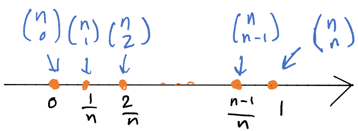
Review

- Fourier Analysis $\quad \mathbb{Z}_{n_1} \times \cdots \times \mathbb{Z}_{n_k}$

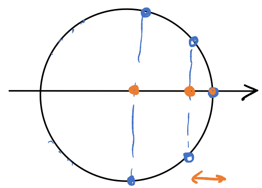
eigenvectors $\leftarrow \omega_1^{x_1} \cdots \omega_n^{x_k}$

$*$ Glauber on $\{0,1\}^n$



$\frac{1}{n}$ spectral gap

$*$ Random walk on Cycle



$\Theta(\frac{1}{n^2})$ spectral gap

- Continuous Time $\quad \nu_t = e^{t(P-I)} \cdot \nu_0$
- Functional Analysis in Continuous Time

$$\frac{d}{dt} D_f(\nu_t \| \mu) \leq -\rho\, D_f(\nu_t \| \mu)$$

$*$ Poincaré: for χ^2 $\quad$ $*$ MLSI: for D_{KL}

Dirichlet form $\rightarrow$ $\mathcal{E}\left(f'\left(\frac{\nu}{\mu}\right), \frac{\nu}{\mu}\right) \geq \rho\, D_f(\nu \| \mu)$

- Dirichlet Form:

$$\mathcal{E}(g,h) := \frac{1}{2} \mathbb{E}_{(x,y)\sim Q}\left[(g(x)-g(y))(h(x)-h(y))\right]$$

- Comparison Method (easy)

P, P' have same stationary

$\forall x \neq y: \quad P(x,y) \geq c \cdot P'(x,y)$

ρ - Poincaré/MLSI/$\cdots$ for P $\Rightarrow$ $c \cdot \rho$ - Poicaré/MLSI/$\cdots$ for P'

Example. Metropolis & Glauber for q-coloring have same Poincaré/MLSI/$\cdots$ consts up to $\frac{q}{q-\Delta}$.

Plan for Today

- Comparison method (hard)
- Trading time for approx
- Sampling matchings

Comparison Method

*All chains time-reversible for now ...

$Q'(x,y) = \mu(x)P'(x,y)$

Idea: Route ergodic flow of P' through that of P with low congestion and length. $\Rightarrow (1-\lambda_2(P)) \geq \frac{1}{C}(1-\lambda_2(P'))$

depends on cong. + len.

Routing: $(s,t) \longmapsto$ convex combination of paths $s = x_0 \sim x_1 \sim \cdots \sim x_\ell = t$

- Easier to think in terms of flows:

π: distribution of path if we sample $(s,t) \sim Q'$.

ergodic flow of P'.

Defining constraint: endpoints (s,t) of path $\sim \pi$ are distributed as

$$Q'(s,t) = \mu(s)P'(s,t) = \mu(t)P'(t,s)$$

Example. (trivial routing)

$$(s,t) \longmapsto s = x_0 \sim x_1 = t$$

Length: We only use paths of length 1.

Congestion: $s \sim t$ has capacity $Q(s,t)$ and we route $Q'(s,t)$ flow through it.

$$\max\left\{\frac{Q'(s,t)}{Q(s,t)}\right\} = \max\left\{\frac{P'(s,t)}{P(s,t)}\right\}$$

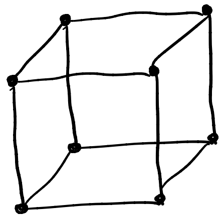
For colorings, P = Metropolis, P' = Glauber, we have congestion $\leq \frac{q}{q-\Delta}$.

Remark. When trivial routing has low congestion we get to transfer not just spectral gap / Poincaré but also MLSI, etc.

Example. (hypercube)

P': $x \longmapsto y \sim \mu$

trivial "ideal" chain



P: Glauber dynamics

μ: uniform on $\{0,1\}^n$

- Demand: $Q'(s,t) = \mu(s)P'(s,t) = \mu(s)\mu(t)$
- Routing: $(s,t) \longmapsto s = x^{(0)} \sim x^{(1)} \sim \cdots \sim x^{(n)} = t$

$x^{(i)} = (t_1, \dots, t_i, \quad s_{i+1}, \dots, s_n)$

"go over bits $1, \dots, n$ and change from s to t one bit at a time"

Length: n

Congestion: There are $2^{n-i} \times 2^{i-1} = 2^{n-1}$ pairs of (s,t) passing through any edge.

$$\text{Congestion} = \frac{2^{n-1} \times \frac{1}{2^n} \times \frac{1}{2^n} \leftarrow \text{flow}}{\frac{1}{2^n} \times \frac{1}{2n} \leftarrow \text{capacity}} = O(n)$$

Remark. When avg. length of path $\sim \pi$ is ℓ, it must be that congestion $\geq \ell$.

total flow $= \ell$ total capacity $= 1$

Theorem. Suppose π is a dist over paths $x_0 \sim x_1 \sim \cdots \sim x_\ell$ such that $(x_0, x_\ell) \sim Q'$.

Let C be:

$$\max \left\{ \frac{\mathbb{E}_{\text{path} \sim \pi}\left[\text{len(path)} \times \mathbb{1}[x \sim y \text{ in path}] \right]}{Q(x,y)} \;\middle|\; \begin{array}{l} x,y \in \Omega \\ x \neq y \end{array} \right\}$$

$\leq$ cong $\times$ len.

Then $\mathcal{E}_P(g,g) \geq \frac{1}{C} \cdot \mathcal{E}_{P'}(g,g)$.

Dirichlet form Dirichlet form

Corollary. $(1 - \lambda_2(P)) \geq \frac{1}{C} \cdot (1 - \lambda_2(P'))$.

Corollary. LSI for $P' \Rightarrow$ LSI for P.

Remark. P' = ideal chain is much-used!

Proof: Let $x_0 \sim x_1 \sim \cdots \sim x_\ell$ be sampled $\sim \pi$:

$$\mathcal{E}_{P'}(g,g) := \frac{1}{2}\mathbb{E}\left[\left(g(x_0)-g(x_\ell)\right)^2\right]$$

By Cauchy-Schwarz we have

$$(\underbrace{1+\cdots+1}_{\ell \text{ times}})\left((g(x_0)-g(x_1))^2+\cdots+(g(x_{\ell-1})-g(x_\ell))^2\right) \geq$$

$$\left((g(x_0)-g(x_1))+\cdots+(g(x_{\ell-1})-g(x_\ell))\right)^2 = (g(x_0)-g(x_\ell))^2$$

Taking expectations we get

$(g(x)-g(\alpha))^2$

$$\text{R.H.S.} = 2\,\mathcal{E}_{P'}(g,g)$$

$$\text{L.H.S.} \leq 2C \cdot \mathcal{E}_P(g,g)$$

So $$\mathcal{E}_P(g,g) \geq \frac{1}{C}\,\mathcal{E}_{P'}(g,g).$$

Example. (hyercube)

Ideal chain mixes in 1 step. In fact

P' is rank 1 so

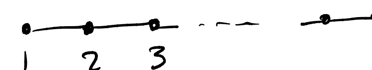
$$\lambda_1(P') = 1 \geq 0 = \lambda_2(P') = \cdots = \lambda_n(P')$$

$$\Rightarrow (1-\lambda_2(\text{Glauber})) \geq \Omega\Big(\frac{1}{\underset{\text{length}}{n} \times \underset{\text{congestion}}{n}}\Big)\cdot 1 = \Omega\Big(\frac{1}{n^2}\Big)$$

not tight

Example. (random walk on path)

P': ideal chain



(similarly for cycle)

P: Metropolis with μ=uniform

Routing: $(s,t) \longmapsto$ subpath from s to t.

Length: $O(n)$

Congestion: $\lesssim \dfrac{n^2 \times \frac{1}{n} \times \frac{1}{n}}{\frac{1}{n} \times \frac{1}{2}} = O(n)$

$\Rightarrow (1-\lambda_2(P)) \geq \Omega\left(\frac{1}{n^2}\right)$ ← this is tight!

Trading time for approx [Jerrum–Sinclair]

Suppose we have α-approx counting ALG for self-reducible problem.

$\Rightarrow$ $(1+\varepsilon)$-approx in time $\text{poly}(n, \alpha, \frac{1}{\varepsilon})$

Corollary. $\alpha = \text{poly}(n) \Longrightarrow$ FPRAS

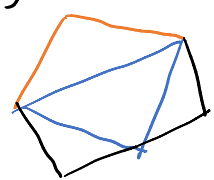
approx counting is all-or-nothing

Example. (colored spanning trees)

Count spanning trees with

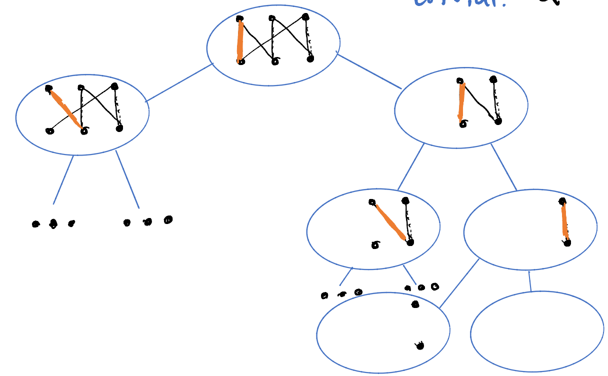
generally matroids

n_1 black, n_2 blue, n_3 orange, ...



Thm: $\exists$ $2^{\tilde{O}(|\text{palette}|)}$-approx ALG

trivial: α^{depth}-time



self-reducibility tree

Idea: Define $\mu'(\text{subproblem}) \propto \text{ALG}(\text{subproblem})$

assume det for simplicity but works with rand too

$\overset{\alpha}{\simeq} \mu(\text{subproblem})$

$\sum_{x \in \text{subproblem}} \mu(x)$

If we sample $\sim \mu'$:

$$\mathbb{P}[\text{base case}] \geq \frac{1}{\alpha \cdot \text{depth}} = \frac{1}{\alpha \cdot \text{poly}(n)}.$$

If we sample base case, we can rejection sample into μ with $\mathbb{P}[\text{accept}] \geq \frac{1}{\alpha}$.

Question: How to sample subproblem w.p. $\propto \mathrm{ALG}(\text{subproblem})$

Idea: Random walk on tree.

put Metropolis filter to get $\propto$ ALG as stationary dist

Claim: $t_{mix} = \mathrm{poly}(n, \alpha)$.

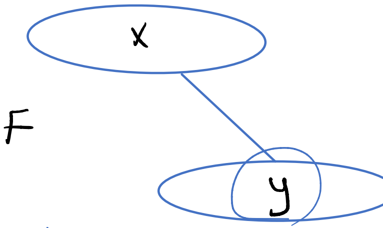
Proof: We compare with ideal chain.

Routing: We route (s,t) demand on the unique tree path.

Length: $2 \cdot \mathrm{depth} = \mathrm{poly}(n)$.

$$\sum_{s \text{ is in } y\text{-}s \text{ subtree}} \mu'(s)\mu'(t) = \sum_s \mu'(s) \cancel{\sum_t \mu'(t)}^{\,1}$$

Congestion:



$$\sum_{s,t} \mu'(s)\mu'(t) =: F$$

routed through $x \sim y$ or $y \sim x$

$$F \le O\Big(\sum_{z \in \text{subtree of } y} \mu'(z)\Big) = \frac{\sum_{z \in y\text{-subtree}} \mathrm{ALG}(z)}{\sum_{z' \in \text{tree}} \mathrm{ALG}(z)} =$$

$$O(\alpha \cdot \mathrm{depth}) \cdot \frac{\mu(y)}{\mu(\mathrm{root})} = O(\alpha\, \mathrm{poly}(n)) \frac{\mu(y)}{\mu(\mathrm{root})}$$

$$Q(x,y) = Q(y,x) \ge \mu'(y) \cdot \frac{1}{\alpha \cdot \mathrm{poly}(n)}$$

$\Rightarrow$ Congestion $= \mathrm{poly}(n, \alpha)$

We conclude $1 - \lambda_2(P) \ge 1/\mathrm{poly}(n, \alpha)$.

If we start at root, we get

$$\lg \chi^2(\mathrm{root} \,\|\, \mu') \le \mathrm{polylg}(n, \alpha).$$

$$\Rightarrow t_{mix}(P, \mathrm{root}, \delta) = \mathrm{poly}(n, \alpha, \lg \tfrac{1}{\delta})$$

Canonical Paths

Suppose we have determinist routing and want:

$$\text{congestion}(x \sim y).$$

specific to $x \sim y$

Encoding: A map from

$$\{(s,t) \mid \underbrace{x \sim y}_{\Omega \times \Omega} \in s\text{-}t \text{ path}\} \longmapsto \Omega \times \underbrace{[M]}_{\text{set of size } M}$$

that's injective & if

$$(s,t) \longmapsto (r, \text{junk}) \text{ we have}$$

$$\mu(s)\mu(t) \leqslant C \cdot \mu(r) Q(x,y)$$

$\Rightarrow$ congestion $\leqslant C \cdot M$ because

$$\sum_{\substack{s,t \\ \text{passing through } x \sim y}} \mu(s)\mu(t) \leqslant C \cdot \underbrace{\sum_{r \in \Omega} \sum_{\substack{\text{junk} \\ \in [M]}} \mu(r)}_{M.} Q(x,y)$$

Example. (hypercube)

Suppose $x \sim_i y$ $\rightarrow$ differ in coord i

Encoding:

$$(s,t) \longmapsto r := (s_1, \ldots, s_i, t_{i+1}, \ldots, t_n)$$

Injective because knowing y & r we can recover both s & t!

$$\cancel{\mu(s)}\,\cancel{\mu(t)} \leqslant O(n) \cdot \cancel{\mu(r)}\,\underbrace{\cancel{\mu(x)}\,\overbrace{P(x,y)}^{\geqslant \frac{1}{n}}}_{Q(x,y)}$$

Note: In general for uniform μ we only need $P(x,y) \geqslant \frac{1}{\text{poly}(n)}$ & injective!

Markov Chain for Matchings

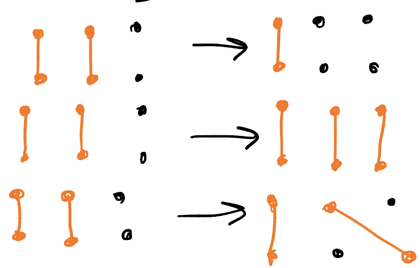
- Goal: Given unweighted graph G, count/sample its matchings.

  not necessarily perfect matchings

- Markov Chain [proposed by Broder, analyzed by Jerrum-Sinclair]

  Move from $M \to M'$ by

  * Removing edge
  * Adding edge
  * $\underbrace{\text{Exchanging edge}}_{\text{not strictly necessary}}$



+ Metropolis rule

Details are unimportant, as long as

$$P(M, M') \geq \frac{1}{\mathrm{poly}(n)}$$ for every valid move from M to M'.

We will design paths for every M to M' & use injective encoding!

enough because uniform

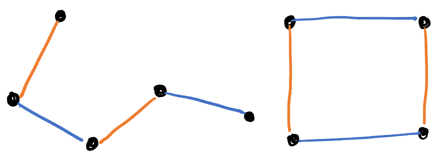
$$\mu(M)\,\mu(M') \leq \mathrm{poly}(n) \cdot \mu(\text{encoding})\, Q(\text{transition})$$

this is automatic

Paths: Consider matchings M, M'.

If we look at $M \underset{\text{xor}}{\oplus} M'$, we get

Collection of paths + cycles.



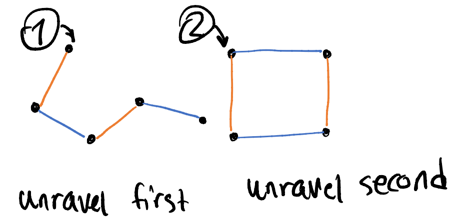
- Path from M to M':

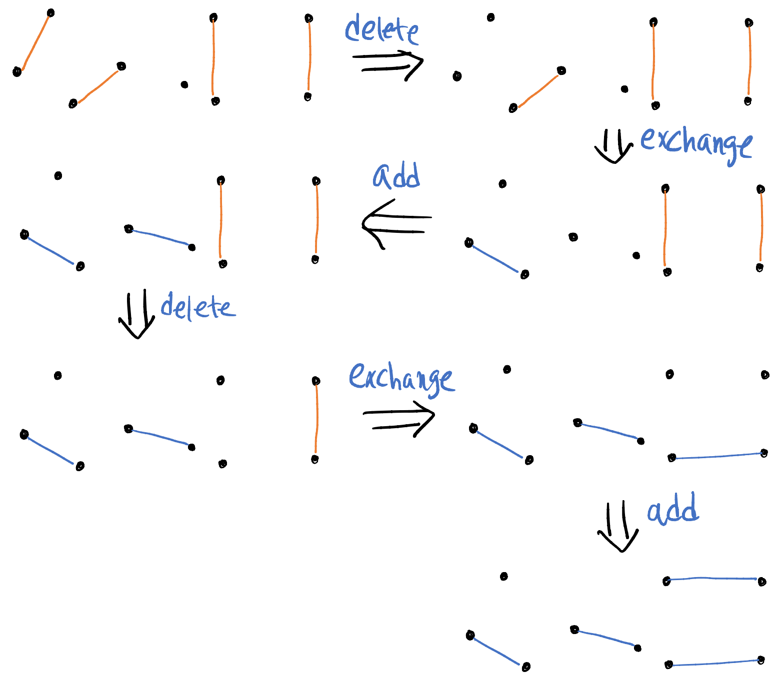
to make it det.

Arb. det. order over possible paths / cycles & arb choice of starting endpoint / vertex.

Go over paths / cycles in this order & unravel one-by-one.

Example:





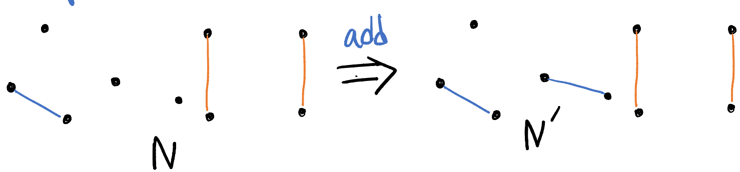
Encoding: For the $N \sim N'$ transition we can encode

$$(M, M') \longmapsto (M \oplus M' \oplus N - \text{few edges},\ \text{few edges})$$

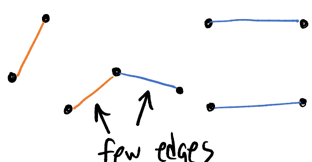
there might be two edges around current vertex

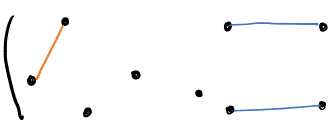
junk part technically not needed

Example.



$M \oplus M' \oplus N =$



Encoding: ( , few edges)

Claim: Injective!

Proof:

- We can recover $M \oplus M'$ from encoding & N.
- We know what path/cycle we are unravelling by $N \sim N'$
- We can rewind unravel starting from N/N' to recover M.
- We can continue unravel from N/N' to recover M'.

Thm: $1 - \lambda_2(\text{chain}) \geq \dfrac{1}{\text{poly}(n)}$:)

Corollary: $t_{mix} = \text{poly}(n)$ ← Open: fast algs!