ELSEVIER

Discrete Applied Mathematics 95 (1999) 241–249

DISCRETE APPLIED MATHEMATICS

Note on a helicopter search problem on graphs

Fedor V. Fomin *

Faculty of Mathematics and Mechanics, St. Petersburg State University, Bibliotechnaya sq.2, St. Petersburg 198904, Russia

Received 31 October 1997; revised 6 April 1998; accepted 23 December 1998

Abstract

We consider a search game on a graph in which one cop in a helicopter flying from vertex to vertex tries to catch the invisible robber. The existence of the winning program for the cop in this problem depends only on the robber's speed. We investigate the problem of finding the minimal robber's speed which prevents the cop from winning. For this parameter we give tight bounds in terms of the linkage and the pathwidth of a graph.

Keywords: Searching a graph; Pathwidth; Linkage

1. Introduction

Here is a search game, played on a finite, undirected topological graph G, that is embedded in a Euclidean space (dimension of this space is not important for us). $V(G)$ is the vertex and $E(G)$ is the edge set of G. In this paper we shall assume that edges of a graph are one unit long. Also we consider only connected graphs with at least two vertices and without multiply edges or loops.

Two players called *Cop* and *Robber* are on G. Cop tries to find Robber, and Robber tries to evade. Cop's actions are defined by a finite sequence of steps called *search program Π*. In the first step, Cop occupies some vertex of G. In each of the following steps, Cop moves (flies by helicopter) to some vertex (not necessarily adjacent to the occupied vertex) of G. So the search program Π is a mapping

$$\Pi : \{1,2,\dots,T\} \to V(G),$$

where $\Pi(i), i \in \{1,\dots,T\}$, is the vertex occupied by Cop in the ith step.

* Correspondence address. Department of Operations Research, St. Petersburg State University, Bibliotechnaya sq.2, St. Petersburg 198904, Russia.

E-mail address: fvf@brdr.usr.pu.ru (F.V. Fomin)

PII: S0166-218X(99)00078-5

A continuous mapping

$$y : [0, T] \to G$$

is interpreted as a trajectory of Robber. We shall suppose that the Robber's speed is restricted by the constant $\mu > 0$, i.e. for any $t_1, t_2 \in [0, T]$, $t_1 \neq t_2$,

$$\left| \frac{\rho(y(t_1), y(t_2))}{t_1 - t_2} \right| \leqslant \mu,$$

where $\rho(y(t_1), y(t_2))$ is the length (in the Euclidean metric) of the shortest curve in G that connects $y(t_1)$, $y(t_2)$. Thus Robber cannot leave G, and can overcome a distance of no more than μ with every step of Cop. When μ^{-1} is an integer then one can interpret μ^{-1} as the number of Cop's steps during which Robber covers an edge.

Cop *finds* Robber in the ith step if and only if there exists $j \in \{1, \dots, i\}$ such that $\rho(\Pi(j), y(j)) < 1$. Loosely speaking, Cop positioned in any vertex 'oversees' all incident edges but he cannot see Robber positioned in adjacent vertices.

The search program $\Pi(i)$, $i \in \{1, \dots, T\}$, is *winning* if for any trajectory of Robber $y(t)$, $t \in [0, T]$, there exists $i \in \{1, \dots, T\}$, such that in the ith step Robber is found.

The existence of the winning program for Cop in this problem depends only on the constant μ. For a graph G we consider the parameter

$$\mu(G) = \inf\{\mu\text{: with } \mu \text{ Cop has no winning program on } G\}.$$

Obviously $\mu(G)$ is at most one. The problem of computing $\mu(G)$ is called *the helicopter search problem*. Note that $\mu(G)$ is a combinatorial invariant.

The author studied in [3] two cases of the helicopter search problem. In the first one Cop can visit each vertex of a graph only once. In the second case Cop cannot afford 'recontamination' of vertices. In the first case the problem of finding the minimal Robber's speed is equivalent to the bandwidth minimization problem and in the second case the problem is equivalent to the natural generalization of the bandwidth problem and is closely approximated by the pathwidth. So it is natural to investigate the case when recontamination is allowed. We think that this case is much harder than 'recontamination-free' ones but for some graphs the solution of the helicopter search problem is easy.

To warm up, let us determine $\mu(G)$ of a path $v_1, v_2, \dots, v_n$ on n vertices. Clearly, $\mu(G) = 1$ because for any $\mu < 1$ Cop has the following winning program on G: for $i \in \{1, \dots, n\}$ $\Pi(i) = v_i$. As another example, if G is a cycle $v_1, v_2, \dots, v_n, v_1$ then $\mu(G)$ is also equal to 1 because for any $\mu < 1$ Cop also has a winning program: in order to win on G he 'runs' the cycle $\lceil n - 2/1 - \mu \rceil$ times.

2. Lower and upper bounds

In this section we find lower and upper bounds for $\mu(G)$. Note that these bounds are tight.

2.1. Upper bound

The linkage (or the width or the colouring number; see [7,11] for further references and discussions) of a graph G, denoted by *linkage* (G), is the maximum min-degree of any subgraph of G. (We use the term *min-degree of a subgraph H of G* to denote the least degree of any of its vertices; the degree of a vertex is taken with respect to the subgraph.)

Theorem 1. *For any graph G, $\mu(G) \leqslant \lceil linkage(G)/2\rceil^{-1}$.*

Proof. Let $\Pi(i)$, $i \in \{1,\dots,T\}$, be a search program and H be a subgraph of G with the largest min-degree $d = linkage(G)$. Let us show how Robber with the speed of $\lceil d/2\rceil^{-1}$ can avoid Cop. We construct the Robber's trajectory inductively. To this end, suppose that for some $i \in \{1,\dots,T-\lceil d/2\rceil+1\}$, $y(i) = u \in V(H)$, and for each $j \in \{i,\dots,\min(i+\lceil d/2\rceil-1,T)\}$, $\Pi(j) \neq u$. Clearly for $i=1$ such u exists. We prove that there exists a moment $i' > i$ such that

(i) $y(i') = v \in V(H)$;
(ii) for each $j \in \{i',\dots,\min(i'+\lceil d/2\rceil-1,T)\}$, $\Pi(j) \neq v$;
(iii) for each $j \in \{i,\dots,i'\}$, $\rho(\Pi(j),y(j)) \geqslant 1$.

If u is not visited by Cop after the ith step then the proof is obvious. Let k be the minimal integer $\geqslant i+\lceil d/2\rceil$ such that $\Pi(k)=u$. If $k > i+\lceil d/2\rceil$, then we put $i' = k-\lceil d/2\rceil$ and Robber simply stays in u from i until i'.

Suppose that $k = i+\lceil d/2\rceil$. Since u is incident with $\geqslant d$ edges in H, there is an edge $(u,v) \in E(H)$ such that for each $j \in \{i+1,\dots,\min(i+d,T)\}$, $\Pi(j) \neq v$. Then Robber starts moving with the speed of $\lceil d/2\rceil^{-1}$ from u to v after the moment i. At the moment $i'=k$ he arrives at v and for each $j \in \{i,\dots,i'\}$, $\rho(\Pi(j),y(j)) \geqslant 1$. Since $2\lceil d/2\rceil - 1 \leqslant d$ then for each $j \in \{i',\dots,\min(i'+\lceil d/2\rceil-1,T)\}$, $\Pi(j) \neq v$. $\square$

Let $\chi(G)$ be the chromatic number of G. It is easy to check that $\chi(G)-1 \leqslant linkage(G)$ (see, e.g. [11]). Then Lemma 1 implies the following.

Corollary 2. *For any graph G, $\mu(G) \leqslant \lceil(\chi(G)-1)/2\rceil^{-1}$.*

2.2. Lower bound

A graph G is an *interval graph*, if and only if one can associate with each vertex $v \in V(G)$ an interval $I_v = [l(v), r(v)]$ on the real line, such that for all $v,w \in V(G)$, $v \neq w$: $(v,w) \in E(G)$, if and only if $I_v \cap I_w \neq \emptyset$. The set of intervals $\mathscr{I} = \{I_v\}_{v\in V(G)}$ is called an (interval) *representation* for G.

A graph G' is a *supergraph* of the graph G if $V(G') = V(G)$ and $E(G) \subseteq E(G')$.

The original definition of the pathwidth can be found in [10]. For our purposes, the following equivalent definition is more convenient (see [8]). The *pathwidth* of a graph

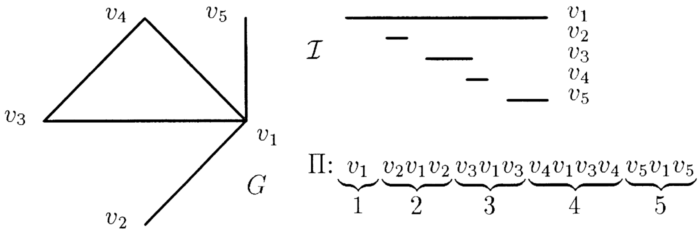

Fig. 1. Graph G, an interval representation $\mathscr{I}$ of G and the associated program Π.

G, denoted by $pw(G)$, is the smallest size of a max-clique over all interval supergraphs of G decreased by one.

It is well known that every interval graph has an interval representation in which the left endpoints are distinct integers $1,2,\ldots,|V(G)|$. Such a representation will be termed *canonical.*

Theorem 3. *For any graph G, $\mu(G)\geqslant 2/(pw(G)+1)$.*

Proof. It is clear that for any supergraph H of G, $\mu^{-1}(G)\leqslant\mu^{-1}(H)$ therefore w.l.o.g. we may assume that G is an interval graph. Let G be an interval graph on n vertices and $\mathscr{I}$ be a canonical representation of G. Let v_i be the vertex associated with an interval $[i,r(i)]\in\mathscr{I}$, $i\in\{1,\ldots,n\}$. Let $\delta(i)$ be the set of all vertices v_j, $j\leqslant i$, such that $r(j)\geqslant i$. Since G is an interval graph it is easy to see that for any $i\in\{1,\ldots,n\}$, $|\delta(i)|\leqslant pw(G)+1$.

Let us describe a search program $\Pi(i)$, $i\in\{1,\ldots,T\}$, where

$$T=\sum_{i\in\{1,\ldots,n\}}(|\delta(i)|+1)-1.$$

Π consists of n grandsteps. At the ith grandstep Cop first visits v_i and then vertices (if such vertices exist) from $\delta(i)$ in the increasing order (see Fig. 1). Thus during the ith grandstep, $i\geqslant 2$, Cop makes $|\delta(i)|+1$ steps.

We claim that if Robber's speed is less than $2/(pw(G)+1)$ then Π is the winning program. Suppose that there exists a trajectory of Robber $y(t)$, $t\in[0,T]$, such that for any $i\in\{1,\ldots,T\}$, $\rho(\Pi(i),y(i))\geqslant 1$. Since Cop visits all vertices of G, there exists a moment t such that at this moment Robber for the first time is in a vertex previously visited by Cop. Let v_i be such a vertex. Let $k\leqslant t$ be the maximal integer subject to $\Pi(k)=v_i$. Note that $t-k>(pw(G)+1)/2$. If Cop visits v_i after k then from the definition of Π, we have that there exists $l\in\{k+1,\ldots,k+pw(G)+1\}$ such that $\Pi(l)=v_i$. Since $l-t<(pw(G)+1)/2$ then Cop finds Robber at the moment l and we conclude that after the kth move Cop cannot visit v_i. Let v_j be the last vertex occupied by Robber before t. Note that v_j is adjacent to v_i.

Due to definitions of t and Π, we have that $j>i$ and $v_i\in\delta(j)$. Hence for some $m\in\{k+1,\ldots,k+pw(G)\}$, $\Pi(m)=v_j$. Robber can start moving from v_j to v_i only after moment k, so Cop finds Robber at the moment m.

This contradiction shows that Π is the winning program. $\square$

2.3. Examples

Let I be an interval graph. Since the linkage of I is at least the size of max-clique minus one then $linkage(I) \geqslant pw(I)$. From the other hand, it is known (see, e.g. [2]) that for any graph G, $linkage(G) \leqslant pw(G)$. Therefore, $linkage(I) \leqslant pw(I)$. As a consequence of Theorems 1 and 3 we have the following result.

Corollary 4. *Let k be the size of max-clique in I. If k is even, then $\mu(I) = \lceil linkage(I)/2 \rceil^{-1} = 2/(pw(I)+1) = 2/k$.*

Let K_n be a complete graph on n vertices.

Theorem 5. $\mu(K_n) = \lfloor n/2 \rfloor^{-1}$.

Proof. K_n is an interval graph and Corollary 4 implies the proof for even n.

Suppose that $n = 2\theta + 1$. Let $v_1, v_2, \ldots, v_n$ be vertices of K_n. We define a program $\Pi(t)$, $t \in [0, n(\theta+1)]$, as follows: for each $t \in \{1, \ldots, n(\theta+1)\}$, $\Pi(t) = v_s$, where $t \equiv s \pmod n$. Thus Π is a sequence of steps:

$$\underbrace{v_1, v_2, \ldots, v_n, v_1, v_2, \ldots, v_n, \ldots, v_1, v_2, \ldots, v_n}_{\theta+1 \text{ times}}.$$

For any vertices v_i, v_j let us define an 'oriented distance'

$$\vec{\rho}(v_i, v_j) = \begin{cases} j - i & \text{if } i \leqslant j, \\ n - i + j & \text{if } i > j. \end{cases}$$

In other words, $\vec{\rho}(v_i, v_j)$ is the number of edges of the directed path between v_i and v_j in the directed cycle $(v_1, v_2, \ldots, v_n, v_1)$. Note that $\vec{\rho}(v_i, v_j) + \vec{\rho}(v_j, v_i) = n$.

We prove the following assertion: if the speed of Robber is less than θ^{-1} then Π is the winning program.

For $k \in \{1, \ldots, T - 2\theta\}$ we denote by r_k the smallest $t \in [k, T]$ such that $y(t) \in V(G)$. The following two claims prove the assertion.

Claim 6. *If $\vec{\rho}(\Pi(k), y(r_k)) \leqslant \theta$ then Cop finds Robber at the moment $k + \theta$.*

Proof of claim. Obvious, since the speed of Robber is less than θ^{-1}. $\square$

Claim 7. *If $\vec{\rho}(\Pi(k), y(r_k)) = M > \theta$ then Cop finds Robber at the moment $k + \theta$ or there is $i \in \{k, \ldots, k+\theta\}$ such that $\vec{\rho}(\Pi(i+\theta), y(r_{i+\theta})) \leqslant M - 1$.*

Proof of claim. Let $y(r_k) = u$ and $y(r_{k+\theta}) = v$. If for each $i \in \{k, \ldots, k+\theta\}$ $v \neq \Pi(i)$ then $\vec{\rho}(\Pi(k+\theta), y(r_{k+\theta})) \leqslant \theta < M$. Suppose that for some $i \in \{k, \ldots, k+\theta\}$, $\Pi(i) = v$. Robber can start moving from u to v only after the moment i. Since Robber's speed is less than θ^{-1}, then he arrives at v only after the moment $i+\theta$. It means that $y(r_{i+\theta})=v$. At the moment $j \in \{k,\dots,k+2\theta\}$ such that $\Pi(j)=u$ Cop positioned in u 'oversees' the edge (u,v); hence $j-i=\vec{\rho}(v,u)$ is more than θ (or Cop finds Robber in the jth step). Thus $\vec{\rho}(u,v)=n-\vec{\rho}(v,u)<n-\theta=\theta+1\leqslant M-1$. $\square$

We proved that $\mu(K_n)\geqslant\lceil n/2\rceil^{-1}$. Since $linkage(K_n)=n-1$ then due to Theorem 1 $\mu(K_n)\leqslant\lceil (n-1)/2\rceil^{-1}=\lceil n/2\rceil^{-1}$. $\square$

3. Recontamination helps

One of the 'main' tools in 'traditional' graph-searching problems is the theorem of LaPaugh [9] which asserts that 'recontamination' does not help to search a graph (see [1,3,5] for further references on graph searching). In other words, excluding search strategies which give the fugitive the possibility of visiting an already searched vertex, does not increase the number of searchers. In the helicopter search problem the usage of the 'recontamination' helps Cop a lot. The author studied in [3] the monotone case of the helicopter search problem in which Cop cannot afford 'recontamination' of previously visited vertices. Let $\mu_{\mathrm{m}}(G)$ be the minimal Robber's speed such that Cop has no winning monotone program on graph G. It is proved [3] that for any graph G,

$$\frac{1}{pw(G)}\geqslant\mu_{\mathrm{m}}(G)\geqslant\frac{1}{pw(G)+1}. \tag{1}$$

Thus Theorem 3 implies the following result. If the pathwidth of a graph G is more than one, then 'recontamination' helps to search G. In this section (Corollary 10) we prove a somewhat stronger result.

A graph G' is called *a homeomorphic image* of a graph G if G' can be obtained from G by subdividing edges in G with an arbitrary number of degree two vertices.

Traditional characterization of an outerplanar graph is that it can be embedded in the plane such that all vertices are on the outer face boundary. The next Lemma follows directly from the definition of outerplanar graphs.

Lemma 8. *Let G be an outerplanar graph. Then there exists an ordering* $(v_1,v_2,\dots,v_n)$, $n=|V(G)|$, *of vertices of G such that for any* $1\leqslant i<k<j<l\leqslant n$, v_i *is adjacent to* v_j *only if* v_k *is not adjacent to* v_l.

Theorem 9. *For any outerplanar graph G there is a homeomorphic image G' of G such that* $\mu(G')\geqslant\frac{2}{3}$.

Proof. Let $v_1,v_2,\dots,v_n$ be a ordering as in Lemma 8. For $p=(u_i,v_j)\in E(G)$ we denote by $b(p)$ $(e(p))$ the smallest (the largest) number from $\{i,j\}$.

Let f be one-to-one mapping

$$f: E(G)\rightarrow\{1,\dots,|E(G)|\}$$

such that for any $p, q \in E(G)$, $f(p) < f(q)$ implies $b(p) \leqslant b(q) < e(q) \leqslant e(p)$ or $e(p) \leqslant b(q)$. Note that the existence of f is due to Lemma 8.

For $p \in E(G)$ we define the set of edges

$$E_p = \{q \in E(G)\colon\ f(q) < f(p) \text{ and } b(p) \leqslant b(q) < e(q) \leqslant e(p)\}.$$

Let G' be the graph obtained from G by replacing every edge $p \in E(G)$ by $l(p)$-edge path $S(p)$ on vertices $(v(p)_1, \ldots, v(p)_{l(p)+1})$, where

$$l(p) = \begin{cases} 1 & \text{if } E_p = \emptyset, \\ 4 \sum\limits_{q \in E_p} l(q) & \text{otherwise} \end{cases}$$

and $v(p)_1 = v_{b(p)}$, and $v(p)_{l(p)+1} = v_{e(p)}$.

To complete the proof we show how Cop can catch Robber on G' for $\mu < \frac{2}{3}$. For describing a winning program for Cop we need more definitions. We say that Cop makes *an increasing visiting-round* of a path $S(p)$ if he visits vertices of S_p in the following order:

$$v(p)_1, v(p)_2, v(p)_3, v(p)_1, v(p)_3, v(p)_4, \ldots, v(p)_1, v(p)_{i-1}, v(p)_i, \ldots,$$
$$v(p)_1, v(p)_{l(p)}, v(p)_{l(p)+1}.$$

We also say that Cop makes *a decreasing visiting-round* of $S(p)$ if he visits vertices as follows:

$$v(p)_{l(p)+1}, v(p)_1, v(p)_2, v(p)_{l(p)+1}, v(p)_2, v(p)_3, \ldots, v(p)_{l(p)+1}, v(p)_{i-1}, v(p)_i, \ldots,$$
$$v(p)_{l(p)+1}, v(p)_{l(p)}, v(p)_{l(p)+1}.$$

We say that Cop *works* on a vertex v_i if for every path $S(p)$ which ends in v_i $(v(p)_{l(p)+1} = v_i)$ he makes the decreasing visiting-round of $S(p)$ and for every path $S(p)$ which starts in v_i $(v(p)_1 = v_i)$, he makes the increasing visiting-round of $S(p)$. The order of visiting-rounds is as follows: first Cop makes decreasing visiting-rounds in decreasing order (if $f(p) < f(q)$ then $S(q)$ is visited before $S(p)$) and then makes increasing visiting-rounds in increasing order (if $f(p) < f(q)$ then $S(p)$ is visited before $S(q)$). Now we are ready to define a program $\Pi(i)$ with $i \in \{1, \ldots, T\}$,

$$[1, T] = \bigcup_{j \in \{1, \ldots, n\}} =[t_j, t_{j+1}]$$

such that from t_j until t_{j+1} Cop works on vertex v_j.

We show that Π is a winning program for $\mu < \frac{2}{3}$. Suppose that Π is not winning. Since $\mu < \frac{2}{3}$, then during Cop's work on a vertex v_k Robber cannot visit v_k. When working on v_i, Cop makes increasing visiting-rounds of every path $S(p)$ which starts in v_k; thus for every $p \in E(G)$, $b(p) = k$, there is $t \in [k, k+1]$ such that $y(t) \notin S(p)$.

We can conclude from this that to avoid Cop, Robber at least once must overcome a path $S(p)$ such that $v_i \in V(G)$, $i = b(p)$, was worked by Cop and $v_j \in V(G)$, $j = e(p)$ was not worked by Cop yet. Suppose that Robber is on $S(p)$ from t_-, $y(t_-) = v_j$ until t_+, $y(t_+) = v_i$.

Because after the t_ith step Cop makes the increasing visiting-round of $S(p)$, then $t_- \geqslant t_i$. Before the t_{j+1}th step Cop makes the decreasing visiting-round of $S(p)$; hence $t_+ \leqslant t_{j+1}$. To conclude a contradiction we shall show that in time $t_{j+1} - t_i \geqslant t_+ - t_-$ Robber cannot overcome a distance $l(p)$. We observe that $l(p) \neq 1$ because if this is not the case, $(v_i, v_j) \in E(G')$ and $t_{i+1} = t_j$.

Cop makes a visiting-round of a path $S(q)$ in at most $3l(q)$ steps. Every path is visited by Cop twice (the first time in increasing order and the second time in decreasing order). Thus

$$t_{j+1} - t_i \leqslant 6 \sum_{q \in E_p} l(q) = \frac{3}{2} l(p).$$

But the Robber's speed is less than $\frac{2}{3}$ and he cannot overcome the distance $l(p)$ in a time $t_+ - t_- \leqslant \frac{3}{2} l(p)$. □

Corollary 10. *For any $k > 0$ there is a graph G such that $\mu(G)/\mu_{\mathrm{m}}(G) \geqslant k$.*

Proof. Let k be an integer and let G be a tree with $pw(G) \geqslant \frac{3}{2}k$. For each k such a tree exists (see, e.g. [6]). For any homeomorphic image G' of a graph G, $pw(G) \leqslant pw(G')$ (see, e.g. [8]). Since every tree is outerplanar, then by Theorem 8 there is a homeomorphic image G' of G such that $\mu(G') \geqslant \frac{2}{3}$.

$$\frac{\mu(G)}{\mu_{\mathrm{m}}(G)} \geqslant \frac{2}{3} \frac{1}{\mu_{\mathrm{m}}(G)} \overset{\text{by (1)}}{\geqslant} \frac{2}{3} pw(G) \geqslant k. \qquad \square$$

4. Concluding remarks

From Theorem 5 we can conclude that for the graph of the tetrahedron $\mu(G) = \frac{1}{2}$. It is interesting to find the parameter μ for another Platonic graphs. The theorems proved above do not provide the exact value of μ. Thus, for example, for the graph of the cube, μ is between $\frac{2}{5}$ and $\frac{1}{2}$; for the graph of the icosahedron between $\frac{2}{7}$ and $\frac{1}{3}$ (see [4] for the solution of similar problems on the Platonic graphs).

Also, it is natural to ask, what about an analogue of Theorem 9 for planar graphs?

References

[1] D. Bienstock, Graph searching, path-width, tree-width and related problems (a survey), DIMACS Ser. Discrete Math. Theoret. Comput. Sci. 5 (1991) 33–49.

[2] N.D. Dendris, L.M. Kirousis, D.M. Thilikos, Fugitive-search games on graphs and related parameters, Theoret. Comput. Sci. 172 (1997) 233–254.

[3] F.V. Fomin, Helicopter search problems, bandwidth and pathwidth, Discrete Appl. Math. 85 (1998) 59–71.

[4] F.V. Fomin, P.A. Golovach, N.N. Petrov, Search problems on 1-skeletons of regular polyhedrons, Internat. J. Math., Game Theory Algebra 7 (1997) 101–111.

[5] F.V. Fomin, N.N. Petrov, Pursuit-evasion and search problems on graphs, Congr. Numer. 122 (1996) 47–58.

[6] L.M. Kirousis, C.H. Papadimitriou, Searching and pebbling, Theoret. Comput. Sci. 47 (1986) 205–218.
[7] L.M. Kirousis, D.M. Thilikos, The linkage of a graph, SIAM J. Comput. 25 (1996) 626–647.
[8] T. Kloks, Treewidth. Computations and Approximations, Lecture Notes in Computer Science, vol. 842, Springer, Berlin, 1994.
[9] A.S. LaPaugh, Recontamination does not help to search a graph, J. ACM 40 (1993) 224–245.
[10] N. Robertson, P.D. Seymour, Graph minors. I. Excluding a forest, J. Combin. Theory Ser. B 35 (1983) 39–61.
[11] B. Toft, Colouring, stable sets and perfect graphs, in: R.L. Graham, M. Grötschel, L. Lovász (Eds.), Handbook of Combinatorics, vol. 1, Elsevier, Amsterdam, 1995, pp. 233–288.