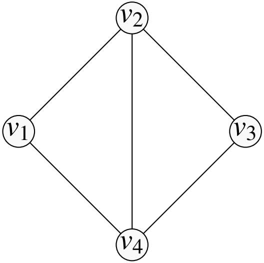
Approximately Counting Graph Homomorphisms

Leslie Ann Goldberg, University of Oxford
Joint work with Andreas Galanis and Mark Jerrum

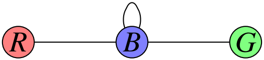
The Classification Program of Counting Complexity
Simons Institute for the Theory of Computing
28 March — 1 April, 2016

H-Colourings

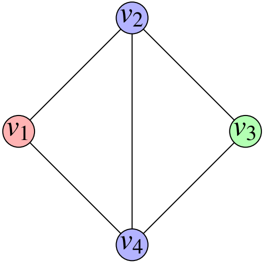
Input: a (simple) graph G



Template: A graph H (possibly with self-loops)

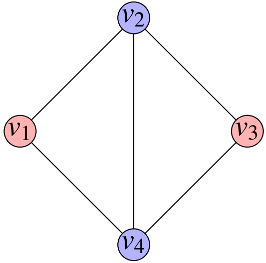


A homomorphism from G to H is a function from $V(G)$ to $V(H)$ which maps every edge of G to an edge of H. It is referred to as an H-colouring of G

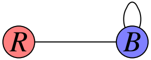


Many combinatorial structures can be represented as H-colourings, for example independent sets

Homomorphism from G to H

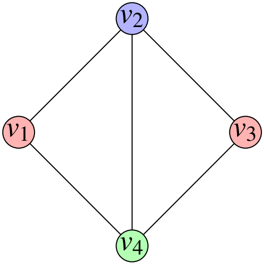


Template H

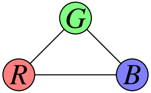


H-colourings of G are independent sets of G.
(Red corresponds to being in the independent set.)

Homomorphism from G to H



Template H



H-colourings of G are proper 3-colourings of G.

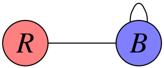
The H-Colouring Decision Problem

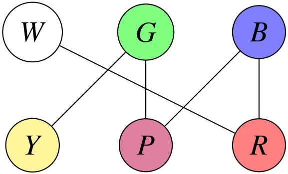
Name H-Col

Instance A simple graph G.

Output Does G have an H-Colouring?

Hell and Nešetřil (1990): H-Col is in P if H has a loop or is bipartite. For all other H it is NP-complete.





The polynomial-time algorithm is easy. The hardness result is difficult.

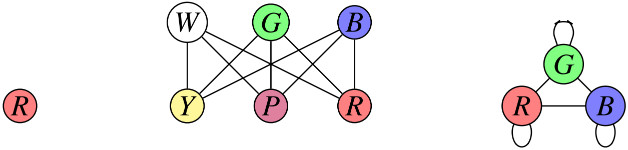
The H-Colouring Counting Problem

Name #H-Col

Instance A simple graph G.

Output #H-Col(G) (the number of H-Colourings of G)

Dyer and Greenhill (2000): #H-Col is in FP if every component of H is trivial (either a clique with all self-loops or a complete bipartite graph with no self-loops). For all other H it is #P-complete.



The polynomial-time algorithm is easy. The hardness result is difficult.

What about approximately counting H-colourings.

FPRAS: randomised algorithm

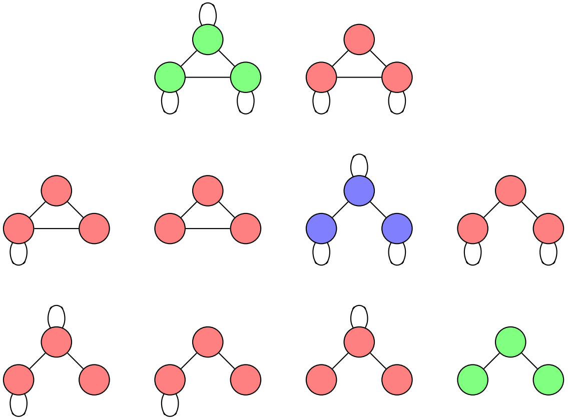
Input: graph G, accuracy parameter $\varepsilon > 0$

Output: number which, with probability at least $3/4$, is in the range

$$[e^{-\varepsilon}\#H\text{-Col}(G), e^{\varepsilon}\#H\text{-Col}(G)].$$

The running time of the algorithm is bounded by a polynomial in $|V(G)|$ and ε^{-1}.

Warm-up: connected 3-vertex H



Green: In FP

Red: $\equiv_{\mathrm{AP}}$ #SAT. No FPRAS unless every problem in #P has an FPRAS (hence NP=RP)

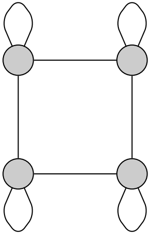
Blue: $\equiv_{\mathrm{AP}}$ #BIS. No FPRAS unless every problem in #RHΠ_1 has an FPRAS.

Dyer, Goldberg, Greenhill, Jerrum, 2003

#RHΠ_1: introduced to classify approximation problems.

#A $\leqslant_{\mathrm{AP}}$ #B $\Rightarrow$ FPRAS for #B yields an FPRAS for #A.

We don't have a trichotomy for all connected H



#BIS-hard but we don't know whether it is #BIS-easy or #SAT-hard.

As hard as counting 4-colourings of a bipartite graph since each vertex is adjacent to all but one.

Other such problems: Kelk 2003

No life below #BIS . . . detour into complexity of sampling

PAS (polynomial approximate sampler) for sampling H-colourings

Input: graph G, accuracy parameter $\varepsilon \in (0, 1]$

Output: total variation distance between the output distribution of the algorithm and the uniform distribution on H-colourings of G is at most ε.

Running time bounded by polynomial in $|V(G)|$ and ε^{-1}.

FPAS if the running time is bounded by a polynomial in $|V(G)|$ and $\log(\varepsilon^{-1})$.

#BIS-hardness of approximate sampling

Theorem. Goldberg, Kelk, Paterson, 2004 Let H be a fixed graph with no trivial components. If there is a PAS for sampling H-colourings then there is an FPRAS for #BIS.

#BIS-hardness of approximate sampling

Theorem. Goldberg, Kelk, Paterson, 2004 Let H be a fixed graph with no trivial components. If there is a PAS for sampling H-colourings then there is an FPRAS for #BIS.

and hence for all problems that are $\equiv_{\text{AP}}$ #BIS

Disconnected H

For counting H-colourings, the existence of a single component H_1 of H such that counting H_1-colourings is #P-complete means that counting H-colourings is #P-complete. The same does not hold for sampling/approximate counting. Consider this H.

There is a PAS for sampling H-colourings.

Theorem. *Goldberg, Kelk, Paterson, 2004* *Let H be a fixed graph with no trivial components. If there is a* PAS *for sampling H-colourings then there is an FPRAS for* #BIS.

- Doesn't imply hardness for approximately counting H-colourings.

Theorem. *Goldberg, Kelk, Paterson, 2004* *Let H be a fixed graph with no trivial components. If there is a* PAS *for sampling H-colourings then there is an FPRAS for* #BIS.

- Doesn't imply hardness for approximately counting H-colourings.
- For self-reducible problems, approximate counting and approximate sampling are equivalent.

Jerrum Valiant Vazirani 1986

Theorem. Goldberg, Kelk, Paterson, 2004 Let H be a fixed graph with no trivial components. If there is a PAS *for sampling H-colourings then there is an FPRAS for* #BIS.

- Doesn't imply hardness for approximately counting H-colourings.
- For self-reducible problems, approximate counting and approximate sampling are equivalent.
- An FPAS for sampling H-colourings implies an FPRAS for counting H-colourings (Dyer, Goldberg, Jerrum 2004) but the reverse direction is open.

gives example of a problem in #P is given which, under usual complexity theory assumptions, admits an FPRAS but not an FPAS.

No life below #BIS

- "Is there a graph H for which approximately counting H-colourings is substantially easier than approximately sampling H-colourings?"
- "Is there a graph H such that #H-Col lies between P and the class of #BIS-hard problems?"

No life below #BIS

- "Is there a graph H for which approximately counting H-colourings is substantially easier than approximately sampling H-colourings?"
- "Is there a graph H such that #H-Col lies between P and the class of #BIS-hard problems?"

Theorem. *(Galanis, Goldberg, Jerrum, 2015) Let H be a graph (possibly with self-loops but without parallel edges), all of whose connected components are non-trivial. Then #BIS $\leqslant_{\mathrm{AP}}$ #H-Col.*

No life below #BIS

- "Is there a graph H for which approximately counting H-colourings is substantially easier than approximately sampling H-colourings?"
- "Is there a graph H such that #H-Col lies between P and the class of #BIS-hard problems?"

Theorem. (Galanis, Goldberg, Jerrum, 2015) *Let H be a graph (possibly with self-loops but without parallel edges), all of whose connected components are non-trivial. Then #BIS $\leqslant_{\mathrm{AP}}$ #H-Col.*

Key Technique: Prove existence of gadgets using tools from graph homomorphism theory (based on work of Lovász1967)

Proof overview

Goal: #BIS $\leqslant_{\mathrm{AP}}$ #H-Col

Proof overview

Goal: #BIS $\leqslant_{\text{AP}}$ #H-Col

Proof by induction on $|V(H)|$

- Base case: 2-vertex graphs.

Proof overview

Goal: #BIS $\leqslant_{\mathsf{AP}}$ #H-Col

Proof by induction on $|V(H)|$

- Base case: 2-vertex graphs.
- Inductive step: find a subgraph H' of H

1. #H'-Col $\leqslant_{\mathsf{AP}}$ #H-Col.
2. $|V(H')| < |V(H)|$
3. H' has no trivial components.

Proof overview

Goal: #BIS $\leqslant_{\mathsf{AP}}$ #H-Col

Proof by induction on $|V(H)|$

- Base case: 2-vertex graphs.
- Inductive step: find a subgraph H' of H

1. #H'-Col $\leqslant_{\mathsf{AP}}$ #H-Col.
2. $|V(H')| < |V(H)|$
3. H' has no trivial components.

By induction, #BIS $\leqslant_{\mathsf{AP}}$ #H'-Col $\leqslant_{\mathsf{AP}}$ #H-Col.

Proof overview

Goal: #BIS $\leqslant_{\text{AP}}$ #H-Col

Proof by induction on $|V(H)|$

- Base case: 2-vertex graphs.
- Inductive step: find a subgraph H' of H

1. #H'-Col $\leqslant_{\text{AP}}$ #H-Col.
2. $|V(H')| < |V(H)|$
3. H' has no trivial components.

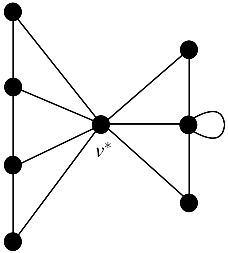
By induction, #BIS $\leqslant_{\text{AP}}$ #H'-Col $\leqslant_{\text{AP}}$ #H-Col.

Main Task: Finding such a subgraph H' of H

A First Attempt

Goal: find subgraph H' of H

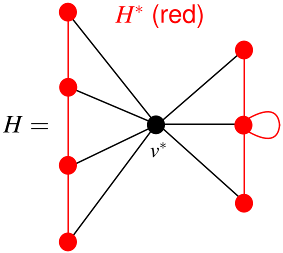
1. $\#H'$-Col $\leqslant_{\text{AP}}$ $\#H$-Col.
2. $|V(H')| < |V(H)|$
3. H' has no trivial components.

$H =$ 

A First Attempt

Goal: find subgraph H' of H

1. $\#H'$-Col $\leqslant_{\mathsf{AP}}$ $\#H$-Col.
2. $|V(H')| < |V(H)|$
3. H' has no trivial components.

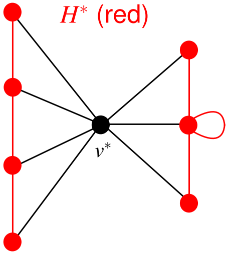


This H has a unique v^* with maximum degree. Let H^* be induced by its neighbourhood.

A First Attempt

Goal: find subgraph H' of H

1. $\#H'$-Col $\leqslant_{\mathrm{AP}}$ $\#H$-Col.
2. $|V(H')| < |V(H)|$
3. H' has no trivial components.

$H =$ 

$\#H^*$-Col $\leqslant_{\mathrm{AP}}$ $\#H$-Col

We will show

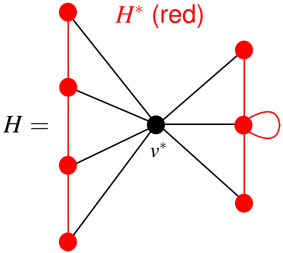
A First Attempt

Goal: find subgraph H' of H

1. $\#H'$-Col $\leqslant_{\text{AP}}$ $\#H$-Col.
2. $|V(H')| < |V(H)|$
3. H' has no trivial components.

G': input to $\#H^*$-Col

I: large independent set



$\#H^*$-Col $\leqslant_{\text{AP}}$ $\#H$-Col

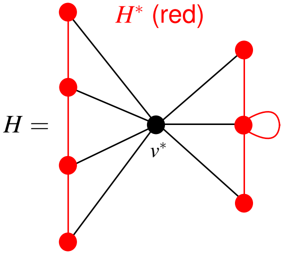
A First Attempt

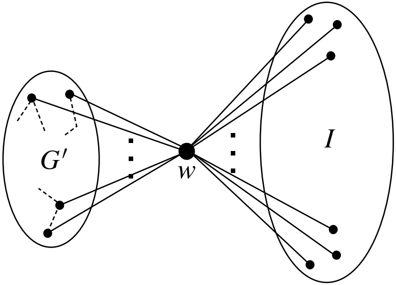
Goal: find subgraph H' of H

1. $\#H'$-Col $\leqslant_{\text{AP}}$ $\#H$-Col.
2. $|V(H')| < |V(H)|$
3. H' has no trivial components.

G': input to $\#H^*$-Col

I: large independent set





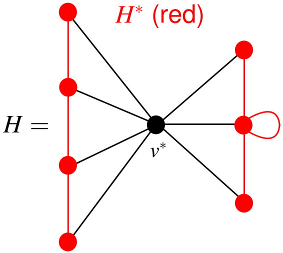
The graph G

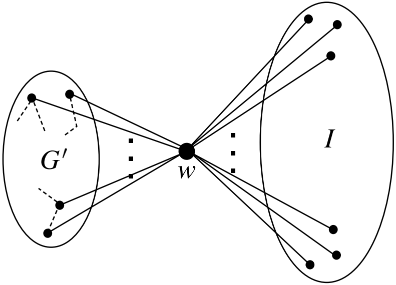
$\#H^*$-Col $\leqslant_{\text{AP}}$ $\#H$-Col

A First Attempt

Goal: find subgraph H' of H

1. $\#H'$-Col $\leqslant_{\mathrm{AP}}$ $\#H$-Col.
2. $|V(H')| < |V(H)|$
3. H' has no trivial components.





The graph G

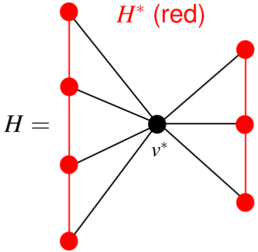
Potential problems for general H: (i) H^* may have trivial components, (ii) it may be that $V(H^*) = V(H)$, (iii) multiple vertices with max degree

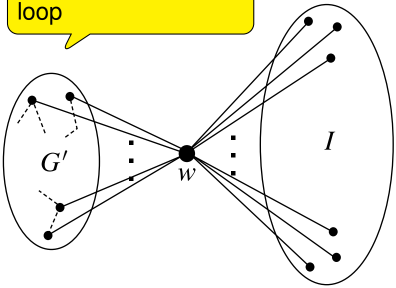
A First Attempt

Goal: find subgraph H' of H

1. $\#H'$-Col $\leqslant_{AP}$ $\#H$-Col.
2. $|V(H')| < |V(H)|$
3. H' has no trivial components.

For example, what if H didn't have its loop





The graph G

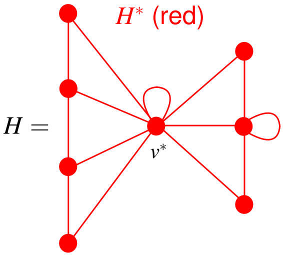
Potential problems for general H: (i) H^* may have trivial components,
(ii) it may be that $V(H^*) = V(H)$,
(iii) multiple vertices with max degree

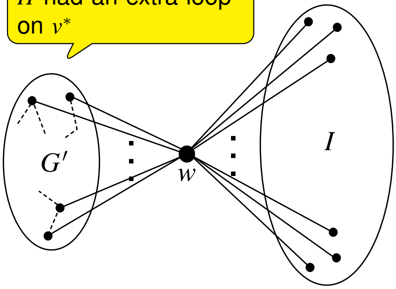
A First Attempt

Goal: find subgraph H' of H

1. $\#H'$-Col $\leqslant_{\text{AP}}$ $\#H$-Col.
2. $|V(H')| < |V(H)|$
3. H' has no trivial components.

For example, what if H had an extra loop on v^*





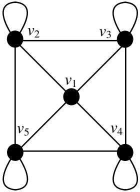
The graph G

Potential problems for general H: (i) H^* may have trivial components,

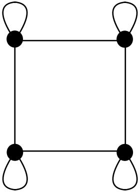
(ii) it may be that $V(H^*) = V(H)$,

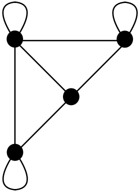
(iii) multiple vertices with max degree

A First Attempt - Multiple vertices with Max Degree



The graph H


The graph H_1

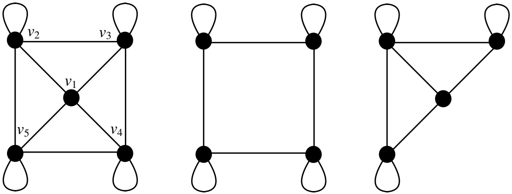

The graph H_2

H_1 induced by the neighbourhood of the max-degree vertex v_1

H_2 induced by the neighbourhood of the max-degree vertex v_2

H_3, H_4, H_5 isomorphic to H_2

A First Attempt - Multiple vertices with Max Degree



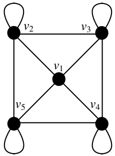
The graph H The graph H_1 The graph H_2

H_1 induced by the neighbourhood of the max-degree vertex v_1

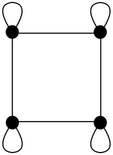
H_2 induced by the neighbourhood of the max-degree vertex v_2

#BIS $\leqslant_{\text{AP}}$ #H_1-Col and #BIS $\leqslant_{\text{AP}}$ #H_2-Col (Kelk 2004)

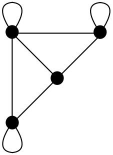
The Sampling World - Gluing Reductions



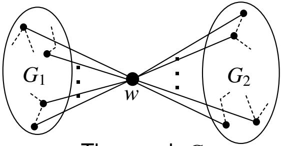
The graph H



The graph H_1



The graph H_2



The graph G

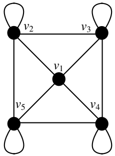
G': Input to SampleBIS.

Using sampling reductions from SampleBIS to Sample-H_1-Col and Sample-H_2-Col, construct G_1 and G_2 such that

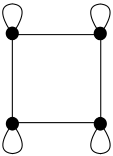
- an (approximately) uniform H_1-colouring of G_1 allows us to construct an (approximately) uniform independent set of G'
- an (approximately) uniform H_2-colouring of G_2 also allows this.

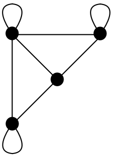
From random H-colouring of G, construct random independent set of G'.

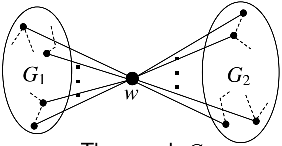
The Sampling World - Gluing Reductions



The graph H


The graph H_1


The graph H_2



The graph G

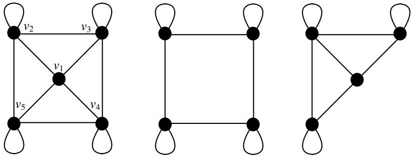
Didn't include I in G because H is regular

G': Input to SampleBIS.

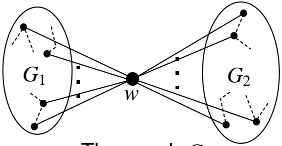
Using sampling reductions from SampleBIS to Sample-H_1-Col and Sample-H_2-Col, construct G_1 and G_2 such that

- an (approximately) uniform H_1-colouring of G_1 allows us to construct an (approximately) uniform independent set of G'
- an (approximately) uniform H_2-colouring of G_2 also allows this.

The Sampling World - Gluing Reductions



The graph H The graph H_1 The graph H_2



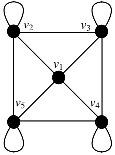
The graph G

This fails in the counting setting!

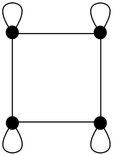
$$\#H\text{-Col}(G) = \#H_1\text{-Col}(G_1)\,\#H_1\text{-Col}(G_2) + 4\#H_2\text{-Col}(G_1)\,\#H_2\text{-Col}(G_2)$$

An approximation Z of $\#H$-Col(G) may not tell us much about $\#H_1$-Col(G_1) or $\#H_2$-Col(G_2).

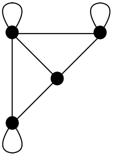
The Sampling World - Gluing Reductions



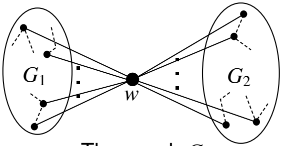
The graph H



The graph H_1



The graph H_2



The graph G

This fails in the counting setting!

$$\#H\text{-Col}(G) = \#H_1\text{-Col}(G_1)\,\#H_1\text{-Col}(G_2) + 4\#H_2\text{-Col}(G_1)\,\#H_2\text{-Col}(G_2)$$

- We have to somehow choose between H_1 and H_2.
- In general, there may be more than two possibilities.

Tools from Graph Homomorphisms

[Lovász '67]: $H_1 \ncong H_2 \implies$ there exists J:
$\#H_1\text{-Col}(J) \neq \#H_2\text{-Col}(J)$

Extension: If $H_1, \ldots, H_t$ are pairwise non-isomorphic,

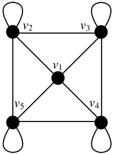
there exists i^* and a graph J so that $\#H_{i^*}\text{-Col}(J) > \#H_i\text{-Col}(J)$ for all $i \neq i^*$.

J will be used to "select" the subgraph H_{i^*}.

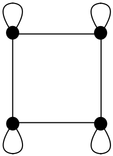
A First Attempt - Multiple vertices with Max Degree

Goal: find subgraph H' of H

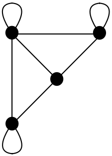
1. $\#H'$-Col $\leqslant_{\text{AP}}$ $\#H$-Col.
2. $|V(H')| < |V(H)|$
3. H' has no trivial components.



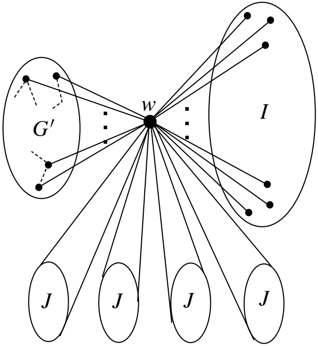
The graph H



The graph H_1



The graph H_2



The graph G

J: $\#H_1$-Col$(J) > \#H_2$-Col(J) (by Lovász)
G': Input to $\#H_1$-Col.

m copies of J.