# Covering Small Independent Sets and Separators (with Applications)

**Recent Advances in Algorithms**
NISER Bhubaneswar
February 10th, 2019

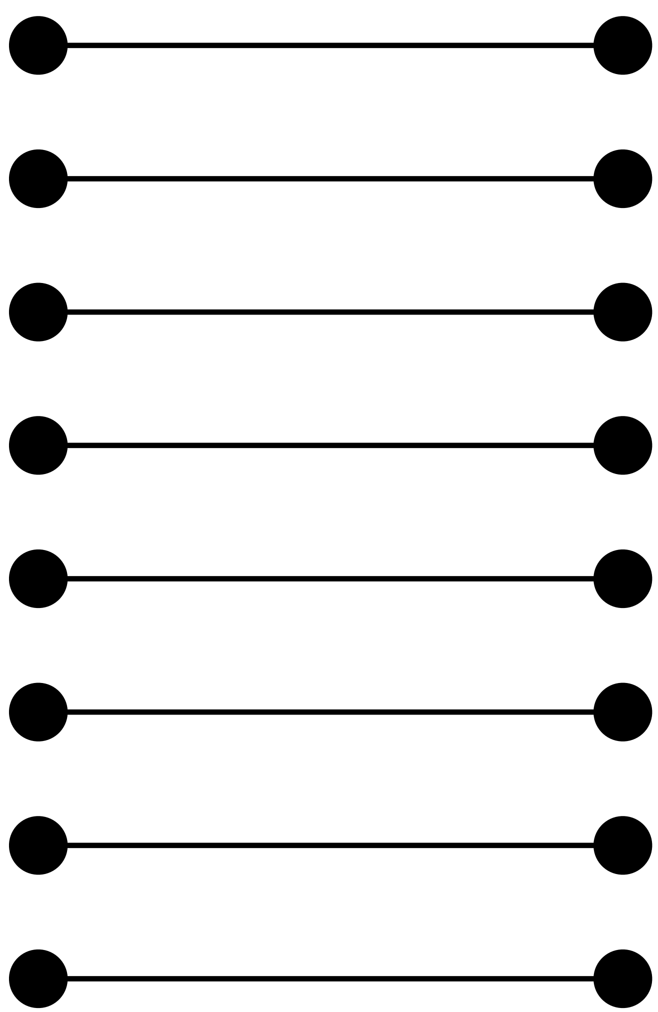
Daniel Lokshtanov, Fahad Panolan, Saket Saurabh,
Roohani Sharma, Meirav Zehavi

# Table of Contents

~~Table of Contents~~ Behind the scenes

1. A Combinatorial Question - Tool 1
2. Applications of Tool 1
3. Stumble upon Literature
4. Resolve some open questions using Tool 1
5. Need for the design of Tool 2
6. Design of Tool 2
7. Concluding Remarks

Given a graph G and an integer k, an independent set covering family (ISCF) for (G,k) is a family of independent sets of G, say $\mathscr{F}$(G,k), such that for any independent set X of G of size at most k, there exists Y ⊆ $\mathscr{F}$(G,k), such that X ⊆ Y.



n edges

| | |
|---|---|
| $\mathscr{F}$(G,1) | 2 |
| $\mathscr{F}$(G,2) | n + 2 |
| $\mathscr{F}$(G,3) | $2\binom{\mathbf{n}}{\mathbf{2}} + 2$ |
| $\mathscr{F}$(G,k) | $2^k\binom{\mathbf{n}}{\mathbf{k}} + 2$ |

$$\approx \mathbf{n}^{\mathcal{O}(\mathbf{k})}$$

Can the dependence of k be removed from the exponent on n?

?

Tool 1:

## Independent Set Covering Lemma (ISCL)

If G is d-degenerate, then for any k, there is an ISCF for (G,k) of size $2^{O(k \log kd)} \log n$.

In fact, such a family can be found in $2^{O(k \log kd)} (n+m) \cdot \log n$ time.

# Towards Randomized Independent Set Covering Lemma

*Goal*

Given: A d-degenerate graph G, an integer k

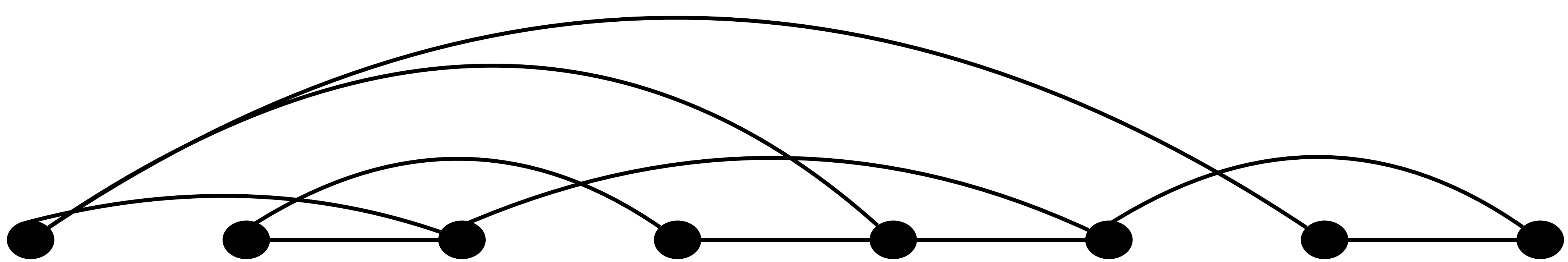
Output: An independent set Y such that

for any independent set X of size at most k, the Pr($X \subseteq Y$) $\geq \frac{1}{2^{k(d+1)}}$

*Experiment*

For each vertex v ∈ V(G), colour it either red or blue, uniformly at random.

Graph G



# Towards Randomized Independent Set Covering Lemma

*Goal*

Given: A d-degenerate graph G, an integer k

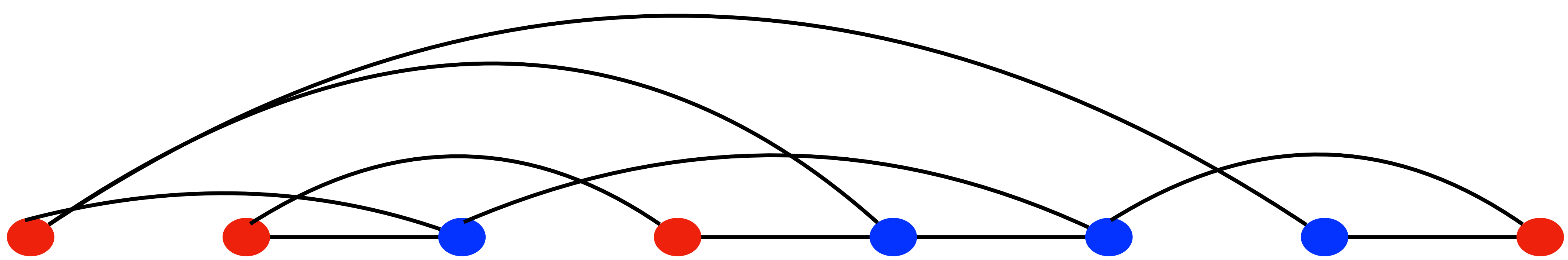
Output: An independent set Y such that

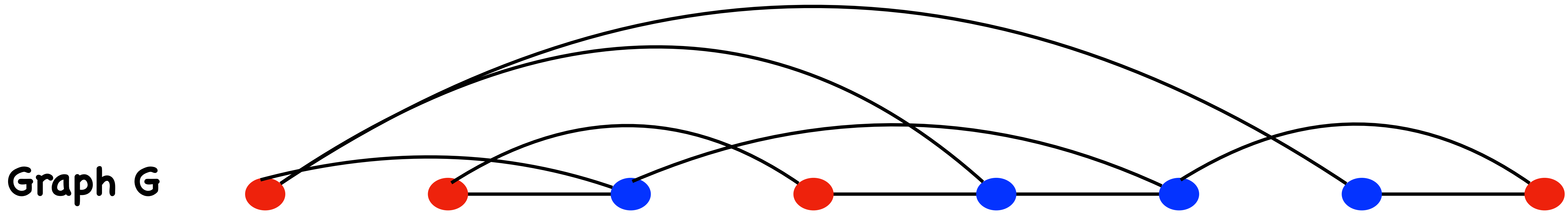
for any independent set X of size at most k, the Pr($X \subseteq Y$)$\geq \frac{1}{2^{k(d+1)}}$

*Experiment*

For each vertex $v \in V(G)$, colour it either red or blue, uniformly at random.

Graph G

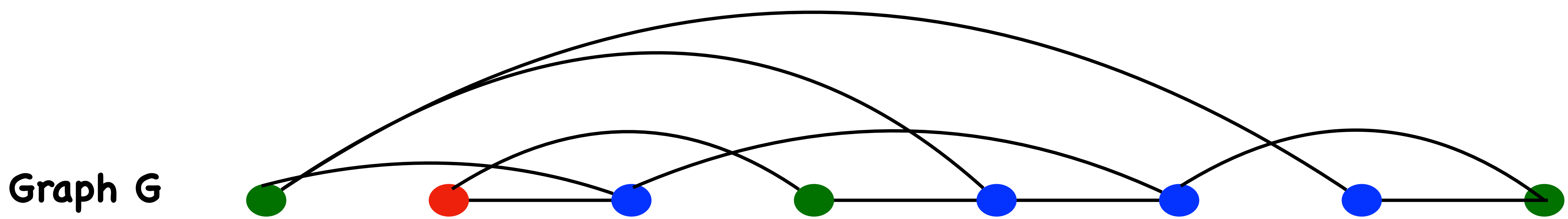





RED = set of all vertices that are coloured red

BLUE = set of all vertices that are coloured blue

GOOD EVENT = RED contains all vertices of X and none of its forward neighbours (i.e. all the forward neighbours of X are in BLUE)




IND_RED= {v: v $\in$ RED and all its forward neighbours in BLUE}

Claim : If GOOD EVENT happens, then X $\subseteq$ IND_RED

Pr(GOOD EVENT) $\geq \frac{1}{2^{|X|}} \frac{1}{2^{|N_f(X)|}} \geq \frac{1}{2^{k(d+1)}}$

# Towards Randomized Independent Set Covering Lemma

*Goal*

Given: A d-degenerate graph G, an integer k

Output: An independent set Y such that

for any independent set X of size at most k, the Pr($X \subseteq Y$) $\geq \frac{1}{2^{k(d+1)}}$

*Experiment*

For each vertex v ∈ V(G), colour it either red or blue, uniformly at random.

# Towards Randomized Independent Set Covering Lemma

*Goal*

Given: A d-degenerate graph G, an integer k

Output: An independent set Y such that

for any independent set X of size at most k, the Pr($X \subseteq Y$) ≥ ~~$\frac{1}{2^{k(d+1)}}$~~

$\frac{1}{2^{\mathcal{O}(k \log kd)}}$

*Experiment*

For each vertex v ∈ V(G), colour it either red or blue, ~~uniformly at random~~.

color v red with probability $\frac{1}{d+1}$

color v blue with probability $\frac{d}{d+1}$

# Randomized Independent Set Covering Lemma

Given: A d-degenerate graph G, an integer k

Output: An independent set Y such that

for any independent set X of size at most k, the Pr($X \subseteq Y$)$\geq \dfrac{1}{2^{\mathcal{O}(k \log kd)}}$

## Randomized Independent Set Covering Lemma (ISCL)

There is an algorithm that given a d-degenerate graph G and an integer k, outputs a family $\mathscr{F}$(G,k) such that:

- $\mathscr{F}$(G,k) is an ISCF for (G,k) with probability at least 1- 1/n,
- $|\mathscr{F}(G,k)| \leq 2^{O(k \log kd)} \log n$
- Running time of the algorithm is $O(|\mathscr{F}(G,k)| \cdot (n+m))$.

# Deterministic Independent Set Covering Lemma

$(n,l,q)$-perfect hash family, $(q \geq l)$

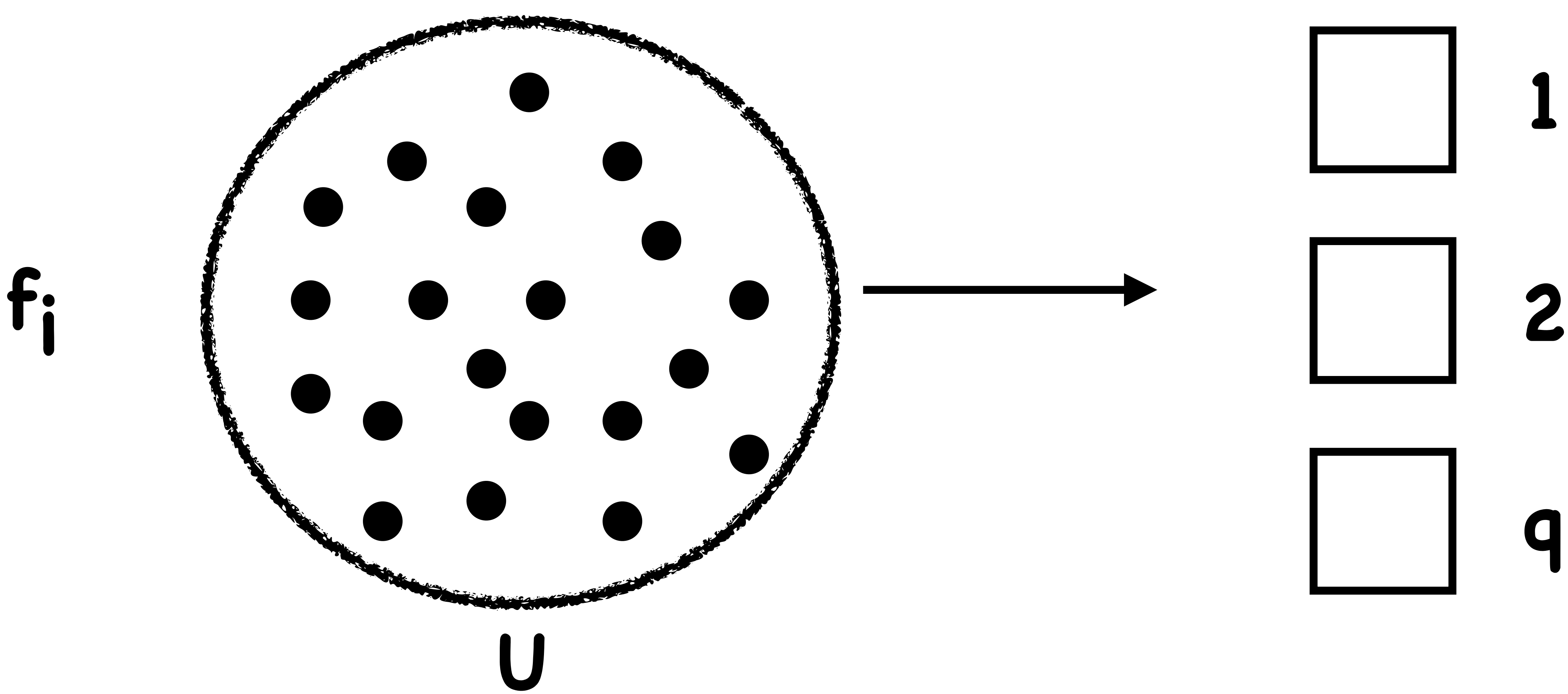
$|U| = n$

Family of functions $\{f_1,\dots,f_t\}$

$f_i : U \to [q]$

For each $S \subseteq U$, $|S| \leq l$,

there exists some $f_i$ such that $f_i$ is injective on $S$



# Deterministic Independent Set Covering Lemma

$(n,l,q)$-perfect hash family, $(q \geq l)$

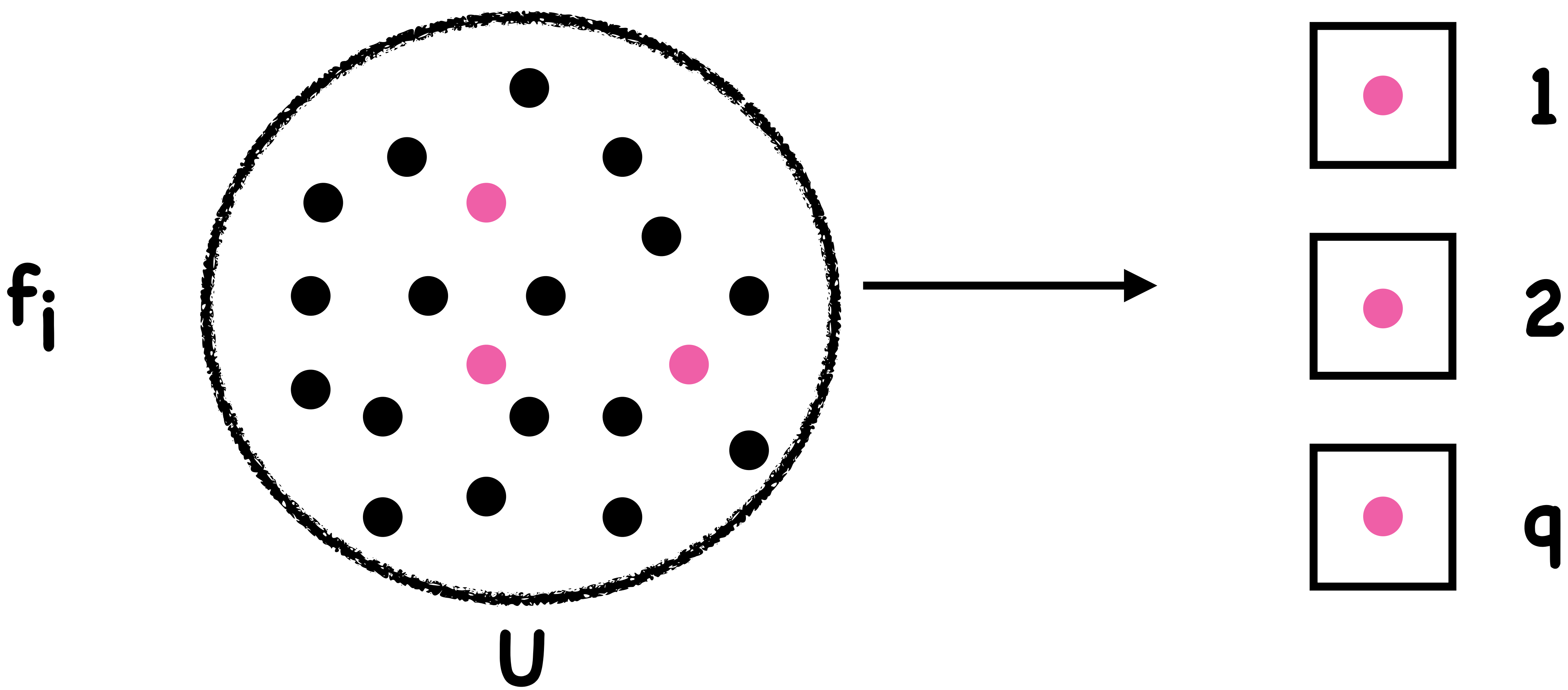
$|U| = n$

Family of functions $\{f_1,\ldots,f_t\}$

$f_i : U \to [q]$

For each $S \subseteq U$, $|S| \leq l$,

there exists some $f_i$ such that $f_i$ is injective on $S$



# Deterministic Independent Set Covering Lemma

## (n,l,q)-perfect hash family

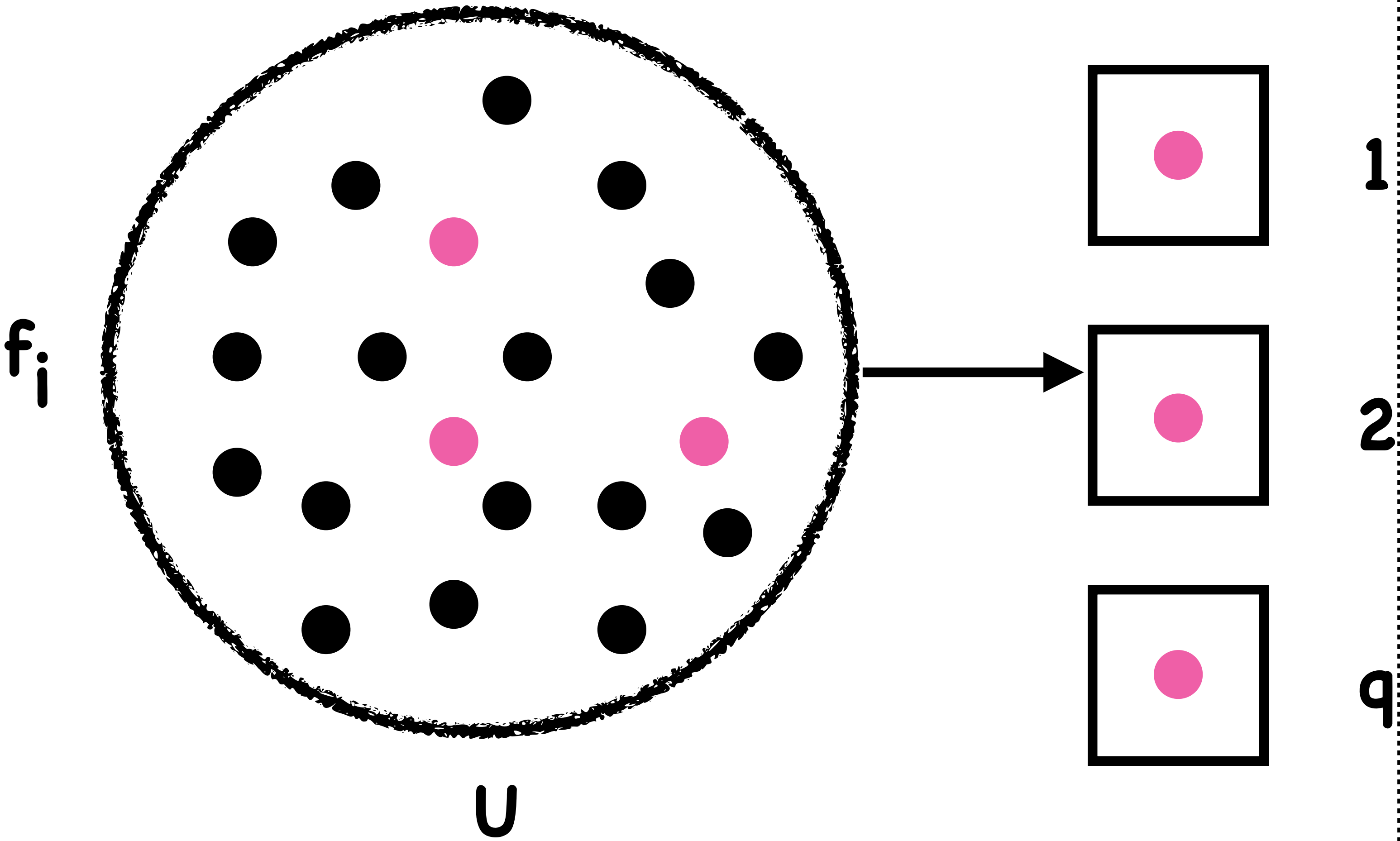
$|U| = n$

Family of functions $\{f_1, \ldots, f_t\}$

$f_i : U \to [q]$

For each $S \subseteq U$, $|S| \leq l$,

there exists some $f_i$ such that $f_i$ is injective on $S$




Fredman, Komlos, Szemeredi [J. ACM '84]

For any n,l, a $(n, l, l^{O(1)})$-perfect hash family of size $l^{O(1)} \log n$ can be computed in time $l^{O(1)} n \log n$.

Compute for $l = k+kd$

## Deterministic Independent Set Covering Lemma (ISCL)

There is an algorithm that given a d-degenerate graph G and an integer k, runs in time $2^{O(k \log kd)} (n+m) \log n$, and outputs an ISCF for (G,k) of size $2^{O(k \log kd)} \log n$.

# Applications: Design of Fixed-Parameter Tractable Algorithms

# Vertex Deletion Problems

**Input**: A graph G , an integer k
**Question**: Does there exist a set of at most k vertices, say S, such that G-S has a property Π?

- Feedback Vertex Set (FVS): Π is a forest.
- Odd Cycle Transversal (OCT): Π is a bipartite graph.
- Planar Vertex Deletion (PVD): Π is a planar graph.
- s-t Separator: Π is no path from s to t.
- ...

# Conflict-free Vertex Deletion Problems

**Input**: A graph G , an integer k

**Question**: Does there exist a set of at most k vertices, say S, such that G-S has a property Π and S is conflict-free (independent set)?

- Conflict-free Feedback Vertex Set (FVS)
- Conflict-free Odd Cycle Transversal (OCT)
- Conflict-free Planar Vertex Deletion (PVD)
- Conflict-free s-t Separator
- ...

"Reusing" algorithms of vertex deletion problems to design algorithms for Conflict-free Vertex Deletion Problems

# FPT Algorithms Conflict-free Vertex Deletion Problems on d-degenerate graphs (using ISCL)

## Deterministic Independent Set Covering Lemma (ISCL)

There is an algorithm that given a d-degenerate graph G and an integer k, runs in time $2^{O(k \log d)} (n+m) \log n$, and outputs an ISCF for (G,k) of size $2^{O(k \log d)} \log n$.

Conflict-free s-t Separator on d-degenerate graphs

# Conflict-free s-t Separator on d-degenerate graphs

**Input**: A graph $G$, an integer $k$, vertices $s$ and $t$

**Question**: Does there exist a set $S$, such that $|S| \leq k$, $S$ is an independent set in $G$ and $G-S$ has no path from $s$ to $t$.

- Compute ISCF for $(G,k)$, say $\mathscr{F} = \{Y_1, \ldots, Y_t\}$, where $t = 2^{O(k \log kd)} \log n$ (from ISCL). Time Taken: $2^{O(k \log kd)} (n+m) \log n$

## Annotated s-t Separator

**Input**: A graph $G$, an integer $k$, vertices $s$ and $t$, $Y \subseteq V(G)$

**Question**: Does there exist a set $S$, such that $|S| \leq k$, $S \subseteq Y$ and $G-S$ has no path from $s$ to $t$.

$(G,k,s,t)$ ⟶ is a YES instance

if and only if

$(G,k,s,t,Y_1)$ $(G,k,s,t,Y_2)$ $(G,k,s,t,Y_t)$ ⟶ one of them is a YES instance

# Annotated s-t Separator

**Input**: A graph $G$, an integer $k$, vertices $s$ and $t$, $Y \subseteq V(G)$

**Question**: Does there exist a set $S$, such that $|S| \leq k$, $S \subseteq Y$ and $G-S$ has no path from $s$ to $t$.

Assign weights to vertices, $w(v) = 1$ if $v \in Y$, otherwise $w(v) = k+1$.

## Weighted s-t Separator

**Input**: A graph $G$, an integer $k$, vertices $s$ and $t$, $w : V(G) \rightarrow \mathbb{N}$

**Question**: Does there exist a set $S$, such that $|S| \leq k$, $w(S) \leq k$ and $G-S$ has no path from $s$ to $t$.

Annotated s-t Separator can be solved in $O(k \cdot (n+m))$ time.

---

Conflict-free s-t Separator on d-degenerate graphs can be solved in $2^{O(k \log kd)} (n+m) \log n$ time.

# FPT Algorithms Conflict-free Vertex Deletion Problems on general graphs

# Conflict-free Feedback Vertex Set on *general graphs*

Approximate Feedback Vertex set,
$|X| \leq c\,k$

Forest,
$R = V(G) \setminus X$

Compute ISCF for (G[R],k), say $\mathcal{F}_1$

$|\mathcal{F}_1| = 2^{O(k \log k)} \log n$

(using ISCL)

Compute ISCF for (G[X],k), say $\mathcal{F}_2$

$|\mathcal{F}_2| = 2^{O(k)}$

(Using Brute force)

$\mathcal{F} = \{Y \cup Z : Y \in \mathcal{F}_1, Z \in \mathcal{F}_2\}$ is an ISCF for G.

$|\mathcal{F}| = 2^{k^{O(1)}} \log n$

storytime

# Open Problem at Dagstuhl Seminar

## Structure Theory and FPT Algorithms for Graphs, Digraphs and Hypergraphs

## 2007

**Almost 2-coloring**

**Henning Fernau**
**U. Trier**
**fernau@uni-trier.de**

Is the following problem fixed-parameter tractable? Given a graph $G$ and a parameter $k$, determine whether $G$ has a vertex 3-coloring such that one color class has at most $k$ vertices. In other words, the goal is to remove an independent set of $k$ vertices such that the remaining graph is bipartite.

Open Problem at Dagstuhl Seminar

Structure Theory and FPT Algorithms for Graphs, Digraphs and Hypergraphs

2007

**Almost 2-coloring**

**Henning Fernau**
**U. Trier**
**fernau@uni-trier.de**

Is the following problem fixed-parameter tractable? Given a graph $G$ and a parameter $k$, determine whether $G$ has a vertex 3-coloring such that one color class has at most $k$ vertices. In other words, the goal is to remove an independent set of $k$ vertices such that the remaining graph is bipartite.

**Is Conflict-free Odd Cycle Transversal FPT?**

- Reed, Smith, Vetta : Finding odd cycle transversals. [Operations Research Letters] (2004)

**Is Conflict-free s-t Separator FPT?**

# Is Conflict-free s-t Separator FPT?

❖ Marx, O'Sullivan, Razgon : Finding small separators in linear time via treewidth reduction. [ACM Trans. Algorithms] (2013)

Yes! $2^{2^{k^{O(1)}}}(n+m)$

## Open Problems from Marx et al.

1. Is it possible to improve the dependence of k to $2^{k^{O(1)}}$ ?
2. Is Conflict-free Multicut FPT?

❖ Lokshtanov, Panolan, Saurabh, S., Zehavi: Covering small independent sets and separators with applications to parameterized algorithms [SODA] (2018)

1. Yes! $2^{k^{O(1)}}(n+m)\log n$
2. Yes! $2^{O(k^3)}n^3(n+m)$

# Overview of Marx et el [TALG 2013] approach

**Is Conflict-free s-t Separator FPT?** (G,k,s,t)

1. Treewidth Reduction Step
2. Dynamic Programming on bounded treewidth graph

---

**Treewidth Reduction Step**

f(k). (n+m) time

(G,k,s,t) ⟶ (G',k,s,t)

preserves all minimal s-t separators of size at most k

treewidth(G') = $2^{k^{O(1)}}$

G' is an "induced subgraph" of G

---

**Dynamic Programming on bounded treewidth graph (G')**

Time taken: exponential in treewidth ⇒ $2^{2^{k^{O(1)}}}$ (n+m)

# Our approach [SODA 2018]

1. Treewidth Reduction Step
2. ~~Dynamic Programming on bounded treewidth graph~~ ISCL

---

Treewidth Reduction Step $f(k).(n+m)$ time

✓ $(G,k,s,t)$ ⟶ $(G',k,s,t)$

preserves all minimal s-t separators of size at most k

$\text{treewidth}(G') = 2^{k^{O(1)}}$

G' is an "induced subgraph" of G

---

✗ ~~Dynamic Programming on bounded treewidth graph (G')~~ ISCL on (G',k)

$\text{degeneracy}(G') \leq \text{treewidth}(G') = 2^{k^{O(1)}}$

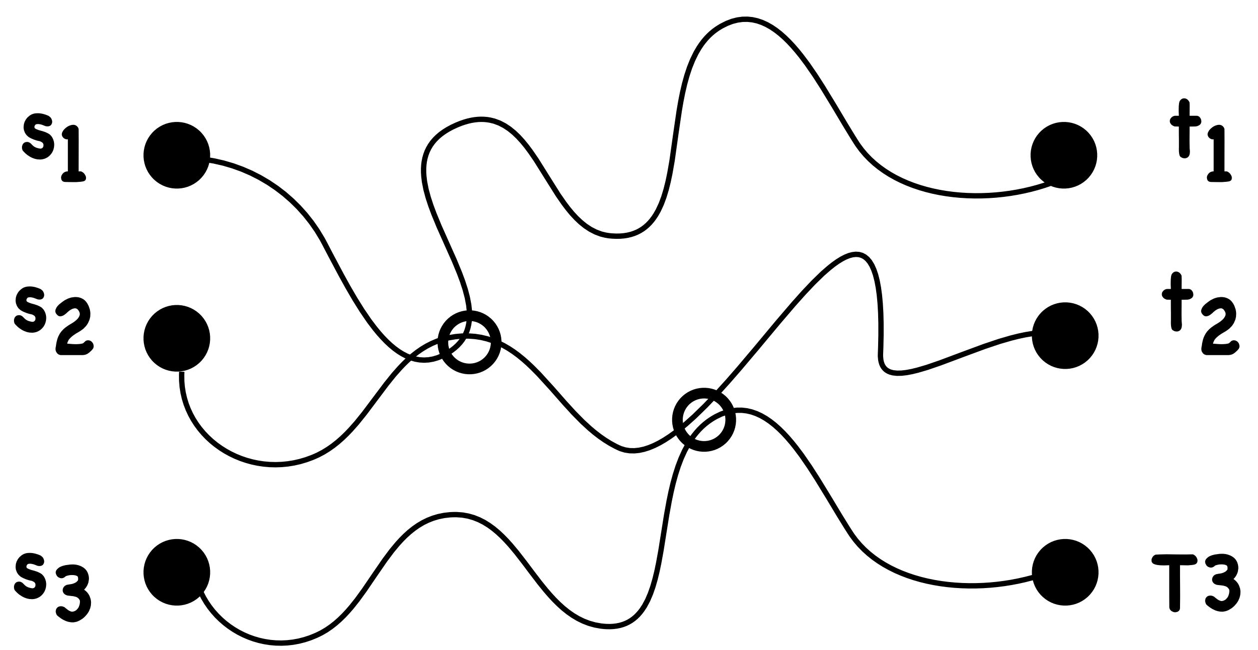
Conflict-free s-t Separator on d-degenerate graphs can be solved in $2^{O(k \log kd)} (n+m) \log n$ time.

★ Conflict-free s-t Separator on *general* graphs can be solved in $2^{k^{O(1)}} (n+m) \log n$ time.

# Conflict-free Multicut

**Input**: A graph $G$, an integer $k$, terminal pairs $T=\{(s_1,t_1), \dots , (s_p,t_p)\}$

**Question**: Does there exist a set $S$, such that $|S| \leq k$, $S$ is an independent set in $G$ and, for all $i \in \{1,\dots,p\}$, there is no path from $s_i$ to $t_i$ in $G-s$.



# Tool 2: Degeneracy Reduction Preserving Minimal Multicuts

There exists a polynomial time algorithm that given a graph $G$, a set of terminal pairs $T=\{(s_1,t_1), \ldots , (s_p,t_p)\}$ and an integer $k$, returns an induced subgraph $G'$ of $G$ and $T' \subseteq T$ such that:

- Every minimal multicut of $T$ in $G$ of size at most $k$ is a minimal multicut of $T'$ in $G'$,
- Every minimal multicut of $T'$ in $G'$ of size at most $k$ is a minimal multicut of $T$ in $G$.
- Degeneracy of $G'$ is $2^{O(k)}$.

$$(G,T,k) \longleftrightarrow (G',T',k)$$

# Concluding Remarks

## Extensions:

ISCL for nowhere dense graphs

## Barriers:

- ISCL for general graphs
- Induced Matching Covering on 1-degenerate graphs
- Acyclic Subgraphs Covering on 2-degenerate graphs
- r-scattered sets covering on 1-degenerate graphs

## Open:

- ISCL beyond nowhere dense graphs?
- Covering other families

# Thank you!