Additive Tree Spanners

Dieter Kratsch*
Institut für Informatik
Fr.-Schiller-Universität Jena

Hoàng-Oanh Le
Wilhelmshavener Str. 22
10551 Berlin

Haiko Müller†
Institut für Informatik
Fr.-Schiller-Universität Jena

Erich Prisner ‡
Mathematisches Seminar
Universität Hamburg

Dorothea Wagner§
Fakultät für Mathematik und Informatik
Universität Konstanz

August 19, 1998

Abstract

A spanning tree of a graph is a k-*additive tree spanner* whenever the distance of every two vertices in the graph or in the tree differs by at most k. In this paper we show that certain classes of graphs, as distance-hereditary graphs, interval graphs, asteroidal-triple free graphs, allow some constant k such that every member of the class has some k-additive tree spanner. On the other hand, there are chordal graphs without k-additive tree spanner for arbitrary large k.

Key words: distance, graph spanners, spanning tree, algorithm, AT-free graph, distance-hereditary graph, chordal graph,
AMS subject classification: 05C12, 05C85, 68R10, 68Q20, 90B80,

1 Introduction

Spanning trees are often used in applications where we want to save edges but maintain connectivity. Since in most of these applications distance matters, not all spanning trees have the same quality. It should be required that vertices of small distance in the original graph should also have relatively small distance in the spanning tree. If we require $d_T(x,y)/d_G(x,y) \leq k$, for all pairs of vertices x, y, then we arrive at the well-known concept of k-multiplicative tree spanners, compare [3]. The stronger property $d_T(x,y) - d_G(x,y) \leq k$, for every pair of vertices x, y, defines k-*additive* spanners, compare [12].

*email: `kratsch@minet.uni-jena.de`
†email: `hm@minet.uni-jena.de`
‡email: `ms6a013@math.uni-hamburg.de`; supported by the Deutsche Forschungsgemeinschaft under grant no. Pr 324/6-1.
§email: `dorothea.wagner@uni-konstanz.de`

In this paper we shall deal with spanning subtrees that are k-additive spanners. We call them also *k-additive tree spanners.* Surely trees are the only graphs with 0-additive tree spanners.

In this paper we will show that certain well-known graph classes Γ allow some constant k_Γ such that every graph in Γ has some k_Γ-additive tree spanner. We present two simple approaches for finding such trees. The first is to consider certain breadth-first search trees. For block graphs, interval graphs and distance-hereditary graphs, we get 1-additive and 2-additive, respectively, tree spanners in linear time. The second approach requires that we have some dominating shortest path. We connect the vertices outside the path to the path in a consistent way to obtain some tree spanner, which, as can be shown, is 4-additive.

Some other classes Γ of graphs do not have such a constant k_Γ. This holds for instance if all cycles belong to Γ, since cycles of length $k+3$ do not have a k-additive tree spanner. But even the class of chordal graphs does not have such a constant, as we shall see.

Parts of the results of the paper appeared in [10] and [14].

2 Spanning trees growing from isometric subtrees

For a set $W \subseteq V$ of vertices in a graph G and any integer $i \geq 0$ let $N^i(W)$ denote the ith neighborhood of W, i.e., the set of all vertices y with $d_G(y, W) = i$, where $d_G(y, W) = \min_{w \in W} d_G(y, w)$.

Definition 2.1 *A tree T is an* isometric subtree *of the graph $G = (V, E)$ if there is a set $W \subseteq V$ such that $T = G[W]$ and $d_G(u, v) = d_T(u, v)$ holds for all $u, v \in W$.*

The most straightforward method of constructing a good additive tree spanner in a connected graph G seems to be the following:

Basic construction:

begin
  choose an isometric subtree $G[W]$ of G with $W \neq \emptyset$;
  $T \leftarrow G[W]$; $i \leftarrow 1$;
  while $N^i(W) \neq \emptyset$ **do begin**
    for $y \in N^i(W)$ **do begin**
      choose a vertex $z \in N(y) \cap N^{i-1}(W)$;
      $V(T) \leftarrow V(T) \cup \{y\}$;
      $E(T) \leftarrow E(T) \cup \{yz\}$;
    end;
    $i \leftarrow i + 1$;
  end;
end.

Since G is connected this construction leads to a spanning tree $T = (V(T), E(T))$ of G. Then every $y \in N^i(W)$ has exactly one neighbor in $N^{i-1}(W)$ on the tree constructed. So, what we do is to construct a spanning tree T where all distances from W towards the other vertices are identical in G and T. Later we will specify a rule f that assigns to each vertex $y \in N^i(W)$ a neighbor $z = f(y)$ in $N^{i-1}(W)$. As isometric subtree $G[W]$ we will use a shortest path of G.

Lemma 2.2 *If $G[W]$ is an isometric subtree of G such that $N^{k+1}(W) = \emptyset$ then a spanning tree T of G constructed by our basic construction is $4k$-additive.*

Proof: We consider a shortest path $(v_0, v_1, \ldots, v_t)$ in G. Let w_0 and w_t be those vertices in W such that $d_T(v_i, w_i) = d_G(v_i, W)$ for $i = 0, t$. Then $d_T(v_i, w_i) \leq k$ since $N^{k+1}(W) = \emptyset$ and $d_T(w_0, w_t) \leq d_G(w_0, v_0) + d_G(v_0, v_t) + d_G(v_t, w_t) \leq 2k + t$ since $G[W]$ is an isometric subtree of G. Consequently, $d_T(v_0, v_t) \leq 4k + t$. □

Now we consider the special case $W = \{x_0\}$. Whether the resulting tree is k-additive depends only on whether $f^i(y) = f^j(z)$ is possible with small i, j, for every edge yz of G. So we do not have to compute all T-distances, but only T-distances between G-adjacent vertices.

Lemma 2.3 *A spanning tree T of G constructed by our basic construction starting from a singleton W is k-additive whenever $d_T(y, z) \leq k + 1$ for every edge yz of G.*

Proof: Necessity of this condition is obvious.

For sufficiency, assume that $d_T(y, z) \leq k + 1$ for every edge yz of G. By induction over t we prove $d_T(v_0, v_t) \leq k + t$ for paths $v_0, v_1, \ldots, v_t$ of length t in G. The case $t = 1$ is just the assumption, now assume that $d_G(u, v) = t > 1$ and assume that $d_T(y, z) \leq d_G(y, z) + k$ whenever $d_G(y, z) < t$ (the induction hypothesis). Let p be the neighbor of v on a shortest $u - v$ path. Now $|d_G(x_0, p) - d_G(x_0, v)| \leq 1$, and $|d_G(x_0, p) - d_G(x_0, u)| \leq t - 1$. By the construction of T there is some vertex q on the $p - v$ path $p = p_r, p_{r-1}, \ldots, p_0 = q = v_0, v_1, \ldots, v_s = v$ of T such that $d_G(x_0, p_i) = d_G(x_0, q) + i$ and $d_G(x_0, v_j) = d_G(x_0, q) + j$. Moreover $r + s \leq k + 1$ and $|r - s| \leq 1$.

The induction hypothesis implies $d_T(u, p) \leq k + t - 1$. Let now $u = u_0, u_1, u_2, \ldots$ $\ldots u_\ell$ be the beginning of the $u - p$ path on T, with u_ℓ the first vertex on the path $p_r, p_{r-1}, \ldots, p_1, q, v_1, \ldots, v_s$. If $u_\ell = q$, then $d_T(u, v) = d_T(u, p) + s - r \leq d_T(u, p) + 1$ since $|r - s| \leq 1$. So we get $d_T(u, v) \leq k + t$ in this case. In case $u_\ell \neq q$ we obtain $d_G(u_{i+1}, x_0) + 1 = d_G(u_i, x_0)$ for every $0 \leq i \leq \ell - 1$, by the special shape of T. But $|d_G(u, x_0) - d_G(p, x_0)| \leq t - 1$, whence $d_T(u, q) \leq t - 1 + r$. Consequently $d_T(u, v) \leq t - 1 + r + s \leq k + t$ in this case also. □

Although starting with singletons has this strong property and works for several graph classes, as we shall see, it has also one disadvantage. It does not appear to construct the optimum spanning tree—that is one which is k-additive for the smallest possible k.

As an example, the graph of Figure 1 has some 1-additive tree spanner, which, however, will not be found by our approach under *any* rule and *any* start vertex x_0.

On the other hand, if we start our construction with maximum isometric subtrees then the assertion of Lemma 2.3 does not hold. As an example, on the graph G depicted in Figure 2 we obtain a spanning tree T with $d_T(y, z) \leq 4$ for every edge yz of G, if we start with a maximum isometric subtree (all such subtrees are shortest paths on 5 vertices). However, T is not 3-additive.

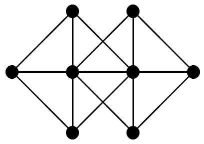

Figure 1: $|W| = 1$ is not optimal

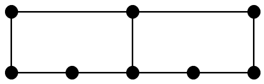

Figure 2: $|W| = \max$ is not optimal

For the algorithms we assume that the vertices of G are linearly ordered, and that the graph is given by means of its ordered neighborhood lists, i.e. for every vertex x there is some list $\textsc{Neigh}(x)$ containing its neighbors in increasing order.

The levels $N^i(W)$ can be computed in linear time $O(|V|+|E|)$ by breadth-first search. Moreover, it is also possible to compute the induced subgraphs $G[N^i(W)]$ in the levels in linear time—we simply check all neighborhood lists one after another, and delete in every list $\textsc{Neigh}(x)$ those neighbors that are in a different level than x. The resulting graph is $\bigcup_i G[N^i(W)]$.

How quickly T can be constructed surely depends on the rules, which themselves depend on the classes considered.

3 Distance-hereditary Graphs

Definition 3.1 *A connected graph $G = (V, E)$ is* distance-hereditary *if every induced x-y path, $x, y \in V$, has length $d_G(x, y)$.*

Several characterizations of distance-hereditary graphs are given in [1] and [8].

Let x_0 be any fixed vertex in a connected distance-hereditary graph G. Given the linear order of the vertices of G, we define the rule

Rule 1 *Connect every $y \in N^{i+1}(x_0)$ with the smallest vertex $f(y) \in N(y) \cap N^i(x_0)$.*

Theorem 3.2 *Every connected distance-hereditary graph G has some 2-additive tree spanner which can be found by our basic construction and rule 1 in linear time.*

Proof: Let T be the tree constructed by the rule above, and let yz be some edge of G.

(1) If $d_G(y, x_0) = d_G(z, x_0) = i + 1$, then $N(y) \cap N^i(x_0) = N(z) \cap N^i(x_0)$ [8], thus $f(y) = f(z)$, and $d_T(y, z) \leq 2$.

(2) If w.l.o.g. $i+1 = d_G(y, x_0) = d_G(z, x_0)+1$, we may assume $f(y) \neq z$ — otherwise we are already done.

Then there are shortest y-x_0 paths in G, one going over $f(y)$, and one over z. In the terminology of [8], $f(y)$ and z are tied, and it has been shown in [8] that this implies

$N(f(y)) \cap N^{i-1}(x_0) = N(z) \cap N^{i-1}(x_0)$. Therefore $f(f(y)) = f(z)$, and $d_T(y,z) = 3$, as desired.

Finally we apply Lemma 2.3.

Finding the tree T is very easy: We simply compute the levels, and then check for every vertex y the neighborhood list in increasing order until we find some vertex one level beyond the level of y — the resulting vertex is $f(y)$. □

At the end of section 5 we will give an example showing that the bound given in Theorem 3.2 is optimal.

4 Block Graphs

Definition 4.1 *A graph is called* block graph *if and only if all its 2-connected components are complete.*

Obviously, the block graphs are just the chordal $(K_4 - e)$-free graphs. Chordal graphs are defined in Definition 7.1 on page 8. Block graphs are distance-hereditary.

Theorem 4.2 *Every connected block graph has some 1-additive tree spanner, that can be found by our basic construction and any rule in linear time.*

Proof: Let x_0 be a vertex in some $(K_4 - e)$- and C_4-free distance-hereditary graph G.

Then every vertex $y \in N^i(x_0)$ must have *exactly* one neighbor in $N^{i-1}(x_0)$.

For, assume otherwise $z_1, z_2 \in N_G(y) \cap N^{i-1}(x_0)$ with $z_1 \neq z_2$. Again z_1 and z_2 are tied, thus every neighbor w of z_1 in $N^{i-2}(x_0)$ is also adjacent to z_2. The four vertices y, z_1, z_2, and w induce a $K_4 - e$ or a C_4 in G, a contradiction.

Thus any rule yields identical results, and case (2) in the proof above cannot occur. Thus every spanning tree constructed by the approach is 1-additive. But block graphs are $(K_4 - e)$- and C_4-free distance-hereditary graphs. □

Again, the bound given in Theorem 4.2 is the best possible, since a complete graph on 3 vertices has no 0-additive tree spanner.

5 Interval Graphs

Definition 5.1 *A graph is an* interval graph *if one can associate with each vertex an interval on the real line such that two vertices are adjacent if and only if the corresponding intervals have a nonempty intersection.*

In [11] the interval graphs are characterized as those chordal graphs (see Definition 7.1 on page 8) without asteroidal triples (see Definition 6.1 on page 7).

By a probably folklore result, chordal graphs have the nice feature that the neighbors of vertices or edges in the lower level look nice:

Lemma 5.2 *Let x_0 be a vertex in the chordal graph G.*

(a) *For every $x \in N^{i+1}(x_0)$, $N(x) \cap N^i(x_0)$ induces a complete graph.*

(b) *For every edge $yz \in N^{i+1}(x_0)$, the two sets $N(y) \cap N^i(x_0)$ and $N(z) \cap N^i(x_0)$ must be comparable.*

Proof: a) Assume $x \in N^{i+1}(x_0)$ had two nonadjacent neighbors y, z in $N^i(x_0)$. Then we choose some chordless y-z path that, except y and z, uses only vertices inside the levels $N^0(x_0)$ up to $N^{i-1}(x_0)$. Together with the edges zx and xy it forms an induced cycle of length 4 or more, a contradiction.

b) Again, assume to the contrary that there are vertices $y', z' \in N^i(x_0)$ such that y' is adjacent to y but not to z, and z' is adjacent to z, but not to y. If y' and z' are adjacent, then we have some induced 4-cycle in G. Otherwise, again we find some induced y'-z' path where all internal vertices have distance less than i to x_0. Together with the edges $z'z, zy, yy'$ this path yields an induced cycle of length at least 5 in G, a contradiction again. $\square$

Lemma 5.3 *For every interval graph and every connected component Q of $G[N^{i+1}(x_0)]$, there is some vertex $f(Q) \in N^i(x_0)$ adjacent to all vertices of Q.*

Proof: We assume that there are two vertices y and z in some common connected component Q of $G[N^{i+1}(x_0)]$, whose sets of neighbors inside $N^i(x_0)$ are not comparable. By Lemma 5.2 (b), y and z are not adjacent. We find y' and z' in $N^i(x_0)$ such that y' is adjacent to y but not to z, and z' is adjacent to z but not to y.

Then the three vertices x_0, y, and z form an asteroidal triple: The y-z path inside $N^{i+1}(x_0)$ avoids x_0 and its neighbors, every shortest y-x_0 path going over y' avoids z and its neighbors, and every shortest z-x_0 path going over z' avoids y and its neighbors. Consequently the nonempty sets $N_G(y) \cap N^i(x_0), y \in V(Q)$, form a chain, thus some element is contained in all these sets. $\square$

Let all the vertices $f(Q)$ for the components Q of the levels $G[N^{i+1}(x_0)]$ be chosen in advance, for instance we could choose the smallest element (in the given ordering of the vertices) in $\bigcap_{y \in Q} N_i(y) \cap N^i(x_0)$. The following rule fits into our general scheme:

Rule 2 *Connect $y \in N^{i+1}(x_0)$ with $f(Q)$, where Q denotes the component of $G[N^{i+1}(x_0)]$ containing y.*

Theorem 5.4 *Every connected interval graph has some 2-additive tree spanner, that can be found by our basic construction and rule 2 in linear time.*

Proof: By Lemma 2.3 it suffices to show that $d_T(y, z) \le 3$ for every edge yz of G for the tree T constructed in this way. The case where both y and z have the same distance $d_G(y, x_0) = d_G(z, x_0) = i + 1$ towards x_0 is easy: Then both lie in the same component Q of $G[N^{i+1}(x_0)]$, whence they are connected over $f(Q)$ in T. So assume that $i + 1 = d_G(y, x_0) = d_G(z, x_0) + 1$, and let Q denote the component of $G[N^{i+1}(x_0)]$ containing y. Since we are done if $f(Q) = z$, assume $f(Q) \ne z$. Then both $f(Q)$ and

z are neighbors of y in $N^i(x_0)$ — by Lemma 5.2 (a) they must be adjacent. By the construction they have distance two in T, whence $d_T(y,z) \leq 3$.

To find T, we first compute the levels, that is, mark every vertex x by some label $d_G(x,x_0)$. Then we compute $\bigcup_i G[N^i(x_0)]$ in linear time, as mentioned above. The components of the levels are just the components of this graph, and they can also be computed in linear time. We also create ordered lists $\textsc{Level}(i)$, $i \geq 0$, of vertices for the levels, simply be checking all vertices in the order and putting them in the appropriate lists.

All what remains to do is to find for every component Q of every level $N^i(x_0)$ some vertex $f(Q)$ in $N^{i-1}(x_0)$ adjacent to all vertices of Q. This is achieved by checking the vertices in the (ordered) list $\textsc{Level}(i-1)$ one after another, and also browsing the neighborhood lists of the elements of Q from left to right in parallel, stopping the pointers every time the entry exceeds the actual entry of $\textsc{Level}(i-1)$. Since the components partition G, every neighborhood list is traversed exactly once, giving in total linear time-complexity. □

Both Theorems 5.4 and 3.2 are best possible. The graph in Figure 3 has no 1-additive tree spanner, but it is both distance-hereditary and an interval graph.

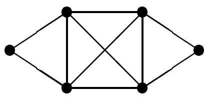

Figure 3: A graph without 1-additive tree spanner.

6 AT-free Graphs

Definition 6.1 *An independent set of three vertices is called an* asteroidal triple *if between each pair in the triple there exists a path that avoids the neighborhood of the third. A graph is asteroidal triple free (AT-free) if it contains no asteroidal triple.*

In this section we use a dominating shortest path (DSP) as isometric subtree. More precisely, a shortest path $(x_0, x_1, \ldots, x_\ell)$ in $G = (V,E)$ is *dominating* if every vertex in $V \setminus W$, $W = \{x_0, x_1, \ldots, x_\ell\}$, is adjacent to at least one vertex in W. By Lemma 2.2 we know that every graph with DSP has a 4-additive tree spanner. This applies to all AT-free graphs as a consequence of the Dominating Pair Theorem given in [6]. It is worth mentioning that a DSP in an AT-free graph G can be found in linear time by $2 \times$ LexBFS [5]: First start a lexicographic breadth-first search (LexBFS) from an arbitrary vertex x of G. Let x_0 be the vertex numbered last by this search. Start a second LexBFS in G from x_0 and let x_ℓ ($\ell = d_G(x_0, x_\ell)$) be the last vertex in the second LexBFS. In [5] is is shown that *every* shortest path $(x_0, x_1, \ldots, x_\ell)$ is a DSP of G.

Next we demonstrate how to use such a DSP in an AT-free graph to show that every AT-free graph admits a 3-additive tree spanner. We need the following claim from [9].

Lemma 6.2 *Let $(x_0, x_1, \ldots, x_\ell)$ be a* DSP *of an AT-free graph $G = (V, E)$ constructed by $2 \times$ LexBFS as described above. Then for $i = 1, 2, \ldots, \ell$ every vertex $z \in N^i(x_0)$ is adjacent to x_i or x_{i-1}.*

Rule 3 *For $i = 1, 2 \ldots, \ell$ connect a vertex $v \in N^i(x_0)$ to $f(v) = x_i$ if v is adjacent to x_i, otherwise connect v to $f(v) = x_{i-1}$.*

Theorem 6.3 *Every connected AT-free graph has a 3-additive tree spanner that can be found in linear time.*

Proof: Let $G = (V, E)$ be a connected AT-free graph and let $(x_0, x_1, \ldots, x_\ell)$ be a DSP of G constructed by $2 \times$ LexBFS. To construct a 3-additive tree spanner of G we use our basic construction with $G[W]$ as isometric subtree, $W = \{x_0, x_1, \ldots, x_\ell\}$ and apply rule 3. This defines a spanning tree T of G, since by Lemma 6.2 for all $v \in V \setminus W$ the vertex $f(v) \in W$ is adjacent to v.

We consider a shortest path $(v_0, v_1, \ldots, v_t)$ in G with $t \geq 1$ (Lemma 2.3 does not apply), and w.l.o.g. $v_0 \in N^i(x_0)$ and $v_t \in N^j(x_0)$, $0 \leq i \leq j \leq \ell$. Then $j - i \leq t$ since $d_G(x_0, v_t) \leq d_G(x_0, v_0) + d_G(v_0, v_t)$. This implies $d_T(f(v_0), f(v_t)) \leq t + 1$ by rule 3. Finally, since $(x_0, x_1, \ldots, x_\ell)$ is a DSP of G we have $d_T(v_0, v_t) \leq t + 3 = d_G(v_0, v_t) + 3$.

Observe that the 3-spanner T constructed in the proof can be found in linear time, since we can find a DSP in AT-free graphs by $2 \times$ LexBFS [5]. □

Moreover, Theorem 6.3 gives the best possible bound since the 5-cycle is an AT-free graph which has no 2-additive tree spanner.

A graph $G = (V, E)$ is a *cocomparability graph* if there is some poset $(V, <)$ such that distinct vertices are adjacent in G if and only if they are not comparable. It is well-known that the cocomparability graphs form a class between the interval graphs and the AT-free graphs [11]. Hence we know by Theorem 6.3 that every cocomparability graph has a 3-additive tree spanner. However, the chordless 4-cycle which has no 1-additive tree spanner gives the best known lower bound for this class.

Conjecture 6.4 *Every cocomparability graph admits a 2-additive tree spanner.*

In a recent algorithm for finding 3-multiplicative tree spanners in interval graphs, the above approach is used implicitly [13]. Actually the tree spanner constructed by this algorithm is 2-additive.

7 Chordal Graphs

Definition 7.1 *A graph is called* chordal *if it does not contain a chordless cycle of length greater than three.*

Since block graphs, as well as interval graphs, are chordal, considering this class might seem promising. However, the following example, independently found by T.A. McKee, shows that for every fixed integer k there are chordal graphs without k-additive tree spanners:

Figure 4: A chordal graph.

Let the graph G_1 be the triangle K_3, and let G_2 be the graph obtained from G_1 by adding three vertices, each one adjacent to distinct two vertices of G_2. Let for every integer $s \geq 2$, the graph G_s be obtained from G_{s-1} and G_{s-2} by adding for every edge in $E(G_{s-1}) \setminus E(G_{s-2})$ one new vertex, adjacent to the two vertices of the edge. These graphs are planar and chordal, even 2-trees and path graphs. G_5, together with some 5-additive tree spanner is given in Figure 5. Look at vertices x and y to see that this particular tree is not 4-additive.

Proposition 7.2 *No $(k-1)$-additive tree spanner and no k-multiplicative tree spanner is possible in G_k.*

Proof: The *eccentricity* $ecc_G(x)$ of x is the largest integer i for which $N^i(x)$ is nonempty (i.e. the largest distance $d_G(x,y), y \in V(G)$).

Look at the canonical embedding of G_k in the plane. The first observation is, that the outer face F_0 of G_k has, as vertex in the dual G_k^*, eccentricity equal to k. In fact, all faces have the same eccentricity k in the dual graph for this example.

Let T be a spanning tree of G_k. The dual tree T^* contains all edges of G_k^* which cross edges of G_k that do *not* belong to T.

Let B be a largest connected component of the forest $T^* - F_0$, and let F_1 be the neighbor of F_0 in B. Note that B contains at least $ecc_{T^*}(F_0) \geq ecc_{G_k{}^*}(F_0) = k$ vertices.

The edge F_0F_1 in T^* crosses an edge xy on the outer cycle in G_k. Since F_0F_1 is an edge in T^*, x and y are not adjacent in T. Moreover $d_T(x,y)+1$ equals the number of edges in G_k^* that start in B and end outside B. Since all vertices except F_0 have degree 3 in G_k^*, this number equals

$$\sum_{F \in V(B)} (3 - d_B(F)),$$

but by the well known degree sum formula, and since B is a tree, this equals

$$3|V(B)| - 2|E(B)| = |V(B)| + 2$$

a number which is greater or equal to $k+2$. Therefore $d_T(x,y) \geq k+1$, and T cannot be $(k-1)$-additive or k-multiplicative. □

The class of *strongly chordal graphs* is between the chordal graphs and the interval graphs, see [7] for definition and characterizations. Strongly chordal graphs allow 3-additive spanning trees, as it has been shown recently by a completely different approach [2].

References

[1] Bandelt, H.-J., Mulder, H.M.: Distance-hereditay graphs, *J. Combin. Th. B* **41** (1986) 182–208.

[2] Brandstädt, A., Chepoi, V., Dragan, F.: Distance approximating trees for chordal and dually chordal graphs, preprint 1997.

[3] Cai, L., Corneil, D.G.: Tree spanners, *SIAM J. Discr. Math.* **8** (1995) 359–387.

[4] Golumbic, M.C.: Algorithmic Graph Theory and Perfect Graphs, Academic Press, London (1980).

[5] Corneil, D.G., Olariu, S., Stewart, L.: A linear time algorithm to compute dominating pairs in asteroidal triple-free graphs, Proceedings of the 22nd International Colloquium on Automata, Languages and Programming, Lecture Notes in Computer Science, Vol. 944, pp. 292–302.

[6] Corneil, D.G., Olariu, S., Stewart, L.: Asteroidal triple-free graphs, *SIAM J. Discr. Math.* **10** (1997) 399–430.

[7] Farber, M.: Characterizations of strongly chordal graphs, *Discr. Math.* **43** (1983) 173–189.

[8] Hammer, P.L., Maffray, F.: Completely separable graphs, *Discrete Appl. Math.* **27** (1990) 85–99.

[9] Kloks, T., Kratsch, D., Müller, H.: Approximating the bandwidth for asteroidal triple-free graphs, Proceedings of the Third European Symposium on Algorithms, Lecture Notes in Computer Science, Vol. 979, pp. 434–447.

[10] Hoàng-Oanh Le, Effiziente Algorithmen für Baumspanner in chordalen Graphen, Diploma Thesis, TU Berlin (1994) .

[11] Lekkerkerker, C., Boland, J.: Representation of a finite graph by a set of intervals on the real line, *Fund. Math.* **51** (1962) 45–64.

[12] Liestman, A.L., Shermer, T.: Additive graph spanners, *Networks* **23** (1993) 343–364.

[13] Madanlal, M.S., Vankatesan, G., Pandu Rangan, C.: Tree 3-spanners on interval, permutation and regular bipartite graphs, *Inform. Process. Lett.* 59 (1996) 97–102.

[14] E. Prisner, Distance approximating spanning trees, Proceedings STACS 97, Lecture Notes in Computer Science, Vol. 1200, (1997) pp. 499-510.

[15] Spinrad, J.: On comparability and permutation graphs, *SIAM J. Computing* **14** (1985) 658–670.