Theoretical Computer Science 407 (2008) 574–582

Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

Note

Matching cutsets in graphs of diameter 2

Mieczysław Borowiecki, Katarzyna Jesse-Józefczyk *

Faculty of Mathematics Computer Science and Econometrics, University of Zielona Góra, prof. Z. Szafrana 4a, 65-516 Zielona Góra, Poland

ARTICLE INFO

Article history:
Received 17 January 2008
Received in revised form 27 May 2008
Accepted 2 July 2008
Communicated by D.-Z. Du

Keywords:
Matching cutset
Stable cutset
Graph algorithms

ABSTRACT

We say that a graph has a matching cutset if its vertices can be coloured in red and blue in such a way that there exists at least one vertex coloured in red and at least one vertex coloured in blue, and every vertex has at most one neighbour coloured in the opposite colour. In this paper we study the algorithmic complexity of a problem of recognizing graphs which possess a matching cutset. In particular we present a polynomial-time algorithm which solves this problem for graphs of diameter two.

1. Introduction

A set M of independent edges in a graph $G = (V, E)$ is called a *matching*. A matching M is called *perfect* if every vertex of $V(G)$ is incident to an edge of M. Moreover a matching M is *almost perfect* if all the vertices of $V(G)$ but one are incident to an edge of M. A set S of edges (or vertices) of a graph G is called a *cutset* (in G) if $G - S$ has more components than G. Furthermore if a cutset is a matching of G, then it is called a *matching cutset*. In other words we say that a graph has a matching cutset if its vertices can be coloured in red and blue in such a way that there exists at least one vertex coloured in red and at least one vertex coloured in blue, and every vertex has at most one neighbour coloured in the opposite colour. The problem of recognizing graphs with a matching cutset (let MATCHING CUTSET denote this problem) is well-studied in the literature. Here are some of the most important results. Chvátal [6] proved that MATCHING CUTSET is NP-complete even for graphs with maximum degree 4. Result of Moshi [13] that MATCHING CUTSET is NP-complete, even if the input is restricted to bipartite graphs of minimum degree 2, was extended by Le and Randerath [10] to bipartite graphs with one colour class consisting only of vertices of degree 3 and the other colour class consisting only of vertices of degree 4. In the same paper [13] Moshi presented an $O(n^3m)$ algorithm for graphs without induced cycles of length greater than 4 which determines whether G has a matching cutset, where $n = |V(G)|$ and $m = |E(G)|$. He also proved that if G is a line graph, then we can determine in $O(m)$ time whether G has a matching cutset. The importance of MATCHING CUTSET follows e.g. from its strong connection with other well-known problem, i.e. problem of deciding whether a given graph has a set I of a pairwise non-adjacent vertices such that I is a cutset (let STABLE CUTSET denote this problem). Brandstädt proved that the solution of STABLE CUTSET for line graphs can be obtained from the solution of MATCHING CUTSET.

Theorem 1 ([1]). *If $L(G)$ has a stable cutset, then G has a matching cutset. If G has a matching cutset, then $L(G)$ has a stable cutset.*

STABLE CUTSET has been studied widely in the literature since Tucker in [14] showed its connection with perfect graphs. For a survey of what is known to date, see [1,4,10,14].

* Corresponding author. Tel.: +48 68 328 2869; fax: +48 68 328 2875.
E-mail addresses: M.Borowiecki@wmie.uz.zgora.pl (M. Borowiecki), K.Jesse-Jozefczyk@wmie.uz.zgora.pl (K. Jesse-Józefczyk).

0304-3975/$ – see front matter
doi:10.1016/j.tcs.2008.07.002

In this paper we consider only finite undirected graphs without loops or multiple edges. The *neighbourhood* of a vertex v is $N(v) = \{u : vu \in E(G)\}$. The *closed neighbourhood* of v is $N[v] = N(v) \cup \{v\}$. For a set $X \subseteq V(G)$, $N(X)$ is the set of vertices outside X which are adjacent to at least one vertex of X. The *degree* of a vertex v will be denoted by $d(v)$ ($d(v) = |N(v)|$). *Minimal (maximal) vertex degree* of the graph G will be denoted by $\delta(G)$ ($\Delta(G)$).

The *distance* $dist_G(u, v)$ in a connected graph G of two vertices u, v is the length of the shortest path in G from u to v. The diameter of a connected graph G, denoted by $diam(G)$, is defined as $\max\{dist_G(u, v) : u, v \in V(G)\}$.

Let $R(G)$ ($B(G)$) denote the set of vertices which are coloured in red (blue) and let us denote by $E_2(G)$ the set of all bichromatic edges of G, i.e. the set of edges whose end vertices are coloured in different colours. When no confusion is possible, we denote $R(G)$, $B(G)$ and $E_2(G)$ by R, B and E_2, respectively. The set of the uncoloured vertices of G will be denoted by $\mathcal{U}(G)$ or briefly by $\mathcal{U}$.

A set W of vertices of a graph G is said to be a *module* of G, if every vertex outside W is either adjacent to all the vertices of W or to none of them. Empty set, singletons and $V(G)$ are *trivial* modules. If G has only trivial modules, then it is called *prime*. *Modular decomposition* is a representation of the modules of a graph. It has been studied widely since in special graph classes it can be used to achieve a linear-time solution for NP-complete problems. There are many polynomial and even linear-time algorithms for finding modular decomposition. For an extensive survey, see [5,7,11,12].

In this paper we will restrict our attention to graphs of diameter 2. There are many papers in which properties of these graphs are investigated, see for example: [2,3,8,9] and their references.

Let $\mathcal{P}(2)$ denote the class of prime graphs of diameter 2 and for a set $F \subseteq E(G)$ let $V_F = \bigcup_{uv\in F}\{u, v\}$.

Many of the papers which concern MATCHING CUTSET focus on graphs which are rather sparse what is implied by the small $\Delta(G)$ of the considered graphs. Even though the graphs are sparse and have so small $\Delta(G)$ most of the results are negative. For a change we would like to present a positive result. The aim of this paper is to prove that MATCHING CUTSET is polynomial for a graph G of diameter 2. One can see that graphs of diameter 2 are quite dense. We also do not put any bounds on $\Delta(G)$. The proof will consist of two parts Theorems 2 and 3 which concern the graphs belonging to $\mathcal{P}(2)$ and graphs with $diam(G) = 2$ which are not prime, respectively. The motivation for this division is a very interesting structure of the graphs in $\mathcal{P}(2)$ (see Corollary 1).

2. Prime graphs of diameter two

Lemma 1. *If $\delta(G) = 1$, then G has a matching cutset.*

It is easy to see that if we colour a vertex v, such that $d(v) = 1$, in red and all the other vertices in blue we obtain the required colouring. In this situation only one edge will belong to the matching cutset.

Lemma 2. *If a graph $G \in \mathcal{P}(2)$ of odd order n has a matching cutset M, then there exists exactly one vertex v such that $deg(v) = \frac{n-1}{2}$ and $v \notin M$.*

Proof. Suppose that a graph $G \in \mathcal{P}(2)$ of odd order n has a matching cutset. Furthermore suppose that there exists a vertex v, such that $v \notin V_M$, $v \in R$ ($v \in B$) and v is not adjacent to a vertex $u \in R$ ($u \in B$) with the property that $u \in V_M$. We know that there exists a vertex $u' \in B$ ($u' \in R$) such that $uu' \in E_2$ and hence $dist(v, u') \geq 3$. This is a contradiction of the fact that $G \in \mathcal{P}(2)$. So v must be adjacent to all the vertices in V_M which are coloured in red (blue), but not adjacent to any blue (red) vertex. Now suppose that there exist vertices $u, v \notin V_M$, $u \in R$, $v \in B$. Once again we get a contradiction because $dist(u, v) \geq 3$. So assume that there exist at least two vertices which do not belong to V_M. We have already shown that they must be either all in R or all in B, but then we get another contradiction since the set of the vertices which do not belong to V_M would create a non-trivial module. Because n is odd there must exist one vertex which does not belong to V_M. This observation completes the proof. ■

From this lemma follow these corollaries.

Corollary 1. *If a graph $G \in \mathcal{P}(2)$ has a matching cutset M, then M is perfect or almost perfect depending on the parity of the order of G.*

Corollary 2. *If a graph $G \in \mathcal{P}(2)$ of order n has a maximum degree $\Delta(G) > \left\lfloor \frac{n+1}{2} \right\rfloor$, then G does not have a matching cutset.*

It is easy to see that the next remark is true.

Remark 1. If an edge e belongs to a triangle of G, then e does not belong to a matching cutset.

Now we can formulate our first theorem.

Theorem 2. *If a graph $G \in \mathcal{P}(2)$, then we can decide in polynomial time whether G has a matching cutset.*

Proof. For a better understanding of the proof it will be based on two algorithms. The reason for existing as many as two algorithms is radically different approach to resolving MATCHING CUTSET for graphs in $\mathcal{P}(2)$ of even (**E Algorithm**) and odd order (**O Algorithm**).

Let G has an even order. It is easy to see that if we have a graph with $\delta(G) = 2$, then we can decide in $O(n^2m)$ time whether G has a matching cutset. It follows from the fact that if a graph G has a matching cutset then vertex v of degree two must have two neighbours x,y such that $d(x) = \frac{n}{2}$ and $d(y) \geq \frac{n}{2} - 1$. So we will consider only graphs with $\delta(G) \geq 3$. Now we can use the **E Algorithm**. Time complexity of this algorithm is $O(n^2m^3)$. However its average time complexity may be better (see Remark 2). The **E Algorithm** uses four subroutines. The correctness of these subroutines will be proved in a sequence of claims.

E Algorithm
Input: a graph $G \in \mathcal{P}(2)$ of order n such that $\Delta(G) \leq \frac{n}{2}$, $\delta(G) \geq 3$ and n is even,
Output: matching cutset of the graph G or a message that the graph G does not have a matching cutset.

$S \leftarrow \phi$;
$S_1 \leftarrow \phi$;
call **Labelling**;
if there exists a vertex v such that for every edge e incident to v $l(e) = 0$ then
    return a message that there does not exist a matching cutset;
    stop **E Algorithm**;
if $S_1 \neq \phi$ then
    call **Initialization**;
    call **Extension**;
    call **Pairs**;
else
    while $S \neq \phi$ do
        call **Initialization**;
        call **Extension**;
        if the **Extension** subroutine did not return a message that
        the initialized colouring cannot be extended then
            call **Pairs**;
return a message that there does not exist a matching cutset;
end **E Algorithm**;

The **Labelling** subroutine is responsible for giving the labels to the edges of the graph. An edge gets a label 0 if it belongs to a triangle in the graph. Such label means that this edge cannot belong to the matching cutset. If there exists a vertex such that all the edges but one, which are incident to it, have label 0 then the last unlabeled edge gets a label 1. We know that the matching cutset is perfect (in this case), so label 1 means that the edge which gets it must belong to the matching cutset. Please notice that during the **Labelling** subroutine not all the edges must be labeled. One can see that this subroutine takes $O(n^3)$ time.
Labelling
Input: a graph $G \in \mathcal{P}(2)$ of order n such that $\Delta(G) \leq \frac{n}{2}$, $\delta(G) \geq 3$ and n is even,
Output: the graph G with partially labeled edges.

Let $l : E \to \{0, 1\}$ be a partial function and

$$l(e) = \begin{cases} 0 & \text{if } e \text{ belongs to a triangle of } G, \\ 1 & \text{if } e \in E(v) - E_0 \text{ and } |E(v) - E_0| = 1, \end{cases}$$

where $E(v) = \{e \in E(G) : e \text{ is incident to } v\}$, $E_0 = \{e \in E(G) : l(e) = 0\}$.
Label the edges of the graph in accordance with definition of function l;
$S_1 \leftarrow \{e \in E : l(e) = 1\}$;
if $S_1 = \phi$ then $S \leftarrow \{e \in E : l(e) \text{ is not given}\}$;
end **Labelling**;

The next subroutine will choose an edge from which we will start the colouring. The end vertices of the chosen edge will be coloured in red and blue. Time complexity of this subroutine is $O(1)$.
Initialization
Input: a graph $G \in \mathcal{P}(2)$ of even order n such that $\Delta(G) \leq \frac{n}{2}$, $\delta(G) \geq 3$ with edges labeled by the **Labelling** subroutine,

Output: the graph G with one bichromatic edge.

if $S_1 \neq \phi$ then
  take an edge $e = uv \in S_1$ and colour u in red and v in blue;
else
  take an edge $e = uv \in S$ colour u in red and v in blue;
  $S \leftarrow S - \{e\}$;
end **Initialization**;

The **Extension** subroutine can be described as the set of rules which enable us to extend the colouring initialized by the previous subroutine. Time complexity of this subroutine is clearly $O(n^2m)$.

Extension

Input: a graph $G \in \mathscr{P}(2)$ of even order n such that $\Delta(G) \leq \frac{n}{2}$, $\delta(G) \geq 3$ with at least one bichromatic edge coloured by the **Initialization** subroutine,

Output: the graph G with a partial colouring of the vertices or the matching cutset of the graph G or a message that the graph does not have a matching cutset or a message that the given colouring cannot be extended.

Use the following rules until none of the rules will be satisfied. (The order of rules is not important.)

(R1) if there exists an uncoloured vertex v such that:
$|N(v) \cap R| \geq 2$ and $|N(v) \cap B| \geq 2$ or $|B| > \frac{n}{2}$ or $|R| > \frac{n}{2}$ or
there exists a coloured vertex which has at least two neighbours coloured in the opposite colour then
  if you have started the colouring with an edge $e \in S_1$ then
    return a message that there does not exist a matching cutset;
    stop **E Algorithm**;
  else
    remove the existing colouring;
    return a message that given colouring cannot be extended;
(R2) if there exists an uncoloured vertex v such that $|N(v) \cap R| \geq 2$ and
$|N(v) \cap B| \leq 1$ then
  colour v in red;
(R3) if there exists an uncoloured vertex v such that $|N(v) \cap R| \leq 1$ and
$|N(v) \cap B| \geq 2$ then
  colour v in blue;
(R4) if there exists an edge $e = uv$ such that $u \in R$, $v \in B$ and not all the neighbours of u and v are coloured then
  colour the uncoloured neighbours of u in red;
  colour the uncoloured neighbours of v in blue;
(R5) if there exists an edge $e = uv$ and $u \in R$ and v is not coloured and $l(e) = 0$ then
  colour v in red; ($*$ *colouring of a triangle in G* $*$)
(R6) if there exists an edge $e = uv$ and $u \in B$ and v is not coloured and $l(e) = 0$ then
  colour v in blue;
(R7) if there exists a vertex v such that $|N(v) \cap R| = |N(v)| - 1$ then
  colour its uncoloured neighbour in blue;
(R8) if there exists a vertex v such that $|N(v) \cap B| = |N(v)| - 1$ then
  colour its uncoloured neighbour in red;
(R9) if every vertex is coloured and the first rule is not satisfied then
  return E_2 as a matching cutset;
  stop **E Algorithm**;
(R10) if there exists an edge $e = uv$ such that $l(e) = 1$ and $u \in R$ and v is not coloured then colour v in blue; ($*$ *colouring of an edge which must belong to a matching cutset* $*$)
(R11) if there exists an edge $e = uv$ such that $l(e) = 1$ and $u \in B$ and v is not coloured then colour v in red;
end **Extension**;

Claim 1. *If the* **Extension** *subroutine of the* **E Algorithm** *was executed and it did not return any message, then for each uncoloured vertex* v

$$|N(v) \cap R| = |N(v) \cap B| = 1.$$

Proof. Let $v \in \mathcal{U}$ and $e = ut$, $c(u) = r$, $c(t) = b$. v is adjacent to neither u nor t otherwise it would be coloured by the **Extension** (R4). From the fact that $diam(G) = 2$, v must be adjacent to a vertex $x \in N(u) - \{t\} \subseteq R$ and a vertex $y \in N(t) - \{u\} \subseteq B$ or else either $dist(v, u) \geq 3$ or $dist(v, t) \geq 3$. Hence $|N(v) \cap R| > 0$ and $|N(v) \cap B| > 0$. Now, let us

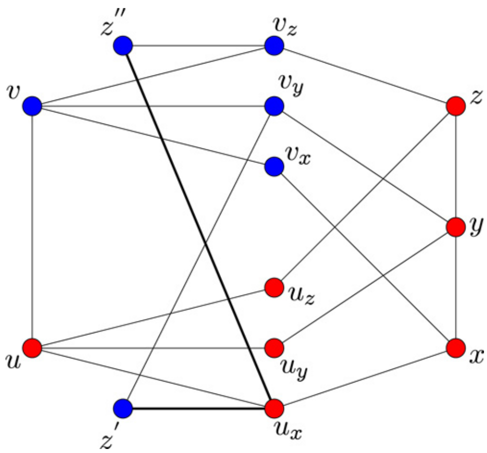

Fig. 1. The colouring of the vertices x, y, z implies the colouring of the vertices z', z''. Bolded edges contradict the existence of the matching cutset. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

suppose that $|N(v) \cap R| \geq 2$ or $|N(v) \cap B| \geq 2$. But it cannot happen because v would be coloured by the **Extension** (R2, R3) or it would return a message that the matching cutset does not exist or that the given colouring cannot be extended properly (R1). Hence $|N(v) \cap R| = |N(v) \cap B| = 1$.

Next three claims give us information about subgraphs induced by the uncoloured vertices.

Claim 2. *If the* **Extension** *subroutine of the* **E Algorithm** *was executed and it was not decided whether a graph G has a matching cutset and the existing colouring can be properly extended, then each connected component Q of the graph G such that* $V(Q) \subseteq \mathcal{U}$ *and* $|V(Q)| \geq 2$ *must be monochromatic to obtain a matching cutset.*

Proof. Let Q be a connected component of the graph G such that $V(Q) \subseteq \mathcal{U}$ and $|V(Q)| \geq 2$ and $x, y \in V(Q)$, $xy \in E$. Furthermore let $e = uv \in E_2$, $c(u) = r$, $c(v) = b$. Neither x nor y is adjacent to u or v otherwise they would be coloured by the **Extension** (R4). From Claim 1 and the fact that $diam(G) = 2$ it follows that both x and y have a neighbour in $N(u)$ and $N(v)$. The vertices x, y have a common neighbour t neither in $N(u)$ nor in $N(v)$ because $\{t, x, y\}$ would induce K_3 which would be coloured by the **Extension** (R5, R6). Let u_x, u_y and v_x, v_y be the neighbours of x and y in $N(u)$ and $N(v)$, respectively (according to rule (R4) $u_x, u_y \in R$ and $v_x, v_y \in B$). It is obvious that there are no bichromatic edges between u_x, u_y, v_x, v_y, otherwise x, y would be coloured by the **Extension** (R4). Now suppose that $c(x) = r$ and $c(y) = b$. We get a contradiction since y would have two neighbours u_y and x coloured in red. Analogous situation occurs if $c(x) = b$ and $c(y) = r$. Hence Q is monochromatic.

Claim 3. *If the* **Extension** *subroutine of the* **E Algorithm** *was executed and it was not decided whether a graph G has a matching cutset and the existing colouring can be properly extended, then* $|V(Q)| \leq 2$ *for each connected component Q of the graph G such that* $V(Q) \subseteq \mathcal{U}$.

Proof. Suppose that, contrary to our Claim, $|V(Q)| \geq 3$. Hence, there must exist a path P_3 on three vertices with $V(P_3) \subseteq Q$ (not necessarily as an induced subgraph). Let $x, y, z \in V(P_3)$ and $xy, yz \in E$. The vertices x, y (y, z) have a common neighbour t neither in R nor in B because $\{t, x, y\}$ $(\{t, y, z\})$ would induce K_3 which would be coloured by the **Extension** (R5, R6). So we must consider three cases. Let $e = uv \in E_2$ and $c(u) = r$, $c(v) = b$. Furthermore let $u_x, u_y, u_z \in N(u)$ be the neighbours of x, y, z, respectively in R and $v_x, v_y, v_z \in N(v)$ be the neighbours of x, y, z, respectively in B. Please notice that there are no bichromatic edges between u_x, u_y, u_z and v_x, v_y, v_z, otherwise x, y, z would be coloured by the **Extension** (R4).
Case 1 (see Fig. 1). u_x, u_y, u_z and v_x, v_y, v_z are pairwise distinct. From $dist(u_x, v_y) = 2$ and $dist(u_x, v_z) = 2$ it follows that there must exist vertices $z', z'' \in \mathcal{U}$ such that $u_xz', v_yz', u_xz'', v_zz'' \in E$. It is obvious that $z', z'' \in \mathcal{U}$ otherwise x, y, z would be coloured by the **Extension** (R4 and R2 or R3). What is more $z' \neq z''$ or else it would have to be coloured (it would have two neighbours coloured in blue). From Claim 2 we know that Q must be monochromatic. Hence, let $c(x) = c(y) = c(z) = r$. Then the edges xv_x, yv_y, zv_z belong to E_2 and z', z'' must be coloured in blue. But in that case u_x would have two neighbours coloured in blue and we get a contradiction with the existence of a matching cutset. Similarly we get a contradiction when $c(x) = c(y) = c(z) = b$.

Case 2 (see Fig. 2). Let x, z have exactly one common neighbour, i.e. $v_x = v_z$ and u_x, u_y, u_z are pairwise distinct. If we colour x, y, z in red, then v_x would have two red neighbours and we would not obtain a matching cutset. So $c(x) = c(y) = c(z) = b$. Since $diam(G) = 2$ there must exist vertices $w, w', w'' \in \mathcal{U}$ such that $wu_x, wv_y, w'u_y, w'v_x, w''u_z, w''v_y \in E$. Hence w, w', w'' must be coloured in red otherwise u_x, u_y, u_z would have two neighbours coloured in blue. But for such colouring we have a vertex v_y which has two neighbours coloured in red, i.e. w and w''. Similarly we get a contradiction when $u_x = u_z$ and v_x, v_y, v_z are pairwise distinct.

Case 3. Let us suppose that vertices x and z have two common neighbours, i.e. $u_x = u_z$ and $v_x = v_z$. It is easy to see that neither red nor blue colour of the component Q would give us a matching cutset. Hence $|V(Q)| \leq 2$.

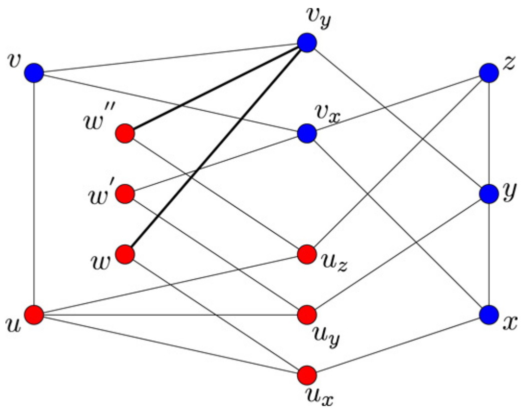

Fig. 2. The colouring of the vertices x, y, z implies the colouring of the vertices w, w', w''. Bolded edges contradict the existence of the matching cutset. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

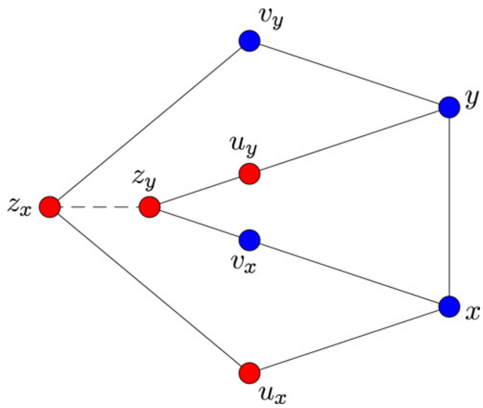

Fig. 3. The colouring of the vertices x, y implies the colouring of the vertices z_x, z_y. Dashed edge can belong to E. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Claim 4. *If the* **Extension** *subroutine of the* **E Algorithm** *was executed and it was not decided whether a graph G has a matching cutset and the existing colouring can be properly extended, then each connected component Q of the graph G such that* $V(Q) \subseteq \mathcal{U}$ *and* $|V(Q)| = 2$ *can be coloured either in red or in blue.*

Proof. Let $x, y \in \mathcal{U}$ and $xy \in E$. x, y have a common neighbour t neither in B nor in R or else $\{t, x, y\}$ would induce K_3 which would be coloured by the **Extension** (R5, R6). So let u_x, u_y and v_x, v_y be the neighbours of x, y in R and B, respectively. Vertices v_x, v_y, u_x, u_y are not adjacent to the vertices in the opposite colour otherwise x, y would be coloured (R4). Because of this and the fact that $diam(G) = 2$, there must exist vertices $z_x, z_y \in \mathcal{U}$ (if $z_x, z_y \notin \mathcal{U}$ then x, y would be coloured by the **Extension** (R4, R2, R3)) such that $u_x z_x, z_x v_y, z_y u_y, z_y v_x \in E$.
Furthermore $z_x \neq z_y$ otherwise it would have two neighbours coloured in red and two coloured in blue. Now, it is easy to see that because of the symmetry of the problem and Claim 2 we can assign either $c(x) = c(y) = r$ and $c(z_x) = c(z_y) = b$ or $c(x) = c(y) = b$ and $c(z_x) = c(z_y) = r$ to properly extend our colouring (see Fig. 3).

The **Pairs** subroutine is based on the above claims. Time complexity of this subroutine is $O(n^2m^2)$.

Pairs

Input: a graph G with the partial colouring of the vertices which was made by the **Extension** subroutine,
Output: the matching cutset of the graph G or a message that the graph does not have a matching cutset or a message that the given colouring cannot be extended.

```
if there exist at least three uncoloured vertices which induce a connected subgraph of the graph G then
    if you have started the colouring with an edge e ∈ S_1 then
        return a message that there does not exist a matching cutset;
        stop E Algorithm;
    else
        remove the existing colouring;
        return a message that given colouring cannot be extended;
else
```

while there exist uncoloured vertices do
take an arbitrary edge with two uncoloured end vertices and colour its end vertices in red;
call **Extension**;
if the **Extension** returned a message that given colouring cannot be extended then
return a message that given colouring cannot be extended;
stop **Pairs**;
return E_2 as a matching cutset;
stop **E Algorithm**;
end **Pairs**;

It is easy to see that E_2 defined by the **E Algorithm** is a matching and a cutset of the graph G.

Let G has an odd order. In this situation we will use the already mentioned **O Algorithm**. It is easy to see that its construction and correctness follows directly from Lemma 2 and its time complexity is bounded by $O(n^2m)$.

O Algorithm
Input: a graph $G \in \mathcal{P}(2)$ of order n such that $\Delta(G) \leq \frac{n-1}{2}$ and n is odd,
Output: either a matching cutset or a message that it does not exist.

if there is no vertex v of degree $\frac{n-1}{2}$ then
return a message that there does not exist a matching cutset;
else
for each vertex v of degree $\frac{n-1}{2}$ do
colour v and every vertex in $N(v)$ in red and all the other vertices in blue;
if E_2 is a matching cutset then
return E_2 as a matching cutset;
stop **O Algorithm**;
return a message that there does not exist a matching cutset;
end **O Algorithm**;

Depending on the number of vertices of a graph $G \in \mathcal{P}(2)$ we can use either the **O Algorithm** or the **E Algorithm** to decide whether G has the matching cutset. Time complexity of both of these algorithms is polynomial and their correctness follows from previous Lemmata and Claims. ■

Remark 2. If in a graph G there is at most one vertex of degree 3 or $\delta(G) \geq 4$, then all the vertices of the graph G will be coloured by the **Extension** subroutine of the **E Algorithm**.

3. Graphs of diameter two with nontrivial module

We now turn our attention to graphs which possess nontrivial module.

Theorem 3. *If $diam(G) = 2$ and G is not prime, then we can decide in polynomial time whether G has a matching cutset.*

Proof. In our proof we will use a modular decomposition of the graph G. Once more we will consider only graphs with $\delta(G) \geq 2$.

Part 1. Suppose that G has a module W^* such that W^* satisfies one of the following conditions:

(1) $|W^*| \geq 3$,
(2) $W^* = \{x, y\}$ and $xy \in E$,
(3) $W^* = \{x, y\}$ and $xy \notin E$ and $|N(W^*)| \geq 3$,
(4) $W^* = \{x, y\}$ and $xy \notin E$ and $N(W^*) = \{u, v\}$ and $uv \in E$,
(5) $W^* = \{x, y\}$ and $xy \notin E$ and $N(W^*) = \{u, v\}$ and $(N(u) - W^*) \cap (N(v) - W^*) \neq \phi$.
(6) $W^* = \{x, y\}$ and $xy \notin E$ and $N(W^*) = \{u, v\}$ and u or v is adjacent to a different module W^{**}.

It is easy to see that if W^* satisfies one of the conditions (1)–(5) then every vertex $v \in N[W^*]$ must have the same colour. Let us consider the case when W^* satisfies only the condition (6) (see Fig. 4). Let u be adjacent to module $W^{**} = \{w_1, w_2\}$ ($W^* \neq W^{**}$). Without loss of generality we can suppose that we would colour x, u in red and y, v in blue. Then we have to colour w_1 and w_2 in red. Because $diam(G) = 2$ and W^* does not satisfy any of the conditions (1)–(5), there must exist a vertex $t \in N(v) - W^*$ which is adjacent to W^{**} (otherwise $dist(w_1, v) = dist(w_2, v) \geq 3$). t must be coloured in red because it has two neighbours w_1 and w_2 coloured in red. Now we have the required contradiction since v has two neighbours coloured in red, i.e. x and t.

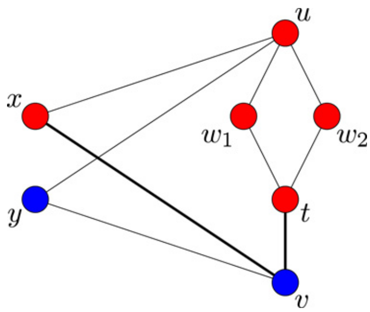

Fig. 4. W^* satisfies only the condition (6). Bolded edges contradict the existence of the matching cutset. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

So without loss of generality we can colour in red all the vertices in $N[W^*]$. According to the fact that no vertex can have two neighbours in the opposite colour, we must colour in red all the neighbours of coloured modules of order at least 2. Furthermore we must colour in red the modules of order at least 3 which are adjacent to a coloured vertex. We also colour in red every module and vertex which has at least two coloured neighbours. We must repeat these operations as long as we can. After that every uncoloured module or vertex has exactly one neighbour coloured in red (if this condition would not be satisfied we would get a contradiction with the fact that $diam(G) = 2$) and the order of these modules will be at most 2.

Claim 5. *Every remaining uncoloured module of order* 2 *must be coloured in red.*

Proof. Let $P = \{x, y\}$ be an uncoloured module of order 2 and $z = N(P) \cap R$. Firstly suppose that $xy \notin E$. Because $\delta(G) \geq 2$ and $|N(P) \cap R| = 1$, then x, y must have an uncoloured neighbour t. Also t must have a neighbour $m \in R$, otherwise we would have a contradiction with the fact that $diam(G) = 2$. If $z = m$, then xz, xt, tz and zt, ty, yz induce triangles in G so x, y, t must be coloured in red. Now suppose that $z \neq m$. If we colour x (y) in blue, then we must colour t in blue and y (x) in red. But in this situation t would have two neighbours m and y coloured in red. So now suppose that $xy \in E$. Then xy, xz, yz induce a triangle in G. Hence x, y must be coloured in red.

So we must colour every uncoloured module of order 2 and all his neighbours in red. After this operation all the remaining uncoloured vertices do not belong to any module. It is obvious that we can also colour in red every pair of vertices x, y such that $xy \in E$ and $N(x) \cap R = N(y) \cap R$ and every vertex v, such that $|N(v) \cap R| \geq 2$. After these operations every vertex $v \in \mathcal{U}$ has exactly one neighbour in R.

Claim 6. *The graph G has a matching cutset if and only if there does not exist a vertex $v \in R$ such that $|N(v) \cap \mathcal{U}| > 1$.*

Proof. ($\Rightarrow$) Suppose that G has a matching cutset and there exists a vertex $v \in R$ such that $|N(v) \cap \mathcal{U}| > 1$. Let $A = N(v) \cap \mathcal{U}$. For every $x, y \in A$ $xy \notin E$ or else they would be already coloured in red. Since $\delta(G) \geq 2$ and $diam(G) = 2$, every vertex $x \in A$ has at least one uncoloured neighbour which does not belong to A. Furthermore, if x is not adjacent to a vertex p which belong to the set $\mathcal{U} - A$, then there exists a vertex $t \in \mathcal{U} - A$ such that $xt, tp \in E$. From these observations follows that if we colour every vertex $x \in A$ in red, then we have to colour in red all the other vertices. This is a clear contradiction of the fact that G has a matching cutset. So suppose that we would colour in blue a vertex $x \in A$. All the other vertices in A would had to be coloured in red. Let y be one of these vertices.

Firstly suppose that x and y have a common neighbour $t \in \mathcal{U}$ ($t \notin A$) and r be a neighbour of t which belongs to the set $(N(t) \cap R) - A$. From previous observations we know that such vertex exists, what is more r is the only one vertex which belongs to the set $(N(t) \cap R) - A$. Now t must be coloured in blue (otherwise x would have two neighbours coloured in red) but then t has two neighbours coloured in red, i.e. y and r. Hence suppose that x and y do not have a common neighbour (see Fig. 5).

Let $t \in N(x) \cap \mathcal{U}$ ($t \notin A$). Since $diam(G) = 2$ there must exist a vertex $z \in N(y) \cap \mathcal{U}$ ($z \notin A$) such that $zt \in E$. As previously t and z must have a neighbour $r_1 \in (N(t) \cap R) - A$ and $r_2 \in (N(z) \cap R) - A$, respectively. Because x is coloured in blue, it follows that t and z must be coloured in blue. Now we have the required contradiction since z has two neighbours r_2 and y coloured in red.

($\Leftarrow$) It is easy to see that if there does not exist a vertex $v \in R$ such that $|N(v) \cap \mathcal{U}| > 1$, then we can colour every vertex in $\mathcal{U}$ in blue to obtain a matching cutset.

Part 2. Suppose that G has no modules which satisfy any of the conditions (1)–(6) of Part 1. Then G has exactly one module $W^* = \{x, y\}$ and $N(W^*) = \{u, v\}$. Without loss of generality we can colour x, u in red and y, v in blue. Now we must colour all the vertices in $N(u)$ in red and all the vertices in $N(v)$ in blue. It is obvious that after this colouring every vertex is coloured. If E_2 is a matching cutset, then we obtained the required solution. So suppose that E_2 is not a matching cutset. Then we must colour every vertex in $N[W^*]$ in red. As previously we can also colour in red every pair of vertices x, y such that $xy \in E$ and $N(x) \cap R = N(y) \cap R$ and every vertex v, such that $|N(v) \cap R| \geq 2$. After that we can use Claim 6 to decide whether G has a matching cutset. ■

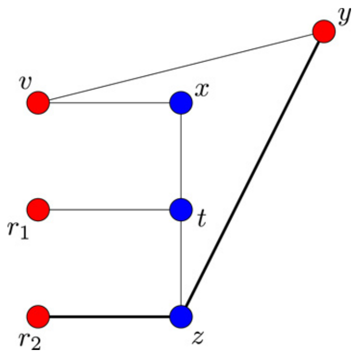

Fig. 5. x and y do not have a common neighbour. Bolded edges contradict the existence of the matching cutset. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

We can now formulate our main result which is obtained by combining the two previous Theorems.

Theorem 4. MATCHING CUTSET *is polynomial for a graph G of diameter* 2*.*

From this Theorem and Theorem 1 follows the next one.

Theorem 5. *If* $diam(L(G)) = 2$*, then* STABLE CUTSET *is polynomial for the graph G.*

4. Conclusion

In this paper we proved that MATCHING CUTSET is polynomial for graphs of diameter 2. This result also gave us the information about complexity of STABLE CUTSET for special graph class, i.e. if an input graph G has a line graph of diameter 2, then the problem is polynomial. An interesting question arises.

Problem 1. What is the greatest value of the diameter of the graph for which MATCHING CUTSET is solvable in polynomial time?

Acknowledgements

The authors would like to thank anonymous reviewer(s) for valuable comments.

References

[1] A. Brandstäd, F. Dragan, V.B. Le, T. Szymczak, On stable cutsets in graphs, Discrete Appl. Math. 105 (2000) 39–50.
[2] C. Barefoot, K. Casey, D. Fisher, K. Fraughnaugh, F. Harary, Size in maximal triangle-free graphs and minimal graphs of diameter 2, Discrete Math. 138 (1995) 93–99.
[3] J.A. Bondy, P. Erdős, S. Fajtlowicz, Graphs of diameter two with no 4-circuits, Discrete Math. 200 (1999) 21–25.
[4] D.G. Coreli, J. Fonlupt, Stable set bonding in perfect graphs and parity graphs, J. Combin. Theory (B) 59 (1993) 1–14.
[5] A. Courier, M. Habib, A new linear algorithm for modular decomposition, in: S. Tison (Ed.), Proceedings of the 19th International Colloquium on Trees in Algebra and Programming, in: Lecture Notes in Computer Science, vol. 787, 1994, pp. 68–82.
[6] V. Chvátal, Recognizing decomposable graphs, J. Graph Theory 8 (1984) 51–53.
[7] E. Dahlhaus, J. Gustedt, R.M. McConnell, Efficient and practical algorithms for sequential modular decomposition, J. Algorithms 41 (2001) 360–387.
[8] A. Hellwig, L. Volkmann, Sufficient conditions for λ'-optimality in graphs of diameter 2, Discrete Math. 283 (2004) 113–120.
[9] J. Huang, A. Yeo, Maximal and minimal vertex-critical graphs of diameter two, J. Combin. Theory B 74 (1998) 311–325.
[10] V.B. Le, B. Randerath, On stable cutsets in line graphs, Theoret. Comput. Sci. 301 (2003) 463–475.
[11] R.M. McConnell, J.P. Spinrad, Linear-time modular decomposition and efficient transitive orientation of comparability graphs, in: Proceedings of the Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA'94, ACM Press, New York, 1994, pp. 536–545.
[12] R.M. McConnell, J.P. Spinrad, Modular decomposition and transitive orientation, Discrete Math. 201 (1999) 189–241.
[13] A.M. Moshi, Matching cutsets in graphs, J. Graph Theory 13 (1989) 527–536.
[14] A. Tucker, Coloring graphs with stable cutsets, J. Combin. Theory (B) 34 (1983) 258–267.