Counting Hypergraph Colourings in the Local Lemma Regime

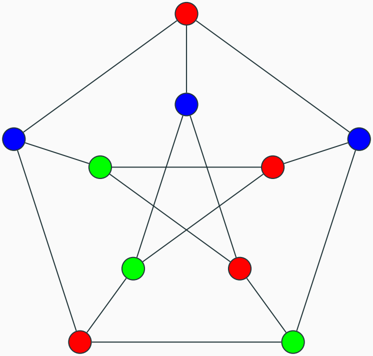
Heng Guo (University of Edinburgh)

Joint with Chao Liao (SJTU), Pinyan Lu (SHUFE), and Chihao Zhang (SJTU)

Berkeley, CA, Mar 22nd, 2019

Colourings

Graph (proper) colouring



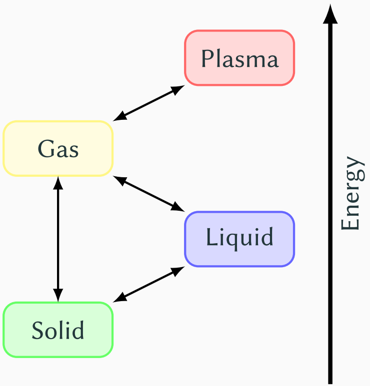
3-colouring of the Petersen graph

Phase transitions

Phase transitions:
as some parameter changes, macroscopic behaviours of the whole system change drastically.

E.g. ice $\to$ water $\to$ water vapor

Computational complexity may also have transitions.



Computational phase transitions

As parameters change, the computational complexity of a problem may change drastically.

Determine whether a graph is q-colourable (or find one if it exists):

- $q = 1, 2$: trivial;
- $q \geqslant 3$: **NP**-hard.

What about graphs with maximum degree Δ?

- $q \geqslant \Delta + 1$: colourable by simple greedy algorithm;
- $q \geqslant \Delta - k_\Delta + 1$: polynomial-time (Molloy, Reed '01 '14);
- $q \leqslant \Delta - k_\Delta$ ($k_\Delta \approx \sqrt{\Delta} - 2$) : **NP**-hard (Embden-Weinert, Hougardy, and Kreuter '98).

Computational phase transitions

As parameters change, the computational complexity of a problem may change drastically.

Determine whether a graph is q-colourable (or find one if it exists):

- $q = 1, 2$: trivial;
- $q \geqslant 3$: **NP**-hard.

What about graphs with maximum degree Δ?

- $q \geqslant \Delta + 1$: colourable by simple greedy algorithm;
- $q \geqslant \Delta - k_\Delta + 1$: polynomial-time (Molloy, Reed '01 '14);
- $q \leqslant \Delta - k_\Delta$ $(k_\Delta \approx \sqrt{\Delta} - 2)$: **NP**-hard (Embden-Weinert, Hougardy, and Kreuter '98).

Computational phase transitions

As parameters change, the computational complexity of a problem may change drastically.

Determine whether a graph is q-colourable (or find one if it exists):

- $q = 1, 2$: trivial;
- $q \geqslant 3$: **NP**-hard.

What about graphs with maximum degree Δ?

- $q \geqslant \Delta + 1$: colourable by simple greedy algorithm;
- $q \geqslant \Delta - k_\Delta + 1$: polynomial-time (Molloy, Reed '01 '14);
- $q \leqslant \Delta - k_\Delta$: **NP**-hard (Embden-Weinert, Hougardy, and Kreuter '98).
  $(k_\Delta \approx \sqrt{\Delta} - 2)$

Thresholds for randomly colouring a graph

Can we generate a uniform proper colouring at random efficiently?
(closely related to approximately count the number of colourings)

- $q > 2\Delta$: rapid mixing of Glauber dynamics by Jerrum (1995); Salas and Sokal (1997);
- $q > \frac{11}{6}\Delta$: rapid mixing of WSK dynamics by Vigoda (2000);
  improved by Chen and Moitra (2019); Delcourt, Perarnau, and Postle (2019) to $q > \left(\frac{11}{6} - \varepsilon\right)\Delta$ for a small constant ε;
- $q < \Delta$ (even q) : **NP**-hard by Galanis, Štefankovič, and Vigoda (2015);

It is conjectured that there is a threshold and $q_c = \Delta + 1$. This is the uniqueness threshold of Gibbs measures in an infinite Δ-regular tree (namely a Bethe lattice), by Jonasson (2002).

Thresholds for randomly colouring a graph

Can we generate a uniform proper colouring at random efficiently?

(closely related to approximately count the number of colourings)

- $q > 2\Delta$: rapid mixing of Glauber dynamics by Jerrum (1995); Salas and Sokal (1997);
- $q > \frac{11}{6}\Delta$: rapid mixing of WSK dynamics by Vigoda (2000);

  improved by Chen and Moitra (2019); Delcourt, Perarnau, and Postle (2019) to $q > \left(\frac{11}{6} - \varepsilon\right)\Delta$ for a small constant ε;
- $q < \Delta$ (even q) : **NP**-hard by Galanis, Štefankovič, and Vigoda (2015);

It is conjectured that there is a threshold and $q_c = \Delta + 1$. This is the uniqueness threshold of Gibbs measures in an infinite Δ-regular tree (namely a Bethe lattice), by Jonasson (2002).

Thresholds for randomly colouring a graph

Can we generate a uniform proper colouring at random efficiently?

(closely related to approximately count the number of colourings)

- $q > 2\Delta$: rapid mixing of Glauber dynamics by Jerrum (1995); Salas and Sokal (1997);
- $q > \frac{11}{6}\Delta$: rapid mixing of WSK dynamics by Vigoda (2000);

  improved by Chen and Moitra (2019); Delcourt, Perarnau, and Postle (2019) to $q > \left(\frac{11}{6} - \varepsilon\right)\Delta$ for a small constant ε;
- $q < \Delta$ (even q) : **NP**-hard by Galanis, Štefankovič, and Vigoda (2015);

It is conjectured that there is a threshold and $q_c = \Delta + 1$. This is the uniqueness threshold of Gibbs measures in an infinite Δ-regular tree (namely a Bethe lattice), by Jonasson (2002).

Thresholds for randomly colouring a graph

Can we generate a uniform proper colouring at random efficiently?

(closely related to approximately count the number of colourings)

- $q > 2\Delta$: rapid mixing of Glauber dynamics by Jerrum (1995); Salas and Sokal (1997);
- $q > \frac{11}{6}\Delta$: rapid mixing of WSK dynamics by Vigoda (2000);

  improved by Chen and Moitra (2019); Delcourt, Perarnau, and Postle (2019) to $q > \left(\frac{11}{6} - \varepsilon\right)\Delta$ for a small constant ε;
- $q < \Delta$ (even q) : **NP**-hard by Galanis, Štefankovič, and Vigoda (2015);

It is conjectured that there is a threshold and $q_c = \Delta + 1$. This is the uniqueness threshold of Gibbs measures in an infinite Δ-regular tree (namely a Bethe lattice), by Jonasson (2002).

Thresholds for randomly colouring a graph

Can we generate a uniform proper colouring at random efficiently?
(closely related to approximately count the number of colourings)

- $q > 2\Delta$: rapid mixing of Glauber dynamics by Jerrum (1995); Salas and Sokal (1997);
- $q > \frac{11}{6}\Delta$: rapid mixing of WSK dynamics by Vigoda (2000);
  improved by Chen and Moitra (2019); Delcourt, Perarnau, and Postle (2019) to $q > \left(\frac{11}{6} - \varepsilon\right)\Delta$ for a small constant ε;
- $q < \Delta$ (even q) : **NP**-hard by Galanis, Štefankovič, and Vigoda (2015);

It is conjectured that there is a threshold and $q_c = \Delta + 1$. This is the uniqueness threshold of Gibbs measures in an infinite Δ-regular tree (namely a Bethe lattice), by Jonasson (2002).

Thresholds for randomly colouring a graph

Can we generate a uniform proper colouring at random efficiently?

(closely related to approximately count the number of colourings)

- $q > 2\Delta$: rapid mixing of Glauber dynamics by Jerrum (1995); Salas and Sokal (1997);
- $q > \frac{11}{6}\Delta$: rapid mixing of WSK dynamics by Vigoda (2000);

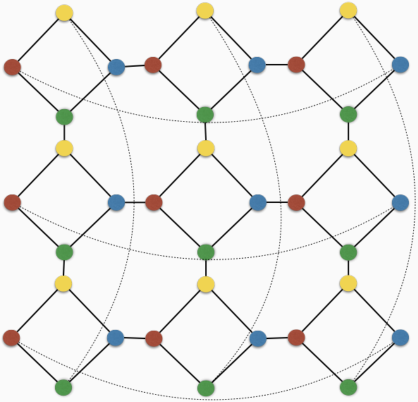
  improved by Chen and Moitra (2019); Delcourt, Perarnau, and Postle (2019) to $q > \left(\frac{11}{6} - \varepsilon\right)\Delta$ for a small constant ε;
- $q < \Delta$ (even q) : **NP**-hard by Galanis, Štefankovič, and Vigoda (2015);

It is conjectured that there is a threshold and $q_c = \Delta + 1$. This is the uniqueness threshold of Gibbs measures in an infinite Δ-regular tree (namely a Bethe lattice), by Jonasson (2002).

Frozen

Sometimes you just cannot let it go.

$$q = \Delta + 1 = 4$$



credit: Chihao Zhang

Deterministic counting of colourings

MCMC is not the only method to count.

Correlation decay:

- Gamarnik, Katz (2012): $q \geqslant \alpha\Delta + \beta$ for large β and $\alpha \approx 2.84$;
- Lu, Yin (2013): $q \geqslant \alpha\Delta + 1$ for $\alpha \approx 2.58$.

Zeros of the chromatic polynomial:

- Patel, Regts (2017): $q > \alpha\Delta$ for $\alpha \approx 6.91$;
- Bencs, Davies, Patel, Regts (2019+): $q \geqslant e\Delta + 1$.

Cannot match randomised algorithms, yet.

Deterministic counting of colourings

MCMC is not the only method to count.

Correlation decay:

- Gamarnik, Katz (2012): $q \geqslant \alpha\Delta + \beta$ for large β and $\alpha \approx 2.84$;
- Lu, Yin (2013): $q \geqslant \alpha\Delta + 1$ for $\alpha \approx 2.58$.

Zeros of the chromatic polynomial:

- Patel, Regts (2017): $q > \alpha\Delta$ for $\alpha \approx 6.91$;
- Bencs, Davies, Patel, Regts (2019+): $q \geqslant e\Delta + 1$.

Cannot match randomised algorithms, yet.

Deterministic counting of colourings

MCMC is not the only method to count.

Correlation decay:

- Gamarnik, Katz (2012): $q \geqslant \alpha\Delta + \beta$ for large β and $\alpha \approx 2.84$;
- Lu, Yin (2013): $q \geqslant \alpha\Delta + 1$ for $\alpha \approx 2.58$.

Zeros of the chromatic polynomial:

- Patel, Regts (2017): $q > \alpha\Delta$ for $\alpha \approx 6.91$;
- Bencs, Davies, Patel, Regts (2019+): $q \geqslant e\Delta + 1$.

Cannot match randomised algorithms, yet.

Deterministic counting of colourings

MCMC is not the only method to count.

Correlation decay:

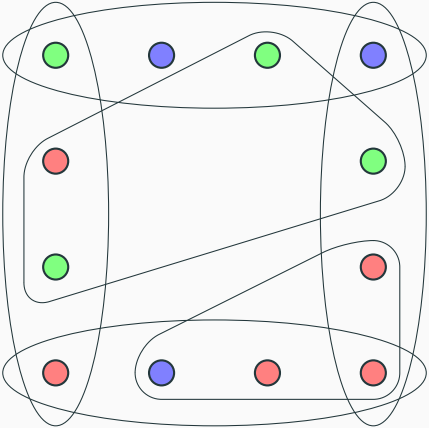
- Gamarnik, Katz (2012): $q \geqslant \alpha\Delta + \beta$ for large β and $\alpha \approx 2.84$;
- Lu, Yin (2013): $q \geqslant \alpha\Delta + 1$ for $\alpha \approx 2.58$.

Zeros of the chromatic polynomial:

- Patel, Regts (2017): $q > \alpha\Delta$ for $\alpha \approx 6.91$;
- Bencs, Davies, Patel, Regts (2019+): $q \geqslant e\Delta + 1$.

Cannot match randomised algorithms, yet.

What about hypergraphs?



A proper hypergraph colouring is one where no edge is monochromatic.

Previous results

For k-uniform hypergraphs, Bordewich, Dyer, and Karpinski (2006) show that Glauber dynamics is rapidly mixing if

$$k \geqslant 4 \text{ and } q > \Delta \qquad \text{or} \qquad k = 3 \text{ and } q > 1.5\Delta.$$

However, Lovász local lemma implies the existence of a proper colouring if $q > (ek\Delta)^{1/(k-1)}$.

Frieze and Melsted (2011) gave examples where $q \ll \Delta$, and there exists a colouring so that no move is possible ("frozen").

Frieze and Anastos (2017) showed that Glauber dynamics still converges rapidly if the hypergraph is simple and $q > \max\{C_k \log n, 500k^3\Delta^{1/(k-1)}\}$.

(Simple: every two hyperedges intersect in at most one vertex.)

Previous results

For k-uniform hypergraphs, Bordewich, Dyer, and Karpinski (2006) show that Glauber dynamics is rapidly mixing if

$$k \geqslant 4 \text{ and } q > \Delta \qquad \text{or} \qquad k = 3 \text{ and } q > 1.5\Delta.$$

However, Lovász local lemma implies the existence of a proper colouring if $q > (ek\Delta)^{1/(k-1)}$.

Frieze and Melsted (2011) gave examples where $q \ll \Delta$, and there exists a colouring so that no move is possible ("frozen").

Frieze and Anastos (2017) showed that Glauber dynamics still converges rapidly if the hypergraph is simple and $q > \max\{C_k \log n, 500k^3\Delta^{1/(k-1)}\}$.

(Simple: every two hyperedges intersect in at most one vertex.)

Previous results

For k-uniform hypergraphs, Bordewich, Dyer, and Karpinski (2006) show that Glauber dynamics is rapidly mixing if

$$k \geqslant 4 \text{ and } q > \Delta \qquad \text{or} \qquad k = 3 \text{ and } q > 1.5\Delta.$$

However, Lovász local lemma implies the existence of a proper colouring if $q > (ek\Delta)^{1/(k-1)}$.

Frieze and Melsted (2011) gave examples where $q \ll \Delta$, and there exists a colouring so that no move is possible ("frozen").

Frieze and Anastos (2017) showed that Glauber dynamics still converges rapidly if the hypergraph is simple and $q > \max\{C_k \log n, 500k^3\Delta^{1/(k-1)}\}$.

(Simple: every two hyperedges intersect in at most one vertex.)

Previous results

For k-uniform hypergraphs, Bordewich, Dyer, and Karpinski (2006) show that Glauber dynamics is rapidly mixing if

$$k \geqslant 4 \text{ and } q > \Delta \qquad \text{or} \qquad k = 3 \text{ and } q > 1.5\Delta.$$

However, Lovász local lemma implies the existence of a proper colouring if $q > (ek\Delta)^{1/(k-1)}$.

Frieze and Melsted (2011) gave examples where $q \ll \Delta$, and there exists a colouring so that no move is possible ("frozen").

Frieze and Anastos (2017) showed that Glauber dynamics still converges rapidly if the hypergraph is simple and $q > \max\{C_k \log n, 500k^3\Delta^{1/(k-1)}\}$.

(Simple: every two hyperedges intersect in at most one vertex.)

Our results

Theorem

For $\Delta \geqslant 2$, $k \geqslant 28$, and $q > 315\Delta^{\frac{14}{k-14}}$, there is an FPTAS for the number of q-colourings in k-uniform hypergraphs with maximum degree Δ.

Theorem

For $\Delta \geqslant 2$, $k \geqslant 28$, and $q > 798\Delta^{\frac{16}{k-16/3}}$, there is also an almost-uniform polynomial-time sampler.

Our approach is a modified version of Moitra (2017) based on the Lovász local lemma. The original approach in this setting would require an extra condition of the form $k > C \log \Delta$.

Our results

Theorem

For $\Delta \geqslant 2$, $k \geqslant 28$, and $q > 315\Delta^{\frac{14}{k-14}}$, there is an FPTAS for the number of q-colourings in k-uniform hypergraphs with maximum degree Δ.

Theorem

For $\Delta \geqslant 2$, $k \geqslant 28$, and $q > 798\Delta^{\frac{16}{k-16/3}}$, there is also an almost-uniform polynomial-time sampler.

Our approach is a modified version of Moitra (2017) based on the Lovász local lemma. The original approach in this setting would require an extra condition of the form $k > C \log \Delta$.

Our results

Theorem

For $\Delta \geqslant 2$, $k \geqslant 28$, *and* $q > 315\Delta^{\frac{14}{k-14}}$, *there is an FPTAS for the number of* q*-colourings in* k*-uniform hypergraphs with maximum degree* Δ.

Theorem

For $\Delta \geqslant 2$, $k \geqslant 28$, *and* $q > 798\Delta^{\frac{16}{k-16/3}}$, *there is also an almost-uniform polynomial-time sampler.*

Our approach is a modified version of Moitra (2017) based on the Lovász local lemma. The original approach in this setting would require an extra condition of the form $k > C \log \Delta$.

Lovász local lemma

(and how it helps with approximate counting)

Lovász local lemma

The original local lemma (Erdős and Lovász 75) was introduced to show the existence of 3-colourings in hypergraphs.

Let $H = (V, \mathcal{E})$ be the hypergraph, and $\Gamma(e)$ be the set of hyperedges intersecting $e \in \mathcal{E}$. Then $|\Gamma(e)| \leqslant (\Delta - 1)k$.

Theorem (Lovász '77)

If there exists an assignment $x : \mathcal{E} \to (0, 1)$ such that for every $e \in \mathcal{E}$ we have

$$\Pr(e \text{ is monochromatic}) \leqslant x(e) \prod_{e' \in \Gamma(e)} (1 - x(e')), \quad (1)$$

then a proper colouring exists.

Typically we set $x(e) = \frac{1}{k\Delta}$. It gives

$$x(e) \prod_{e' \in \Gamma(e)} (1 - x(e')) \geqslant \frac{1}{k\Delta} \left(1 - \frac{1}{k\Delta}\right)^{k(\Delta-1)} \geqslant \frac{1}{ek\Delta}. \quad (2)$$

Notice that $\Pr(e \text{ is monochromatic}) = \frac{q}{q^k} = \frac{1}{q^{k-1}}$.

Thus $q^{k-1} \geqslant ek\Delta$, or equivalently $q \geqslant (ek\Delta)^{\frac{1}{k-1}}$ suffices.

Lovász local lemma

The original local lemma (Erdős and Lovász 75) was introduced to show the existence of 3-colourings in hypergraphs.

Let $H = (V, \mathcal{E})$ be the hypergraph, and $\Gamma(e)$ be the set of hyperedges intersecting $e \in \mathcal{E}$. Then $|\Gamma(e)| \leqslant (\Delta - 1)k$.

Theorem (Lovász '77)

If there exists an assignment $x : \mathcal{E} \to (0, 1)$ such that for every $e \in \mathcal{E}$ we have

$$\Pr(e \text{ is monochromatic}) \leqslant x(e) \prod_{e' \in \Gamma(e)} (1 - x(e')), \tag{1}$$

then a proper colouring exists.

Typically we set $x(e) = \frac{1}{k\Delta}$. It gives

$$x(e) \prod_{e' \in \Gamma(e)} (1 - x(e')) \geqslant \frac{1}{k\Delta} \left(1 - \frac{1}{k\Delta}\right)^{k(\Delta-1)} \geqslant \frac{1}{ek\Delta}. \tag{2}$$

Notice that $\Pr(e \text{ is monochromatic}) = \frac{q}{q^k} = \frac{1}{q^{k-1}}$.

Thus $q^{k-1} \geqslant ek\Delta$, or equivalently $q \geqslant (ek\Delta)^{\frac{1}{k-1}}$ suffices.

Lovász local lemma

The original local lemma (Erdős and Lovász 75) was introduced to show the existence of 3-colourings in hypergraphs.

Let $H = (V, \mathcal{E})$ be the hypergraph, and $\Gamma(e)$ be the set of hyperedges intersecting $e \in \mathcal{E}$. Then $|\Gamma(e)| \leqslant (\Delta - 1)k$.

Theorem (Lovász '77)

If there exists an assignment $x : \mathcal{E} \to (0, 1)$ such that for every $e \in \mathcal{E}$ we have

$$\Pr(e \textit{ is monochromatic}) \leqslant x(e) \prod_{e' \in \Gamma(e)} (1 - x(e')), \tag{1}$$

then a proper colouring exists.

Typically we set $x(e) = \frac{1}{k\Delta}$. It gives

$$x(e) \prod_{e' \in \Gamma(e)} (1 - x(e')) \geqslant \frac{1}{k\Delta}\left(1 - \frac{1}{k\Delta}\right)^{k(\Delta-1)} \geqslant \frac{1}{ek\Delta}. \tag{2}$$

Notice that $\Pr(e \text{ is monochromatic}) = \frac{q}{q^k} = \frac{1}{q^{k-1}}$.

Thus $q^{k-1} \geqslant ek\Delta$, or equivalently $q \geqslant (ek\Delta)^{\frac{1}{k-1}}$ suffices.

Lovász local lemma

The original local lemma (Erdős and Lovász 75) was introduced to show the existence of 3-colourings in hypergraphs.

Let $H = (V, \mathcal{E})$ be the hypergraph, and $\Gamma(e)$ be the set of hyperedges intersecting $e \in \mathcal{E}$. Then $|\Gamma(e)| \leqslant (\Delta - 1)k$.

Theorem (Lovász '77)

If there exists an assignment $x : \mathcal{E} \to (0, 1)$ *such that for every* $e \in \mathcal{E}$ *we have*

$$\Pr(e \textit{ is monochromatic}) \leqslant x(e) \prod_{e' \in \Gamma(e)} (1 - x(e')), \tag{1}$$

then a proper colouring exists.

Typically we set $x(e) = \frac{1}{k\Delta}$. It gives

$$x(e) \prod_{e' \in \Gamma(e)} (1 - x(e')) \geqslant \frac{1}{k\Delta} \left(1 - \frac{1}{k\Delta}\right)^{k(\Delta - 1)} \geqslant \frac{1}{ek\Delta}. \tag{2}$$

Notice that $\Pr(e \text{ is monochromatic}) = \frac{q}{q^k} = \frac{1}{q^{k-1}}$.

Thus $q^{k-1} \geqslant ek\Delta$, or equivalently $q \geqslant (ek\Delta)^{\frac{1}{k-1}}$ suffices.

Lovász local lemma

The original local lemma (Erdős and Lovász 75) was introduced to show the existence of 3-colourings in hypergraphs.

Let $H = (V, \mathcal{E})$ be the hypergraph, and $\Gamma(e)$ be the set of hyperedges intersecting $e \in \mathcal{E}$. Then $|\Gamma(e)| \leqslant (\Delta - 1)k$.

Theorem (Lovász '77)

If there exists an assignment $x : \mathcal{E} \to (0, 1)$ such that for every $e \in \mathcal{E}$ we have

$$\Pr(e \textit{ is monochromatic}) \leqslant x(e) \prod_{e' \in \Gamma(e)} (1 - x(e')), \qquad (1)$$

then a proper colouring exists.

Typically we set $x(e) = \frac{1}{k\Delta}$. It gives

$$x(e) \prod_{e' \in \Gamma(e)} (1 - x(e')) \geqslant \frac{1}{k\Delta} \left(1 - \frac{1}{k\Delta}\right)^{k(\Delta-1)} \geqslant \frac{1}{ek\Delta}. \qquad (2)$$

Notice that $\Pr(e \text{ is monochromatic}) = \frac{q}{q^k} = \frac{1}{q^{k-1}}$.

Thus $q^{k-1} \geqslant ek\Delta$, or equivalently $q \geqslant (ek\Delta)^{\frac{1}{k-1}}$ suffices.

Lovász local lemma

The original local lemma (Erdős and Lovász 75) was introduced to show the existence of 3-colourings in hypergraphs.

Let $H = (V, \mathcal{E})$ be the hypergraph, and $\Gamma(e)$ be the set of hyperedges intersecting $e \in \mathcal{E}$. Then $|\Gamma(e)| \leqslant (\Delta - 1)k$.

Theorem (Lovász '77)

If there exists an assignment $x : \mathcal{E} \to (0, 1)$ *such that for every* $e \in \mathcal{E}$ *we have*

$$\Pr(e \textit{ is monochromatic}) \leqslant x(e) \prod_{e' \in \Gamma(e)} (1 - x(e')), \tag{1}$$

then a proper colouring exists.

Typically we set $x(e) = \frac{1}{k\Delta}$. It gives

$$x(e) \prod_{e' \in \Gamma(e)} (1 - x(e')) \geqslant \frac{1}{k\Delta} \left(1 - \frac{1}{k\Delta}\right)^{k(\Delta - 1)} \geqslant \frac{1}{ek\Delta}. \tag{2}$$

Notice that $\Pr(e \text{ is monochromatic}) = \frac{q}{q^k} = \frac{1}{q^{k-1}}$.

Thus $q^{k-1} \geqslant ek\Delta$, or equivalently $q \geqslant (ek\Delta)^{\frac{1}{k-1}}$ suffices.

Reducing to computing marginal probabilities

For approximate counting, we use the (algorithmic) local lemma to find a partial colouring τ so that every hyperedge is satisfied by the first k_1 vertices.

(This will succeed as long as $q > (ek_1\Delta)^{\frac{1}{k_1-1}}$. k_1 will eventually be set to $\frac{k}{14}$.)

Then we compute the probability of τ by "pinning" vertices one by one.

Let $U = \{u_1, \dots, u_r\}$ be the support of τ, and $\mu(\cdot)$ be the Gibbs (uniform) distribution on all proper colourings.

$$\frac{q^{n-r}}{|\mathcal{C}|} = \Pr_{\sigma\sim\mu}(\sigma \models \tau)$$

Reducing to computing marginal probabilities

For approximate counting, we use the (algorithmic) local lemma to find a partial colouring τ so that every hyperedge is satisfied by the first k_1 vertices.

(This will succeed as long as $q > (ek_1\Delta)^{\frac{1}{k_1-1}}$. k_1 will eventually be set to $\frac{k}{14}$.)

Then we compute the probability of τ by "pinning" vertices one by one.

Let $U = \{u_1, \ldots, u_r\}$ be the support of τ, and $\mu(\cdot)$ be the Gibbs (uniform) distribution on all proper colourings.

$$\begin{aligned} \frac{q^{n-r}}{|\mathcal{C}|} &= \Pr_{\sigma\sim\mu}(\sigma \models \tau) \\ &= \Pr_{\sigma\sim\mu}(\forall u \in U,\ \sigma(u) = \tau(u)) \end{aligned}$$

Reducing to computing marginal probabilities

For approximate counting, we use the (algorithmic) local lemma to find a partial colouring τ so that every hyperedge is satisfied by the first k_1 vertices.

(This will succeed as long as $q > (ek_1\Delta)^{\frac{1}{k_1-1}}$. k_1 will eventually be set to $\frac{k}{14}$.)

Then we compute the probability of τ by "pinning" vertices one by one.

Let $U = \{u_1, \ldots, u_r\}$ be the support of τ, and $\mu(\cdot)$ be the Gibbs (uniform) distribution on all proper colourings.

$$\begin{aligned}\frac{q^{n-r}}{|\mathcal{C}|} &= \Pr_{\sigma\sim\mu}(\sigma \models \tau)\\ &= \Pr_{\sigma\sim\mu}(\forall u \in U,\ \sigma(u) = \tau(u))\\ &= \prod_{i=1}^{r} \Pr_{\sigma\sim\mu}(\sigma(u_i) = \tau(u_i) \mid \forall j < i, \sigma(u_j) = \tau(u_j))\end{aligned}$$

Reducing to computing marginal probabilities

For approximate counting, we use the (algorithmic) local lemma to find a partial colouring τ so that every hyperedge is satisfied by the first k_1 vertices.

(This will succeed as long as $q > (ek_1\Delta)^{\frac{1}{k_1-1}}$. k_1 will eventually be set to $\frac{k}{14}$.)

Then we compute the probability of τ by "pinning" vertices one by one.

Let $U = \{u_1, \ldots, u_r\}$ be the support of τ, and $\mu(\cdot)$ be the Gibbs (uniform) distribution on all proper colourings.

$$\begin{aligned}
\frac{q^{n-r}}{|\mathcal{C}|} &= \Pr_{\sigma\sim\mu}(\sigma \models \tau) \\
&= \Pr_{\sigma\sim\mu}(\forall u \in U,\ \sigma(u) = \tau(u)) \\
&= \prod_{i=1}^{r} \Pr_{\sigma\sim\mu}(\sigma(u_i) = \tau(u_i) \mid \forall j < i, \sigma(u_j) = \tau(u_j))
\end{aligned}$$

Thus the key is to estimate marginal probabilities under partial colourings (up to $1 \pm \frac{\varepsilon}{n}$ error), where at least $k - k_1$ vertices are uncoloured in every edge.

Local lemma controls the conditional distribution

Let $\mathcal{C}$ be the set of all proper colourings.

Let $\mu(\cdot)$ be the Gibbs (uniform) distribution on all proper colourings (namely the product distribution conditioned on no monochromatic edge).

The local lemma also gives an upper bound for any event under $\mu(\cdot)$.

Theorem (Haeupler, Saha, and Srinivasan '11)
If LLL condition (1) holds for every $e \in \mathcal{E}$, then for any event B, it holds that

$$\mu(B) \leqslant \Pr(B) \prod_{e \in \Gamma(B)} (1 - x(e))^{-1}.$$

Local lemma controls the conditional distribution

Let $\mathcal{C}$ be the set of all proper colourings.

Let $\mu(\cdot)$ be the Gibbs (uniform) distribution on all proper colourings

(namely the product distribution conditioned on no monochromatic edge).

The local lemma also gives an upper bound for any event under $\mu(\cdot)$.

Theorem (Haeupler, Saha, and Srinivasan '11)
If LLL condition (1) holds for every $e \in \mathcal{E}$, then for any event B, it holds that

$$\mu(B) \leqslant \Pr(B) \prod_{e \in \Gamma(B)} (1 - x(e))^{-1}.$$

Local uniformity

Lemma

If $\forall e \in \mathcal{E}, k' \leqslant |e| \leqslant k, t \geqslant k$ *and* $q \geqslant (et\Delta)^{\frac{1}{k'-1}}$*, then for any* $v \in V$ *and any colour* $c \in [q]$,

$$\frac{1}{q}\left(1 - \frac{1}{t}\right) \leqslant \Pr_{\sigma \sim \mu}(\sigma(v) = c) \leqslant \frac{1}{q}\left(1 + \frac{4}{t}\right).$$

The upper bound comes from a direct application.

The lower bound is obtained by giving upper bounds for “blocking cases”.

Local uniformity

Lemma

If $\forall e \in \mathcal{E}, k' \leqslant |e| \leqslant k, t \geqslant k$ *and* $q \geqslant (et\Delta)^{\frac{1}{k'-1}}$*, then for any* $v \in V$ *and any colour* $c \in [q]$,

$$\frac{1}{q}\left(1 - \frac{1}{t}\right) \leqslant \Pr_{\sigma \sim \mu}(\sigma(v) = c) \leqslant \frac{1}{q}\left(1 + \frac{4}{t}\right).$$

The upper bound comes from a direct application.

The lower bound is obtained by giving upper bounds for “blocking cases”.

Local uniformity

Lemma

If $\forall e \in \mathcal{E}, k' \leqslant |e| \leqslant k, t \geqslant k$ *and* $q \geqslant (et\Delta)^{\frac{1}{k'-1}}$*, then for any* $v \in V$ *and any colour* $c \in [q]$,

$$\frac{1}{q}\left(1 - \frac{1}{t}\right) \leqslant \Pr_{\sigma \sim \mu}(\sigma(v) = c) \leqslant \frac{1}{q}\left(1 + \frac{4}{t}\right).$$

We use this lemma with $t \approx \Delta^C$ at various places with various C. Recall that our assumption is of the form $q \geqslant C_1 \Delta^{\frac{C_2}{k-C_3}}$.

Under μ, all vertices are very close to uniform.

We use this lemma when some vertices are already coloured, namely for μ conditioned on a partial colouring.

The quantity k' is the minimum number of uncoloured vertices among all unsatisfied hyperedges (namely $k' = k - k_1$).

A good start, but not enough. The goal is $\left(1 \pm \frac{\varepsilon}{n}\right)$-approximation of the marginals.

Local uniformity

Lemma

If $\forall e \in \mathcal{E}, k' \leqslant |e| \leqslant k, t \geqslant k$ *and* $q \geqslant (et\Delta)^{\frac{1}{k'-1}}$, *then for any* $v \in V$ *and any colour* $c \in [q]$,

$$\frac{1}{q}\left(1-\frac{1}{t}\right) \leqslant \Pr_{\sigma\sim\mu}(\sigma(v)=c) \leqslant \frac{1}{q}\left(1+\frac{4}{t}\right).$$

We use this lemma with $t \approx \Delta^{C}$ at various places with various C. Recall that our assumption is of the form $q \geqslant C_1\Delta^{\frac{C_2}{k-C_3}}$.

Under μ, all vertices are very close to uniform.

We use this lemma when some vertices are already coloured, namely for μ conditioned on a partial colouring.

The quantity k' is the minimum number of uncoloured vertices among all unsatisfied hyperedges (namely $k' = k - k_1$).

A good start, but not enough. The goal is $\left(1 \pm \frac{\varepsilon}{n}\right)$-approximation of the marginals.

Local uniformity

Lemma

If $\forall e \in \mathcal{E}, k' \leqslant |e| \leqslant k, t \geqslant k$ *and* $q \geqslant (et\Delta)^{\frac{1}{k'-1}}$, *then for any* $v \in V$ *and any colour* $c \in [q]$,

$$\frac{1}{q}\left(1 - \frac{1}{t}\right) \leqslant \Pr_{\sigma \sim \mu}(\sigma(v) = c) \leqslant \frac{1}{q}\left(1 + \frac{4}{t}\right).$$

We use this lemma with $t \approx \Delta^C$ at various places with various C. Recall that our assumption is of the form $q \geqslant C_1 \Delta^{\frac{C_2}{k - C_3}}$.

Under μ, all vertices are ~~very close to~~ uniform.
We use this lemma when some vertices are already coloured, namely for μ conditioned on a partial colouring.
The quantity k' is the minimum number of uncoloured vertices among all unsatisfied hyperedges (namely $k' = k - k_1$).

A good start, but not enough. The goal is $\left(1 \pm \frac{\varepsilon}{n}\right)$-approximation of the marginals.

Local uniformity

Lemma

If $\forall e \in \mathcal{E}, k' \leqslant |e| \leqslant k, t \geqslant k$ *and* $q \geqslant (et\Delta)^{\frac{1}{k'-1}}$, *then for any* $v \in V$ *and any colour* $c \in [q]$,

$$\frac{1}{q}\left(1 - \frac{1}{t}\right) \leqslant \Pr_{\sigma\sim\mu}(\sigma(v) = c) \leqslant \frac{1}{q}\left(1 + \frac{4}{t}\right).$$

We use this lemma with $t \approx \Delta^C$ at various places with various C. Recall that our assumption is of the form $q \geqslant C_1\Delta^{\frac{C_2}{k-C_3}}$.

Under μ, all vertices are ~~very close to~~ uniform.

We use this lemma when some vertices are already coloured, namely for μ conditioned on a partial colouring.

The quantity k' is the minimum number of uncoloured vertices among all unsatisfied hyperedges (namely $k' = k - k_1$).

A good start, but not enough. The goal is $\left(1 \pm \frac{\varepsilon}{n}\right)$-approximation of the marginals.

Coupling

Say we want to compute the marginal probability of v.

Let $\mathcal{C}_i$ be the set of colourings where v is coloured i, and μ_i be uniform over $\mathcal{C}_i$. We want to couple μ_1 and μ_2.

Start: $V_1 = \{v\}$, $V_{col} = \{v\}$. Maintain $V_2 = V \setminus V_1$.

Body:

1. For any hyperedge e intersecting both V_1 and V_2, let u be its first vertex. Couple u maximally assuming its marginal probabilities are known.
2. Remove all hyperedges that are satisfied in both copies.
3. If an edge has k_2 vertices coloured, put all remaining vertices in V_1 (failed) and remove the edge.

Stop: all hyperedges intersecting V_1 are removed.

(The constant k_2 is eventually set to $\frac{3k}{7}$ for approximate counting.)

Coupling

Say we want to compute the marginal probability of v.

Let $\mathcal{C}_i$ be the set of colourings where v is coloured i, and μ_i be uniform over $\mathcal{C}_i$. We want to couple μ_1 and μ_2.

Start: $V_1 = \{v\}$, $V_{col} = \{v\}$. Maintain $V_2 = V \setminus V_1$.

Body:

1. For any hyperedge e intersecting both V_1 and V_2, let u be its first vertex. Couple u maximally assuming its marginal probabilities are known.
2. Remove all hyperedges that are satisfied in both copies.
3. If an edge has k_2 vertices coloured, put all remaining vertices in V_1 (failed) and remove the edge.

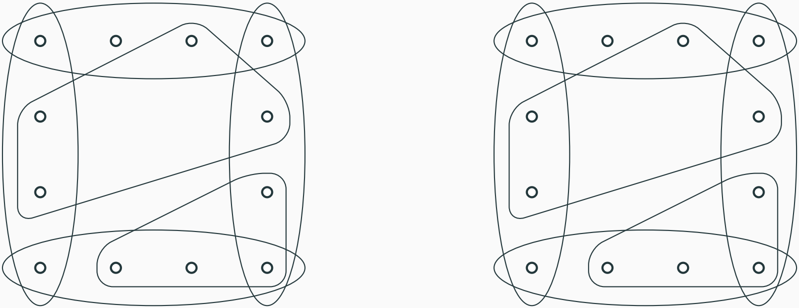
Stop: all hyperedges intersecting V_1 are removed.

(The constant k_2 is eventually set to $\frac{3k}{7}$ for approximate counting.)

Coupling - An example

V_1 : descrepency. V_{col} : coloured. $V_2 := V \setminus V_1$.

Stop: all hyperedges intersecting V_1 are removed.



At any time, there are at least $k' - k_2$ empty vertices in any hyperedge.

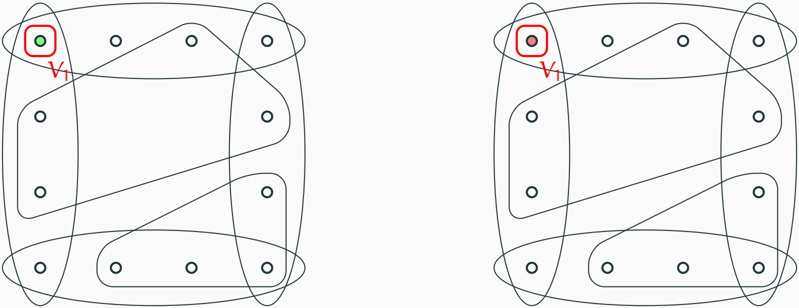
If $q > C\Delta^{\frac{3}{k'-k_2}}$, then the coupling stops in $O(\log n)$ steps with probability $1 - O\left(\frac{1}{n^c}\right)$.

Moitra (2017) marks what vertices to couple in advance, whereas our coupling is adaptive.

Coupling - An example

V_1 : descrepency. V_{col} : coloured. $V_2 := V \setminus V_1$.

Stop: all hyperedges intersecting V_1 are removed.



At any time, there are at least $k' - k_2$ empty vertices in any hyperedge.

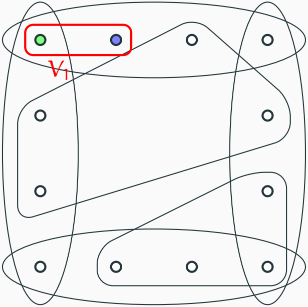
If $q > C\Delta^{\frac{3}{k'-k_2}}$, then the coupling stops in $O(\log n)$ steps with probability $1 - O\left(\frac{1}{n^c}\right)$.

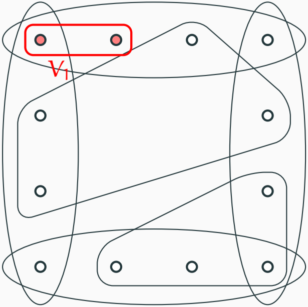
Moitra (2017) marks what vertices to couple in advance, whereas our coupling is adaptive.

Coupling - An example

V_1 : descrepency. V_{col} : coloured. $V_2 := V \setminus V_1$.

Stop: all hyperedges intersecting V_1 are removed.





At any time, there are at least $k' - k_2$ empty vertices in any hyperedge.

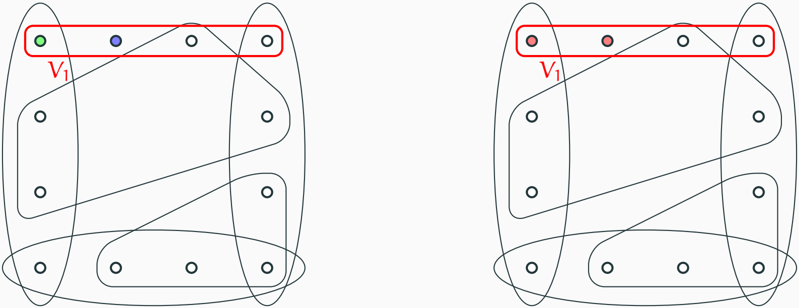
If $q > C\Delta^{\frac{3}{k'-k_2}}$, then the coupling stops in $O(\log n)$ steps with probability $1 - O\left(\frac{1}{n^c}\right)$.

Moitra (2017) marks what vertices to couple in advance, whereas our coupling is adaptive.

Coupling - An example

V_1 : descrepency. V_{col} : coloured. $V_2 := V \setminus V_1$.

Stop: all hyperedges intersecting V_1 are removed.



At any time, there are at least $k' - k_2$ empty vertices in any hyperedge.

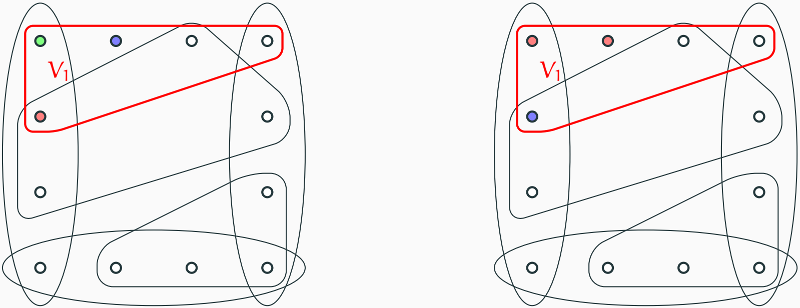
If $q > C\Delta^{\frac{3}{k'-k_2}}$, then the coupling stops in $O(\log n)$ steps with probability $1 - O\left(\frac{1}{n^c}\right)$.

Moitra (2017) marks what vertices to couple in advance, whereas our coupling is adaptive.

Coupling - An example

V_1 : descrepency. V_{col} : coloured. $V_2 := V \setminus V_1$.

Stop: all hyperedges intersecting V_1 are removed.



At any time, there are at least $k' - k_2$ empty vertices in any hyperedge.

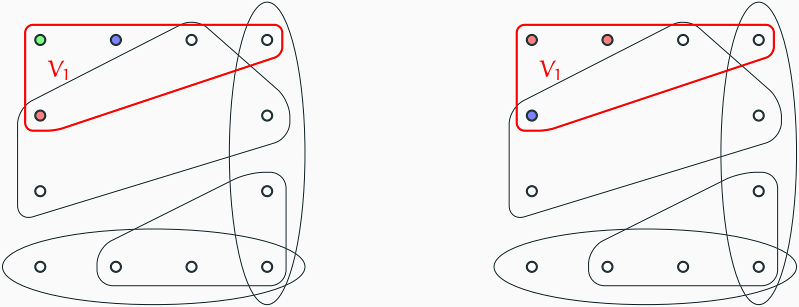
If $q > C\Delta^{\frac{3}{k'-k_2}}$, then the coupling stops in $O(\log n)$ steps with probability $1 - O\left(\frac{1}{n^c}\right)$.

Moitra (2017) marks what vertices to couple in advance, whereas our coupling is adaptive.

Coupling - An example

V_1 : descrepency. V_{col} : coloured. $V_2 := V \setminus V_1$.

Stop: all hyperedges intersecting V_1 are removed.



At any time, there are at least $k' - k_2$ empty vertices in any hyperedge.

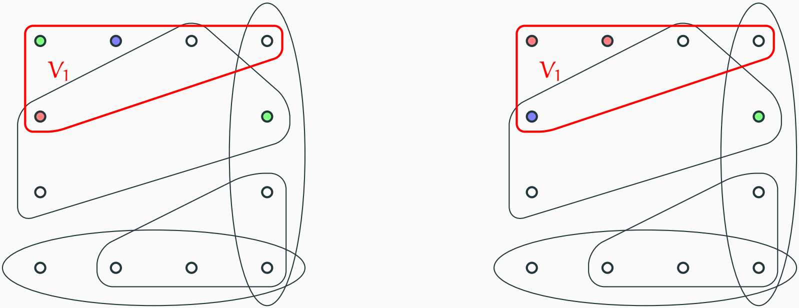
If $q > C\Delta^{\frac{3}{k'-k_2}}$, then the coupling stops in $O(\log n)$ steps with probability $1 - O\left(\frac{1}{n^c}\right)$.

Moitra (2017) marks what vertices to couple in advance, whereas our coupling is adaptive.

Coupling - An example

V_1 : descrepency. V_{col} : coloured. $V_2 := V \setminus V_1$.

Stop: all hyperedges intersecting V_1 are removed.



At any time, there are at least $k' - k_2$ empty vertices in any hyperedge.

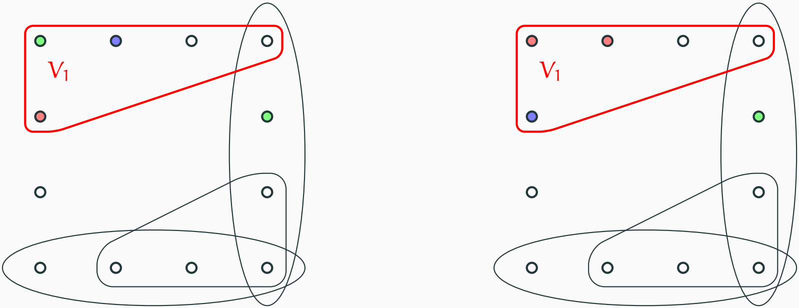
If $q > C\Delta^{\frac{3}{k'-k_2}}$, then the coupling stops in $O(\log n)$ steps with probability $1 - O\left(\frac{1}{n^c}\right)$.

Moitra (2017) marks what vertices to couple in advance, whereas our coupling is adaptive.

Coupling - An example

V_1 : descrepency. V_{col} : coloured. $V_2 := V \setminus V_1$.

Stop: all hyperedges intersecting V_1 are removed.



At any time, there are at least $k' - k_2$ empty vertices in any hyperedge.

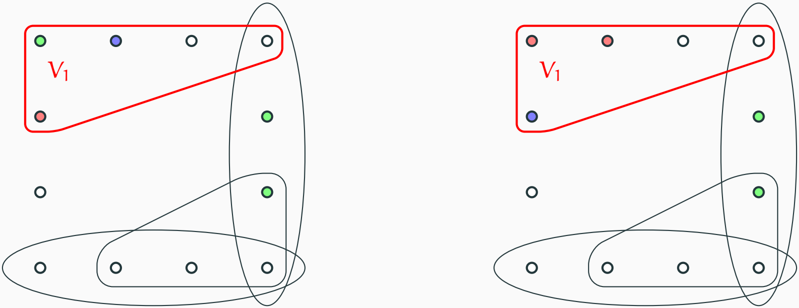
If $q > C\Delta^{\frac{3}{k'-k_2}}$, then the coupling stops in $O(\log n)$ steps with probability $1 - O\left(\frac{1}{n^c}\right)$.

Moitra (2017) marks what vertices to couple in advance, whereas our coupling is adaptive.

Coupling - An example

V_1 : descrepency. V_{col} : coloured. $V_2 := V \setminus V_1$.

Stop: all hyperedges intersecting V_1 are removed.



At any time, there are at least $k' - k_2$ empty vertices in any hyperedge.

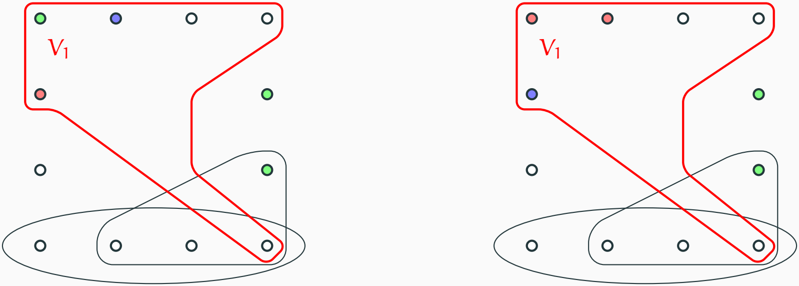
If $q > C\Delta^{\frac{3}{k'-k_2}}$, then the coupling stops in $O(\log n)$ steps with probability $1 - O\left(\frac{1}{n^c}\right)$.

Moitra (2017) marks what vertices to couple in advance, whereas our coupling is adaptive.

Coupling - An example

V_1 : descrepency. $\quad V_{col}$: coloured. $\quad V_2 := V \setminus V_1$.

Stop: all hyperedges intersecting V_1 are removed.



At any time, there are at least $k' - k_2$ empty vertices in any hyperedge.

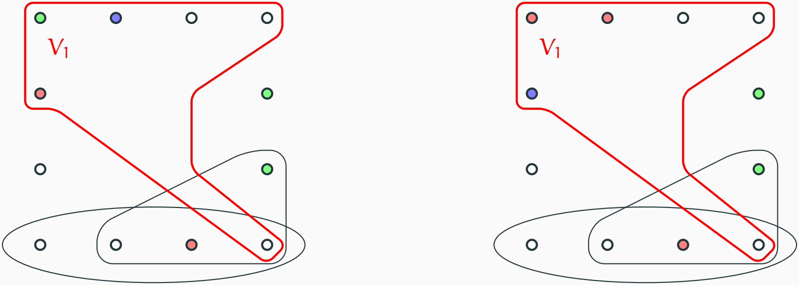
If $q > C\Delta^{\frac{3}{k'-k_2}}$, then the coupling stops in $O(\log n)$ steps with probability $1 - O\left(\frac{1}{n^c}\right)$.

Moitra (2017) marks what vertices to couple in advance, whereas our coupling is adaptive.

Coupling - An example

V_1 : descrepency. V_{col} : coloured. $V_2 := V \setminus V_1$.

Stop: all hyperedges intersecting V_1 are removed.



At any time, there are at least $k' - k_2$ empty vertices in any hyperedge.

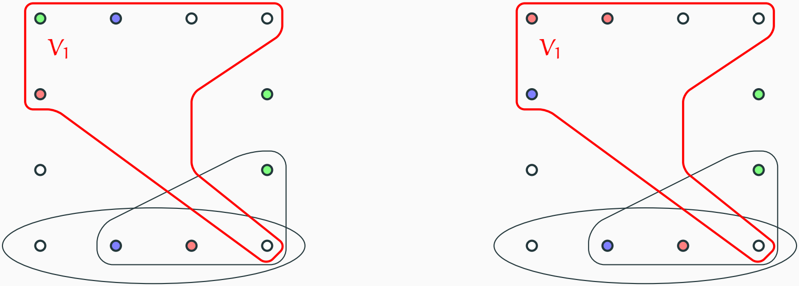
If $q > C\Delta^{\frac{3}{k'-k_2}}$, then the coupling stops in $O(\log n)$ steps with probability $1 - O\left(\frac{1}{n^c}\right)$.

Moitra (2017) marks what vertices to couple in advance, whereas our coupling is adaptive.

Coupling - An example

V_1 : descrepency. V_{col} : coloured. $V_2 := V \setminus V_1$.

Stop: all hyperedges intersecting V_1 are removed.



At any time, there are at least $k' - k_2$ empty vertices in any hyperedge.

If $q > C\Delta^{\frac{3}{k'-k_2}}$, then the coupling stops in $O(\log n)$ steps with probability $1 - O\left(\frac{1}{n^c}\right)$.

Moitra (2017) marks what vertices to couple in advance, whereas our coupling is adaptive.

Coupling - An example

V_1 : descrepency. V_{col} : coloured. $V_2 := V \setminus V_1$.

Stop: all hyperedges intersecting V_1 are removed.

At any time, there are at least $k' - k_2$ empty vertices in any hyperedge.

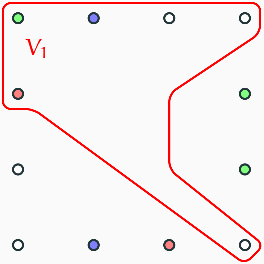
If $q > C\Delta^{\frac{3}{k'-k_2}}$, then the coupling stops in $O(\log n)$ steps with probability $1 - O\left(\frac{1}{n^c}\right)$.

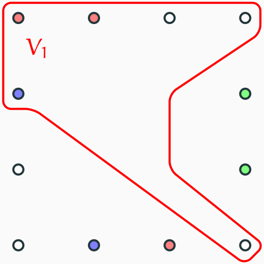
Moitra (2017) marks what vertices to couple in advance, whereas our coupling is adaptive.

Coupling - An example

V_1 : descrepency. V_{col} : coloured. $V_2 := V \setminus V_1$.

Stop: all hyperedges intersecting V_1 are removed.





At any time, there are at least $k' - k_2$ empty vertices in any hyperedge.

If $q > C\Delta^{\frac{3}{k'-k_2}}$, then the coupling stops in $O(\log n)$ steps with probability $1 - O\left(\frac{1}{n^c}\right)$.

Moitra (2017) marks what vertices to couple in advance, whereas our coupling is adaptive.

Coupling - An example

V_1 : descrepency. V_{col} : coloured. $V_2 := V \setminus V_1$.

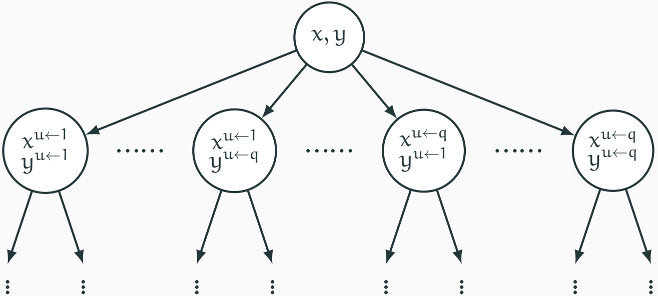
Stop: all hyperedges intersecting V_1 are removed.

At any time, there are at least $k' - k_2$ empty vertices in any hyperedge.

If $q > C\Delta^{\frac{3}{k'-k_2}}$, then the coupling stops in $O(\log n)$ steps with probability $1 - O\left(\frac{1}{n^c}\right)$.

Moitra (2017) marks what vertices to couple in advance, whereas our coupling is adaptive.

Coupling tree



Coupling tree $\mathcal{T}$: each node is a pair of partial colourings (x, y).

The children of (x, y) are all q^2 ways to extend them to the next vertex.

Linear program

We cannot really run the coupling. Instead, we set up a linear program. The variables are to mimic:

$$p^x_{x,y} = \frac{|\mathcal{C}_1|}{|\mathcal{C}_x|} \cdot \mu_{cp}(x, y),$$
$$p^y_{x,y} = \frac{|\mathcal{C}_2|}{|\mathcal{C}_y|} \cdot \mu_{cp}(x, y),$$

where $\mathcal{C}_i$ is the set of colourings s.t. $v \leftarrow i$ for $i = 1, 2$, and $\mathcal{C}_x$ (or $\mathcal{C}_y$) is the set of colourings consistent with x (or y).

Note that $\sum_y p^x_{x,y} = 1$ and thus $0 \leqslant p^x_{x,y}, p^y_{x,y} \leqslant 1$.

We can write down linear constraints for these variables.

Linear program

We cannot really run the coupling. Instead, we set up a linear program. The variables are to mimic:

$$p^x_{x,y} = \frac{|\mathcal{C}_1|}{|\mathcal{C}_x|} \cdot \mu_{cp}(x, y),$$
$$p^y_{x,y} = \frac{|\mathcal{C}_2|}{|\mathcal{C}_y|} \cdot \mu_{cp}(x, y),$$

where $\mathcal{C}_i$ is the set of colourings s.t. $v \leftarrow i$ for $i = 1, 2$, and $\mathcal{C}_x$ (or $\mathcal{C}_y$) is the set of colourings consistent with x (or y).

Note that $\sum_y p^x_{x,y} = 1$ and thus $0 \leqslant p^x_{x,y}, p^y_{x,y} \leqslant 1$.

We can write down linear constraints for these variables.

Linear program

We cannot really run the coupling. Instead, we set up a linear program. The variables are to mimic:

$$p^x_{x,y} = \frac{|\mathcal{C}_1|}{|\mathcal{C}_x|} \cdot \mu_{cp}(x, y),$$
$$p^y_{x,y} = \frac{|\mathcal{C}_2|}{|\mathcal{C}_y|} \cdot \mu_{cp}(x, y),$$

where $\mathcal{C}_i$ is the set of colourings s.t. $v \leftarrow i$ for $i = 1, 2$, and $\mathcal{C}_x$ (or $\mathcal{C}_y$) is the set of colourings consistent with x (or y).

Note that $\sum_y p^x_{x,y} = 1$ and thus $0 \leqslant p^x_{x,y}, p^y_{x,y} \leqslant 1$.

We can write down linear constraints for these variables.

Constraints 1

From the definition: $\frac{|\mathcal{C}_1|}{|\mathcal{C}_2|} = \frac{p^x_{x,y}}{p^y_{x,y}} \cdot \frac{|\mathcal{C}_x|}{|\mathcal{C}_y|}$.

If (x, y) is a leaf in $\mathcal{T}$, then we can compute $\frac{|\mathcal{C}_x|}{|\mathcal{C}_y|}$ in time $\exp(|V_1 \setminus V_{col}|)$.

Constraints 1: For every leaf (x, y), we have the constraints:

$$\underline{r} \leqslant \frac{p^x_{x,y}}{p^y_{x,y}} \cdot \frac{|\mathcal{C}_x|}{|\mathcal{C}_y|} \leqslant \overline{r}.$$

Here $\underline{r}$ and $\overline{r}$ are our guessed lower and upper bounds for $\frac{|\mathcal{C}_1|}{|\mathcal{C}_2|}$.

Constraints 1

From the definition: $\frac{|\mathcal{C}_1|}{|\mathcal{C}_2|} = \frac{p^x_{x,y}}{p^y_{x,y}} \cdot \frac{|\mathcal{C}_x|}{|\mathcal{C}_y|}$.

If (x, y) is a leaf in $\mathcal{T}$, then we can compute $\frac{|\mathcal{C}_x|}{|\mathcal{C}_y|}$ in time $\exp(|V_1 \setminus V_{col}|)$.

Constraints 1: For every leaf (x, y), we have the constraints:

$$\underline{r} \leqslant \frac{p^x_{x,y}}{p^y_{x,y}} \cdot \frac{|\mathcal{C}_x|}{|\mathcal{C}_y|} \leqslant \overline{r}.$$

Here $\underline{r}$ and $\overline{r}$ are our guessed lower and upper bounds for $\frac{|\mathcal{C}_1|}{|\mathcal{C}_2|}$.

Constraints 1

From the definition: $\frac{|\mathcal{C}_1|}{|\mathcal{C}_2|} = \frac{p^x_{x,y}}{p^y_{x,y}} \cdot \frac{|\mathcal{C}_x|}{|\mathcal{C}_y|}$.

If (x, y) is a leaf in $\mathcal{T}$, then we can compute $\frac{|\mathcal{C}_x|}{|\mathcal{C}_y|}$ in time $\exp(|V_1 \setminus V_{col}|)$.

Constraints 1: For every leaf (x, y), we have the constraints:

$$\underline{r} \leqslant \frac{p^x_{x,y}}{p^y_{x,y}} \cdot \frac{|\mathcal{C}_x|}{|\mathcal{C}_y|} \leqslant \bar{r}.$$

Here $\underline{r}$ and $\bar{r}$ are our guessed lower and upper bounds for $\frac{|\mathcal{C}_1|}{|\mathcal{C}_2|}$.

Constraints 2

Constraints 2: For the root $(x_0, y_0) \in \mathcal{T}$, we have

$$p^{x_0}_{x_0,y_0} = p^{y_0}_{x_0,y_0} = 1.$$

Moreover, for every non-leaf $(x, y) \in \mathcal{T}$, let u be the next vertex to couple. For every $c \in [q]$,

$$\sum_{c' \in [q]} p^{x^{u\leftarrow c}}_{x^{u\leftarrow c}, y^{u\leftarrow c'}} = \frac{|\mathcal{C}_1|}{|\mathcal{C}_{x^{u\leftarrow c}}|} \cdot \frac{|\mathcal{C}_{x^{u\leftarrow c}}|}{|\mathcal{C}_x|} \cdot \mu_{cp}(x, y) = p^x_{x,y};$$

$$\sum_{c' \in [q]} p^{y^{u\leftarrow c}}_{x^{u\leftarrow c'}, y^{u\leftarrow c}} = \frac{|\mathcal{C}_2|}{|\mathcal{C}_{y^{u\leftarrow c}}|} \cdot \frac{|\mathcal{C}_{y^{u\leftarrow c}}|}{|\mathcal{C}_y|} \cdot \mu_{cp}(x, y) = p^y_{x,y}.$$

Recover the marginals

Due to **Constraints 2**, a simple induction shows that for every $\sigma \in \mathcal{C}_1$,

$$\sum_{(x,y)\in\mathcal{L}(\mathcal{T}):\ \sigma\models x} p_{x,y}^{x} = 1.$$

Rewrite $|\mathcal{C}_1|$:

$$\begin{aligned}|\mathcal{C}_1| &= \sum_{\sigma\in\mathcal{C}_1} 1 = \sum_{\sigma\in\mathcal{C}_1}\ \sum_{(x,y)\in\mathcal{L}(\mathcal{T}):\sigma\models x} p_{x,y}^{x} \\ &= \sum_{(x,y)\in\mathcal{L}(\mathcal{T})}\ \sum_{\sigma\models x} p_{x,y}^{x} \\ &= \sum_{(x,y)\in\mathcal{L}(\mathcal{T})} p_{x,y}^{x}\,|C_x|.\end{aligned}$$

Similar equalities hold on the y side, implying:

$$\frac{|\mathcal{C}_1|}{|\mathcal{C}_2|} = \frac{\sum_{(x,y)\in\mathcal{L}(\mathcal{T})} p_{x,y}^{x}\,|C_x|}{\sum_{(x,y)\in\mathcal{L}(\mathcal{T})} p_{x,y}^{y}\,|C_y|}.$$

Recover the marginals (cont.)

$$\frac{|\mathcal{C}_1|}{|\mathcal{C}_2|} = \frac{\sum_{(x,y)\in\mathcal{L}(\mathcal{T})} p^x_{x,y} |C_x|}{\sum_{(x,y)\in\mathcal{L}(\mathcal{T})} p^y_{x,y} |C_y|}$$

Recall **Constraints 1**. For any $(x, y) \in \mathcal{L}(\mathcal{T})$,

$$\underline{r} \leqslant \frac{p^x_{x,y} |C_x|}{p^y_{x,y} |C_y|} \leqslant \overline{r}.$$

It implies that

$$\underline{r} \leqslant \frac{|\mathcal{C}_1|}{|\mathcal{C}_2|} \leqslant \overline{r}.$$

Constraints 3

Unfortunately, the whole linear program is exponentially large. The saving grace is that the coupling stops at $O(\log n)$ size whp.

If we truncate at $O(\log n)$ levels, the error should be small, due to local uniformity.

Constraints 3: For every $c, c' \in [q]$ that $c \neq c'$:

$$p^{x^{u \leftarrow c}}_{x^{u \leftarrow c}, y^{u \leftarrow c'}} \leqslant \frac{5}{t} \cdot p^{x}_{x,y};$$
$$p^{y^{u \leftarrow c'}}_{x^{u \leftarrow c}, y^{u \leftarrow c'}} \leqslant \frac{5}{t} \cdot p^{y}_{x,y}.$$

The quantity t will eventually be set as $C(k\Delta)^6$.

Constraints 3

Unfortunately, the whole linear program is exponentially large. The saving grace is that the coupling stops at $O(\log n)$ size whp.

If we truncate at $O(\log n)$ levels, the error should be small, due to local uniformity.

Constraints 3: For every $c, c' \in [q]$ that $c \neq c'$:

$$p^{x^{u\leftarrow c}}_{x^{u\leftarrow c}, y^{u\leftarrow c'}} \leqslant \frac{5}{t} \cdot p^{x}_{x,y};$$
$$p^{y^{u\leftarrow c'}}_{x^{u\leftarrow c}, y^{u\leftarrow c'}} \leqslant \frac{5}{t} \cdot p^{y}_{x,y}.$$

The quantity t will eventually be set as $C(k\Delta)^6$.

Truncation error

Recall that

$$|\mathcal{C}_1| = \sum_{\sigma \in \mathcal{C}_1} \sum_{(x,y) \in \mathcal{L}(\mathcal{T}) : \sigma \models x} p^x_{x,y}.$$

The truncation error from a particular $\sigma \in \mathcal{C}_1$ comes from conditioned on outputing σ, the coupling lasts too long.

Such "bad" colourings do exist (all early vertices are monochromatic).

We prove two things:

1. The fraction of "bad" colourings is small;
2. For every "good" colouring, the truncation error is small because of **Constraints 3**.

Truncation error

Recall that

$$|\mathcal{C}_1| = \sum_{\sigma \in \mathcal{C}_1} \sum_{(x,y) \in \mathcal{L}(\mathcal{T}) : \sigma \models x} p^{x}_{x,y}.$$

The truncation error from a particular $\sigma \in \mathcal{C}_1$ comes from conditioned on outputing σ, the coupling lasts too long.

Such "bad" colourings do exist (all early vertices are monochromatic).

We prove two things:

1. The fraction of "bad" colourings is small;
2. For every "good" colouring, the truncation error is small because of **Constraints 3**.

Truncation error

Recall that

$$|\mathcal{C}_1| = \sum_{\sigma \in \mathcal{C}_1} \sum_{(x,y) \in \mathcal{L}(\mathcal{T}) : \sigma \models x} p^x_{x,y}.$$

The truncation error from a particular $\sigma \in \mathcal{C}_1$ comes from conditioned on outputing σ, the coupling lasts too long.

Such "bad" colourings do exist (all early vertices are monochromatic).

We prove two things:

1. The fraction of "bad" colourings is small;
2. For every "good" colouring, the truncation error is small because of **Constraints 3**.

Bound the error

A "bad" colouring must fail many hyperedges during the coupling, but we couple k_2 vertices of every hyperedge.

Thus its fraction is small if k_2 is sufficiently large.

The error allowed by **Constraints 3** is controlled by the number of empty vertices in the coupling process, namely the quantity $k' - k_2$.

The larger $k' - k_2$, the more uniform all vertices are and the smaller coupling errors.

We solve an optimization problem to get the best k_2 balancing the two points above.

Bound the error

A “bad” colouring must fail many hyperedges during the coupling, but we couple k_2 vertices of every hyperedge.

Thus its fraction is small if k_2 is sufficiently large.

The error allowed by **Constraints 3** is controlled by the number of empty vertices in the coupling process, namely the quantity $k' - k_2$.

The larger $k' - k_2$, the more uniform all vertices are and the smaller coupling errors.

We solve an optimization problem to get the best k_2 balancing the two points above.

Bound the error

A “bad” colouring must fail many hyperedges during the coupling, but we couple k_2 vertices of every hyperedge.

Thus its fraction is small if k_2 is sufficiently large.

The error allowed by **Constraints 3** is controlled by the number of empty vertices in the coupling process, namely the quantity $k' - k_2$.

The larger $k' - k_2$, the more uniform all vertices are and the smaller coupling errors.

We solve an optimization problem to get the best k_2 balancing the two points above.

Counting and sampling

So far we are calculating the marginal probability, which requires that there are sufficiently many uncoloured vertices in all hyperedges.

- For approximate counting, we use the local lemma to find a partial colouring so that every hyperedge is satisfied by its first $\frac{k}{14}$ vertices.

  Then we compute the marginal probability of this partial colouring by pinning vertices one by one.

- For sampling, we use the marginal to colour vertices, similar to the coupling process. We colour $\frac{3k}{16}$ vertices of every hyperedge.

  With high probability, every remaining connected component has size $O(\log n)$.

Counting and sampling

So far we are calculating the marginal probability, which requires that there are sufficiently many uncoloured vertices in all hyperedges.

- For approximate counting, we use the local lemma to find a partial colouring so that every hyperedge is satisfied by its first $\frac{k}{14}$ vertices.

  Then we compute the marginal probability of this partial colouring by pinning vertices one by one.

- For sampling, we use the marginal to colour vertices, similar to the coupling process. We colour $\frac{3k}{16}$ vertices of every hyperedge.

  With high probability, every remaining connected component has size $O(\log n)$.

Counting and sampling

So far we are calculating the marginal probability, which requires that there are sufficiently many uncoloured vertices in all hyperedges.

- For approximate counting, we use the local lemma to find a partial colouring so that every hyperedge is satisfied by its first $\frac{k}{14}$ vertices.

  Then we compute the marginal probability of this partial colouring by pinning vertices one by one.

- For sampling, we use the marginal to colour vertices, similar to the coupling process. We colour $\frac{3k}{16}$ vertices of every hyperedge.

  With high probability, every remaining connected component has size $O(\log n)$.

Concluding remarks

Open problems

- What is the correct threshold for hypergraph colouring?
  - Is it $q \asymp \Delta^{\frac{2}{k}}$?
- What about **NP**-hardness of sampling hypergraph colourings?
- Does this method work for graph colourings?
- Can we show anything about the zeros of hypergraph colourings?

Open problems

- What is the correct threshold for hypergraph colouring?
    - Is it $q \asymp \Delta^{\frac{2}{k}}$?
- What about **NP**-hardness of sampling hypergraph colourings?
- Does this method work for graph colourings?
- Can we show anything about the zeros of hypergraph colourings?

Open problems

- What is the correct threshold for hypergraph colouring?
    - Is it $q \asymp \Delta^{\frac{2}{k}}$?
- What about **NP**-hardness of sampling hypergraph colourings?
- Does this method work for graph colourings?
- Can we show anything about the zeros of hypergraph colourings?

Open problems

- What is the correct threshold for hypergraph colouring?
    - Is it $q \asymp \Delta^{\frac{2}{k}}$?
- What about **NP**-hardness of sampling hypergraph colourings?
- Does this method work for graph colourings?
- Can we show anything about the zeros of hypergraph colourings?

Thank you!

arXiv:1711.03396