Feedback Vertex Set with Bounded Cycle Length: Approximation, Tractability and Beyond the Worst-Case Analysis

Martin Sokolov

Utrecht University

June 24, 2025

Outline

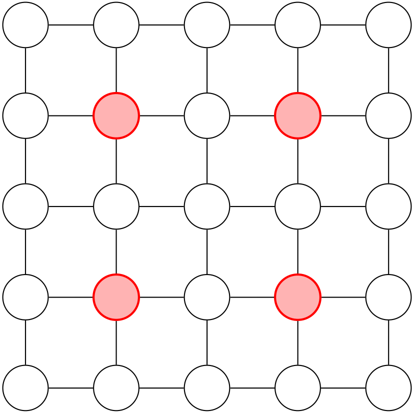
1. Problem Overview
2. Approximation Schemes
3. Fixed-Parameter Tractability of FVS-BCL
4. Certified Algorithms
5. Conclusion

Problems Studied (Emphasis on Focus Areas)

- Dominating Set - BWCA is studied
- Feedback Vertex Set (FVS) - Approximation is studied
- Feedback Vertex Set with Four Cycle Length (FVS-4CL)
- Feedback Vertex Set with Bounded Cycle Length (FVS-BCL)
- r-Dominating Set (r-DS)

FVS-4CL is the primary focus of this work.
We aim to generalize results to **FVS-BCL**.
(Our hope in the beginning was generalizing our results to **FVS**).

FVS-4CL on a 5x5 grid



Approximation Algorithms for Feedback Vertex Set (FVS)

- $\min\{2\Delta^2,\ 4\log n\}$ **where Δ is the max. degree in G. (Bar-Yehuda et al., 1994)**
  Primal-dual algorithm on undirected graphs with general vertex weights.
- **2-Approximation (Bafna et al., 1995)**
  Local ratio technique with improved efficiency.
- **2-Approximation (Becker and Geiger, 1996)**
  Greedy-like approximation algorithm.
- **2-Approximation (Chudak et al., 1998)**
  A primal-dual algorithm.
- **Hardness: APX-Complete (Dinur and Safra, 2005)**
  NP-hard to approximate within a factor better than 1.36 via reduction from Vertex Cover.

Approximation Algorithms for Feedback Vertex Set (FVS) in planar graphs

- **PTAS for FVS (Kleinberg and Kumar, 2001)**
- **PTAS for FVS (Le and Zheng, 2020)**
  Using a local search heuristic
- **EPTAS for unweighted FVS (Demaine and Hajiaghayi)**
  Using bidimensionality
- **PTAS for weighted FVS (Cohen-Addad et al., 2016)**
  Reduction from weighted feedback vertex set to vertex-weighted connected dominating set
- **EPTAS for weighted FVS (Open question.)**

FVS with Bounded Cycle Length (FVS-BCL)

- **Inapproximability of Feedback Vertex Set for Bounded Length Cycles (Guruswami and Lee, 2014)**
  For any integer constant $\rho \geq 3$ and $\epsilon > 0$, it is hard to find a $(\rho - 1 - \epsilon)$-approximate solution to the problem of intersecting every cycle of length at most ρ.

EPTAS via Baker's Technique

We obtain:

- $(1+2\epsilon)$-approximation algorithm for the FVS-4CL problem with a running time of $2^{\mathcal{O}(\mathrm{tw}^2)} \cdot n^{\mathcal{O}(1)}$.
- $\left(1+\frac{\lfloor \rho/2 \rfloor}{\epsilon}\right)$-approximation for the FVS-BCL problem with a running time of $f(\mathrm{tw}, \rho) \cdot n^{\mathcal{O}(1)}$ for some computable function f.

Algorithm Baker's technique for the unweighted FVS-4CL

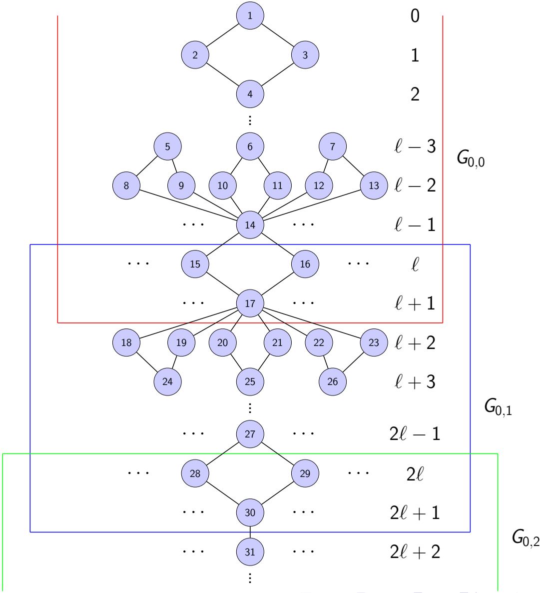
Require: Planar graph $G = (V, E)$, parameter $\ell \leftarrow \frac{1}{\epsilon}$
Ensure: $(1+\epsilon)$-approximation for $\mathrm{FVS\text{-}4CL}$

```
Perform BFS from some arbitrary vertex r
S ← ∅
i: shift; j: slice
for each i = 0 to ℓ − 1 do
    Let G_{i,j} be the subgraph induced on vertices at levels j·ℓ + i through
    (j + 1)·ℓ + i + 1 for all j ≥ 0.
    Let S_{i,j} be the minimum unweighted FVS-4CL of G_{i,j} using Algo-
    rithm 21 as a subroutine (weights are all ones).
    Let S_i = ⋃_j S_{i,j}
    S ← S ∪ {S_i}
end for
return S_{i*} from S with minimum cardinality
```

BFS Layered Tree Structure

Idea: Break graph into layers via BFS.

- Nodes grouped by distance from root.
- Overlap grouped by distance from root mod ℓ
- Subgraphs $G_{0,0}, G_{0,1}, G_{0,2}, \ldots$



Approximation Bound via Subgraph Solutions

- Let $S_{i,j}$ be the minimum unweighted feedback vertex set (FVS-4CL) for subgraph $G_{i,j}$, computed using Algorithm 21.
- Let $F \equiv \text{OPT}$ be the optimum solution for the full graph G, and define $F_{i,j} := F \cap V(G_{i,j})$, the restriction of F to subgraph $G_{i,j}$.
- We obtain the following bound on the total solution size:

$$\underbrace{|S_{i^*}| \leq \sum_j |S_{i^*,j}|}_{\text{by Lemma 7 and Result 1}} \underbrace{\leq \sum_j |F_{i^*,j}|}_{\text{by Lemma 8}} \underbrace{\leq \left(1 + \frac{2}{\ell}\right) \cdot |F|}_{\text{by Lemma 9}} = (1 + 2\epsilon) \cdot |F|$$

Weighted Bounds for FVS-BCL Problems

Weighted FVS-4CL:

$$w(S_{i^*}) \leq \sum_j w(S_{i^*,j}) \leq \sum_j w(F_{i^*,j}) \leq \left(1 + \frac{2}{\ell}\right) \cdot w(F) = (1 + 2\epsilon) \cdot w(F)$$

Weighted FVS-BCL (break cycles of length ρ):

$$w(S_{i^*}) \leq \sum_j w(S_{i^*,j}) \leq \sum_j w(F_{i^*,j}) \leq \left(1 + \frac{\lfloor \rho/2 \rfloor}{\ell}\right) \cdot w(F)$$

Fixed-Parameter Tractability of FVS-BCL

Fixed-Parameter Tractability of FVS-BCL

- Monadic Second Order Logic for FVS-BCL
- Dynamic Programming Algorithm for FVS-4CL using Nice Tree Decompositions

MSOL Formulation

- An extension of First-Order Logic
- Object variables: *vertices*: $v_1, v_2, \ldots$ and *edges*: $e_1, e_2, \ldots$.
- Set variables: sets of vertices $V_1, V_2, \ldots$ and sets of edges $E_1, E_2, \ldots$.
- Binary relation $\in$: $\{\text{object variable}\} \times \{\text{set variable}\} \to \{0, 1\}$.
  Therefore $v \in V$ *iff* v is an element of the corresponding set V.
- The $Adj(e, v_i, v_j)$ relation. It detects whether edge e is an edge from vertex v_i to vertex v_j where $v_i \neq v_j$.
- Quantification over set variables: $\forall V_i, \forall E_i$ and $\exists V_i, \exists E_i$.

MSOL Formulation: Unweighted FVS-4CL

$$
\begin{aligned}
\min_{F \subseteq V} \; & |F| : \\
& \forall v : \forall u : \forall w : \forall z : \\
& v \in (V \backslash F) \land u \in (V \backslash F) \land w \in (V \backslash F) \land z \in (V \backslash F) \\
& \land\, v \neq w \land u \neq z \\
& \land\, \neg((v, u) \in E \land (u, w) \in E \land (w, z) \in E \land (v, z) \in E)
\end{aligned} \tag{1}
$$

MSOL Formulation: Weighted FVS-4CL

$$\begin{aligned}
&\min_{F \subseteq V} w(F) : \\
&\quad \forall v : \forall u : \forall w : \forall z : \\
&\quad v \in (V \setminus F) \wedge u \in (V \setminus F) \wedge w \in (V \setminus F) \wedge z \in (V \setminus F) \\
&\quad \wedge\, v \neq w \wedge u \neq z \\
&\quad \wedge\, \neg((v,u) \in E \wedge (u,w) \in E \wedge (w,z) \in E \wedge (v,z) \in E)
\end{aligned} \tag{2}$$

MSOL Formulation: Weighted FVS-ρCL

$$
\begin{aligned}
\min_{F \subseteq V} \; & w(F) : \\
& \forall v_1 : \forall v_2 : \ldots \forall v_\rho : \\
& v_1 \in (V \setminus F) \land v_2 \in (V \setminus F) \cdots \land v_\rho \in (V \setminus F) \\
& \land (v_1 \neq v_2 \land v_1 \neq v_3 \land \cdots \land v_1 \neq v_\rho) \\
& \land (v_2 \neq v_1 \land v_2 \neq v_3 \land \cdots \land v_2 \neq v_\rho) \\
& \ldots \\
& \land (v_\rho \neq v_1 \land v_\rho \neq v_2 \land \cdots \land v_\rho \neq v_{\rho-1}) \\
& \land \neg((v_1, v_2) \in E \land (v_2, v_3) \in E \land \cdots \land (v_{\rho-1}, v_\rho) \in E \land (v_\rho, v_1) \in E)
\end{aligned} \tag{3}
$$

Courcelle's Theorem and MSOL Solvability of FVS-ρCL

Courcelle's Theorem states that any graph property definable in MSOL can be decided in linear time on graphs of bounded treewidth.

Theorem 7.11, p. 183
Parameterized Algorithms, Cygan et al., 2015

Corollary (MSOL Solvability of FVS-ρCL on Bounded-Treewidth Graphs)

Let $\rho \geq 3$ be a constant and $G = (V, E)$ be a graph of treewidth at most tw, *with vertex-weight function $w : V \to \mathbb{N}$. Then the minimum-weight set breaking all ρ-cycles (FVS-ρCL) can be computed in time:*

$$f(\rho, \mathrm{tw}) \cdot n$$

for some computable function f depending only on ρ and tw.

Dynamic Programming Algorithm for FVS-4CL over Nice Tree Decompositions

Algorithm Design for FVS-4CL

1. **Solution:** For the FVS-4CL problem, a solution for graph G is a set F such that $G - F$ contains no 4-cycles.
2. **Partial Solution:** For subgraph $G_i = (V_i, E_i)$, a partial solution F_i is a subset $F_i \subseteq V_i$, a restriction of a full solution.
3. **Extension of Partial Solution:** A solution F extends F_i if $F \cap V_i = F_i$.
4. **Characteristic of a Partial Solution:** For X_i, vertices are partitioned as:
   - $I \subseteq X_i$: vertices in the partial solution.
   - $\mathcal{F} = \{(v, u) \in X_i \times X_i : \exists w \in V_i \setminus X_i, (v, w), (u, w) \in E_i, w \notin F_i\}$

   $$ch(G_i, F_i) = (I, \mathcal{F})$$

   The valuation table $c[i, I, \mathcal{F}] \in \mathbb{N} \cup \{\infty\}$ gives the min. weight of F_i:

   $$c[i, I, \mathcal{F}] = \min\{w(W) : W \text{ is a FVS-4CL of } G_i \wedge W \cap X_i = I\}$$

5. **Full Set of Characteristics:** For node X_i, valuations exist for all

   $$I \in \{0,1\}^{|X_i|} \quad \text{and} \quad \mathcal{F} \in \{0,1\}^{|X_i \times X_i|}$$

   There are at most $2^{(\mathrm{tw}+1)} \cdot 2^{\binom{\mathrm{tw}+1}{2}} = 2^{\mathcal{O}(\mathrm{tw}^2)}$ entries.

DP Transitions over Tree Decomposition Nodes

Leaf Node: $X_i = \emptyset$

$$c[i, \emptyset, \emptyset] = 0$$

Introduce Node: Let $X_i = X_j \cup \{v\}$

$$c[i, I \cup \{v\}, \mathcal{F}] = w(v) + c[j, I, \mathcal{F}]$$

$$c[i, I, \mathcal{F}] = \begin{cases} \infty & \text{if } \exists u, w \notin I : (u, w) \in \mathcal{F},\ (v, u), (v, w) \in E_i \\ c[j, I, \mathcal{F}] & \text{otherwise} \end{cases}$$

Forget Node: Let $X_i = X_j \setminus \{v\}$

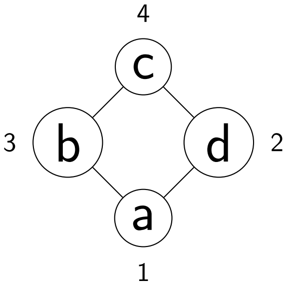
$$c[i, I, \mathcal{F}] = \min \begin{pmatrix} c[j, I \cup \{v\}, \mathcal{F}], \\ c[j, I, \mathcal{F} \cup \{(u, w) : (v, u), (v, w) \in E_i\}] \end{pmatrix}$$

Join Node: Let $X_i = X_{j_1} = X_{j_2}$

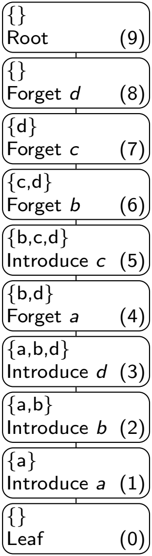
$$c[i, I, \mathcal{F}] = \min_{\mathcal{F}_1 \cup \mathcal{F}_2 = \mathcal{F}} (c[j_1, I, \mathcal{F}_1] + c[j_2, I, \mathcal{F}_2] - w(I))$$

Infeasible if $\exists u, w \notin I : (u, w) \in \mathcal{F}_1 \cap \mathcal{F}_2 \Rightarrow c[i, I, \mathcal{F}] = \infty$

Rhombus Graph and Tree Decomposition

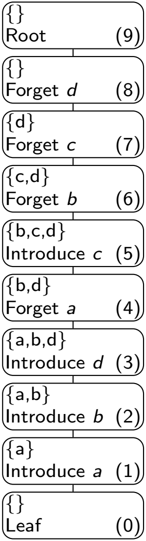


Rhombus Graph G



Nice Tree Decomposition
of the graph G with a width of 2.

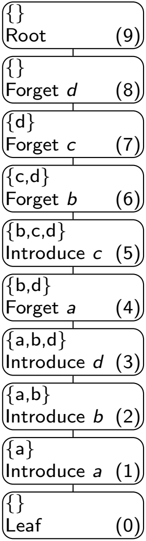
Rhombus Graph and Tree Decomposition



i	I	$\mathcal{F}$	**Value**
0	$\emptyset$	$\emptyset$	0
1	$\{a\}$	$\emptyset$	1
1	$\emptyset$	$\emptyset$	0
2	$\{a, b\}$	$\emptyset$	4
2	$\{a\}$	$\emptyset$	1
2	$\{b\}$	$\emptyset$	3
2	$\emptyset$	$\emptyset$	0
3	$\{a, b, d\}$	$\emptyset$	6
3	$\{a, b\}$	$\emptyset$	4
3	$\{a, d\}$	$\emptyset$	3
3	$\{b, d\}$	$\emptyset$	5
3	$\{a\}$	$\emptyset$	1
3	$\{b\}$	$\emptyset$	3
3	$\{d\}$	$\emptyset$	2
3	$\emptyset$	$\emptyset$	0
4	$\{b, d\}$	$\emptyset$	6
4	$\{b, d\}$	$\{(b, d)\}$	5
4	$\{b\}$	$\emptyset$	4
4	$\{b\}$	$\{(b, d)\}$	3

i	I	$\mathcal{F}$	**Value**
4	$\{d\}$	$\emptyset$	3
4	$\{d\}$	$\{(b, d)\}$	2
4	$\emptyset$	$\emptyset$	1
4	$\emptyset$	$\{(b, d)\}$	0
5	$\{b, c, d\}$	$\emptyset$	10
5	$\{b, c, d\}$	$\{(b, d)\}$	9
5	$\{b, d\}$	$\emptyset$	6
5	$\{b, d\}$	$\{(b, d)\}$	5
5	$\{b, c\}$	$\emptyset$	8
5	$\{b, c\}$	$\{(b, d)\}$	7
5	$\{b\}$	$\emptyset$	4
5	$\{b\}$	$\{(b, d)\}$	3
5	$\{c, d\}$	$\emptyset$	7
5	$\{c, d\}$	$\{(b, d)\}$	6
5	$\{d\}$	$\emptyset$	3
5	$\{d\}$	$\{(b, d)\}$	2
5	$\{c\}$	$\emptyset$	5
5	$\{c\}$	$\{(b, d)\}$	4
5	$\emptyset$	$\emptyset$	1
5	$\emptyset$	$\{(b, d)\}$	∞

Rhombus Graph and Tree Decomposition



i	I	$\mathcal{F}$	**Value**
6	$\{c, d\}$	$\emptyset$	10
6	$\{c, d\}$	$\emptyset$	9
6	$\{d\}$	$\emptyset$	6
6	$\{d\}$	$\emptyset$	5
6	$\{c\}$	$\emptyset$	8
6	$\{c\}$	$\emptyset$	7
6	$\emptyset$	$\emptyset$	4
6	$\emptyset$	$\emptyset$	3
6	$\{c, d\}$	$\emptyset$	7
6	$\{c, d\}$	$\emptyset$	6
6	$\{d\}$	$\emptyset$	3
6	$\{d\}$	$\emptyset$	2
6	$\{c\}$	$\emptyset$	5
6	$\{c\}$	$\emptyset$	4
6	$\emptyset$	$\emptyset$	1
6	$\emptyset$	$\emptyset$	∞

i	I	$\mathcal{F}$	**Value**
6	$\{c, d\}$	$\emptyset$	6
6	$\{d\}$	$\emptyset$	2
6	$\{c\}$	$\emptyset$	4
6	$\emptyset$	$\emptyset$	1
7	$\{d\}$	$\emptyset$	2
7	$\emptyset$	$\emptyset$	1
8	$\emptyset$	$\emptyset$	1
9	$\emptyset$	$\emptyset$	1

Beyond the Worst-Case Analysis

Beyond the Worst-Case Analysis

- Perturbation resilience
- m-Stitching and Π-m-Stitching
- Certified Algorithms

Definitions: γ-Perturbation and γ-Stability

Definition (γ-Perturbation for Vertex-Optimization Problems)

Let (G, w) be a weighted graph. For any $\gamma \in \mathbb{R}_{\geq 0}$, a γ-**perturbation** of the weight function $w : V \to \mathbb{N}$ is a function $w' : V \to \mathbb{R}$ such that:

$$w(v) \leq w'(v) \leq \gamma \cdot w(v) \quad \forall v \in V.$$

the number may be different for each parameter!

Definition (γ-Stability)

Let Π be a vertex-minimization problem. For any $\gamma \in \mathbb{R}_{\geq 0}$, a weighted graph (G, w) is called a γ-**stable instance** of Π if it admits a unique optimal solution S that remains optimal under all γ-perturbations of the weight function w.

Definition: Certified Algorithm

Definition (Certified Algorithm)

A γ-**certified solution** to an instance (G, w) of a weighted vertex-optimization problem Π is a pair (S, w'), where:

- w' is a γ-**perturbation** of the original weight function w, and
- S is an **optimal solution** for the instance (G, w').

A γ-**certified algorithm** for Π maps each instance (G, w) to a γ-certified solution.

Certified Algorithms and Perturbation Resilience

A γ-**certified algorithm**:

- **Exactly solves all γ-perturbation-resilient instances**.
- Always returns a γ-**approximate solution**.
- Gives a γ-**approximate solution** for the compliment problem.

Definition: *m*-stitching

Definition (*m*-stitching)

Assume $m \geq 0$ is an integer, J is an induced subgraph of G, and $S_1, S_2 \subseteq V(G)$. Then we define the *m-stitch* of S_2 onto S_1 along J as the set:

$$S_3 := (S_1 \setminus J) \cup (S_2 \cap N_G^m[J]).$$

Illustration of 2-stitching

$$S_3 := (S_1 \setminus J) \cup (S_2 \cap N_G^2[J]).$$

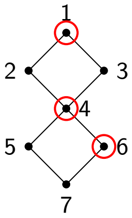


Figure: Vertex set S_1

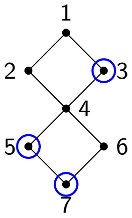


Figure: Vertex set S_2

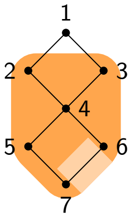


Figure: J and $N_G^2[J]$

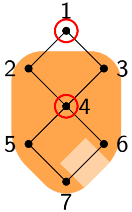


Figure: $S_1 \setminus J$

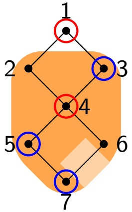


Figure: $S_2 \cap N_G^2[J]$

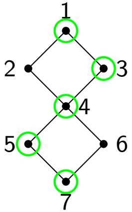


Figure: Final set S_3

Meta-Theorem for Minor-Closed Graph Classes

Theorem (Meta-Theorem for Minor-Closed Graph Classes)

Let $\mathcal{G}$ be a minor-closed graph class whose local treewidth is bounded by $g(r) = \lambda \cdot r$, for fixed $\lambda \in \mathbb{R}$ and $r \in \mathbb{N}$.
Let Π be a vertex-minimization problem such that:

1. *Π is guessable.*
2. *Π is m-stitchable.*
3. *There exists an algorithm A_Π that solves Π-m-stitching in time $f(t) \cdot |V(G)|^{\mathcal{O}(1)}$, where $t = \mathrm{tw}(G[N_G^m(J)])$ and f is computable.*

Then, for each $\epsilon > 0$, there exists a $(1+\epsilon)$-certified algorithm for Π running in time $f(\lambda \cdot m/\epsilon) \cdot |V(G)|^{\mathcal{O}(1)}$ on any input $(G, w : V(G) \to \mathbb{N})$, with $G \in \mathcal{G}$ and polynomially-bounded weights.

(Bumpus et al., 2024)

$(1+\epsilon)$-Certified Algorithm for FVS-4CL

Algorithm $(1+\epsilon)$-Certified algorithm for FVS-4CL

Require: Vertex-weighted planar graph $(G, w : V(G) \to \mathbb{N})$, $\epsilon > 0$

Ensure: A vertex set $S^* \subseteq V(G)$ and a $(1+\epsilon)$-perturbation w' of w such that S^* is optimal for FVS-4CL on (G, w')

1: $\kappa \leftarrow \left\lceil \frac{2m}{\epsilon} \right\rceil + 2m$, where $m \leftarrow 2$

2: Let S^* be a feasible solution (FVS-4CL is guessable)

3: Perform BFS from an arbitrary vertex r

4: **while** there exists a subgraph $J_{\kappa-2m}$ of width $\kappa - 2m$ such that $w_A((G,w), S^*, J_{\kappa-2m}) < w(S^*)$ **do**

5: $\quad S^* \leftarrow A((G,w), S^*, J_{\kappa-2m})$

6: **end while**

7: Define $w' : V(G) \to \mathbb{R}^+$ by:

$$w'(x) = \begin{cases} w(x) & \text{if } x \in S^* \\ (1+\epsilon)w(x) & \text{otherwise} \end{cases}$$

Summary of Contributions

- Establish **EPTAS for FVS-BCL** in planar graphs.
- Tractability:
  - Solved the **FVS-4CL** problem using **dynamic programming over nice tree decompositions**.
  - Solved the **FVS-BCL** problem using a **MSOL** formulation.
- Designed $(1+\epsilon)$-certified algorithms for **FVS-4CL** and **FVS-BCL**

Future Work

- Extend the DP algorithm over nice tree decompositions to FVS-BCL
- Improve the current algorithm that has a running time of $2^{\mathcal{O}(\mathrm{tw}^2)} \cdot n^{\mathcal{O}(1)}$
  - Cut & Count technique, which obtained a $3^{\mathcal{O}(\mathrm{tw})} \cdot n^{\mathcal{O}(1)}$ randomized algorithm **(Cygan et al., 2011)**
  - Deterministic $2^{\mathcal{O}(\mathrm{tw})} \cdot n^{\mathcal{O}(1)}$ rank-based approach **(Bodlaender et al., 2015)**
- Improve the certified algorithm for FVS-BCL.

Thank you!

Questions?

Appendix

- A.1 Approximation Schemes
- A.2 Certified Algorithms

Appendix Approximation Schemes

Bounds on the Optimum Solution (Unweighted Case)

Lemma (Bound on the Optimum Solution in Subgraphs)

Let $S_{i,j}$ be the minimum unweighted feedback vertex set (FVS-4CL) for subgraph $G_{i,j}$, computed using Algorithm 21, and let $F \equiv OPT$ be the optimum solution for the full graph G.
Define $F_{i,j} := F \cap V(G_{i,j})$, i.e., the restriction of the global solution F to the subgraph $G_{i,j}$.
Then:

$$|S_{i,j}| \leq |F_{i,j}|$$

Result (Optimum solution bound for the whole graph)

Let S_i be the union of optimal solutions defined on Line 6 of Algorithm 1 for some shift i. Let $F \equiv OPT$ be the optimum solution in G and let $F_{i,j} = G_{i,j} \cap F$. For those sets it holds that:

$$|S_i| \leq \sum_j |S_{i,j}| \leq \sum_j |F_{i,j}|$$

Bounds on the Optimum Solution (Unweighted Case)

Lemma (Bound for the vertices on the boundaries (Unweighted Case))

Let $F_i = F \cap \{\textit{vertices at levels } i \bmod \ell\}$*. The sets* F_i *are disjoint and* $\bigcup_i F_i = F$*.*
We claim that:

$$\exists q \in \{0, 1, \dots, \ell - 1\} : |F_q| + |F_{q+1}| \leq \frac{2}{\ell} \cdot |F|.$$

Lemma (Bound for the cardinality of the intersection sets)

Let $F \equiv OPT$ *be the optimal solution and* $F_{i,j} = F \cap G_{i,j}$ *be the intersection sets with the subgraphs. Then we have:*

$$\sum_j |F_{i^*,j}| \leq |F| + \frac{2}{\ell} \cdot |F|$$

for some specific integer i^**.*

$$|S_{i^*}| \leq \sum_j |S_{i^*,j}| \leq \sum_j |F_{i^*,j}| \leq (1 + \frac{2}{\ell}) \cdot |F| = (1 + 2\epsilon) \cdot |F|$$

Appendix Certified Algorithms

Meta-Theorem for Minor-Closed Graph Classes

Theorem (Meta-Theorem for Minor-Closed Graph Classes)

Let $\mathcal{G}$ be a minor-closed graph class whose local treewidth is bounded by $g(r) = \lambda \cdot r$, for fixed $\lambda \in \mathbb{R}$ and $r \in \mathbb{N}$.
Let Π be a vertex-minimization problem such that:

1. *Π is guessable.*
2. *Π is m-stitchable.*
3. *There exists an algorithm A_Π that solves Π-m-stitching in time $f(t) \cdot |V(G)|^{\mathcal{O}(1)}$, where $t = \mathrm{tw}(G[N_G^m(J)])$ and f is computable.*

Then, for each $\epsilon > 0$, there exists a $(1+\epsilon)$-certified algorithm for Π running in time $f(\lambda \cdot m/\epsilon) \cdot |V(G)|^{\mathcal{O}(1)}$ on any input $(G, w : V(G) \to \mathbb{N})$, with $G \in \mathcal{G}$ and polynomially-bounded weights.

Guessable Problems

Definition (Guessable)

A problem Π is *guessable* if there is an algorithm that outputs a feasible solution with no requirement for optimality in polynomial time.

In the case of FVS-BCL:
This set is $F \leftarrow V$ for a graph $G = (V, E)$.

Lemma: FVS-4CL is 2-stitchable

Lemma (FVS-4CL is 2-stitchable)

Let (G, w) be any instance of the FVS-4CL problem, let $J \subseteq V(G)$ be a subset of the vertices in the graph, and let S_1 and S_2 be any two feasible solutions to the problem in G. Then the set

$$S_3 := (S_1 \setminus J) \cup (S_2 \cap N_G^2[J])$$

is a feasible solution to the problem.

Stitch-FVS-4CL Algorithm

Algorithm Stitch-FVS-4CL

Require: Vertex-weighted planar graph $(G, w : V(G) \to \mathbb{N})$, a feasible solution S_1 on G, and a vertex set $J \subseteq V(G)$

Ensure: Feasible solution S' on G, such that for all feasible solutions S^*, $w(S') \leq w(S^* \oplus^2_{G,J} S_1)$

```
H ← G[N²_G[J] \ (S₁ \ J)]
Let S₂ be the output of algorithm A on input (H, w)
if w(S₂ ⊕²_{G,J} S₁) < w(S₁) then
    return S₂ ⊕²_{G,J} S₁
else
    return S₁
end if
```

Case-distinction proof

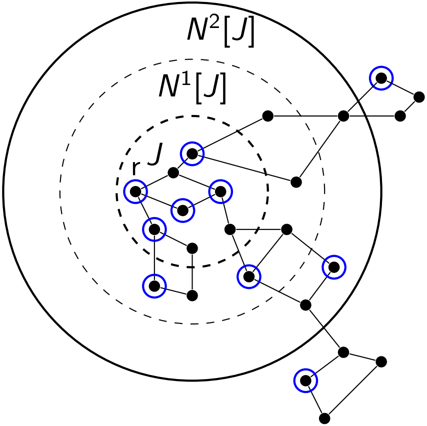


Figure: Sets $J \subseteq N^1_G[J] \subseteq N^2_G[J]$; Solution S_1 in blue

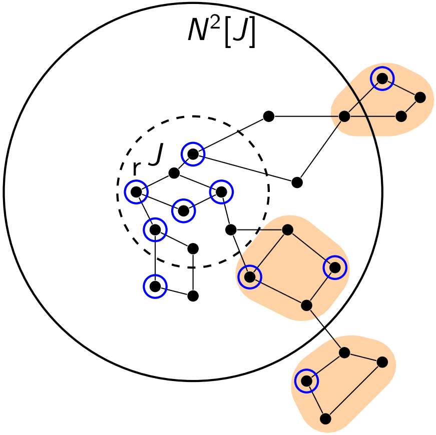


Figure: $\forall v \in \{v_1, v_2, v_3, v_4\}$, such that $v \in S_1$, it holds that $v \notin J$

Case-distinction proof

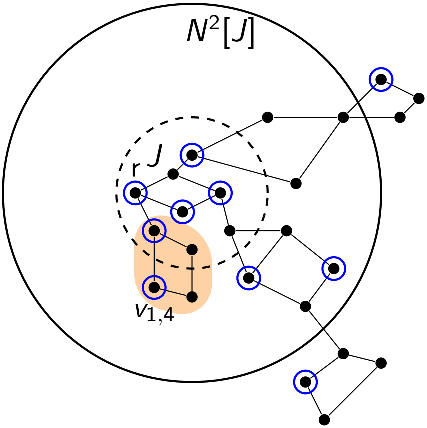


Figure: $\exists v \in \{v_1, v_2, v_3, v_4\}$, such that $v \in S_1$: $v \in J$

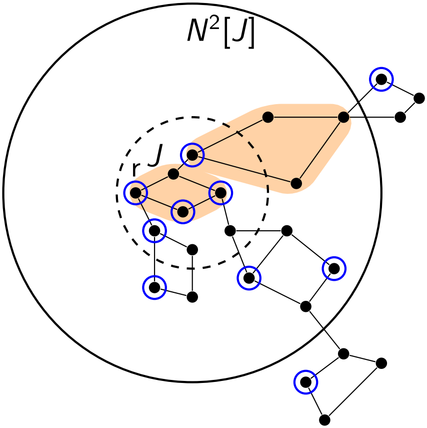


Figure: $\forall v \in \{v_1, v_2, v_3, v_4\}$, such that $v \in S_1$: $v \in J$

Meta-Theorem for Minor-Closed Graph Classes

Theorem (Meta-Theorem for Minor-Closed Graph Classes)

Let $\mathcal{G}$ be a minor-closed graph class whose local treewidth is bounded by $g(r) = \lambda \cdot r$, for fixed $\lambda \in \mathbb{R}$ and $r \in \mathbb{N}$.
Let Π be a vertex-minimization problem such that:

1. *Π is guessable.*
2. *Π is m-stitchable.*
3. *There exists an algorithm A_Π that solves Π-m-stitching in time $f(t) \cdot |V(G)|^{\mathcal{O}(1)}$, where $t = \mathrm{tw}(G[N_G^m(J)])$ and f is computable.*

Then, for each $\epsilon > 0$, there exists a $(1+\epsilon)$-certified algorithm for Π running in time $f(\lambda \cdot m/\epsilon) \cdot |V(G)|^{\mathcal{O}(1)}$ on any input $(G, w : V(G) \to \mathbb{N})$, with $G \in \mathcal{G}$ and polynomially-bounded weights.