# Improved Fixed-Parameter Algorithms for Two Feedback Set Problems

Jiong Guo   Jens Gramm   Falk Hüffner   Rolf Niedermeier
Sebastian Wernicke

Friedrich-Schiller-Universität Jena

# Outline

Introduction

Exact Algorithm for Feedback Vertex Set

Exact Algorithm for Edge Bipartization

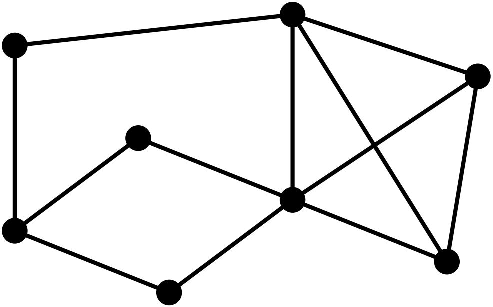
# Feedback Vertex Set

FEEDBACK VERTEX SET

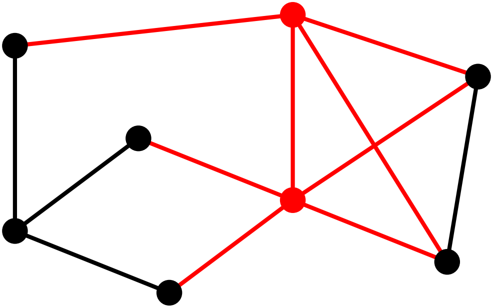
**Input:** *An undirected graph* $G = (V, E)$ *and a nonnegative integer* $k$.

**Task:** *Find a subset* $X \subseteq V$ *of vertices with* $|X| \leq k$ *such that* $G \setminus X$ *is cycle-free.*

# Feedback Vertex Set Example



# Feedback Vertex Set Example



# Known Results for Feedback Vertex Set

- Applications in VLSI design, Genome assembly, ...
- NP-complete

  [Karp 1972]
- MaxSNP-hard

  [Lund&Yannakakis, ICALP'93]
- Best known approximation: factor 2

  [Bafna, Berman&Fujito, SIDMA 1999]

# Parameterized Approach to Hard Problems

We want an exact algorithm, but that implies exponential runtime.
**Parameterized approach:** Try to confine the combinatorial explosion to $k$.

# Parameterized Approach to Hard Problems

We want an exact algorithm, but that implies exponential runtime.
**Parameterized approach:** Try to confine the combinatorial explosion to $k$.

## Definition

For some *parameter* $k$ of a problem of size $n$, the problem is called *fixed-parameter tractable* with respect to $k$ if there is an algorithm that solves it in $f(k) \cdot n^{O(1)}$ time.

# Parameterized Approach to Hard Problems

We want an exact algorithm, but that implies exponential runtime.
**Parameterized approach:** Try to confine the combinatorial explosion to $k$.

## Definition

For some *parameter* $k$ of a problem of size $n$, the problem is called *fixed-parameter tractable* with respect to $k$ if there is an algorithm that solves it in $f(k) \cdot n^{O(1)}$ time.

## Theorem (Downey&Fellows 1992)

Feedback Vertex Set *is fixed-parameter tractable with respect to* $k$.

# Parameterized Algorithms for Feedback Vertex Set

## Known Results

- $O((2k+1)^k \cdot n^2$
  [Downey&Fellows 1999]
- $O((4 \log k)^k \cdot n^{2.4})$
  [Raman, Saurabh&Subramanian, ISAAC'02]
- $O((2 \log k)^k \cdot n^2)$
  [Kanj, Pelsmajer&Schaefer, IWPEC'04]
- $O\left(\left(\frac{12 \log k}{\log \log k}\right)^k \cdot n^{2.4}\right)$
  [Raman, Saurabh&Subramanian, GRACO'05]

# Parameterized Algorithms for Feedback Vertex Set

## Known Results

- $O((2k+1)^k \cdot n^2)$
  [Downey&Fellows 1999]
- $O((4 \log k)^k \cdot n^{2.4})$
  [Raman, Saurabh&Subramanian, ISAAC'02]
- $O((2 \log k)^k \cdot n^2)$
  [Kanj, Pelsmajer&Schaefer, IWPEC'04]
- $O\left(\left(\frac{12 \log k}{\log \log k}\right)^k \cdot n^{2.4}\right)$
  [Raman, Saurabh&Subramanian, GRACO'05]

## Open Question

Is there an $O(c^k \cdot n^{O(1)})$ time algorithm for Feedback Vertex Set for some constant $c$?

# Iterative Compression for Feedback Vertex Set

## Idea

Use a *compression routine* iteratively.

[Reed, Smith&Vetta, Oper. Res. Lett. 2004]

*Compression routine*: Given a size-$(k+1)$ solution, either compute a size-$k$ solution or prove that there is no size-$k$ solution.

# Iterative Compression for Feedback Vertex Set

## Idea

Use a *compression routine* iteratively.

[Reed, Smith&Vetta, Oper. Res. Lett. 2004]

*Compression routine*: Given a size-$(k+1)$ solution, either compute a size-$k$ solution or prove that there is no size-$k$ solution.

*Algorithm:*

Start with empty graph $G'$ and empty feedback vertex set $X$

For each vertex $v$ in $G$:

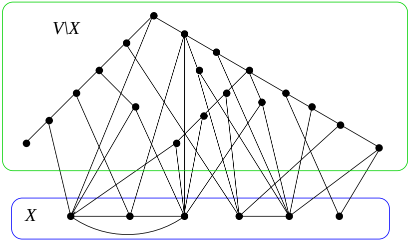
  Add $v$ to both $G'$ and $X$

  Compress $X$ using the compression routine

# Compression routine

## Task

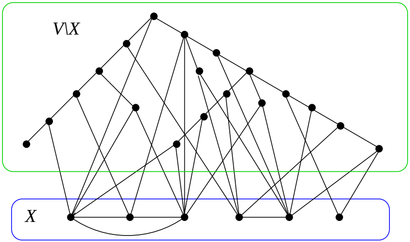
Given a graph $G$ and a feedback vertex set $X$ of size $k+1$. Find a feedback vertex set $X'$ of size $k$ or prove that there is none.



# Compression routine

## Task

Given a graph $G$ and a feedback vertex set $X$ of size $k+1$. Find a feedback vertex set $X'$ of size $k$ or prove that there is none.



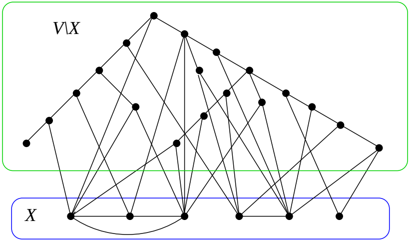

## Idea

Restrict our search for smaller solutions to *disjoint* solutions.

# Compression routine

## Task

Given a graph $G$ and a feedback vertex set $X$ of size $k+1$. Find a feedback vertex set $X'$ of size $k$ or prove that there is none.




## Idea

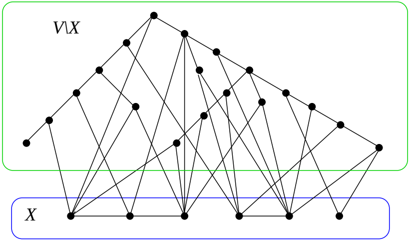
Restrict our search for smaller solutions to *disjoint* solutions.

Use brute force: Try all $2^{|X|}$ partitions of $X$ into a part to keep and a part to exchange.

# Compression routine (2)

## Task

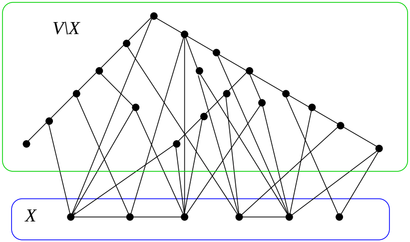
Given a graph $G$ and a feedback vertex set $X$ of size $k+1$. Find a feedback vertex set $X'$ of size $k$ in $V \setminus X$ or prove that there is none.



# Compression routine (2)

## Task

Given a graph $G$ and a feedback vertex set $X$ of size $k+1$. Find a feedback vertex set $X'$ of size $k$ in $V \setminus X$ or prove that there is none.



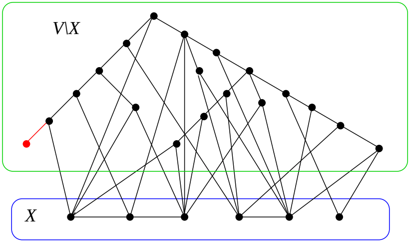

## Idea

Shrink the set of candidates for $X'$ by using data reduction rules.

# Compression routine (2)

## Task

Given a graph $G$ and a feedback vertex set $X$ of size $k+1$. Find a feedback vertex set $X'$ of size $k$ in $V \setminus X$ or prove that there is none.



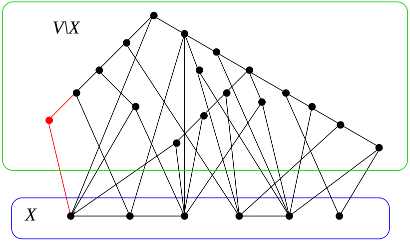

## Idea

Shrink the set of candidates for $X'$ by using data reduction rules.

# Compression routine (2)

## Task

Given a graph $G$ and a feedback vertex set $X$ of size $k+1$. Find a feedback vertex set $X'$ of size $k$ in $V \setminus X$ or prove that there is none.



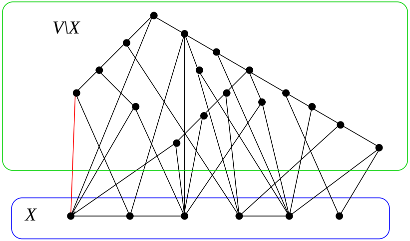

## Idea

Shrink the set of candidates for $X'$ by using data reduction rules.

# Compression routine (2)

## Task

Given a graph $G$ and a feedback vertex set $X$ of size $k+1$. Find a feedback vertex set $X'$ of size $k$ in $V \setminus X$ or prove that there is none.



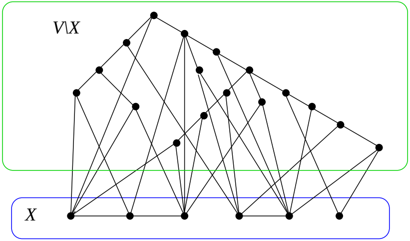

## Idea

Shrink the set of candidates for $X'$ by using data reduction rules.

# Compression routine (3)

### Task

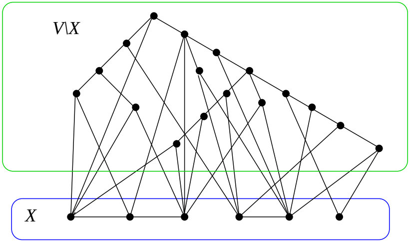
Given a graph $G$ with minimum vertex degree 3 and a feedback vertex set $X$ of size $k+1$. Find a feedback vertex set $X'$ of size $k$ in $V \setminus X$ or prove that there is none.



# Compression routine (3)

## Task

Given a graph $G$ with minimum vertex degree 3 and a feedback vertex set $X$ of size $k+1$. Find a feedback vertex set $X'$ of size $k$ in $V \setminus X$ or prove that there is none.




## Idea

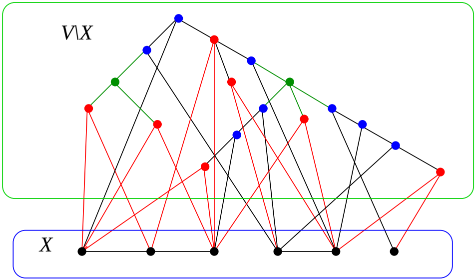
If $V \setminus X$ is too large as compared to $X$, then there is no solution $X'$ $\rightsquigarrow$ Use brute force!

# Bounding the Candidate Set

## Theorem

*If $|V \setminus X| > 14|X|$, then there is no solution $X'$.*

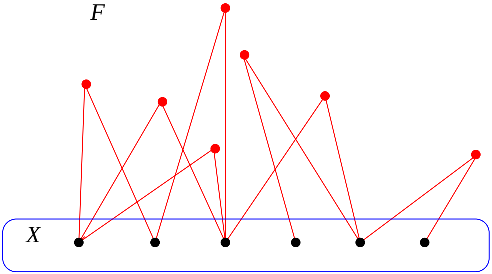
Separately bound the number of red, green, and blue vertices.




- at least two neighbors in $X$
- at least three neighbors in $V\backslash X$
- the rest

# Bounding the Candidate Set (2)

$F$: at least 2 neighbors in $X$

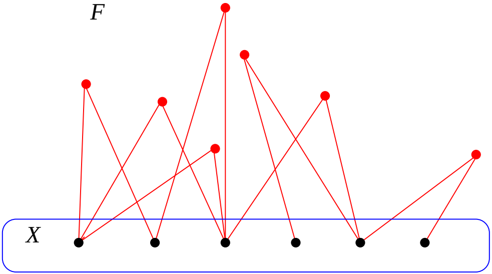



**Lemma**

*If $|F| \geq |X|$, then there is a cycle.*

# Bounding the Candidate Set (2)

$F$: at least 2 neighbors in $X$



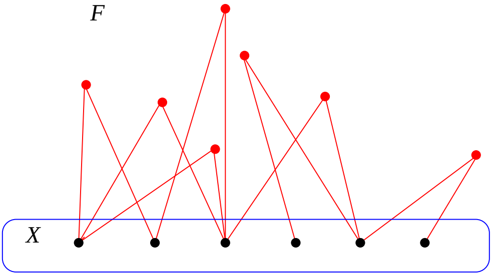

**Lemma**

*If $|F| \geq |X|$, then there is a cycle.*

- If $|F| \geq 2|X|$, then with $k = |X| - 1$ deletions we cannot get rid of all cycles.

# Bounding the Candidate Set (2)

$F$: at least 2 neighbors in $X$




### Lemma

*If $|F| \geq |X|$, then there is a cycle.*

- If $|F| \geq 2|X|$, then with $k = |X| - 1$ deletions we cannot get rid of all cycles.
- If there is a solution, there are at most $2k$ $F$-vertices.

# Feedback Vertex Set Algorithm

**Theorem**

*If* $|V \setminus X| > 14|X|$, *then we cannot compress the solution* $X$.

Running time by brute force for the compression step: $O(37.7^k \cdot m)$

# Feedback Vertex Set Algorithm

**Theorem**

*If* $|V \setminus X| > 14|X|$*, then we cannot compress the solution* $X$*.*

Running time by brute force for the compression step: $O(37.7^k \cdot m)$

**Theorem**

FEEDBACK VERTEX SET *with* $k$ *vertex deletions can be solved in* $O(c^k \cdot mn)$ *time for a constant* $c$*.*

[Independently shown by DEHNE et al., COCOON'05]

# Edge Bipartization

Edge Bipartization

**Input:** *An undirected graph* $G = (V, E)$ *and a nonnegative integer* $k$.

**Task:** *Find a subset* $X \subseteq E$ *of edges with* $|X| \leq k$ *such that* $G - X$ *is bipartite (i.e., 2-colorable).*

# Edge Bipartization

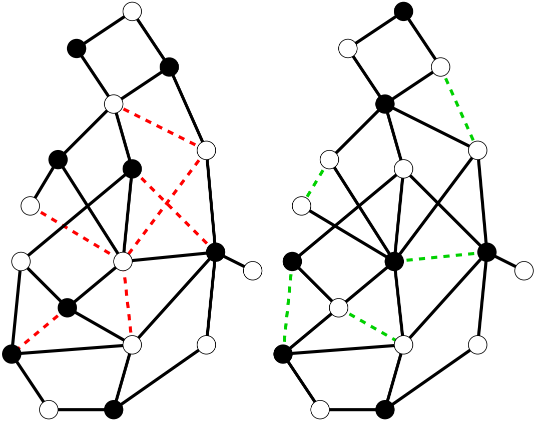
EDGE BIPARTIZATION
**Input:** *An undirected graph* $G = (V, E)$ *and a nonnegative integer* $k$.
**Task:** *Find a subset* $X \subseteq E$ *of edges with* $|X| \leq k$ *such that* $G - X$ *is bipartite (i.e., 2-colorable).*

- Numerous applications in computational biology, VLSI, register allocation, ...
- NP-complete
  [LEWIS&YANNAKAKIS, J. Comput. Syst. Sci. 1980]
- MaxSNP-hard
  [PAPADIMITRIOU&YANNAKAKIS, J. Comput. Syst. Sci. 1991]
- Best known approximation: factor of $\log |V|$
  [GARG, VAZIRANI&YANNAKAKIS, SIAM J. Comput. 1996]

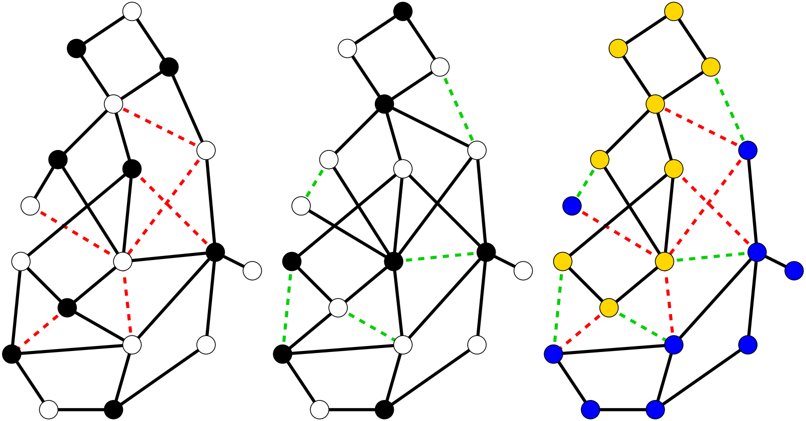
# Iterative Compression for Edge Bipartization

- Use a compression routine iteratively
- Restrict the compression routine to look for disjoint solutions by a simple input transformation

# Comparing Disjoint Edge Bipartization Sets

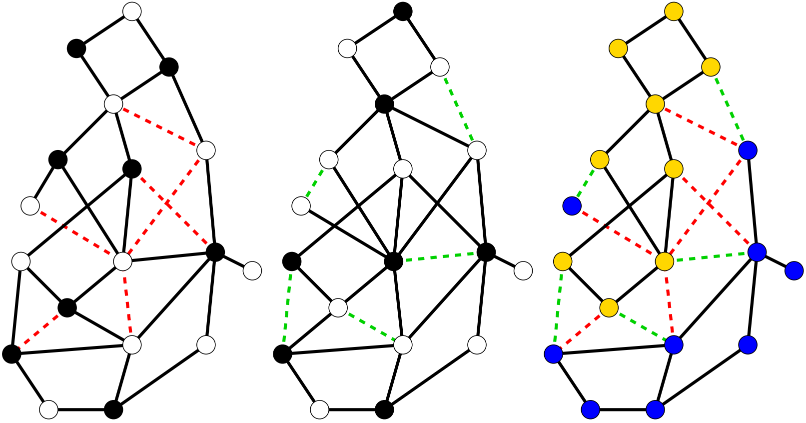


# Comparing Disjoint Edge Bipartization Sets



$$\Phi := \begin{cases} \text{🟡 for } (\bullet, \circ) \text{ or } (\circ, \bullet) \\ \text{🔵 for } (\bullet, \bullet) \text{ or } (\circ, \circ) \end{cases}$$

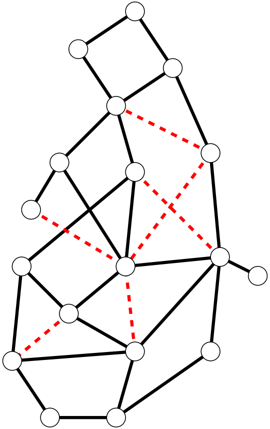
# Comparing Disjoint Edge Bipartization Sets



$$\Phi := \begin{cases} \text{yellow} \circ \text{ for } (\bullet, \circ) \text{ or } (\circ, \bullet) \\ \text{blue} \bullet \text{ for } (\bullet, \bullet) \text{ or } (\circ, \circ) \end{cases}$$

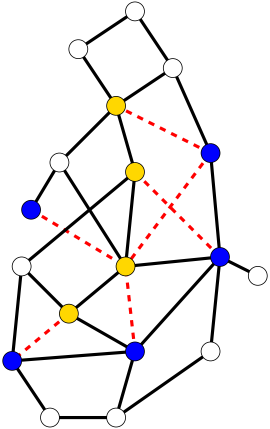
{red dashed, green dashed} is an edge cut between {yellow ●} and {blue ●}

# Discovering a smaller edge bipartization set



Given: $G = (V, E)$ and an edge bipartization $X \subseteq E$ without redundant edges (⁞)

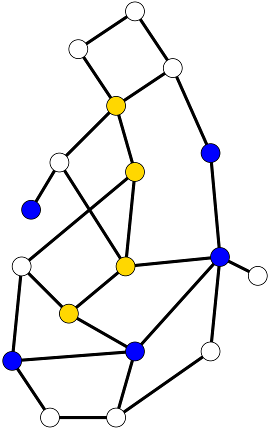
# Discovering a smaller edge bipartization set



Given: $G = (V, E)$ and an edge bipartization $X \subseteq E$ without redundant edges (⁞)

- Guess $\Phi$ at the endpoints of the edges in $X$

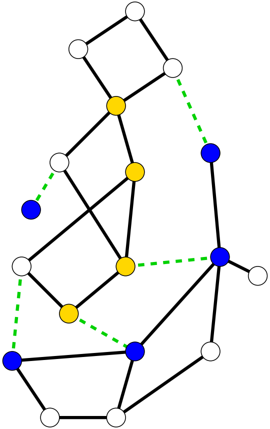
# Discovering a smaller edge bipartization set



Given: $G = (V, E)$ and an edge bipartization $X \subseteq E$ without redundant edges (⁝)

- Guess $\Phi$ at the endpoints of the edges in $X$
- Find a minimum edge cut between {●} and {●} with the Edmonds–Karp MaxFlow algorithm

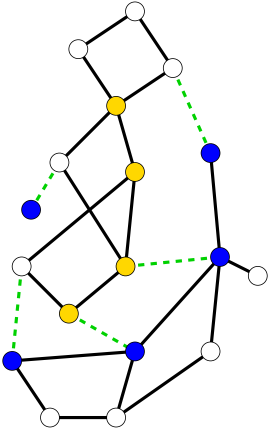
# Discovering a smaller edge bipartization set



Given: $G = (V, E)$ and an edge bipartization $X \subseteq E$ without redundant edges (⁝)

- Guess $\Phi$ at the endpoints of the edges in $X$
- Find a minimum edge cut between {●} and {●} with the Edmonds–Karp MaxFlow algorithm

# Discovering a smaller edge bipartization set



Given: $G = (V, E)$ and an edge bipartization $X \subseteq E$ without redundant edges (⋮)

- Guess $\Phi$ at the endpoints of the edges in $X$
- Find a minimum edge cut between {●} and {●} with the Edmonds–Karp MaxFlow algorithm
- Any such cut is a solution!

# Run Time for Edge Bipartization

- Compress $m$ times
- Try $2^k$ values for $\Phi$
- Find $k$ times an augmenting path in time $O(m)$

### Theorem

EDGE BIPARTIZATION *can be solved in* $O(2^k \cdot km^2)$ *time.*

# Conclusions and Outlook

- Significantly improved parameterized algorithms for FEEDBACK VERTEX SET and EDGE BIPARTIZATION
- Iterative compression is a promising method for the design of efficient fixed-parameter algorithms

Future work and open questions:

- Implementation and experiments
- Problem kernels
- DIRECTED FEEDBACK VERTEX SET