Disjoint cycles intersecting a set of vertices

M. Pontecorvi[*], P. Wollan[†]

May 28, 2012

Abstract

A classic theorem of Erdős and Pósa states that there exists a constant c such that for all positive integers k and graphs G, either G contains k vertex disjoint cycles, or there exists a subset of at most $ck \log k$ vertices intersecting every cycle of G. We consider the following generalization of the problem: fix a subset S of vertices of G. An *S-cycle* is a cycle containing at least one vertex of S. We show that again there exists a constant c' such that G either contains k disjoint S-cycles, or there exists a set of at most $c'k \log k$ vertices intersecting every S-cycle. The proof yields an algorithm for finding either the disjoint S-cycles or the set of vertices intersecting every S-cycle. An immediate consequence is an $O(\log n)$-approximation algorithm for finding disjoint S-cycles.

1 Introduction

A classic theorem of Erdős and Pósa states that every graph G either contains many disjoint cycles or there exists a bounded subset of vertices intersecting every cycle in G. Specifically, they give the following theorem.

Theorem 1 ([8]) *There exists a constant c such that for all graphs G and for all positive integers k, G either contains k vertex disjoint cycles, or there exists a set $X \subseteq V(G)$ with $|X| \leq ck \log k$ such that $G - X$ contains no cycle.*

Erdős and Pósa also show that the bound on the size of the set X is best up to the value of the constant c, as for all positive integers k, there exist graphs G_k and a constant c' such that G_k does not contain either k disjoint cycles nor does G_k contain a set X of size at most $c'k \log k$ intersecting all the cycles.

In this paper, we consider the following question. Let G be a graph and S be a subset of the vertices. An *S-cycle* in G is a cycle containing at least one vertex in S. An *S-cycle vertex (edge) hitting set* is a subset $X \subseteq V(G)$ ($X \subseteq E(G)$) such that every S-cycle contains at least one vertex (edge) of X. We consider the extension of Erdős and Pósa's theorem to S-cycles. Kakimura, Kawarabayashi and Marx [12] give the first result along these lines, showing that for every graph G and set S of vertices, either there exist k disjoint S-cycles or there exists an S-cycle vertex hitting set of size at most $40k^2 \log_2 k$. The main result of this article, given in the next theorem, shows that we can achieve the same bound on the hitting set as in the original theorem of Erdős and Pósa. The proof yields an algorithm for finding such cycles or a hitting set. Throughout the article, when we refer to algorithms for a fixed graph, we will use n to indicate the number of vertices and m for the number of edges of the graph.

[*]Department of Computer Science, University of Austin, TX.

[†]Department of Computer Science, University of Rome, "La Sapienza" Via Salaria 113, Rome, Italy 00198. wollan@di.uniroma1.it The work of this author was partially supported by the European Research Council under the European Unions Seventh Framework Programme (FP7/2007-2013)/ERC grant agreement no. 279558.

Theorem 2 *There exists a constant c_v such that for all graphs G, $S \subseteq V(G)$, and positive integers k, either there exist k vertex disjoint S-cycles, or there exists an S-cycle vertex hitting set X with $|X| \leq c_v k \log k$. There exists an $O(nm)$-time algorithm which given in input G, S and k, returns either k vertex disjoint S-cycles or an S-cycle vertex hitting set of size at most $c_v k \log k$.*

The bound of $c_v k \log k$ on the size of the hitting set is again best possible, as we can consider the same graphs G_k showing Theorem 1 has the optimal bound and fix $S := V(G_k)$.

This paper is organized as follows. In the next section, we see that the analogous statement for edge disjoint S-cycles follows by a simple construction from the vertex disjoint version. We also present an approximation algorithmic result which follows immediately as a consequence of Theorem 2. Section 3 contains the proof of Theorem 2.

2 Related questions

In this section, we first consider the problem of finding edge disjoint cycles instead of vertex disjoint cycles. We then move on to an application of Theorem 2.

2.1 Edge versus vertex disjoint cycles

Historically, when looked at from an algorithmic point of view, the problem of finding edge disjoint (instead of vertex disjoint) cycles has perhaps received more attention. Here we note that the natural edge disjoint version of Theorem 2 follows by an easy construction.

Corollary 1 *There exists a constant c_e such that the following holds for all graphs G, $S \subseteq V(G)$, and positive integers k. Either there exist k edge disjoint S-cycles, or there exists an S-cycle edge hitting set $Z \subseteq E(G)$ with $|Z| \leq c_e k \log k$. There exists an $O(nm^2)$-time algorithm which takes as input G, S, and k, and outputs either k edge disjoint S-cycles, or the hitting set $Z \subseteq E(G)$ with $|Z| \leq c_e k \log k$.*

Proof. Let G, S, and k be given. For all vertices $v \in V(G)$, let $d(v)$ be the degree of v. We define disjoint sets of vertices $X_v = \{x_v^1, x_v^2, \dots, x_v^{d(v)+1}\}$ for all vertices $v \in V(G)$. For every edge uv in G, we define new vertices x_{uv}^u, x_{uv}^v and x_{uv}^{mid}. Let $D = \{x_{uv}^u, x_{uv}^v, x_{uv}^{mid} : uv \in E(G)\}$. We let $\bar{G}$ be the graph with vertex set

$$V(\bar{G}) = \bigcup_{v \in V(G)} X_v \cup D$$

and edge set

$$\begin{aligned} E(\bar{G}) = &\{x_{uv}^v x_v^i, x_{uv}^u x_u^j : uv \in E(G) \text{ and } 1 \leq i \leq d(v)+1, 1 \leq j \leq d(u)+1\} \\ &\cup \{x_{uv}^u x_{uv}^{mid}, x_{uv}^v x_{uv}^{mid} : uv \in E(G)\}. \end{aligned}$$

Thus $\bar{G}$ is obtained from G by subdividing every edge exactly 3 times, and then repeatedly replicating each vertex v of the original graph exactly $d(v)$ times. We let $\bar{S} := \{x_{uv}^{mid} : \text{ either } u \text{ or } v \in S\}$. For every cycle in $\bar{G}$ which uses at least one vertex of the form x_{uv}^{mid}, there is a corresponding cycle in G. Moreover, if we consider two vertex disjoint cycles, each of which uses a vertex of the form x_{uv}^{mid}, then they will correspond to edge disjoint cycles in the original graph G. Also, note $|V(\bar{G})| \leq \sum_{v \in V(G)}(d(v)+1) + 3|E(G)| \leq 6|E(G)|$ and $|E(\bar{G})| = \sum_{v \in V(G)} d(v)(d(v)+1) + 2|E(G)| \leq 4|V(G)||E(G)|$.

By Theorem 2, either there exist k vertex disjoint $\bar{S}$-cycles in $\bar{G}$, corresponding to k edge disjoint S-cycles in G, or there exists a hitting set $X \subseteq V(\bar{G})$ of the $\bar{S}$-cycles. We claim that we can assume $x_v^i \notin X$ for all $v \in V(G)$, $1 \leq i \leq d(v)+1$. Assume such a vertex x_v^i is contained in X. If there exists an index $j \neq i$ such that $x_v^j \notin X$, then $X \setminus \{x_v^i\}$ is also a

$\bar{S}$-cycle hitting set. Alternatively, if no such j exists, then the set $X' := (X \setminus \{x_v^i : 1 \leq i \leq d(v)+1\}) \cup \{x_{uv}^v :$ u is adjacent v in G $\}$ is a $\bar{S}$-cycle hitting set on strictly fewer vertices. We conclude that we may assume every vertex in X is of the form x_{uv}^u, x_{uv}^v, or x_{uv}^{mid} for some edge uv. Thus, X directly yields a set of edges in G intersecting every S-cycle. We also have the desired algorithm which either finds the disjoint cycles or the hitting set. The bound on the running time follows from the bounds on the size of $\bar{G}$ and the run time given in Theorem 2. This completes the proof of the corollary. □

2.2 Approximation algorithms for finding disjoint S-cycles

The problem of finding a maximum number of either edge or vertex disjoint cycles in a graph has been widely studied, and it is known to be NP-hard, even in the special case when attempting to determine if the edge set of a graph can be decomposed into edge disjoint triangles [7]. This has lead to the study of approximation algorithms to find many disjoint cycles. An *α-approximation algorithm* (for a maximization problem) returns an answer at least $1/\alpha$ of the optimal value. The value α is called the *approximation factor* of the algorithm.

Caprara, Panconesi, and Rizzi [3] showed that a modified greedy algorithm yields a $O(\log n)$-approximation algorithm to find a maximum set of edge disjoint cycles. Krivelevich et al. [14] later improved the analysis to yield an $O(\log^{1/2} n)$-approximation factor. This is essentially best possible, as Friggstad and Salavatipour [10] show that for all $\epsilon > 0$ it is impossible to achieve a $\Omega(\log^{1/2-\epsilon} n)$-approximation algorithm unless $NP \subseteq DTIME(n^{polylog(n)})$.

If we consider the problems of finding a maximum set of edge or vertex disjoint S-cycles, the directed variant of the problem has been studied. In [14], the authors show that there exists an $O(n^{\frac{2}{3}})$-approximation algorithm to find a maximum number of edge disjoint S-cycles in directed graphs. In a recent result, Kamikura and Kawarabayashi [11] have shown that while there does not exist an analogous theorem to Theorem 2 in directed graphs, there does exist a function $f(k)$ such that for any directed graph D and subset S of the vertices, either there exist k directed S-cycles such that every vertex of D is in at most five of the cycles, or there exists a set of at most $f(k)$ vertices intersecting every such S-cycle.

An immediate corollary to Theorem 2 and Corollary 1 is the following approximation algorithm.

Corollary 2 *There exists an $O(\log n)$-approximation algorithm to find the maximum number of (either vertex or edge) disjoint S-cycles in a graph.*

We quickly outline the proof. Given a graph G and $S \subseteq V(G)$, we repeatedly apply the algorithm given in Theorem 2 to find an integer k and disjoint S-cycles $C_1, C_2, \ldots, C_k$ such that there do not exist $k+1$ disjoint S-cycles. Therefore, there exists an S-cycle hitting set of size at most $c(k+1)\log(k+1)$, and thus our set $C_1, C_2, \ldots, C_k$ of disjoint S-cycles is within a factor of $c' \log(k) \leq c' \log n$ of the optimal value for some slightly larger constant c'. It would be interesting to know if the analysis of the algorithm can be improved, as in [14], to give a $\log^{1/2} n$-approximation factor.

The dual problem of finding a minimal set of vertices intersecting every cycle, known as the feedback vertex set, is a classic problem in the theory of computing. There exists a 2-approximation algorithm for the feedback vertex set problem as well as its natural weighted generalization. See [1], [2], [4]. The S-feedback vertex set problem seeks a minimum set of vertices intersecting every S-cycle. There exists an 8-approximation algorithm for the problem, due to Even, Noar, and Zosin [9] and the problem has recently been shown to be fixed parameter tractable by Cygan et al [5] and independently by Kawarabayashi and Kobayashi [13].

3 Proof of Theorem 2

In this section, we give the proof of Theorem 2. We follow the main idea of Simonovits' [15] (see also [6]) proof of the original theorem of Erdős and Pósa, by generalizing it to consider the case when we are only interested in cycles that intersect a fixed set of vertices. The proof of Simonovits immediately gives a fast algorithm which either finds disjoint cycles or a bounded size hitting set. Our proof also gives an algorithm, although we must work slightly harder to extract it.

Before proceeding, we will need several definitions. Let G be a graph. Given a path P in the graph, the subpath of P with endpoints x and y in $V(P)$ is denoted xPy. If we let $X \subseteq V(G)$ be a subset of the vertices, then P is an X*-path* if P has at least two vertices, both endpoints of P are in X, and P has no internal vertex in X. Similarly, if H is a subgraph of G, then an H-path is a $V(H)$-path which is not an edge of H. Given a second path Q, we say that P and Q are *internally disjoint* if no internal vertex of P is contained in $V(Q)$ and vice versa. Finally, given a second graph J, we denote by $G \cup J$ the graph with vertex set $V(G) \cup V(J)$ and edge set $E(G) \cup E(J)$.

We define a $(2,3)$*-graph* to be a graph with minimum degree two and maximum degree three. Let G be a $(2,3)$-graph. We will refer to the set $B := \{v \in V(G) : \deg(v) = 3\}$ as the *branch vertices* of G. A *segment* of G is either a B-path or a cycle forming a component of G. Let $S \subseteq V(G)$ be a subset of the vertices. A subset $Z \subseteq V(G)$ is an S*-segment hitting set* if for every segment P of G such that $(V(P) \setminus B) \cap S \neq \emptyset$, there exists $z \in Z \cap (V(P) \setminus B)$ such that $z \in S$. We will specifically focus on $(2,3)$-graphs where every cycle in the graph is an S-cycle. Call such a graph a $(2,3)$ S*-cycle* graph.

The proof of Simonovits proceeds by considering a $(2,3)$-subgraph H in a graph G which is maximal by containment as a subgraph. If H is sufficiently large (in terms of the number of branch vertices) then H will contain k disjoint cycles. Otherwise, one can find a hitting set for the cycles contained in H using the maximality of H. In our proof, we will instead consider a maximal $(2,3)$-subgraph where every cycle in the subgraph is an S-cycle. Thus, in order to use the algorithm given by the proof, we need to show that one can efficiently find such a maximal $(2,3)$ S-cycle subgraph. We first give a characterization of maximal $(2,3)$ S-cycle subgraphs.

Lemma 3 *Let G be a graph and $\Sigma \subseteq E(G)$. Let H be a $(2,3)$ S-cycle subgraph of G, and let B be the branch vertices of H. Then there exists a $(2,3)$ S-cycle subgraph H' which strictly contains H as a subgraph if and only if one of the following holds.*

1. *There exists an S-cycle C in $G - V(H)$.*
2. *There exists an $(H - B)$-path P in $G - B$ such that $V(P) \cap S \neq \emptyset$.*
3. *Let $Y_1, Y_2, \dots, Y_l$ the vertex sets of the components of $H - S$. There exists an $(H-B)$-path P in $G - B$ with endpoints in distinct pieces of the partition $Y_1 \setminus B, Y_2 \setminus B, \dots, Y_l \setminus B$.*

Proof. It is easy to see that if the cycle C in 1 or the path P in 2 or 3 exists, then either $H \cup C$ or $H \cup P$ is a $(2,3)$ S-cycle subgraph. Thus, it remains to show that if there exists a $(2,3)$ S-cycle subgraph H' strictly containing H, then one of the outcomes $1-3$ holds. Fix such a subgraph H'. If H' contains a cycle which is disjoint from $V(H)$, then 1 holds. Otherwise, there exists a $(H - B)$-path P. We may assume that $V(P) \cap S = \emptyset$, lest 2 hold. If P violates the condition for 3, then there exists a path Q linking the endpoints of P in $H - S$. But then $P \cup Q$ is a cycle contained in $H \cup P \subseteq H'$ which does not intersect S, a contradiction. This completes the proof of the lemma. □

In order to take advantage of Lemma 3, we will need to find either a cycle C or path P satisfying one of the possibilities $1-3$ in the statement of Lemma 3. We will give a linear time algorithm to do so which depends on the so called block decomposition of the graph. Let G be a graph. Recall that a *block* of G is a maximal subgraph containing no cut-vertex. As shown by Tarjan [16], the blocks of a connected graph intersect to form a tree-structure called the *block decomposition*. Moreover, such a block decomposition can be found in $O(m)$ time. In the same work, Tarjan shows that given a vertex x and a pair of vertices $\{u, v\}$, if there does not exist a cut vertex separating x from $\{u, v\}$, then in time $O(m)$, one can find two internally disjoint paths from x to the set $\{u, v\}$ with distinct ends in $\{u, v\}$.

Lemma 4 *Let G be a graph, and let $S \subseteq V(G)$ be a subset of the vertices. There exists an $O(nm)$-time algorithm which finds a maximal (by containment as a subgraph) $(2,3)$ S-cycle subgraph H.*

Proof. Let H be a $(2,3)$ S-cycle subgraph of G. We will see that in time $O(m)$, we can find either a path or cycle satisfying 1, 2, or 3 in the statement of Lemma 3, or alternately determine that no such path or cycle exists. Thus, by iterating this process at most $O(n)$ times, we find a maximal by subgraph containment $(2,3)$ S-cycle subgraph of G. Fix B to be the set of branch vertices of H.

First, in time $O(m)$, we can find block decompositions of the components of $G - V(H)$. If some block contains a vertex of S and at least three vertices total, then by Tarjan's algorithm for finding two internally disjoint paths, we can find an S-cycle avoiding $V(H)$ satisfying 1. If no such block exists, then no such S-cycle can exist avoiding $V(H)$.

We now attempt to find a path satisfying 2. Again, in time $O(m)$, we find block decompositions of the components of $(G - B) - E(H)$. Each leaf L of the block decomposition has a unique vertex x_L which is a cut vertex separating $L - x_L$ from the rest of the graph $(G - B) - E(H)$. By repeatedly deleting the set of vertices $V(L) - x_L$ of leaves L of the block decompositions for which $L - x_L$ does not contain a vertex of $V(H) - B$, we find the maximum subgraph J of $(G - B) - E(H)$ such that for any vertex $x \in V(J) - V(H)$, there does not exist a cut vertex separating x from the set $V(H) - B$. Note that if there exists a path satisfying 2, it must be contained in the subgraph J. If there exists a component of J containing at least two vertices of $\{u, v\} \subseteq V(H) - B$ and a vertex $x \in S$, we can find two paths from x to $\{u, v\}$ in time $O(m)$, yielding a $V(H) - B$ path containing a vertex of S as desired by 2.

Finally, we find the partition of $V(H) - S$ into subsets $Y_1, \ldots, Y_l$ inducing connected components of $H - S$ for some value l. In time $O(m)$, we can determine the connected components of $(G - B) - E(H)$ and specifically determine if there exists a component K containing vertices from Y_i and Y_j for $i \neq j$. If such a component K and indices i and j exist, then there exists a path satisfying either 2 or 3 in the statement of Lemma 3. If no such component and pair of indices exist, then there does not exist an $H - B$ path satisfying 3. We conclude that H is a maximal $(2,3)$ S-cycle subgraph, as desired. □

The next lemma contains the majority of the work in the proof of the main theorem.

Lemma 5 *Let G be a graph and $S \subseteq V(G)$. Let k be a positive integer. Let H be a maximal $(2,3)$-subgraph such that every cycle contained in H is an S-cycle. Let B be the set of branch vertices of H. Then either G contains k disjoint S-cycles, or there exists an S-cycle hitting set Z with $|Z| \leq \frac{5}{2}|B| + 2k$. There exists an $O(nm)$-time algorithm which given in input G, H, S and k outputs either the k disjoint cycles or the hitting set Z.*

Proof. Let G, S, and H be given. Let B be the set of branch vertices of H. Note by Lemma 3, there does not exist either an S-cycle C which is disjoint from $V(H)$, nor an $(H - B)$-path P in $G - B$ such that $V(P) \cap S \neq \emptyset$.

Let X be an inclusion-wise minimal S-segment hitting set of H. Note, that X contains at most one vertex in each segment of H, and exactly one from each segment of length at least two which intersects S in a vertex of $V(P) \setminus B$. Note that H has at most k components which are cycles and at most $\frac{3}{2}|B|$ segments which are paths. Consequently, $|X| \leq \frac{3}{2}|B| + k$.

Consider the graph $H - (B \cup X)$. Every component is a subpath of some segment of H. Note that here we are using the property that each segment of H which is a cycle must be an S-cycle and therefore contains a vertex of X. We will see that the set $B \cup X$ is almost an S-cycle hitting set. Every S-cycle which does not intersect $B \cup X$ can only intersect the graph H in a limited manner. For a subset $A \subseteq V(H)$, we will refer to the vertices of H which have a neighbor (in H) in the set A as the *H-neighbors of A*.

Claim 6 *There does not exist an S-cycle C disjoint from $B \cup X$ which intersects $V(H)$ in at least two vertices.*

Proof. Assume the claim is false and pick such a C to minimize $E(H - (B \cup X)) \cup E(C)$. We will derive a contradiction to the maximality of H.

As a case, assume that there exists an $(H - (B \cup X))$-path P contained in C such that P has both endpoints in the same component of $H - (B \cup X)$. Let the endpoints of P be x and y, and let the component of $H - (B \cup X)$ containing x and y be labeled Q. The path P does not contain a vertex of S, lest we violate the maximality of H. Assume that the subpath xQy does contain a vertex of S. We claim that in this case, $H \cup P$ violates our choice of H maximal. The cycle $P \cup xQy$ is an S-cycle. Any other cycle of $H \cup P$ which contains P must contain the H-neighbors of $V(Q)$ in $B \cup X$. Since Q contains a vertex of S, it follows that at least one neighbor of Q in $B \cup X$ must be in the set S. Thus, every cycle of $H \cup P$ contains a vertex of S, a contradiction.

It follows that the path xQy does not contain a vertex in S. Let R be the subpath of C connecting x and y which does not contain the edges $E(P)$. Then $R \cup xQy$ is a closed walk, which may or may not repeat edges and vertices in xQy. Consider the subgraph formed by the edges $E(R) \triangle E(xQy)$ where $\triangle$ denotes the symmetric difference. It is a union of edge disjoint cycles and isolated vertices. Moreover, any isolated vertex must be an internal vertex of xQy. It follows that the symmetric difference must contain a cycle with at least one vertex in S. Fix C' to be such an S-cycle contained in $E(R) \triangle E(xQy)$. We claim that C' intersects $V(H)$ in at least two vertices. The cycle C' must intersect H by the maximality of H. Assume that C' intersects H in exactly one vertex. Every edge of C' which is not contained in H is an edge of C. Thus, it follows that C' is a proper subgraph of C, a contradiction. This proves that C' intersects H in at least two vertices. However, this contradicts our choice of C to minimize the edge set $E(H - (B \cup X)) \cup E(C)$, as we have avoided the edges of P.

We conclude that every $(H - (B \cup X))$-path P contained in C has endpoints in distinct components of $H - (B \cup X)$. As C is an S-cycle, there exists some component, again call it Q, of $H - (B \cup X)$ and internally disjoint subpaths P_1, P_2, P_3 of C such that

1. P_2 shares exactly one endpoint with P_1 and one endpoint with P_3,
2. P_2 is a subpath of Q and $V(P_2) \cap S \neq \emptyset$,
3. and both P_1 and P_3 are $(H - (B \cup X))$-paths.

Since Q contains a vertex of S, $V(Q)$ has at least one H-neighbor in $S \cap (X \cup B)$. It follows that for either $i = 1$ or $i = 3$, every cycle of $H \cup P_i$ must contain a vertex of S, contradicting the maximality of H. This contradiction completes the proof of the claim that there does not exist an S-cycle which intersects $H - (B \cup X)$ in at least two vertices. □

There do not exist S-cycles in $G-(B \cup X)$ which are disjoint from $V(H)$, nor do there exist such cycles intersecting $V(H)$ in two or more vertices. Consider now two S-cycles, C_1 and C_2 in $G - (B \cup X)$, which each intersect $V(H)$ in exactly one vertex. Let $V(C_i) \cap V(H) = v_i$ for $i = 1, 2$ and assume that $v_1 \neq v_2$. It follows, then that $V(C_1) \cap V(C_2) = \emptyset$. Otherwise, there exists a path P contained in $C_1 \cup C_2$ from v_1 to v_2 which intersects S and is internally disjoint from $V(H)$. However, in this case, the subgraph $H \cup P$ contradicts the maximality of H.

Let $\mathcal{C}$ be the set of all S-cycles in G that avoid $B \cup X$ and meet $V(H)$ in exactly one vertex (necessarily of degree 2 in H). Let $Y \subseteq V(H) \setminus (B \cup X)$ be the set of those vertices. For each $y \in Y$ pick an S-cycle $C_y \in \mathcal{C}$ that meets H in the vertex y. Consider the set $\mathcal{C}' := \{C_y \,|\, y \in Y\}$. As we observed above, the cycles $\{C_y : y \in Y\}$ are pairwise disjoint. Thus, if $|Y| \geq k$, we see that there exist k pairwise disjoint S-cycles, and otherwise the set $Z := B \cup X \cup Y$ is the desired S-cycle hitting set.

The only remaining claim is to show that there exists an algorithm to find either the disjoint S-cycles or the hitting set Z. We first greedily select a minimal S-segment hitting set X. Set $\mathcal{C}$ and Y to be the empty set. As long as the $|Y| \leq k$, we check whether there exists an S-cycle in $G - (B \cup X \cup Y)$. If such a cycle C exists, it must intersect $V(H) \setminus (B \cup X)$ in exactly one vertex y. We let $\mathcal{C} = \mathcal{C} \cup C$ and $Y = Y \cup y$ and repeat. We return $\mathcal{C}$ if $|Y| \geq k$, and otherwise return $X \cup B \cup Y$. This completes the proof of the lemma. □

The proof of Simonovits proceeds by defining the following constants:

Definition For $k \in \mathbb{N}$, let

$$r_k := \log k + \log\log k + 4 \tag{1}$$

$$s_k := \begin{cases} 4kr_k & \text{if } k \geq 2 \\ 2 & \text{if } k \leq 1 \end{cases} \tag{2}$$

and proving the following lemma.

Lemma 7 ([15]) *Let $k \in \mathbb{N}$, and let H be a 3-regular multigraph. If $|V(H)| \geq s_k$, then H contains at least k disjoint cycles.*

An immediate consequence of the proof of the lemma is that the cycles can be greedily selected, i.e. in time $O(n)$ we can find a cycle of length $O(\log k)$. Consequently, to find the k cycles, repeatedly find such a minimal cycle, delete it, and recurse on the smaller graph. Thus the k disjoint cycles can be found in time $O(nm)$.

The proof of Theorem 2 now follows easily from the lemmas.

Proof. [Theorem 2] Assume we are given a graph G, $S \subseteq V(G)$, and $k \geq 1$ a positive integer. Let $n = |V(G)|$ and $m = |E(G)|$. Let s_k be the value given in Lemma 7. We let H be a maximal $(2,3)$-subgraph of G such that every cycle in H is an S-cycle. By Lemma 4, we can find such an H in time $O(nm)$. Let B be the set of branch vertices of H. If $|B| \geq s_k$, then H contains k disjoint cycles, all of which are necessarily S-cycles. Moreover, we can find them in $O(nm)$ time. Thus, we may assume that $|B| < s_k$. By Lemma 5, either there exist k disjoint S-cycles or there exists an S-cycle hitting set of size at most $\frac{5}{2}s_k + 2k \leq c_v k \log k$ for an appropriate choice of c_v. Moreover, Lemma 5 gives an algorithm to find either the cycles or the hitting set in time $O(nm)$, as desired. □

Acknowledgements

The authors are indebted to the anonymous referees who greatly improved the readability of the article.

References

[1] V. Bafna, P. Berman, and T. Fujito, Constant ratio approximation of the weighted feedback vertex set problem for undirected graphs, *Proceedings of the 6th International Symposium on Algorithms and Computation (ISAAC)* Lecture Notes in Computer Science, Springer-Verlag, (1995) pp. 142 - 151.

[2] A. Becker and D. Geiger, Approximation algorithms for the loop cutset problem *Proceedings of the Tenth Annual Conference on Uncertainty in Artificial Intelligence, July 29-31, 1994* Seattle, Washington, pp. 60 - 68.

[3] A. Caprara, A. Panconesi, and R. Rizzi, Packing cycles in undirected graphs, *J. Algorithms* **48** (2003), pp. 239 - 256.

[4] F. A. Chudak, M. X. Goemans, D. S. Hochbaum, and D. P. Williamson, A primal-dual interpretation of the two 2-approximation algorithms for the feedback vertex set problem in undirected graphs, *Oper. Res. Lett.* **22** 4-5, pp. 111 - 118.

[5] M. Cygan, M. Pilipczuk, M. Pilipczuk, J. Wojtaszczyk, Subset feedback vertex set is fixed parameter tractable, *Proceedings of the 38th International Colloquium on Automata, Languages, and Programming (ICALP)* Lecture Notes in Computer Science, Springer-Verlag, (2011), pp. 449 - 461.

[6] R. Diestel, Graph Theory, 3rd edition Springer Verlag (2005).

[7] D. Dor and M. Tarsi, Graph decomposition is NPC - a complete proof of Holyer's conjecture, *Proceedings of the twenty-fourth annual ACM symposium on Theory of Computing (STOC)*, (1992), pp. 252 - 263.

[8] P. Erdős and L. Pósa, On the independent circuits contained in a graph, *Canadian Journal of Mathematics* **17** (1965), pp. 347 - 352.

[9] G. Even, J. Naor, and L. Zosin. An 8-approximation algorithm for the subset feedback vertex set problem. *SIAM J. Comput.* **30(4)**, (2000), pp. 1231 - 1252.

[10] Z. Friggstad and M. Salavatipour, Approximability of packing disjoint cycles, *Proceedings of the 18th international conference on Algorithms and computation* Lecture Notes in Computer Science, Springer-Verlag, (2007), pp. 304 - 315.

[11] N. Kakimura and K. Kawarabayashi, Packing directed circuits through prescribed vertices bounded-fractionally, *manuscript*.

[12] N. Kakimura, K. Kawarabayashi, and D. Marx, Packing cycles through prescribed vertices, *J. Combin. Theory, Ser. B* **101** (2011), pp. 378 - 381.

[13] K. Kawarabayashi and Y. Kobayashi, Fixed-parameter tractability for the subset feedback vertex set problem and the S-cycle packing problem, *manuscript*, (2010).

[14] M. Krivelevich, Z. Nutov, M. Salavatipour, J. Verstraete, and R. Yuster, Approximation algorithms and hardness results for cycle packing problems, *ACM Transactions on Algorithms*, **3** (2007), Article 48.

[15] M. Simonovits, A new proof and generalizations of a theorem of Erdős and Pósa on graphs without k+1 independent circuits, *Acta Mathematica Academiae Scientiarum Hungaricae*, **18** (1967), pp. 191 - 206.

[16] R. Tarjan, Depth - first search and linear graph algorithms. *SIAM J. Comput.* **1** (1972), pp. 146 - 160.