

Nodal Domains and Diffusion Processes

Jacob Denson

University of Wisconsin Madison

October 2, 2022

- ▶ *Nodal Geometry, Heat Diffusion, and Brownian Motion*
Georgiev, Mukherjee, 2018.
- ▶ *Lower Bounds on Nodal Sets of Eigenfunctions via the Heat Flow*
Steinerberger, 2014.
- ▶ *Stochastic Differential Equations*
Øksendal, Springer Universitext, 2003.
- ▶ *Green, Brown, and Probability and Brownian Motion*
Chung, World Scientific, 2002.

Nodal Domains

- ▶ Let M^d be a compact Riemannian manifold.
- ▶ Let $e_\lambda \in C^\infty(M)$, $\Delta e_\lambda = -\lambda^2 e_\lambda$.
- ▶ *Nodal set* of e_λ : $Z(e_\lambda) = \{x \in M : e_\lambda(x) = 0\}$.
- ▶ *Nodal domain* D_λ : a connected component of $M - Z(e_\lambda)$.
- ▶ Goal: Study the 'asymptotic geometry' of Ω_λ as $\lambda \rightarrow \infty$.

Nodal Domains

- ▶ Let M^d be a compact Riemannian manifold.
- ▶ Let $e_\lambda \in C^\infty(M)$, $\Delta e_\lambda = -\lambda^2 e_\lambda$.
- ▶ *Nodal set* of e_λ : $Z(e_\lambda) = \{x \in M : e_\lambda(x) = 0\}$.
- ▶ *Nodal domain* D_λ : a connected component of $M - Z(e_\lambda)$.
- ▶ Goal: Study the 'asymptotic geometry' of Ω_λ as $\lambda \rightarrow \infty$.
- ▶ (Lieb, 1990) To calculate $BL(B,p)$, plug in Gaussians.

Nodal Domains

- ▶ Let M^d be a compact Riemannian manifold.
- ▶ Let $e_\lambda \in C^\infty(M)$, $\Delta e_\lambda = -\lambda^2 e_\lambda$.
- ▶ *Nodal set* of e_λ : $Z(e_\lambda) = \{x \in M : e_\lambda(x) = 0\}$.
- ▶ *Nodal domain* D_λ : a connected component of $M - Z(e_\lambda)$.
- ▶ Goal: Study the 'asymptotic geometry' of Ω_λ as $\lambda \rightarrow \infty$.
- ▶ (Lieb, 1990) To calculate $\text{BL}(B,p)$, plug in Gaussians.
- ▶ If $f_i(x) = e^{-\pi(A_i x) \cdot x}$ for $A_i \succ 0$, then

Nodal Domains

- ▶ Let M^d be a compact Riemannian manifold.
- ▶ Let $e_\lambda \in C^\infty(M)$, $\Delta e_\lambda = -\lambda^2 e_\lambda$.
- ▶ *Nodal set* of e_λ : $Z(e_\lambda) = \{x \in M : e_\lambda(x) = 0\}$.
- ▶ *Nodal domain* D_λ : a connected component of $M - Z(e_\lambda)$.
- ▶ Goal: Study the 'asymptotic geometry' of Ω_λ as $\lambda \rightarrow \infty$.
- ▶ (Lieb, 1990) To calculate $\text{BL}(B,p)$, plug in Gaussians.
- ▶ If $f_i(x) = e^{-\pi(A_i x) \cdot x}$ for $A_i \succ 0$, then

$$\int_{\mathbb{R}^n} \prod_{i=1}^m |f_i(B_i x)|^{p_i} dx = \det(\sum p_i B_i^* A_i B_i)^{-1/2}$$

Nodal Domains

- ▶ Let M^d be a compact Riemannian manifold.
- ▶ Let $e_\lambda \in C^\infty(M)$, $\Delta e_\lambda = -\lambda^2 e_\lambda$.
- ▶ *Nodal set* of e_λ : $Z(e_\lambda) = \{x \in M : e_\lambda(x) = 0\}$.
- ▶ *Nodal domain* D_λ : a connected component of $M - Z(e_\lambda)$.
- ▶ Goal: Study the 'asymptotic geometry' of Ω_λ as $\lambda \rightarrow \infty$.
- ▶ (Lieb, 1990) To calculate $\text{BL}(B,p)$, plug in Gaussians.
- ▶ If $f_i(x) = e^{-\pi(A_i x) \cdot x}$ for $A_i \succ 0$, then

$$\int_{\mathbb{R}^n} \prod_{i=1}^m |f_i(B_i x)|^{p_i} dx = \det\left(\sum p_i B_i^* A_i B_i\right)^{-1/2}$$

$$\prod_{i=1}^m \|f_i\|_{L^1(\mathbb{R}^{n_i})}^{p_i} = \left(\prod_{i=1}^m \det(A_i)^{p_i}\right)^{-1/2}.$$

Nodal Domains

- ▶ Let M^d be a compact Riemannian manifold.
- ▶ Let $e_\lambda \in C^\infty(M)$, $\Delta e_\lambda = -\lambda^2 e_\lambda$.
- ▶ *Nodal set* of e_λ : $Z(e_\lambda) = \{x \in M : e_\lambda(x) = 0\}$.
- ▶ *Nodal domain* D_λ : a connected component of $M - Z(e_\lambda)$.
- ▶ Goal: Study the 'asymptotic geometry' of Ω_λ as $\lambda \rightarrow \infty$.
- ▶ (Lieb, 1990) To calculate $\text{BL}(B, p)$, plug in Gaussians.
- ▶ If $f_i(x) = e^{-\pi(A_i x) \cdot x}$ for $A_i \succ 0$, then

$$\int_{\mathbb{R}^n} \prod_{i=1}^m |f_i(B_i x)|^{p_i} dx = \det\left(\sum p_i B_i^* A_i B_i\right)^{-1/2}$$

$$\prod_{i=1}^m \|f_i\|_{L^1(\mathbb{R}^{n_i})}^{p_i} = \left(\prod_{i=1}^m \det(A_i)^{p_i}\right)^{-1/2}.$$

Thus

$$\det\left(\sum p_i B_i^* A_i B_i\right)^{-1/2} \leq \text{BL}(B, p) \cdot \left(\prod_{i=1}^m \det(A_i)^{p_i}\right)^{-1/2}$$

Lieb's Theorem

- Rearranging gives

$$\text{BL}(B, p) = \left(\sup_{A_1, \dots, A_m \succ 0} \frac{\prod_i \det(A_i)^{p_i}}{\det(\sum p_i B_i^* A_i B_i)} \right)^{1/2}.$$

Lieb's Theorem

- ▶ Rearranging gives

$$\text{BL}(B, p) = \left(\sup_{A_1, \dots, A_m \succ 0} \frac{\prod_i \det(A_i)^{p_i}}{\det(\sum p_i B_i^* A_i B_i)} \right)^{1/2}.$$

- ▶ Non convex objective, so tricky to optimize (NP hard).

Lieb's Theorem

- ▶ Rearranging gives

$$\text{BL}(B, p) = \left(\sup_{A_1, \dots, A_m \succ 0} \frac{\prod_i \det(A_i)^{p_i}}{\det(\sum p_i B_i^* A_i B_i)} \right)^{1/2}.$$

- ▶ Non convex objective, so tricky to optimize (NP hard).
- ▶ Checking finiteness efficiently also non-obvious.

Lieb's Theorem

- ▶ Rearranging gives

$$\mathrm{BL}(B, p) = \left(\sup_{A_1, \dots, A_m \succ 0} \frac{\prod_i \det(A_i)^{p_i}}{\det(\sum p_i B_i^* A_i B_i)} \right)^{1/2}.$$

- ▶ Non convex objective, so tricky to optimize (NP hard).
- ▶ Checking finiteness efficiently also non-obvious.
 - ▶ BCCT gives a family of conditions to check finiteness.

$$\dim(V) \leq \sum p_i \dim(B_i V).$$

Lieb's Theorem

- ▶ Rearranging gives

$$\text{BL}(B, p) = \left(\sup_{A_1, \dots, A_m \succ 0} \frac{\prod_i \det(A_i)^{p_i}}{\det(\sum p_i B_i^* A_i B_i)} \right)^{1/2}.$$

- ▶ Non convex objective, so tricky to optimize (NP hard).
- ▶ Checking finiteness efficiently also non-obvious.
 - ▶ BCCT gives a family of conditions to check finiteness.

$$\dim(V) \leq \sum p_i \dim(B_i V).$$

Since $0 \leq \dim(V) \leq n$ and $0 \leq \dim(B_i V) \leq n_i$, only finitely many linear conditions on the exponents p_i .

Lieb's Theorem

- ▶ Rearranging gives

$$\text{BL}(B, p) = \left(\sup_{A_1, \dots, A_m \succ 0} \frac{\prod_i \det(A_i)^{p_i}}{\det(\sum p_i B_i^* A_i B_i)} \right)^{1/2}.$$

- ▶ Non convex objective, so tricky to optimize (NP hard).
- ▶ Checking finiteness efficiently also non-obvious.
 - ▶ BCCT gives a family of conditions to check finiteness.

$$\dim(V) \leq \sum p_i \dim(B_i V).$$

Since $0 \leq \dim(V) \leq n$ and $0 \leq \dim(B_i V) \leq n_i$, only finitely many linear conditions on the exponents p_i .

- ▶ Can be exponentially many constraints, so inefficient.

Geometric Brascamp Lieb

$$\int_{\mathbf{R}^n} \prod_{i=1}^m |f_i(B_i x)|^{p_i} dx \leq \text{BL}(B, p) \cdot \prod_{i=1}^m \|f_i\|_{L^1(\mathbf{R}^{n_i})}^{p_i}$$

Geometric Brascamp Lieb

$$\int_{\mathbf{R}^n} \prod_{i=1}^m |f_i(B_i x)|^{p_i} dx \leq \text{BL}(B, p) \cdot \prod_{i=1}^m \|f_i\|_{L^1(\mathbf{R}^{n_i})}^{p_i}$$

- ▶ A Brascamp-Lieb inequality is *geometric* if
 - ▶ (Projection Property): $B_i B_i^* = I$.
 - ▶ (Isotropy Property): $\sum_i p_i B_i^* B_i = I$.

Geometric Brascamp Lieb

$$\int_{\mathbf{R}^n} \prod_{i=1}^m |f_i(B_i x)|^{p_i} dx \leq \text{BL}(B, p) \cdot \prod_{i=1}^m \|f_i\|_{L^1(\mathbf{R}^{n_i})}^{p_i}$$

- ▶ A Brascamp-Lieb inequality is *geometric* if
 - ▶ (Projection Property): $B_i B_i^* = I$.
 - ▶ (Isotropy Property): $\sum_i p_i B_i^* B_i = I$.
- ▶ (Barthe, 1998), generalizing (Ball, 1989), showed that if (B, p) is geometric, $\text{BL}(B, p) = 1$.

Operator Rescaling

- ▶ What happens to $\text{BL}(B, p)$ if we 'change coordinates'.

Operator Rescaling

- ▶ What happens to $\text{BL}(B, p)$ if we ‘change coordinates’.
- ▶ Fix invertible matrices M and M_1, \dots, M_m , and consider the Brascamp-Lieb inequality with matrices $B'_i = M_i B_i M$,

$$\int_{\mathbf{R}^n} \prod_{i=1}^m |f_i(M_i B_i M x)|^{p_i} dx \leq \text{BL}(B', p) \cdot \prod_{i=1}^m \|f_i\|_{L^1(\mathbf{R}^{n_i})}^{p_i}.$$

Operator Rescaling

- ▶ What happens to $\text{BL}(B, p)$ if we 'change coordinates'.
- ▶ Fix invertible matrices M and M_1, \dots, M_m , and consider the Brascamp-Lieb inequality with matrices $B'_i = M_i B_i M$,

$$\int_{\mathbf{R}^n} \prod_{i=1}^m |f_i(M_i B_i M x)|^{p_i} dx \leq \text{BL}(B', p) \cdot \prod_{i=1}^m \|f_i\|_{L^1(\mathbf{R}^{n_i})}^{p_i}.$$

$$\begin{aligned} \text{BL}(B', p) &= \left(\sup_{A_1, \dots, A_m \succ 0} \frac{\prod_i \det(A_i)^{p_i}}{\det(\sum_i p_i M_i^* B_i^* M_i^* A_i M_i B_i M)} \right)^{1/2} \\ &= \det(M)^{-1} \prod_i \det(M_i)^{-p_i} \cdot \text{BL}(B, p). \end{aligned}$$

Operator Rescaling

$$\mathrm{BL}(B', p) = \det(M)^{-1} \prod_i \det(M_i)^{-p_i} \cdot \mathrm{BL}(B, p)$$

Operator Rescaling

$$\mathrm{BL}(B', p) = \det(M)^{-1} \prod_i \det(M_i)^{-p_i} \cdot \mathrm{BL}(B, p)$$

- If (B', p) is geometric, $\mathrm{BL}(B', p) = 1$, so

$$\mathrm{BL}(B, p) = \det(M) \prod_i \det(M_i)^{p_i}.$$

Operator Rescaling

$$\mathrm{BL}(B', p) = \det(M)^{-1} \prod_i \det(M_i)^{-p_i} \cdot \mathrm{BL}(B, p)$$

- ▶ If (B', p) is geometric, $\mathrm{BL}(B', p) = 1$, so

$$\mathrm{BL}(B, p) = \det(M) \prod_i \det(M_i)^{p_i}.$$

- ▶ (BCCT) Geometric rescaling possible iff extremizers exist.

Main Result

- ▶ (Garg, Gurvits, Oliveira, Wigderson, 2016) If $\text{BL}(B, p) < \infty$, we can rescale to (B', p) with $\text{BL}(B', p) \leq 1 + \varepsilon$, for any $\varepsilon > 0$.

Main Result

- ▶ (Garg, Gurvits, Oliveira, Wigderson, 2016) If $\text{BL}(B, p) < \infty$, we can rescale to (B', p) with $\text{BL}(B', p) \leq 1 + \varepsilon$, for any $\varepsilon > 0$.
 - ▶ For $\varepsilon \ll 1$, $\text{BL}(B, p) \approx \det(M)^2 \prod_i \det(M_i)^{2p_i}$.

Main Result

- ▶ (Garg, Gurvits, Oliveira, Wigderson, 2016) If $\text{BL}(B, p) < \infty$, we can rescale to (B', p) with $\text{BL}(B', p) \leq 1 + \varepsilon$, for any $\varepsilon > 0$.
 - ▶ For $\varepsilon \ll 1$, $\text{BL}(B, p) \approx \det(M)^2 \prod_i \det(M_i)^{2p_i}$.
 - ▶ We can do this algorithmically, i.e. if p_i are rationals, with common denominator d , a computer can compute a ε -approximate geometric rescaling in $\text{Poly}(\text{Bits}(B, p), d, 1/\varepsilon)$ computations.

Main Result

- ▶ (Garg, Gurvits, Oliveira, Wigderson, 2016) If $\text{BL}(B, p) < \infty$, we can rescale to (B', p) with $\text{BL}(B', p) \leq 1 + \varepsilon$, for any $\varepsilon > 0$.
 - ▶ For $\varepsilon \ll 1$, $\text{BL}(B, p) \approx \det(M)^2 \prod_i \det(M_i)^{2p_i}$.
 - ▶ We can do this algorithmically, i.e. if p_i are rationals, with common denominator d , a computer can compute a ε -approximate geometric rescaling in $\text{Poly}(\text{Bits}(B, p), d, 1/\varepsilon)$ computations.
 - ▶ Conversely, we can determine if $\text{BL}(B, p) = \infty$, and find a violating subspace V to the condition $\dim(V) \leq \sum p_i \dim(B_i V)$ in this case, in $\text{Poly}(\text{Bits}(B, p), d)$ computations.

Main Result

- ▶ (Garg, Gurvits, Oliveira, Wigderson, 2016) If $\text{BL}(B, p) < \infty$, we can rescale to (B', p) with $\text{BL}(B', p) \leq 1 + \varepsilon$, for any $\varepsilon > 0$.
 - ▶ For $\varepsilon \ll 1$, $\text{BL}(B, p) \approx \det(M)^2 \prod_i \det(M_i)^{2p_i}$.
 - ▶ We can do this algorithmically, i.e. if p_i are rationals, with common denominator d , a computer can compute a ε -approximate geometric rescaling in $\text{Poly}(\text{Bits}(B, p), d, 1/\varepsilon)$ computations.
 - ▶ Conversely, we can determine if $\text{BL}(B, p) = \infty$, and find a violating subspace V to the condition $\dim(V) \leq \sum p_i \dim(B_i V)$ in this case, in $\text{Poly}(\text{Bits}(B, p), d)$ computations.
 - ▶ Open Problem: Can we improve this to $\text{Poly}(\text{Bits}(B, p), d, \log(1/\varepsilon))$ computations?

Analytic Consequences

Analytic Consequences

- ▶ Typing a single digit is a single unit of computation.

Analytic Consequences

- ▶ Typing a single digit is a single unit of computation.
 - ▶ Thus given (B, p) , either $\text{BL}(B, p) < \infty$, or

$$\text{BL}(B, p) \leq 2^{\text{Poly}(\text{Bits}(B, p), d, \log(1/\varepsilon))}.$$

and

$$\text{BL}(B, p) \geq 2^{-\text{Poly}(\text{Bits}(B, p), d, \log(1/\varepsilon))}$$

Analytic Consequences

- ▶ Typing a single digit is a single unit of computation.
 - ▶ Thus given (B, p) , either $\text{BL}(B, p) < \infty$, or

$$\text{BL}(B, p) \leq 2^{\text{Poly}(\text{Bits}(B, p), d, \log(1/\varepsilon))}.$$

and

$$\text{BL}(B, p) \geq 2^{-\text{Poly}(\text{Bits}(B, p), d, \log(1/\varepsilon))}$$

- ▶ Calculation is also stable.

Analytic Consequences

- ▶ Typing a single digit is a single unit of computation.
 - ▶ Thus given (B, p) , either $\text{BL}(B, p) < \infty$, or

$$\text{BL}(B, p) \leq 2^{\text{Poly}(\text{Bits}(B, p), d, \log(1/\varepsilon))}.$$

and

$$\text{BL}(B, p) \geq 2^{-\text{Poly}(\text{Bits}(B, p), d, \log(1/\varepsilon))}$$

- ▶ Calculation is also stable.
 - ▶ Thus we conclude a *continuity bound*:

Analytic Consequences

- ▶ Typing a single digit is a single unit of computation.
 - ▶ Thus given (B, p) , either $\text{BL}(B, p) < \infty$, or

$$\text{BL}(B, p) \leq 2^{\text{Poly}(\text{Bits}(B, p), d, \log(1/\varepsilon))}.$$

and

$$\text{BL}(B, p) \geq 2^{-\text{Poly}(\text{Bits}(B, p), d, \log(1/\varepsilon))}$$

- ▶ Calculation is also stable.
 - ▶ Thus we conclude a *continuity bound*:
 - ▶ For each (B, p) with $\text{BL}(B, p) < \infty$, and each $\varepsilon > 0$,

Analytic Consequences

- ▶ Typing a single digit is a single unit of computation.
 - ▶ Thus given (B, p) , either $\text{BL}(B, p) < \infty$, or

$$\text{BL}(B, p) \leq 2^{\text{Poly}(\text{Bits}(B, p), d, \log(1/\varepsilon))}.$$

and

$$\text{BL}(B, p) \geq 2^{-\text{Poly}(\text{Bits}(B, p), d, \log(1/\varepsilon))}$$

- ▶ Calculation is also stable.
 - ▶ Thus we conclude a *continuity bound*:
 - ▶ For each (B, p) with $\text{BL}(B, p) < \infty$, and each $\varepsilon > 0$, if

$$\|B_1 - B\| \leq \exp\left(-\frac{\text{Poly}(\text{Bits}(B, p), d)}{\varepsilon^3}\right),$$

Analytic Consequences

- ▶ Typing a single digit is a single unit of computation.
 - ▶ Thus given (B, p) , either $\text{BL}(B, p) < \infty$, or

$$\text{BL}(B, p) \leq 2^{\text{Poly}(\text{Bits}(B, p), d, \log(1/\varepsilon))}.$$

and

$$\text{BL}(B, p) \geq 2^{-\text{Poly}(\text{Bits}(B, p), d, \log(1/\varepsilon))}$$

- ▶ Calculation is also stable.
 - ▶ Thus we conclude a *continuity bound*:
 - ▶ For each (B, p) with $\text{BL}(B, p) < \infty$, and each $\varepsilon > 0$, if

$$\|B_1 - B\| \leq \exp\left(-\frac{\text{Poly}(\text{Bits}(B, p), d)}{\varepsilon^3}\right),$$

then

$$(1 - \varepsilon)\text{BL}(B, p) \leq \text{BL}(B_1, p) \leq (1 + \varepsilon)\text{BL}(B, p).$$

Computing Permanents (An Analogous Problem)

Computing Permanents (An Analogous Problem)

- ▶ Given an $n \times n$ matrix A with non-negative entries, define

$$\text{Perm}(A) = \sum_{\sigma \in S_n} \prod_i A_{i\sigma(i)}.$$

Computing Permanents (An Analogous Problem)

- ▶ Given an $n \times n$ matrix A with non-negative entries, define

$$\text{Perm}(A) = \sum_{\sigma \in S_n} \prod_i A_{i\sigma(i)}.$$

- ▶ This is (surprisingly) NP hard to compute.

Computing Permanents (An Analogous Problem)

- ▶ Given an $n \times n$ matrix A with non-negative entries, define

$$\text{Perm}(A) = \sum_{\sigma \in S_n} \prod_i A_{i\sigma(i)}.$$

- ▶ This is (surprisingly) NP hard to compute.
 - ▶ If

$$R = \begin{pmatrix} \lambda_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \lambda_n \end{pmatrix} \quad \text{and} \quad C = \begin{pmatrix} \gamma_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \gamma_n \end{pmatrix},$$

then

$$\begin{aligned} \text{Perm}(RAC) &= \text{Perm}(\lambda_i A_{ij} \gamma_j) \\ &= (\lambda_1 \dots \lambda_n)(\gamma_1 \dots \gamma_n) \text{Perm}(A) \\ &= \det(R) \det(C) \text{Perm}(A). \end{aligned}$$

Computing Permanents (An Analogous Problem)

$$\text{Perm}(RAC) = \det(R) \det(C) \text{Perm}(A).$$

Computing Permanents (An Analogous Problem)

$$\text{Perm}(RAC) = \det(R) \det(C) \text{Perm}(A).$$

- ▶ (Egorychev, 1981), (Falikman, 1981) If A is *doubly stochastic*, $e^{-n} \leq \text{Perm}(A) \leq 1$.

Computing Permanents (An Analogous Problem)

$$\text{Perm}(RAC) = \det(R) \det(C) \text{Perm}(A).$$

- ▶ (Egorychev, 1981), (Falikman, 1981) If A is *doubly stochastic*, $e^{-n} \leq \text{Perm}(A) \leq 1$.
- ▶ If RAC is doubly stochastic, then

$$\text{Perm}(A) \leq \det(R)^{-1} \det(C)^{-1} \leq e^n \cdot \text{Perm}(A),$$

so $\text{Perm}(A) \approx \det(R)^{-1} \det(C)^{-1}$.

The Algorithm

- ▶ Given a non-negative matrix A_0 , alternatively apply row and column normalization, obtaining a sequence

$$A_0 \rightarrow A_1 \rightarrow A_2 \rightarrow \dots$$

- ▶ Given a non-negative matrix A_0 , alternatively apply row and column normalization, obtaining a sequence

$$A_0 \rightarrow A_1 \rightarrow A_2 \rightarrow \dots$$

- ▶ Claim: If $\text{Per}(A_0) > 0$, $d(A_i, \mathbf{S}) \rightarrow 0$, where \mathbf{S} is the family of doubly stochastic matrices.

- ▶ Given a non-negative matrix A_0 , alternatively apply row and column normalization, obtaining a sequence

$$A_0 \rightarrow A_1 \rightarrow A_2 \rightarrow \dots$$

- ▶ Claim: If $\text{Per}(A_0) > 0$, $d(A_i, \mathbf{S}) \rightarrow 0$, where \mathbf{S} is the family of doubly stochastic matrices.
- ▶ For even (odd) i , let γ_{ij} be the j th row (column) sum of A_i , so that $\text{Per}(A_{i+1}) = (\gamma_{i1} \dots \gamma_{in})^{-1} \cdot \text{Per}(A_i)$.

- ▶ Given a non-negative matrix A_0 , alternatively apply row and column normalization, obtaining a sequence

$$A_0 \rightarrow A_1 \rightarrow A_2 \rightarrow \dots$$

- ▶ Claim: If $\text{Per}(A_0) > 0$, $d(A_i, \mathbf{S}) \rightarrow 0$, where \mathbf{S} is the family of doubly stochastic matrices.
- ▶ For even (odd) i , let γ_{ij} be the j th row (column) sum of A_i , so that $\text{Per}(A_{i+1}) = (\gamma_{i1} \dots \gamma_{in})^{-1} \cdot \text{Per}(A_i)$.
- ▶ **Two Key Facts Ensuring Convergence:**

- ▶ Given a non-negative matrix A_0 , alternatively apply row and column normalization, obtaining a sequence

$$A_0 \rightarrow A_1 \rightarrow A_2 \rightarrow \dots$$

- ▶ Claim: If $\text{Per}(A_0) > 0$, $d(A_i, \mathbf{S}) \rightarrow 0$, where \mathbf{S} is the family of doubly stochastic matrices.
- ▶ For even (odd) i , let γ_{ij} be the j th row (column) sum of A_i , so that $\text{Per}(A_{i+1}) = (\gamma_{i1} \dots \gamma_{in})^{-1} \cdot \text{Per}(A_i)$.
- ▶ **Two Key Facts Ensuring Convergence:**
 - (1) $\text{Per}(A) \leq 1$ if A is partially normalized.

- ▶ Given a non-negative matrix A_0 , alternatively apply row and column normalization, obtaining a sequence

$$A_0 \rightarrow A_1 \rightarrow A_2 \rightarrow \dots$$

- ▶ Claim: If $\text{Per}(A_0) > 0$, $d(A_i, \mathbf{S}) \rightarrow 0$, where \mathbf{S} is the family of doubly stochastic matrices.
- ▶ For even (odd) i , let γ_{ij} be the j th row (column) sum of A_i , so that $\text{Per}(A_{i+1}) = (\gamma_{i1} \dots \gamma_{in})^{-1} \cdot \text{Per}(A_i)$.
- ▶ **Two Key Facts Ensuring Convergence:**
 - (1) $\text{Per}(A) \leq 1$ if A is partially normalized.
 - (2) If $\Delta_i = \sum_j (\gamma_{ij} - 1)^2$, $\text{Per}(A_{i+1}) \geq (1 + C\Delta_i) \cdot \text{Per}(A_i)$.

- ▶ Given a non-negative matrix A_0 , alternatively apply row and column normalization, obtaining a sequence

$$A_0 \rightarrow A_1 \rightarrow A_2 \rightarrow \dots$$

- ▶ Claim: If $\text{Per}(A_0) > 0$, $d(A_i, \mathbf{S}) \rightarrow 0$, where \mathbf{S} is the family of doubly stochastic matrices.
- ▶ For even (odd) i , let γ_{ij} be the j th row (column) sum of A_i , so that $\text{Per}(A_{i+1}) = (\gamma_{i1} \dots \gamma_{in})^{-1} \cdot \text{Per}(A_i)$.
- ▶ **Two Key Facts Ensuring Convergence:**
 - (1) $\text{Per}(A) \leq 1$ if A is partially normalized.
 - (2) If $\Delta_i = \sum_j (\gamma_{ij} - 1)^2$, $\text{Per}(A_{i+1}) \geq (1 + C\Delta_i) \cdot \text{Per}(A_i)$.
- ▶ Thus $\text{Per}(A_i)$ is bounded, monotonic, converges to $P \leq 1$.

- ▶ Given a non-negative matrix A_0 , alternatively apply row and column normalization, obtaining a sequence

$$A_0 \rightarrow A_1 \rightarrow A_2 \rightarrow \dots$$

- ▶ Claim: If $\text{Per}(A_0) > 0$, $d(A_i, \mathbf{S}) \rightarrow 0$, where \mathbf{S} is the family of doubly stochastic matrices.
- ▶ For even (odd) i , let γ_{ij} be the j th row (column) sum of A_i , so that $\text{Per}(A_{i+1}) = (\gamma_{i1} \dots \gamma_{in})^{-1} \cdot \text{Per}(A_i)$.
- ▶ **Two Key Facts Ensuring Convergence:**
 - (1) $\text{Per}(A) \leq 1$ if A is partially normalized.
 - (2) If $\Delta_i = \sum_j (\gamma_{ij} - 1)^2$, $\text{Per}(A_{i+1}) \geq (1 + C\Delta_i) \cdot \text{Per}(A_i)$.
- ▶ Thus $\text{Per}(A_i)$ is bounded, monotonic, converges to $P \leq 1$.
- ▶ If $\text{Per}(A_i) \geq P - \varepsilon$ for $\varepsilon \ll 1$, then

$$P \geq \text{Per}(A_{i+1}) \geq (1 + C \cdot \Delta_i) \cdot \text{Per}(A_i) \geq (1 + C \cdot \Delta_i)(P - \varepsilon).$$

Thus $\Delta_i \lesssim \varepsilon$. Taking $\varepsilon \rightarrow 0$ shows $\Delta_i \rightarrow 0$.

Sinkhorn Iteration

- ▶ **Proof that $\text{Per}(A_i) \leq 1$:**

Sinkhorn Iteration

- ▶ **Proof that $\text{Per}(A_i) \leq 1$:**
 - ▶ Simple inductive argument: the hypothesis is that if each row of a matrix B all sum up to less than one, then $\text{Per}(B) \leq 1$.

Sinkhorn Iteration

- ▶ **Proof that $\text{Per}(A_i) \leq 1$:**
 - ▶ Simple inductive argument: the hypothesis is that if each row of a matrix B all sum up to less than one, then $\text{Per}(B) \leq 1$.
- ▶ **Proof that $\text{Per}(A_{i+1}) \geq (1 + C\Delta_i) \cdot \text{Per}(A_i)$:**

Sinkhorn Iteration

- ▶ **Proof that $\text{Per}(A_i) \leq 1$:**
 - ▶ Simple inductive argument: the hypothesis is that if each row of a matrix B all sum up to less than one, then $\text{Per}(B) \leq 1$.
- ▶ **Proof that $\text{Per}(A_{i+1}) \geq (1 + C\Delta_i) \cdot \text{Per}(A_i)$:**
 - ▶ Really just more robust form of AGM inequality.

Sinkhorn Iteration

- ▶ **Proof that $\text{Per}(A_i) \leq 1$:**
 - ▶ Simple inductive argument: the hypothesis is that if each row of a matrix B all sum up to less than one, then $\text{Per}(B) \leq 1$.
- ▶ **Proof that $\text{Per}(A_{i+1}) \geq (1 + C\Delta_i) \cdot \text{Per}(A_i)$:**
 - ▶ Really just more robust form of AGM inequality.
 - ▶ If γ_{ij} are the row sums, then because the column sums are all one,

$$\frac{1}{n} \sum_j \gamma_{ij} = 1.$$

Sinkhorn Iteration

- ▶ **Proof that $\text{Per}(A_i) \leq 1$:**

- ▶ Simple inductive argument: the hypothesis is that if each row of a matrix B all sum up to less than one, then $\text{Per}(B) \leq 1$.

- ▶ **Proof that $\text{Per}(A_{i+1}) \geq (1 + C\Delta_i) \cdot \text{Per}(A_i)$:**

- ▶ Really just more robust form of AGM inequality.
- ▶ If γ_{ij} are the row sums, then because the column sums are all one,

$$\frac{1}{n} \sum_j \gamma_{ij} = 1.$$

- ▶ AGM implies $\gamma_{i1} \dots \gamma_{in} \geq 1$, and monotonicity follows from

$$\text{Per}(A_{i+1}) = (\gamma_{i1} \dots \gamma_{in})^{-1} \text{Per}(A_i).$$

And now, back to our regularly scheduled programming

$$\text{BL}(B, p) = \sqrt{\sup_{A_1, \dots, A_m \succ 0} \frac{\prod_i \det(A_i)^{p_i}}{\det(\sum p_i \cdot B_i^* A_i B_i)}}.$$

And now, back to our regularly scheduled programming

$$\text{BL}(B, p) = \sqrt{\sup_{A_1, \dots, A_m \succ 0} \frac{\prod_i \det(A_i)^{p_i}}{\det(\sum p_i \cdot B_i^* A_i B_i)}}.$$

- Goal: Rescale our inputs so that

And now, back to our regularly scheduled programming

$$\text{BL}(B, p) = \sqrt{\sup_{A_1, \dots, A_m \succ 0} \frac{\prod_i \det(A_i)^{p_i}}{\det(\sum p_i \cdot B_i^* A_i B_i)}}.$$

- ▶ Goal: Rescale our inputs so that
 - ▶ (Isotropy) $\sum p_i B_i^* B_i = I$.

And now, back to our regularly scheduled programming

$$\text{BL}(B, p) = \sqrt{\sup_{A_1, \dots, A_m \succ 0} \frac{\prod_i \det(A_i)^{p_i}}{\det(\sum p_i \cdot B_i^* A_i B_i)}}.$$

- ▶ Goal: Rescale our inputs so that
 - ▶ (Isotropy) $\sum p_i B_i^* B_i = I$.
 - ▶ (Projection) $B_i B_i^* = I$ for each i .

Iteration of Operator Rescaling

- ▶ Sinkhorn: Alternately apply the following two procedures:

Iteration of Operator Rescaling

- ▶ Sinkhorn: Alternately apply the following two procedures:
 - ▶ (Isotropy Normalization)

Iteration of Operator Rescaling

- ▶ Sinkhorn: Alternately apply the following two procedures:
 - ▶ (Isotropy Normalization)
 - ▶ Let $M = \sum_i p_i B_i^* B_i$.

Iteration of Operator Rescaling

- ▶ Sinkhorn: Alternately apply the following two procedures:
 - ▶ (Isotropy Normalization)
 - ▶ Let $M = \sum_i p_i B_i^* B_i$.
 - ▶ Replace B_i with $B'_i = B_i M^{-1/2}$.

Iteration of Operator Rescaling

- ▶ Sinkhorn: Alternately apply the following two procedures:
 - ▶ (Isotropy Normalization)
 - ▶ Let $M = \sum_i p_i B_i^* B_i$.
 - ▶ Replace B_i with $B'_i = B_i M^{-1/2}$.
 - ▶ Then $\sum p_i (B'_i)^* B'_i = 1$, i.e. isotropy holds.

Iteration of Operator Rescaling

- ▶ Sinkhorn: Alternately apply the following two procedures:
 - ▶ (Isotropy Normalization)
 - ▶ Let $M = \sum_i p_i B_i^* B_i$.
 - ▶ Replace B_i with $B'_i = B_i M^{-1/2}$.
 - ▶ Then $\sum p_i (B'_i)^* B'_i = 1$, i.e. isotropy holds.
 - ▶ (Projection Normalization)

Iteration of Operator Rescaling

- ▶ Sinkhorn: Alternately apply the following two procedures:
 - ▶ (Isotropy Normalization)
 - ▶ Let $M = \sum_i p_i B_i^* B_i$.
 - ▶ Replace B_i with $B'_i = B_i M^{-1/2}$.
 - ▶ Then $\sum p_i (B'_i)^* B'_i = 1$, i.e. isotropy holds.
 - ▶ (Projection Normalization)
 - ▶ Let $M_i = B_i B_i^*$.

Iteration of Operator Rescaling

- ▶ Sinkhorn: Alternately apply the following two procedures:
 - ▶ (Isotropy Normalization)
 - ▶ Let $M = \sum_i p_i B_i^* B_i$.
 - ▶ Replace B_i with $B'_i = B_i M^{-1/2}$.
 - ▶ Then $\sum p_i (B'_i)^* B'_i = 1$, i.e. isotropy holds.
 - ▶ (Projection Normalization)
 - ▶ Let $M_i = B_i B_i^*$.
 - ▶ Replace B_i with $B'_i = M_i^{-1/2} B_i$.

Iteration of Operator Rescaling

- ▶ Sinkhorn: Alternately apply the following two procedures:
 - ▶ (Isotropy Normalization)
 - ▶ Let $M = \sum_i p_i B_i^* B_i$.
 - ▶ Replace B_i with $B'_i = B_i M^{-1/2}$.
 - ▶ Then $\sum p_i (B'_i)^* B'_i = 1$, i.e. isotropy holds.
 - ▶ (Projection Normalization)
 - ▶ Let $M_i = B_i B_i^*$.
 - ▶ Replace B_i with $B'_i = M_i^{-1/2} B_i$.
 - ▶ Then $(B'_i)^* B'_i = I$ for each i .

Iteration of Operator Rescaling

- ▶ Sinkhorn: Alternately apply the following two procedures:
 - ▶ (Isotropy Normalization)
 - ▶ Let $M = \sum_i p_i B_i^* B_i$.
 - ▶ Replace B_i with $B'_i = B_i M^{-1/2}$.
 - ▶ Then $\sum p_i (B'_i)^* B'_i = 1$, i.e. isotropy holds.
 - ▶ (Projection Normalization)
 - ▶ Let $M_i = B_i B_i^*$.
 - ▶ Replace B_i with $B'_i = M_i^{-1/2} B_i$.
 - ▶ Then $(B'_i)^* B'_i = I$ for each i .
- ▶ We obtain a sequence $B \rightarrow B_1 \rightarrow B_2 \rightarrow \dots$

Iteration of Operator Rescaling

- ▶ **Two Key Facts Ensuring Convergence:**

Iteration of Operator Rescaling

► **Two Key Facts Ensuring Convergence:**

(1) $\text{BL}(B, p) \geq 1$ if (B, p) is partially normalized.

Iteration of Operator Rescaling

► **Two Key Facts Ensuring Convergence:**

- (1) $\text{BL}(B, p) \geq 1$ if (B, p) is partially normalized.
- (2) For some $r \leq \text{Poly}(\text{Bits}(B), d, n)$, if

$$\text{BL}(B_i, p) \geq 1 + \varepsilon,$$

then

$$\text{BL}(B_{i+1}, p) \leq (1 - C\varepsilon^r)\text{BL}(B_i, p).$$

Iteration of Operator Rescaling

► **Two Key Facts Ensuring Convergence:**

- (1) $\text{BL}(B, p) \geq 1$ if (B, p) is partially normalized.
- (2) For some $r \leq \text{Poly}(\text{Bits}(B), d, n)$, if

$$\text{BL}(B_i, p) \geq 1 + \varepsilon,$$

then

$$\text{BL}(B_{i+1}, p) \leq (1 - C\varepsilon^r)\text{BL}(B_i, p).$$

- Thus convergence occurs as with Sinkhorn iteration provided that $\text{BL}(B, p) < \infty$.

Iteration of Operator Rescaling

► **Two Key Facts Ensuring Convergence:**

- (1) $\text{BL}(B, p) \geq 1$ if (B, p) is partially normalized.
- (2) For some $r \leq \text{Poly}(\text{Bits}(B), d, n)$, if

$$\text{BL}(B_i, p) \geq 1 + \varepsilon,$$

then

$$\text{BL}(B_{i+1}, p) \leq (1 - C\varepsilon^r)\text{BL}(B_i, p).$$

- Thus convergence occurs as with Sinkhorn iteration provided that $\text{BL}(B, p) < \infty$.
- (1) and (2) follow from techniques in the study of *positive operators*.

Positive Operators

- ▶ A linear map $T : M_n \rightarrow M_n$ is *completely positive* if there are $n \times n$ matrices B_1, \dots, B_K and $p_i > 0$ such that

$$T(A) = \sum p_i B_i A B_i^*.$$

Positive Operators

- ▶ A linear map $T : M_n \rightarrow M_n$ is *completely positive* if there are $n \times n$ matrices B_1, \dots, B_K and $p_i > 0$ such that

$$T(A) = \sum p_i B_i A B_i^*.$$

- ▶ $T : M_n \rightarrow M_n$ is *positive* if $A \succeq 0$ implies $T(A) \succeq 0$.

Positive Operators

- ▶ A linear map $T : M_n \rightarrow M_n$ is *completely positive* if there are $n \times n$ matrices B_1, \dots, B_K and $p_i > 0$ such that

$$T(A) = \sum p_i B_i A B_i^*.$$

- ▶ $T : M_n \rightarrow M_n$ is *positive* if $A \succeq 0$ implies $T(A) \succeq 0$.
- ▶ Can reduce the study of non-negative matrices to positive operators: For a non-negative matrix S , $T(A)$ is the diagonal matrix whose entries are precisely the vector Sa , where a is the vector formed from the diagonal entries of A .

Positive Operators

- ▶ A linear map $T : M_n \rightarrow M_n$ is *completely positive* if there are $n \times n$ matrices B_1, \dots, B_K and $p_i > 0$ such that

$$T(A) = \sum p_i B_i A B_i^*.$$

- ▶ $T : M_n \rightarrow M_n$ is *positive* if $A \succeq 0$ implies $T(A) \succeq 0$.
- ▶ Can reduce the study of non-negative matrices to positive operators: For a non-negative matrix S , $T(A)$ is the diagonal matrix whose entries are precisely the vector Sa , where a is the vector formed from the diagonal entries of A .
- ▶ Given T , we have $T^*(A) = \sum p_i B_i^* A B_i$.

Further Connections

- For simplicity, look at Brascamp Lieb where all spaces have the same dimension (all B_i are square matrices).

Further Connections

- ▶ For simplicity, look at Brascamp Lieb where all spaces have the same dimension (all B_i are square matrices).
- ▶ $\text{BL}(B, p) < \infty$ can only hold if $\sum p_i = 1$.

Further Connections

- ▶ For simplicity, look at Brascamp Lieb where all spaces have the same dimension (all B_i are square matrices).
- ▶ $\text{BL}(B, p) < \infty$ can only hold if $\sum p_i = 1$.
- ▶ Consider optimizing the quantity

$$\inf_{A \succ 0} \frac{\det(\sum p_i B_i^* A B_i)}{\det(A)}$$

analogous to

$$\text{BL}(B, p) = \sup_{A_1, \dots, A_m \succ 0} \sqrt{\frac{\prod_i \det(A_i)^{p_i}}{\det(\sum p_i \cdot B_i^* A_i B_i)}},$$

if all A_i are equal.

Capacity of Operators

$$\inf_{A \succ 0} \frac{\det(\sum p_i B_i^* A B_i)}{\det(A)}$$

Capacity of Operators

$$\inf_{A \succ 0} \frac{\det(\sum p_i B_i^* A B_i)}{\det(A)}$$

- (Gurvits, 2004) The *capacity* of $T : M_n \rightarrow M_n$ is

$$\text{Cap}(T) = \inf_{A \succ 0} \frac{\det(TA)}{\det(A)}.$$

Capacity of Operators

$$\inf_{A \succ 0} \frac{\det(\sum p_i B_i^* A B_i)}{\det(A)}$$

- ▶ (Gurvits, 2004) The *capacity* of $T : M_n \rightarrow M_n$ is

$$\text{Cap}(T) = \inf_{A \succ 0} \frac{\det(TA)}{\det(A)}.$$

- ▶ For any Brascamp-Lieb data (B, p) , there exists a positive $T : M_n \rightarrow M_n$ and k such that $\text{Cap}(T) = 1/\text{BL}(B, p)^{2k}$.

Capacity of Operators

$$\inf_{A \succ 0} \frac{\det(\sum p_i B_i^* A B_i)}{\det(A)}$$

- ▶ (Gurvits, 2004) The *capacity* of $T : M_n \rightarrow M_n$ is

$$\text{Cap}(T) = \inf_{A \succ 0} \frac{\det(TA)}{\det(A)}.$$

- ▶ For any Brascamp-Lieb data (B, p) , there exists a positive $T : M_n \rightarrow M_n$ and k such that $\text{Cap}(T) = 1/\text{BL}(B, p)^{2k}$.
- ▶ Positive operators are well studied in the quantum information theory literature, so reduction of BL to this theory is useful.

Doubly Stochastic Positive Operators

- ▶ (Isotropy) Let $T(A) = \sum p_i B_i^* A B_i$.

Doubly Stochastic Positive Operators

- ▶ (Isotropy) Let $T(A) = \sum p_i B_i^* A B_i$.
 - ▶ $\sum p_i B_i^* B_i = I$ holds iff $T(I) = I$.

Doubly Stochastic Positive Operators

- ▶ (Isotropy) Let $T(A) = \sum p_i B_i^* A B_i$.
 - ▶ $\sum p_i B_i^* B_i = I$ holds iff $T(I) = I$.
- ▶ (Projection) Let $T(A) = B_i^* A B_i$.

Doubly Stochastic Positive Operators

- ▶ (Isotropy) Let $T(A) = \sum p_i B_i^* A B_i$.
 - ▶ $\sum p_i B_i^* B_i = I$ holds iff $T(I) = I$.
- ▶ (Projection) Let $T(A) = B_i^* A B_i$.
 - ▶ $B_i B_i^* = I$ if and only if $T^*(I) = I$.

Doubly Stochastic Positive Operators

- ▶ (Isotropy) Let $T(A) = \sum p_i B_i^* A B_i$.
 - ▶ $\sum p_i B_i^* B_i = I$ holds iff $T(I) = I$.
- ▶ (Projection) Let $T(A) = B_i^* A B_i$.
 - ▶ $B_i B_i^* = I$ if and only if $T^*(I) = I$.
- ▶ If (B, p) is a Brascamp-Lieb datum with associated operator $T : M_n \rightarrow M_n$, then (B, p) is geometric if and only if T is *doubly stochastic*, i.e. $T(I) = I$ and $T^*(I) = I$.

Operator Rescaling for Positive Operators

- ▶ If T is doubly stochastic, $\text{Cap}(T) = 1$.

Operator Rescaling for Positive Operators

- ▶ If T is doubly stochastic, $\text{Cap}(T) = 1$.
- ▶ We can rescale. If

$$T_{M_1 M_2}(A) = M_2^* T(M_1^* A M_1) M_2,$$

then $\text{Cap}(T_{M_1, M_2}) = \det(M_1)^2 \det(M_2)^2 \cdot \text{Cap}(T)$.

Operator Rescaling for Positive Operators

- ▶ If T is doubly stochastic, $\text{Cap}(T) = 1$.
- ▶ We can rescale. If

$$T_{M_1 M_2}(A) = M_2^* T(M_1^* A M_1) M_2,$$

then $\text{Cap}(T_{M_1, M_2}) = \det(M_1)^2 \det(M_2)^2 \cdot \text{Cap}(T)$.

- ▶ Sinkhorn says to iterate

$$T \mapsto T_{I, T(I)^{-1/2}} \quad \text{and} \quad T \mapsto T_{T^*(I)^{-1/2}, I}.$$

Operator Rescaling for Positive Operators

- ▶ If T is doubly stochastic, $\text{Cap}(T) = 1$.
- ▶ We can rescale. If

$$T_{M_1 M_2}(A) = M_2^* T(M_1^* A M_1) M_2,$$

then $\text{Cap}(T_{M_1, M_2}) = \det(M_1)^2 \det(M_2)^2 \cdot \text{Cap}(T)$.

- ▶ Sinkhorn says to iterate

$$T \mapsto T_{I, T(I)^{-1/2}} \quad \text{and} \quad T \mapsto T_{T^*(I)^{-1/2}, I}.$$

- ▶ If $\text{Cap}(T) > 0$, iteration yields a rescaling arbitrarily close to a doubly stochastic operator, in $\text{Poly}(\text{Bits}(B), 1/\varepsilon)$ time.

Upper Bounds For Capacity

Upper Bounds For Capacity

- ▶ To guarantee efficiency, we need to show that for partially normalized T , $\text{Cap}(T) \geq 1/e^{\text{Poly}(\text{Bits}(B))}$.

Upper Bounds For Capacity

- ▶ To guarantee efficiency, we need to show that for partially normalized T , $\text{Cap}(T) \geq 1/e^{\text{Poly}(\text{Bits}(B))}$.
- ▶ (Gurvits, 2004) If $T(A) = \sum B_i A B_i^*$, and $\det(\sum B_i) \neq 0$, then $\text{Cap}(T) \gtrsim (\text{Bits}(B) \cdot n)^{-O(n)}$.

Upper Bounds For Capacity

- ▶ To guarantee efficiency, we need to show that for partially normalized T , $\text{Cap}(T) \geq 1/e^{\text{Poly}(\text{Bits}(B))}$.
- ▶ (Gurvits, 2004) If $T(A) = \sum B_i A B_i^*$, and $\det(\sum B_i) \neq 0$, then $\text{Cap}(T) \gtrsim (\text{Bits}(B) \cdot n)^{-O(n)}$.
- ▶ If $\text{Cap}(T) > 0$, there is $d > 0$ and $d \times d$ matrices C_i s.t.
$$\det(\sum C_i \otimes B_i) \neq 0 \quad \text{and} \quad \text{Bits}(C) \leq \text{Poly}(d, \text{Bits}(B)).$$

Upper Bounds For Capacity

- ▶ To guarantee efficiency, we need to show that for partially normalized T , $\text{Cap}(T) \geq 1/e^{\text{Poly}(\text{Bits}(B))}$.
- ▶ (Gurvits, 2004) If $T(A) = \sum B_i A B_i^*$, and $\det(\sum B_i) \neq 0$, then $\text{Cap}(T) \gtrsim (\text{Bits}(B) \cdot n)^{-O(n)}$.
- ▶ If $\text{Cap}(T) > 0$, there is $d > 0$ and $d \times d$ matrices C_i s.t.

$$\det(\sum C_i \otimes B_i) \neq 0 \quad \text{and} \quad \text{Bits}(C) \leq \text{Poly}(d, \text{Bits}(B)).$$

- ▶ 'Fairly simple' to show that if $S(A) = \sum C_i A C_i^*$, and $(S \otimes T)(A) = \sum (C_i \otimes B_i) A (C_i \otimes B_i)^*$, then

$$\text{Cap}(S \otimes T) \leq \text{Cap}(S)^n \text{Cap}(T)^d.$$

Upper Bounds For Capacity

- ▶ To guarantee efficiency, we need to show that for partially normalized T , $\text{Cap}(T) \geq 1/e^{\text{Poly}(\text{Bits}(B))}$.
- ▶ (Gurvits, 2004) If $T(A) = \sum B_i A B_i^*$, and $\det(\sum B_i) \neq 0$, then $\text{Cap}(T) \gtrsim (\text{Bits}(B) \cdot n)^{-O(n)}$.
- ▶ If $\text{Cap}(T) > 0$, there is $d > 0$ and $d \times d$ matrices C_i s.t.

$$\det(\sum C_i \otimes B_i) \neq 0 \quad \text{and} \quad \text{Bits}(C) \leq \text{Poly}(d, \text{Bits}(B)).$$

- ▶ 'Fairly simple' to show that if $S(A) = \sum C_i A C_i^*$, and $(S \otimes T)(A) = \sum (C_i \otimes B_i) A (C_i \otimes B_i)^*$, then

$$\text{Cap}(S \otimes T) \leq \text{Cap}(S)^n \text{Cap}(T)^d.$$

- ▶ Since $\text{Cap}(L) \leq 1$, it follows that

$$\text{Cap}(T) \geq \text{Cap}(S \otimes T)^{1/d} \gtrsim (\text{Poly}(d, \text{Bits}(B))n)^{-O(n)}.$$

Upper Bounds For Capacity

- ▶ To guarantee efficiency, we need to show that for partially normalized T , $\text{Cap}(T) \geq 1/e^{\text{Poly}(\text{Bits}(B))}$.
- ▶ (Gurvits, 2004) If $T(A) = \sum B_i A B_i^*$, and $\det(\sum B_i) \neq 0$, then $\text{Cap}(T) \gtrsim (\text{Bits}(B) \cdot n)^{-O(n)}$.
- ▶ If $\text{Cap}(T) > 0$, there is $d > 0$ and $d \times d$ matrices C_i s.t.

$$\det(\sum C_i \otimes B_i) \neq 0 \quad \text{and} \quad \text{Bits}(C) \leq \text{Poly}(d, \text{Bits}(B)).$$

- ▶ 'Fairly simple' to show that if $S(A) = \sum C_i A C_i^*$, and $(S \otimes T)(A) = \sum (C_i \otimes B_i) A (C_i \otimes B_i)^*$, then

$$\text{Cap}(S \otimes T) \leq \text{Cap}(S)^n \text{Cap}(T)^d.$$

- ▶ Since $\text{Cap}(L) \leq 1$, it follows that

$$\text{Cap}(T) \geq \text{Cap}(S \otimes T)^{1/d} \gtrsim (\text{Poly}(d, \text{Bits}(B))n)^{-O(n)}.$$

- ▶ Invariant theory shows we can choose $d \leq n^4[(n+1)!]^2$.

The Invariant Theory

The Invariant Theory

- ▶ We have a group action of $SL_n \times SL_n$ on tuples $B = (B_1, \dots, B_m)$, such that

$$(M, N) \circ B = (MB_1N, \dots, MB_mN).$$

The Invariant Theory

- ▶ We have a group action of $SL_n \times SL_n$ on tuples $B = (B_1, \dots, B_m)$, such that

$$(M, N) \circ B = (MB_1N, \dots, MB_mN).$$

- ▶ Invariant Theory: Find the ring R of all 'invariant polynomials' $f(B)$ such that

$$f((M, N) \circ B) = f(B)$$

for all $(M, N) \in SL_n \times SL_n$ and all tuples B .

The Invariant Theory

- ▶ We have a group action of $SL_n \times SL_n$ on tuples $B = (B_1, \dots, B_m)$, such that

$$(M, N) \circ B = (MB_1N, \dots, MB_mN).$$

- ▶ Invariant Theory: Find the ring R of all 'invariant polynomials' $f(B)$ such that

$$f((M, N) \circ B) = f(B)$$

for all $(M, N) \in SL_n \times SL_n$ and all tuples B .

- ▶ Note: $f_C(B) = \det(\sum C_i \otimes B_i)$ is an invariant homogeneous polynomial under this action for any C_i .

The Invariant Theory

- ▶ (Nayak and Subrahmanyam, 2010) R is a ring generated by the homogeneous polynomials $f_C(A) = \det(\sum C_i \circ A_i)$, for $d \times d$ matrices C_i , for all $d > 0$.

The Invariant Theory

- ▶ (Nayak and Subrahmanyam, 2010) R is a ring generated by the homogeneous polynomials $f_C(A) = \det(\sum C_i \circ A_i)$, for $d \times d$ matrices C_i , for all $d > 0$.
 - ▶ (1890) Hilbert showed that for a fairly general family of group actions, R is finitely generated, so we should expect there is some d_0 such that R is generated by the polynomials above for $d \leq d_0$. Killed the field for 100 years.

The Invariant Theory

- ▶ (Nayak and Subrahmanyam, 2010) R is a ring generated by the homogeneous polynomials $f_C(A) = \det(\sum C_i \circ A_i)$, for $d \times d$ matrices C_i , for all $d > 0$.
 - ▶ (1890) Hilbert showed that for a fairly general family of group actions, R is finitely generated, so we should expect there is some d_0 such that R is generated by the polynomials above for $d \leq d_0$. Killed the field for 100 years.
 - ▶ (Ivanyos, Qiao, Subrahmanyam, 2015)
 $d_0 \lesssim n^4[(n+1)!]^2$.

The Invariant Theory

- ▶ (Nayak and Subrahmanyam, 2010) R is a ring generated by the homogeneous polynomials $f_C(A) = \det(\sum C_i \circ A_i)$, for $d \times d$ matrices C_i , for all $d > 0$.
 - ▶ (1890) Hilbert showed that for a fairly general family of group actions, R is finitely generated, so we should expect there is some d_0 such that R is generated by the polynomials above for $d \leq d_0$. Killed the field for 100 years.
 - ▶ (Ivanyos, Qiao, Subrahmanyam, 2015)
 $d_0 \lesssim n^4[(n+1)!]^2$.
 - ▶ Suppose there are $d \times d$ matrices C_i (with d minimal) such that $f_C(B) \neq 0$.

The Invariant Theory

- ▶ (Nayak and Subrahmanyam, 2010) R is a ring generated by the homogeneous polynomials $f_C(A) = \det(\sum C_i \circ A_i)$, for $d \times d$ matrices C_i , for all $d > 0$.
 - ▶ (1890) Hilbert showed that for a fairly general family of group actions, R is finitely generated, so we should expect there is some d_0 such that R is generated by the polynomials above for $d \leq d_0$. Killed the field for 100 years.
 - ▶ (Ivanyos, Qiao, Subrahmanyam, 2015)
 $d_0 \lesssim n^4[(n+1)!]^2$.
 - ▶ Suppose there are $d \times d$ matrices C_i (with d minimal) such that $f_C(B) \neq 0$.
 - ▶ Find families of matrices $C(1), \dots, C(n)$ of dimension at most d_0 such that

$$f_C(B) = \sum c_\alpha f_{C(1)}(B)^{\alpha_1} \dots f_{C(n)}(B)^{\alpha_n}.$$

The Invariant Theory

- ▶ (Nayak and Subrahmanyam, 2010) R is a ring generated by the homogeneous polynomials $f_C(A) = \det(\sum C_i \circ A_i)$, for $d \times d$ matrices C_i , for all $d > 0$.
 - ▶ (1890) Hilbert showed that for a fairly general family of group actions, R is finitely generated, so we should expect there is some d_0 such that R is generated by the polynomials above for $d \leq d_0$. Killed the field for 100 years.
 - ▶ (Ivanyos, Qiao, Subrahmanyam, 2015)
 $d_0 \lesssim n^4[(n+1)!]^2$.
 - ▶ Suppose there are $d \times d$ matrices C_i (with d minimal) such that $f_C(B) \neq 0$.
 - ▶ Find families of matrices $C(1), \dots, C(n)$ of dimension at most d_0 such that

$$f_C(B) = \sum c_\alpha f_{C(1)}(B)^{\alpha_1} \dots f_{C(n)}(B)^{\alpha_n}.$$

- ▶ Since $f_C(B) \neq 0$, there must exist i with $f_{C(i)}(B) \neq 0$.

The Invariant Theory

- ▶ (Nayak and Subrahmanyam, 2010) R is a ring generated by the homogeneous polynomials $f_C(A) = \det(\sum C_i \circ A_i)$, for $d \times d$ matrices C_i , for all $d > 0$.
 - ▶ (1890) Hilbert showed that for a fairly general family of group actions, R is finitely generated, so we should expect there is some d_0 such that R is generated by the polynomials above for $d \leq d_0$. Killed the field for 100 years.
 - ▶ (Ivanyos, Qiao, Subrahmanyam, 2015)
 $d_0 \lesssim n^4[(n+1)!]^2$.
 - ▶ Suppose there are $d \times d$ matrices C_i (with d minimal) such that $f_C(B) \neq 0$.
 - ▶ Find families of matrices $C(1), \dots, C(n)$ of dimension at most d_0 such that

$$f_C(B) = \sum c_\alpha f_{C(1)}(B)^{\alpha_1} \dots f_{C(n)}(B)^{\alpha_n}.$$

- ▶ Since $f_C(B) \neq 0$, there must exist i with $f_{C(i)}(B) \neq 0$.
- ▶ Thus $d \leq d_0$.

Rank Decreasing Operators

$$\int_{\mathbf{R}^n} \prod_{i=1}^m |f_i(B_i x)|^{p_i} dx \leq \text{BL}(B, p) \cdot \prod_{i=1}^m \|f_i\|_{L^1(\mathbf{R}^{n_i})}^{p_i}.$$

Rank Decreasing Operators

$$\int_{\mathbf{R}^n} \prod_{i=1}^m |f_i(B_i x)|^{p_i} dx \leq \text{BL}(B, p) \cdot \prod_{i=1}^m \|f_i\|_{L^1(\mathbf{R}^{n_i})}^{p_i}.$$

- (Bennett et al, 2008) implies that $\text{BL}(B, p) < \infty$ if and only if $\sum p_i n_i = n$, and for any subspace $V \subset \mathbf{R}^n$,

$$\dim(V) \leq \sum p_i \dim(B_i V).$$

Rank Decreasing Operators

$$\int_{\mathbf{R}^n} \prod_{i=1}^m |f_i(B_i x)|^{p_i} dx \leq \text{BL}(B, p) \cdot \prod_{i=1}^m \|f_i\|_{L^1(\mathbf{R}^{n_i})}^{p_i}.$$

- ▶ (Bennett et al, 2008) implies that $\text{BL}(B, p) < \infty$ if and only if $\sum p_i n_i = n$, and for any subspace $V \subset \mathbf{R}^n$,

$$\dim(V) \leq \sum p_i \dim(B_i V).$$

- ▶ An operator $T : M_n \rightarrow M_n$ is *rank non-decreasing* if for any $A \succeq 0$, $\text{Rank}(TA) \geq \text{Rank}(A)$.

Rank Decreasing Operators

$$\int_{\mathbf{R}^n} \prod_{i=1}^m |f_i(B_i x)|^{p_i} dx \leq \text{BL}(B, p) \cdot \prod_{i=1}^m \|f_i\|_{L^1(\mathbf{R}^{n_i})}^{p_i}.$$

- ▶ (Bennett et al, 2008) implies that $\text{BL}(B, p) < \infty$ if and only if $\sum p_i n_i = n$, and for any subspace $V \subset \mathbf{R}^n$,

$$\dim(V) \leq \sum p_i \dim(B_i V).$$

- ▶ An operator $T : M_n \rightarrow M_n$ is *rank non-decreasing* if for any $A \succeq 0$, $\text{Rank}(TA) \geq \text{Rank}(A)$.
- ▶ (Gurvits, 2004) $T : M_n \rightarrow M_n$ is rank non-decreasing if and only if $\text{Cap}(T) > 0$.

Proof Idea

(Gurvits, 2004) $T : M_n \rightarrow M_n$ is rank non-decreasing if and only if $\text{Cap}(T) > 0$.

Proof Idea

(Gurvits, 2004) $T : M_n \rightarrow M_n$ is rank non-decreasing if and only if $\text{Cap}(T) > 0$.

- ▶ Results from (Gurvits and Samorodnitsky, 2002) show the result is true if $T(X) = \sum X_{ij} A_i$, where $A_i \succeq 0$. The general case can be reduced to this case.

Proof Idea

(Gurvits, 2004) $T : M_n \rightarrow M_n$ is rank non-decreasing if and only if $\text{Cap}(T) > 0$.

- ▶ Results from (Gurvits and Samorodnitsky, 2002) show the result is true if $T(X) = \sum X_{ii} A_i$, where $A_i \succeq 0$. The general case can be reduced to this case.
- ▶ Given an orthonormal basis $U = \{u_1, \dots, u_N\}$, we define the *decoherence operator* $D_U(A) = \sum \langle Au_i, u_i \rangle \cdot u_i u_i^*$.

Proof Idea

(Gurvits, 2004) $T : M_n \rightarrow M_n$ is rank non-decreasing if and only if $\text{Cap}(T) > 0$.

- ▶ Results from (Gurvits and Samorodnitsky, 2002) show the result is true if $T(X) = \sum X_{ii} A_i$, where $A_i \succeq 0$. The general case can be reduced to this case.
- ▶ Given an orthonormal basis $U = \{u_1, \dots, u_N\}$, we define the *decoherence operator* $D_U(A) = \sum \langle Au_i, u_i \rangle \cdot u_i u_i^*$.
- ▶ Define $T_U = D_U \circ T$.

Proof Idea

(Gurvits, 2004) $T : M_n \rightarrow M_n$ is rank non-decreasing if and only if $\text{Cap}(T) > 0$.

- ▶ Results from (Gurvits and Samorodnitsky, 2002) show the result is true if $T(X) = \sum X_{ii} A_i$, where $A_i \succeq 0$. The general case can be reduced to this case.
- ▶ Given an orthonormal basis $U = \{u_1, \dots, u_N\}$, we define the *decoherence operator* $D_U(A) = \sum \langle Au_i, u_i \rangle \cdot u_i u_i^*$.
- ▶ Define $T_U = D_U \circ T$.
- ▶ Result follows from the following two facts:

Proof Idea

(Gurvits, 2004) $T : M_n \rightarrow M_n$ is rank non-decreasing if and only if $\text{Cap}(T) > 0$.

- ▶ Results from (Gurvits and Samorodnitsky, 2002) show the result is true if $T(X) = \sum X_{ii} A_i$, where $A_i \succeq 0$. The general case can be reduced to this case.
- ▶ Given an orthonormal basis $U = \{u_1, \dots, u_N\}$, we define the *decoherence operator* $D_U(A) = \sum \langle Au_i, u_i \rangle \cdot u_i u_i^*$.
- ▶ Define $T_U = D_U \circ T$.
- ▶ Result follows from the following two facts:
 - (1) T is rank non-decreasing if and only if T_U is rank non-decreasing for all U .

Proof Idea

(Gurvits, 2004) $T : M_n \rightarrow M_n$ is rank non-decreasing if and only if $\text{Cap}(T) > 0$.

- ▶ Results from (Gurvits and Samorodnitsky, 2002) show the result is true if $T(X) = \sum X_{ii} A_i$, where $A_i \succeq 0$. The general case can be reduced to this case.
- ▶ Given an orthonormal basis $U = \{u_1, \dots, u_N\}$, we define the *decoherence operator* $D_U(A) = \sum \langle Au_i, u_i \rangle \cdot u_i u_i^*$.
- ▶ Define $T_U = D_U \circ T$.
- ▶ Result follows from the following two facts:
 - (1) T is rank non-decreasing if and only if T_U is rank non-decreasing for all U .
 - (2) $\text{Cap}(T) = \inf_U \text{Cap}(T_U)$.

Proof Idea

(Gurvits, 2004) $T : M_n \rightarrow M_n$ is rank non-decreasing if and only if $\text{Cap}(T) > 0$.

- ▶ Results from (Gurvits and Samorodnitsky, 2002) show the result is true if $T(X) = \sum X_{ii} A_i$, where $A_i \succeq 0$. The general case can be reduced to this case.
- ▶ Given an orthonormal basis $U = \{u_1, \dots, u_N\}$, we define the *decoherence operator* $D_U(A) = \sum \langle Au_i, u_i \rangle \cdot u_i u_i^*$.
- ▶ Define $T_U = D_U \circ T$.
- ▶ Result follows from the following two facts:
 - (1) T is rank non-decreasing if and only if T_U is rank non-decreasing for all U .
 - (2) $\text{Cap}(T) = \inf_U \text{Cap}(T_U)$.
- ▶ To prove (1) and (2), use a simple trick: Given $A \succeq 0$, find U diagonalizing $T(A)$. Then $T(A) = T_U(A)$.

Thanks For Listening!