Available online at www.sciencedirect.com
SCIENCE DIRECT®
Discrete Mathematics 285 (2004) 83–95
DISCRETE MATHEMATICS
www.elsevier.com/locate/disc

Spanning spiders and light-splitting switches☆

Luisa Gargano[a], Mikael Hammar[a], Pavol Hell[b], Ladislav Stacho[c], Ugo Vaccaro[a]

[a] *Dipartimento di Informatica ed Applicazioni, Università di Salerno, 84081 Baronissi, SA, Italy*
[b] *School of Computing Science, Simon Fraser University, Burnaby, BC, Canada V5A 1S6*
[c] *Department of Mathematics, Simon Fraser University, Burnaby, BC, Canada V5A 1S6*

Received 28 May 2002; received in revised form 3 February 2004; accepted 19 April 2004

Abstract

Motivated by a problem in the design of optical networks, we ask when a graph has a spanning spider (subdivision of a star), or, more generally, a spanning tree with a bounded number of branch vertices. We investigate the existence of these spanning subgraphs in analogy to classical studies of Hamiltonicity.

Keywords: Spanning tree; Branch vertex; Hamilton path

1. Introduction

The existence of a Hamilton path in a given graph G is a much studied problem. It is known that deciding if such a path exists is an NP-complete problem, even for cubic graphs G [12]. On the other hand, if the graph G satisfies any of a number of density conditions, a Hamilton path is guaranteed to exist. The best known of these density conditions, due to Dirac [6], requires each vertex of G to have a degree of at least $n/2$, where n is the number of vertices in G. Other conditions relax the degree constraint somewhat, while requiring at the same time that $K_{1,3}$ (or sometimes $K_{1,4}$) is not an induced subgraph of G. Excluding these subgraphs has the effect of forcing each neighborhood of a vertex to have many edges, allowing us to guarantee the existence of a Hamilton path with a somewhat weaker degree condition.

There are several natural optimization versions of the Hamilton path problem. For instance, one may want to minimize the number of leaves [16], or minimize the maximum degree in a spanning tree of G [1,13,15,21]; either of these numbers is equal to two if and only if G has a Hamilton path. The best known optimization problem of this sort is the longest path problem [2,7,14] (G has a Hamilton path if and only if the longest path has n vertices). It is known that, unless $P = NP$, there is no polynomial time constant ratio approximation algorithm for the longest path problem, even when restricted to cubic graphs which have a Hamilton path, cf. [2,7] where a number of other nonapproximability results are also discussed. In this paper, we introduce another possible optimization problem—minimizing the number of branch vertices in a spanning tree of G.

A *branch vertex* of G is a vertex of degree greater than two. If G is a connected graph, we let $s(G)$ denote the smallest number of branch vertices in any spanning tree of G. Since a spanning tree without branch vertices is a Hamilton path of G, we have $s(G) = 0$ if and only if G admits a Hamilton path. A tree with at most one branch vertex will be called a *spider*. Note that a spider may in fact be a path, i.e., have no branch vertices. Thus a graph G with $s(G) \leqslant 1$ admits a spanning subgraph that is a spider; we will say that G admits a *spanning spider*. There is an interesting intermediate

☆ Research of the first, second and fifth authors was partially supported by the European Community under the RTN project: "Approximation and Randomized Algorithms in Communication Networks (ARACNE)", and by the Italian Ministry of University and of Scientific Research under the PRIN project: "Resource Allocation in Computer Networks". Research of the third and fourth authors was supported by NSERC of Canada. This paper is an expansion and coalescence of the two extended abstracts [8] and [9].
E-mail address: lstacho@sfu.ca (L. Stacho).

0012-365X/$ - see front matter
doi:10.1016/j.disc.2004.04.005

possibility: We will call a graph G *arachnoid*, if it admits a spanning spider centered at each vertex of G. (A spider with a branch vertex is said to be *centered* at the branch vertex; a spider without branch vertices, i.e., a path, is viewed as centered at any vertex.) It follows from these definitions that the set of arachnoid graphs contains the set of graphs G with $s(G)=0$, and is contained in the set of graphs G with $s(G)\leqslant 1$.

Our interest in the problem of minimizing the number of branch vertices arose from a problem in optical networks. The wave division multiplexing technology of optical communication supports the propagation of multiple laser beams through a single optical fiber, as long as each beam has a different wavelength. A *lightpath* connects two nodes of the network by a sequence of fiber links, with a fixed wavelength. Thus two lightpaths using the same link must use different wavelengths. This situation gives rise to many interesting combinatorial problems, cf. [3,10].

We consider a different situation, resulting from a new technology allowing a switch to replicate the signal by *splitting* light. A *light-tree* connects one node to a set of other nodes in the network—allowing *multicast* communication from the source to a set of destinations (including the possibility of the set of destinations consisting of all other nodes). The switches which correspond to the nodes of degree greater than two have to be able to split light (except for the source of the multicast, which can transmit to any number of neighbors). Typical optical networks will have a limited number of these more sophisticated switches, and one has to position them in such a way that all possible multicasts can be performed. Thus we are lead to the problem of finding spanning trees with as few branch vertices as possible.

Specifically, let G be the graph whose vertices are the switches of the network, and whose edges are the fiber links. With $s(G)$ light-splitting switches, placed at the branch vertices of an optimal spanning tree, we can perform all possible multicasts. In particular, if $s(G)=1$, i.e., if G has a spanning spider, we can do with just one special switch. If G is an arachnoid graph, no switches are needed. (Recall that the source of the multicast can transmit to any number of neighbors.) If $s(G)\geqslant 0$, the minimum number of light-splitting switches needed for all possible multicasts in G, is in fact equal to $s(G)$. Indeed, if k vertices of G are allowed to be light-splitting switches, then multicasting from one of these vertices results in a spanning tree of G with at most k branch vertices, thus $k\geqslant s(G)$.

In this paper, we investigate the parameter $s(G)$, with emphasis on graphs which admit a spanning spider, or which are arachnoid, in analogy with the study of graphs which admit a Hamilton path. We show that the recognition problems are all NP-complete, and that $s(G)$ is even hard to approximate. We explore several density conditions, similar to those for Hamilton paths, which are sufficient to give interesting upper bounds on $s(G)$. Finally, we also relate the parameter $s(G)$ to other well studied graph parameters, such as connectivity, independence number, and the length of a longest path.

In dealing with branch vertices of spanning trees, it is helpful to observe that a cut vertex v of a graph G such that $G-v$ has at least three components must be a branch vertex of any spanning tree of G. We will use this observation throughout our arguments.

Let $G=(V,E)$ be a graph on n vertices. (We shall reserve n to denote the number of vertices of G, and to avoid trivialities we shall always assume that $n\geqslant 3$.) For a vertex $v\in V$ we let $d(v)$ denote the degree of v in G. More generally, for a subset $X\subseteq V$ we denote by $d_X(v)$ the number of vertices of X that are adjacent to v in G. We write $\delta(G)=\min_{v\in V} d(v)$, to denote the minimum degree in G, and denote by $\delta_k(G)$ the minimum sum of the degrees of k independent vertices in G.

The neighborhood of a vertex x in G is denoted by $N(x)$. For a subset $X\subseteq V$, the neighborhood of $v\in V$ with respect to X is defined as

$$N_X(v)=\{u\in X \mid uv\in E\}.$$

For sake of simplicity, whenever it is clear from the context, we will identify the vertex set of a subgraph H of G with H itself. Hence, we will use $|H|$ to denote the number of vertices in the graph and $d_H(v)$ and $N_H(v)$ will represent, respectively, the degree and the neighborhood of v with respect to the vertex set of H.

2. Complexity results

In this section, we observe that all of the problems we introduced are NP-complete. We start with graphs admitting a spanning spider, i.e., graphs G with $s(G)\leqslant 1$.

Proposition 1. *It is* NP-*complete to decide whether a given graph G admits a spanning spider*.

Proof. Suppose G is a given graph, and v a given vertex of G. Construct a new graph G' which consists of three copies of G and one additional vertex adjacent to the vertex v of all three copies of G. It is then easy to see that G' has a spanning spider (necessarily centered at the additional vertex), if and only if G admits a Hamilton path starting at v.

Recall that it is NP-complete to decide whether, given a graph G and a vertex v, there exists a Hamilton path in G which starts at v [12]. □

The problem of recognizing arachnoid graphs is also intractable:

Proposition 2. *It is* NP-*complete to decide whether a given graph G is arachnoid.*

Proof. Suppose G is a given graph, and v a given vertex of G. Construct a new graph G' by including a new vertex w in G and adding an edge between v and w. The graph G' is arachnoid if and only if G has a Hamilton path starting from v. □

We close this section by showing that even approximating $s(G)$ seems to be an intractable problem. (More results on nonapproximability can be obtained by the same technique from other results of [2].)

Proposition 3. *Let $k = \mathrm{O}(n^{1-\varepsilon})$, for ε fixed and $0 < \varepsilon < 1$. There is no polynomial time algorithm to check whether $s(G) \leqslant k$, unless* $\mathrm{P} = \mathrm{NP}$.

Proof. Let again G be a given graph, and v a given vertex of G. This time we construct a graph G' from k disjoint copies of G and an additional vertex v' by making the vertex v of every copy of G adjacent to v'. If G admits a Hamilton path starting at v, then G' contains a spanning tree with one branch vertex, centered at v'. On the other hand, if no such Hamilton path exists within G then $s(G') \geqslant k+1$, since every spanning tree of G' must contain at least one branch vertex for every copy of G. □

Note that the result is easily adapted to c-connected graphs, for fixed c. Even among cubic graphs approximation is intractable:

Proposition 4. *Let k be any fixed positive integer. If* $\mathrm{P} \neq \mathrm{NP}$, *then there is no polynomial time algorithm to check whether $s(G) \leqslant k$, even among cubic graphs with $s(G) = 0$.*

Proof. This will follow from [2], and the following observation. Let $\ell(G)$ denote the maximum length of a path in G. (Thus $\ell(G) = n$ if and only if G admits a Hamilton path.) We claim that in any cubic graph G

$$\ell(G) \geqslant \frac{n}{s(G)+1}. \tag{1}$$

Consider a spanning tree T of G with $s(G)$ branch vertices. Noticing that each branch vertex of T has degree 3, we partition the vertices of T into a set of $s(G)+1$ paths as follows: Consider any path in T connecting two leaves and containing exactly one branch vertex; add this path to the partition. Continue recursively on the tree obtained from T by removing all the vertices of the above path (the new tree has one less branch vertex), until all the branch vertices are removed (that is, the last tree is a path).

Hence, the number of paths so constructed is $s(G)+1$. Since the set constructed is a partition of G, there will be at least one path of length $n/s(G)+1$.

It is shown in [2] that there is no polynomial time algorithm guaranteed to test whether or not $\ell(G) \geqslant n/k$, even among cubic graphs G with $\ell(G) = n$, hence the proposition follows. □

3. A density result

In this section we begin to study density criteria which assure that G has a small value of $s(G)$. Recall that $s(G) = 0$ if and only if G admits a Hamilton path. Therefore, a classical result of Dirac (or, more generally, of Ore) can be formulated as follows:

Theorem 1 (Dirac [6], Ore [20]). *If $\delta(G) \geqslant (n-1)/2$ (or, more generally, if $\delta_2(G) \geqslant n-1$), then $s(G) = 0$.*

We believe the following generalization of Dirac's (and Ore's) theorem may hold:

Conjecture 1. Let G be a connected graph and k a nonnegative integer.

If $\delta(G) \geqslant (n-1)/(k+2)$ (or, more generally, if $\delta_{k+2}(G) \geqslant n-1$), then $s(G) \leqslant k$.

While we cannot at this stage prove the general conjecture, we have a proof for the case $k=1$: (For more results of this type, restricted to bipartite graphs, we refer the reader to [8].)

Theorem 2. *Let G be a connected graph. If $\delta(G) \geqslant (n-1)/3$ (or, more generally, if $\delta_3(G) \geqslant n-1$), then G contains a spanning spider.*

Furthermore, there is an $O(n^3)$ time algorithm that finds a spanning spider in G.

In order to construct a spanning spider we first find a suitable long path in the graph. This path will then be turned into a spider that, in the last step, can be extended to span the whole graph. Before we can describe the paths we are looking for, we present some definitions.

Let $P=[v_0v_1\ldots v_t]$ denote a path in G. The *left neighborhood* of $x \in V$ on P is the set

$$N_P^-(x) = \{v_i \mid v_{i+1}x \in E\}.$$

The *right neighborhood* of $x \in V$ on P is defined analogously as

$$N_P^+(x) = \{v_i \mid v_{i-1}x \in E\}.$$

When the underlying path is evident from the context we write $N^-(x)$ and $N^+(x)$ for the left and right neighborhoods, respectively.

Any left neighbor $v_i \in N^-(v_0)$ of v_0 is an end point of the path $P - v_iv_{i+1} + v_0v_{i+1}$ containing the same set of vertices as P; by symmetry, the same holds for $N^+(v_t)$; see Fig. 1. Therefore, we say that the elements in $N^-(v_0)$ and $N^+(v_t)$ are *potential endpoints* with respect to P.

The following set of *maximality criteria* implicitly suggests a local optimization heuristic to find suitably long paths. We obtain this heuristic by showing how to find paths that satisfy the criteria.

Definition 1. A path $P=[v_0\ldots v_t]$ is called *maximal* if either it is a Hamilton path or it satisfies each of the following conditions:

(i) $N(r) \cap N^-(v_0) = \emptyset = N(r) \cap N^+(v_t)$, for every $r \in V-P$.
(ii) $N(v_0) \cap N^+(v_t) = \emptyset$. $\{v_0, v_t\} \cup N^-(v_0) \cap N^+(v_t)$ is an independent set.
(iii) $N^-(r)$ is an independent set, for every $r \in V-P$.
(iv) If $N^-(v_0) \cap N^+(v_t) \neq \emptyset$ then
 (a) no two consecutive vertices in P both have neighbors in $V-P$,
 (b) $V-P$ is an independent set.

We show now that any nonmaximal path $P=[v_0\ldots v_t]$ can be extended in polynomial time.

If condition (i) is violated then there is a vertex r outside P that is adjacent to a potential end point of a path P'. Thus, we construct P' (if r is adjacent to v_0 or v_t then $P'=P$) as described in Fig. 1 and add r to this path.

If condition (ii) is violated then we can find a cycle in G that contains all the vertices of P; see Fig. 2. Since G is connected and P is not a Hamilton path there is a vertex r outside P that is adjacent to a vertex v in P. Thus, we can extend P by constructing the path P' obtained by adding rv to the cycle and removing any other edge incident to v.

If condition (iii) is violated we find an edge between two vertices in $N^-(r)$ and extend P as described in Fig. 3.

If condition (iv) is violated then we have two cases to consider: either there are two consecutive vertices on P that are both adjacent to vertices in the subgraph $G-P$, or $V-P$ is not an independent set.

In the first case we identify the two vertices v_i and v_{i+1} that are both adjacent to vertices outside P. If they are both adjacent to the same vertex $r \in V-P$ then we directly add this new vertex to P obtaining the longer path $[v_0\ldots v_irv_{i+1}\ldots v_t]$. If they are adjacent to different vertices in $V-P$, we construct the cycle C containing all but one vertex of P as described in Fig. 4. Let v denote the excluded vertex. Note that $v_iv_{i+1} \in E(C)$ (otherwise either $v_i = v$ or $v_{i+1} = v$; but $v \in N^-(v_0)$ and such a vertex is not adjacent to vertices in $V-P$, by condition (i)). Assume that $v_ir_1 \in E$ and $v_{i+1}r_2 \in E$, where $r_1, r_2 \notin P$. By removing v_iv_{i+1} from C and adding r_1v_i and $v_{i+1}r_2$ to C, we create a new path of size $|P|+1$, with end points r_1 and r_2.

For the second case, we observe that if $V-P$ is not an independent set then we can construct a path P' in $G-P$ containing at least two vertices, which by the connectivity of G can be connected to the cycle C described above. In this way we create a new path with size at least $|P|+1$.

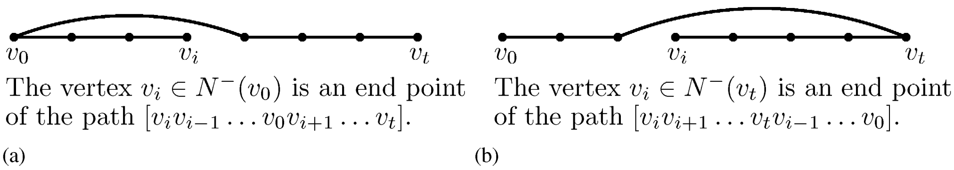

Fig. 1. Potential end points in a path.

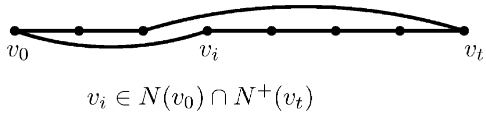

Fig. 2. If $N(v_0) \cap N^+(v_t) \neq \emptyset$ then there is a cycle in G that contains all vertices in P.

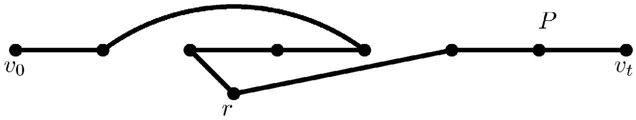

Fig. 3. If the left neighborhood on P of a vertex r outside P is not an independent set then P can be extended to include r.

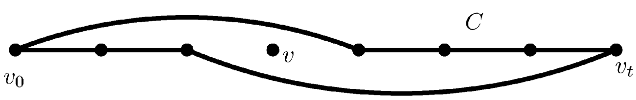

Fig. 4. The cycle C includes all vertices of the path $[v_0 \dots v_t]$ except $v \in N^-(v_0) \cap N^+(v_t)$.

A careful analysis of the violation checks above shows that an algorithm to find a maximal path can be implemented to run in $O(n^3)$ time. Thus, we have proved

Theorem 3. *A maximal path in a connected graph can be found in* $O(n^3)$ *time.*

Let P denote a maximal path in G according to Definition 1, with $P = [v_0 v_1 \dots v_{t-1} v_t]$ and let $R = V - P$ denote the vertices of G outside P.

Recall that Definition 1 includes two additional conditions if the set $N^-(v_0) \cap N^+(v_t)$ is non-empty. We start considering the other case, i.e., $N^-(v_0) \cap N^+(v_t) = \emptyset$.

Lemma 1. *If P is maximal then either $N^-(v_0) \cap N^+(v_t) \neq \emptyset$ or there is a spanning spider in G whose center is adjacent to all vertices outside P.*

Proof. We prove the equivalent statement that if P is maximal and $N^-(v_0) \cap N^+(v_t) = \emptyset$ then there is a vertex in P that is adjacent to all vertices in R. Thus, assume that P is maximal and that $N^-(v_0) \cap N^+(v_t) = \emptyset$. Let $X = N^-(v_0) \cup N^+(v_t)$. Take an arbitrary vertex $r \in V - P$. From condition (i) of Definition 1 it follows that r, v_0, and v_t are independent. Since $N^-(v_0) \cap N^+(v_t) = \emptyset$,

$$d(r) + |X| = d(r) + d(v_0) + d(v_t) \geqslant n - 1. \tag{2}$$

By condition (i) of Definition 1, we get that r is adjacent only to vertices in $V - X$. Since $r \notin P$, r is adjacent to at most $n - |P| - 1$ vertices in $G - P$. The remaining edges from r are adjacent to vertices in $P - X$. The number of these edges is

$$d_{P-X}(r) \geqslant d(r) - n + |P| + 1.$$

By (2) we get

$$d_{P-X}(r) \geqslant n - 1 - |X| - n + |P| + 1$$
$$= |P - X|.$$

That is, r is adjacent to all vertices in $P - X$, and since r was chosen arbitrarily from R, it follows that any vertex in $P - X$ is adjacent to all vertices in R, and is hence the center of a spanning spider in G. □

Assume from now on that

$$N^-(v_0) \cap N^+(v_t) \neq \emptyset.$$

This implies that conditions (iv) (a) and (b) of Definition 1 hold.

We will give an algorithm proving the following theorem. Later we will extend it to the general case of Theorem 2.

Theorem 4. *Any connected graph G with $\delta(G) \geqslant (n-1)/3$ contains a spanning spider. Furthermore, there is an $O(n^3)$ time algorithm that finds a spanning spider in G.*

The following lemma gives Theorem 4 when the size of R is small.

Lemma 2. *If $|R| \leqslant 2$ then G contains a spanning spider.*

Proof. If R is empty then P is a Hamilton path.

If $|R| = 1$ then by the connectivity of G, the vertex in R is adjacent to a vertex in P, yielding a spanning spider of G.

If R contains two vertices r_1 and r_2, and if both r_1 and r_2 are neighbors of $v_i \in P$, then $r_1 v_i$ and $r_2 v_i$ together with P form a spanning spider, centered at v_i. Thus, to prove that there exists a spanning spider in G it is sufficient to prove that $N(r_1) \cap N(r_2) \neq \emptyset$. For a contradiction, assume that $N(r_1) \cap N(r_2) = \emptyset$. By Definition 1, condition (iv) point (a), $N(r_1) \cap N^-(r_2) = \emptyset$. Hence, $N(r_1) \subseteq V - R - (N(r_2) \cup N^-(r_2))$ and the size of $N(r_1)$ is

$$|N(r_1)| \leqslant n - |R| - |N(r_2) \cup N^-(r_2)|.$$

By applying condition iv(a) again, we get that $N(r_2) \cap N^-(r_2) = \emptyset$ implying that $|N(r_2) \cup N^-(r_2)| = |N(r_2)| + |N^-(r_2)|$. It follows that:

$$|N(r_1)| \leqslant n - |R| - |N(r_2)| - |N^-(r_2)|$$
$$\leqslant n - 2 - 2(n-1)/3 = (n-5)/3$$
$$< (n-1)/3,$$

contradicting the degree condition of G, i.e., that $\delta(G) \geqslant (n-1)/3$. □

Assume now that $|R| \geqslant 3$, with $R = \{r_1, r_2, \ldots, r_{|R|}\}$, and let r^* denote an arbitrary vertex in R. In order to prove Theorem 4, we construct a spanning spider out of the maximal path P. First we need to find a suitable center for the spider. It turns out that a convenient property of such a center is to be adjacent to many independent vertices which in turn are independent of R.

Lemma 3. *The set $N^-(r^*) \cup R$ is independent, with size $|R| + (n-1)/3$.*

$$\frac{(n-1)}{6} + \frac{3|R|-1}{4}.$$

Proof. The independence is given by Definition 1 as follows. If $r \in R$ and $v \in N^-(r^*)$ then $rv \notin E$ by condition (iv) point (a). R is an independent set by condition (iv) point (b). Left is to prove that $N^-(r^*)$ is independent, but this follows from condition (iii). The size of the union follows from the degree condition on r^*, and the fact that R and $N^-(r^*)$ are disjoint.

For the second part of the proof, consider the vertices in $N^-(r^*) \cup R$. Each of them is adjacent only to vertices in $P - N^-(r^*)$, since $N^-(r^*) \cup R$ is an independent set and $R \cap P = \emptyset$. By the pigeonhole principle there exists a vertex

Table 1
The spider construction algorithm for general graphs

Algorithm. Spider construction in general graphs
  Input: A graph $G=(V,E)$, a maximal path P, and a vertex v_i satisfying the condition of Lemma 3.
  Output: A spider S, centered at v_i, and a tail T, that collectively span P and a portion of R.

1 Initially let $S := P$.
2 **For** each $r \in R$ such that $v_i r \in E$: add the edge $v_i r$ to S.
3 **If** all $r \in R$ are adjacent to v_i: **return** the spanning spider S.
  Otherwise,
4 **For** each $v_j \in P$ such that both $v_{j-1}v_i$ and $v_j r^*$ are in E:
  remove $v_{j-1}v_j$ from S, and
  add the edge $v_i v_{j-1}$ to S.
5 **If** there is an edge $v_i v_j \in S$ with $j > i+1$:
  remove the edge $v_i v_{i+1}$ from S (recall that v_i is the center of the spider).
6 **Return** the spider S and the tail $T := P - S$.
End Spider construction in general graphs.

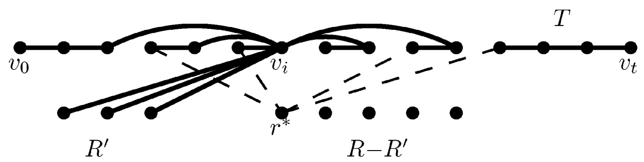

Fig. 5. The spider S, the tail T and the set $R - R'$, after the spider construction algorithm.

$v_i \in P - N^-(r^*)$ adjacent to at least

$$\frac{\frac{n-1}{3}|N^-(r^*) \cup R|}{|P - N^-(r^*)|} = \frac{\frac{n-1}{3}\left(\frac{n-1}{3} + |R|\right)}{n - \frac{n-1}{3} - |R|} \tag{3}$$

$$= \frac{\frac{n-1}{3}\left(\frac{n-1}{3} - \frac{|R|-1}{2} + \frac{3|R|-1}{2}\right)}{2\left(\frac{n-1}{3} - \frac{|R|-1}{2}\right)}$$

$$\geqslant \frac{n-1}{6} + \frac{3|R|-1}{4} \tag{4}$$

vertices in $N^-(r^*) \cup R$. □

Let v_i be a vertex in $P - N^-(r^*)$ satisfying the condition given in Lemma 3.[1] Let Δ be the number of vertices in $N^-(r^*) \cup R$ adjacent to v_i, i.e.,

$$\Delta \geqslant \frac{n-1}{6} + \frac{3|R|-1}{4}. \tag{5}$$

Using the algorithm in Table 1 we construct a spider S, centered at v_i, with branches beginning at vertices in $N^-(r^*)$ and ending at vertices in $N(r^*)$. Note that S fails to include the tail of P. We let T denote this tail; see Fig. 5.

Let L denote the leaves in S and let $R' = S - P - r^* \subset R$. We note that the number of leaves in S is at least $\Delta + 1$ but more importantly, the number of leaves adjacent to r^* is

$$d_L(r^*) \geqslant \Delta - |R'| - 2. \tag{6}$$

To see this, note first of all that r^* is not adjacent to any leaf that belongs to R'. Secondly, the tail T is not in S, but contains exactly one vertex in $N(v_i)$ that also lies in $N^-(r^*)$. Finally, if v_i is adjacent to r^*, then r^* is itself a leaf in S, but is of course not adjacent to itself.

[1] Notice that $i < t$, since by condition (i) of Definition 1, $(N^-(r^*) \cup R) \cap N(v_t) = \emptyset$.

If there is a matching between the vertices in $R-R'$ and $L-R'$, then we can construct a spider covering G. Next we prove that there is such a matching. A vertex v in $S\cup T$ is called an internal vertex if $v\notin L$. We let I denote the set of internal vertices.

Lemma 4. *There exists a matching between $R-R'$ and $L-R'$.*

Proof. Since r^* is adjacent to more than $|R-R'|$ leaves in S, it suffices to show that there is a matching between $R-R'-\{r^*\}$ and $L-R'$. Let r denote an arbitrary vertex in $R-R'-\{r^*\}$. By definition,

$$d(r)=d_I(r)+d_L(r). \tag{7}$$

Since r is not adjacent to v_i, and v_{i+1} is a leaf by construction, neither v_i nor v_{i+1} is counted in $d_I(r)$. Neither are they counted in $d_L(r^*)$. This time, v_i is not counted, since it is not a leaf, and v_{i+1} is not counted because $v_i\notin N^-(r^*)$.

Therefore,

$$d_I(r)+d_L(r^*)\leqslant(|P|-2)/2,$$

since r and r^* cannot be adjacent to v_0 or v_t (Definition 1, condition (ii)), nor to consecutive vertices on P (Definition 1, point (a) of condition (iv)). Hence,

$$d_L(r)\geqslant d(r)+d_L(r^*)-(|P|-2)/2. \tag{8}$$

Recalling that $d_L(r^*)\geqslant \Delta-|R'|-2$ (by (6)) and that $|P|=n-|R|$, by using (8) we get

$$d_L(r)\geqslant\frac{n-1}{3}+(\Delta-|R'|-2)-\frac{n-|R|-2}{2}. \tag{9}$$

By using (5) we obtain

$$\begin{aligned} d_L(r)&\geqslant\frac{n-1}{3}+\frac{n-1}{6}+\frac{3|R|-1}{4}-|R'|-2-\frac{n-|R|-2}{2}\\ &=|R|-|R'|+(|R|-7)/4\\ &\geqslant|R-R'|-1. \end{aligned}$$

The last inequality holds since $|R|\geqslant 3$ by our assumption. Thus, each vertex in $R-R'-\{r^*\}$ is adjacent to at least $|R-R'|-1$ leaves in S, so there exists a matching between $R-R'-\{r^*\}$ and $L-R'$. $\square$

Given the above guarantee of a matching we construct the spider as follows. Compute a matching between $R-R'$ and L. This gives us a new spider S' that contains all vertices except the tail T. The head of the tail is adjacent to r^*, and r^* is a leaf in S'. Add to S' the edge between r^* and the head of the tail to complete the spanning spider. This concludes the proof of Theorem 4.

Our main theorem follows easily from previous discussion:

Proof of Theorem 2. We begin with the following observation. In any independent set, there can be at most two vertices with degree less than $(n-1)/3$. This follows directly from the degree sum criteria. Thus, in the set R there are at most two vertices with degree less than $(n-1)/3$, call these r and r'. It is easy to modify any maximal path so to contain the eventual low degree vertices, i.e., every vertex in R has at least $(n-1)/3$ neighbors. The trick is to make these vertices part of the maximal path. We do this as follows. Consider a maximal path $P=[v_1\dots v_t]$ and its corresponding set $N^-(v_0)\cap N^+(v_t)$. This set is nonempty by Lemma 1. Hence, there is a cycle containing all vertices except one, call this vertex v. The excluded vertex is independent from R, being a potential end point of the maximal path. Hence we can create a new path $P'=[v_0'\dots v_t']$, replacing r by v in R. Note that r will be an end point of the path P'. At this point we need to check whether P' is maximal. If not, we recompute a maximal path out of P' and repeat the procedure.

Now, reconsider the sets $N^-(v_0)$ and $N^+(v_t)$ of the new path. Again, the intersection is non-empty. Thus, we can repeat the above procedure, replacing $r'\in R$ with a vertex v' in the path. Note that v' is independent from r and r', and has therefore degree higher than $(n-1)/3$.

Given this new maximal path we proceed as before. The existence of two vertices with degree less than $(n-1)/3$ makes the counting argument stronger. $\square$

4. Claw-free graphs

We now return to density conditions. In the case of Hamilton paths, in addition to density conditions on graphs in general, research has focused on claw-free graphs, where a weaker density condition is sufficient to assure Hamiltonicity:

Theorem 5 (Liu et al. [17], Mathews and Sumner [19]). *Let G be a connected graph without an induced $K_{1,3}$.*

If $\delta(G) \geqslant (n-2)/3$ (or, more generally, if $\delta_3(G) \geqslant n-2$), then $s(G)=0$ (in other words, then G has a Hamilton path).

In this context we were able to prove the following full generalization to spanning trees with at most k branch vertices:

Theorem 6. *Let G be a connected graph without an induced $K_{1,3}$, and let k be a nonnegative integer.*

If $\delta(G) \geqslant (n-k-2)/(k+3)$ (or, more generally, if $\delta_{k+3}(G) \geqslant n-k-2$), then $s(G) \leqslant k$.

In particular, we have the following corollary:

Corollary 1. *Let G be a connected graph without an induced $K_{1,3}$.*

If each vertex of G has degree at least $(n-3)/4$ (or, more generally, if $\delta_4(G) \geqslant n-3$), then G has a spanning spider.

Remark 1. In the case $k=1$ Theorem 6 assures the existence of a spanning spider in any claw free graph with $\delta_4(G) \geqslant n-3$. On the other hand, there is no new result of this type for arachnoid graphs, because the smallest lower bound on the degrees of a connected graph G without $K_{1,3}$ which would guarantee that G is arachnoid is $(n-2)/3$: The graph R_p obtained from a triangle abc by attaching a separate copy of K_p, $p \geqslant 3$, to each vertex a, b, c, is a $K_{1,3}$-free graph with minimum degree $(n-3)/3$, which is not arachnoid (all spanning spiders must have the center in the triangle abc). However, if all degrees of a connected graph G without $K_{1,3}$ are at least $(n-2)/3$, then already Theorem 5 implies that G has a Hamilton path (and hence is arachnoid).

We remark that Theorem 6 does not hold if $K_{1,3}$ is not excluded as an induced subgraph: For $k=0$ this is well-known, and easily seen by considering, say the complete bipartite graph $K_{p,p+2}$ ($p \geqslant 1$). For $k \geqslant 1$ we can take a path on $k+1$ vertices and attach a K_p ($p \geqslant 2$) to every vertex of the path. Moreover, we attach an extra K_p to the first and the last vertex of the path. The resulting graph is not $K_{1,3}$-free, and has no spanning tree with k branch vertices. However, the degree sum of any $k+3$ independent vertices in the graph is at least $n-k-1$, where $n=(k+3)p-2$.

We also note that the bound $n-k-2$ in the theorem is nearly best possible. For $k=0$, this is again well-known, and can be seen by considering, say, the above graph R_p. For $k \geqslant 1$, we consider following example. Take two copies of R_p, where $p=k+1$. Shrink one K_p back to a vertex in one of the two copies of R_p, and attach the vertex to a vertex of degree $p-1$ in the other copy of R_p. The resulting graph has four copies of K_p, each with $p-1$ vertices of degree $p-1$, and one copy K_p, denoted by K, with $p-2$ vertices of degree $p-1$. Take $k-1$ vertices of degree $p-1$ in K, and attach a copy of K_p to each. The resulting graph is $K_{1,3}$-free, and has no spanning tree with at most k branch vertices. The degree sum of any $k+3$ independent vertices is at least $n-k-5$, where $n=(k+3)p+2$.

Proof of Theorem 6. Let $T \subseteq G$ be a tree and let L be the set of all leaves and S be the set of all branch vertices of T. For every leaf $u \in L$, there is a unique branch vertex $s(u) \in S$ (closest to u). These two vertices are joined by a unique $s(u)-u$ path in T called the *branch of u*. The path joining two branch vertices of T is called a *trunk* of T. Note that a tree with ℓ branch vertices has at least $\ell+2$ branches (and hence at least $\ell+2$ leaves). For every internal vertex v on a branch, we denote by v^- and v^+ the predecessor and the successor of v on the branch, respectively. Similarly, by $s(u)^+$ and u^- we denote the successor of $s(u)$ and the predecessor of u on the branch of u, respectively. Note that it is possible to have $s(u)^+ = u$ and/or $s(u)=s(v)$ for some $u, v \in L$.

Suppose that G is a counterexample to the theorem. Let T be a tree of G with at most k branch vertices, and therefore, not spanning G. We assume that T is chosen so that it contains the greatest number of vertices, and, subject to this condition, so that

(i) the sum of the degrees of the branch vertices of T is as small as possible; and
(ii) subject to (i), the sum, over all leaves u of T, of the distance in T between u and $s(u)$ is as small as possible; and
(iii) subject to (i) and (ii), the length of the longest branch of T is as large as possible, and, for each $i=2,3,\ldots,$ the length of the ith longest branch is as large as possible subject to (i) and (ii), having the longer branches as long as possible.

Since G is a counterexample to the theorem, we can assume that the number of branch vertices in T is exactly k and that $G - T \neq \emptyset$.

The assumptions imply certain properties of T, which we now explore. Assume that u and v are any two distinct leaves of T $(u, v \in L)$ and r is any vertex not in T.

Fact 1. *The vertex u cannot be adjacent to $s(v)^+$.*

Otherwise, the tree $T + us(v)^+ - s(v)s(v)^+$ violates (i).

Fact 2. *The vertex u cannot be adjacent to two consecutive vertices on a trunk.*

Indeed, if a, b are such vertices, then $T + ua + ub - ab - uu^-$ violates (ii).

Fact 3. *If the branch of v is at least as long as the branch of u, then v cannot be adjacent to any vertex x $(\neq s(u))$ on the branch of u.*

Otherwise, $T + vx - xx^-$ violates (iii).

Fact 4. *If the branch of v is shorter than the branch of u, then v cannot be adjacent to two consecutive vertices on the branch of u.*

Indeed, if x, y are such consecutive vertices, then the tree $T + vx + vy - xy - vv^-$ violates (iii).

Fact 5. *The vertex v cannot be adjacent to an internal vertex x on the branch of u such that x^- is adjacent to u.*

Otherwise, the tree $T + x^-u + vx - x^-x - s(v)s(v)^+$ violates (i).

Fact 6. *The vertex r cannot be adjacent to any leaf or branch vertex of T, or to any two consecutive vertices of T.*

This follows by the maximality of T.

Fact 7. *The vertex r cannot be adjacent to an internal vertex x on the branch of u such that x^- is adjacent to u.*

Otherwise, the tree $T + x^-u + rx - xx^-$ is larger than T, thus violating the initial assumption.
We now fix a vertex r outside of T, and make the following claims:

Claim 1. *The set $L + r$ is independent.*

Fact 3 implies that no two leaves can be adjacent, and Fact 6 implies that r cannot be adjacent to a leaf.

Claim 2. *No vertex of T is adjacent to two distinct leaves of T.*

Suppose x is adjacent to both leaves u and v. Then x cannot be a vertex of a nontrivial trunk of T, because Fact 2 would imply that u, v, x, and a trunk neighbor of x induce a $K_{1,3}$. Similarly, x cannot be an internal vertex of a branch of T, other than the branch of u or v, because Facts 3 and 4 would imply that u, v, x, and a branch neighbor of x induce a $K_{1,3}$. Finally, x cannot be an internal vertex of the branch of u or v, because Facts 3–5 would imply that u, v, x, x^- induce a $K_{1,3}$. Thus x must be the unique branch vertex of T, and it is again easy to see that Fact 1 implies that u, v, x, and a neighbor of x (on a third branch/trunk) induce a $K_{1,3}$.

Claim 3. *No vertex of T is adjacent to both r and a leaf of T.*

This follows in the same manner as Claim 2, substituting as necessary Facts 6 and 7 for Facts 2–5.

Claim 4. *Each vertex of G is adjacent to at most one vertex in $L + r$.*

This follows from Claims 2 and 3, and the fact that no vertex outside of T can be adjacent to a leaf of T, by the maximality of T.

The set L has at least $k+2$ vertices, since T has exactly k branch vertices. Let I be any subset of $L \cup r$, with exactly $k+3$ vertices. Then I is an independent set with $k+3$ vertices, and the sum of their degrees is at most $n-k-3$, contrary to the assumption. □

As in the previous section, it is not difficult to turn the above proof into a polynomial time algorithm to actually find a spanning tree of G with at most k branch vertices. (The obvious approach yields an $O(n^4)$ algorithm.)

Using the same methods we can prove a similar result for $K_{1,4}$-free graphs. (For the existence of Hamilton paths and cycles, such results can be found in [4,18].) Naturally, there is a tradeoff between this weaker assumption and the minimum degrees in G one has to assume. We only state it here for $k \geqslant 1$, the main emphasis of our paper:

Theorem 7. *Let G be a connected graph without an induced $K_{1,4}$, and let k be a positive integer.*
If $\delta(G) \geqslant (n+3k+1)/(k+3)$ (or, more generally, if $\delta_{k+3}(G) \geqslant n+3k+1$), then $s(G) \leqslant k$.

Proof. The proof is along the same lines as the proof of Theorem 6; we only point out the differences.

Three claims are modified to obtain the following weaker statements: *Each branch vertex is adjacent to at most three leaves of T; At most one internal vertex of a branch of u is a common neighbor of u and another leaf v of T. In that case, the branch of u must be longer than the branch of v; At most one internal vertex of a branch of u is a common neighbor of u and r.* Using the modified set of claims we can show that there exists an independent set I with $k+3$ vertices and degree sum at most $n-(k+3)+2k+(k+1)+(k+2)=n+3k$, a contradiction. □

5. Relation to other problems

In this section we relate our parameter $s(G)$ to other classical graph theoretic parameters.

First we make some additional remarks about arachnoid graphs.

Proposition 5. *If G is an arachnoid graph, then, for any set S of vertices, the graph $G-S$ has at most $|S|+1$ components.*

Proof. If the deletion of S leaves at least $|S|+2$ components, then no spider centered in one of the components can be spanning. □

The condition in the proposition is a well-known necessary condition for a graph to have a Hamilton path. Recall that we have also observed in Section 4, that we do not have a density condition which implies that a graph is arachnoid, without also implying that it has a Hamilton path. Thus we are led to ask whether or not every arachnoid graph must have a Hamilton path. This is, in fact, not the case, but examples are not easy to find. One can, for instance, take a *hypotraceable graph* G, that is a graph which does not have a Hamilton path, but such that for each vertex v, the graph $G-v$ has a Hamilton path. Hypotraceable graphs are constructed in [22,23].

Proposition 6. *Every hypotraceable graph is arachnoid.*

Proof. For any vertex x, consider the Hamilton path in $G-v$, where v is any neighbor of x. Adding the edge xv yields a spanning spider of G centered at x. □

We note that we only know two types of arachnoid graphs: There are the graphs with a Hamilton path (sometimes called *traceable* graphs), and then there are the hypotraceable graphs. Moreover, the traceable graphs have at each vertex a spider with at most two branches, and the hypotraceable graphs have at each vertex a spider with at most three branches (one consisting of only one edge, cf. the proof above). It would be interesting to construct arachnoid graphs which are neither traceable nor hypotraceable. In the same vein, it would be interesting to construct arachnoid graphs in which some vertex is center only to spiders with more than three branches.

The following observation shows the relationship between path coverings and $s(G)$.

Proposition 7. *If G is a connected graph whose vertices can be covered by k disjoint paths, then G has a spanning tree with at most $2k-2$ branch vertices, i.e. $s(G) \leqslant 2k-2$.*

Let $\alpha(G)$ denote the *independence number* of G, i.e. maximum size of an independent set of vertices in G, and let $\kappa(G)$ denote the *connectivity* of G, i.e. minimum number of vertices removal of which disconnects G or results in the empty graph. Chvátal and Erdős [5] proved that vertices of any graph G can be covered by at most $\lceil \alpha(G)/\kappa(G) \rceil$ vertex disjoint paths. Using the previous proposition we have

Theorem 8. *Let G be a connected graph. Then $s(G) \leqslant 2\lceil \alpha(G)/\kappa(G) \rceil - 2$.*

Thus, for 1-connected graphs, previous theorem gives $s(G) \leqslant 2\alpha(G) - 2$. One may in fact do a little bit better:

Proposition 8. *Let $G = (V, E)$ be a connected graph that is not complete. Then $s(G) \leqslant \alpha(G) - 2$, and a spanning tree with at most $\alpha(G) - 2$ branch vertices can be found in $O(|V| + |E|)$ time.*

Proof. Consider a depth first search tree T of G. This is a spanning tree where the set of all leaves, save possibly the root, forms an independent set. If either the root r of T is a branch vertex or r is independent from the other leaves, then G has a spanning tree with at most $\alpha(G)$ leaves and hence, at most $\alpha(G) - 2$ branch vertices.

In case the root r has degree one and it is adjacent to another leaf u, then we can remove in T the edge $s(u)s(u)^+$ and add the edge ru (recall that $s(u)$ is the branch vertex closest to u in T and $s(u)^+$ is its successor on the path from $s(u)$ to u in T). Again, we get a spanning tree in which the number of leaves is at most $\alpha(G)$, thus implying that the number of branch vertices is at most $\alpha(G) - 2$. $\square$

A caterpillar in which all branch vertices are of degree 3 shows that Proposition 8 is the best possible.

Acknowledgements

Our thanks to Olivier Cogis for interesting discussions and suggestions, and to Peter Damaschke for correcting our proof of Proposition 2. We also thank Bruno Beauquier for his interest and inspiration.

References

[1] M. Aung, A. Kyaw, Maximal trees with bounded maximum degree in a graph, Graphs Combin. 14 (1998) 209–221.
[2] C. Bazgan, M. Santha, Z. Tuza, On the approximation of finding a(nother) Hamiltonian cycle in cubic Hamiltonian graphs, J. Algorithms 31 (1) (1999) 249–268.
[3] B. Beauquier, J.-C. Bermond, L. Gargano, P. Hell, S. Perennes, U. Vaccaro, Graph problems arising from wavelength-routing in all-optical networks, Proceedings of WOCS, Geneve, Switzerland, 1997.
[4] G. Chen, R.H. Schelp, Hamiltonicity for $K_{1,r}$-free graphs, J. Graph Theory 20 (4) (1995) 423–439.
[5] V. Chvátal, P. Erdős, A note on Hamiltonian circuits, Discrete Math. 2 (1972) 111–113.
[6] G.A. Dirac, Some theorems on abstract graphs, Proc. London Math. Soc. (3) 2 (1952) 69–81.
[7] T. Feder, R. Motwani, C. Subi, Finding long paths and cycles in sparse Hamiltonian graphs, Proceedings of the ACM Symposium on Theory of Computing, STOC'00, 2000.
[8] L. Gargano, M. Hammar, There are spanning spiders in dense graphs (and we know how to find them), ICALP'03, June 30–July 04, 2002, Eindhoven, The Netherlands.
[9] L. Gargano, P. Hell, L. Stacho, U. Vaccaro, Spanning trees with bounded number of branch vertices, ICALP'02, July 8–13, Málaga, Spain, 2002.
[10] L. Gargano, U. Vaccaro, Routing in all-optical networks: algorithmic and graph-theoretic problems, in: I. Althofer, et al. (Eds.), Numbers, Information and Complexity, Kluwer Academic Publisher, Dordrecht, 2000, pp. 555–578.
[12] R.M. Karp, Reducibility among combinatorial problems, in: Complexity of Computer Computations (Proceedings of the Symposium, IBM Thomas J. Watson Research Center, Yorktown Heights, N.Y., 1972), Plenum, New York, 1972, pp. 85–103.
[13] J. Könemann, R. Ravi, A matter of degree: improved approximation algorithms for degree-bounded minimum spanning trees, Proceedings of ACM Symposium on Theory of Computing, STOC'00, 2000.
[14] D. Krager, R. Motwani, D.S. Ramkumar, On approximating the longest path in a graph, Algorithmica 18 (1997) 82–98.
[15] A. Kyaw, A sufficient condition for a graph to have a k tree, Graphs Combin. 17 (2001) 113–121.
[16] M. Las Vergnas, Sur une proprieté des arbres maximaux dans un graphe, C. R. Acad. Sci. Paris, Ser. A 271 (1971) 1297–1300.
[17] Y. Liu, F. Tian, Z. Wu, Some results on longest paths and cycles in $K_{1,3}$-free graphs, J. Changsha Railway Inst. 4 (1986) 105–106.
[18] L.R. Markus, Hamiltonian results in $K_{1,r}$-free graphs, in: Proceedings of the 24th Southeastern International Conference on Combinatorics, Graph Theory, and Computing, Vol. 98, Boca Raton, Florida, 1993, pp. 143–149.

[19] M.M. Matthews, D.P. Sumner, Longest paths and cycles in $K_{1,3}$-free graphs, J. Graph Theory 9 (2) (1985) 269–277.
[20] O. Ore, A note on Hamilton circuits, Am. Math. Monthly 67 (1960) 55.
[21] B. Raghavachari, Algorithms for finding low-degree structures, in: D.S. Hochbaum (Ed.), Approximation Algorithms for NP-Hard Problems, PWS Publishing Company, Boston, 1997, pp. 266–295.
[22] C. Thomassen, Hypohamiltonian and hypotraceable graphs, Discrete Math. 9 (1974) 91–96.
[23] C. Thomassen, Planar cubic hypo-Hamiltonian and hypotraceable graphs, J. Combin. Theory Ser. B 30 (1) (1981) 36–44.