The complexity of approximate counting
Part 1

Leslie Ann Goldberg, University of Oxford

Counting Complexity and Phase Transitions Boot Camp

Simons Institute for the Theory of Computing

January 2016

The Complexity of Approximate Counting

- Relative Complexity and #BIS: This talk
- Markov Chain Monte Carlo: Ivona Bezáková, Nayantara Bhatnagar (next talk!)
- Approximate Counting and Constraint Satisfaction Problems David Richerby (Part 2)
- Correlation Decay and Phase Transitions Yitong Yin

Credits

Cai, Chebolu, Dyer, Galanis, Greenhill, Guo, Gysel, Jerrum, Kelk, Lapinskas, Martin, Paterson, Štefankovič, Vigoda

Example: The Potts model

Interaction strength $\gamma \geqslant -1$.

Set of spins $[q]$.

Graph $G = (V, E)$.

1 if spins are the same and 0 otherwise.

partition function

$$Z_{\text{Potts}}(G; q, \gamma) = \sum_{\sigma : V \to [q]} \prod_{e=\{u,v\} \in E} \big(1 + \gamma\, \delta(\{\sigma(u), \sigma(v)\})\big)$$

Configuration σ: assigns spins to vertices

Example: The Potts model

Interaction strength $\gamma \geqslant -1$.

Set of spins $[q]$.

Ising $q = 2$

Graph $G = (V, E)$.

partition function

ferromagnetic $\gamma > 0$

$$Z_{\text{Potts}}(G; q, \gamma) = \sum_{\sigma : V \to [q]} \prod_{e=\{u,v\} \in E} \big(1 + \gamma\, \delta(\{\sigma(u), \sigma(v)\})\big)$$

Example: The Potts model

Interaction strength $\gamma \geqslant -1$.

Set of spins $[q]$.

Graph $G = (V, E)$.

$\gamma = -1$ counts proper q-colourings

partition function

$$Z_{\text{Potts}}(G; q, \gamma) = \sum_{\sigma : V \to [q]} \prod_{e = \{u,v\} \in E} \big(1 + \gamma\, \delta(\{\sigma(u), \sigma(v)\})\big)$$

Example: The Potts model

Interaction strength $\gamma \geqslant -1$.

Set of spins $[q]$.

Graph $G = (V, E)$.

partition function

$$Z_{\mathrm{Potts}}(G; q, \gamma) = \sum_{\sigma : V \to [q]} \prod_{e = \{u,v\} \in E} \big(1 + \gamma\, \delta(\{\sigma(u), \sigma(v)\})\big)$$

- "computational counting": computing sums of products.
- In $\mathrm{FP}^{\#\mathrm{P}}$. We'll focus on approximate counting within #P (or within $\mathrm{FP}^{\#\mathrm{P}}$).

In #P up to "easily-computable factor"

Example: The Potts model

Interaction strength $\gamma \geqslant -1$.

Set of spins $[q]$.

Graph $G = (V, E)$.

partition function

later λ_e

$$Z_{\text{Potts}}(G; q, \gamma) = \sum_{\sigma: V \to [q]} \prod_{e=\{u,v\} \in E} \big(1 + \gamma\, \delta(\{\sigma(u), \sigma(v)\})\big)$$

- "computational counting": computing sums of products.
- In $\text{FP}^{\#\text{P}}$. We'll focus on approximate counting within #P (or within $\text{FP}^{\#\text{P}}$).

The goal: an FPRAS

A randomised approximation scheme (RAS) is an algorithm for approximately computing the value of a function $f : \Sigma^* \to \mathbb{R}$.

finite alphabet Σ

Input:

- instance $x \in \Sigma^*$
- rational error tolerance $\varepsilon \in (0, 1)$

e.g., if $f = Z_{\text{Potts}}$ then x encodes G

The goal: an FPRAS

A randomised approximation scheme (RAS) is an algorithm for approximately computing the value of a function $f : \Sigma^* \to \mathbb{R}$.

Input:

- instance $x \in \Sigma^*$
- rational error tolerance $\varepsilon \in (0, 1)$

Output: Rational number z such that, for all x,

$$\Pr\left(e^{-\varepsilon} f(x) \leqslant z \leqslant e^{\varepsilon} f(x)\right) \geqslant \frac{3}{4}.$$

z is a random variable, depending on the "coin tosses" made by the algorithm"

The goal: an FPRAS

A randomised approximation scheme (RAS) is an algorithm for approximately computing the value of a function $f : \Sigma^* \to \mathbb{R}$.

Input:

- instance $x \in \Sigma^*$
- rational error tolerance $\varepsilon \in (0, 1)$

Output: Rational number z such that, for all x,

$$\Pr\left(e^{-\varepsilon} f(x) \leqslant z \leqslant e^{\varepsilon} f(x)\right) \geqslant \frac{3}{4}.$$

FPRAS: Running time bounded by a polynomial in $|x|$ and ε^{-1}.

e.g., if $f = Z_{\text{Potts}}$ then x encodes G and $|x| = n$.

“No FPRAS”: typically, we can’t even get close!

- $k \cdot G$: k disjoint copies of G.
- $Z_{\text{Potts}}(k \cdot G; q, \gamma) = Z_{\text{Potts}}(G; q, \gamma)^k$.
- Set $k = O\left(\frac{1}{\varepsilon}\right)$
  - given a constant factor approximation to $Z_{\text{Potts}}(k \cdot G; q, \gamma)$
  - take k’th root
  - get FPRAS for $Z_{\text{Potts}}(G; q, \gamma)$.

contrast with optimisation!

An approximation within a polynomial factor would also suffice.

How difficult is it to FPRAS a problem in #P?

under (randomised) polynomial-time Turing reductions

- It can be NP-hard

Obviously, an FPRAS for counting satisfying assignments will tell you, with high probability, whether there is one.

How difficult is it to FPRAS a problem in #P?

- It can be NP-hard
- But it can't be much harder
  - Valiant, Vazirani 1986 bisection technique
  - #SAT can be approximated by a probabilistic polynomial-time Turing machine using an oracle for SAT.

FPRAS for #SAT

How difficult is it to FPRAS a problem in #P?

- It can be NP-hard
- But it can't be much harder
  - Valiant, Vazirani 1986 bisection technique
  - #SAT can be approximated by a probabilistic polynomial-time Turing machine using an oracle for SAT.
  - Given an FPRAS for #SAT, obtain an FPRAS for any problem in #P

Cook's theorem is parsimonious

number of accepting computations of Turing machine/input pair $=$ number of satisfying assignments of the constructed formula

Relative Complexity of Approximate Counting

f, g: functions from Σ^* to $\mathbb{N}$.

AP-reduction from f to g:

apologies to Pilu Crescenzi (1997)!

Relative Complexity of Approximate Counting

f, g: functions from Σ^* to $\mathbb{N}$.

AP-reduction from f to g: randomised algorithm $\mathcal{A}$ for computing f using an oracle for g.

Input: $(x, \varepsilon) \in \Sigma^* \times (0, 1)$.
x is an instance of f

Relative Complexity of Approximate Counting

f, g: functions from Σ^* to $\mathbb{N}$.

AP-reduction from f to g: randomised algorithm $\mathcal{A}$ for computing f using an oracle for g.

Input: $(x, \varepsilon) \in \Sigma^* \times (0, 1)$.
x is an instance of f

1. $\mathcal{A}$ makes oracle calls (w, δ)

w is an instance of g. $0 < \delta < 1$ is an error bound satisfying $\delta^{-1} \leqslant \text{poly}(|x|, \varepsilon^{-1})$

Relative Complexity of Approximate Counting

f, g: functions from Σ^* to $\mathbb{N}$.

AP-reduction from f to g: randomised algorithm $\mathcal{A}$ for computing f using an oracle for g.

Input: $(x, \varepsilon) \in \Sigma^* \times (0, 1)$.
x is an instance of f

1. $\mathcal{A}$ makes oracle calls (w, δ)
2. $\mathcal{A}$ meets the specification for being a RAS for f whenever the oracle meets the specification for being a RAS for g

Relative Complexity of Approximate Counting

f, g: functions from Σ^* to $\mathbb{N}$.

AP-reduction from f to g: randomised algorithm $\mathcal{A}$ for computing f using an oracle for g.

Input: $(x, \varepsilon) \in \Sigma^* \times (0, 1)$.
x is an instance of f

1. $\mathcal{A}$ makes oracle calls (w, δ)
2. $\mathcal{A}$ meets the specification for being a RAS for f whenever the oracle meets the specification for being a RAS for g
3. the run-time of $\mathcal{A}$ is polynomial in $|x|$ and ε^{-1}.

Relative Complexity of Approximate Counting

f, g: functions from Σ^* to $\mathbb{N}$.

AP-reduction from f to g: randomised algorithm $\mathcal{A}$ for computing f using an oracle for g.

Input: $(x, \varepsilon) \in \Sigma^* \times (0, 1)$.
x is an instance of f

1. $\mathcal{A}$ makes oracle calls (w, δ)
2. $\mathcal{A}$ meets the specification for being a RAS for f whenever the oracle meets the specification for being a RAS for g
3. the run-time of $\mathcal{A}$ is polynomial in $|x|$ and ε^{-1}.

The class of functions with an FPRAS is closed under AP-reducibility.

An impossible goal (if NP $\neq$ RP)

A dichotomy within #P:

- FPRASable problems.
- The rest.

All AP-interreducible

All AP-interreducible.
No FPRAS unless NP $=$ RP

An impossible goal (if NP $\neq$ RP)

A dichotomy within #P:

- FPRASable problems.
- The rest.

Bordewich 2010

Like Ladner 1975 for P versus NP.

Let π be a problem in #P such that there is no FPRAS for π.
Then there is a problem $\pi' \in$ #P such that

- there is no FPRAS for π', and
- $\pi \not\leq_{\mathrm{AP}} \pi'$.

Three classes of interreducible classes within #P

- FPRASable problems
- Problems AP-interreducible with #BIS
- Problems AP-interreducible with #SAT

Name #BIS

Instance A bipartite graph B.

Output The number of independent sets in B.

All problems in #P are AP-reducible to #SAT (since a parsimonious reduction is an AP-reduction)

Another impossible goal

A trichotomy within #P:

- FPRASable problems
- Problems $\equiv_{AP}$ #BIS
- Problems $\equiv_{AP}$ #SAT

Bordewich 2010

If there is no FPRAS for #BIS then there is a problem π in #P that does not have an FPRAS such that #BIS $\not\leq_{AP} \pi$.

(Infinite hierarchy below #BIS)

We have candidates above #BIS but nothing natural below

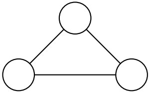
some settings where trichotomies arise ...

Graph Homomorphisms

Homomorphism from G to H

$\sigma\colon V(G) \to V(H)$

for every edge $(u,v) \in E(G), \quad (\sigma(u),\sigma(v)) \in E(H)$

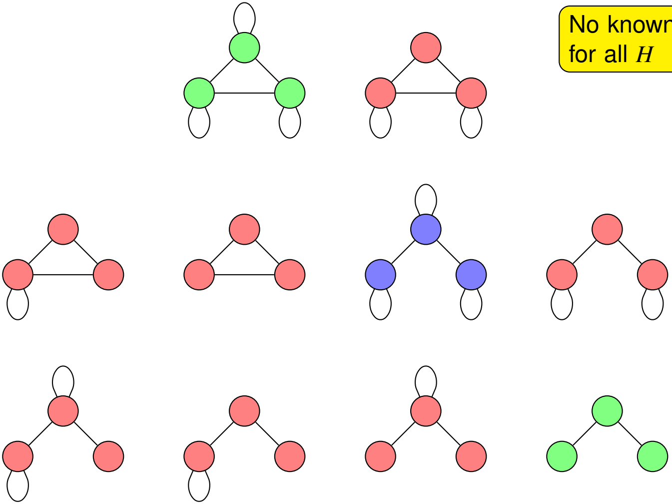


Name #HomsTo(H).

Instance Graph G.

Output The number of homomorphisms from G to H.

connected 3-vertex H



No known trichotomy for all H

A trichotomy for weighted homomorphisms to trees

Weighting function $w\colon V(H) \to \mathbb{Q}_{\geqslant 0}$ assigns non-negative rational weight to each vertex of H.

A trichotomy for weighted homomorphisms to trees

Weighting function $w\colon V(H) \to \mathbb{Q}_{\geqslant 0}$ assigns non-negative rational weight to each vertex of H.

weighting function for each $v \in V(G)$

$$W(G,H) = \{w_v \mid v \in V(G)\}.$$

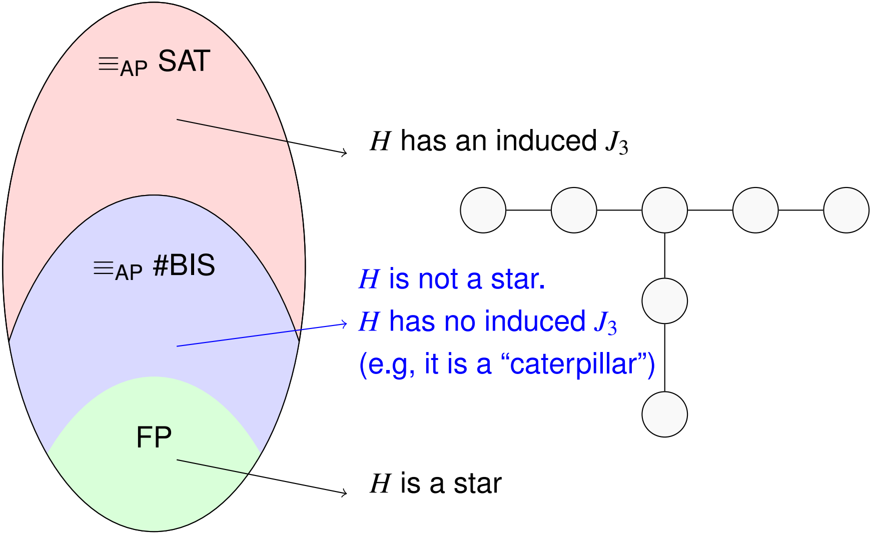
A trichotomy for weighted homomorphisms to trees

Weighting function $w \colon V(H) \to \mathbb{Q}_{\geqslant 0}$ assigns non-negative rational weight to each vertex of H.

$$W(G,H) = \{w_v \mid v \in V(G)\}.$$

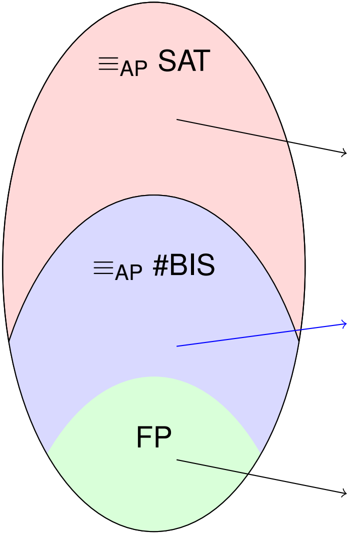
$$Z_H(G, W(G,H)) = \sum_{\sigma \in \mathrm{Hom}(G,H)} \prod_{v \in V(G)} w_v(\sigma(v)).$$

A trichotomy for weighted homomorphisms to trees

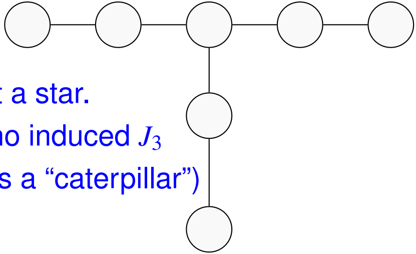


A trichotomy for weighted homomorphisms to trees

Open problems if H not a tree



H has an induced J_3

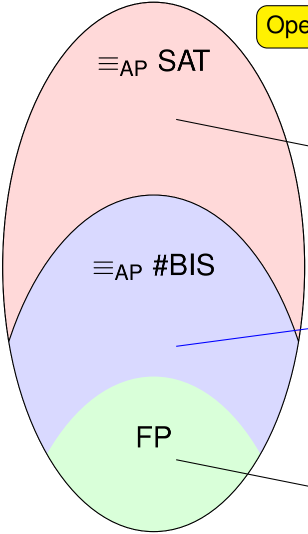


H is not a star.
H has no induced J_3
(e.g, it is a "caterpillar")

H is a star

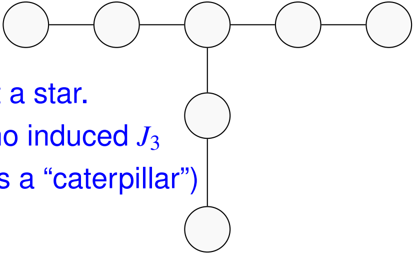
A trichotomy for weighted homomorphisms to trees

Open problems unweighted



Some of red region is still #SAT-hard

H has an induced J_3



H is not a star.

H has no induced J_3

(e.g, it is a “caterpillar”)

H is a star

Connection to the Potts model

For any $q > 2$ and any (efficiently approximable) γ, counting homomorphisms to J_q is AP-equivalent to computing the partition of the ferromagnetic q-state Potts model $Z_{\text{Potts}}(\cdot; q, \gamma)$.

J_q is like J_3 but with q branches

We'll come back to the Potts model

Approximate counting problems which are $\equiv_{AP}$ #BIS

- Counting downsets in a partial order
- Graph homomorphism counting problems
- Counting Constraint Satisfaction (#CSP) problems
- Ferromagnetic Ising with mixed fields
- Ferromagnetic Ising in a hypergraph (even without fields)
- Counting stable matchings (in general, or for geometric preference models)

We've seen some. See also [Kelk 2003]

part 2

follows also from #CSP results

The ferro Ising partition function of a hypergraph

Interaction strength $\gamma > 0$.

$$Z_{\text{Ising}}(H;\gamma) = \sum_{\sigma:V\to\{0,1\}} \prod_{f\in E} \bigl(1+\gamma\,\delta(\{\sigma(v) \mid v\in f\})\bigr)$$

$\delta(S) = 1$ if its argument is a singleton and 0 otherwise.

Back to Binary Matroid Ising

The ferro Ising partition function of a hypergraph

Interaction strength $\gamma > 0$.

$$Z_{\text{Ising}}(H;\gamma) = \sum_{\sigma : V \to \{0,1\}} \prod_{f \in E} \big(1 + \gamma\, \delta(\{\sigma(v) \mid v \in f\})\big)$$

By (Fortuin, Kasteleyn 1972) this is the same as the Tutte polynomial version

$$Z_{\text{Tutte}}(H;\gamma) = \sum_{F \subseteq E} 2^{\kappa(F)} \gamma^{|F|},$$

$\kappa(F)$ is the number of connected components in (V,F): think of the connected components of the underlying graph if you replace hyperedges by cliques

Back to Binary Matroid Ising

The ferro Ising partition function of a hypergraph

Interaction strength $\gamma > 0$.

$$Z_{\text{Ising}}(H;\gamma) = \sum_{\sigma:V\to\{0,1\}} \prod_{f\in E} \left(1 + \gamma\, \delta(\{\sigma(v) \mid v \in f\})\right)$$

By (Fortuin, Kasteleyn 1972) this is the same as the Tutte polynomial version

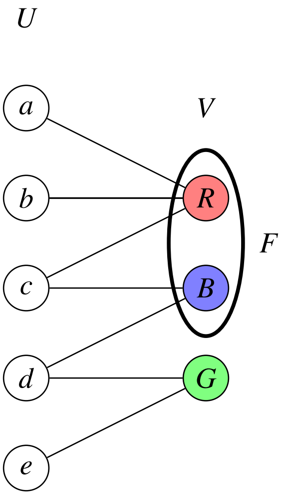
$$Z_{\text{Tutte}}(H;\gamma) = \sum_{F\subseteq E} 2^{\kappa(F)}\gamma^{|F|},$$

Proof: "Integrate out" one of the sums in

$$\sum_{\sigma:V\to\{0,1\}} \sum_{F\subseteq E} \prod_{f\in F} \gamma\, \delta(\{\sigma(v) \mid v \in f\})$$

Back to Binary Matroid Ising

#BIS $\leqslant_{AP}$ Ferromagnetic Hypergraph Ising



IS with F on RHS: $2^{|U|-|\Gamma(F)|}$

Back to FerroPotts

$$Z_{\text{Tutte}}(H; 1) = \sum_{F \subseteq E} 2^{\kappa(F)}$$

- Vertices $U \cup \{v\}$
- Hyperedges
  $R = \{a, b, c, v\}$
  $B = \{c, d, v\}$
  $G = \{d, e, v\}$

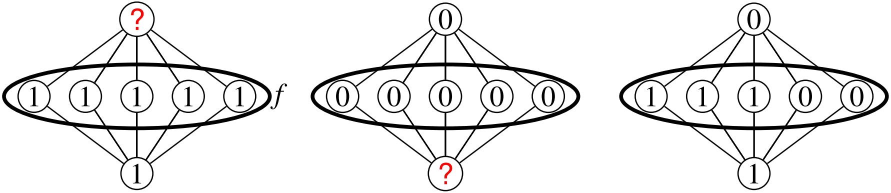
Contribution of F:
$2^{\kappa(F)} = 2^{|U|-\Gamma(F)+1}$.

Ferromagnetic Hypergraph Ising $\leqslant_{AP}$ Downsets

Downsets in a partial order: Represent partial order as directed graph (drawn on the slides with edges pointing down). Spin 1 forces all vertices below to have spin 1

Ferromagnetic Hypergraph Ising $\leqslant_{\mathrm{AP}}$ Downsets

$$Z_{\mathrm{Ising}}(H; 1) = \sum_{\sigma : V \to \{0,1\}} \prod_{f \in E} \left(1 + \delta(\{\sigma(v) \mid v \in f\})\right)$$



Then stretch and thicken to get other γ

Approximate counting problems which are $\equiv_{AP}$ #BIS

- Counting downsets in a partial order
- Graph homomorphism counting problems
- Counting Constraint Satisfaction (#CSP) problems
- Ferromagnetic Ising with mixed fields
- Ferromagnetic Ising in a hypergraph
  (even without fields)
- Counting stable matchings (in general, or for geometric preference models)

Now place #BIS in logically defined class