KAIST, School of Computing, Spring 2024
Algorithms for NP-hard problems (CS492)
Lecture: Eunjung KIM

Scribed By: Eunjung KIM
Lecture #5
11 March 2024

Contents

1 Quadratic kernel for FEEDBACK VERTEX SET

Basic Preprocessing. The input graph G for FEEDBACK VERTEX SET may have a loop (i.e. an edge whose both endpoints are identical) or parallel edges (i.e. multiple edges between a pair of vertices) as we may create loops and parallel edges while applying reduction rules. Graphs without loops and parallel edges are called *simple graphs*, and graphs with possible loops and parallel edges are called *multigraphs*. For formal definition of multigraphs, see Chapter 1.10 of Diestel's book. Very often, we assume that the graphs under consideration are simple graphs, but there are situations where multigraphs are more natural objects to consider. The kernelization of FEEDBACK VERTEX SET is such a situation.

In a multigraph, the degree of a vertex v counts the number of edges incident with v, where a loop at v contributes 2 to the degree of v. One can have a cycle of length 1 (a loop) or of length 2 (two parallel edges).

In each of the following reduction rules, the obtained graph is referred to as G'.

- Reduction Rule 1: If a vertex v has a loop, then delete v and set $k' := k - 1$.
- Reduction Rule 2: If a vertex v has degree at most 1 in G, then delete v and $k' := k$.
- Reduction Rule 3: If there are more than two parallel edges between u and v, delete all edges but two of them and set $k' := k$.
- Reduction Rule 4: If a vertex v is incident with exactly two edges vx and vy, delete v and add an edge xy. Let $k' := k$. Note that this operation can create parallel edges between x and y (if they are already adjacent) or a loop (if $x = y$).

The next two lemmas are straightforward and their proof is left to the readers.

Lemma 1. *Each of Reduction Rule 1-4 is safe and can be applied in polynomial time.*

Lemma 2. *Let (G, k) be an instance to* FEEDBACK VERTEX SET *which is irreducible with respect to Reduction Rules 1-4. Then we have $deg(v) \geq 3$.*

What next? An overview. Suppose that (G, k) is irreducible with respect to the basic Reduction Rules 1-4. If the maximum degree of G is bounded by d, then one can easily bound the size of G.

Lemma 3. *Let (G, k) be an instance to* FEEDBACK VERTEX SET *which is irreducible with respect to Reduction Rules 1-3. If $deg(v) < d$, then $|V| < (d + 1)k$ and $|E| < 2dk$ unless (G, k) is a* NO*-instance.*

Proof: If (G, k)-is a NO-instance, we have nothing to prove. Suppose that X is a feedback vertex set of G of size at most k. Let $E(V - X, X)$ be the set of edges with one endpoint in $V - X$ and another endpoint in X, and $E(V - X)$ is the set of edges with both endpoints in $V - X$. The former is upper bounded by $d|X|$ due to $deg(v) \leq d$ and the latter is upper bounded by $|V - X| - 1$ because $G[V - X]$ is acyclic. Now

$$3 \cdot |V - X| \leq \sum_{v \in V - X} deg(v) = |E(V - X, X)| + 2 \cdot E(V - X) < d|X| + 2(|V - X| - 1),$$

from which we deduce $|V| < (d + 1)|X| \leq (d + 1)k$ and

$$\begin{aligned} |E| &= |\text{edges incident with } X| + |\text{edges with both endpoints in } V - X| \\ &< d|X| + |V - X| - 1 \\ &\leq |V| + (d - 1)|X| < 2dk. \end{aligned}$$

□

Based on Lemma 3, our strategy for getting an $O(k^2)$-vertex kernel is to upper bound the maximum degree of G by $O(k)$. A priori, there is no reason why the input graph G has degree $O(k)$ - a vertex may have an arbitrarily large degree. In what follows, if there is a vertex x of degree large enough ($\geq 8k$ suffices), either there is a certificate forcing x in any feedback vertex set of size at most k or there is a vertex set $S'_x \subseteq V - x$ such that one may assume any optimal feedback vertex set contains either x or S'_x. Each of these two cases can be handled with suitable reduction rules.

An *x-cycle* is a cycle traversing x, which can be one, two or at least three in general. But as we can assume that the instance at hand is irreducible with respect to Reduction Rule 1, the length of an x-cycle will be at least two whenever a relevant reduction rule or lemmas are applied, unless stated otherwise.

An *x-flower* is a collection of x-cycles such that any pair of cycles intersect exactly at $\{x\}$. We say that x-flower is of order k if there are k cycles in the collection. When there is an x-flower of order at least $k+1$, then any feedback vertex set of size at most k must contain x and the safeness of the next reduction rule is immediate.

- Reduction Rule 5: If there is an x-flower of order $k+1$, then delete x and set $k' := k - 1$.

What if there is no x-flower of order $k+1$? We shall see that one of the following happens; recall that x does not have a loop.

(i) Either there an x-flower of order $k+1$,

(ii) or, no x-flower of order $k+1$ exists. In this case, there is a vertex set $S_x \subseteq V - x$ of size at most $2k$ which hits all x-cycles.

That one of (i) and (ii) must happen is Corollary 1. The eligible one can be decided in polynomial time by Theorem 1 and Corollary 1. When (i) is the case, one can apply Reduction Rule 5. What if (ii) is the case? Lemma 9 says that if (ii) is the case **AND** $deg(x) \geq 8k$, one can identify a set $\emptyset \neq S'_x \subseteq S_x$ such that there is an optimal feedback vertex set (which must hit all x-cycles) containing x or there is one containing S'_x. This feature can be implemented as a modified instance with x having double edges with all vertices of S'_x. Reduction Rule 6 describes the detail of how to implement this feature so as to strictly decrease some measure on the input graph.

Below we see more details, and why things work as they do. We maintain the assumption that the input instance (G, k) is irreducible with respect to Reduction Rules 1-4.

What next? Distinguishing between (i) and (ii). For a vertex subset A of G, we say that a path P is an *A-path* if P has length at least one and the vertex set of P intersects with A precisely at the two endpoints of P; i.e. both endpoints of P are in A and no internal vertex of P is in A.

For $i = 1, 2$, let $A_i \subseteq N(x)$ be the set of neighbors of x connected with x by i edges. How does a hitting set S_x of all x-cycles avoiding x look like? First of all, it must contain A_2 entirely and this hits all x-cycles of length 2. Consider an x-cycle C of length at least three. For sure, the path $P := C - x$ must start from a vertex of $N(x)$ and end in one of $N(x)$. As S_x already contains A_2, we may consider only the case when the endpoints of P are in A_1. The path P may traverse A_1 many times, but one can choose a subpath P' of P whose internal vertices are outside of A_1, that is, P always contains a subpath P' which is an A_1-path. Note that P' together with x forms an x-cycle whose vertex set is contained in $V(C)$.

This summarizes to the next observation.

Lemma 4. *Let x be a vertex of $G = (V, E)$.*

- *The maximum order of x-flower equals $|A_2|$ plus the maximum number of vertex-disjoint A_1-paths in $G - x - A_2$.*
- *$S_x \subseteq V - x$ is a hitting set of all x-cycles if and only if S_x contains A_2 and $S_x - A_2$ hits all A_1-paths of $G - x - A_2$.*

Proof: To see the $\leq$-inequality in the first statement, let $\mathcal{F}$ be an x-flower which minimizes $|\bigcup_{C\subseteq\mathcal{F}} C|$. it is easy to see that if $C \in \mathcal{F}$ intersects A_2 then C is a 2-cycle consisting of x and a vertex $u \in A_2$; otherwise contradicting the minimum cardinality assumption on $\mathcal{F}$. In particular, this means that all the double edges incident with x (and with A_2) are members of $\mathcal{F}$. Therefore, for any x-cycle $C \in \mathcal{F}$ of length at least three, we have $C \cap N(x) \subseteq A_1$, $C \cap A_2 = \emptyset$ and the path $C - x$ connects two distinct vertices of A_1. Again by the minimum cardinality assumption, we conclude that $C - x$ is an A_2-path. The $\geq$-inequality is immediate from the fact that A_2 and a collection of vertex disjoint A_1-paths can be extended to an x-flower.

The second statement is easy to verify and left to the readers. □

Let $\ell := k - |A_2|$. Thanks to Lemma 4, the case of (i) and (ii) above translates to the following two cases. Now let $A := A_1$ and $H := G - A_2 - x$.

(i') Either there are $\ell + 1$ vertex-disjoint A-paths in H,

(ii") or, the maximum number of vertex-disjoint A-paths in H is at most ℓ. In this case, there is a vertex set $S_x \subseteq V(H)$ of size at most 2ℓ which hits all A-paths of H.

We use the next theorem by Gallai to show that one of the case (i') and (ii') happens, and to decide which case applies can be done in polynomial time. The proof of Gallai's Theorem can be found in Chapter 9.1.1 of the textbook by Cygan et. al.

Theorem 1 (Gallai 1961, Schrijver 2001)**.** *Let $G = (V, E)$ and $A \subseteq V$. Then the maximum number of pairwise vertex-disjoint A-paths equals the minimum of*

$$|X| + \sum_{C\in\mathcal{C}(G-X)} \lfloor \frac{|A \cap C|}{2} \rfloor,$$

where the minimum is taken over all vertex subsets $X \subseteq V$, and $\mathcal{C}(G - X)$ is the collection of all connected components of $G - X$. Moreover, both a maximum collection of pairwise vertex-disjoint A-paths and a vertex set X minimizing the right-hand side of the equality can be computed in polynomial-time.

Lemma 5. *Let $G = (V, E)$ be a graph. If the maximum number of pairwise vertex-disjoint A-paths is at most k, then is a hitting set Z of all A-paths with $|Z| \leq 2k$, i.e. a vertex set $Z \subseteq V$ such that there is no A-path in $G - Z$.*

Proof: By Theorem 1, there exists $X \subseteq V$ such that $|X| + \sum_{C\in\mathcal{C}(G-X)} \lfloor \frac{|A\cap C|}{2} \rfloor \leq k$. We add X to Z, and additionally $2 \cdot \frac{|A\cap C|}{2}$ vertices to Z from $A \cap C$ for each connected component of C of $G - X$. Note that

$$|Z| = |X| + \sum_{C\in\mathcal{C}(G-X)} 2 \cdot \lfloor \frac{|A \cap C|}{2} \rfloor \leq 2\big(|X| + \sum_{C\in\mathcal{C}(G-X)} \lfloor \frac{|A \cap C|}{2} \rfloor\big) \leq 2k.$$

It remains to see that $G - Z$ contains no A-path. Suppose that P is an A-path in $G - Z$. Then there exists a connected component C of $G - Z$ entirely containing P. As the two (distinct) endpoints of P must be in A, the path P witnesses that $|(C \cap A) - Z| \geq 2$. Recall that C is in turn contained in some connected component C^* of $G - X$ as $X \subseteq Z$. This means that $|(C^* \cap A) - Z| \geq 2$, which contradicts the fact that Z takes all but at most one vertex from $K \cap A$ for each connected component K of $G - X$. $\square$

Corollary 1. *There is a polynomial-time algorithm which, given a graph $G = (V, E)$, a vertex x without loop, and a non-negative integer k, either detects an x-flower of order $k + 1$ or returns a vertex set $S_x \subseteq V - x$ of size at most $2k$ hitting all x-cycles.*

Proof: Let $A_2 \subseteq N(x)$ be the set of neighbors of x connected with x by at least two edges. Let $\ell := k - |A_2|$ and let $H := G - A_2 - x$. By Lemma 5, there a collection of $\ell + 1$ vertex-disjoint A-paths in H or there is a vertex set $S_x \subseteq V - A_2 - x$ of size at most 2ℓ hitting all A-paths of H. Moreover, one can detect such a collection of $\ell + 1$ or return a vertex set S_x in polynomial time by Theorem 1. In the former case, there is an x-flower of order $k + 1$ in G by Lemma 4. In the latter case, $S_x \cup A_2$ is a hitting set of all x-cycles of G by the same lemma. It remains to note that $|S_x \cup A_2| \leq 2\ell + |A_2| \leq 2k - |A_2| \leq 2k$. $\square$

What next? Further into the case (ii). Let $S_x \subseteq V - x$ be a vertex set of size at most $2k$ hitting all x-cycles of G. Our goal is to show that the next, last Reduction Rule is applicable in this case *provided that* $deg(x) \geq 8k$. We first present the reduction rule.

- Reduction Rule 6: Suppose there exist a vertex x, a vertex subset $S \subseteq V - x$ and a set $\mathcal{T}$ (not necessarily all the connected components of $G - S - x$) of connected components of $G - S - x$ such that

  1. $G[C]$ is a tree for each $C \in \mathcal{T}$,
  2. there is exactly one edge between x and C for each $C \in \mathcal{T}$,
  3. for every $Z \subseteq S$, there are at least $2|Z|$ connected components of $\mathcal{T}$ neighboring Z.

  Then the new graph G' is obtained by

  - deleting the edges connecting x and the connected components of $\mathcal{T}$, and
  - adding double edges between x and S (and deleting some edges so that there are exactly two edges between x and each vertex of S).

  We set $k' := k$.

We need to prove two things; (a) Reduction Rule is safe, and (b) if $deg(x) \geq 8x$ and $S_x \subseteq V - x$ is a hitting set of all x-cycles of size at most $2k$, there exists $S'_x \subseteq S_x$ which satisfies the condition of Reduction Rule 5 by setting $S := S'_x$ unless (G, k) is a trivial NO-instance.

Lemma 6. *Reduction Rule 6 is safe.*

Proof: Let X' be an optimal feedback vertex set of G'. If $x \in X'$, then X' is a feedback vertex set of G as well because G and G' differ only at edges incident with x, in particular any cycle not traversing x is present in G if and only if it is in G'.

Let X be an optimal feedback vertex set of G. If $x \in X$, the same argument in the previous paragraph applies, and X is a feedback vertex set of G'. So, assume that $x \notin X$. We show that there is an optimal feedback vertex set X^* which entirely contain S. If $S \subseteq X$, the claim trivially holds. If $Z := S \setminus X \neq \emptyset$, consider a vertex $z \in Z$ and the tree components of $\mathcal{T}$ neighboring z. There are at least two such tree components (Condition 3), and X must intersect all those tree components save at most one; otherwise, the tree components free from X together with z and x contain a cycle, contradiction that X is a feedback vertex set. Let us count the number of tree components of $\mathcal{T}$ neighboring Z, which is

$$\leq \text{\# tree components intersecting } X + \text{\# tree components free from } X \leq |X \cap \bigcup_{T \in \mathcal{T}} V(T)| + |Z|.$$

On the other hand, Condition 3 says that the number of tree components of $\mathcal{T}$ neighboring Z is at least $2|Z|$. Therefore, we have $|Z| \leq |X \cap \bigcup_{T \in \mathcal{T}} V(T)|$. Take a set

$$X^* := \big(X \setminus \bigcup_{T \in \mathcal{T}} V(T)\big) \cup Z$$

and observe that $|X^*| \leq |X|$. It remains to verify that X^* is indeed a feedback vertex set of G. Note that any cycle of $G - X^*$ must traverse some tree component T of $\mathcal{T}$. As the neighborhood of T are in $\{x\} \cup S$ and $S \cup X^*$, this means that C is a cycle fully contained in $\{x\} \cup V(T)$. This is impossible

Now we argue that X^* is a feedback vertex set of G'. If there is a cycle C in $G' - X^*$, then it must be an x-cycle. Moreover, C must contain an edge G' which was not present in G. Such edges are between x and S, and thus C must traverse a vertex of S, contradicting that C is a cycle in $G' - X^*$ and X^* contain S. This is impossible due to Condition 2, which subsumes that $G[\{x\} \cup V(T)]$ is a tree. □

The last piece is Lemma 9 which states that if $deg(x) \geq 8k$ and the latter case of Corollary 1 holds, then Reduction Rule 6 is applicable. For this, we need two technical lemmata.

Lemma 7. *Let H be a bipartite graph with a bipartition $S \uplus T$. In polynomial time, one can find a set $Z \subseteq S$ with $|N(Z)| < 2|Z|$ if such Z exists.*

Proof: Copy each vertex v of S, say v' and make it adjacent with $N(v)$. The new graph H' is a bipartite graph as well. Use König Theorem (Lecture Note #3 on Kernelization Part I) to compute a maximum matching M of H'. If M saturates $S \cup S'$, we know that $|N(Z)| \geq |Z|$ for every $Z \subseteq S \cup S'$. In particular, this holds when Z takes both vertices of a pair $\{v, v'\}$. This means that $|N(Z)| \geq 2|Z|$ for every $Z \subseteq S$ in H. □

Lemma 8. *Let H be a bipartite graph with a bipartition $S \uplus T$ satisfying*

- *every vertex of T is adjacent with a vertex of S,*
- $|T| \geq 2|S|$.

Then in polynomial time one can find vertex sets $S' \subseteq S$ and $T' \subseteq T$ such that

- *$N(T') = S'$, and*
- *$|N(Z)| \geq 2|Z|$ for every $Z \subseteq S'$.*

Proof: The proof uses the same argument as in the proof of Theorem 4 in Lecture Note #3 on Kernelization Part I. □

Lemma 9. *Let (G, k) be an instance irreducible with respect to Reduction 1-4. Let x be a vertex of degree at least $8x$ and assume that there is no x-flower of order at least $k+1$. Then in polynomial time, one can do one of the following:*

- *correctly decide that* (G, k) *is a* NO-*instance, or*
- *return a vertex set* S *meeting the condition of Reduction Rule 6.*

Proof: By Corollary 1, one can find a vertex set $S_x \subseteq V - x$ of size at most $2k$ hitting all x-cycles in polynomial time.

If x have more than $3k$ incident edges whose other endpoints are in S_x, then there are double edges between x and at least $k+1$ vertices of S_x. They form an x-flower of order at least $k+1$, a contradiction.

Therefore, x and S_x are connected by at most $3k$ edges. Note that x is connected with any connected component C of $G - S_x - x$ by at most one edge; otherwise, x and C with two edges between them create a cycle avoiding S_x. So any $C \in \mathcal{C}$ and x is connected by at most one edge. Let $\mathcal{C}$ be the set of connected components of $G - S_x - x$ joined by a single edge with x. There are at least $5k$ members in $\mathcal{C}$ because any edge incident with x is either incident with S_x (and there are at most $3k$ such edges) or incident with some member of $\mathcal{C}$, and due to the assumption $deg(x) \geq 8k$.

There are two cases.

1. If there are at least $k+1$ connected components in $\mathcal{C}$ containing a cycle, then (G, k) is a NO-instance.
2. Otherwise, the set $\mathcal{T} \subseteq \mathcal{C}$ of connected components of $G - S_x - x$ each of which forming a tree has at least $4k$ members.

In the first case, we're done. In the second case, note that $|\mathcal{T}| \geq 2|S|$. Moreover, every connected component C of $\mathcal{T}$ is adjacent with at least one vertex of S; if not, C is a tree and a leaf of the tree C has degree at most two in G, thus Reduction Rule 2 or 4 is applicable. Now one can apply the algorithm of Lemma 8 to the auxiliary graph $H = (S, \mathcal{T})$ and obtain sets $S' \subseteq S$ and $\mathcal{T}' \subseteq \mathcal{T}$ such that

- S' is exactly the set of vertices of S which are adjacent with tree components of $\mathcal{T}'_0$, and
- every $Z \subseteq S'$ is adjacent with at least $2|Z|$ tree components of $\mathcal{T}'_0$.

It remains to observe that S' and $\mathcal{T}'$ satisfies the condition of Reduction Rule 6. This completes the proof. □

The kernelization algorithm is summarized as Algorithm **FVS-kernel**(G, k).

Correctness of FVS-kernel(G, k).The equivalence of (G, k) and the output instance at lines 2, 3 is trivial. Line 4 is correct due to the correctness of Reduction Rules 1-4. The output at line 6 is correct due to Lemma 3. The correctness of Reduction Rule 5 ensures that line 9 is correct. At line 11, the existence of such S_x is given by Lemma 1.

At line 16, the number of components in $\mathcal{T}$ is at least $4k$. Indeed x can have at most $3k$ edges incident with S_x since otherwise, there is an x-flower (consisting of 2-cycles) of order $k+1$. All other edges incident with x must have the other endpoint in one of the members of $\mathcal{C}$, and thus we have $|\mathcal{C}| \geq 5k$. As there are at most k connected components of $\mathcal{C}$ containing a cycle, it follows that $|\mathcal{T}| \geq 4k$.

Therefore, the second condition of Lemma 8 are satisfied by S_x and $\mathcal{T}$ in the bipartite graph obtained by contracting each connected component of $\mathcal{T}$ to a single vertex (and ignoring all edges between vertices of S_x). The first condition of Lemma 8 is satisfied, note that each member $T \in \mathcal{T}$ has exactly one edge connecting T and x because S_x is a hitting set of all x-cycles. If T is not adjacent with S_x, then $x + T$ is a tree and one can find a leaf and Reduction Rule 2 is applicable. Hence, both conditions of Lemma 8 are satisfied, and one can obtain in polynomial time $S'_x \subseteq S_x$ and $\mathcal{T}' \subseteq \mathcal{T}$ which meet Condition 3 of

Algorithm 1 Kernelization for FEEDBACK VERTEX SET

1: **procedure FVS-kernel**(G, k)
2:     **if** $k < 0$ or ($k = 0$ and G contains a cycle) **then return** NO
3:     **if** G is a forest **then return** YES
4:     **if** One of Reduction Rules 1-4 is applicable with output (G', k') **then return FVS-kernel**(G', k')
        ▷ Now, G is loopless and every vertex has degree at least 3
5:     **if** $|V| < 8k^2 + k$ **then return** (G, k).
6:     **else if** $deg(v) < 8k$ for every $v \in V$ **then return** NO.
7:     **else**
8:         Pick a vertex x of degree at least $8k$.
9:         **if** there is an x-flower of order $k + 1$ **then** return **FVS-kernel**$(G - x, k - 1)$
10:         **else**
11:             Let $S_x \subseteq V - x$ be a hitting set of all x-flowers of size at most $2k$.
12:             ▷ x is joined with S_x with more than $3k$ edges.
13:             Let $\mathcal{C}$ be the connected components of $G - S_x - x$ joined with x.
14:             ▷ $|\mathcal{C}| \geq 5k$.
15:             **if** there are more than k components of $\mathcal{C}$ containing a cycle **then** return NO
16:             Let $\mathcal{T} \subseteq \mathcal{C}$ be the tree components of $G - S_x - x$ joined with x by a single edge.
17:             ▷ $|\mathcal{T}| \geq 4k$.
18:             Let $\mathcal{T}' \subseteq \mathcal{T}$ and $S' \subseteq S$ as described in Lemma 8.
19:             Let G' be the output of Reduction Rule 6. **return FVS-kernel**(G', k)

Reduction Rule 6 (line 18). Condition 1 of Reduction Rule 6 is met by x, S'_x and $\mathcal{T}'$ clearly. To verify that $\mathcal{T}'$ is a set of connected components of $G - S'_x - x$, it suffices to recall that the neighboring vertices of $\mathcal{T}'$ in S_x are precisely S'_x by Lemma 8. This, together with the fact that each $T\mathcal{T}$ is adjacent with at least one vertex of S_x, also means that each connected component of $\mathcal{T}'$ is adjacent with at least one vertex of S'_x. The correctness of line 19 follows from Lemma 6.

That the algorithm outputs a kernel on at most $8k^2 + k$ vertices on polynomial time is straightforward, left to the readers.

2 Remarks

In Dell and van Melkebeek (2010), it was proved that for both VERTEX COVER and FEEDBACK VERTEX SET, the quadratic-size kernels are essentially the best possible. For FEEDBACK VERTEX SET, the best known kernelization ends up with $2k^2 + k$ vertices and $4k^2$ edges. Whether one can obtain a kernel with $o(k^2)$ vertices for FEEDBACK VERTEX SET remains open. This result on quadratic-size lower bound for both VERTEX COVER and FEEDBACK VERTEX SET builds on the machinery for proving a lower bound of a kernel by Dell and van Melkebeek (2010).

- Holger Dell, Dieter van Melkebeek, Satisfiability allows no nontrivial sparsification unless the polynomial-time hierarchy collapses, STOC 2010, J. ACM 61(4): 23:1-23:27 (2014). Link

The kernelization for FEEDBACK VERTEX SET can be obtained in linear time.

- Yoichi Iwata, Linear-time Kernelization for Feedback Vertex Set, ICALP 2017: 68:1-68:14. Link