ELSEVIER
Discrete Applied Mathematics 72 (1997) 209–218
DISCRETE APPLIED MATHEMATICS

An algorithm for finding homogeneous pairs ☆

Hazel Everett [a], Sulamita Klein [b,*], Bruce Reed [c]

[a] Dep. d'informatique, Université du Québec à Montréal, Montréal, Québec, H3C 3P8 Canada
[b] Dep. Ciência da Computação, I.M., Universidade Federal do Rio de Janeiro, RJ 21944, Brazil
[c] Dept. of Mathematics, Carnegie Mellon University, Pittsburgh, PA 15213-3890, USA

Received 13 April 1994; revised 9 May 1995

Abstract

A homogeneous pair in a graph $G = (V,E)$ is a pair $\{Q_1, Q_2\}$ of disjoint sets of vertices in this graph such that every vertex of $V \setminus (Q_1 \cup Q_2)$ is adjacent either to all vertices of Q_1 or to none of the vertices of Q_1 and is adjacent either to all vertices of Q_2 or to none of the vertices of Q_2. Also $|Q_1| \geqslant 2$ or $|Q_2| \geqslant 2$ and $|V \setminus (Q_1 \cup Q_2)| \geqslant 2$. In this paper we present an $O(mn^3)$-time algorithm which determines whether a graph contains a homogeneous pair, and if possible finds one.

1. Introduction

The purpose of this paper is to present an algorithm to determine whether a graph G has a homogeneous pair and to find such a pair, if it exists, in polynomial time. We consider only finite undirected graphs $G = (V,E)$ with no loops or multiple edges, where V is the set of vertices of G and E is the set of edges of G.

A *homogeneous pair* in a graph G is a pair $\{Q_1, Q_2\}$ of disjoint sets of vertices in this graph such that:

- every vertex of $V \setminus (Q_1 \cup Q_2)$ is adjacent to either all vertices of Q_1 or to no vertex of Q_1,
- every vertex of $V \setminus (Q_1 \cup Q_2)$ is adjacent to either all vertices of Q_2 or to no vertex of Q_2,
- $|Q_1| \geqslant 2$ or $|Q_2| \geqslant 2$ and
- $|V \setminus (Q_1 \cup Q_2)| \geqslant 2$.

See Fig. 1 for an example.

Homogeneous pairs are a generalization of homogeneous sets. By a *homogeneous set* in a graph G, we shall mean a set Q of vertices of G such that each vertex of $V \setminus Q$ is either adjacent to all vertices of Q or to none of the vertices of Q, $|Q| \geqslant 2$

☆ This research was partially supported by NSERC, FCAR and CNPq.
* Corresponding author. E-mail: sula@cos.ufrj.br.

SSDI 0166-218X(95)00090-9

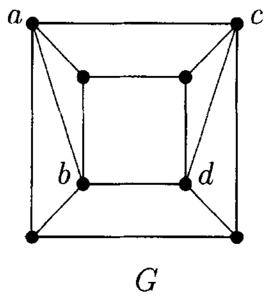

Fig. 1. $\{\{a,b\},\{c,d\}\}$ is a homogeneous pair in graph G

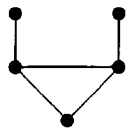

Fig. 2. Bull.

and $|V \setminus Q| \geqslant 1$. Note that a homogeneous set Q with $|V \setminus Q| \geqslant 2$ is a homogeneous pair where Q_2 is empty. A key result in perfect graph theory (for an introduction see [7]) is that no minimal imperfect graph contains a homogeneous set. In fact, this is an important lemma used by Lovász in the proof of his celebrated Perfect Graph Theorem [10]: "G is perfect if and only if its complement is perfect". This result has also been used to prove that various classes of graphs are perfect, for example comparability graphs [8] and P_4-free graphs [4].

Many optimization problems, such as the clique, independent set, chromatic number, and clique cover problems, which are NP-complete for general graphs are solvable in polynomial time for perfect graphs using the ellipsoid method [9]. For many classes of perfect graphs, simpler polynomial time algorithms for solving these problems have been found which are based on a structural decomposition of the graph. See [2] for a summary. In particular, homogeneous sets have been used to obtain a structural decomposition for the class of comparability graphs and P_4-free graphs and, consequently, simple algorithms for many optimization problems for these graphs [8, 4, 12].

Polynomial time algorithms for finding homogeneous sets are given in [11, 12, 5]. The fastest existing algorithm is the $O(m)$ algorithm by Spinrad (unpublished; see [12] for an $O((m\alpha(m,n))$ algorithm).

Homogeneous pairs were introduced by Chvátal and Sbihi [3] in 1987. They showed that no minimal imperfect graph contains a homogeneous pair. They used this result to prove that bull-free Berge graphs are perfect. A graph is Berge when neither the graph nor its complement contains an induced odd cycle of size at least five. A *Bull-free graph* is a graph that does not contain a certain induced subgraph with five vertices. This graph, shown in Fig. 2, is called a *bull.*

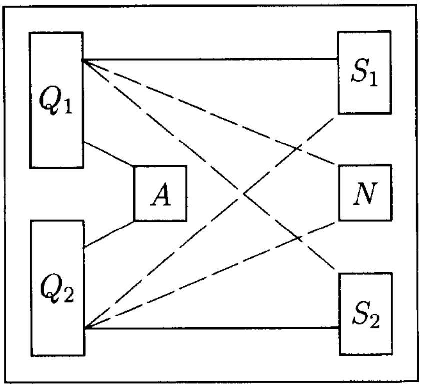

Fig. 3. Graph G with homogeneous pair $\{Q_1, Q_2\}$.

Currently, no simple algorithms exist for the above mentioned optimization problems for the class of bull-free graphs. A structural decomposition approach based on homogeneous pairs would use a fast algorithm for finding a homogeneous pair.

Observe that the existence of a homogeneous pair $\{Q_1, Q_2\}$ in a graph G implies that the set of vertices $V \setminus (Q_1 \cup Q_2)$ can be partitioned into four sets A, N, S_1 and S_2 such that

$A = \{v \in V \mid \mathcal{N}(v) \cap (Q_1 \cup Q_2) = Q_1 \cup Q_2\}$,
$N = \{v \in V \mid \mathcal{N}(v) \cap (Q_1 \cup Q_2) = \emptyset\}$,
$S_1 = \{v \in V \mid \mathcal{N}(v) \cap Q_1 = Q_1\ \mathcal{N}(v) \cap Q_2 = \emptyset\}$,
$S_2 = \{v \in V \mid \mathcal{N}(v) \cap Q_2 = Q_2, \mathcal{N}(v) \cap Q_1 = \emptyset\}$,

where $\mathcal{N}(v) = \{x \in V \mid \{v,x\} \in E\}$. We can represent a graph G that has a homogeneous pair by the diagram in Fig. 3, where a continuous line between two sets represents the property that each vertex of one set is adjacent to each vertex of the other set. A broken line represents the property that no vertex of a set is adjacent to any vertex of the other set.

2. The algorithm

Our algorithm looks for a homogeneous pair in two stages. First we check if G has a homogeneous pair which is a homogeneous set; that is, if G has a homogeneous set H with $|V \setminus H| \geqslant 2$. The simple fact below implies that we can test for such an H in $O(m)$-time using Spinrad's algorithm.

Fact. *If G is a graph with $|V| \geqslant 4$ and H is a homogeneous set in G with $|V \setminus H| = 1$ then G has a homogeneous set H' with $|V \setminus H'| \geqslant 2$ if and only if there is a homogeneous set in the subgraph $G[H]$ of G induced by H.*

Proof. Let G be a graph with $|V| \geqslant 4$, let H be a homogeneous set in G with $|H| = |V - 1|$, and let $V \setminus H = \{x\}$. We note that x is either adjacent to all of H or to none of H. Thus if H' is a homogeneous set in $G[H]$ then H' is a homogeneous set in G with $|V \setminus H'| \geqslant 2$. It remains to show that if there is a homogeneous set H_1 in G with $|V \setminus H_1| \geqslant 2$ then $G[H]$ has a homogeneous set. So let H_1 be a homogeneous set in G with $|V \setminus H_1| \geqslant 2$. If $|H_1| \geqslant 3$ or $x \notin H_1$ then $H_1 \cap H$ is a homogeneous set in $G[H]$ and we are done. Thus $H_1 = \{x, y\}$ for some $y \in H$. But now x and y either both are adjacent to all of $V \setminus \{x, y\}$ or to none of $V \setminus \{x, y\}$. Thus $V \setminus \{x, y\}$ is a homogeneous set in G with fewer than $|V| - 1$ vertices as required. □

In the second stage we look for a homogeneous pair with both Q_1 and Q_2 non-empty and one of S_1 or S_2 non-empty. Without loss of generality we can assume S_1 is non-empty. We start by making a list $\mathcal{L}$ of all possible ordered triples of vertices (q_1, q_2, s_1) such that $\{q_1, s_1\} \in E$ and $\{q_2, s_1\} \notin E$. Now G has such a homogeneous pair if and only if it has a homogeneous pair with q_1 in Q_1, q_2 in Q_2 and s_1 in S_1 for some triple of $\mathcal{L}$.

We describe an $O(n^2)$ algorithm to test for a particular ordered triple (q_1, q_2, s_1) if G has a homogeneous pair with q_1 in Q_1, q_2 in Q_2 and s_1 in S_1. We apply this algorithm to all the possibilities in turn. If we ever find a homogeneous pair we return it. Otherwise we return *NO*. The total time taken by the algorithm is $O(n^5)$. [1]

We partition the vertex set $V^* = V \setminus \{q_1, q_2, s_1\}$ of G into the following eight sets

$$AQ_1 = \{x \in V^* \mid \{x, q_1\} \in E,\ \{x, q_2\} \in E \text{ and } \{x, s_1\} \in E\};$$

$$AQ_2 = \{x \in V^* \mid \{x, q_1\} \in E,\ \{x, q_2\} \in E \text{ and } \{x, s_1\} \notin E\};$$

$$S_1Q_1 = \{x \in V^* \mid \{x, q_1\} \in E,\ \{x, q_2\} \notin E \text{ and } \{x, s_1\} \in E\};$$

$$S_2Q_1 = \{x \in V^* \mid \{x, q_1\} \notin E,\ \{x, q_2\} \in E \text{ and } \{x, s_1\} \in E\};$$

$$S_1Q_2 = \{x \in V^* \mid \{x, q_1\} \in E,\ \{x, q_2\} \notin E \text{ and } \{x, s_1\} \notin E\};$$

$$S_2Q_2 = \{x \in V^* \mid \{x, q_1\} \notin E,\ \{x, q_2\} \in E \text{ and } \{x, s_1\} \notin E\};$$

$$NQ_1 = \{x \in V^* \mid \{x, q_1\} \notin E,\ \{x, q_2\} \notin E \text{ and } \{x, s_1\} \in E\};$$

$$NQ_2 = \{x \in V^* \mid \{x, q_1\} \notin E,\ \{x, q_2\} \notin E \text{ and } \{x, s_1\} \notin E\}.$$

Consider a vertex $x \in AQ_1$. Since x is adjacent to $q_1 \in Q_1$, to $q_2 \in Q_2$ and to $s_1 \in S_1$ it cannot be in the sets S_1 or S_2 or Q_2 or N. In other words x must be assigned to either A or Q_1 which we indicate with the notation $x \to A$ or $x \to Q_1$. So, with a similar analysis for the other sets, we have

$$AQ_1 \subseteq \{x \in V^* \mid x \to A \text{ or } x \to Q_1\},$$

$$AQ_2 \subseteq \{x \in V^* \mid x \to A \text{ or } x \to Q_2\},$$

$$S_1Q_1 \subseteq \{x \in V^* \mid x \to S_1 \text{ or } x \to Q_1\},$$

[1] It was noted by one of the referees that this procedure can be generalized to yield a polynomial algorithm for finding a homogeneous k-set.

$$S_2Q_1 \subseteq \{x \in V^* \mid x \to S_2 \text{ or } x \to Q_1\},$$

$$S_1Q_2 \subseteq \{x \in V^* \mid x \to S_1 \text{ or } x \to Q_2\},$$

$$S_2Q_2 \subseteq \{x \in V^* \mid x \to S_2 \text{ or } x \to Q_2\},$$

$$NQ_1 \subseteq \{x \in V^* \mid x \to N \text{ or } x \to Q_1\},$$

$$NQ_2 \subseteq \{x \in V^* \mid x \to N \text{ or } x \to Q_2\}.$$

The idea of the algorithm, which follows closely that of Apsvall et al. [1], is to transform the sets AQ_1, AQ_2, S_1Q_1, S_2Q_1, S_1Q_2, S_2Q_2, NQ_1 and NQ_2 into the sets A, N, S_1, S_2, Q_1 and Q_2, by specifying for each vertex whether or not it should be placed in $Q_1 \cup Q_2$. We say that a vertex is *internal* if we place it in $Q_1 \cup Q_2$ and *external* otherwise. Thus, after all our choices have been made, A is the union of the external vertices in AQ_1 and AQ_2, S_1 is the union of the external vertices in S_1Q_1 and S_1Q_2, S_2 is the union of the external vertices of S_2Q_1 and S_2Q_2 and N is the union of the external vertices of NQ_1 and NQ_2. Then Q_1 will be formed from the internal vertices of AQ_1, S_1Q_1, S_2Q_1 and NQ_1, and Q_2 will be formed from the internal vertices of AQ_2, S_1Q_2, S_2Q_2 and NQ_2.

It is easy to see that G contains a homogeneous pair with $q_1 \in Q_1$, $q_2 \in Q_2$ and $s_1 \in S_1$ if and only if for every pair of vertices x and y, x, $y \in V^*$, the following conditions are satisfied. Conditions (I)–(VI) ensure that vertices are placed so that all constraints on the existence of edges and non-edges are satisfied and condition (VII) ensures that the technical requirements of the definition of a homogeneous pair that at least one of Q_1 or Q_2 has at least two vertices and that there are at least two vertices outside $Q_1 \cup Q_2$ are satisfied.

(I) If x and y are not adjacent and they are both in one of AQ_1, AQ_2, S_1Q_1 or S_2Q_2, then they are either both internal or both external.

(II) If x and y are adjacent and they are both in one of S_1Q_2, S_2Q_1, NQ_1 or NQ_2, then either they are both internal or both external.

(III) If $x \in AQ_1$, $y \in AQ_2$ and x and y are not adjacent then either they are both internal or both external. Similarly for $x \in AQ_1$ and $y \in S_1Q_1$ or $y \in S_1Q_2$; $x \in AQ_2$ and $y \in S_2Q_1$ or $y \in S_2Q_2$; $x \in S_1Q_2$ and $y \in S_2Q_1$.

(IV) If $x \in S_1Q_1$, $y \in S_2Q_2$ and x and y are adjacent then either they are both internal or both external. Similarly for $x \in S_1$ and $y \in NQ_2$; $x \in S_2Q_2$ and $y \in NQ_1$; $x \in S_1Q_2$ and $y \in NQ_2$; $x \in S_2Q_1$ and $y \in NQ_1$; $x \in NQ_1$ and $y \in NQ_2$.

(V) If $x \in AQ_1$, $y \in S_2Q_1$ and x and y are not adjacent then if x is external y is also external. Similarly for $x \in AQ_1$ and $y \in S_2Q_2$ or $y \in NQ_1$ or $y \in NQ_2$; $x \in AQ_2$ and $y \in S_1Q_1$ or $y \in S_1Q_2$ or $y \in NQ_1$ or $y \in NQ_2$; $x \in S_1Q_1$ and $y \in S_2Q_1$ or $y \in NQ_1$; $x \in S_1Q_2$ and $y \in S_1Q_1$ or $y \in NQ_1$; $x \in S_2Q_1$ and $y \in S_2Q_2$ or $y \in NQ_2$.

(VI) If $x \in AQ_1$, $y \in S_2Q_1$ and x and y are adjacent then if x is internal y is also internal. Similarly for $x \in AQ_1$ and $y \in S_2Q_2$ or $y \in NQ_1$ or $y \in NQ_2$; $x \in AQ_2$ and $y \in S_1Q_1$ or $y \in S_1Q_2$ or $y \in NQ_1$ or $y \in NQ_2$; $x \in S_1Q_1$ and $y \in S_2Q_1$ or $y \in NQ_1$; $x \in S_1Q_2$ and $y \in S_1Q_1$ or $y \in NQ_1$; $x \in S_2Q_1$ and $y \in S_2Q_2$ or $y \in NQ_2$.

(VII) There is at least one external vertex and one internal vertex.

We remark that we could enforce condition (VII) by trying all possible choices for external and internal vertices in V^*. We could then apply the algorithm of Apsvall et al. [1] directly to a 2-SAT instance describing our conditions. This yields an $O(n^7)$ algorithm. We do not see how to apply Apsvall et al. [1] to obtain an $O(mn^3)$ algorithm for our problem. Our algorithm for solving the problem however mimics their algorithm quite closely.

We will need the following definitions. A directed graph $\vec{G}$ is *strongly connected* if there is a path from any vertex to any other. The maximal strongly connected subgraphs of a graph are vertex disjoint and are called its *strong components*. We say that $(C_1, C_2, ..., C_k)$ is a *reverse topological ordering* of the strong components of $\vec{G}$ if there is no arc directed from a vertex of C_i to a vertex of C_j, for $i < j$. If C_k and C_l are two strong components such that there is an arc directed from a vertex of C_k to a vertex of C_l then C_k is called a *predecessor* of C_l and C_l is called a *successor* of C_k.

Homogeneous pair-algorithm

Input: a graph $G = (V, E)$ with $|V| \geqslant 4$.

Output: *YES-G has a homogeneous pair* or *NO-G does not have a homogeneous pair*. If the answer is *YES*, the algorithm also returns the homogeneous pair $\{Q_1, Q_2\}$.

Step 0 Use Spinrad's $O(m)$ algorithm to test if G has a homogeneous set. If it returns with a homogeneous set H with $|H| \leqslant |V-2|$ then return *YES-G has a homogeneous pair,* $Q_1 = H$ *and* $Q_2 = \emptyset$ and stop. If it returns with a homogeneous set H such that $V \setminus H$ is a single vertex x then run it again on $G[H]$. If $G[H]$ has a homogeneous set H' then return *YES-G has a homogeneous pair,* $Q_1 = H'$ *and* $Q_2 = \emptyset$ and stop.

Step 1 Make a list $\mathcal{L}$ of all ordered triples (a, b, c) of vertices of G such that $\{a, c\} \in E$ and $\{b, c\} \notin E$.

Step 2 If $\mathcal{L}$ is empty return *NO-G does not have a homogeneous pair* and stop. Otherwise let T be the first triple in $\mathcal{L}$. Let q_1 be the first vertex of T, let q_2 be the second and let s_1 be the third. Remove T from $\mathcal{L}$.

Step 3 Partition the set $V^* = V \setminus \{q_1, q_2, s_1\}$ into the eight sets AQ_1, AQ_2, S_1Q_1, S_2Q_1, S_1Q_2, S_2Q_2, NQ_1, NQ_2 as defined above.

Step 4 Construct a directed graph $\vec{G}_{q_1,q_2,s_1} = (V(\vec{G}_{q_1,q_2,s_1}), E(\vec{G}_{q_1,q_2,s_1}))$ as follows. The set $V(\vec{G}_{q_1,q_2,s_1}) = V^*$. The set $E(\vec{G}_{q_1,q_2,s_1}))$ is given by the next two tables. We denote by $\langle x, y \rangle$ the fact that (x, y) and (y, x) are edges of $\vec{G}$. Table 1 shows the directed edges corresponding to the edge $\{x, y\}$ of G, where x belongs to a set in the first column and y belongs to a set in the first row. Table 2 represents the directed edges corresponding to the fact that $\{x, y\}$ is not an edge of G.

Step 5 Find the strong components of $\vec{G}_{q_1,q_2,s_1}$ and return them in reverse topological order.

Step 6 Process the strong components of $\vec{G}_{q_1,q_2,s_1}$ (in reverse topological order) as follows. If $\vec{G}_{q_1,q_2,s_1}$ has only one strong component, then G does not have a homogeneous pair for this chosen triple of vertices, so return to Step 2.

Table 1

(edges)	AQ_1	AQ_2	S_1Q_1	S_2Q_2	S_1Q_2	S_2Q_1	NQ_1	NQ_2
AQ_1				(y,x)		(y,x)	(y,x)	(y,x)
AQ_2			(y,x)		(y,x)		(y,x)	(y,x)
S_1Q_1		(x,y)		$\langle x,y\rangle$	(x,y)	(y,x)	(y,x)	$\langle x,y\rangle$
S_2Q_2	(x,y)		$\langle x,y\rangle$		(y,x)	(x,y)	$\langle x,y\rangle$	(y,x)
S_1Q_2		(x,y)	(y,x)	(x,y)	$\langle x,y\rangle$		(y,x)	$\langle x,y\rangle$
S_2Q_1	(x,y)		(x,y)	(y,x)		$\langle x,y\rangle$	$\langle x,y\rangle$	(y,x)
NQ_1	(x,y)	(x,y)	(x,y)	$\langle x,y\rangle$	(x,y)	$\langle x,y\rangle$	$\langle x,y\rangle$	$\langle x,y\rangle$
NQ_2	(x,y)	(x,y)	$\langle x,y\rangle$	(x,y)	$\langle x,y\rangle$	(x,y)	$\langle x,y\rangle$	$\langle x,y\rangle$

Table 2

non-edges	AQ_1	AQ_2	S_1Q_1	S_2Q_2	S_1Q_2	S_2Q_1	NQ_1	NQ_2
AQ_1	$\langle x,y\rangle$	$\langle x,y\rangle$	$\langle x,y\rangle$	(x,y)	$\langle x,y\rangle$	(x,y)	(x,y)	(x,y)
AQ_2	$\langle x,y\rangle$	$\langle x,y\rangle$	(x,y)	$\langle x,y\rangle$	(x,y)	$\langle x,y\rangle$	(x,y)	(x,y)
S_1Q_1	$\langle x,y\rangle$	(y,x)	$\langle x,y\rangle$		(y,x)	(x,y)	(x,y)	
S_2Q_2	(y,x)	$\langle x,y\rangle$		$\langle x,y\rangle$	(x,y)	(y,x)		(x,y)
S_1Q_2	$\langle x,y\rangle$	(y,x)	(x,y)	(y,x)		$\langle x,y\rangle$	(x,y)	
S_2Q_1	(y,x)	$\langle x,y\rangle$	(y,x)	(x,y)	$\langle x,y\rangle$			(x,y)
NQ_1	(x,y)	(x,y)	(x,y)		(x,y)			
NQ_2	(x,y)	(x,y)		(x,y)		(x,y)		

Otherwise mark all the vertices of the strong components that have predecessors with E and the vertices of the other components (that have no predecessors) with I. If there are only isolated components (i.e. components that have no predecessors and no successors), choose one of them and mark its vertices with I and mark all the vertices of the other components with E.

Step 7 Set:

Q_1 = {vertices of AQ_1 marked I } ∪ {vertices of S_1Q_1 marked I}∪ {vertices of S_2Q_1 marked I} ∪ {vertices of NQ_1 marked I} ∪ $\{q_1\}$.

Q_2 = {vertices of AQ_2 marked I } ∪ {vertices of S_1Q_2 marked I} ∪ {vertices of S_2Q_2 marked I} ∪ {vertices of NQ_2 marked I} ∪ $\{q_2\}$.

Return *YES-G has a homogeneous pair* and the sets Q_1 and Q_2.

3. Why it works

In this section, assume we have a triple of vertices $\{q_1, q_2, s_1\}$ given by Step 2. We show that the algorithm correctly marks the vertices of the directed graph $\vec{G}_{q_1,q_2,s_1}$ so that conditions (I)–(VII) are satisfied if and only if G has a homogeneous pair with $q_1 \in Q_1$, $q_2 \in Q_2$ and $s_1 \in S_1$.

Consider the directed edges of $\vec{G}_{q_1,q_2,s_1}$. The edge (x, y), for example, where $x \in S_2Q_2$ and $y \in AQ_1$ represents the fact that if x is placed in S_2, that is, if x is external, then y must not be placed in Q_1, that is, y cannot be internal. This is exactly the constraint required by condition (VI). Thus the vertices of $\vec{G}_{q_1,q_2,s_1}$ can be marked so that there is no edge (x,y) with x external and y internal iff conditions (I)–(VI) can be satisfied. So we have proved:

Lemma 1. *G has a homogeneous pair with $q_1 \in Q_1$, $q_2 \in Q_2$ and $s_1 \in S_1$ if and only if the vertices of $\vec{G}_{q_1,q_2,s_1}$ can be marked internal and external (or I and E) such that the next two conditions are satisfied:*

(i) *no directed edge (x,y), can have x external and y internal;*

(ii) *there is at least one external vertex and one internal vertex.*

Clearly, for each strong component of $\vec{G}$, condition (i) implies that all the vertices are either external or internal. This fact, needed for the correctness proof of the algorithm, is stated explicitly in the next lemma.

Lemma 2. *Suppose G has a homogeneous pair with $q_1 \in Q_1$, $q_2 \in Q_2$ and $s_1 \in S_1$. Then each strong component of $\vec{G}_{q_1,q_2,s_1}$ either has only external vertices or only internal vertices.*

We are now ready to state our main theorem.

Theorem 1. *The algorithm correctly determines whether G has a homogeneous pair.*

Proof. Suppose the algorithm does not report failure. By our fact about homogeneous sets which are homogeneous pairs, if the algorithm terminates in Step 0 then it does indeed return a homogeneous pair. Otherwise, the algorithm terminates because for some triple $\{q_1, q_2, s_1\}$ the directed graph $\vec{G}_{q_1,q_2,s_1}$ has more than one strong component. Since all vertices of a strong component receive the same mark and only the components that have no predecessor (in the case we do not have all isolated components) are marked with I this implies that there is no directed edge (x,y) with x marked E and y marked I. It is clear that this is also true for the case of all isolated components. Since there is at least one component marked E and one component marked I there will be at least one internal and one external vertex. Then by Lemma 1 G has a homogeneous pair.

Now, let us assume that G has a homogeneous pair $\{Q_1, Q_2\}$. In this case we must show that in this case our algorithm finds a homogeneous pair in G. If for this homogeneous pair one of Q_1, Q_2, or $Q_1 \cup Q_2$ is a homogeneous set containing fewer than $|V| - 1$ vertices then by our fact about homogeneous sets which are homogeneous pairs we will find such a homogeneous set, which is also a homogeneous pair in Step 0. Otherwise, G has a homogeneous pair $\{Q_1, Q_2\}$ such that for some triple of vertices $\{a, b, c\}$ in G we have $a \in Q_1$, $b \in Q_2$, $c \in S_1$. Then the vertices of $\vec{G}_{a,b,c}$ can be marked so that the conditions of Lemma 1 are satisfied. We want to show that the algorithm finds such a marking in the iteration in which $T = \{a, b, c\}$ (we can assume the algorithm performs this iteration as otherwise it halted earlier and therefore found a homogeneous pair as required). Suppose the algorithm fails to find a marking. This implies that $\vec{G}_{a,b,c}$ has only one strong component. According to Lemmas 1 and 2 the vertices of this component have to be all marked E or all marked I. If all are marked E, then all vertices are going to be external and we have $Q_1 = q_1$ and $Q_2 = q_2$, contradicting the definition of homogeneous pair. If all are marked I, then all vertices are going to be internal and $V = Q_1 \cup Q_2$, also contradicting the definition of a homogeneous pair. $\square$

4. Time complexity

We can estimate the running time of the homogeneous pair-algorithm as follows:

Step 0 can be implemented in $O(m)$-time. Step 1 takes $O(mn)$-time since the triples are made up of one edge and one other vertex. Step 2 can be done in constant time. In Step 3 it is clear that the construction of each of the eight sets takes time proportional to n. In Step 4 the construction of the directed graph $\vec{G}_{q_1,q_2,s_1}$ clearly takes $O(n^2)$. Step 5 can be done using Tarjan's algorithm [13] for finding strong components and returning them in reverse topological order in a time proportional to the size of the graph. Clearly Steps 6 and 7 can be done in linear time. Then Steps 2–5 take $O(n^2)$-time. Since we will apply Steps 2–5 at most mn times the algorithm takes $O(mn^3)$-time in total. Then it follows:

Theorem 2. *Determining whether a graph has a homogeneous pair can be computed in $O(mn^3)$-time.*

Acknowledgements

We would like to thank Celina M.H. de Figueiredo and Marcia R. Cerioli for careful readings of earlier drafts of this paper and René Ferland for giving invaluable advice on LATEX.

References

[1] B. Aspvall, F. Plass and R.E. Tarjan, A linear-time algorithm for testing the truth of certain quantified Boolean formulas, Inform. Process. Lett. 8 (1979) 121–123.

[2] A. Brandstädt, Special graph classes – a survey, Schriftenreihe des Fachbereihs Mathematik, SM-DU-199, Universität Duisburg Gesamthochschule (1993).

[3] V. Chvátal and N. Sbihi, Bull-free Berge graphs are perfect, Graphs Combin. 3 (1987) 127–139.

[4] D.G. Corneil, Y. Perl and L. Stewart, Cographs: recognition, application and algorithms, Congr. Numer. 43 (1984) 249–258.

[5] A. Cournier, Sur quelques Algorithmes de Décomposition de Graphes, Thése, Université Montpellier II (1993).

[6] M.R. Garey and D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, (W.H. Freeman, San Francisco, 1979).

[7] M.C. Golumbic, Algorithmic Graph Theory and Perfect Graphs (Academic Press, New York, 1980).

[8] M.C. Golumbic, Comparability graphs and a new matroid, J. Combin. Theory. Drt. B 22 (1977) 68–90.

[9] M. Grotschel, L. Lovasz and A. Schrijver, Polynomial algorithms for perfect graphs, Ann. Discrete Math. 21 (1984) 325–356.
[10] L. Lovász, Normal hypergraphs and the perfect graph conjecture, Discrete Math. 2 (1972) 253–267.
[11] J.H. Muller and J. Spinrad, Incremental modular decomposition, J. ACM 1 (1989) 1–19.
[12] J. Spinrad, P_4-trees and substitution decomposition, Discrete Appl. Math. 39 (1992) 263–291.
[13] R.E. Tarjan, Depth first search and linear graph algorithms, SIAM J. Comput. 1 (1972) 146–160.