 \sin^2 \frac{\alpha}{2} \geq 1 - \alpha.$$

For the other extreme in the interval we get

$$g\left(\frac{\pi - \alpha}{2}\right) = \frac{\sin(\pi - \alpha)/2 - \sin \alpha}{\sin((\alpha - \alpha)/2)} = 1 - \frac{\sin \alpha}{\sin((\pi - \alpha)/2)}$$

$$= 1 - \frac{\sin \alpha}{\cos \alpha/2} = 1 - 2 \sin \frac{\alpha}{2} \geq 1 - \alpha.$$

We complete the proof by showing that the derivative $g'(\gamma)$ has no zero in the open interval $(0, (\pi - \alpha)/2)$. Clearly,

$$g'(\gamma) = \frac{-\cos(\pi - \alpha - \gamma) \sin \gamma - [\sin(\pi - \alpha - \gamma) - \sin \alpha] \cos \gamma}{\sin^2 \gamma}.$$

An application of the addition formula for the sinus yields

$$g'(\gamma) = \frac{-\sin(\pi - \alpha) + \sin \alpha \cos \gamma}{\sin^2 \gamma}.$$

The numerator equals $-\sin \alpha + \sin \alpha \cos \gamma$ and is smaller than 0 for all $0 < \gamma < (\pi - \alpha)/2$. $\square$

Lemma 13. *In the Euclidean norm $\|\cdot\|_2$, every triangle with side lengths $c \le b$ and angle $\alpha < 2\pi/3$ satisfies $(b - a)/c \ge 1 - \alpha$.*

Proof. Apply the sinus theorem $a/\sin \alpha = b/\sin \beta = c/\sin \gamma$ to Lemma 12. ☐

In the rest of the section we use the following notation. The angle between any two points u and v with respect to the origin in R^d will be written as $\angle(u, v)$. Also $\sin \angle(u, v)$ will be written in short as $\sin(u, v)$.

Lemma 14. *Consider the norm $\|\cdot\|_2$ and any other norm $\|\cdot\|$ in R^d, such that $\|u\| \le q\|u\|_2$ for all u in R^d, where $q > 0$ is an appropriate constant. Then*

$$\left| \frac{\|u\|}{\|u\|_2} - \frac{\|v\|}{\|v\|_2} \right| \le \sqrt{2}q|\sin(u, v)| \quad \text{for all } u, v \text{ other than the origin.}$$

Proof. The triangle inequalities are $\|u\| - \|v\| \le \|u - v\|$ and $\|u\| - \|v\| \le \|u + v\|$. Without loss of generality assume $\|u\|_2 = \|v\|_2 = 1$ and $\|u\| \ge \|v\|$. Let $\langle \cdot, \cdot \rangle$ denote the standard scalar product.

Case 1: $\langle u, v \rangle \ge 0$. By the first triangle inequality we get

$$\left| \frac{\|u\|}{\|u\|_2} - \frac{\|v\|}{\|v\|_2} \right|^2 \le \|u - v\|^2 \le q^2\|u - v\|_2^2 = q^2\langle u - v, u - v \rangle$$

$$= q^2(\langle u, u \rangle + \langle v, v \rangle - 2\langle u, v \rangle) = 2q^2(1 - \langle u, v \rangle).$$

On the other hand,

$$|\sin(u, v)|^2 = 1 - \cos^2(u, v) = 1 - \langle u, v \rangle^2 = (1 - \langle u, v \rangle)1 + \langle u, v \rangle)$$

$$\ge 1 - \langle u, v \rangle.$$

Case 2: $\langle u, v \rangle < 0$. This case is solved analogously by applying the other triangle inequality. ☐

Lemma 15. *Let $\|u\|_2 \le \|u\| \le q\|u\|_2$ for some $q \ge 1$ and all u in R^d. Let $0 < \varepsilon < 1/(1 + 2\sqrt{2})$. Then*

$$\frac{\|u\| - \|u - v\|}{\|v\|} > 1 - (1 + 2\sqrt{2})\varepsilon$$

for all u, v other than the origin with $\|u\|_2 \ge \|v\|_2$ and $\angle(u, v) < \varepsilon/q$.

Proof. By Lemma 13 we have

$$\frac{\|u\|_2 - \|u - v\|_2}{\|v\|_2} > 1 - \frac{\varepsilon}{q}. \tag{1}$$

Consider the expression $(\|u\| - \|v - u\|)/\|v\|$. By Lemma 14 and multiplying the numerator and denominator by $\|u\|_2/\|u\|$ we get

$$
\frac{\|u\| - \|u - v\|}{\|v\|} \geq \frac{\|u\| - \|v - u\|_2[\|u\|/\|u\|_2 + \sqrt{2q}|\sin(u, v - u)|]}{\|v\|_2[\|u\|/\|u\|_2 + \sqrt{2q}|\sin(u, v)|]}
$$

$$
= \frac{\|u\|_2 - \|v - u\|_2[1 + (\|u\|_2/\|u\|)\sqrt{2q}|\sin(u, v - u)|]}{\|v\|_2[1 + (\|u\|_2/\|u\|)\sqrt{2q}|\sin(u, v)|]}
$$

$$
\geq \frac{\|u\|_2 - \|v - u\|_2[1 + \sqrt{2q}|\sin(u, v - u)|]}{\|v\|_2[1 + \sqrt{2q}|\sin(u, v)|]}
$$

$$
= \frac{1}{1 + \sqrt{2q}|\sin(u, v)|}
$$

$$
\times \left[\frac{\|u\|_2 - \|v - u\|_2}{\|v\|_2} - \frac{\|v - u\|_2\sqrt{2q}|\sin(u, v - u)|}{\|v\|_2} \right]. \quad (2)
$$

By the sinus theorem we have $\sin(u, v - u) = (\|v\|_2/\|v - u\|_2)\sin(u, v)$. Together with (1) this shows that (2) is greater than

$$
\frac{1}{1 + \sqrt{2q}\sin(\varepsilon/q)} \cdot \left[1 - \frac{\varepsilon}{q} - \sqrt{2q}\sin\left(\frac{\varepsilon}{q}\right) \right] \geq \frac{1}{1 + \sqrt{2q}(\varepsilon/q)} \cdot \left[1 - \varepsilon - \sqrt{2q}\frac{\varepsilon}{q} \right]
$$

$$
= \frac{1 - (1 + \sqrt{2})\varepsilon}{1 + \sqrt{2}\varepsilon}
$$

$$
> 1 - (1 + 2\sqrt{2})\varepsilon. \qquad \square
$$

Proof of Lemma 11. Let the norm under consideration be $\|\cdot\|$ such that $\|u\|_2 \leq \|u\| \leq q\|u\|_2$ for some $q \geq 1$ and all u in R^d. Select the angle $\delta(\varepsilon)$ to be ε/q. Recall the manner in which a path from v to u is constructed in the graph G'. Let the vertices along this path be $v = z_0, \ldots, z_p = u$. The following are true for all i:

$$
\angle(z_p - z_i, z_{i+1} - z_i) < \frac{\varepsilon}{q} \quad \text{and} \quad \|z_p - z_i\|_2 \geq \|z_{i+1} - z_i\|_2.
$$

By Lemma 15 we get

$$
\|z_{i+1} - z_i\| < \frac{1}{1 - (1 + 2\sqrt{2})\varepsilon} [\|z_p - z_i\| - \|z_p - z_{i+1}\|].
$$

Thus,

$$
\sum_{i=0}^{p-1} \|z_{i+1} - z_i\| < \frac{1}{1 - (1 + 2\sqrt{2})\alpha} \sum_{i=0}^{p-1} [\|z_p - z_i\| - \|z_p - z_{i+1}\|]
$$

$$= \frac{1}{1 - (1 + 2\sqrt{2})\varepsilon} \left[\|z_p - z_0\| - z_p - z_p\| \right]$$

$$= \frac{1}{1 - (1 + 2\sqrt{2})\varepsilon} \|zp - z_0\|. \qquad \square$$

A direct application of Lemma 13 shows that in the case of the $\| \cdot \|_2$ norm we get $1/(1 - \varepsilon)$ spanners with $s(\delta(\varepsilon)) \cdot n$ edges. In $R^2, s(\delta(\varepsilon))$ is known to be $\lfloor 2\pi/\varepsilon \rfloor + 1$.

6. Open Problems

We conclude with some open problems.

(1) In Theorem 1 the bound in the weight does not agree with the lower bound as nicely as in the other results. Can it be improved? We feel that our strategy of dividing the graph into planar components cannot be extended to yield the optimal answer.

(2) Even in the other results, there are gaps between upper and lower bounds. For instance, in Theorem 1 the upper bound for the size is $O(n^{1 + 1/t})$, while the lower bound is $\Omega(n^{1 + 4/(6t + 9)})$. Only for $t = 1$ are the known bounds, $\Theta(n^{3/2})$, of the same order. The lower bound was proved in [L], and the upper bound is mentioned in [PUP].

(3) For dimensions higher than 2, Euclidean spanners with linear sizes exist. Do Euclidean spanners exist with weights within a constant multiple of the weight of the minimum spanning tree?

(4) What spanners do random graphs have? The Euclidean random case has been examined in [SV].

(5) Consider R^d, some fixed norm, and $t > 1$. What are the worst (or at least bad) point configurations V with respect to the number of edges in optimal t-spanners?

(6) Do spanners have other applications?

Acknowledgments

References

[A] I. Althöfer, On Optimal Realizations of Finite Metric Spaces by Graphs, *Discrete Comput. Geom.* 3 (1988), 103–122.

[Aw] B. Awerbuch, Complexity of Network Synchronization, *J. Assoc. Comput. Mach.* (1985), 804–823.

[ABLP] B. Awerbuch, D. Bar-Noy, N. Linial, D. Peleg, Compact Distributed Data Structures for Adaptive Routing, *Proc. STOC*, 1989, pp. 479–489.

[AP] B. Awerbuch, D. Peleg, Routing with Polynomial Communication–Space Tradeoff, *SIAM J. Discrete Math.*, to appear.

[BD] H. J. Bandelt, A. W. M. Dress, Reconstructing the Shape of a Tree from Observed Dissimilarity Data, *Adv. in Appl. Math.* 7 (1986), 309–343.

[Be] M. Bern, Private Communication to David Dobkin, 1989.

[Bo] B. Bollobás, *Extremal Graph Theory*, Academic Press, New York, 1978.

[C] L. P. Chew, There is a Planar Graph Almost as Good as the Complete Graph, *Proc. ACM Symp. on Computational Geometry*, 1986, pp. 169–177.

[CS] J. H. Conway, N. J. A. Sloane, *Sphere Packing, Lattices, and Groups*, Springer-Verlag, New York, 1988.

[D] A. W. M. Dress, Trees, Tight Extensions of Metric Spaces, *Adv. in Math.* 53 (1984), 321–402.

[DFS] D. P. Dobkin, S. J. Friedman, K. J. Supowit, Delaunay Graphs Are Almost as Good as Complete Graphs, *Proc. FOCS*, 1987, pp. 20–26.

[DJ] G. Das, D. A. Joseph, Which Triangulations Approximate the Complete Graph?, *Proc. Internat. Symp. on Optimal Algorithms*, 1989, pp. 168–192, LNCS, Vol. 401, Springer-Verlag, Berlin.

[ES] P. Erdös, H. Sachs, Reguläre Graphen gegebener Taillenweite mit minimaler Knotenzahl, *Wiss. Z. Martin-Luther-Univ. Halle-Wittenberg Math.-Natur. Reihe* 12 (1963), 251–257.

[K] J. M. Keil, Approximating the Complete Euclidean Graph, *Proc. SWAT*, 1988, pp. 208–213, LNCS, Vol. 318, Springer-Verlag, Berlin.

[KG] J. M. Keil, C. A. Gutwin, The Delaunay Triangulation Closely Approximates the Complete Euclidean Graph, *Proc. WADS*, 1989, pp. 47–56, LNCS, Vol. 382, Springer-Verlag, Berlin.

[LL] C. Levcopoulos, A. Lingas, There Are Planar Graphs Almost as Good as the Complete Graphs and as Short as Minimum Spanning Trees, *Proc. Internat. Symp. on Optimal Algorithms*, 1989, pp. 9–13, LNCS, Vol. 401, Springer-Verlag, Berlin.

[L] J. Longyear, Regular d-valent Graphs of Girth 6 and $2(d*d-d+1)$ Vertices, *J. Combin. Theory* 9 (1970), 420–422.

[LPS] A. Lubotzky, R. Phillips, P. Sarnak, Ramanujan Graphs, *Combinatorica* 8(3) (1988), 261–277.

[M] G. A. Margulis, Explicit Group-Theoretical Constructions of Combinatorial Schemes and Their Application to the Design of Expanders and Concentrators, *Problems Inform. Transmission* 24(1) (1988), 39–46 (translated from *Problemy Peredachi Informatsii*).

[PS] D. Peleg, A. Schäffer, Graph Spanners, *J. Graph Theory* 13(1) (1989), 99–116.

[PUl] D. Peleg, J. Ullman, An Optimal Synchronizer for the Hypercube, *SIAM J. Comput.* 18 (1989), 740–747.

[PU] P. Peleg, E. Upfal, A Tradeoff Between Space and Efficiency for Routing Tables, *Proc. STOC*, 1988, pp. 43–52.

[RS] P. Raghaven, M. Snir, Memory Versus Randomness in Online Algorithms, *Proc. ICALP*, 1989, pp. 687–703, LNCS, Vol. 372, Springer-Verlag, Berlin.

[SV] R. Sedgewick, J. S. Vitter, Shortest Paths in Euclidean Graphs, *Algorithmica* 1 (1986), 31–48.

[S] J. M. S. Simões-Pereira, A Note on the Tree Realizability of a Distance Matrix, *J. Combin. Theory* 6 (1969), 303–310.

[T] R. E. Tarjan, *Data Structures and Network Algorithms*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1983.