FPT algorithms to recognize well covered graphs

Rudini Sampaio
Universidade Federal do Ceará (UFC)
Fortaleza, Brazil

This is a joint work with
Sulamita Klein (UFRJ, Rio de Janeiro, Brazil)
Rafael Teixeira (UFC, Fortaleza, Brazil)

ICGT-2018, July 10, 10h30

Well covered graphs

- A graph G is **well covered** if all minimal vertex covers have the same size (are minimum)
- C is **vertex cover** if every edge has an endpoint in C
- C is **minimal** if it is not contained in any other vertex cover
- $vc(G)$ and $vc^+(G)$: minimum vs maximum minimal
- G is well covered if and only if $vc(G) = vc^+(G)$

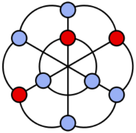


Figure: Vertex cover / Independent set

- C is a vertex cover $\iff V - C$ is independent (no edges)
- C minimal vertex cover $\iff V - C$ maximal independent
- A graph G is **well covered** if every maximal independent set has the same size $\alpha(G) = n - vc(G)$ (are maximum).

Example: Small well covered graphs

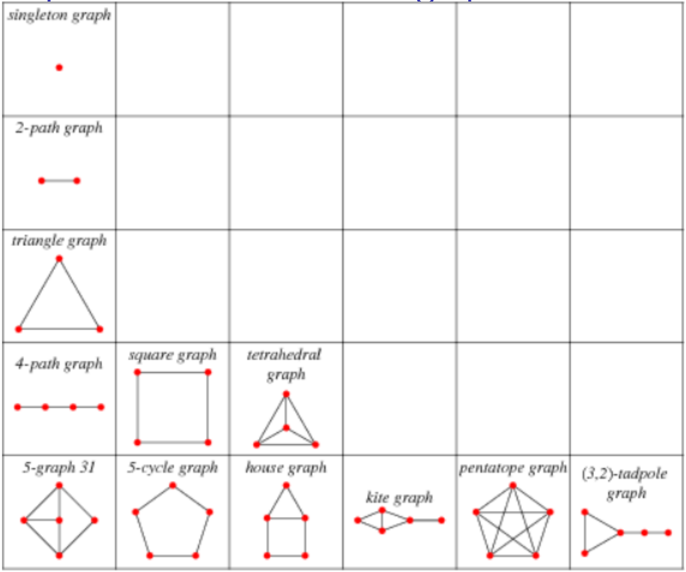


Figure: Small connected well covered graphs

Example: Rook's graph

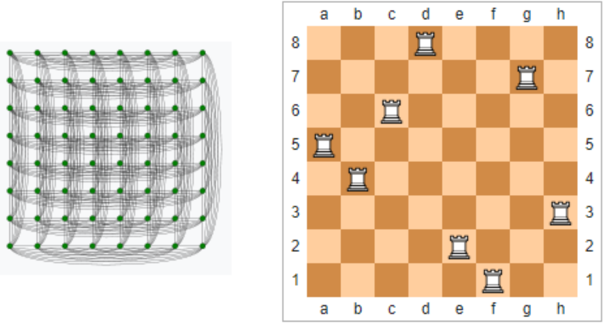


Figure: Rook's graph

A rook's graph is **well covered**: given any set of non-attacking rooks in a chesboard $n \times n$, we can place more non-attacking rooks until we have n rooks (one in each row and one in each column).

Results

Known results

- Introduced by [Plummer'70, JCTB]
- coNP-Completo in $K_{1,4}$-free graphs [Caro et al.'96, JAlg]
- Roller Coaster Conjecture [Michael and Traves'03, G&C]
- Cartesian product [Fradkin et al.'09, DM]
- Graphs with few P_4's [Klein et al.'13, G&C]
- FPT $O^*(2.83^{vc})$ and $O^*(1.54^{vc^+})$ [Boria et al.'15, DAM]
- FPT cliquewidth and neighborhood diversity $O^*(2^{nd}) = O^*(2^{2^{vc}})$ [Alves et al.'18, to appear in TCS]

New results

- FPT $O^*(2^{vc})$ parameterized by $vc(G)$
- FPT $O^*(1.4656^{vc^+})$ parameterized by $vc^+(G)$
- FPT in $\alpha(G) = n - vc(G)$ in d-degenerate graphs
- Linear algorit in $(q, q-4)$-graphs: FPT parameterized by q

Graph classes with few P_4's

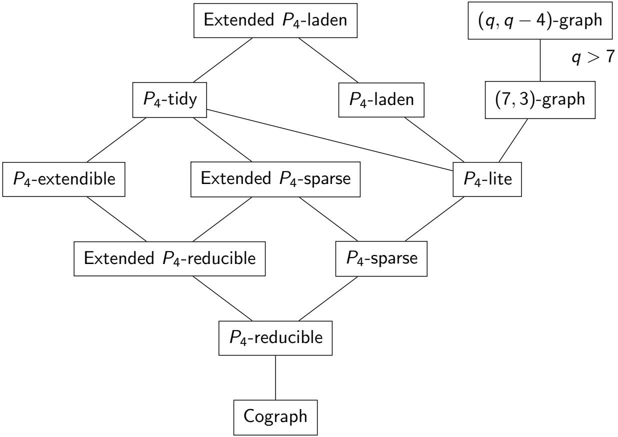


Figure: Known hierarchy of graphs with few P_4's

Graph classes with few P_4's

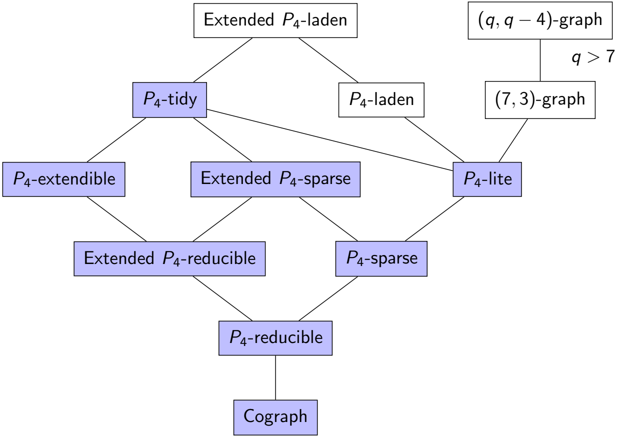
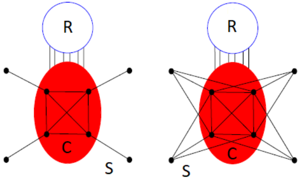


Figure: Blue graph classes: solved in [Klein et al., 2013, G&C]

Some operations: union, join and spider

- **Union** $G = G_1 \cup G_2$: $V(G) = V(G_1) \cup V(G_2)$, $E(G) = E(G_1) \cup E(G_2)$
- **Join** $G = G_1 \vee G_2$: $V(G) = V(G_1) \cup V(G_2)$, $E(G) = E(G_1) \cup E(G_2) \cup \{uv : u \in V(G_1), v \in V(G_2)\}$.



G is a **pseudo-split** (R, C, S) if $V(G) = R \cup C \cup S$ st:

- C induces a clique and S induces an independent set
- All edges from R to C and no edges from R to S

G is a **spider** if it is a pseudo-split (R, C, S) st:

- $C = \{c_1, \ldots, c_k\}$ and $S = \{s_1, \ldots, s_k\}$ for some $k \geq 2$
- **Thin spider**: s_i is adjacent to c_j if and only if $i = j$
- **Thick spider**: s_i is adjacent to c_j if and only if $i \neq j$

Graphs with few P_4's

Theorem [Klein et al., 2013]: solved Union and Join

- $G_1 \cup G_2$ well covered if and only if G_1 and G_2 are well covered
- $G_1 \vee G_2$ well covered if and only if G_1 and G_2 are well covered and $\alpha(G_1) = \alpha(G_2)$.

Theorem: Pseudo-split and quasi-spiders

- Pseudo-split (R, C, S) well covered **iff** $R = \emptyset$ and every vertex of C has exactly one neighbor in S.
- Quasi spider (R, C, S) well covered **iff** $R = \emptyset$ and G is a thin spider with a vertex possibly substituted by a K_2.
- **Separable p-components** also OK.

$(q, q-4)$-graphs and Extended P_4-laden graphs

- Decomposition theorems in terms of union, join, pseudo-splits, quasi-spiders and separable p-components
- Compute $\alpha(G)$ and decide well coverdness in a DP manner

Graph classes with few P_4's

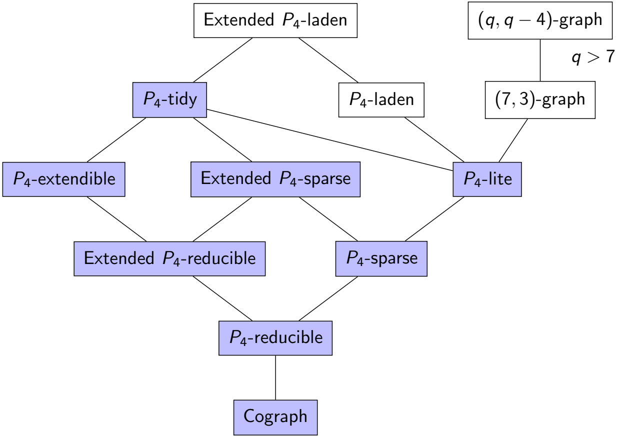


Figure: Blue graph classes: solved in [Klein et al., 2013, G&C]

FPT in $vc(G)$ - time $O^*(2^{vc})$

Technique similar to Iterative Compression

- Obtain a minimum vertex cover C in time $O^*(2^{vc})$
- For every partition of C in two sets A and B:
  - Check if $A \cup (N(B) \setminus B)$ is a minimal vertex cover
- Return the maximum minimal vertex cover
- Number of partitions: 2^{vc}

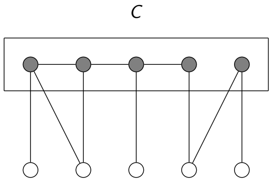


Figure: Vertex cover C

FPT in $vc(G)$ - time $O^*(2^{vc})$

Techinque similar to Iterative Compression

- Obtain a minimum vertex cover C in time $O^*(2^{vc})$
- For every partition of C in two sets A and B:
  - Check if $A \cup (N(B) \setminus B)$ is a minimal vertex cover
- Return the maximum minimal vertex cover
- Number of partitions: 2^{vc}

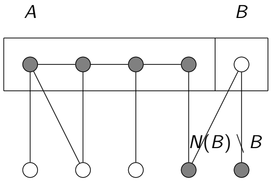


Figure: Vertex cover C

FPT in $vc(G)$ - time $O^*(2^{vc})$

Techinque similar to Iterative Compression

- Obtain a minimum vertex cover C in time $O^*(2^{vc})$
- For every partition of C in two sets A and B:
  - Check if $A \cup (N(B) \setminus B)$ is a minimal vertex cover
- Return the maximum minimal vertex cover
- Number of partitions: 2^{vc}

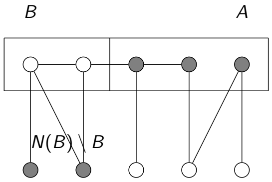


Figure: Vertex cover C

FPT in $vc(G)$ - time $O^*(2^{vc})$

Techinque similar to Iterative Compression

- Obtain a minimum vertex cover C in time $O^*(2^{vc})$
- For every partition of C in two sets A and B:
  - Check if $A \cup (N(B) \setminus B)$ is a minimal vertex cover
- Return the maximum minimal vertex cover
- Number of partitions: 2^{vc}

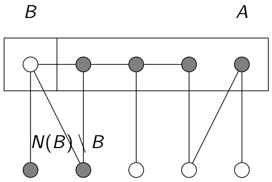


Figure: Vertex cover C

FPT in $vc^+(G)$ - time $O^*(1.4656^{vc^+})$

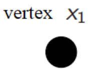


Figure: Branch in the vertices

FPT in $vc^{+}(G)$ - time $O^{*}(1.4656^{vc^{+}})$

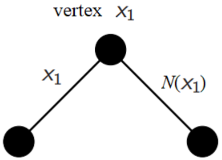


Figure: Branch in the vertices

FPT in $vc^+(G)$ - time $O^*(1.4656^{vc^+})$

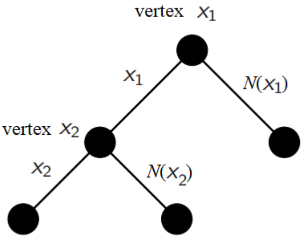


Figure: Branch in the vertices

FPT in $vc^+(G)$ - time $O^*(1.4656^{vc^+})$

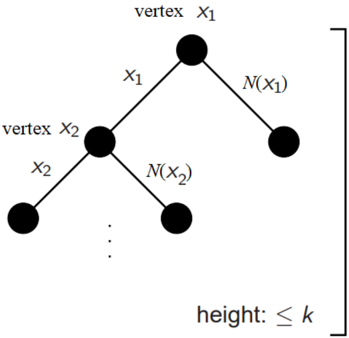


Figure: Branch in the vertices

FPT in $vc^+(G)$ - time $O^*(1.4656^{vc^+})$

- Recurrence in the number of leaves: $F(k) = F(k-1) + F(k-3)$
- $F(k) = 1.4656^k$
- Time $O(1.4656^k \cdot n^2) \quad = \quad O^*(1.4656^k)$

FPT in $vc^{+}(G)$ - time $O^{*}(1.4656^{vc^{+}})$

- Recurrence in the number of leaves: $F(k) = F(k-1) + F(k-3)$
- $F(k) = 1.4656^{k}$
- Time $O(1.4656^{k} \cdot n^{2}) \quad = \quad O^{*}(1.4656^{k})$

- Analyse each leaf: checking if it is a minimal vertex cover
- Two leaves with different heights $\Leftrightarrow$ G not well covered
- FPT time $O^{*}(1.4656^{vc^{+}})$
- Improving [Boria et al.'15, DAM] $O^{*}(1.5397^{vc^{+}})$

FPT in $\alpha = n - vc(G)$ for graph classes

- coW[2]-hard problem [Alves et al., 2018, TCS]
- 1st-order logic formulas:

$$* \ Indep(X) \ := \ \forall x, y \ (x \in X \land y \in X) \rightarrow \neg E(x, y)$$

$$* \ Maximal(X) \ := \ \forall y \ \exists x \ (y \notin X) \rightarrow (x \in X) \land E(x, y)$$

$$* \ WellCov_k \ := \ \forall_{\neq} x_1, \ldots, x_k \forall_{\neq} y_1, \ldots, y_{k-1} : \ Indep(\{x_1, \ldots, x_k\})$$

$$\implies \neg\Big(Indep(\{y_1, \ldots, y_{k-1}\}) \ \land \ Maximal(\{y_1, \ldots, y_{k-1}\})\Big)$$

$$* \ WellCov \ := \bigwedge_{2 \leq k \leq \alpha} WellCov_k$$

- Frick-Grohe Theorem: *WellCov* tem $\leq \alpha^2$ variables
- Well coveredness decision problem is FPT parameterized by $\alpha(G)$ in time $O(n^2)$ for graphs with bounded local treewidth

FPT in $\alpha = n - vc(G)$ for graph classes

Lemma

- Fixed integer d, hereditary graph class $\mathcal{C}$
- Every $G \in \mathcal{C}$ has a vertex with degree at most d
- **THEN** FPT in $\alpha(G) = n - vc(G)$ in time $O((d+1)^\alpha \cdot (m+n))$ in the class $\mathcal{C}$

Proof

- Search tree with height $\alpha(G)$
- Branch in a vertex v with $N[v] = \{u_1, \ldots, u_\ell\}$, $\ell \leq d+1$
- In child i, remove $N[u_i]$. The remaining graph is in $\mathcal{C}$
- Repeat. A leaf has height α or is empty
- Return NO, if there are two leaf nodes with different heights: two maximal independent sets with different sizes
- Return YES, otherwise
- Complexity $O((d+1)^\alpha \cdot (m+n))$

FPT in $\alpha = n - vc(G)$ for graph classes

Corollary

Well coveredness problem is FPT in $\alpha(G) = n - vc(G)$:

- Graphs with bounded genus in time $O(7^{\alpha} \cdot (m+n))$
- d-degenerate graphs in time $O((d+1)^{\alpha} \cdot (m+n))$
  - Graphs max degree Δ: $d = \Delta$
  - Outerplanar graphs ($d = 2$), Planar graphs ($d = 5$)

Proof

Ok d-degenerate graphs

- Euler for bounded genus g: $m = n + f - 2 + 2g$
- Triangulated faces (otherwise, add edges): $3f = 2m$
- Assume $n \geq 12g$ (otherwise, constant time)
- $m = n + (2/3)m - 2 + 2g \implies m = 3n + 6g - 6 \leq (3.5)n - 6$
- $\sum_{v \in V} d(v) = 2m \leq 7n - 12$
- G has a vertex of degree less than 7

Thank you !!

THANK YOU !!

Thank you !!

THANK YOU !!

Thank you !!

THANK YOU !!

The Roller Coaster Conjecture

- $i_t(G)$: number of independent sets of size $0 \leq t \leq \alpha(G)$

  [Alavi, Erdős, Malde, Schwenk, 1987] For every integer $q \geq 0$ and permutation π of $[q]$, there is a graph G with $\alpha(G) = q$ and $i_{\pi(1)}(G) < i_{\pi(2)}(G) < \ldots < i_{\pi(q)}(G)$.
- A sequence $(i_0, \ldots, i_{\alpha(G)})$ is unimodal if there is k such that $i_0 \leq i_1 \leq \ldots \leq i_k \quad \geq \quad i_{k+1} \geq \ldots \geq i_{\alpha(G)}$.

  [Brown et al., 2000] conjectured the independence sequence of any **well-covered** graph is unimodal

  [Michael, Traves, 2003] disproved this conjecture, but showed that it is increasing in the 1st half: $i_0 < i_1 < \ldots < i_{\lceil \alpha/2 \rceil}$.

  [Roller Coaster Conjecture]: 2nd half is any-ordered. That is, for any q and permutation π of $\{\lceil q/2 \rceil, \ldots, q\}$, there is a well covered graph with $\alpha(G) = q$ and $i_{\pi(\lceil q/2 \rceil)} < \ldots < i_{\pi(q)}$.

  [Matchett, 2004] proved for $\alpha \leq 11$.

  [Cuttle and Pebody, JCTB, 2017] proved the conjecture.