Stability in Graphs with Matroid Constraints

Fedor V. Fomin, Petr A. Golovach, Tuukka Korhonen, and Saket Saurabh

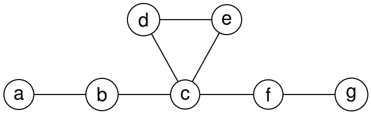
UNIVERSITY OF BERGEN

SWAT 2024

12 June 2024

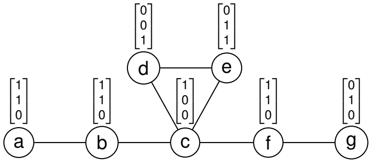
Frameworks

- Framework is a pair (G, M), where
  - $G = (V, E)$ is an undirected graph



Frameworks

- Framework is a pair (G, M), where
  - $G = (V, E)$ is an undirected graph
  - $M = (V, \mathcal{I})$ is a matroid

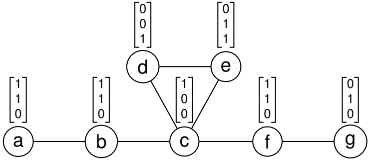


$$\begin{array}{ccccccc} a & b & c & d & e & f & g \end{array}$$
$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

Frameworks

- Framework is a pair (G, M), where
  - $G = (V, E)$ is an undirected graph
  - $M = (V, \mathcal{I})$ is a matroid

[Lovász '77]



$$
\begin{array}{ccccccc} a & b & c & d & e & f & g \end{array}
$$

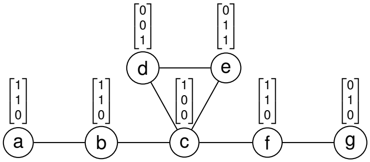
$$
\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}
$$

Frameworks

- Framework is a pair (G, M), where
  - $G = (V, E)$ is an undirected graph
  - $M = (V, \mathcal{I})$ is a matroid

[Lovász '77]

- Matroid is a pair $(V, \mathcal{I})$, where V is the ground set, and $\mathcal{I}$ is a family of *independent sets* $X \subseteq V$

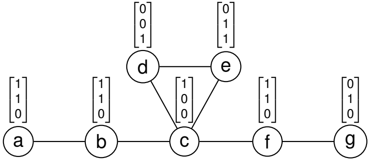


$$\begin{array}{ccccccc} a & b & c & d & e & f & g \end{array}$$
$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

Frameworks

- Framework is a pair (G, M), where
  - $G = (V, E)$ is an undirected graph
  - $M = (V, \mathcal{I})$ is a matroid

[Lovász '77]



a	b	c	d	e	f	g
1	1	1	0	0	1	0
1	1	0	0	1	1	1
0	0	0	1	1	0	0

- Matroid is a pair $(V, \mathcal{I})$, where V is the ground set, and $\mathcal{I}$ is a family of *independent sets* $X \subseteq V$ satisfying
  1. $\emptyset \in \mathcal{I}$
  2. if $X \in \mathcal{I}$ and $X' \subseteq X$ then $X' \in \mathcal{I}$
  3. if $X, Y \in \mathcal{I}$ and $|X| > |Y|$, then $\exists x \in X \setminus Y$ s.t. $Y \cup \{x\} \in \mathcal{I}$

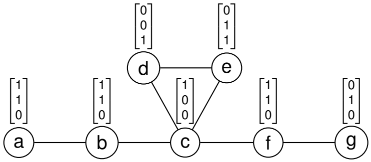
Frameworks

- Framework is a pair (G, M), where
  - $G = (V, E)$ is an undirected graph
  - $M = (V, \mathcal{I})$ is a matroid

[Lovász '77]

- Matroid is a pair $(V, \mathcal{I})$, where V is the ground set, and $\mathcal{I}$ is a family of *independent sets* $X \subseteq V$ satisfying
  1. $\emptyset \in \mathcal{I}$
  2. if $X \in \mathcal{I}$ and $X' \subseteq X$ then $X' \in \mathcal{I}$
  3. if $X, Y \in \mathcal{I}$ and $|X| > |Y|$, then $\exists x \in X \setminus Y$ s.t. $Y \cup \{x\} \in \mathcal{I}$

- Example: Linear matroid defined by a matrix A



$$\begin{array}{ccccccc} a & b & c & d & e & f & g \end{array}$$
$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

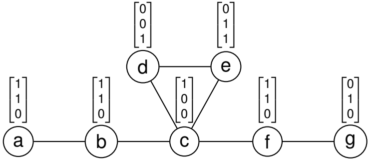
Frameworks

- Framework is a pair (G, M), where
  - $G = (V, E)$ is an undirected graph
  - $M = (V, \mathcal{I})$ is a matroid

[Lovász '77]

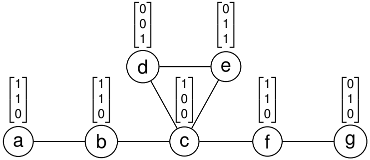
- Matroid is a pair $(V, \mathcal{I})$, where V is the ground set, and $\mathcal{I}$ is a family of *independent sets* $X \subseteq V$ satisfying
  1. $\emptyset \in \mathcal{I}$
  2. if $X \in \mathcal{I}$ and $X' \subseteq X$ then $X' \in \mathcal{I}$
  3. if $X, Y \in \mathcal{I}$ and $|X| > |Y|$, then $\exists x \in X \setminus Y$ s.t. $Y \cup \{x\} \in \mathcal{I}$

- Example: Linear matroid defined by a matrix A
  - V corresponds to the columns of A



$$\begin{array}{ccccccc} a & b & c & d & e & f & g \end{array}$$
$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

Frameworks



- Framework is a pair (G, M), where
  - $G = (V, E)$ is an undirected graph
  - $M = (V, \mathcal{I})$ is a matroid

[Lovász '77]

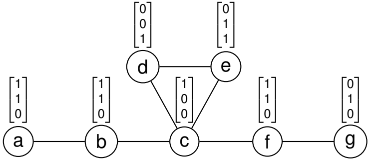
- Matroid is a pair $(V, \mathcal{I})$, where V is the ground set, and $\mathcal{I}$ is a family of *independent sets* $X \subseteq V$ satisfying
  1. $\emptyset \in \mathcal{I}$
  2. if $X \in \mathcal{I}$ and $X' \subseteq X$ then $X' \in \mathcal{I}$
  3. if $X, Y \in \mathcal{I}$ and $|X| > |Y|$, then $\exists x \in X \setminus Y$ s.t. $Y \cup \{x\} \in \mathcal{I}$

$$\begin{array}{ccccccc} a & b & c & d & e & f & g \end{array}$$
$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

- Example: Linear matroid defined by a matrix A
  - V corresponds to the columns of A
  - $X \subseteq V$ is independent if the columns X of A are linearly independent

Frameworks

- Framework is a pair (G, M), where
  - $G = (V, E)$ is an undirected graph
  - $M = (V, \mathcal{I})$ is a matroid



$$\begin{array}{ccccccc} a & b & c & d & e & f & g \end{array}$$
$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

[Lovász '77]

- Matroid is a pair $(V, \mathcal{I})$, where V is the ground set, and $\mathcal{I}$ is a family of *independent sets* $X \subseteq V$ satisfying
  1. $\emptyset \in \mathcal{I}$
  2. if $X \in \mathcal{I}$ and $X' \subseteq X$ then $X' \in \mathcal{I}$
  3. if $X, Y \in \mathcal{I}$ and $|X| > |Y|$, then $\exists x \in X \setminus Y$ s.t. $Y \cup \{x\} \in \mathcal{I}$

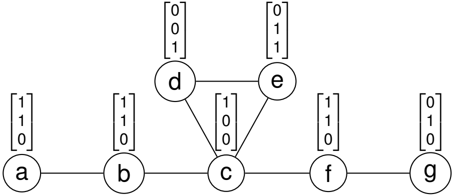
- Example: Linear matroid defined by a matrix A
  - V corresponds to the columns of A
  - $X \subseteq V$ is independent if the columns X of A are linearly independent

- This work: M is either given by an independence oracle or represented by a matrix

Independent Stable Set problem

Independent Stable Set problem

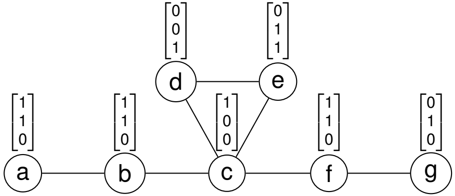
- Input:
  - Framework (G, M)



a	b	c	d	e	f	g
1	1	1	0	0	1	0
1	1	0	0	1	1	1
0	0	0	1	1	0	0

Independent Stable Set problem

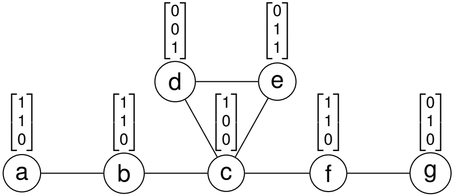
- Input:
  - Framework (G, M)
  - Integer k



$$\begin{array}{ccccccc} a & b & c & d & e & f & g \end{array}$$
$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

Independent Stable Set problem

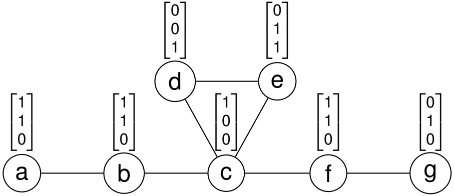
- Input:
  - Framework (G, M)
  - Integer k
- Output: Set $X \subseteq V$ of size $|X| = k$ that is
  - an independent set in M
  - a stable set in G (independent set)



a	b	c	d	e	f	g
1	1	1	0	0	1	0
1	1	0	0	1	1	1
0	0	0	1	1	0	0

Independent Stable Set problem

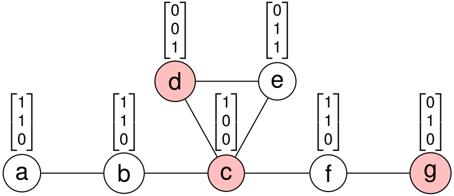
- Input:
  - Framework (G, M)
  - Integer k
- Output: Set $X \subseteq V$ of size $|X| = k$ that is
  - an independent set in M
  - a stable set in G (independent set)
- Example: $k = 3$



$$\begin{array}{ccccccc} a & b & c & d & e & f & g \end{array}$$
$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

Independent Stable Set problem

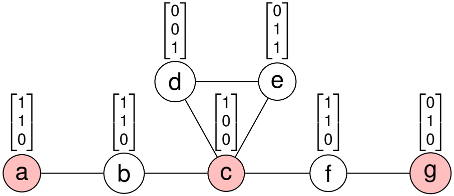
- Input:
  - Framework (G, M)
  - Integer k
- Output: Set $X \subseteq V$ of size $|X| = k$ that is
  - an independent set in M
  - a stable set in G (independent set)
- Example: $k = 3$



$$\begin{array}{ccccccc} a & b & c & d & e & f & g \end{array}$$
$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

Independent Stable Set problem

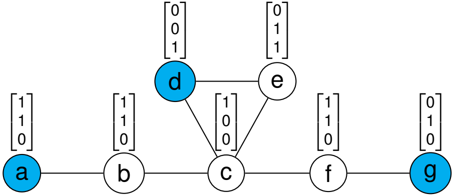
- Input:
  - Framework (G, M)
  - Integer k
- Output: Set $X \subseteq V$ of size $|X| = k$ that is
  - an independent set in M
  - a stable set in G (independent set)
- Example: $k = 3$



$$\begin{array}{ccccccc} a & b & c & d & e & f & g \end{array}$$
$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

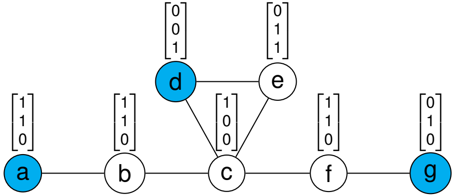
Independent Stable Set problem

- Input:
  - Framework (G, M)
  - Integer k
- Output: Set $X \subseteq V$ of size $|X| = k$ that is
  - an independent set in M
  - a stable set in G (independent set)
- Example: $k = 3$



$$\begin{array}{ccccccc} a & b & c & d & e & f & g \end{array}$$
$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

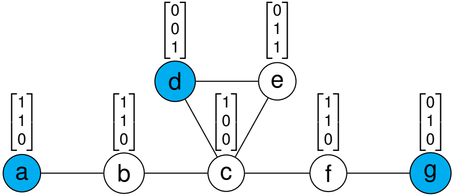
Independent Stable Set: Special cases



$$\begin{array}{ccccccc} a & b & c & d & e & f & g \end{array}$$
$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

Independent Stable Set: Special cases

- Partition matroid

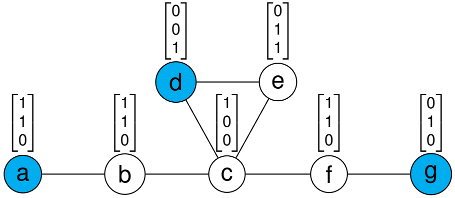


$$
\begin{array}{ccccccc}
a & b & c & d & e & f & g
\end{array}
$$

$$
\begin{bmatrix}
1 & 1 & 1 & 0 & 0 & 1 & 0 \\
1 & 1 & 0 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 1 & 1 & 0 & 0
\end{bmatrix}
$$

Independent Stable Set: Special cases

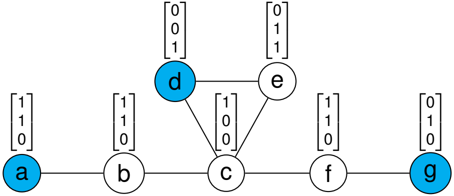
- Partition matroid
  - Rainbow stable set/independent transversal [Aharoni et al. ’23, Kim et al. ’22, ...]



$$\begin{array}{ccccccc} a & b & c & d & e & f & g \end{array}$$
$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

Independent Stable Set: Special cases

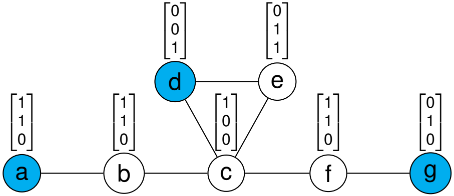
- Partition matroid
  - Rainbow stable set/independent transversal [Aharoni et al. ’23, Kim et al. ’22, ...]
  - Rainbow matching [Drisko ’98, Itai Rodeh, Tanimoto ’78, ...]



a	b	c	d	e	f	g
1	1	1	0	0	1	0
1	1	0	0	1	1	1
0	0	0	1	1	0	0

Independent Stable Set: Special cases

- Partition matroid
  - Rainbow stable set/independent transversal [Aharoni et al. '23, Kim et al. '22, ...]
  - Rainbow matching [Drisko '98, Itai Rodeh, Tanimoto '78, ...]
- Transversal matroid
  - Bipartite matching with separation [Manurangsi, Segal-Halevi, Suksompong '23]



$$\begin{array}{ccccccc} a & b & c & d & e & f & g \end{array}$$
$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

Hardness of Independent Stable Set

Hardness of Independent Stable Set

- NP-hard even when G is a path and M is a transversal matroid [Manurangsi, Segal-Halevi, Suksompong '23]

Hardness of Independent Stable Set

- NP-hard even when G is a path and M is a transversal matroid [Manurangsi, Segal-Halevi, Suksompong '23]
- Directly from stable set: No $f(k) \cdot n^{o(k)}$ time algorithm assuming ETH, even when M is uniform matroid

Hardness of Independent Stable Set

- NP-hard even when G is a path and M is a transversal matroid [Manurangsi, Segal-Halevi, Suksompong '23]
- Directly from stable set: No $f(k) \cdot n^{o(k)}$ time algorithm assuming ETH, even when M is uniform matroid

Theorem (This work)

When M is given by independence oracle, no $f(k) \cdot n^{o(k)}$ time algorithm (unconditionally)

Hardness of Independent Stable Set

- NP-hard even when G is a path and M is a transversal matroid [Manurangsi, Segal-Halevi, Suksompong '23]
- Directly from stable set: No $f(k) \cdot n^{o(k)}$ time algorithm assuming ETH, even when M is uniform matroid

Theorem (This work)

When M is given by independence oracle, no $f(k) \cdot n^{o(k)}$ time algorithm (unconditionally), even on

- bipartite graphs
- chordal graphs
- claw-free graphs
- AT-free graphs

Algorithms on sparse graphs

Algorithms on sparse graphs

Setting:

- M given by independence oracle
- parameterization by solution size k and degeneracy d

Algorithms on sparse graphs

Setting:

- M given by independence oracle
- parameterization by solution size k and degeneracy d

Theorem (This work)

$\mathcal{O}((d+1)^k \cdot n)$ time algorithm

Algorithms on sparse graphs

Setting:

- M given by independence oracle
- parameterization by solution size k and degeneracy d

Theorem (This work)

$\mathcal{O}((d+1)^k \cdot n)$ time algorithm

Theorem (This work)

Kernel of size $k^{\mathcal{O}(d)}$ with running time $n^{\mathcal{O}(d)}$

Algorithms on sparse graphs

Setting:

- M given by independence oracle
- parameterization by solution size k and degeneracy d

Theorem (This work)

$\mathcal{O}((d+1)^k \cdot n)$ time algorithm

Theorem (This work)

Kernel of size $k^{\mathcal{O}(d)}$ with running time $n^{\mathcal{O}(d)}$

(the kernel only deletes vertices)

Algorithms on sparse graphs

Setting:

- M given by independence oracle
- parameterization by solution size k and degeneracy d

Theorem (This work)

$\mathcal{O}((d+1)^k \cdot n)$ time algorithm

Theorem (This work)

Kernel of size $k^{\mathcal{O}(d)}$ with running time $n^{\mathcal{O}(d)}$

(the kernel only deletes vertices)

Theorem (This work)

No polynomial kernel parameterized by $k + d$, unless NP $\subseteq$ coNP$/$poly

FPT algorithm parameterized by k, d

Theorem (This work)

$\mathcal{O}((d+1)^k \cdot n)$ time algorithm

Proof:

FPT algorithm parameterized by k, d

Theorem (This work)

$\mathcal{O}((d+1)^k \cdot n)$ time algorithm

Proof:

Step 1: Remove vertices v s.t. $\{v\}$ not independent

FPT algorithm parameterized by k, d

Theorem (This work)

$\mathcal{O}((d+1)^k \cdot n)$ time algorithm

Proof:

Step 1: Remove vertices v s.t. $\{v\}$ not independent

Lemma

For every vertex v, exists an optimal solution that intersects $N[v]$

FPT algorithm parameterized by k, d

Theorem (This work)

$\mathcal{O}((d+1)^k \cdot n)$ time algorithm

Proof:

Step 1: Remove vertices v s.t. $\{v\}$ not independent

Lemma

For every vertex v, exists an optimal solution that intersects $N[v]$

Step 2: Pick a vertex v with degree $\leq d$ and branch

poly(k) kernel on trees

poly(k) kernel on trees

Assume M truncated to rank k

poly(k) kernel on trees

Assume M truncated to rank k

Step 1: $k^2\Delta$ size kernel, where Δ is max degree

poly(k) kernel on trees

Assume M truncated to rank k

Step 1: $k^2\Delta$ size kernel, where Δ is max degree

- Greedily pick $k\Delta$ disjoint maximal independent sets

poly(k) kernel on trees

Assume M truncated to rank k

Step 1: $k^2\Delta$ size kernel, where Δ is max degree

- Greedily pick $k\Delta$ disjoint maximal independent sets
- Discard all other vertices

poly(k) kernel on trees

Assume M truncated to rank k

Step 1: $k^2\Delta$ size kernel, where Δ is max degree

- Greedily pick $k\Delta$ disjoint maximal independent sets
- Discard all other vertices
- Any discarded vertex in an independent set can be exchanged to at least $k\Delta$ different vertices not discarded

poly(k) kernel on trees

Assume M truncated to rank k

Step 1: $k^2\Delta$ size kernel, where Δ is max degree

- Greedily pick $k\Delta$ disjoint maximal independent sets
- Discard all other vertices
- Any discarded vertex in an independent set can be exchanged to at least $k\Delta$ different vertices not discarded

Step 2: Reducing the degree

poly(k) kernel on trees

Assume M truncated to rank k

Step 1: $k^2\Delta$ size kernel, where Δ is max degree

- Greedily pick $k\Delta$ disjoint maximal independent sets
- Discard all other vertices
- Any discarded vertex in an independent set can be exchanged to at least $k\Delta$ different vertices not discarded

Step 2: Reducing the degree

- Let v be a vertex of degree $> k^2$

poly(k) kernel on trees

Assume M truncated to rank k

Step 1: $k^2\Delta$ size kernel, where Δ is max degree

- Greedily pick $k\Delta$ disjoint maximal independent sets
- Discard all other vertices
- Any discarded vertex in an independent set can be exchanged to at least $k\Delta$ different vertices not discarded

Step 2: Reducing the degree

- Let v be a vertex of degree $> k^2$
- Pick k disjoint maximal independent sets in $N(v)$, and discard other vertices in $N(v)$

poly(k) kernel on trees

Assume M truncated to rank k

Step 1: $k^2\Delta$ size kernel, where Δ is max degree

- Greedily pick $k\Delta$ disjoint maximal independent sets
- Discard all other vertices
- Any discarded vertex in an independent set can be exchanged to at least $k\Delta$ different vertices not discarded

Step 2: Reducing the degree

- Let v be a vertex of degree $> k^2$
- Pick k disjoint maximal independent sets in $N(v)$, and discard other vertices in $N(v)$

$\Rightarrow k^4$ size kernel

Conclusion

Conclusion

- Independent Stable Set problem: Given a framework (G, M), find a set $X \subseteq V$ that is independent in M and stable in G

Conclusion

- Independent Stable Set problem: Given a framework (G, M), find a set $X \subseteq V$ that is independent in M and stable in G
- NP-hard already when G is a path

Conclusion

- Independent Stable Set problem: Given a framework (G, M), find a set $X \subseteq V$ that is independent in M and stable in G
- NP-hard already when G is a path
- This work:

Conclusion

- Independent Stable Set problem: Given a framework (G, M), find a set $X \subseteq V$ that is independent in M and stable in G
- NP-hard already when G is a path
- This work:
  - FPT algorithm and kernel on bounded-degeneracy graphs

Conclusion

- Independent Stable Set problem: Given a framework (G, M), find a set $X \subseteq V$ that is independent in M and stable in G
- NP-hard already when G is a path
- This work:
  - FPT algorithm and kernel on bounded-degeneracy graphs
  - Unconditional $f(k) \cdot n^{o(k)}$ lower bounds when M given by oracle

Conclusion

- Independent Stable Set problem: Given a framework (G, M), find a set $X \subseteq V$ that is independent in M and stable in G
- NP-hard already when G is a path
- This work:
  - FPT algorithm and kernel on bounded-degeneracy graphs
  - Unconditional $f(k) \cdot n^{o(k)}$ lower bounds when M given by oracle
- Future directions:

Conclusion

- Independent Stable Set problem: Given a framework (G, M), find a set $X \subseteq V$ that is independent in M and stable in G
- NP-hard already when G is a path
- This work:
  - FPT algorithm and kernel on bounded-degeneracy graphs
  - Unconditional $f(k) \cdot n^{o(k)}$ lower bounds when M given by oracle
- Future directions:
  - FPT algorithms for linear matroids?

Conclusion

- Independent Stable Set problem: Given a framework (G, M), find a set $X \subseteq V$ that is independent in M and stable in G
- NP-hard already when G is a path
- This work:
  - FPT algorithm and kernel on bounded-degeneracy graphs
  - Unconditional $f(k) \cdot n^{o(k)}$ lower bounds when M given by oracle
- Future directions:
  - FPT algorithms for linear matroids?
  - Weighted stable set?

Conclusion

- Independent Stable Set problem: Given a framework (G, M), find a set $X \subseteq V$ that is independent in M and stable in G
- NP-hard already when G is a path
- This work:
  - FPT algorithm and kernel on bounded-degeneracy graphs
  - Unconditional $f(k) \cdot n^{o(k)}$ lower bounds when M given by oracle
- Future directions:
  - FPT algorithms for linear matroids?
  - Weighted stable set?
  - (Induced) subgraph isomorphism on frameworks?

Conclusion

- Independent Stable Set problem: Given a framework (G, M), find a set $X \subseteq V$ that is independent in M and stable in G
- NP-hard already when G is a path
- This work:
  - FPT algorithm and kernel on bounded-degeneracy graphs
  - Unconditional $f(k) \cdot n^{o(k)}$ lower bounds when M given by oracle
- Future directions:
  - FPT algorithms for linear matroids?
  - Weighted stable set?
  - (Induced) subgraph isomorphism on frameworks?

Thank you!

Linear matroids

Recall: No $f(k) \cdot n^{o(k)}$ algorithm on chordal graphs when M given by independence oracle

Theorem (This work)

There is a $2^{\mathcal{O}(k)} n^{\mathcal{O}(1)}$ time algorithm for Independent Stable Set when G is chordal and M is a linear matroid given by its representation.

(but no polynomial kernel, unless NP $\subseteq$ coNP/poly)

Idea:

- Dynamic programming over tree decomposition with representative sets

In contrast, Independent Stable Set is W[1]-hard when G is bipartite and M is linear