Journal of Algorithms **30,** 166–184 (1999)
Article ID jagm.1998.0962, available online at http://www.idealibrary.com on IDEAL®

Distance Approximating Trees for Chordal and Dually Chordal Graphs*

Andreas Brandstädt

Universität Rostock, Fachbereich Informatik, Lehrstuhl für Theoretische Informatik, D-18051 Rostock, Germany
E-mail: ab@informatik.uni-rostock.de

Victor Chepoi*

Laboratoire de Biomathématiques, Université d'Aix Marseille II, F-13385 Marseille Cedex 5, France
E-mail: aria@pacwan.mm-soft.fr

and

Feodor Dragan

Universität Rostock, Fachbereich Informatik, Lehrstuhl für Theoretische Informatik, D-18051 Rostock, Germany
E-mail: dragan@informatik.uni.rostock.de

Received April 23, 1997; revised March 10, 1998

In this paper we show that, for each chordal graph G, there is a tree T such that T is a spanning tree of the square G^2 of G and, for every two vertices, the distance between them in T is not larger than the distance in G plus 2. Moreover, we prove that, if G is a strongly chordal graph or even a dually chordal graph, then there exists a spanning tree T of G that is an additive 3-spanner as well as a multiplicative 4-spanner of G. In all cases the tree T can be computed in linear time.

1. INTRODUCTION

Many combinatorial and algorithmic problems concern the distance d_G on the vertices of a possibly weighted graph $G = (V, E)$. Approximating

* The second and third authors were supported by VW, Project no. I/69041; the third author was also supported by DFG. The results were presented at ESA'97 (Graz [6]). The second and third authors are on leave from the Universitatea de stat din Moldova, Chişinău.

0196-6774/99 $30.00

d_G by a simpler distance (in particular, by tree distance) is useful in many areas such as communication networks, data analysis, motion planning, image processing, network design, and phylogenetic analysis (see [1, 2, 4, 10, 12, 22, 27, 28, 30, 32]). The goal is, for a given graph G, to find a sparse graph $G = (V, E')$ with the same vertex set, such that the distance $d_H(u,v)$ in H between two vertices $u, v \in V$ is reasonably close to the corresponding distance $d_G(u,v)$ in the original graph G. There are several ways to measure the quality of this approximation, two of them leading to the notion of a spanner. For $t \geq 1$ a spanning subgraph H of G is called a *multiplicative t-spanner* of G [12, 27, 28] if $d_H(u,v) \leq t \cdot d_G(u,v)$ for all $u, v \in V$. If $r \geq 0$ and $d_H(u,v) \leq d_G(u,v) + r$ for all $u, v \in V$, then H is called an *additive r-spanner* [22].

For many applications (e.g., in numerical taxonomy or in phylogeny reconstruction) the condition that H must be a spanning subgraph of G can be dropped (see [2, 30, 32]). In this case there is a striking way to measure how sharp d_H approximates d_G, based on the notion of a pseudoisometry between two metric spaces. This idea is borrowed from the geometry of hyperbolic groups [15, 18]. For graphs and finite metric spaces a related notion of a near-isometry has been already used by Linial et al. [23]. For our purposes we present a simplified version of this notion (the interested reader can consult [15, pp. 71–72] and [18] for the general definition and related material).

Let $t \geq 1$ and $r \geq 0$ be real numbers. Two graphs $G = (V, E)$ and $H = (V, E')$ are called *(t, r)-pseudoisometric* if

$$d_H(u,v) \leq t \cdot d_G(u,v) + r \quad \text{and} \quad d_G(u,v) \leq t \cdot d_H(u,v) + r$$

for all $u, v \in V$. In this case we will say that H is a *distance (t, r)-approximating graph* for G (and conversely, G will be a distance (t, r)-approximating graph for H). The graphs G and H are $(t, 0)$-pseudoisometric iff

$$\frac{1}{t} \cdot d_G(u,v) \leq d_H(u,v) \leq t \cdot d_G(u,v)$$

for $u, v \in V$. If, in addition, H is a spanning subgraph of G, then we obtain the notion of the multiplicative t-spanner. Clearly, G and H are $(1, r)$-pseudoisometric iff $|d_G(u,v) - d_H(u,v)| \leq r$ for $u, v \in V$. Again, if H is a spanning subgraph of G, this is the usual notion of the additive r-spanner.

Recently Cai and Corneil [10] have considered multiplicative tree spanners in graphs. They showed that for a given graph G, the problem of deciding whether G has a multiplicative tree t-spanner is NP-complete for any fixed $t \geq 4$ and is linearly solvable for $t = 1, 2$. The status of the case

$t = 3$ is still open. For *NP*-completeness results on planar spanners see [5]. Multiplicative tree 3-spanners exist for interval and permutation graphs, and they can be found in linear time [24]. Similar results are known for the additive tree r-spanner problem. Prisner [29] proposes a simple approach to constructing additive tree 2-spanners in interval and distance-hereditary graphs, and such 4-spanners in cocomparability graphs. Both papers [10, 29] ask which important graph classes have tree t-spanners and r-spanners with small t and r. As mentioned in [29], McKee showed that for every fixed integer t there is a chordal graph without tree t-spanners (additive as well as multiplicative). Nevertheless, from the metric point of view, chordal graphs look like trees. In this paper we prove that for every chordal graph G there exists a tree T (actually, T is a spanning tree of the square G^2) such that

$$d_T(u,v) \leq 3 \cdot d_G(u,v) \quad \text{and} \quad d_T(u,v) \leq d_G(u,v) + 2$$

for all vertices u, v of G. In other words, T is a $(3,0)$- and $(1,2)$-approximating tree for G. Moreover, if G is a strongly chordal graph, then there exists a spanning tree T of G that is an additive 3-spanner and a multiplicative 4-spanner. Thus, this answers the question whether strongly chordal graphs have tree t-spanners with small t, posed in [29]. Furthermore, we show that the method elaborated for strongly chordal graphs works for a more general graph class, for the dually chordal graphs. In all cases the tree T can be computed in linear time.

2. PRELIMINARIES

All graphs occurring in this paper are connected, finite, undirected, loopless, and without multiple edges. A graph $G = (V, E)$ is *chordal* [9, 13] if it does not contain any induced (chordless) cycle of length at least four. In a graph G the *length* of a path from a vertex v to a vertex u is the number of edges in the path. The *distance* $d_G(u, v)$ between the vertices u and v is the length of a shortest (u, v)-path, and the *interval* $I(u, v)$ between these vertices is the set $I(u, v) = \{w \in V\colon d_G(u, v) = d_G(u, w) + d_G(w, v)\}$. Let k be a positive integer. The kth *power* G^k of G has the same vertices as G, and two vertices are joined by an edge in G^k if and only if the distance in G is at most k. The *disk* of radius k centered at v is the set of all vertices at distance at most k from v, $D_k(v) = \{w \in V\colon d_G(v,w) \leq k\}$, and the kth *neighborhood* $N_k(v)$ of v is defined as the set of all vertices at distance k from v, that is, $N_k(v) = \{w \in V\colon d_G(v, w) = k\}$. By $N(v)$ we denote the *neighborhood* of v, i.e., $N(v) := N_1(v)$. More generally, for a subset $S \subseteq V$ let $N(S) = \bigcup_{v \in S} N(v)$ denote the *neighborhood* of S.

A subset $S \subseteq V$ of a graph G is called *m-convex* [17] if for any pair of vertices $u, v \in S$ each induced path connecting u and v is contained in S. If two vertices x and y of a m-convex set S can be joined outside S by a path, then by definition of m-convexity, x and y must be adjacent.

PROPOSITION 1 [17]. *Any disk $D_k(v)$ of a chordal graph G is m-convex.*

This basic metric property of chordal graphs has some immediate but important consequences (some of them already being used by different authors). In the subsequent results the graph G is assumed to be chordal and u is an arbitrary but fixed vertex of G.

LEMMA 1. *For every vertex v of G and every k, $0 < k < d_G(u, v)$, the set $N_k(u) \cap I(u, v)$ induces a complete subgraph of G.*

Proof. Pick vertices $x, y \in N_k(u) \cap I(u, v)$. Concatenating arbitrary shortest paths that connect v with x and y, we will obtain a path outside the disk $D_k(u)$. By the m-convexity of $D_k(u)$ we conclude that x and y are adjacent. ∎

LEMMA 2. *If two vertices $v, w \in N_k(u)$ are adjacent, then they have a common neighbor in the set $N_{k-1}(u)$.*

Proof. Indeed, take $x, y \in N_{k-1}(u)$ such that v is adjacent to x and w is adjacent to y. By m-convexity x and y are adjacent or coincide. If $x = y$ we are done. Otherwise, from the chordality of G we will deduce that either $vy \in E$ or $wx \in E$, concluding the proof. ∎

LEMMA 3. *For any connected component S of the subgraph of G induced by $N_k(u)$, the set $N(S) \cap N_{k-1}(u)$ induces a complete subgraph. Moreover, there exists a vertex $w \in S$ such that $N(S) \cap N_{k-1}(u) = N(w) \cap N_{k-1}(u)$ (Fig. 1).*

Proof. Let $x, y \in N(S) \cap N_{k-1}(u)$. Then we can find two vertices $v, w \in S$ such that x is adjacent to w and y is adjacent to v. Consider a path P of S connecting the vertices v and w. Then P together with the edges xw and yv will give a path that intersects the disk $D_{k-1}(u)$ only in the vertices x and y. Since $D_{k-1}(u)$ is m-convex, we deduce that the vertices x and y are adjacent.

Now consider a vertex $w \in S$ with a maximal number of neighbors in $N_{k-1}(u)$ and assume that $wy \notin E$ for some $y \in N(S) \cap N_{k-1}(u)$. Among neighbors of y in S we choose a vertex v for which the distance $d_S(w, v)$ is minimal. Let P be a shortest path in S connecting the vertices w and v. For every neighbor x of w in $N_{k-1}(u)$ consider the cycle formed by path P and the edges wx, xy, and yv. Since G is chordal and $d_S(w, v)$ is minimal, the vertices x and v must be adjacent. So, every neighbor of w in $N_{k-1}(u)$ is a neighbor of v as well. But then from $wy \notin E$ and $vy \in E$ we get a contradiction to the choice of the vertex w. ∎

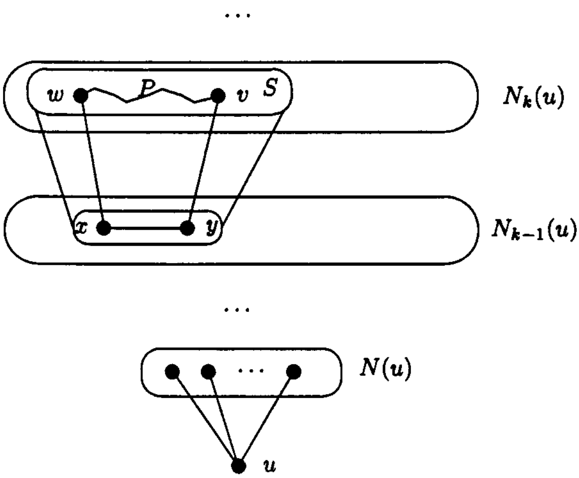

FIG. 1. Illustration of Lemma 3.

Let $q = \max\{d_G(u, v) : v \in V\}$. For a given k, $0 \le k \le q$, let $S_1^k, \dots, S_{p_k}^k$ be the connected components of the subgraph of G induced by the kth neighborhood of u. Define a new graph Γ whose vertices are the connected components S_i^k, $k = 0, \dots, q$ and $i = 1, \dots, p_k$. Two vertices S_i^k and S_j^{k-1} are adjacent if and only if there is an edge of G with one end in S_i^k and another end in S_j^{k-1}. Lemma 3 implies that every S_i^k, $k > 0$, is adjacent in Γ to exactly one connected component of $N_{k-1}(u)$. This shows that the following holds.

LEMMA 4. *Γ is a tree.*

In what follows we consider Γ rooted at the vertex $u = S_1^0$. As usual, the *nearest common ancestor* $nca(S_i^k, S_j^l)$ of two vertices S_i^k and S_j^l of Γ is the root of the smallest subtree of Γ that contains both of the vertices S_i^k and S_j^l.

3. DISTANCE APPROXIMATING TREES FOR CHORDAL GRAPHS

In this section for a given chordal graph $G = (V, E)$ we construct a tree $T = (V, E')$ that is a distance (3, 0)- and (1, 2)-approximating tree for G. As in the previous section, we fix an arbitrary vertex u of G. To construct T, for each connected component S_j^k $(k \ge 1)$ we select a vertex $v \in N(S_j^k) \cap N_{k-1}(u)$ and make v adjacent in T to all vertices of S_j^k (a formal description is given below).

PROCEDURE 1.

$E' := \emptyset$;
for $k := q$ **downto** 1 **do**
  for $j := 1$ to p_k **do**
    pick an arbitrary vertex v in $N(S_j^k) \cap N_{k-1}(u)$;
    for all $x \in S_j^k$ add xv to E'.

By Lemma 4 it follows easily that T is a tree. For any edge xv of T that is not an edge of G we have $d_G(x, v) = 2$. Indeed, by Lemma 3 every neighbor of x in $N_{k-1}(u)$ is adjacent to v. Therefore, T is a spanning tree of G^2. Since for constructing T we need only find the connected components of the kth neighborhoods of u, the complexity of this procedure is $O(|V| + |E|)$.

THEOREM 1. *T is a distance $(3, 0)$- and $(1, 2)$-approximating tree of G.*

Proof. First we will show that for any edge xy of G we have $d_T(x, y) \leq 3$. If $d_G(u, x) = d_G(u, y) = k$, then x and y belong to a common connected component of $N_k(u)$. Therefore, in T both x and y are adjacent to the same vertex v. In this case $d_T(x, y) = 2$. Now suppose that $d_G(u, x) - 1 = d_G(u, y) = k$ and xy is not an edge of T. Let v be the neighbor of x on the path connecting u and x in the tree T. By Lemma 3 the vertices v and y are adjacent. Since $d_G(u, v) = k$, from the previous case we obtain that $d_T(v, y) = 2$. Hence $d_T(x, y) = 3$. On the other hand, as we mentioned above, for every edge xy of T, $d_G(x, y) \leq 2$.

Now consider two arbitrary vertices x and y of G and a shortest (x, y) path. Applying to every edge of this path the obtained inequalities, we will get

$$\tfrac{1}{2} \cdot d_G(x, y) \leq d_T(x, y) \leq 3 \cdot d_G(x, y).$$

To prove that $|d_G(x, y) - d_T(x, y)| \leq 2$, we proceed as follows. Suppose that $x \in S_i^k$ and $y \in S_j^l$. Let $S := nca(S_i^k, S_j^l)$ and assume that S is a connected component of the sth neighborhood of u. Denote by S' and S'' the neighbors of S in Γ on the paths between S, S_i^k, and S, S_j^l, respectively. From the definition of the trees Γ and T we obtain that in T and G the distances between u and any other vertex of V are the same. Any shortest path in G between u and each of the vertices x and y shares a common vertex with S. Therefore, one can select two vertices $x', y' \in S$ such that $d_G(x, x') = k - s$ and $d_G(y, y') = l - s$ (note that x' and y' are vertices of S closest to x and y, respectively). Since Γ is a tree, one can easily show that in G any shortest (x, y) path $P(x, y)$ between x and y passes through the set S. Let x'', y'' denote the first and last vertices of $P(x, y)$ in S.

Since $x', x'' \in N(S')$ and $y', y'' \in N(S'')$, by Lemma 3, $d_G(x, x'') = k - s$ and $d_G(y, y'') = l - s$. Therefore, $d_G(x, y) = k + l - 2 \cdot s + d_G(x'', y'')$. From Lemma 3 we have $d_G(x'', y'') \leq 3$. On the other hand, from the algorithm we obtain that $d_T(x, y) = k + l - 2 \cdot s + \alpha$, where $\alpha = 0$ if in T the nearest common ancestor of x and y is a vertex of S, and $\alpha = 2$ if this ancestor belongs to the father of S in Γ. In the first case necessarily $N(S')$ and $N(S'')$ share a vertex in S. Since by Lemma 3 $N(S') \cap S$ and $N(S'') \cap S$ are complete subgraphs, one can easily show that $d_G(x'', y'') \leq 2$. Therefore, in this case $|d_G(x, y) - d_T(x, y)| \leq 2$. The same inequality is evidently true if $\alpha = 2$. This completes the proof of the theorem. ∎

It is an open problem whether the distance matrix D of a chordal graph $G = (V, E)$ can be computed in $O(|V|^2)$ time. From the second assertion of Theorem 1 we obtain that within these time bounds we can compute the elements of D with an error of at most 2. Even more, given a pair of vertices $x, y \in V$ by the algorithm of Harel and Tarjan [20], the nearest common ancestor $nca(x, y)$ of x and y in T can be computed in $O(1)$ time after a linear time preprocessing of T. Since $d_T(x, y) = d_T(x, u) + d_T(y, u) - 2 \cdot d_T(u, nca(x, y))$, the distance $d_T(x, y)$ can be found by using a constant number of operations. Therefore, after a linear time preprocessing, in only $O(1)$ time we can compute $d_G(x, y)$ with an error of at most 2.

A chordal graph together with some distance $(1, 2)$-approximating tree produced by Procedure 1 is given in Fig. 2. Note that this chordal graph has no additive tree r-spanner for $r \leq 3$ (see [29]). It remains an open question whether every chordal graph admits an $(1, 1)$-approximating tree.

4. SPANNERS OF SOME CLASSES OF CHORDAL GRAPHS

As we already mentioned in the Introduction, chordal graphs do not have multiplicative or additive t-spanners with a fixed t. However, we will show that chordal graphs, which do not contain (extended) suns as induced subgraphs, have multiplicative 4-spanners and additive 3-spanners. We present an $O(|V| + |E|)$ time algorithm for computing such spanners. This class of graphs comprises the well-known strongly chordal graphs.

A *p-sun* S_p ($p \geq 3$) is a chordal graph on $2p$ vertices whose vertex set can be partitioned into two sets, $U = \{u_1, \dots, u_p\}$ and $W = \{w_1, \dots, w_p\}$, such that W is independent and every w_i is adjacent only to u_i and u_{i+1} (mod p). An extended sun S_p^* is a sun S_p with two additional vertices x and y such that x is adjacent to w_2 and not to w_i, u_j, with $i \neq 2$, $j \neq 3$, and y is adjacent to w_p and not to w_i, u_i, with $i \neq p$. Note that x may be

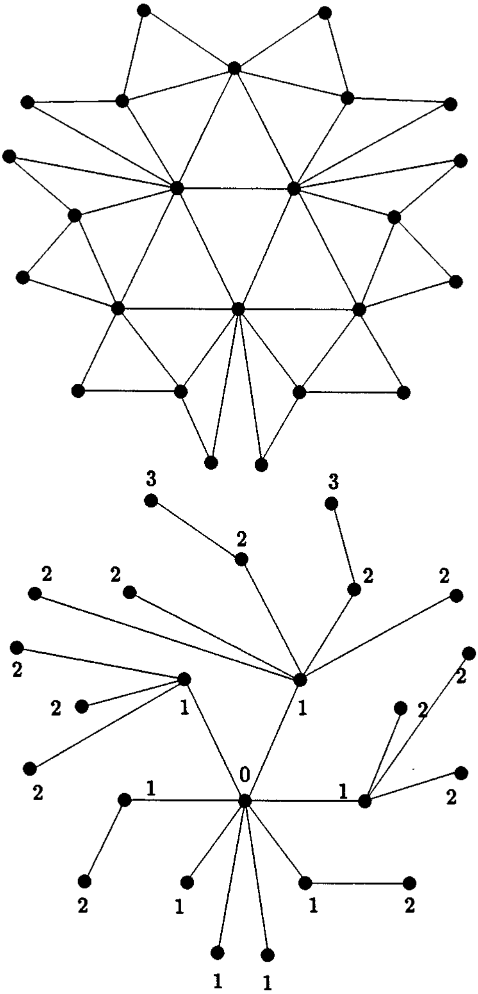

FIG. 2. A chordal graph and a distance $(1, 2)$-approximating tree of it.

adjacent to u_3 and y may be adjacent to u_p. Suns S_3 and S_4 and extensions of S_4 are presented in Fig. 3.

A chordal graph G is called *strongly chordal* if it does not contain any sun S_p as an induced subgraph [11, 16]. For equivalent definitions and properties of strongly chordal graphs see [16], and for the recognition problem see [26, 31].

Pick an arbitrary vertex u of G, and let $S_1^{k+1}, \dots, S_{p_{k+1}}^{k+1}$ be the connected components of the subgraph of G induced by $N_{k+1}(u)$, $k \geq 1$. Let

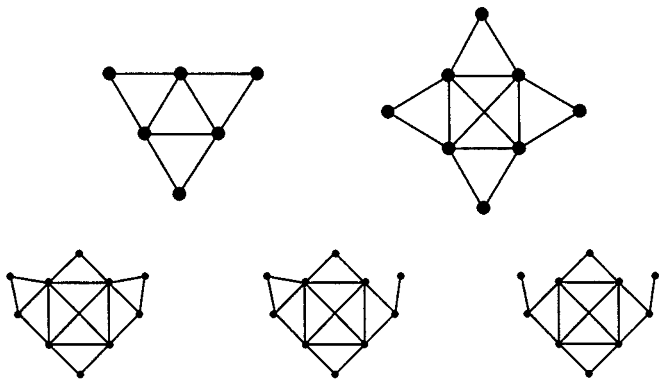

FIG. 3. Suns S_3 and S_4 and extensions of S_4.

$C_j = N(S_j^{k+1}) \cap N_k(u)$ $(j = 1, \dots, p_{k+1})$ (because of Lemma 3, C_j is a clique) and denote by H_k the graph with $\bigcup_{j=1}^{p_{k+1}} C_j$ as the vertex set, and two vertices x, y are adjacent in H_k if and only if they belong to a common clique C_j.

By a *two-set* of a graph G we will mean a subset $M \subseteq V$ such that $d_G(x, y) \le 2$ for any $x, y \in M$.

LEMMA 5. *Let G be a chordal graph that does not contain S_3 as an induced subgraph. For any two-set $M \subseteq N_k(u)$ of G there exists a vertex $v \in N_{k-1}(u)$ that is adjacent to all vertices of M.*

Proof. By induction on the cardinality $q = |M|$ of M. First assume that $q = 2$ and $M = \{x, y\}$. If no vertex from $N_{k-1}(u)$ is adjacent to both x and y, then by Lemmas 2 and 3 we will have $d_G(x, y) = 2$ and $N(x) \cap N(y) \subseteq N_k(u)$. Let z be an arbitrary vertex from $N(x) \cap N(y)$. Since $x, y, z \in N_k(u)$, by Lemma 2, there exist two vertices a and b in $N_{k-1}(u)$ such that a is adjacent to x, z and b is adjacent to y, z. From Lemmas 2 and 3 we deduce that the vertices a and b must be adjacent, and hence there is a common neighbor $w \in N_{k-2}(u)$ of a and b. Thus, we have constructed a 3-sun, induced by the vertices x, y, w, z, a, b, contrary to our assumption. Hence, x and y must have a common neighbor in $N_{k-1}(u)$.

Assume now that every $q - 1$ $(q \ge 3)$ vertices of $M = \{x_1, x_2, \dots, x_q\}$ have a common neighbor in $N_{k-1}(u)$ but not all q vertices have such a neighbor. Let $y_p = \bigcap_{i=1, i \ne p}^{q} N(x_i) \cap N_{k-1}(u)$, $p = 1, 2, 3$. Since x_p is adjacent to y_{p+1} and y_{p-1} (mod 3) by Lemma 3, the vertices y_1, y_2, and y_3 are pairwise adjacent. Moreover, the vertices x_1, x_2, and x_3 induce an independent set; otherwise we will obtain an induced 4-cycle. Hence, again we have constructed an induced 3-sun formed by $x_1, x_2, x_3, y_1, y_2, y_3$. ∎

LEMMA 6. *Let G be a chordal graph that does not contain extended suns S_p^* $(p \geq 4)$ as induced subgraphs. Then every connected component of the graph H_k $(k \geq 1)$ is a two-set of G.*

Proof. Let x, y be two vertices of a connected component F of the graph H_k such that $d_G(x, y) \geq 3$. Pick a shortest path $P = (x_0 = x, x_1, \ldots, x_{l-1}, x_l = y)$ in F connecting x and y. By Lemma 3 and since P is an induced path of F, we will find l pairwise nonadjacent vertices $z_1, \ldots, z_l$ in $N_{k+1}(u)$ such that every z_i $(i = 1, \ldots, l)$ is adjacent to x_{i-1}, x_i only. Let a and b be neighbors in $N_{k-1}(u)$ of x and y, respectively. Note that $a \notin N(y)$ and $b \notin N(x)$. From Lemma 3 we obtain that a and b must be adjacent. Furthermore, by Lemma 2 there is a common neighbor $w \in N_{k-2}(u)$ of a and b. Now we construct an extended sun S_p^* with $p \geq 4$ in the following way. Pick the smallest i $(i = 1, \ldots, l-1)$ such that x_i is adjacent to y. Then pick the largest j $(j = 1, \ldots, i)$ such that x_j is adjacent to x. Since $d_G(x, y) \geq 3$, we have $i \geq 2$ and $1 \leq j < i$. It is easy to see that the vertices $w, x, z_{j+1}, \ldots, z_i, y$ together with the vertices $a, x_j, x_{j+1}, \ldots, x_i, b$ induce a sun in G. Adding to this sun the vertices z_1 and z_l, we obtain an extended sun. ∎

Now we have all of the prerequisites needed to formulate the main result of this section. Let G be a chordal graph that does not contain a sun S_3 and extended suns S_p^* $(p \geq 4)$ as induced subgraphs. Consider an arbitrary fixed vertex u and a tree Γ rooted at u, which is defined as in Section 2. Let $q := \max\{d(u, v) : v \in V\}$ and $S_1^k, \ldots, S_{p_k}^k$ be the connected components of the kth neighborhood $N_k(u)$ of u. We construct a distance-approximating spanning tree $T = (V, E')$ of G step by step, starting from the leaves of Γ, i.e., from $N_q(u)$. Initially E' is empty. At first, for each vertex $v \in N_q(u)$, we add to E' one edge of the form vw, where w is a neighbor of v in $N_{q-1}(u)$. Now consider an arbitrary k that runs from $q-1$ to 1 and the cliques $C_j = N(S_j^{k+1}) \cap N_k(u)$ $(j = 1, \ldots, p_{k+1})$. By Lemma 6, every connected component F of the graph H_k is a two-set. Therefore all vertices of F have a common neighbor v_F in $N_{k-1}(u)$ (see Lemma 5). Now if x is a vertex of H_k, say $x \in F$ for some connected component F of H_k, then we add to the current E' the edge xv_F. For every vertex y of $N_k(u)$ that does not belong to H_k, we add to E' an edge connecting y with an arbitrary neighbor of it in $N_{k-1}(u)$.

PROCEDURE 2.

$E' := \emptyset$;

for every $v \in N_q(u)$ pick a neighbor w in $N_{q-1}(u)$ and
add the edge vw to E';

for $k := q-1$ **downto** 1 **do**

compute the connected components $S_1^{k+1}, \ldots, S_{p_{k+1}}^{k+1}$ of $N_{k+1}(u)$;

determine the graph H_k and compute its connected components;

for every $y \in N_k(u) \setminus H_k$ pick a neighbor w in $N_{k-1}(u)$ and add the edge yw to E';

for every connected component F of the graph H_k **do**

choose in $N_{k-1}(u)$ a common neighbor v_F of all vertices of F;

for every $x \in F$ add the edge xv_F to E'.

One can easily show that the graph $T = (V, E')$ constructed by this procedure is a spanning tree of G. Next we will show that the procedure can be implemented in linear time. In the preprocessing step we apply the breadth-first search to compute the kth neighborhoods of the vertex u in $O(|V| + |E|)$ time. Denote by $deg(v)$ the degree of a vertex v in G, i.e., $deg(v) = |N(v)|$. The second line of the procedure requires at most $\sum_{v \in N_q(u)} deg(v)$ operations. We spend $O(|N_{k+1}(u)| + \sum_{v \in N_{k+1}(u)} deg(v))$ time for computing the connected components in $N_{k+1}(u)$.

To determine the graph H_k we need to find the cliques $C_j = N(S_j^{k+1}) \cap N_k(u)$, $(j = 1, \ldots, p_{k+1})$. To do this we proceed in the following way. By Lemma 3 a vertex w_j of the connected component S_j^{k+1}, which has the maximum number of neighbors in $N_k(u)$, obeys the condition $N(w_j) \cap N_k(u) = N(S_j^{k+1}) \cap N_k(u)$. Therefore, we have to find the vertices $\{w_1, \ldots, w_{p_{k+1}}\}$ and put $C_j := N(w_j) \cap N_k(u)$. This can be done in $\sum_{v \in N_{k+1}(u)} deg(v)$ time. Having the cliques $C_1, \ldots, C_{p_{k+1}}$, the connected components of the graph H_k can be computed by constructing a special bipartite graph $B_k = (W, K; U)$. In this graph $W = \{w_1, \ldots, w_{p_{k+1}}\}$, $K = \bigcup_{i=1}^{p_{k+1}} C_i$, and a vertex of W and a vertex of K are adjacent in B_k if and only if they are adjacent in G. This graph B_k can be constructed in $O(\sum_{j=1,\ldots,p_{k+1}} |C_j|) = O(\sum_{v \in N_{k+1}(u)} deg(v))$. One can easily see that the connected components of H_k are exactly the intersections of the connected components of B_k with the set K, and thus can be found within the same time bounds.

Finally, to decide which connected components of the graph H_k are contained in the neighborhood $N(v)$ of some vertex $v \in N_{k-1}(u)$, we do the following. In total $O(deg(v))$ time, for each connected component F_i $(i = 1, \ldots, r)$ with $N(v) \cap F_i \neq \emptyset$, we compute the value n_i, which is the number of vertices of F_i from $N(v)$. If $n_i = |F_i|$, then we put $v_{F_i} := v$. Thus, this line of the procedure can be implemented in $O(\sum_{v \in N_{k-1}(u)} deg(v))$ time, too.

Summarizing, the whole procedure requires only

$$O\left(\sum_{k=1}^{q}\left(\sum_{v\in N_k(u)} deg(v) + |N_k(u)|\right)\right) = O(|V| + |E|)$$

time.

A strongly chordal graph together with some additive tree 3-spanner produced by our procedure is given in Fig. 4. Note that this strongly chordal graph has no additive tree 2-spanner [21]. The following theorem shows that not only this graph but every strongly chordal graph has an additive tree 3-spanner. So this result is the best possible.

THEOREM 2. *Let G be a chordal graph that does not contain a sun S_3 and extended suns S_p^* $(p \geq 4)$ as induced subgraphs. The tree $T = (V, E')$ constructed by Procedure* 2 *is a multiplicative* 4-*spanner as well as an additive* 3-*spanner of G.*

Proof. First we will show that $d_T(x, y) \leq 4$ holds for any edge xy of G. Suppose that $d_G(u, x) = d_G(u, y) = k$, say $x, y \in S_j^k$. If x and y belong to a common connected component F of the graph H_k, then in T they have a common father v_F, and thus $d_T(x, y) = 2$. Otherwise, let x' and y' be the fathers of x and y, respectively, in T. Since $x', y' \in C_j$, in H_{k-1} the vertices x' and y' lie in a common connected component. As we have already shown $d_T(x', y') = 2$. Therefore, in this case $d_T(x, y) = 4$.

Now suppose that $x \in N_k(u)$ and $y \in N_{k-1}(u)$. Let x' be the father of x in T. If $x' = y$ we are done. Assume $x' \neq y$. Both x' and y have some neighbor in the same component of $N_k(u)$, and thus they are adjacent in H_{k-1}. According to the algorithm, for an edge of this type, $d_T(x', y) = 2$ holds. Hence $d_T(x, y) = 3$.

Now consider two arbitrary vertices v and w of G and a shortest (v, w) path. Applying to every edge xy of this path the inequality $d_T(x, y) \leq 4$, we will get $d_T(v, w) \leq 4 \cdot d_G(v, w)$, i.e., T is a multiplicative 4-spanner for G.

That T is an additive 3-spanner of G follows already from the previous part of our proof and from [29, Lemma 1]. For the sake of completeness we present here our proof. For this suppose that T is rooted at the vertex u. From the algorithm it follows that the distances in G and T between a vertex and any of its ancestors are the same. Pick two vertices $v, w \in V$ and proceed by induction on $d_G(v, w)$. If v and w are adjacent, then we are done, because then $d_T(v, w) \leq 4$. Now suppose that $d_G(v, w) = s \geq 2$ and let z be a neighbor of v on a shortest path between v and w. From the induction assumption we have $d_T(z, w) \leq s - 1 + 3 = s + 2$ and $d_T(v, z) \leq 4$. Let $a = nca(v, z)$ be the nearest common ancestor of v and

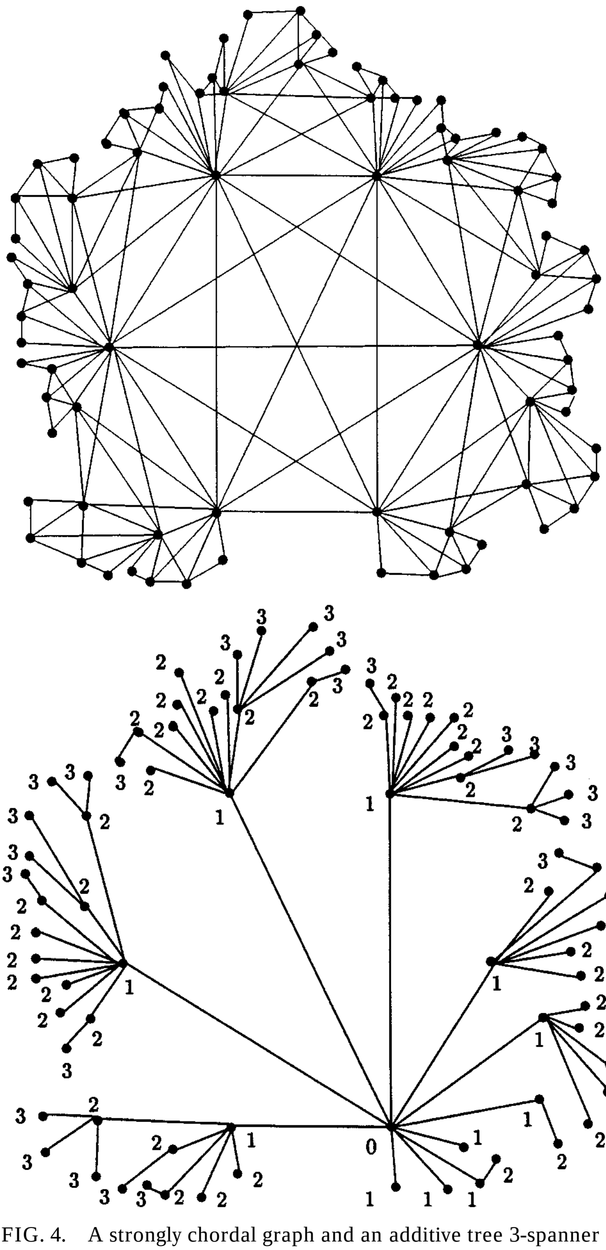

FIG. 4. A strongly chordal graph and an additive tree 3-spanner of it.

z in the tree T. Since $d_T(a, v) = d_G(a, v)$, $d_T(a, z) = d_G(a, z)$, and $vz \in E$, we obtain that $|d_T(v, a) - d_T(z, a)| \leq 1$.

We can further assume that $d_T(z, w) < d_T(v, w) - 1$; otherwise we immediately conclude that $d_T(v, w) \leq d_G(v, w) + 3$. From this and the previous inequality we deduce that the vertex $nca(w, z)$ lies on the path of T between the vertices a and z. Therefore, a is an ancestor of w, and thus $d_T(a, w) = d_G(a, w)$. Notice that the distance sums $d_T(v, w) + d_T(a, z)$ and $d_T(v, z) + d_T(a, w)$ are equal. Hence

$$\begin{aligned} d_T(v, w) &= d_T(a, w) - d_T(a, z) + d_T(v, z) \\ &= d_G(a, w) - d_G(a, z) + d_T(v, z) \\ &\leq d_G(w, z) + 4 \leq d_G(v, w) + 3, \end{aligned}$$

concluding the proof. ∎

COROLLARY 1. *Every strongly chordal graph G admits a spanning tree T that is a multiplicative 4-spanner as well as an additive 3-spanner of G. Such a tree T can be constructed in linear time.*

Figure 5 gives a small strongly chordal graph G without additive tree 2-spanners. This graph G is obtained from four copies H_1, H_2, H_3, and H_4 of a graph H by identifying some vertices and adding some new edges (see Fig. 5). Now assume that G has an additive tree 2-spanner T. Then there must be an index $i \in \{1, 2, 3, 4\}$ such that the vertices a_i and b_i of H_i are connected in T by a path (of length at most three) that does not contain any other vertices of H_i. Indeed, if for every index i vertices a_i and b_i would be connected in T by a path using only the vertices of H_i, then the edge $a_1 b_4$ of G cannot be an edge of T, but $d_T(a_1, b_4) \geq 4$ will hold. Hence, to get a contradiction of our assumption it is enough to show that the graph H_i^+ with tree edges $a_i x$ and $b_i x$ does not have any additive tree

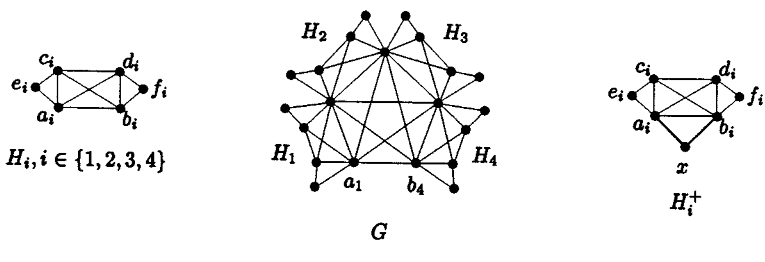

FIG. 5. A strongly chordal graph G that does not have an additive tree 2-spanner.

2-spanner that contains both edges $a_i x$ and $b_i x$. But this can be done directly. The case in which a_i and b_i are connected outside H_i by a path of length 3 of T is even simpler.

5. SPANNERS OF DUALLY CHORDAL GRAPHS

Below we show that Procedure 2 from the previous section can be applied to produce multiplicative 4-spanners and additive 3-spanners in dually chordal graphs. These graphs were introduced in [14] as a generalization of strongly chordal graphs (which are the hereditary dually chordal graphs) where the Steiner tree problem and many domination-like problems still have efficient solutions. It turns out that the dually chordal graphs are exactly the intersection graphs of maximal cliques of chordal graphs (see [7, 33]).

To define dually chordal graphs, we need some notions from the theory of hypergraphs [3]. Let $\mathscr{E}$ be a hypergraph with underlying set V, i.e., $\mathscr{E}$ is a collection of subsets of V. The *dual hypergraph* $\mathscr{E}^*$ has $\mathscr{E}$ as its vertex set, and for every $v \in V$ a hyperedge $\{e \in \mathscr{E}\colon v \in e\}$. The *line graph* $L(\mathscr{E}) = (\mathscr{E}, E)$ of $\mathscr{E}$ is the intersection graph of $\mathscr{E}$, i.e., $ee' \in E$ if and only if $e \cap e' \neq \emptyset$. A *Helly hypergraph* is one whose edges satisfy the Helly property, that is, any subfamily $\mathscr{E}' \subseteq \mathscr{E}$ of pairwise intersecting edges has a nonempty intersection. A hypergraph $\mathscr{E}$ is a *hypertree* if there is a tree T with vertex set V such that every edge $e \in \mathscr{E}$ induces a subtree in T. Equivalently, $\mathscr{E}$ is a hypertree if and only if the line graph $L(\mathscr{E})$ is chordal and $\mathscr{E}$ is a Helly hypergraph. A hypergraph $\mathscr{E}$ is a *dual hypertree* (*α-acyclic hypergraph*) if there is a tree T with vertex set $\mathscr{E}$ such that, for every vertex $v \in V$, $T_v = \{e \in \mathscr{E}\colon v \in e\}$ induces a subtree of T. Observe that $\mathscr{E}$ is a hypertree if and only if $\mathscr{E}^*$ is a dual hypertree.

For a graph G by $\mathscr{C}(G) = \{C : C$ is a maximal clique in $G\}$ we denote the *clique hypergraph*. Furthermore, let $\mathscr{D}(G) = \{D_k(v)\colon v \in V,\ k$ a nonnegative integer$\}$ be the *disk hypergraph* of G. A graph G is called *dually chordal* if the clique hypergraph $\mathscr{C}(G)$ is a hypertree [7, 14, 33]. In [7, 14] it is shown that dually chordal graphs are the graphs G whose disk hypergraphs $\mathscr{D}(G)$ are hypertrees (see [7, 14] for other characterizations, particularly in terms of certain elimination schemes, and [8] for their algorithmic use). From the definition of hypertrees we deduce that for dually chordal graphs the line graphs of the clique and disk hypergraphs are chordal. Conversely, if G is a chordal graph, then $\mathscr{C}(G)$ is a dual hypertree, and, therefore, the line graph $L(\mathscr{C}(G))$ is a dually chordal graph, justifying the term "dually chordal graphs." Finally note that graphs that are both chordal and dually chordal were dubbed *doubly chordal* and were investigated in [7, 14, 25].

Henceforward, we will suppose that G is a dully chordal graph. To prove that Procedure 2 indeed constructs an additive 3-spanner of G, we need some properties of two-sets and graphs H_k constructed from G.

LEMMA 7. *For any two-set $M \subseteq N_k(u)$ of G there exists a vertex $v \in N_{k-1}(u)$ that is adjacent to all vertices of M.*

Proof. Indeed, consider the disks $\{D_1(w): w \in M\}$ and $D_{k-1}(u)$. Since these disks intersect pairwise, by the Helly property they share a common vertex v. Necessarily, $v \in N_{k-1}(u)$ because $M \subseteq N_k(u)$. ∎

LEMMA 8. *Let S be a connected component of the subgraph of G induced by $N_k(u)$. Then $M := N(S) \cap N_{k-1}(u)$ is a two-set, and, moreover, any two nonadjacent vertices of M have a common neighbor in M. In particular, the graph Γ, defined in Section 2, is a tree.*

Proof. Pick two arbitrary nonadjacent vertices $x, y \in M$. Then we can find two vertices $v, w \in S$ such that x is adjacent to v and y is adjacent to w. Consider a path P of S connecting the vertices v and w. The disk $D_{k-2}(u)$, together with the disks of the family $\mathscr{M} := \{D_1(z): z \in P \cup \{x, y\}\}$, forms a cycle in the line graph $L(\mathscr{D}(G))$. From the chordality of $L(\mathscr{D}(G))$ and since $D_{k-2}(u) \cap D_1(z) = \emptyset$ for all $z \in P$, we deduce that $D_1(x) \cap D_1(y) \neq \emptyset$, i.e., $d_G(x, y) = 2$. (Here we used the following property of chordal graphs: for each vertex t on a cycle C of length at least 4, either the neighbors of t in C are adjacent or C has a chord incident to t.) Applying once more the property to the cycle of $L(\mathscr{D}(G))$ formed by the disks of $\mathscr{M}$, we will find in this cycle a disk $D_1(z')$, which together with $D_1(x)$ and $D_1(y)$, induces a triangle in $L(\mathscr{D}(G))$. By the Helly property the pairwise intersecting disks $D_1(z')$. $D_1(x)$, and $D_1(y)$ have a common vertex s. If $s \in N_{k-1}(u)$ we are done. So assume that $s \in S$. The disks $D_{k-1}(u)$, $D_1(s)$, $D_1(x)$, and $D_1(y)$ intersect pairwise. Again, by the Helly property and since $d_G(s, u) = k$, we can find in M a common neighbor of the vertices x, y, and s. ∎

As in Procedure 2, let $S_1^{k+1}, \dots, S_{p_{k+1}}^{k+1}$ denote the connected components of the subgraph of G induced by $N_{k+1}(u)$, $k \geq 1$. Set $M_j = N(S_j^{k+1}) \cap N_k(u)$, $j = 1, \dots, p_{k+1}$ (the two-sets M_j play the role of the cliques C_j in the case of strongly chordal graphs). Denote by H_k the graph with $\bigcup_{j=1}^{p_{k+1}} M_j$ as the vertex set, and two vertices x, y are adjacent in H_k if and only if they belong to a common set M_j. We continue with the following result, which is analogous to Lemma 6.

LEMMA 9. *Every connected component F of the graph H_k $(k \geq 1)$ is a two-set of G.*

Proof. Let x, y be two nonadjacent vertices of a connected component F of the graph H_k. Then we can find a collection of two-sets $M_{i_1}, M_{i_2}, \dots, M_{i_h}$ such that $x \in M_{i_1}, y \in M_{i_h}$, and $M_{i_j} \cap M_{i_{j+1}} \neq \emptyset$ for all $j = 1, \dots, h-1$. Pick $z_j \in M_{i_j} \cap M_{i_{j+1}}$, $j = 1, \dots, h-1$, and let $z_0 := x$ and $z_h := y$. Since $z_{j-1}, z_j \in M_{i_j}$, $j = 1, \dots, h$, we can find two vertices $v'_j, v''_j \in S_{i_j}^{k+1}$ adjacent to z_{j-1} and z_j, respectively. Let P_j be a path of $S_{i_j}^{k+1}$ connecting the vertices v'_j and v''_j. The disk $D_{k-1}(u)$, together with $D_1(x), D_1(y)$ and the disks of the family $\{D_1(z): z \in \bigcup_{j=1}^h P_j\}$, forms a cycle in the line graph $L(\mathscr{D}(G))$. From chordality of this graph and since $D_{k-1}(u) \cap D_1(z) = \emptyset$ holds for all $z \in \bigcup_{j=1}^h P_j$, we deduce that $D_1(x) \cap D_1(y) \neq \emptyset$, i.e., $d_G(x, y) = 2$. ∎

To find the connected components of the graph H_k we construct again a special bipartite graph $B_k = (W, K; U)$. In this graph $W = \{s_1, \dots, s_{p_{k+1}}\}$ (a vertex s_j represents a set S_j^{k+1}), and K is the vertex set of H_k. A vertex $s_j \in W$ and a vertex $v \in K$ are adjacent in B_k if and only if $v \in N(S_j^{k+1})$. The graph B_k can be constructed in $O(\sum_{v \in N_{k+1}(u) \cup N_k(u)} deg(v))$ time. Note that a vertex $v \in N_k(u)$ belongs to H_k if and only if it has a neighbor in $N_{k+1}(u)$. The connected components of H_k are exactly the intersections of the connected components of B_k with the set K, and thus can be found within the same time bound. All other steps of the procedure can be implemented in the same way as for strongly chordal graphs. Summarizing, the whole procedure for a dually chordal graph G requires

$$O\left(\sum_{k=1}^{q}\left(\sum_{v \in N_k(u)} deg(v) + |N_k(u)|\right)\right) = O(|V| + |E|)$$

time.

THEOREM 3. *Let G be a dually chordal graph. The tree $T = (V, E')$ constructed by Procedure 2 is a multiplicative 4-spanner as well as an additive 3-spanner of G.*

Proof. The proof is the same as the proof of Theorem 2. First, from Lemma 8 we conclude that T is a spanning tree of G. We will prove only that $d_T(x, y) \leq 4$ for any edge xy of G. First suppose that $x, y \in S_j^k$. If x and y belong to a common connected component of H_k, by Lemmas 7 and 9 we deduce that x and y have a common father in T, and thus $d_T(x, y) = 2$. Otherwise, let x' and y' be their fathers in T. By Lemma 8 we obtain that x' and y' belong to a common connected component of H_{k-1}. But we know that in this case $d_T(x', y') = 2$; hence $d_T(x, y) = 4$.

Now suppose that $x \in N_k(u)$ and $y \in N_{k-1}(u)$. Let $x' \neq y$ be the father of x in T (if $x' = y$ we are done). By Lemma 8 x' and y either are adjacent or have a common neighbor in $N_{k-1}(u)$. In any case they both belong to a common connected component of H_{k-1}. Then $d_T(x', y) = 2$, and hence $d_T(x, y) = 3$. ∎

REFERENCES

1. I. Althöfer, G. Das, D. Dobkin, D. Joseph, and J. Soares, On sparse spanners of weighted graphs, *Discrete Comput. Geom.* **9** (1993), 81–100.
2. J.-P. Barthélemy and A. Guénoche, "Trees and Proximity Representations," Wiley, New York, 1991.
3. C. Berge, "Hypergraphs," Horth-Holland, Amsterdam, 1989.
4. S. Bhatt, F. Chung, F. Leighton, and A. Rosenberg, Optimal simulations of tree machines, *in* "27th IEEE Foundations of Computer Science, Toronto, 1986," pp. 274–282.
5. U. Brandes and D. Handke, "*NP*-Completeness Results for Minimum Planar Spanners," preprint, University of Konstanz, *Konstanzer Schrift. Math. Inform.* **16** (1996).
6. A. Brandstädt, V. Chepoi, and F. Dragan, Distance approximating trees for chordal and dually chordal graphs, Graz, Austria, September 1997, *Lecture Notes in Comp. Sci.* **1284** (1997), 78–91.
7. A. Brandstädt, F. Dragan, V. Chepoi, and V. Voloshin, Dually chordal graphs, Graph-Theoretic Concepts in Computer Science, *Lecture Notes in Comput. Sci.* **790** (1994), 237–251 (to appear in SIAM J. Discrete Math.)
8. A. Brandstädt, V. Chepoi, and F. Dragan, Clique *r*-domination and clique *r*-packing problems on dually chordal graphs, *SIAM J. Discrete Math.* **10** (1997), 109–127.
9. P. Buneman, A characterization of rigid circuit graphs, *Discrete Math.* **9** (1974), 205–212.
10. L. Cai and D. G. Corneil, Tree spanners, *SIAM J. Discrete Math.* **8** (1995), 359–387.
11. G. J. Chang and G. L. Nemhauser, The *k*-domination and *k*-stability problems on sun-free chordal graphs, *SIAM J. Alg. Discrete Math.* **5** (1984), 332–345.
12. L. P. Chew, There are planar graphs almost as good as the complete graph, *J. Comput. Syst. Sci.* **39** (1989), 205–219.
13. G. A. Dirac, On rigid circuit graphs, *Abh. Math. Sem. Univ. Hamburg* **25** (1961), 71–76.
14. F. Dragan, C. Prisacaru, and V. Chepoi, Location problems in graphs and the Helly property (in Russian), *Discrete Math. Moscow* **4** (1992), 67–73.
15. D. B. A. Epstein, J. W. Cannon, D. F. Holt, S. V. F. Levy, M. S. Paterson, and W. P. Thurston, "Word Processing in Groups," Jones and Bartlett, Boston, 1992.
16. M. Farber, Characterization of strongly chordal graphs, *Discrete Math.* **43** (1983), 173–189.
17. M. Farber and R. E. Jamison, Convexity in graphs and hypergraphs, *SIAM J. Alg. Discrete Math.* **7** (1986), 433–444.
18. E. Ghys and P. de la Harpe, Les Groupes Hyperboliques d'après M. Gromov, *Prog. Math.* **83** (1990).
19. M. C. Golumbic, "Algorithmic Graph Theory and Perfect Graphs," Academic Press, London, 1980.
20. D. Harel and R. E. Tarjan, Fast algorithms for finding nearest common ancestors. *SIAM J. Comput.* **13** (1984), 338–355.
21. Hoang-Oanh Le, Personal communication.
22. A. L. Liestman and T. Shermer, Additive graph spanners, *Networks* **23** (1993), 343–364.
23. N. Linial, E. London, and Y. Rabinovich, The geometry of graphs and some of its algorithmic applications, *Combinatorica* **15** (1995), 215–245.
24. M. S. Madanlal, G. Venkatesan, and C. Pandu Rangan, Tree 3-spanners on interval, permutation and regular bipartite graphs, *Inform. Proc. Lett.* **59** (1996), 97–102.

25. M. Moscarini, Doubly chordal graphs, Steiner trees and connected domination, *Networks* **23** (1993), 59–69.
26. R. Paige and R. E. Tarjan, Three partition refinement algorithms, *SIAM J. Comput.* **16** (1987), 973–989.
27. D. Peleg and A. A. Schäfer, Graph spanners, *J. Graph Theory* **13** (1989), 99–116.
28. D. Peleg and J. D. Ullman, An optimal synchronizer for the hypercube, *in* ''Proceedings of the 6th ACM Symposium on Principles of Distributed Computing, Vancouver, 1987,'' pp. 77–85.
29. E. Prisner, Distance approximating spanning trees, Proceedings of STACS'97, Lecture Notes in Comput. Sci. **1200** (1997), 499–510.
30. P. H. A. Sneath and R. R. Sokal, ''Numerical Taxonomy,'' San Francisco, 1973.
31. J. P. Spinrad, Doubly lexical ordering of dense 0–1-matrices, *Inform. Proc. Lett.* **45** (1993), 229–235.
32. D. L. Swofford and G. J. Olsen, Phylogeny reconstruction, *in* (D. M. Hills and C. Moritz, Eds.), ''Molecular Systematics,'' pp. 411–501, Sinauer Associates, Sunderland, MA, 1990.
33. J. L. Szwarcfiter and C. F. Bornstein, Clique graphs of chordal and path graphs, *SIAM J. Discrete Math.* **7** (1994), 331–336.