Approximation algorithms for noncommutative CSPs

Taro Spirig

QMATH, University of Copenhagen

Work in collaboration with Eric Culf (Waterloo) and Hamoon Mousavi (UC Berkeley)

arxiv.org/abs/2312.16765

Overview

- **Motivation:** Classically we know a lot about NP-hard CSPs and their approximability e.g. 3SAT or MaxCut. Quantumly (noncommutatively) we know very little.
- **Key idea:** Inspired by past classical algorithm (Goemans-Williamson) and quantum algorithm (Tsirelson), develop new framework to approximate noncommutative CSPs.
- **Main result:** 0.864-approximation algorithm for NC-Max-3-Cut.

Noncommutative-CSPs

Classical Max-Cut

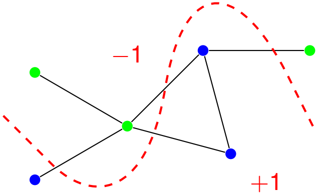


Figure: An instance of Max-Cut, Graph G

The classical value of the instance $\omega_c(G)$ is maximum partition value over assignments:

$$\text{maximize:} \quad \frac{1}{|E|} \sum_{(i,j)\in E} \frac{1 - x_i x_j}{2}$$

$$\text{subject to:} \quad x_i \in \{-1, +1\}.$$

Noncommutative Max-Cut

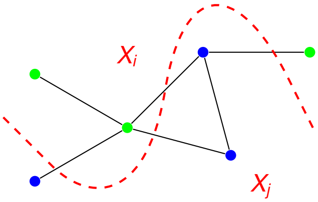


Figure: An instance of Max-Cut, Graph G

The noncommutative value of the instance $\omega_{\mathrm{nc}}(G)$:

maximize: $\frac{1}{|E|} \sum_{(i,j) \in E} \frac{1 - \langle X_i, X_j \rangle}{2}$

subject to: X_i unitary with eigenvalues ± 1.

Note:

- $\langle X, Y \rangle = \mathrm{tr}(X^* Y)$,
- X unitary iff $XX^* = X^* X = 1$,
- eigenvalues ± 1 same as $X^2 = 1$.

(Quantum Max-Cut)

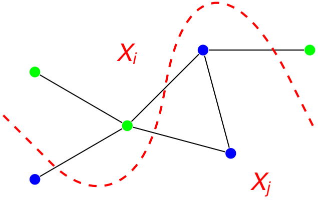


Figure: An instance of Max-Cut, Graph G

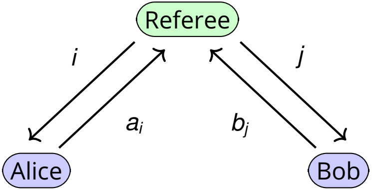
The quantum value of the instance $\omega_q(G)$ is:

$$\begin{aligned} \text{maximize:} \quad & \frac{1}{|E|} \sum_{(i,j)\in E} \frac{1 - \langle X_i, X_j \rangle}{2} \\ \text{subject to:} \quad & X_i \text{ unitary with} \\ & \text{eigenvalues } \pm 1, \\ & [X_i, X_j] = 0,\ (i,j) \in E. \end{aligned}$$

Motivation

Two-player games

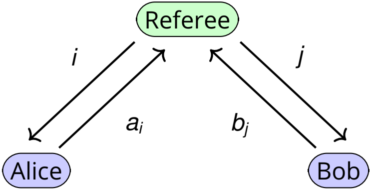
Every CSP equivalent to a two-players game, e.g. Max-Cut:



- Questions i, j are vertices, answers $a_i, b_j \in \pm 1$.
- If asked question $(i, j) \in E$, win with $a_i \neq b_j$.
- If asked question (i, i), win if $a_i = b_i$.
- winning probability of best strategy $\sim \omega_c(G)$

Two-player games

Every CSP equivalent to a two-players game, e.g. Max-Cut:

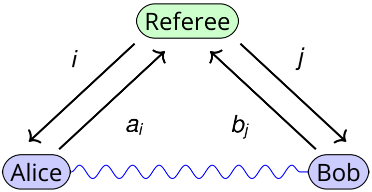


- Questions i, j are vertices, answers $a_i, b_j \in \pm 1$.
- If asked question $(i, j) \in E$, win with $a_i \neq b_j$.
- If asked question (i, i), win if $a_i = b_i$.
- winning probability of best strategy $\sim \omega_c(G)$

Remark: same thing as saying Max-Cut$_{c,s} \in$ MIP(2,1).

Nonlocal Games

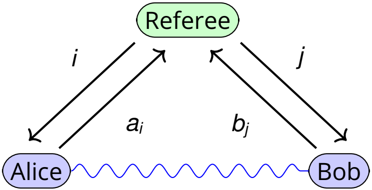
Similarly players could be quantum scientists and share entanglement:



- Strategies for players are generalization of probabilistic strategies: share entangled state $\rho \in M_{d^2}$, operators $A_i, B_j \in U_d$. Alice and Bob perform measurements to decide what to answer.
- Similarly, winning probability of best (synchronous) strategy $\sim \omega_{\mathrm{nc}}(G)$.

Nonlocal Games

Similarly players could be quantum scientists and share entanglement:



- Strategies for players are generalization of probabilistic strategies: share entangled state $\rho \in M_{d^2}$, operators $A_i, B_j \in U_d$. Alice and Bob perform measurements to decide what to answer.
- Similarly, winning probability of best (synchronous) strategy $\sim \omega_{\mathrm{nc}}(G)$.

Remark:

- Nonlocal games originally studied to understand entanglement but also useful in crypto, delegation of quantum computation, etc.
- Same thing as saying NC-Max-Cut$_{c,s} \in \mathrm{MIP}^*$.

NC-CSPs and nonlocal games

So far, motivated why NC-CSPs aren't just an arbitrary noncommutative generalization of CSPs, as NC-CSPs are equivalent to nonlocal games.

NC-CSPs and nonlocal games

So far, motivated why NC-CSPs aren't just an arbitrary noncommutative generalization of CSPs, as NC-CSPs are equivalent to nonlocal games.

What do we know about NC-CSPs? Not much.

- Hardness side: Analog of PCP theorem MIP*=RE. In general NC-CSPs are RE-hard to approximate within any constant factor.
- Algorithmic side: NC-Max-Cut is in fact easy. There is an algorithm by Tsirelson to compute $\omega_{\mathrm{nc}}(G)$ in polytime. (+ two more algorithms).

NC-CSPs and nonlocal games

So far, motivated why NC-CSPs aren't just an arbitrary noncommutative generalization of CSPs, as NC-CSPs are equivalent to nonlocal games.

What do we know about NC-CSPs? Not much.

- Hardness side: Analog of PCP theorem MIP*=RE. In general NC-CSPs are RE-hard to approximate within any constant factor.

- Algorithmic side: NC-Max-Cut is in fact easy. There is an algorithm by Tsirelson to compute $\omega_{\mathrm{nc}}(G)$ in polytime. (+ two more algorithms).

Remarks:

- Interesting insights into entanglement and noncommutative maths.
- Second result is surprising as Max-Cut is NP-hard. Intuitively NC-Max-Cut which is optimizing over matrices of any dimension should be harder (this is the intuition for MIP*=RE).

Main questions

Questions

Which NC-CSPs are RE-hard to approximate? Which ones are easy? Can we show tight approximation ratios?

More generally, we explore how noncommutativity reshapes the landscape of CSPs. In this work, we make progress on the algorithmic side.

Result

Specific question

If NC-Max-Cut is easy, what about its generalisations?

Result

Specific question

If NC-Max-Cut is easy, what about its generalisations?

NC-Max-3-Cut (3-coloring)

NC-Max-3-Cut is RE-hard but we show a polytime 0.864-approximation algorithm for it.

Here, we measure quality of algo $\mathcal{A}$ with approximation ratio $0.864 = \inf_I \frac{\mathcal{A}(I)}{OPT(I)}$.

Classically there is a 0.836-approximation algorithm for Max-3-Cut. We see that as with Max-Cut it is easier to compute the NC-value!

Max-3-Cut

Max-3-Cut

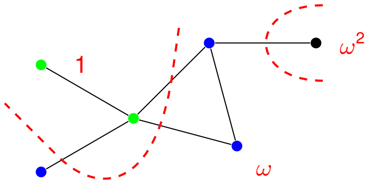


Figure: An instance of Max-3-Cut, Graph G

The classical value of the instance $\omega_c(G)$ is

$$\text{maximize:} \quad \sum_{i,j=1}^{N} w_{ij} \frac{2 - x_i^* x_j - x_j^* x_i}{3}$$

$$\text{subject to:} \quad x_i^3 = 1 \text{ and } x_i \in \mathbb{C}$$

Max-3-Cut

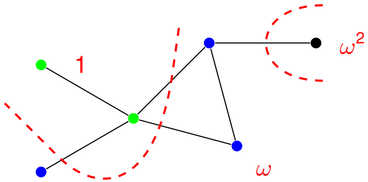


Figure: An instance of Max-3-Cut, Graph G

The classical value of the instance $\omega_c(G)$ is

$$\text{maximize:} \quad \sum_{i,j=1}^{N} w_{ij} \frac{2 - x_i^* x_j - x_j^* x_i}{3}$$

$$\text{subject to:} \quad x_i^3 = 1 \text{ and } x_i \in \mathbb{C}$$

Remarks:

- Real as $x_i^* x_j + x_j^* x_i = 2\Re(x_i^* x_j)$,

Max-3-Cut

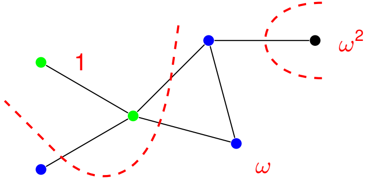


Figure: An instance of Max-3-Cut, Graph G

The classical value of the instance $\omega_c(G)$ is

$$\text{maximize:} \quad \sum_{i,j=1}^{N} w_{ij} \frac{2 - x_i^* x_j - x_j^* x_i}{3}$$

$$\text{subject to:} \quad x_i^3 = 1 \text{ and } x_i \in \mathbb{C}$$

Remarks:

- Real as $x_i^* x_j + x_j^* x_i = 2\Re(x_i^* x_j)$,
- As $x_i^* = x_i^2$, then

$$2 - x_i^* x_j - x_j^* x_i = 2 - x_i^* x_j - (x_i^* x_j)^2 = \begin{cases} 0, & x_i^* x_j = 1 \\ 2 - \omega^s - \omega^{2s} = 3, & x_i^* x_j = \omega^s,\ s = \{1, 2\}. \end{cases}$$

NC-Max-3-Cut

The noncommutative value $\omega_{\mathrm{nc}}(G)$ is

$$\begin{aligned} \text{maximize:} \quad & \sum_{i,j=1}^{N} w_{ij} \frac{2 - \langle X_i, X_j \rangle - \langle X_j, X_i \rangle}{3} \\ \text{subject to:} \quad & X_i^* X_i = X_i X_i^* = X_i^3 = 1. \end{aligned}$$

The algorithm

Tsirelson

Idea 1: Copy Tsirelson

What did he do? He showed Max-Cut equivalent to SDP relaxation.

OPT

$$\text{maximize:} \quad \sum_{i,j=1}^{N} w_{ij} \frac{1 - \langle X_i, X_j \rangle}{2}$$

$$\text{subject to:} \quad X_i^2 = X_i^* X_i = 1.$$

Tsirelson

Idea 1: Copy Tsirelson

What did he do? He showed Max-Cut equivalent to SDP relaxation.

OPT

$$\text{maximize:} \quad \sum_{i,j=1}^{N} w_{ij} \frac{1 - \langle X_i, X_j \rangle}{2}$$

$$\text{subject to:} \quad X_i^2 = X_i^* X_i = 1.$$

SDP

$$\text{maximize:} \quad \sum_{i,j=1}^{N} w_{ij} \frac{1 - \langle \vec{x}_i, \vec{x}_j \rangle}{2}$$

$$\text{subject to:} \quad \|\vec{x}_i\| = 1 \text{ and } \vec{x}_i \in \mathbb{R}^N.$$

Tsirelson

Idea 1: Copy Tsirelson

What did he do? He showed Max-Cut equivalent to SDP relaxation.

OPT

$$\text{maximize:} \quad \sum_{i,j=1}^{N} w_{ij} \frac{1 - \langle X_i, X_j \rangle}{2}$$

$$\text{subject to:} \quad X_i^2 = X_i^* X_i = 1.$$

SDP

$$\text{maximize:} \quad \sum_{i,j=1}^{N} w_{ij} \frac{1 - \langle \vec{x}_i, \vec{x}_j \rangle}{2}$$

$$\text{subject to:} \quad \|\vec{x}_i\| = 1 \text{ and } \vec{x}_i \in \mathbb{R}^N.$$

OPT $\Rightarrow$ SDP (OPT$\leq$SDP): easy

(Let matrix $K \in M_{n,d}$ such that ith row is $vec(X_i)^*$. Then $A = KK^* \in \mathrm{PSD}_n$ with $A_{i,i} = 1$ and $A_{i,j} = \langle X_i, X_j \rangle$.)

Tsirelson

OPT $\Leftarrow$ SDP (OPT$\geq$SDP): hard

Weyl-Brauer operators are defined as $\sigma_1 = X \otimes I \otimes \cdots$, $\sigma_2 = Y \otimes X \otimes I \cdots$, $\sigma_3 = Z \otimes X \otimes I \cdots$, ... They are pairwise anticommuting.

Vector-to-unitary construction Given vector $\vec{x}$, construct $U_{\vec{x}} = x_1\sigma_1 + \cdots + x_N\sigma_N$ which is Hermitian unitary and *isometric* $\langle U_{\vec{x}}, U_{\vec{y}} \rangle = \langle \vec{x}, \vec{y} \rangle$.

So given SDP vectors $\vec{x}_i$, $\vec{x}_j$, form the operators $U_{\vec{x}_i}$ and $U_{\vec{x}_j}$, which are feasible and have same value as SDP.

Tsirelson for NC-Max-3-Cut

Can we use the same idea to show **NC-Max-3-Cut=SDP?**

OPT

$$\text{maximize:} \quad \sum_{i,j=1}^{N} w_{ij} \frac{2 - \langle X_i, X_j \rangle - \langle X_j, X_i \rangle}{3}$$

$$\text{subject to:} \quad X_i^* X_i = X_i^3 = 1.$$

SDP

$$\text{maximize:} \quad \sum_{i,j=1}^{N} w_{ij} \frac{2 - \langle \vec{x}_i, \vec{x}_j \rangle - \langle \vec{x}_j, \vec{x}_i \rangle}{3}$$

$$\text{subject to:} \quad \|\vec{x}_i\| = 1 \text{ and } \vec{x}_i \in \mathbb{R}^N, \quad \langle \vec{x}_i, \vec{x}_j \rangle \geq -\frac{1}{2}.$$

Tsirelson for NC-Max-3-Cut

Can we use the same idea to show **NC-Max-3-Cut=SDP?**

OPT

maximize: $\displaystyle\sum_{i,j=1}^{N} w_{ij}\frac{2-\langle X_i, X_j\rangle - \langle X_j, X_i\rangle}{3}$

subject to: $X_i^* X_i = X_i^3 = 1.$

SDP

maximize: $\displaystyle\sum_{i,j=1}^{N} w_{ij}\frac{2-\langle \vec{x}_i, \vec{x}_j\rangle - \langle \vec{x}_j, \vec{x}_i\rangle}{3}$

subject to: $\|\vec{x}_i\| = 1$ and $\vec{x}_i \in \mathbb{R}^N$,

$$\langle \vec{x}_i, \vec{x}_j\rangle \geq -\frac{1}{2}.$$

OPT $\Rightarrow$ SDP: easy

($X = \frac{1}{2}(\langle X_i, X_j\rangle + \langle X_j, X_i\rangle)$ is real PSD, satisfies $X_{i,i} = 1$ and $X_{i,j} \geq -\frac{1}{2}$ since $\langle X_i, X_j\rangle \in \Delta_3$.)

Tsirelson for NC-Max-3-Cut

OPT$\Leftarrow$SDP?:

Vector-to-unitary fails: $U_{\vec{x}_i}$ is unitary but not order-3, i.e. $U_{\vec{x}_i}^3 \neq 1$.

Better group than Weyl-Brauer? No (at least no group that would work perfectly).

Rounding idea

Idea 2: rounding Tsirelson

Key observation: For spectral decomposition $X = \sum_s \alpha_s P_s$, closest order-3 is $\widetilde{X} = \sum_s \tilde{\alpha}_s P_s$ where $\tilde{\alpha}_s$ closest 3rd-root of unity to α_s.

Is that it? Use $\widetilde{U}_{\vec{x}_i}$? That does not do very good. But with some randomness à la Goemans-Williamson, it all works out!

Algorithm

Algorithm 1

1. Solve the SDP to obtain vectors $\vec{x}_1, \ldots, \vec{x}_N$.
2. Apply the vector-to-unitary construction to obtain unitary operators $X_i = U_{\vec{x}_i}$.
3. Sample a Haar random unitary U.
4. Return order-3 unitaries $\widetilde{UX}_1, \ldots, \widetilde{UX}_N$.

Algorithm

Algorithm 1

1. Solve the SDP to obtain vectors $\vec{x}_1, \ldots, \vec{x}_N$.
2. Apply the vector-to-unitary construction to obtain unitary operators $X_i = U_{\vec{x}_i}$.
3. Sample a Haar random unitary U.
4. Return order-3 unitaries $\widetilde{UX}_1, \ldots, \widetilde{UX}_N$.

Caveat: In fact we consider operators $X_i \otimes I_m$ in the $m \to \infty$ limit. Note also that it is not an efficient algorithm in the sense that it outputs efficiently an assignment. We only certify that there exists a good assignment whose value we can efficiently compute.

The analysis

Main theorem

Theorem: NC-Max-Cut(3) approximation ratio

Let X_i as in Algorithm 1, then

$$\mathbb{E}_U[2 - \langle \widetilde{UX_i}, \widetilde{UX_j} \rangle - \langle \widetilde{UX_j}, \widetilde{UX_i} \rangle] \geq 0.864 \left(2 - 2\langle \vec{x}_i, \vec{x}_j \rangle\right) .$$

Main theorem

Theorem: NC-Max-Cut(3) approximation ratio

Let X_i as in Algorithm 1, then

$$\mathbb{E}_U[2 - \langle \widetilde{UX_i}, \widetilde{UX_j} \rangle - \langle \widetilde{UX_j}, \widetilde{UX_i} \rangle] \geq 0.864 \left(2 - 2\langle \vec{x}_i, \vec{x}_j \rangle\right).$$

Remarks:

- $2 - \langle \vec{x}_i, \vec{x}_j \rangle - \langle \vec{x}_j, \vec{x}_i \rangle = 2 - 2\langle \vec{x}_i, \vec{x}_j \rangle$ as real vectors.
- Approximation ratio $\inf_I \frac{\mathcal{A}(I)}{OPT(I)} \geq \inf_I \frac{\mathcal{A}(I)}{SDP(I)} \geq 0.864$.

Spectral decomposition

Rewriting the problem

Let $UX_i = \sum_r \alpha_r P_r$ and $UX_j = \sum_s \beta_s Q_s$ as in algorithm. Then LHS of theorem equivalent to:

$$\mathbb{E}_U[2 - \langle \widetilde{UX_i}, \widetilde{UX_j} \rangle - \langle \widetilde{UX_j}, \widetilde{UX_i} \rangle] = \mathbb{E}_U \sum_{r,s} (2 - \tilde{\alpha}_r^* \tilde{\beta}_s - \tilde{\beta}_s^* \tilde{\alpha}_r) \langle P_r, Q_s \rangle$$

To analyze this quantity, need to define two objects: the fidelity and the relative distribution.

Fidelity

Definition: $\mathrm{fid}(\theta) = \mathbb{E}[2 - \tilde{\alpha}^* \tilde{\beta} - \tilde{\beta}^* \tilde{\alpha}]$, expectation is over uniformly random α, β on unit circle with fixed relative angle $e^{i\theta} = \alpha^* \beta$.

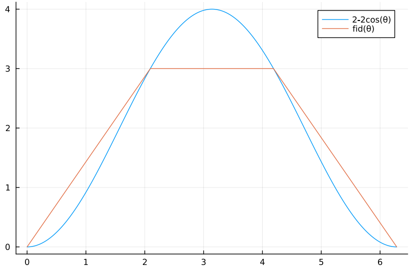


Figure: $\mathrm{fid}(\theta)$ vs. $2 - \alpha^* \beta - \beta^* \alpha = 2 - 2\cos(\theta)$.

Relative distribution

Definition

Relative distribution Δ_{x_i,x_j} is distribution of relative angle between pair of eigenvalues α_r and β_s, sampled with probability $\langle P_r, Q_s \rangle$.

Cauchy law

Theorem: Cauchy law

$X_i, X_j \in U_d(\mathbb{C})$ such that $\lambda = \langle X_i, X_j \rangle$. In the limit of $m \to \infty$, the relative distribution $\Delta_{X_i \otimes I_m, X_j \otimes I_m}$ converges to a wrapped Cauchy distribution Δ_λ.

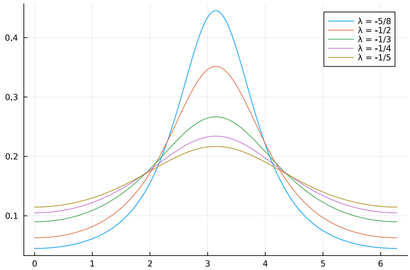


Figure: PDF of wrapped Cauchy distribution Δ_λ

Putting everything together

We had $D_{ij} := \mathbb{E}_U(2 - \langle \widetilde{UX_i}, \widetilde{UX_j} \rangle - \langle \widetilde{UX_j}, \widetilde{UX_i} \rangle) = \mathbb{E}_U \sum_{r,s}(2 - \tilde{\alpha}_r^* \tilde{\beta}_s - \tilde{\beta}_s^* \tilde{\alpha}_r)\langle P_r, Q_s \rangle$.

Putting everything together

We had $D_{ij} := \mathbb{E}_U(2 - \langle \widetilde{UX_i}, \widetilde{UX_j}\rangle - \langle \widetilde{UX_j}, \widetilde{UX_i}\rangle) = \mathbb{E}_U \sum_{r,s}(2 - \tilde{\alpha}_r^* \tilde{\beta}_s - \tilde{\beta}_s^* \tilde{\alpha}_r)\langle P_r, Q_s\rangle$.

By Haar invariance, multiply U by any root of unity $\zeta = e^{i\phi}$ which gives $\widetilde{UX_i} \to \widetilde{\zeta UX_i} = \sum_r \widetilde{\zeta \alpha_r} P_r$. Then

$$D_{ij} = \mathbb{E}_U \sum_{r,s} \frac{1}{2\pi} \int_0^{2\pi} (2 - \widetilde{\alpha_r e^{i\phi}}^* \widetilde{\beta_s e^{i\phi}} - \widetilde{\beta_s e^{i\phi}}^* \widetilde{\alpha_r e^{i\phi}}) d\phi \, \langle P_r, Q_s\rangle$$

$$= \mathbb{E}_U \sum_{r,s} \mathrm{fid}(\theta_{r,s})\langle P_r, Q_s\rangle = \int_\theta \mathrm{fid}(\theta) d\Delta_{X_i,X_j}(\theta)$$

Putting everything together

We had $D_{ij} := \mathbb{E}_U(2 - \langle \widetilde{UX_i}, \widetilde{UX_j}\rangle - \langle \widetilde{UX_j}, \widetilde{UX_i}\rangle) = \mathbb{E}_U \sum_{r,s}(2 - \tilde{\alpha}_r^* \tilde{\beta}_s - \tilde{\beta}_s^* \tilde{\alpha}_r)\langle P_r, Q_s\rangle$.

By Haar invariance, multiply U by any root of unity $\zeta = e^{i\phi}$ which gives $\widetilde{UX_i} \to \zeta \widetilde{UX_i} = \sum_r \widetilde{\zeta\alpha_r} P_r$. Then

$$D_{ij} = \mathbb{E}_U \sum_{r,s} \frac{1}{2\pi} \int_0^{2\pi} (2 - \widetilde{\alpha_r e^{i\phi}}^* \widetilde{\beta_s e^{i\phi}} - \widetilde{\beta_s e^{i\phi}}^* \widetilde{\alpha_r e^{i\phi}}) d\phi \, \langle P_r, Q_s\rangle$$

$$= \mathbb{E}_U \sum_{r,s} \mathrm{fid}(\theta_{r,s})\langle P_r, Q_s\rangle = \int_\theta \mathrm{fid}(\theta) d\Delta_{X_i,X_j}(\theta)$$

Hard to calculate. Instead consider the same reasoning but with operators $X_i \otimes I_m$ as $m \to \infty$. By Cauchy law:

$$D_{ij} = \int_\theta \mathrm{fid}(\theta) d\Delta_{X_i \otimes I_m, X_j \otimes I_m}(\theta) \to \int_\theta \mathrm{fid}(\theta) d\Delta_\lambda(\theta) =: D_\lambda.$$

The ratio

Compute

$$D_\lambda = \int_\theta \mathrm{fid}(\theta) d\Delta_\lambda(\theta) = 2 - \frac{18}{\pi^2} \sum_{n=1}^{\infty} \frac{\sin^2(\frac{\pi}{3} n)}{n^2} \lambda^n.$$

Thus, as $\lambda = \langle X_i, X_j \rangle = \langle \vec{x}_i, \vec{x}_j \rangle$ by the isometric property, we have

$$\inf_\lambda \frac{D_\lambda}{2 - 2\lambda} = \inf_\lambda \frac{2 - \frac{18}{\pi^2} \sum_{n=1}^{\infty} \frac{\sin^2(\frac{\pi}{3} n)}{n^2} \lambda^n.}{(2 - 2\lambda)} \geq 0.864,$$

as it is increasing, so only need to evaluate at smallest feasible value $\lambda = -\frac{1}{2}$.

Summary

- **Motivation:** Classically we know a lot about NP-hard CSPs and their approximability. Quantumly (noncommutatively) we know very little, apart from an analog of the PCP theorem (MIP*=RE) and an efficient algorithm for NC-Max-Cut (Tsirelson).
- **Key idea:** Inspired by past classical and noncommutative algorithms, develop new framework to approximate noncommutative CSPs.
- **Main result:** 0.864-approximation algorithm for NC-Max-3-Cut.

Future directions

Future directions

On the hardness front.

Tight hardness for Max-3-Cut

Could it be that 0.864 is tight for Max-3-Cut? I.e. is Max-3-Cut RE-hard to approximate within any factor greater than 0.864?

Future directions

On the hardness front.

Tight hardness for Max-3-Cut

Could it be that 0.864 is tight for Max-3-Cut? I.e. is Max-3-Cut RE-hard to approximate within any factor greater than 0.864?

Two recent results:

- Quantum CSPs seem to have the same landscape as classical CSPs [Mousavi, S 2024]. Can define quantum analogs to the Unique Games Conjecture and lift Fourier analytic techniques to the operator setting.
- Gapped reductions in the NC setting [Mastel, Slofstra 2024],[Culf, Mastel 2024]. There is $s > 0$ such that Max-3-Cut$_{1,s}$ is uncomputable.

Future directions

On the algorithmic front. There should be a better algorithm for NC-Max-k-Cut with $k \geq 5$.

Relation with classical CSPs. Given an operator assignment to a NC-Max-Cut instance, one can round it naturally to a classical assignment that achieves the best approximation ratio. Could this be a more general phenomenon? Can NC-CSPs shed some light on classical CSPs?

Thank you!