# Complexity of the identifying code problem in restricted graph classes

Florent Foucaud

Universitat Politècnica de Catalunya, Barcelona (Spain)

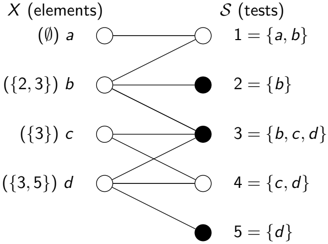
Rouen, July 11th, 2013

IWOCA

# The test cover problem

**Definition** - TEST COVER (mentioned in Garey, Johnson, 1979)

**INPUT:** set system (i.e. hypergraph) $(X, \mathcal{S})$
**TASK:** find the minimum subset $\mathcal{T} \subseteq \mathcal{S}$ such that each element $x \in X$ belongs to a different set of sets in $\mathcal{T}$.

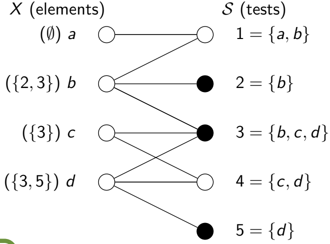



example: $\mathcal{T} = \{2, 3, 5\}$

# The test cover problem

**Definition** - TEST COVER (mentioned in Garey, Johnson, 1979)

**INPUT:** set system (i.e. hypergraph) $(X, \mathcal{S})$
**TASK:** find the minimum subset $\mathcal{T} \subseteq \mathcal{S}$ such that each element $x \in X$ belongs to a different set of sets in $\mathcal{T}$.




example: $\mathcal{T} = \{2, 3, 5\}$

**Remark**

Equivalently: for any pair $x, y$ of elements of $X$, there is a set in $\mathcal{T}$ that contains **exactly** one of $x, y$, i.e. the symmetric difference of the sets of tests covering $x, y$ is **nonempty**.

# General bounds

**Theorem** (Folklore)

Given a set system $(X, \mathcal{S})$, a solution to TEST COVER has size at least $\log_2(|X|)$.

**Proof:** Must assign to each element of $X$, a distinct subset of $\mathcal{T}$.
Hence $|X| \leq 2^{|\mathcal{T}|}$. □

# General bounds

**Theorem** (Folklore)

Given a set system $(X, \mathcal{S})$, a solution to TEST COVER has size at least $\log_2(|X|)$.

**Proof:** Must assign to each element of $X$, a distinct subset of $\mathcal{T}$. Hence $|X| \leq 2^{|\mathcal{T}|}$. □

**Theorem** (Bondy's theorem, 1972)

Given a set system $(X, \mathcal{S})$, a minimal solution to TEST COVER has size at most $|X| - 1$.

**Proof:** nice and short graph-theoretic argument. □

# Complexity results

**Theorem** (Garey, Johnson, 1979)

TEST COVER is NP-complete.

**Theorem** (Charon, Cohen, Hudry, Lobstein, 2008)

TEST COVER is NP-complete, even for set systems with a planar incidence graph.

# Complexity results

**Theorem** (Garey, Johnson, 1979)

TEST COVER is NP-complete.

**Theorem** (Charon, Cohen, Hudry, Lobstein, 2008)

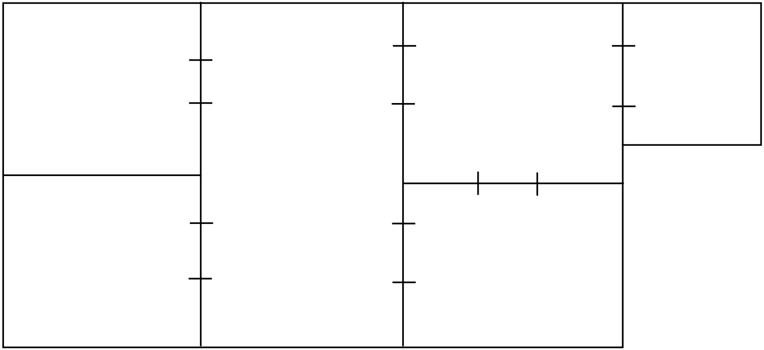
TEST COVER is NP-complete, even for set systems with a planar incidence graph.

**Theorem** (De Bontridder, Haldorsson, Haldorsson, Hurkens, Lenstra, Ravi, Stougie, 2003)

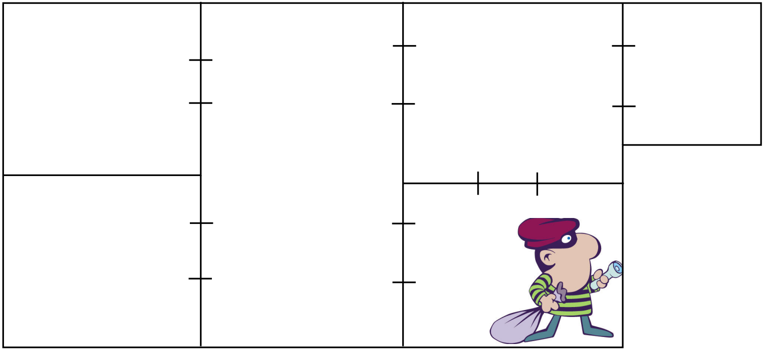
MIN TEST COVER is $O(\log(|X|))$-approximable, but NP-hard to approximate within $o(log(|X|))$.

**Proof:** Reductions from and to MIN SET COVER. □

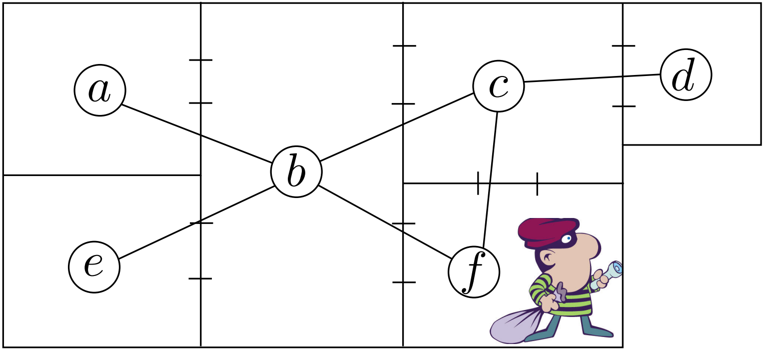
# A special case: identifying the rooms of a building



# A special case: identifying the rooms of a building

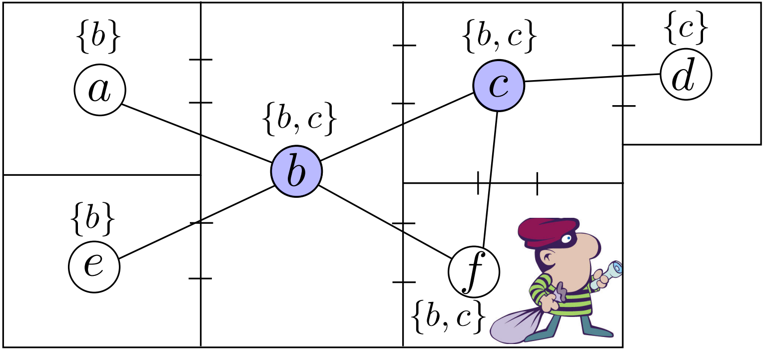


# A special case: identifying the rooms of a building



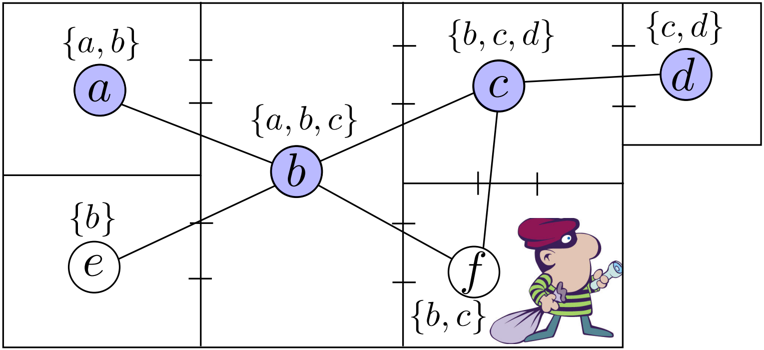

Graph $G = (V, E)$. $V$: vertices (rooms), $E \subseteq V \times V$: edges (doors)

# A special case: identifying the rooms of a building



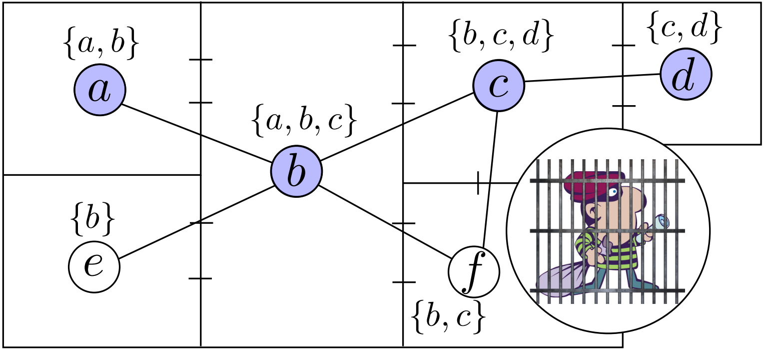

Graph $G = (V, E)$. $V$: vertices (rooms), $E \subseteq V \times V$: edges (doors)

# A special case: identifying the rooms of a building




Graph $G = (V, E)$. $V$: vertices (rooms), $E \subseteq V \times V$: edges (doors)

# A special case: identifying the rooms of a building




Graph $G = (V, E)$. $V$: vertices (rooms), $E \subseteq V \times V$: edges (doors)

# Identifying codes, a special case of test covers

$G$: undirected graph
$N[u]$: set of vertices $v$ s.t. $d(u,v) \leq 1$

**Definition** - Identifying code (Karpovsky, Chakrabarty, Levitin, 1998)

Subset $C$ of $V(G)$ such that:

- $C$ is a **dominating set** in $G$: $\forall u \in V(G)$, $N[u] \cap C \neq \emptyset$, and
- $C$ is a **separating code** in $G$: $\forall u \neq v$ of $V(G)$, $N[u] \cap C \neq N[v] \cap C$

# Identifying codes, a special case of test covers

$G$: undirected graph
$N[u]$: set of vertices $v$ s.t. $d(u, v) \leq 1$

**Definition** - Identifying code (Karpovsky, Chakrabarty, Levitin, 1998)

Subset $C$ of $V(G)$ such that:

- $C$ is a **dominating set** in $G$: $\forall u \in V(G)$, $N[u] \cap C \neq \emptyset$, and
- $C$ is a **separating code** in $G$: $\forall u \neq v$ of $V(G)$,
  $N[u] \cap C \neq N[v] \cap C$
  Equivalently: $(N[u] \Delta N[v]) \cap C \neq \emptyset$
  (i.e. covering symmetric differences)

# Identifying codes, a special case of test covers

$G$: undirected graph
$N[u]$: set of vertices $v$ s.t. $d(u, v) \leq 1$

**Definition** - Identifying code (Karpovsky, Chakrabarty, Levitin, 1998)

Subset $C$ of $V(G)$ such that:

- $C$ is a **dominating set** in $G$: $\forall u \in V(G)$, $N[u] \cap C \neq \emptyset$, and
- $C$ is a **separating code** in $G$: $\forall u \neq v$ of $V(G)$,
  $N[u] \cap C \neq N[v] \cap C$
  Equivalently: $(N[u] \Delta N[v]) \cap C \neq \emptyset$
  (i.e. covering symmetric differences)

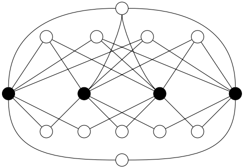
# Examples

**Definition** - Identifying code (Karpovsky, Chakrabarty, Levitin, 1998)

Subset $C$ of $V(G)$ such that:

- $C$ is a **dominating set** in $G$: $\forall u \in V(G)$, $N[u] \cap C \neq \emptyset$, and
- $C$ is a **separating code** in $G$: $\forall u \neq v$ of $V(G)$, $N[u] \cap C \neq N[v] \cap C$
  Equivalently: $(N[u] \Delta N[v]) \cap C \neq \emptyset$
  (i.e. covering symmetric differences)

$|C| = \log_2(n+1)$



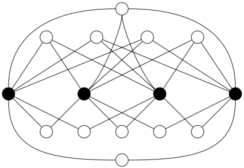
# Examples

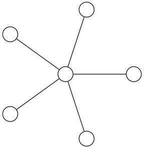
**Definition** - Identifying code (Karpovsky, Chakrabarty, Levitin, 1998)

Subset $C$ of $V(G)$ such that:

- $C$ is a **dominating set** in $G$: $\forall u \in V(G)$, $N[u] \cap C \neq \emptyset$, and
- $C$ is a **separating code** in $G$: $\forall u \neq v$ of $V(G)$,
  $N[u] \cap C \neq N[v] \cap C$
  Equivalently: $(N[u] \Delta N[v]) \cap C \neq \emptyset$
  (i.e. covering symmetric differences)

$|C| = \log_2(n+1)$





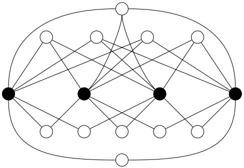
# Examples

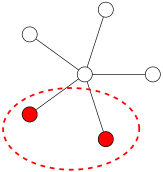
**Definition** - Identifying code (Karpovsky, Chakrabarty, Levitin, 1998)

Subset $C$ of $V(G)$ such that:

- $C$ is a **dominating set** in $G$: $\forall u \in V(G)$, $N[u] \cap C \neq \emptyset$, and
- $C$ is a **separating code** in $G$: $\forall u \neq v$ of $V(G)$, $N[u] \cap C \neq N[v] \cap C$
  Equivalently: $(N[u] \Delta N[v]) \cap C \neq \emptyset$
  (i.e. covering symmetric differences)

$|C| = \log_2(n+1)$





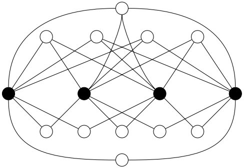
# Examples

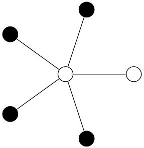
**Definition** - Identifying code (Karpovsky, Chakrabarty, Levitin, 1998)

Subset $C$ of $V(G)$ such that:

- $C$ is a **dominating set** in $G$: $\forall u \in V(G)$, $N[u] \cap C \neq \emptyset$, and
- $C$ is a **separating code** in $G$: $\forall u \neq v$ of $V(G)$,
  $N[u] \cap C \neq N[v] \cap C$
  Equivalently: $(N[u] \Delta N[v]) \cap C \neq \emptyset$
  (i.e. covering symmetric differences)

$|C| = \log_2(n+1)$





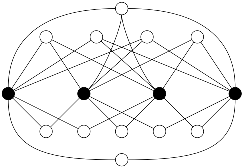
# Examples

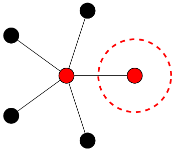
**Definition** - Identifying code (Karpovsky, Chakrabarty, Levitin, 1998)

Subset $C$ of $V(G)$ such that:

- $C$ is a **dominating set** in $G$: $\forall u \in V(G)$, $N[u] \cap C \neq \emptyset$, and
- $C$ is a **separating code** in $G$: $\forall u \neq v$ of $V(G)$,
  $N[u] \cap C \neq N[v] \cap C$
  Equivalently: $(N[u] \Delta N[v]) \cap C \neq \emptyset$
  (i.e. covering symmetric differences)

$|C| = \log_2(n+1)$





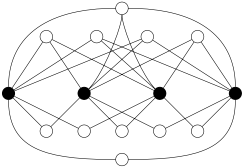
# Examples

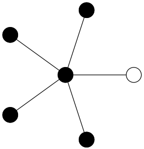
**Definition** - Identifying code (Karpovsky, Chakrabarty, Levitin, 1998)

Subset $C$ of $V(G)$ such that:

- $C$ is a **dominating set** in $G$: $\forall u \in V(G)$, $N[u] \cap C \neq \emptyset$, and
- $C$ is a **separating code** in $G$: $\forall u \neq v$ of $V(G)$,
  $N[u] \cap C \neq N[v] \cap C$
  Equivalently: $(N[u] \Delta N[v]) \cap C \neq \emptyset$
  (i.e. covering symmetric differences)

$|C| = \log_2(n+1)$





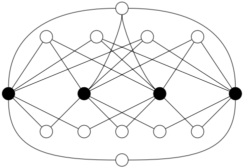
# Examples

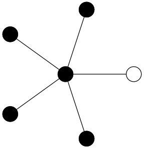
**Definition** - Identifying code (Karpovsky, Chakrabarty, Levitin, 1998)

Subset $C$ of $V(G)$ such that:

- $C$ is a **dominating set** in $G$: $\forall u \in V(G)$, $N[u] \cap C \neq \emptyset$, and
- $C$ is a **separating code** in $G$: $\forall u \neq v$ of $V(G)$,
  $N[u] \cap C \neq N[v] \cap C$
  Equivalently: $(N[u] \Delta N[v]) \cap C \neq \emptyset$
  (i.e. covering symmetric differences)

$|C| = \log_2(n+1)$





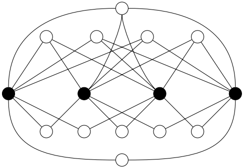
# Examples

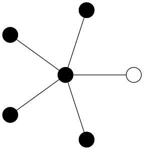
**Definition** - Identifying code (Karpovsky, Chakrabarty, Levitin, 1998)

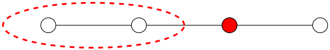
Subset $C$ of $V(G)$ such that:

- $C$ is a **dominating set** in $G$: $\forall u \in V(G)$, $N[u] \cap C \neq \emptyset$, and
- $C$ is a **separating code** in $G$: $\forall u \neq v$ of $V(G)$,
  $N[u] \cap C \neq N[v] \cap C$
  Equivalently: $(N[u] \Delta N[v]) \cap C \neq \emptyset$
  (i.e. covering symmetric differences)

$|C| = \log_2(n+1)$







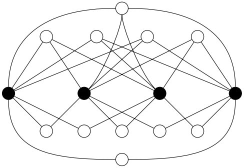
# Examples

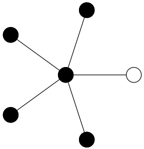
**Definition** - Identifying code (Karpovsky, Chakrabarty, Levitin, 1998)

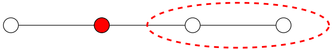
Subset $C$ of $V(G)$ such that:

- $C$ is a **dominating set** in $G$: $\forall u \in V(G)$, $N[u] \cap C \neq \emptyset$, and
- $C$ is a **separating code** in $G$: $\forall u \neq v$ of $V(G)$,
  $N[u] \cap C \neq N[v] \cap C$
  Equivalently: $(N[u] \Delta N[v]) \cap C \neq \emptyset$
  (i.e. covering symmetric differences)

$|C| = \log_2(n+1)$







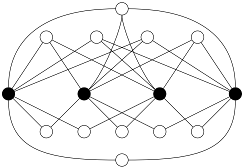
# Examples

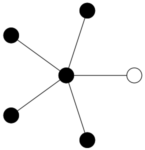
**Definition** - Identifying code (Karpovsky, Chakrabarty, Levitin, 1998)

Subset $C$ of $V(G)$ such that:

- $C$ is a **dominating set** in $G$: $\forall u \in V(G)$, $N[u] \cap C \neq \emptyset$, and
- $C$ is a **separating code** in $G$: $\forall u \neq v$ of $V(G)$,
  $N[u] \cap C \neq N[v] \cap C$
  Equivalently: $(N[u] \Delta N[v]) \cap C \neq \emptyset$
  (i.e. covering symmetric differences)

$|C| = \log_2(n+1)$





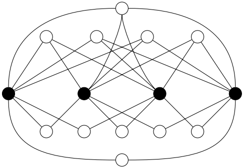
# Examples

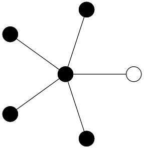
**Definition** - Identifying code (Karpovsky, Chakrabarty, Levitin, 1998)

Subset $C$ of $V(G)$ such that:

- $C$ is a **dominating set** in $G$: $\forall u \in V(G)$, $N[u] \cap C \neq \emptyset$, and
- $C$ is a **separating code** in $G$: $\forall u \neq v$ of $V(G)$,
  $N[u] \cap C \neq N[v] \cap C$
  Equivalently: $(N[u] \Delta N[v]) \cap C \neq \emptyset$
  (i.e. covering symmetric differences)

$|C| = \log_2(n+1)$





# Computational problems

**Definition** - MIN IDCODE

**INPUT:** graph $G$
**TASK:** find a minimum-size identifying code of $G$

**Theorem** (Cohen, Honkala, Lobstein, Zémor, 1999)

MIN IDCODE NP-hard (reduction from 3SAT).

NP-completeness also holds for planar subcubic graphs, planar bipartite unit disk graphs, line graphs, etc.

# Computational problems

**Definition** - MIN IDCODE

**INPUT:** graph $G$
**TASK:** find a minimum-size identifying code of $G$

**Theorem** (Cohen, Honkala, Lobstein, Zémor, 1999)

MIN IDCODE NP-hard (reduction from 3SAT).

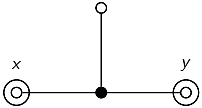
NP-completeness also holds for planar subcubic graphs, planar bipartite unit disk graphs, line graphs, etc.

**Theorem** (Berger-Wolf, Laifenfeld, Trachtenberg, 2006)

MIN IDCODE is approximable within $O(\log(n))$, but NP-hard to approximate within $o(\log(n))$ (reduction from MIN SET COVER).

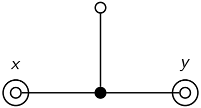
# A simple reduction from MIN VERTEX COVER

**Reduction:** subdivide each edge $xy$ of $G$ once, add pendant vertex.



# A simple reduction from MIN VERTEX COVER

**Reduction:** subdivide each edge $xy$ of $G$ once, add pendant vertex.

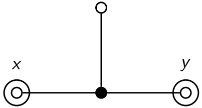



**Proposition**

If $G$ has min. degree 2, $G$ has a vertex cover of size $k$ iff $f(G)$ has an id. code of size $k + |E(G)|$.

# A simple reduction from MIN VERTEX COVER

**Reduction:** subdivide each edge $xy$ of $G$ once, add pendant vertex.




**Proposition**

If $G$ has min. degree 2, $G$ has a vertex cover of size $k$ iff $f(G)$ has an id. code of size $k + |E(G)|$.

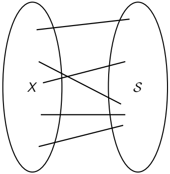
MIN VERTEX COVER hard for planar cubic graphs.

**Theorem** (F.)

MIN IDCODE is NP-hard for subcubic bipartite planar graphs.

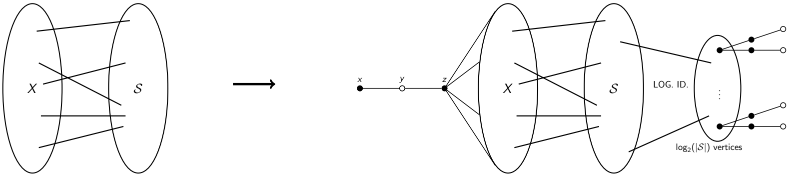
# New non-approximability reductions

Reduction: MIN TEST COVER to MIN IDCODE for bipartite graphs.



# New non-approximability reductions

Reduction: MIN TEST COVER to MIN IDCODE for bipartite graphs.

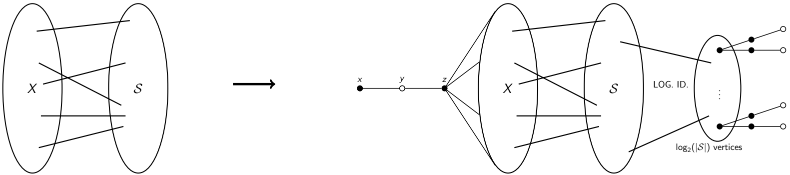



**Theorem** (F.)

- $(X, \mathcal{S})$ has a test cover of size $k$ if and only if $G(X, \mathcal{S})$ has an identifying code of size $k + 3\lceil \log_2(|\mathcal{S}| + 1) \rceil + 2$. Constructive.
- If MIN IDCODE has an $\alpha$-approximation algorithm, then MIN TEST COVER has a $4\alpha$-approximation algorithm.

# New non-approximability reductions

Reduction: MIN TEST COVER to MIN IDCODE for bipartite graphs.




**Theorem** (F.)

- $(X, \mathcal{S})$ has a test cover of size $k$ if and only if $G(X, \mathcal{S})$ has an identifying code of size $k + 3\lceil \log_2(|\mathcal{S}| + 1)\rceil + 2$. Constructive.
- If MIN IDCODE has an $\alpha$-approximation algorithm, then MIN TEST COVER has a $4\alpha$-approximation algorithm.

**Proof:** Build approximate id. code $C$ with $|C| \leq \alpha OPT_{ID}$

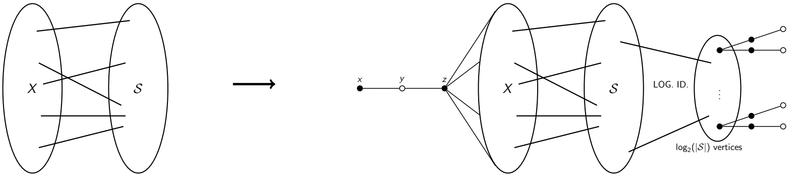
Build test cover $T$: $|T| \leq \alpha OPT_{ID} - 3\log_2(|\mathcal{S}|) - 2$

$$\leq \alpha(OPT_{TC} + 3\log_2(|\mathcal{S}|) + 2) - 3\log_2(|\mathcal{S}|) - 2$$
$$\leq \alpha OPT_{TC} + (\alpha - 1)3\log_2(|\mathcal{S}|)$$
$$\leq 4\alpha OPT_{TC}$$

□

# New non-approximability reductions

Reduction: MIN TEST COVER to MIN IDCODE for bipartite graphs.




**Theorem** (F.)

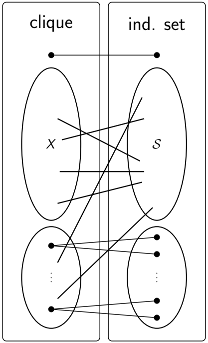
- $(X, \mathcal{S})$ has a test cover of size $k$ if and only if $G(X, \mathcal{S})$ has an identifying code of size $k + 3\lceil \log_2(|\mathcal{S}| + 1) \rceil + 2$. Constructive.
- If MIN IDCODE has an $\alpha$-approximation algorithm, then MIN TEST COVER has a $4\alpha$-approximation algorithm.

**Corollary**

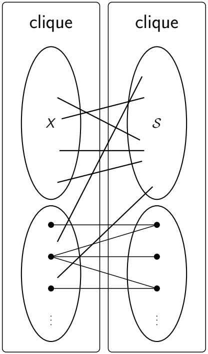
It is NP-hard to approximate MIN IDCODE within $o(\log(n))$, even for bipartite graphs.

# New non-approximability reductions

Similar reductions for split graphs and co-bipartite graphs.




split graphs




co-bipartite graphs

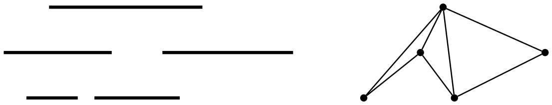
**Theorem** (F.)

It is NP-hard to approximate MIN IDCODE within $o(\log(n))$, even for split graphs and even for co-bipartite graphs.

# Interval graphs

**Definition** - Interval graph

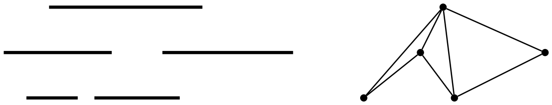
Intersection graph of intervals of the real line.



# Interval graphs

**Definition** - Interval graph

Intersection graph of intervals of the real line.



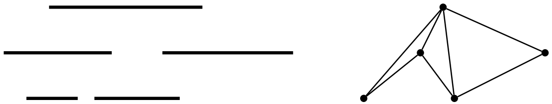
**Theorem** (F., Mertzios, Valicov)

MIN IDCODE is NP-hard for interval graphs. Reduction from 3-DIMENSIONAL MATCHING.

# Interval graphs

**Definition** - Interval graph

Intersection graph of intervals of the real line.



**Theorem** (F., Mertzios, Valicov)

MIN IDCODE is NP-hard for interval graphs. Reduction from 3-DIMENSIONAL MATCHING.

**Main idea**: an interval can separate pairs of intervals lying **far away** from each other (without affecting what lies in between).

# MIN IDCODE for unit interval graphs

**Definition** - Unit interval graph

Intersection graph of intervals of the real line all having unit length. Equivalent to *proper* interval graphs (no interval contains another).

# MIN IDCODE for unit interval graphs

**Definition** - Unit interval graph

Intersection graph of intervals of the real line all having unit length.
Equivalent to *proper* interval graphs (no interval contains another).

**Observation**

Our reduction creates interval graphs that are far from proper/unit.

# MIN IDCODE for unit interval graphs

**Definition** - Unit interval graph

Intersection graph of intervals of the real line all having unit length. Equivalent to *proper* interval graphs (no interval contains another).

**Observation**

Our reduction creates interval graphs that are far from proper/unit.

**Question**

What is the complexity of MIN IDCODE for *unit* interval graphs?

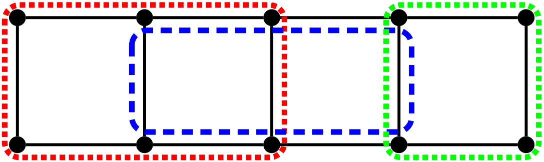
# MIN IDCODE for unit interval graphs

**Definition** - Ladder graph $L_m$

$L_m$ is the grid graph $P_2 \square P_m$.

**Definition** - Cycle cover

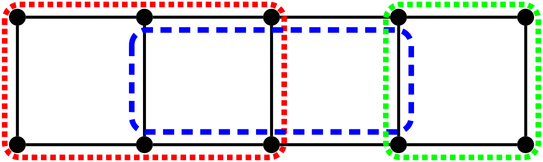
Set $\mathcal{S}$ of cycles of graph $G$ s.t. $\bigcup_{S \in \mathcal{S}} E(S) = E(G)$.



# MIN IDCODE for unit interval graphs

**Definition** - LADDER CYCLE COVER

**INPUT:** An integer $m$ and an integer $k$, and a set $\mathcal{S}$ of cycles of $L_m$.
**TASK:** Find a minimum-size cycle cover $\mathcal{S}' \subseteq \mathcal{S}$ of $L_m$.



# MIN IDCODE for unit interval graphs

**Definition** - LADDER CYCLE COVER

**INPUT:** An integer $m$ and an integer $k$, and a set $\mathcal{S}$ of cycles of $L_m$.
**TASK:** Find a minimum-size cycle cover $\mathcal{S}' \subseteq \mathcal{S}$ of $L_m$.

**Theorem** (F., Naserasr, Parreau, Valicov)

MIN IDCODE for unit interval graphs of order $n$ can be reduced to LADDER CYCLE COVER for $L_{n+1}$ and an input of $n$ cycles.

# MIN IDCODE for unit interval graphs

**Definition** - LADDER CYCLE COVER

**INPUT:** An integer $m$ and an integer $k$, and a set $\mathcal{S}$ of cycles of $L_m$.
**TASK:** Find a minimum-size cycle cover $\mathcal{S}' \subseteq \mathcal{S}$ of $L_m$.

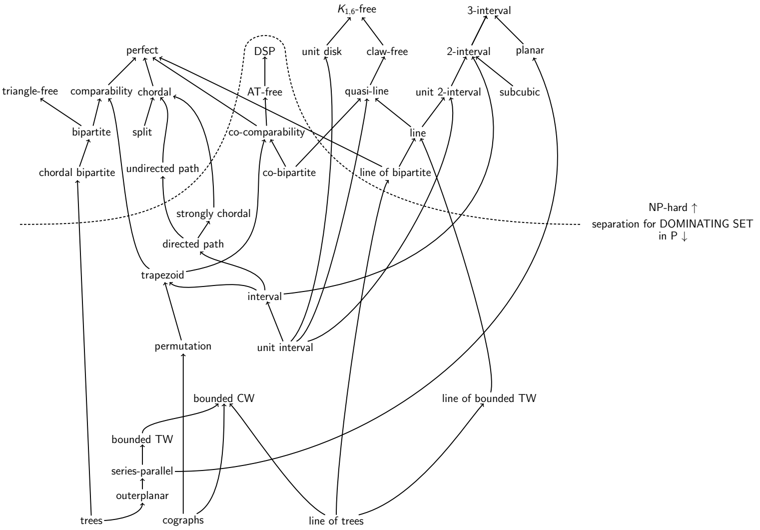
**Theorem** (F., Naserasr, Parreau, Valicov)

MIN IDCODE for unit interval graphs of order $n$ can be reduced to LADDER CYCLE COVER for $L_{n+1}$ and an input of $n$ cycles.

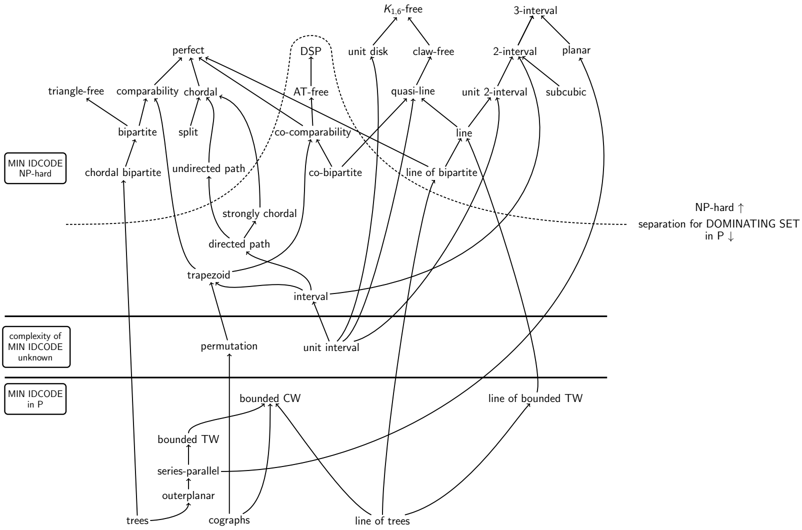
**Question**

What is the complexity of LADDER CYCLE COVER?

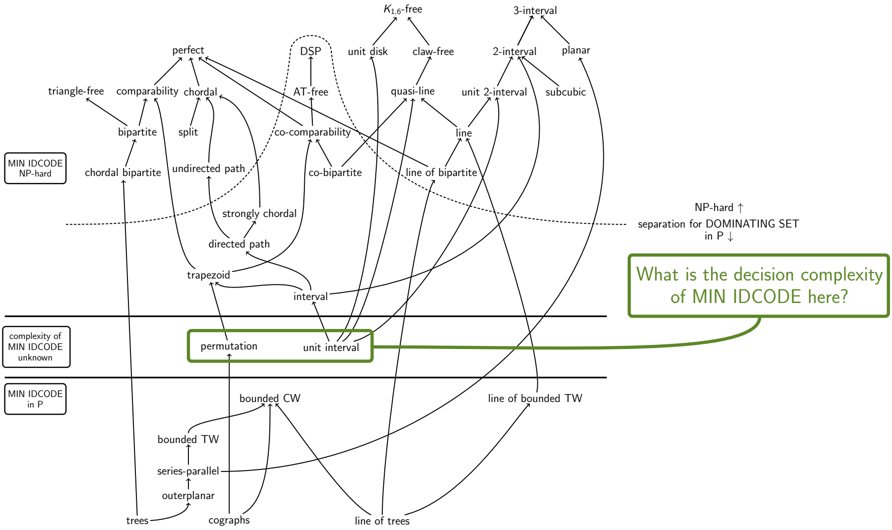
# Complexity of MIN IDCODE for various graph classes



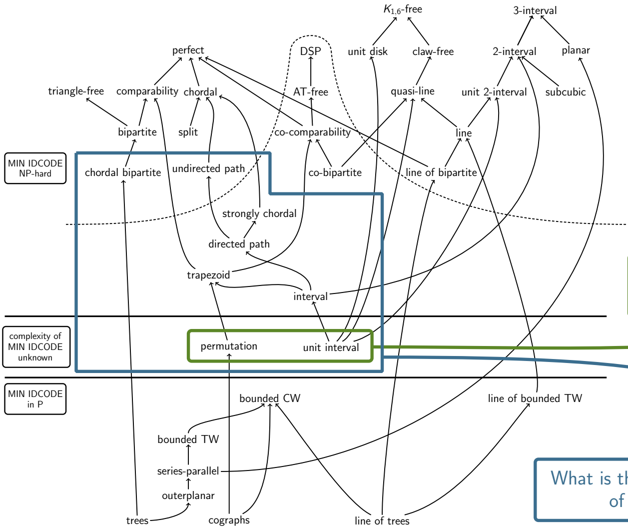
# Complexity of MIN IDCODE for various graph classes



# Complexity of MIN IDCODE for various graph classes



# Complexity of MIN IDCODE for various graph classes




What is the decision complexity of MIN IDCODE here?

What is the approximation complexity of MIN IDCODE here?