The parameterised complexity of subgraph counting problems

Kitty Meeks

Queen Mary, University of London

Joint work with Mark Jerrum (QMUL)

What is a counting problem?

Decision problems

Given a graph G, does G contain a Hamilton cycle?

Given a bipartite graph G, does G contain a perfect matching?

What is a counting problem?

Decision problems	**Counting problems**
Given a graph G, does G contain a Hamilton cycle?	How many Hamilton cycles are there in the graph G?
Given a bipartite graph G, does G contain a perfect matching?	How many perfect matchings are there in the bipartite graph G?

What is a parameterised counting problem?

- Introduced by Flum and Grohe (2004)
- Measure running time in terms of a parameter as well as the total input size
- Examples:

What is a parameterised counting problem?

- Introduced by Flum and Grohe (2004)
- Measure running time in terms of a parameter as well as the total input size
- Examples:
  - How many vertex-covers of size k are there in G?

What is a parameterised counting problem?

- Introduced by Flum and Grohe (2004)
- Measure running time in terms of a parameter as well as the total input size
- Examples:
  - How many vertex-covers of size k are there in G?
  - How many k-cliques are there in G?

What is a parameterised counting problem?

- Introduced by Flum and Grohe (2004)
- Measure running time in terms of a parameter as well as the total input size
- Examples:
  - How many vertex-covers of size k are there in G?
  - How many k-cliques are there in G?
  - Given a graph G of treewidth at most k, how many Hamilton cycles are there in G?

The theory of parameterised counting

Efficient algorithms: Fixed parameter tractable (FPT)
Running time $f(k) \cdot n^{O(1)}$

The theory of parameterised counting

Efficient algorithms: Fixed parameter tractable (FPT)
Running time $f(k) \cdot n^{O(1)}$

Intractable problems: #W[1]-hard
A $\#$W[1]-complete problem: **p-#**$\textsc{Clique}$.

#W[1]-completeness

- To show the problem Π' (with parameter κ') is #W[1]-hard, we give a reduction from a problem Π (with parameter κ) to Π'.

#W[1]-completeness

- To show the problem Π' (with parameter κ') is #W[1]-hard, we give a reduction from a problem Π (with parameter κ) to Π'.
- An fpt Turing reduction from (Π, κ) to (Π', κ') is an algorithm A with an oracle to Π' such that
  1. A computes Π,
  2. A is an fpt-algorithm with respect to κ, and
  3. there is a computable function $g : \mathbb{N} \to \mathbb{N}$ such that for all oracle queries "$\Pi'(y) =?$" posed by A on input x we have $\kappa'(y) \leq g(\kappa(x))$.

  In this case we write $(\Pi, \kappa) \ \leq_T^{fpt} \ (\Pi', \kappa')$.

Subgraph Counting Model

Let Φ be a family $(\phi_1, \phi_2, \ldots)$ of functions, such that ϕ_k is a mapping from labelled graphs on k-vertices to $\{0, 1\}$.

Subgraph Counting Model

Let Φ be a family $(\phi_1, \phi_2, \ldots)$ of functions, such that ϕ_k is a mapping from labelled graphs on k-vertices to $\{0, 1\}$.

p-#Induced Subgraph With Property(Φ)
Input: A graph $G = (V, E)$ and an integer k.
Parameter: k.
Question: What is the cardinality of the set
$\{(v_1, \ldots, v_k) \in V^k : \phi_k(G[v_1, \ldots, v_k]) = 1\}$?

Examples

- **p-#**$\textsc{Clique}$

Examples

- **p-#**$\textsc{Clique}$
- **p-#**$\textsc{Path}$
- **p-#**$\textsc{Cycle}$

Examples

- **p-#**Clique
- **p-#**Path
- **p-#**Cycle
- **p-#**Matching

Examples

- **p-#**CLIQUE
- **p-#**PATH
- **p-#**CYCLE
- **p-#**MATCHING
- **p-#**CONNECTED INDUCED SUBGRAPH

Examples

- **p-#**Clique
- **p-#**Path
- **p-#**Cycle
- **p-#**Matching
- **p-#**Connected Induced Subgraph
- **p-#**Clique + Independent Set

Examples

- **p-#**Clique
- **p-#**Path
- **p-#**Cycle
- **p-#**Matching
- **p-#**Connected Induced Subgraph
- **p-#**Clique + Independent Set
- **p-#**Planar Induced Subgraph

Complexity Questions

- Is the corresponding decision problem in FPT?
- Is there a fixed parameter algorithm for **p-#**Induced Subgraph With Property(Φ)?
- Can we approximate **p-#**Induced Subgraph With Property(Φ) efficiently?

Approximation Algorithms

An FPTRAS for a parameterised counting problem Π with parameter k is a randomised approximation scheme that takes an instance I of Π (with $|I| = n$), and real numbers $\epsilon > 0$ and $0 < \delta < 1$, and in time $f(k) \cdot g(n, 1/\epsilon, \log(1/\delta))$ (where f is any function, and g is a polynomial in n, $1/\epsilon$ and $\log(1/\delta)$) outputs a rational number z such that

$$\mathbb{P}[(1-\epsilon)\Pi(I) \leq z \leq (1+\epsilon)\Pi(I)] \geq 1-\delta.$$

Problems in our model

	Decision FPT?	FPTRAS?	Exact counting FPT?
p-#Clique	N	N	N
p-#Path **p-#**Cycle	Y	Y	N
p-#Matching	Y	Y	N
p-#Connected Induced Subgraph	Y	Y	N
p-#Clique + Independent Set	Y	Y	N

Flum & Grohe '04, Curticapean '13, Arvind & Raman '02, Jerrum & M. '13

The Colourful Version

Suppose the vertices of G are coloured with k colours.

We say a subset of the vertices (or a subgraph) is *colourful* if it contains exactly one vertex of each colour.

We define another problem, **p-#**$\textsc{Multicolour Induced Subgraph with Property}(\Phi)$, where we only count *colourful* labelled subgraphs satisfying Φ.

Colouring can make problems easier

- If the uncoloured version of a parameterised counting problem is in FPT, the multicolour version must also be in FPT: use inclusion-exclusion.

Colouring can make problems easier

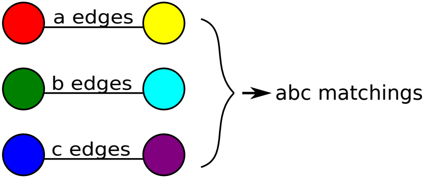
- If the uncoloured version of a parameterised counting problem is in FPT, the multicolour version must also be in FPT: use inclusion-exclusion.
- **p-#**$\textsc{Matching}$ is $\#$W[1]-complete.
- **p-#**$\textsc{Multicolour Matching}$ is in FPT:

Colouring can make problems easier

- If the uncoloured version of a parameterised counting problem is in FPT, the multicolour version must also be in FPT: use inclusion-exclusion.
- **p-#**$\textsc{Matching}$ is $\#$W[1]-complete.
- **p-#**$\textsc{Multicolour Matching}$ is in FPT:
  - There are $\frac{k!}{(\frac{k}{2})!2^{\frac{k}{2}}}$ ways to pair up the colours

Colouring can make problems easier

- If the uncoloured version of a parameterised counting problem is in FPT, the multicolour version must also be in FPT: use inclusion-exclusion.
- **p-#**$\textsc{Matching}$ is #W[1]-complete.
- **p-#**$\textsc{Multicolour Matching}$ is in FPT:
  - There are $\frac{k!}{(\frac{k}{2})!2^{\frac{k}{2}}}$ ways to pair up the colours
  - For each way of pairing up the colours, the number of matchings can easily be calculated:



Colouring can make problems harder

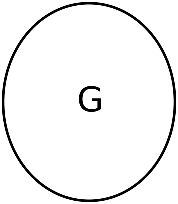
- **p**-CLIQUE + INDEPENDENT SET is in FPT:
  - By Ramsey, for sufficiently large graphs the answer is always "yes".

Colouring can make problems harder

- **p**-Clique + Independent Set is in FPT:
  - By Ramsey, for sufficiently large graphs the answer is always "yes".
- **p**-Multicolour Clique + Independent Set is W[1]-complete:
  - Reduction from **p**-Multicolour Clique.

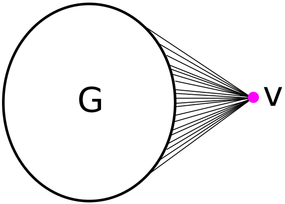
Colouring can make problems harder

- **p**-CLIQUE + INDEPENDENT SET is in FPT:
  - By Ramsey, for sufficiently large graphs the answer is always "yes".
- **p**-MULTICOLOUR CLIQUE + INDEPENDENT SET is W[1]-complete:
  - Reduction from **p**-MULTICOLOUR CLIQUE.



Colouring can make problems harder

- **p**-CLIQUE + INDEPENDENT SET is in FPT:
  - By Ramsey, for sufficiently large graphs the answer is always "yes".
- **p**-MULTICOLOUR CLIQUE + INDEPENDENT SET is W[1]-complete:
  - Reduction from **p**-MULTICOLOUR CLIQUE.



Hardness I: Properties that hold for few distinct edge densities

Theorem

Let Φ be a family $(\phi_1, \phi_2, \ldots)$ of functions $\phi_k : \{0,1\}^{\binom{k}{2}} \to \{0,1\}$ that are not identically zero, such that the function mapping $k \mapsto \phi_k$ is computable. Suppose that

$$|\{|E(H)| : |V(H)| = k \text{ and } \Phi \text{ is true for } H\}| = o(k^2).$$

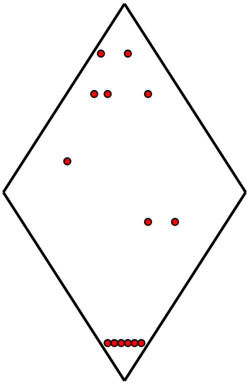
Then **p-#**$\textsc{Induced Subgraph With Property}$*($\Phi$) is #W[1]-complete.*

Hardness I: Properties that hold for few distinct edge densities

- We prove hardness of **p-#**MULTICOLOUR INDUCED SUBGRAPH WITH PROPERTY(Φ) by means of a reduction from **p-#**MULTICOLOUR CLIQUE.

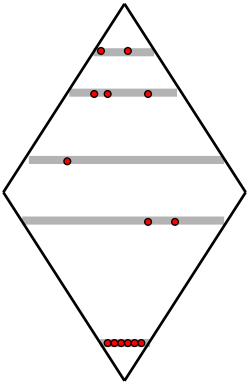
Hardness I: Properties that hold for few distinct edge densities

- We prove hardness of **p-#**MULTICOLOUR INDUCED SUBGRAPH WITH PROPERTY(Φ) by means of a reduction from **p-#**MULTICOLOUR CLIQUE.

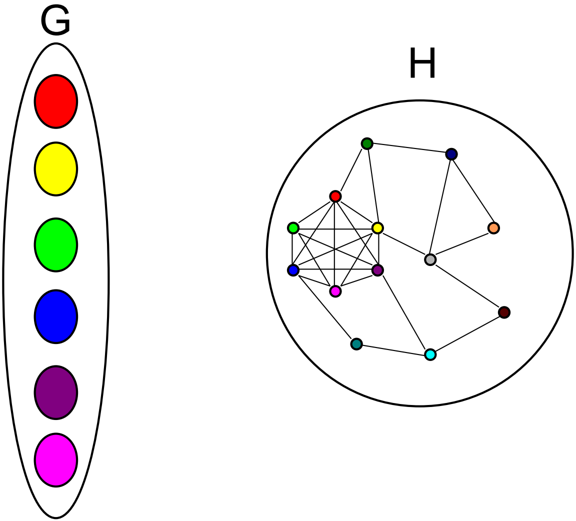


Hardness I: Properties that hold for few distinct edge densities

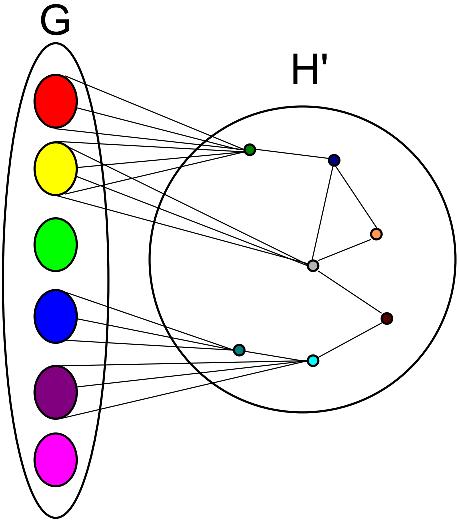
- We prove hardness of **p-#**MULTICOLOUR INDUCED SUBGRAPH WITH PROPERTY(Φ) by means of a reduction from **p-#**MULTICOLOUR CLIQUE.



Hardness I: Properties that hold for few distinct edge densities



Hardness I: Properties that hold for few distinct edge densities



Hardness I: Properties that hold for few distinct edge densities

Lemma

Let $G = (V, E)$ be an n-vertex graph, where $n \geq 2^k$. Then the number of k-vertex subsets $U \subset V$ such that U induces either a clique or independent set in G is at least

$$\frac{(2^k - k)!}{(2^k)!} \frac{n!}{(n-k)!}.$$

Hardness II: Connected subgraphs

Theorem

p-#Connected Induced Subgraph *is #W[1]-complete under fpt Turing reductions.*

Hardness II: Connected subgraphs

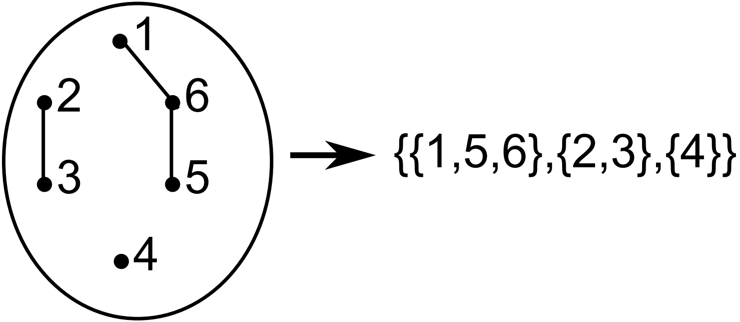
Theorem

p-#CONNECTED INDUCED SUBGRAPH is #W[1]-complete under fpt Turing reductions.

- Prove hardness of **p-#**MULTICOLOUR CONNECTED INDUCED SUBGRAPH
- Reduction from **p-#**MULTICOLOUR INDEPENDENT SET

Hardness II: Connected subgraphs

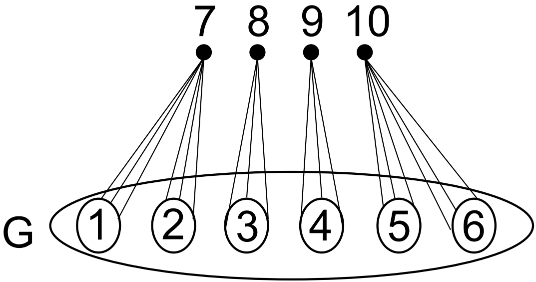
Associate each colourful set of vertices U with a partition $P(U)$ of $\{1, \ldots, k\}$.



Hardness II: Connected subgraphs

For any partition P_i of $\{1, \ldots, k\}$, construct G_{P_i}.

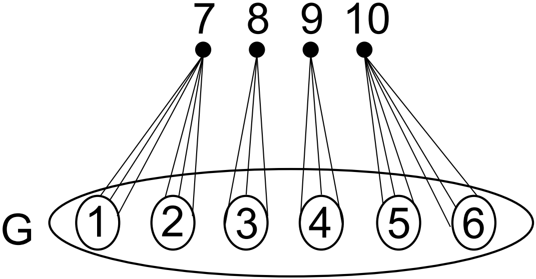
- Suppose $P_i = \{\{1,2\},\{3\},\{4\},\{5,6\}\}$:



Hardness II: Connected subgraphs

For any partition P_i of $\{1,\ldots,k\}$, construct G_{P_i}.

- Suppose $P_i = \{\{1,2\},\{3\},\{4\},\{5,6\}\}$:



- Number of colourful connected induced subgraphs
  $=$ Number of colourful subsets $U \in V(G)^{(k)}$ such that
  $P(U) \wedge P_i = \{\{1,\ldots,k\}\}$.

Hardness II: Connected subgraphs

- Let N_i be the number of colourful subsets $U \in V^{(k)}$ such that $P(U) = P_i$.

Hardness II: Connected subgraphs

- Let N_i be the number of colourful subsets $U \in V^{(k)}$ such that $P(U) = P_i$.
- Set
$$a_{ij} = \begin{cases} 1 & \text{if } P_i \wedge P_j = \{\{1, \ldots, k\}\} \\ 0 & \text{otherwise.} \end{cases}$$

Hardness II: Connected subgraphs

- Let N_i be the number of colourful subsets $U \in V^{(k)}$ such that $P(U) = P_i$.
- Set

$$a_{ij} = \begin{cases} 1 & \text{if } P_i \wedge P_j = \{\{1, \ldots, k\}\} \\ 0 & \text{otherwise.} \end{cases}$$

- We can compute

$$\begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,B_k} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,B_k} \\ \vdots & \vdots & \ddots & \vdots \\ a_{B_k,1} & a_{B_k,2} & \cdots & a_{B_k,B_k} \end{pmatrix} \cdot \begin{pmatrix} N_0 \\ N_1 \\ \vdots \\ N_{B_k} \end{pmatrix}$$

Approximation algorithm

Theorem

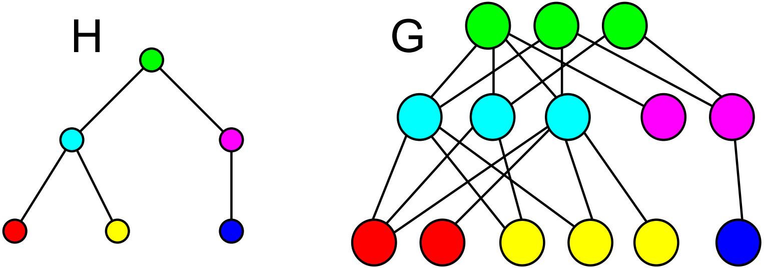
Let $\Phi = (\phi_1, \phi_2, \ldots)$ be a monotone property, and suppose there exists a positive integer t such that, for each ϕ_k, all edge-minimal labelled k-vertex graphs (H, π) such that $\phi_k(H) = 1$ satisfy treewidth$(H) \leq t$. Then there is an FPTRAS for **p-#**$\textsc{Induced Subgraph With Property}$*($\Phi$).*

Approximation algorithm

- Colour the vertices of G with k colours.

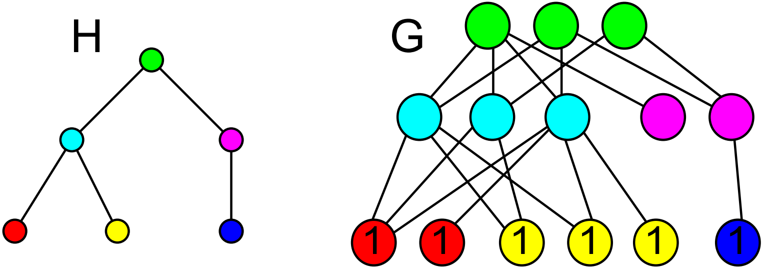
Approximation algorithm

- Colour the vertices of G with k colours.
- For each minimal element H, and each colouring of H with k colours:



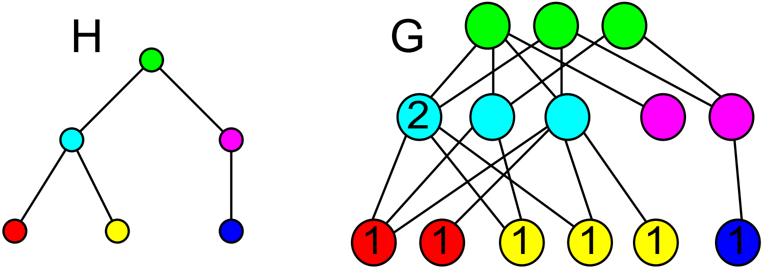
Approximation algorithm

- Colour the vertices of G with k colours.
- For each minimal element H, and each colouring of H with k colours:



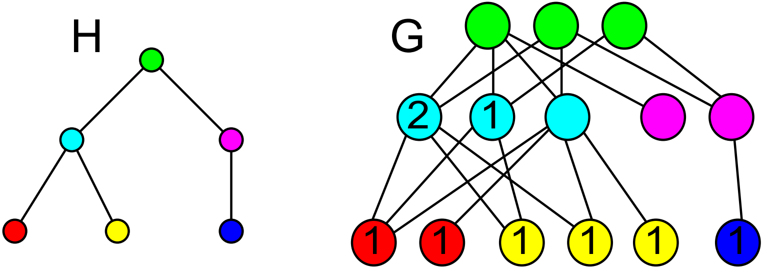
Approximation algorithm

- Colour the vertices of G with k colours.
- For each minimal element H, and each colouring of H with k colours:



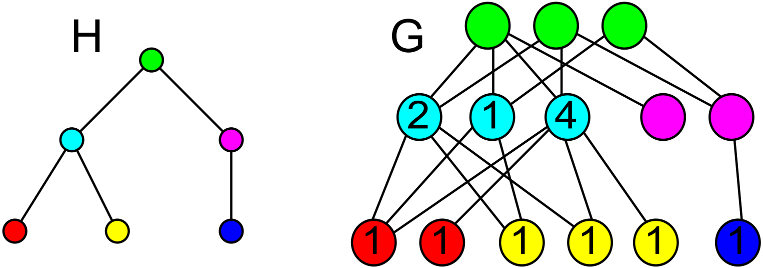
Approximation algorithm

- Colour the vertices of G with k colours.
- For each minimal element H, and each colouring of H with k colours:



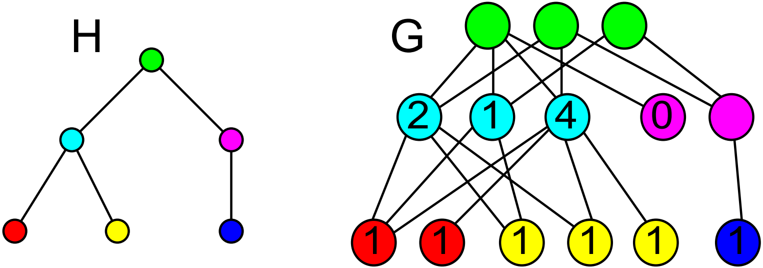
Approximation algorithm

- Colour the vertices of G with k colours.
- For each minimal element H, and each colouring of H with k colours:



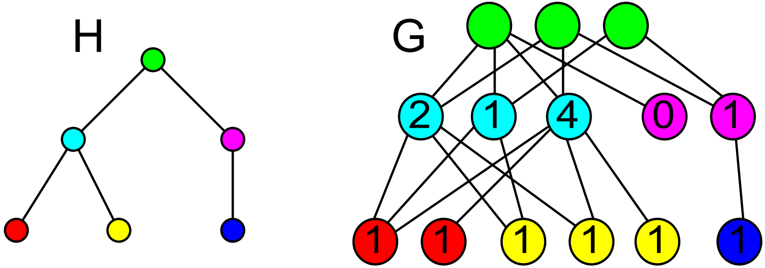
Approximation algorithm

- Colour the vertices of G with k colours.
- For each minimal element H, and each colouring of H with k colours:



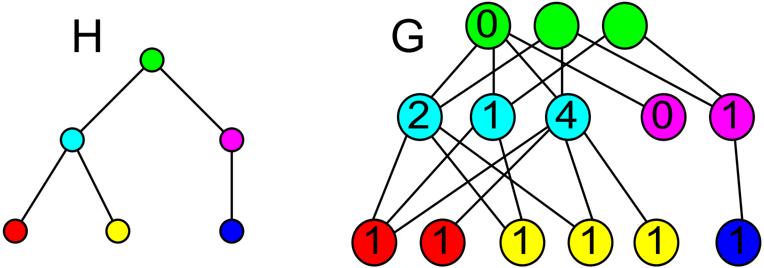
Approximation algorithm

- Colour the vertices of G with k colours.
- For each minimal element H, and each colouring of H with k colours:



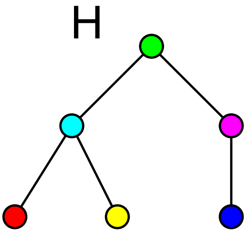
Approximation algorithm

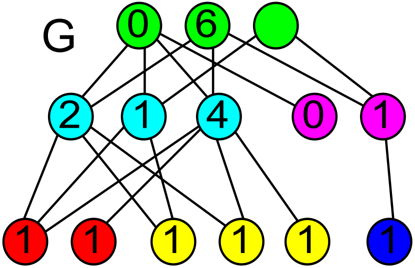
- Colour the vertices of G with k colours.
- For each minimal element H, and each colouring of H with k colours:



Approximation algorithm

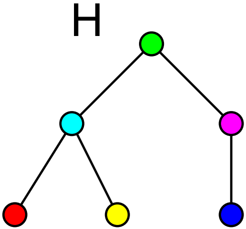
- Colour the vertices of G with k colours.
- For each minimal element H, and each colouring of H with k colours:

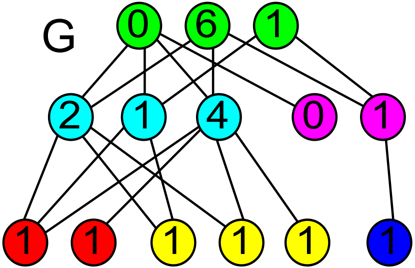




Approximation algorithm

- Colour the vertices of G with k colours.
- For each minimal element H, and each colouring of H with k colours:





Open problems

Open problems

- Are there any non-trivial properties in this model that can be counted exactly in FPT time?

Open problems

- Are there any non-trivial properties in this model that can be counted exactly in FPT time?
- Is there an FPTRAS for any monotone property where the minimal elements with the property do **not** all have bounded treewidth?

Open problems

- Are there any non-trivial properties in this model that can be counted exactly in FPT time?
- Is there an FPTRAS for any monotone property where the minimal elements with the property do **not** all have bounded treewidth?
- What is the complexity of **p-**#Induced Subgraph With Property(Φ) when ϕ_k is true precisely on k-vertex induced subgraphs which have an even number of edges?

THANK YOU

http://arxiv.org/abs/1308.1575

http://arxiv.org/abs/1310.6524