Remote spanners: what to know beyond neighbors

Philippe Jacquet, Laurent Viennot

Remote-Spanners: What to Know beyond Neighbors

Philippe Jacquet*
INRIA
Rocquencourt, France
`philippe.jacquet@inria.fr`

Laurent Viennot†
INRIA
Paris, France
`laurent.viennot@inria.fr`

Abstract

Motivated by the fact that neighbors are generally known in practical routing algorithms, we introduce the notion of remote-spanner. Given an unweighted graph G, a sub-graph H with vertex set $V(H) = V(G)$ is an (α, β)-remote-spanner *if for each pair of points u and v the distance between u and v in H_u, the graph H augmented by all the edges between u and its neighbors in G, is at most α times the distance between u and v in G plus β. We extend this definition to k-connected graphs by considering the minimum length sum over k disjoint paths as a distance. We then say that an (α, β)-remote-spanner is* k-connecting.

In this paper, we give distributed algorithms for computing $(1 + \varepsilon, 1 - 2\varepsilon)$-remote-spanners for any $\varepsilon > 0$, k-connecting $(1, 0)$-remote-spanners for any $k \geq 1$ (yielding $(1, 0)$-remote-spanners for $k = 1$) and 2-connecting $(2, -1)$-remote-spanners. All these algorithms run in constant time for any unweighted input graph. The number of edges obtained for k-connecting $(1, 0)$-remote-spanner is within a logarithmic factor from optimal (compared to the best k-connecting $(1, 0)$-remote-spanner of the input graph). Interestingly, sparse $(1, 0)$-remote-spanners (i.e. preserving exact distances) with $O(n^{4/3})$ edges exist in random unit disk graphs. The number of edges obtained for $(1+\varepsilon, 1-2\varepsilon)$-remote-spanners and 2-connecting $(2, -1)$-remote-spanners is linear if the input graph is the unit ball graph of a doubling metric (even if distances between nodes are unknown). Our methodology consists in characterizing remote-spanners as sub-graphs containing the union of small depth tree sub-graphs dominating nearby nodes. This leads to simple local distributed algorithms.

1. Introduction

This paper concerns the characterization and the distributed computation of sparse remote-spanners. Given an unweighted graph G, a sub-graph H with vertex set $V(H) = V(G)$ is an (α, β)-*remote-spanner* if it approximates distances in G from any node u when it is completed with all neighboring links of u. More precisely, for any two nonadjacent nodes u, v, the inequality $d_{H_u}(u, v) \leq \alpha\, d_G(u, v) + \beta$ is satisfied, where H_u is the sub-graph with edge set $E(H) \cup \{uv \mid v \in N(u)\}$ and d_{H_u} is the distance in H_u. (Note that $d_{H_u}(u, v) = 1 = d_G(u, v)$ when u and v are adjacent). (α, β) is called the stretch. This can be reformulated as follows: for each pair of nodes u and v, there exist a node x adjacent to u in G such that the distance between x and v in H is at most α times the distance between u and v in G plus $\beta - 1$. Figure 1 illustrates an example of a graph (a), a $(1, 0)$-remote-spanner (b) of this graph and a $(1, 1)$-remote-spanner (c) which is also a $(2, -1)$-remote-spanner.

We introduce this notion based on the functioning of routing protocols used in practical networks where each router generally knows its list of neighbors. This is particularly the case for link state routing that was introduced by McQuillian et al. [19] as a replacement for distance vector routing. It was then standardized as OSPF protocol [20], [21] which is widely used in the Internet. With a very high level description, link state routing basically consists in two periodic procedures. First, each router sends regularly probing messages on its network interfaces to discover its neighbors. Second, it regularly floods the network with link state advertisement messages containing its list of neighbors. Each node then knows its list of neighbors and the whole network topology. The next hop for each destination is then deduced from a shortest path computation.

This can be very costly in a large and dense network, a case that can be encountered in ad hoc networks where wireless connections may provide many neighbors to each node. To optimize link state routing in such situations, it was proposed more recently to alleviate the cost of link state advertisements by flooding only a subset of links [14]. This was standardized by IETF as the OLSR routing protocol [3]. This principle can be indeed applied to any link state routing protocol: broadcast only a subset of links to all nodes, thus

* *INRIA Project-team "HIPERCOM".*
† *Project-Team "GANG" between INRIA and LIAFA. Supported by ANR project "ALADDIN" and CRC "MARDI" with France-Telecom Orange.*

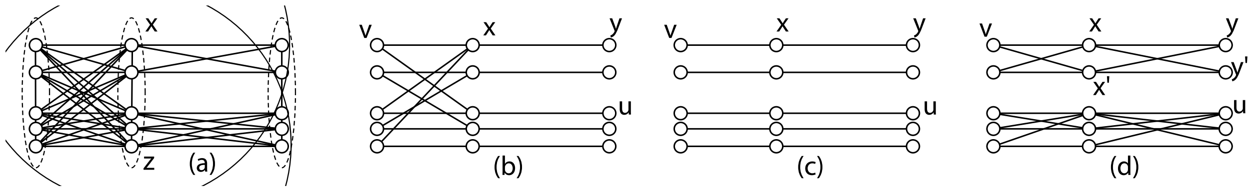

Figure 1. (a) A unit disk graph G^a where two nodes are connected if one is in the unit disk centered at the other (nodes in a dashed oval are all pairwise connected). The unit disks centered at x and z are partially plotted. (b) A $(1,0)$-remote-spanner H^b of G^a. For example, $d_{H^b_u}(u,x) = 2 = d_{G^a}(u,x)$. (Edge uy is in not in H^a, but is in H^a_u since $y \in N(u)$ in G^a). (c) A $(2,-1)$-remote-spanner H^c of G^a. For example, $d_{H^c_u}(u,v) = 3 = 2d_{G^a}(u,v) - 1$ through the path $uyxv$. (d) A 2-connecting $(2,-1)$-remote-spanner H^d of G^a. H^d_u contains two disjoint paths of length 3 from u to v: $uyxv$ and $uy'x'v$.

defining a sub-graph H. As each node u regularly discovers its neighbors, it can augment this graph with its neighboring links, to obtain a sub-graph H_u with edge set $E(H) \cup \{uv \mid v \in N(u)\}$. It then computes its routing tables according to distances in H_u: it forwards packets with destination v to a closest neighbor u' to v in H_u. u' then forwards similarly the packet and so on. This results in a classical greedy routing scheme. As the path from u' to v in H_u is included in H, it is known by u', implying $d_{H_{u'}}(u',v) \leq d_{H_u}(u,v) - 1$. We thus see that this greedy routing from u to v results in a route of length at most $d_{H_u}(u,v)$. The notion of (α,β)-remote-spanner thus formalizes the required properties on the broadcasted sub-graph H to ensure that greedy routing performs with stretch at most (α,β). Note that the definition of distances in the remote spanner, i.e. $d_{H_u}(u,v)$ versus $d_{H_v}(v,u)$, is asymmetric with respect to u and v as is the knowledge of u and v in a link state routing protocol.

Our formalization is inspired by the regular notion of graph spanner introduced by Peleg et al. [23], [22]. An (α,β)-spanner is a sub-graph H preserving (u,v) distance by ensuring $d_H(u,v) \leq \alpha\, d_G(u,v) + \beta$ for all nodes u,v. In comparison, for remote spanners, the preservation of distance is aided by including all edges incident to the source node u even if some are not part of the spanner H. Spanners are key ingredients of various distributed applications, e.g., synchronizers, compact routing, distance oracles broadcasting, etc. Recent reviews of the literature on spanners can be found in [24], [26]. We believe that part of this work can be investigated in the context where a node knows its neighboring links in addition to the spanner as this information is usually accessible in practical routing context. This is the reason why we introduce remote-spanners.

Similarly to spanners, remote-spanner constructions are to be evaluated along three worst-case measures: approximation quality (i.e. small values of α and β), number of edges and construction time. Additionally, we are interested in multi-conectivity properties. Preserving multi-connectivity has practical interest for improving reliability of the network and to allow multi-path routing. We say that two nonadjacent nodes u and v are k-connected in G if there exists k pairwise disjoint paths from u to v (i.e. having no internal node in common). A remote-spanner H is said to be k-*connecting* if for all nonadjacent nodes u,v and all positive integer $k' \leq k$, u and v are k'-connected in H_u if they are k'-connected in G. Additionally, we require that the stretch of the length sum of these paths is bounded, i.e. $d^{k'}_{H_u}(u,v) \leq \alpha\, d^{k'}_G(u,v) + k'\beta$ where $d^{k'}_K(u,v)$ is minimum length sum of k' disjoint paths in a sub-graph K. Figure 1(d) illustrates an example of 2-connecting $(2,-1)$-remote-spanner.

An (α,β)-spanner is always an (α,β)-remote-spanner. We thus obtain a wider class of sub-graphs that allows several improvements over regular spanners.

- First, $(1,0)$-remote-spanners (i.e. exact distances are preserved) can be *sparse* (e.g.$O(n^{4/3})$ on average in random unit disk graphs) whereas a $(1,0)$-spanner must obviously include all edges (see Figure 1 (b) for an example). Additionally, such $(1,0)$-remote-spanners can be computed within a logarithmic factor from optimal (compared to the number of edges of the best $(1,0)$-remote-spanner of the input graph). In comparison, spanner construction algorithms usually give a controlled approximation ratio compared to the best spanner of the worst possible graph.
- We show that remote-spanners have local characterizations that yield simple distributed algorithms for computing them. In particular, no synchronization between node decisions is necessary (based on the topology knowledge up to some constant distance, a node can decide which edges to add to the remote-spanner independently from other node decisions). This is not the case for existing distributed algorithms computing spanners [2], [10].
- Remote-spanners allow to extend the notion of stretch to multi-connected graphs in a novel man-

ner: by considering sum of lengths of disjoint paths. Similar properties were only studied in the context of (fault-tolerant) geometrical spanners where the graph is given by all pair distances in an euclidean space [17], [18], [7]. This setting cannot be extended to graphs in general.

1.1. Our results

We characterize some remote-spanner classes as unions of small depth tree sub-graphs dominating nearby nodes. More precisely, given a node u we define an (r,β)*-dominating tree* T *for* u as a tree sub-graph rooted at node u such that for all v at distance r' from u with $2 \leq r' \leq r$, there exists $x \in N(v) \cap V(T)$ with $d_T(u,x) \leq r'-1+\beta$. In other words, $V(T)$ dominates the ball $B_G(u,r)$ of radius r centered at u in the graph G and T induces paths of stretch $(1,\beta)$ to any node $v \in B_G(u,r)$. (We mainly consider $\beta = 0$ or $\beta = 1$). We say that a sub-graph H *induces* (r,β)-dominating trees if it contains an (r,β)-dominating tree for each node in the graph, i.e. for all node u there exist an (r,β)-dominating tree T for u with $E(T) \subseteq E(H)$.

In Section 2, we study low stretch remote-spanners. We provide a distributed algorithm computing a $(1+\varepsilon, 1-2\varepsilon)$-remote-spanner in $O(\varepsilon^{-1})$ time. It has $O(n\varepsilon^{-(p+1)})$ edges if the input graph is the unit ball graph (UBG) of a metric e with constant doubling dimension p, i.e. two nodes are neighbors iff $e(u,v) \leq 1$ and any ball of radius R in the metric e can be covered by 2^p balls of radius $R/2$. A particular case of such unit ball graph is the unit disk graph where e is the distance in the plane and two nodes in the plane are neighbors if one is in the unit disk centered at the other. Such graph models are often used to model ad hoc networks. The unit ball centered on a node then corresponds to the area where a radio emission of the node can be successfully received. The algorithm is obtained by proving that for any ε with $0 < \varepsilon \leq 1$, a sub-graph is a $(1+\varepsilon, 1-2\varepsilon)$-remote-spanner iff it induces $\left(\left\lceil \frac{1}{\varepsilon} \right\rceil + 1, 1\right)$-dominating trees.

In Section 3, we study k-connecting remote-spanners. We provide a distributed algorithm computing a k-connecting $(1,0)$-remote-spanner in constant time and with optimal number of edges up to a factor $2(1+\log\Delta)$. Interestingly, its expected number of edges is $O(k^{\frac{2}{3}} n^{\frac{4}{3}} \log n)$ in the unit disk graph model with a uniform Poisson distribution of nodes (compared to $\Omega(n^2)$ for the full topology). Additionally, we propose a distributed algorithm computing in time $O(1)$ a 2-connecting $(2,-1)$-remote-spanner which has $O(n)$ edges if the input graph is the unit ball graph of a metric with constant doubling dimension. To obtain these results, we generalize the $(2,\beta)$-dominating trees as follows. A k*-connecting* $(2,\beta)$*-dominating tree* T for a node u is a tree sub-graph rooted at node u such that for all node v at distance 2 from u, either $uw \in E(T)$ for all $w \in N(u) \cap N(v)$ or v has k neighbors in $V(T)$ such that the paths connecting them to u in T are disjoint (i.e. share only node u) and have length at most $1+\beta$. We then show that a sub-graph is a k-connecting $(1,0)$-remote-spanner iff it induces k-connecting $(2,0)$-dominating trees and that any sub-graph inducing 2-connecting $(2,1)$-dominating trees is a 2-connecting $(2,-1)$-remote-spanner. Algorithms computing such dominating trees allow to obtain the two previously mentioned distributed algorithms. These results are summarized in Table 1 which compares them to the following related results.

1.2. Related work

One can easily see that any (α,β)-spanner is also an (α,β)-remote-spanner and even an $(\alpha,\beta-\alpha+1)$-remote-spanner for $\alpha \geq 1$ (simply consider the spanner stretch from u' to v where u' the first node on a shortest path from u to v in G). All existing algorithms for computing spanners thus also yield remote-spanners (see [24] for a review of best known algorithms). Classical spanner results show that any graph admits a $(2k-1,0)$-spanner with $O(n^{1+1/k})$ edges (see e.g. [24]) and this is believed to be tight, i.e. stretch (α,β) with $\alpha+\beta < 2k-1$ cannot be obtained with $o(n^{1+1/k})$ edges (see e.g. [26]). On the positive side, any graph thus admits a $(2k-1,-2k+2)$-remote-spanner with $O(n^{1+1/k})$ edges. Moreover, the construction in [2] of $(k,k-1)$-spanners with $O(n^{1+1/k})$ edges leads to $(k,0)$-remote-spanners. We suspect that these bounds are also tight for remote-spanners in the sense that stretch (α,β) with $2\alpha+\beta-1 < 2k-1$ cannot be obtained with $o(n^{1+1/k})$ edges on some graphs.

Most notably, compared to our results on remote-spanners, it is known how to distributively compute $(1+\varepsilon,0)$-spanners with $O(n)$ edges in the unit ball graph of a doubling metric [8], [9]. These two papers consider that the unit ball graph is weighted by edge lengths (stretch is considered with respect to path length obtained by summing edge lengths). In particular, they assume that two neighbors are always informed of their relative distance in the underlying metric. Computation of a linear size $(1+\varepsilon,0)$-spanner is made in that setting in $O(\log^* n)$ time [8]. A more general class of graphs motivated by radio propagation models is considered in [9], requiring logarithmic time. Both algorithms make use of maximal independent sets (MIS). Our setting is different: the input is reduced to the graph, and distances in the underlying metric are unknown. This setting appears to be less tractable. For example, the MIS compu-

Type of input	Type of spanner	Number of edges	Comp. time	Ref.
Any graph	$(k, k-1)$-span.	$O(kn^{1+1/k})$	O(k)	[2]
Any graph	$(k, 0)$-rem.-span.	$O(kn^{1+1/k})$	O(k)	using [2]
Any graph	$(1, 0)$-span.	m (all edges)	–	(trivial)
Any graph	k-conn. $(1, 0)$-rem.-span.	$O(\log n)$ from opt.	O(1)	**Th. 2**
rand. UDG	$(1, 0)$-rem.-span.	$O(n^{4/3} \log n)$	$O(1)$	**Th. 2** & [13]
UBG known dist.	$(1+\varepsilon, 0)$-span.	$O(n)$	$O(\log^* n)$	[8]
UBG unknown dist.	$(1+\varepsilon, 1-2\varepsilon)$-rem.-span.	$O(n)$	$O(1)$	**Th. 1**
Points in $\mathbb{R}^d$	k-fault-tol. $(1+\varepsilon, 0)$-span.	$O(kn)$	seq.	[7]
UBG unknown dist.	2-conn. $(2, -1)$-rem.-span.	$O(n)$	$O(1)$	**Th. 3**

Table 1. Remote spanners versus regular spanners depending on assumptions on the input graph. UBG stands for Unit Ball Graph (of a doubling metric) and "rand. UDG" for a unit disk graph with a uniform Poisson distribution of nodes. In both cases, distances in the underlying metric can be known, i.e. are part of the input, or not.

tation can be done in time $O(\log^* n)$ when distances are known [16] whereas the best algorithm in general [15] (up to our knowledge) requires $O(\log \Delta \cdot \log^* n)$ time. Note that we may obviously have $\Delta = \Omega(n)$ in unit ball graphs. With remote-spanners we get a constant time algorithm with similar stretch and number of edges in the more general setting where the underlying metric distances are not given. In fact, our algorithm works properly on any graph, i.e. computes a $(1+\varepsilon, 1-\varepsilon)$-remote-spanner whatever the input is. The linear size of the spanner is guaranteed only in the case of a unit ball graph of a doubling metric.

On the other hand it is possible to compute sparse multi-connected spanners of an euclidean space [6] or a planar graph [5], but stretch is not controlled there. Sparse geometrical spanners with low stretch avoiding a given region in the plane were introduced in [1]. The closest work concerns fault-tolerant geometrical spanners [17], [18], [7]. In that setting, the input is a set of nodes in an euclidean space. The spanned graph is thus the complete graph where edges are again weighted by distances in the plane. A spanner is k-fault-tolerant if the stretch is preserved after removal of any subset of at most k nodes. Given $t > 1$ and $k > 0$ it is always possible to construct a k-fault-tolerant $(t, 0)$-spanner with maximal degree $O(k)$ and with edge length sum within a factor $O(k^2)$ from that of a minimum spanning tree. (The best complexities are obtained in [7]). Note that this is not possible for a graph in general. Considered for example a long cycle. As soon as a node is deleted, the distance between its two neighbors increases from 2 to the length of the cycle minus 2. This definition of stretch in the context of multi-connectivity is well adapted for geometrical spanners but not for spanners in general. Our generalization of stretch to multi-connected graphs overcomes this restriction. (Note that our definition with length of disjoint paths can be used in the context of regular spanners also). As far as we know, remote-spanners are the first skeleton structure enabling at the same time tractability, sparsity, and low stretch with respect to disjoint-path length sum, a natural distance when considering multi-connected graphs.

Interestingly, our dominating trees generalize the notions of multipoint relays introduced in ad hoc networks [14], [3] for optimizing flooding and shortest path routing. They were extended in [27] for providing small connected dominating sets. However the concept of remote-spanner was never introduced before and its relationships with multipoint relays were largely ignored. In our terms, multipoint relays as defined in [14], [3] can be seen as $(2, 0)$-dominating trees. It was already known that they provide shortest path routes, i.e. their union forms a $(1, 0)$-remote-spanner. However, it was not noticed that they are also necessary: any $(1, 0)$-remote-spanner must induce multipoint relays. As multipoint relays are also used for optimizing flooding, this definition was extended to obtain better reliability of flooding with the k-coverage feature [3], [4]. This extension is equivalent to k-connecting $(2, 0)$-dominating trees. It was never proved that this extension indeed ensures k-connectivity. On the other hand, the extended multipoint relays defined in [27] are $(2, 1)$-dominating trees in our terms. They were introduced for computing small connected dominating sets. It was not noticed that they provide $(2, -1)$-remote-spanners. Our definitions of dominating trees extend these notions of multipoint relays in various ways.

2. Remote-spanners with low stretch

We prove in Section 3 that a sub-graph is a $(1, 0)$-remote-spanner iff it induces $(2, 0)$-dominating trees in the more general setting of k-connecting $(1, 0)$-remote-spanners. We now consider $(1 + \varepsilon, O(1))$-remote-spanners.

2.1. $(1+\varepsilon, 1-2\varepsilon)$-remote-spanners

The main idea is to consider a radius r and to require $d_{H_u}(u,v) \le d_G(u,v)+1$ for u,v such that $d_G(u,v) \le r$. This is clearly satisfied if H induces a $(r,1)$-dominating tree for v. Indeed, we can obtain the following characterization.

Proposition 1: For any ε with $0 < \varepsilon \le 1$, a sub-graph is a $(1+\varepsilon, 1-2\varepsilon)$-remote-spanner iff it induces $\left(\left\lceil \frac{1}{\varepsilon} \right\rceil + 1, 1\right)$-dominating trees.

Proof: Set $r = \left\lceil \frac{1}{\varepsilon} \right\rceil + 1$. We first show that inducing $(r,1)$-dominating trees is a necessary condition. If H is a $(1+\varepsilon, 1-2\varepsilon)$-remote-spanner, it satisfies $d_{H_v}(v,u) \le (1+\varepsilon)r' + 1 - 2\varepsilon$ for all nodes u and v such that $d_G(u,v) = r' \ge 2$. For $r' \le r$, we obtain $d_{H_v}(v,u) \le r' + 1 + \varepsilon(r'-2) \le r' + 1 + \varepsilon(\left\lceil \frac{1}{\varepsilon} \right\rceil - 1) < r' + 2$. As $d_{H_v}(v,u)$ is integral, we must have $d_{H_v}(v,u) \le r'+1$. In other words, H contains a path of length at most r' from u to some node in $N(v)$. By considering the union of such paths for all $v \in B_G(u,r) \setminus B_G(u,1)$, we obtain a $(r,1)$-dominating tree for u included in H.

Now consider a sub-graph H inducing $(r,1)$-dominating trees and a pair of nodes u,v. Let $\ell = d_G(u,v)$ denote their distance. We show $d_{H_u}(u,v) \le \left(1 + \frac{1}{r-1}\right)\ell + 1 - \frac{2}{r-1}$ by induction on ℓ. It is verified for $\ell = 1$ since uv is then in $E(H_u)$ and $2 - \frac{1}{r-1} \ge 1$ since $r \ge 2$. For $2 \le \ell \le r$, H contains a $(r,1)$-dominating tree for v. This implies $d_{H_u}(u,v) \le \ell + 1 \le \left(1 + \frac{1}{r-1}\right)\ell + 1 - \frac{2}{r-1}$ as $\ell \ge 2$. Now consider $\ell > r$. Let v' be the node at distance r from v in a shortest path from v to u. As H induces an $(r,1)$-dominating tree T for v, there exists $x \in N(v')$ with $d_T(v,x) \le r$. As a neighbor of v', x is thus at distance at most $\ell - r + 1$ from u. Applying the induction hypothesis, we obtain $d_{H_u}(u,x) \le \left(1 + \frac{1}{r-1}\right)(\ell - (r-1)) + 1 - \frac{2}{r-1} = \left(1 + \frac{1}{r-1}\right)\ell - r + 1 - \frac{2}{r-1}$. As $d_H(x,v) \le r$, we have $d_{H_u}(u,v) \le \left(1 + \frac{1}{r-1}\right)\ell + 1 - \frac{2}{r-1}$. H is thus a $(1+\varepsilon', 1-2\varepsilon')$-remote-spanner with $\varepsilon' = \frac{1}{r-1} = \frac{1}{\lceil \varepsilon^{-1} \rceil} \le \varepsilon$. It is thus a $(1+\varepsilon, 1-2\varepsilon)$-remote-spanner as $(1+\varepsilon')\ell + 1 - 2\varepsilon' \le \ell + (\ell-2)\varepsilon' + 1 \le (1+\varepsilon)\ell + 1 - 2\varepsilon$ since $\ell \ge 2$. □

2.2. Computing dominating trees

It is always possible to compute a (r,β)-dominating tree for node u with size within a logarithmic factor from the minimal such dominating tree for u. However, due to space limitations, we only include algorithms that are used in the results of the paper. The particular case of $(2,0)$-dominating trees is treated in Algorithm 3 ($\textsc{DomTreeGdy}_{2,0,1}(u)$) presented in Section 3.

Algorithm 1 ($\textsc{DomTreeMIS}_{r,1}(u)$) computes a $(r,1)$-dominating tree for node u. It consists in computing greedily a local maximum independent set (MIS) for dominating nodes at distance at most r. This is particularly interesting if the input graph is the unit ball graph of a doubling metric as the size of a MIS is then bounded.

```
T := ({u}, ∅)
M := ∅, B := B_G(u, r) \ B_G(u, 1)
while B ≠ ∅ do
    Pick x ∈ B at minimal distance from u
      (i.e. d_G(u, v) ≥ d_G(u, x) for all v ∈ B).
    M := M ∪ {x}
    Add to T a shortest path from u to x in G.
    B := B \ B_G(x, 1)
```

Algorithm 1: Algorithm $\textsc{DomTreeMIS}_{r,1}(u)$ for a node u. The tree T is the dominating tree computed for u.

Proposition 2: Algorithm 1 $\textsc{DomTreeMIS}_{r,1}(u)$ computes an $(r,1)$-dominating tree for node u. Additionally, if the input graph is the unit ball graph of a metric with constant doubling dimension p, then the computed tree has $O(r^{p+1})$ edges.

Proof: Consider a node v at distance r' from u. Either v is added to T and $d_T(u,x) \le r' - 1$ where x is the next node on the path from v to u in T. Or v is in $N(x)$ for some node x added to M. The choice of x in the while loop implies $r' = d_G(u,v) \ge d_G(u,x)$. As a shortest path from u to x is added to T, we have $d_T(u,x) = d_G(u,x) \le r'$.

Note that the set M is a maximum independent set (MIS) of $B_G(u,r) \setminus B_G(u,1)$ at the end of the algorithm: for x,y in M where y was added after x in M, we have $y \notin B_G(x,1)$. As the metric is doubling, the metric ball of center u and radius r can be covered by $2^{p(\lceil \log_2 r \rceil + 1)}$ metric balls of radius $\frac{1}{2}$. Such a ball of radius $\frac{1}{2}$ contains one node of M at most since M is a MIS. We thus deduce $|M| \le (4r)^p$ and $|E(T)| \le r\,|M| \le 4^p r^{p+1}$. □

2.3. Computing remote-spanners

According to the previous characterizations, distributed algorithms for computing remote-spanners can be obtained by locally computing dominating trees. The general form of our distributed algorithms for computing remote-spanner is thus given by Algorithm 2 which give the procedure $\textsc{RemSpan}_{r,\beta}(u)$ run by each node u.

Running Algorithm $\textsc{RemSpan}_{r,\beta}(u)$ for all u in parallel allows to compute a remote-spanner inducing

Send u to all neighbors and receive identities of neighbors.
Send $N(u)$ to all nodes in $B_G(u, r-1+\beta)$ (and receive $N(v)$ from each v in $B_G(u, r-1+\beta)$).
Compute an (r,β)-dominating tree T_u for u.
Send T_u to all nodes in $B_G(u, r-1+\beta)$.

Algorithm 2: Algorithm $\text{REMSPAN}_{r,\beta}(u)$ for node u.

(r,β)-dominating trees as the union of all T_u in time $2r-1+2\beta$. Using Algorithm $\text{DOMTREEMIS}_{r,1}(u)$ and relying on Proposition 1, we obtain the following result.

Theorem 1: For any $\varepsilon > 0$ with $\varepsilon \leq 1$, a $(1+\varepsilon, 1-2\varepsilon)$-remote-spanner can be computed in time $O(\varepsilon^{-1})$ such that its number of edges is $O(\varepsilon^{-(p+1)}n)$ if the input graph is the unit ball graph of a metric with constant doubling dimension p.

Note that Algorithm $\text{REMSPAN}_{r,\beta}(u)$ can be run as in practical link state routing protocols by regularly performing its four operations in an asynchronous fashion every period of time T and using regular flooding of neighbor lists and computed trees. If a topology change occurs, the computed spanner will stabilize after a time period of $T + 2F$ where F is the time duration of a flooding up to distance $r-1+\beta$.

3. Remote-spanners providing multi-connectivity

We now consider the case where multiple paths from a node s to a node t can be found in G. We consider only simple paths, i.e. a node appears at most once in a path. We say that two paths are *disjoint* if they do not have any internal node in common. Several paths are disjoint if they are pairwise disjoint. We define the *k-connecting distance* $d^k_G(s,t)$ between two nodes s and t as the minimum length sum obtained over all sets of k disjoint paths from s to t. (We set $d^k_G(s,t) = \infty$ if there do not exist k disjoint paths from s to t). We thus have $d^1_G(s,t) = d_G(s,t)$. We similarly define $d^k_H(s,t)$ for any sub-graph H.

Recall that an (α,β)-remote-spanner H is said to be *k-connecting* if it satisfies $d^{k'}_{H_s}(s,t) \leq \alpha\, d^{k'}_G(s,t) + k'\beta$ for all nonadjacent nodes s and t and all positive integer $k' \leq k$. This definition is equivalent to the (α,β)-remote-spanner definition for $k=1$. Let us recall also the definition of a k-connecting $(2,\beta)$-dominating tree T. For that purpose, let $B_T(u,r)$ denote the ball of radius r centered at u in a tree sub-graph T. Given a node u, a *k-connecting $(2,\beta)$-dominating tree T for u* is a tree sub-graph rooted at node u dominating every node v at distance 2 from u in the following sense: either v has k neighbors in $B_T(u, 1+\beta)$ such that the paths connecting them to u in T are disjoint, or $uw \in E(T)$ for all $w \in N(u) \cap N(v)$. This definition is equivalent to the $(2,\beta)$-dominating tree definition for $k=1$.

3.1. 2-connecting $(2,-1)$-remote-spanners

We now show the following result.

Proposition 3: Any sub-graph H inducing 2-connecting $(2,1)$-dominating trees is a 2-connecting $(2,-1)$-remote-spanner.

In the rest of the section, we consider such a sub-graph H and two nonadjacent nodes s and t such that $d^2_G(s,t) < \infty$.

Let $|P|$ denote the length of a path P in number of edges. If two nodes u and v belong to a path P, let $P[u,v]$ denote the sub-path from u to v. If P is a path from u to v and Q is a path from v to w (for disjoint P and Q and $u \neq w$), let $P+Q$ denote the path from u to w obtained by concatenation of P and Q.

If H is a sub-graph and P a path from s to t, we say that P *lies outside* H by i edges if its last $|P|-i$ edges are in $E(H)$, i.e. it has an internal node w such that $|P[s,w]| = i$ and all edges of $P[w,t]$ are in $E(H)$. The proof relies on the following lemma.

Lemma 1: Among all pairs P,Q of disjoint paths from s to t such that P lies outside H by $i \geq 2$ edges and Q lies outside H by $j \geq 1$ edges, consider one with minimal length sum ℓ. Then there exists two disjoint paths P', Q' from s to t with length sum $\ell+1$ such that P' lies outside H by $i' \geq 1$ edges and Q' lies outside H by $j' \geq 1$ edges with $i'+j' < i+j$.

Proof: The proof consists in considering the nodes u and w on P at respective distances $i-2$ and i from s in P. The 2-connecting $(2,1)$-dominating tree rooted at w either contains a branch disjoint from $P[s,w]$ and Q that dominates u or two disjoint branches intersecting both Q. In each case, we can construct P' and Q' from P, Q and these branches. We now go more into details.

Let u,v,w be the nodes of P before the $|P|-i$ last edges of P, i.e. $P = P[s,u] + uv + vw + P[w,t]$ with $|P[s,w]| = i$. The minimal length condition implies that u and w are nonadjacent. Note that H induces a 2-connected $(2,1)$-dominating tree T for w. If $wv \in E(T)$, P and Q then satisfy the desired property. Otherwise, T must contain two disjoint paths of length at most 2 from w to two neighbors of u. Let R and S denote the two disjoint paths thus obtained from u to w. They have length at most 3 and lie outside H by one edge.

The minimality of $|P|+|Q|$ implies that the internal nodes of R and S cannot belong to $P[s,u]$ or $P[w,t]$. Suppose first that one of these paths has no internal node in Q. Assume without loss of generality that it is R. $P' = P[s,u] + R + P[w,t]$ has length at most

$|P|+1$ and lies outside H by $i-1$ edges. P' and Q thus satisfy the desired property.

Now consider the case where both R and S intersect Q. The minimality of $|P|+|Q|$ implies that each of them has at most one internal node in Q. Let x (resp. y) denote the node of R (resp. S) belonging to Q. Without loss of generality, suppose that x is closer to s than y in Q. Then set $P' = Q[s,x] + R[x,w] + P[w,t]$ and $Q' = P[s,u] + S[u,y] + Q[y,t]$. Note that $R[x,w]$ and $S[u,y]$ are disjoint and share no extremity. Moreover, their length is at most 2. P' and Q' are thus disjoint and their length sum is at most $|P|+|Q|-|P[u,w]|-|Q[x,y]|+|R[x,w]|+|S[u,y]| \leq \ell+1$. P' lies outside H by $\min\{|Q[s,x]|, j\}$ edges.

If all edges of $Q[y,t]$ are in $E(H)$, then $S[u,y]+Q[y,t]$ lies outside H by one edge and Q' lies outside H by $i-1$ edges. In that case, P' lies outside H by j edges at most. Otherwise, $Q[y,t]$ lies outside H by $j-|Q[s,y]|$ edges and Q' lies outside H by $|P[s,u]|+|S[u,y]|+j-|Q[s,y]| < i+j-|Q[s,x]|$ edges. In that case, P' lies outside H by $|Q[s,x]|$ edges. In both cases, P' and Q' satisfy the desired properties. □

of Proposition 3: Using Proposition 1 with $\varepsilon = 1$, we already know that H is a $(2,-1)$-remote-spanner. Consider two nonadjacent nodes s and t such that there exists two internally node-disjoint paths from s to t. Let $\ell = d^2_G(s,t)$ denote the minimal length sum of such a pair of paths.

By applying p times Lemma 1, we deduce that there exists two disjoint paths P and Q from s to t with length sum at most $\ell+p$ such that P lies outside H by $i \geq 1$ edges and Q lies outside H by $j \geq 1$ edges with $i+j \leq \ell-p$.

For $p = \ell-2$, we obtain two disjoint paths of length sum at most $2\ell-2$ connecting s to t in H_s. We thus deduce $d^2_{H_s}(u,v) \leq 2\ell-2$. □

3.2. k-connecting $(1,0)$-remote-spanners

We now characterize k-connecting $(1,0)$-remote-spanners as sub-graphs inducing k-connecting $(2,0)$-dominating trees. (Note that $(2,0)$-dominating trees have depth 1 and are thus stars). It is clearly a necessary condition: if H is a k-connecting $(1,0)$-remote-spanner, consider two nodes u and v such that $d_G(u,v) = 2$. If u and v have k' common neighbors with $1 \leq k' \leq k$, then the stretch condition implies that $d^{k'}_{H_v}(v,u) \leq 2k'$. As minimal path length between u and v is 2, u and v must thus have at least k' common neighbors in H_v. H must thus contain a k-connecting $(1,0)$-dominating tree for u. Indeed, we can obtain the following characterization.

Proposition 4: A sub-graph is a k-connecting $(1,0)$-remote-spanner iff it induces k-connecting $(2,0)$-dominating trees.

Consider a sub-graph H inducing k-connecting $(2,0)$-dominating trees and two nonadjacent nodes s and t such that $d^{k'}_G(s,t) < \infty$ for some k' with $1 \leq k' \leq k$. In that case, we can generalize Lemma 1 to k' paths as follows.

Lemma 2: Among all tuples $P_1, \ldots, P_{k'}$ of k' disjoint paths from s to t, consider one with minimal length sum. If P_1 lies outside H by $i \geq 2$ edges, then there exists a path P'_1 from s to t with same length as P_1 such that $P'_1, P_2, \ldots, P_{k'}$ are disjoint and P'_1 lies outside H by $i-1$ edges.

Let us first mention that Proposition 4 easily follows from this lemma. By iteratively applying Lemma 2, we obtain that there exist k' disjoint paths with minimal length sum, all of them lying 1-outside H. This implies that $d^{k'}_{H_s}(s,t) = d^{k'}_G(s,t)$ and H is thus a k-connecting $(1,0)$-remote-spanner.

of Lemma 2: Similarly to the proof of Lemma 1, let u, v, w be the nodes of P_1 before the $|P_1|-i$ last edges of P_1, i.e. $P_1 = P_1[s,u] + uv + vw + P_1[w,t]$ with $|P_1[s,w]| = i$. Note that H induces a k-connected $(2,0)$-dominating tree T for w. If $wv \in E(H)$, then $P_1, \ldots, P_{k'}$ satisfy the desired property. Otherwise, T must contain k disjoint paths of length at most 1 from w to k neighbors of u. In other words, k neighbors of u are adjacent to w in H.

We now show that each path P_j contains at most one of these k common neighbors of u and w. Suppose that by contradiction that two of them, say x and y lie on path P_j. Suppose without loss of generality that x is closer to s than y in P_j. Then we can set $P'_1 = P_j[s,x] + xw + P_1[w,t]$, $P'_j = P_1[s,u] + uy + P_j[y,t]$, and $P'_a = P_a$ for $a \notin \{1, j\}$. We then we have a contradiction since P'_1 lies outside H by i edges, $P'_1, \ldots, P'_{k'}$ are disjoint and have length sum less than $|P_1| + \cdots + |P_{k'}|$.

The minimality of the length sum of the paths and $wv \notin E(H)$ implies that P_1 contains none of the k neighbors of u adjacent to w in H. As $k' \leq k$, one of them, say x, is not in any of the paths $P_1, \ldots, P_{k'}$. Then $P_1[s,u] + ux + xw + P_1[w,t], P_2, \ldots, P_{k'}$ has the desired property. □

Interestingly, we can bound the expected number of edges of a k-connecting $(1,0)$-remote-spanner in the unit disk graph model where nodes are placed in a fixed square in the plane according to a uniform Poisson distribution (two nodes are neighbors if their distance in the plane is at most one unit). The average number of edges of an optimal k-connecting $(1,0)$-remote-spanner in such a random graph is $O(k^{\frac{2}{3}} n^{\frac{4}{3}})$ where n is the average number of nodes. In comparison, a $(1,0)$-spanner must contain all edges and has $\Omega(n^2)$ edges. This analysis is proved in [13]. For $k = 1$, this result can be deduced from the analysis of the average number of multipoint relays in [12].

3.3. Computing k-connecting remote-spanners

Algorithm 3 ($\textsc{DomTreeGdy}_{2,0,k}(u)$) computes a k-connecting $(2,0)$-dominating tree. It consists in solving greedily a set cover problem for dominating k times nodes at distance 2 from u. We use the heuristic consisting in adding iteratively in the dominating tree a node covering a maximal number of nodes at distance 2 that are still not covered by k nodes. This classical greedy heuristic in this generalization of the set-cover problem performs within a factor $1 + \log \Delta$ from optimal [11], [25] where Δ denotes the maximum degree of a node.

```
T := ({u}, ∅)
M := ∅, S := B_G(u,2) \ B_G(u,1), X := N(u)
while S ≠ ∅ do
    Pick x ∈ X \ M such that |B_G(x,1) ∩ S| is
      maximal.
    M := M ∪ {x}
    Add edge ux to T.
    S := S \ {v ∈ S | N(v) ∩ N(u) ⊆ M
                        or |N(v) ∩ M| ≥ k}
```

Algorithm 3: Algorithm $\textsc{DomTreeGdy}_{2,0,k}(u)$ for a node u. The tree T is the dominating tree computed for u.

Note that M is the set of nodes added as leaves of T. If there remains a node v in S which is initially the set of nodes at distance 2 from u, then v is not dominated k times and it has a common neighbor x with u which is not in M. It is thus always possible pick some x at the beginning of the while loop until S is empty. We can thus state the following proposition.

Proposition 5: Algorithm $\textsc{DomTreeGdy}_{2,0,k}(u)$ computes a k-connecting $(2,0)$-dominating tree for node u with minimal number of edges up to a factor $1 + \log \Delta$.

According to Proposition 4, Algorithm $\textsc{RemSpan}_{2,0}$ in conjunction with $\textsc{DomTreeGdy}_{2,0,k}$ then leads to the following result.

Theorem 2: A k-connecting $(1,0)$-remote-spanner with number of edges within a factor $2(1 + \log \Delta)$ from optimal can be computed in time $O(1)$. If the input graph is the unit disk graph of a uniform Poisson distribution in a fixed square, its average number of edges is $O(k^{2/3} n^{4/3} \log n)$.

The approximation ratio on the number of edges of the computed $(1,0)$-remote-spanner comes from the following remarks. An optimal k connecting $(1,0)$-remote-spanner H^* induces k-connecting $(2,0)$-dominating trees for each node u. As such a tree has depth 1 the degree of u in H^* is at least the size of an optimal k-connecting $(2,0)$-dominating tree T_u^* for u. We thus obtain $2|E(H^*)| \geq \sum_{u \in V(G)} |E(T_u^*)|$. As the computed dominating tree for node u with Algorithm $\textsc{DomTreeGdy}_{2,0,k}(u)$ has at most $(1+\log\Delta)|E(T_u^*)|$ edges, the remote-spanner made of the union of these trees has thus at most $2(1 + \log \Delta)|E(H^*)|$ edges. As mentioned at the end of Section 3.2, we have $|E(H^*)| = O(k^{2/3} n^{4/3})$ in expectation in the unit disk graph of a uniform Poisson distribution in a fixed square [13]. The average number of edges in the remote-spanner computed by our algorithm is thus $O(k^{2/3} n^{4/3} \log n)$.

Additionally, Algorithm 4 generalizes Algorithm $\textsc{DomTreeMIS}_{2,1}(u)$ for computing k-connecting $(2,1)$-dominating trees. It consists in dominating nodes at distance 2 from u with k maximum independent sets computed greedily.

```
T := ({u}, ∅)
S := B_G(u,2) \ B_G(u,1)
for k' := 1 to k do
    M := ∅, X := S
    while X ≠ ∅ and S ≠ ∅ do
        Pick x ∈ S ∩ X.
        M := M ∪ {x}
        k' := min{k, |(N(x) ∩ N(u)) \ V(T)|}
        Pick y_1, ..., y_k' in (N(x) ∩ N(u)) \ V(T).
        Add path uy_1 + y_1x and edges
        uy_2, ..., uy_k' to T.
        S := S \ {v ∈ S | N(v) ∩ N(u) ⊆ V(T)
          or v has k neighbors in B_T(u,2) con-
          nected to u by k disjoint paths in T}
        X := X \ B_G(x,1)
```

Algorithm 4: Algorithm $\textsc{DomTreeMIS}_{2,1,k}(u)$ for a node u. The tree T is the dominating tree computed for u.

Proposition 6: Algorithm $\textsc{DomTreeMIS}_{2,1,k}(u)$ computes a k-connecting $(2,1)$-dominating tree for node u. This tree has $O(k^2)$ edges if the input graph is the unit ball graph of a doubling metric.

Proof: The k-connected $(2,1)$-dominating tree condition on nodes v at distance 2 from u is clearly verified for nodes $x \in X \cap S$ added to T as we connect $\min\{k, |N(x) \cap N(u)|\}$ of their neighbors to u in the tree T.

At each iteration of the for loop, the set M contains the nodes added to T in that iteration. As these nodes are picked in X, the last instruction of the while loop implies that M is a maximum independent set of $M \cup S$ at the end of the iteration. At end of iteration k', the nodes remaining in S are thus dominated by k' nodes in $V(T)$: one in each computed MIS. Note additionally, that for each node x added to M, we add at least one path $uy_1 + y_1x$ disjoint from all previous paths added to T (otherwise x would have been removed from S previously). Each node remaining in S is thus dominated by k' nodes in T connected to u in T by disjoint paths of length 2. At the end of the last iteration, we thus have $S = \emptyset$ and T is a k-connecting $(2, 1)$-dominating tree.

If the input graph is the unit ball graph of a doubling metric, each computed MIS set has size $O(1)$. We thus add $O(k)$ edges to T in each iteration and $O(k^2)$ in total. □

According to Proposition 3, Algorithm $\text{REMSPAN}_{2,0}$ in conjunction with $\text{DOMTREEMIS}_{2,1,2}$ then leads to the following result.

Theorem 3: A 2-connecting $(2, -1)$-remote-spanner can be computed in time $O(1)$. Its number of edges is $O(n)$ if the input graph is the unit ball graph of a doubling metric.

4. Concluding remarks

We have introduced the notion of remote-spanner which is well suited for grasping the trade-offs when optimizing the subset of links advertised in a link state routing protocol. Most strikingly, we have proposed distributed construction of sparse remote-spanners providing at the same time multi-connectivity and controlled stretch for any input graph. Their size is optimal up to a poly-logarithmic factor for $(1, 0)$-remote-spanners, and linear if the input graph is the unit ball graph of a doubling metric. An interesting followup resides in constructing sparse k-connecting $(1 + \varepsilon, O(1))$-remote-spanners for any $\varepsilon > 0$ and $k > 1$. Additionally, it seems possible to extend our results to edge-connectivity where we consider paths that are edge-disjoint rather than internal-node disjoint.

References

[1] Mohammad Ali Abam, Mark de Berg, M. Farshi, and Joachim Gudmundsson. Region-fault tolerant geometric spanners. In 18^{th} *Symp. on Discrete Algorithms (SODA)*, pages 1–10. ACM-SIAM, January 2007.

[2] Surender Baswana, Telikepalli Kavitha, Kurt Mehlhorn, and Seth Pettie. New constructions of (α, β)-spanners and purely additive spanners. In 16^{th} *Symp. on Discrete Algorithms (SODA)*, pages 672–681. ACM-SIAM, January 2005.

[3] T. Clausen, P. Jacquet (editors), C. Adjih, A. Laouit i, P. Minet, P. Muhlethaler, A. Qayyum, and L.Viennot. Optimized link state routing protocol (OLSR). RFC 3626, October 2003. Network Working Group.

[4] T. Clausen, P. Jacquet, and L. Viennot. Investigating the impact of partial topology in proactive manet routing protocols. In *The 5th Int. Symp. on Wireless Personal Multimedia Communications (WPMC)*, october 2002.

[5] Artur Czumaj, Michelangelo Grigni, Papa Sissokho, and Hairong Zhao. Approximation schemes for minimum 2-edge-connected and biconnected subgraphs in planar graphs. In *SODA '04: Proc. of the fifteenth annual ACM-SIAM Symposium on Discrete Algorithms*, pages 496–505, Philadelphia, PA, USA, 2004. Society for Industrial and Applied Mathematics.

[6] Artur Czumaj and Andrzej Lingas. Fast approximation schemes for euclidean multi-connectivity problems. In *Proc. of the 27th Int. Colloquium on Automata, Languages and Programming (ICALP)*, pages 856–868. Springer-Verlag, 2000.

[7] Artur Czumaj and Hairong Zhao. Fault-tolerant geometric spanners. *Discrete Comput. Geom.*, 32(2):207–230, 2004.

[8] Mirela Damian, Saurav Pandit, and Sriram Pemmaraju. Distributed spanner construction in doubling metric spaces. In *Principles of Distributed Systems*, volume 4305 of *Lecture Notes in Computer Science*, pages 157–171. Springer, 2006.

[9] Mirela Damian, Saurav Pandit, and Sriram Pemmaraju. Local approximation schemes for topology control. In *PODC '06: Proc. of the twenty-fifth annual ACM Symposium on Principles of Distributed Computing*, pages 208–217. ACM, 2006.

[10] B. Derbel, C. Gavoille, D. Peleg, and L. Viennot. On the locality of distributed sparse spanner construction. In ACM Press, editor, *27th Annual ACM Symp. on Principles of Distributed Computing (PODC)*, pages 273–282, 2008.

[11] Gregory Dobson. Worst-case analysis of greedy heuristics for integer programming with nonnegative data. *Mathematics of Operations Research*, 7(4):515–531, 1982.

[12] P. Jacquet, A. Laouiti, P. Minet, and L. Viennot. Performance analysis of OLSR multipoint relay flooding in two ad hoc wireless network models. In *The second IFIP-TC6 NETWORKING Conference*, may 2002.

[13] P. Jacquet and L. Viennot. Average size of unstretched remote-spanners. In *5th ACM Workshop on Analytic Algorithmics and Combinatorics (ANALCO)*, pages 23–33, 2009.

[14] Philippe Jacquet, Pascale Minet, Paul Muhlethaler, and Ni colas Rivierre. Increasing reliability in cable-free radio LANs: Low level forwarding in HIPERLAN. *Wireless Personal Communications*, 4(1):65–80, January 1997.

[15] Fabian Kuhn, Thomas Moscibroda, Tim Nieberg, and Roger Wattenhofer. Fast deterministic distributed maximal independent set computation on growth-bounded graphs. In *DISC*, pages 273–287, 2005.

[16] Fabian Kuhn, Thomas Moscibroda, and Roger Wattenhofer. On the locality of bounded growth. In *PODC '05: Proc. of the twenty-fourth annual ACM Symposium on Principles of Distributed Computing*, pages 60–68. ACM, 2005.

[17] Christos Levcopoulos, Giri Narasimhan, and Michiel H. M. Smid. Improved algorithms for constructing fault-tolerant spanners. *Algorithmica*, 32(1):144–156, 2002.

[18] Tamas Lukovszki. New results on fault tolerant geometric spanners. In *Proc. of the 6th Workshop on Algorithms and Data Structures*, volume 1663 of *Lecture Notes in Computer Science*, pages 193–204. Springer-Verlag,

1999.
[19] J. McQuillan, I. Richer, and E. Rosen. The new routing algorithm for the arpanet. *IEEE Transactions on Communications*, 28(5):711–719, 1980.
[20] John Moy. Open shortest path first (OSPF) specification. RFC 1131, October 1989. Network Working Group.
[21] John Moy. OSPF version 2. RFC 2328, April 1998. Network Working Group.
[22] D. Peleg and A. A. Schaffer. Graph spanners. *Journal of graph theory*, 13(1):99–116, 1989.
[23] David Peleg and Jeffrey D. Ullman. An optimal synchornizer for the hypercube. *SIAM Journal on Computing*, 18(4):740–747, 1989.
[24] Seth Pettie. Low distortion spanners. In 34^{th} *Int. Colloquium on Automata, Languages and Programming (ICALP)*, pages 78–89, July 2007.
[25] L. A. Wolsey. An analysis of the greedy algorithm for the submodular set covering problem. *Combinatorica*, 2(4):385–393, 1982.
[26] David P. Woodruff. Lower bounds for additive spanners, emulators, and more. In 47^{th} *Annual IEEE Symp. on Foundations of Computer Science (FOCS)*, pages 389–398, October 2006.
[27] Jie Wu, Wei Lou, and Fei Dai. Extended multipoint relays to determine connected dominating sets in manets. *IEEE Trans. Comput.*, 55(3):334–347, 2006.