New Upper and Lower bounds for Entanglement Testing

Aram W. Harrow, Anand Natarajan, **Xiaodi Wu**

MIT Center for Theoretical Physics

MSR Redmond, May, 2015

Entanglement Detection

Definition (Separable and Entangled States)

A bi-partitie state $\rho \in \mathrm{D}(\mathcal{X} \otimes \mathcal{Y})$ is *separable* if $\exists$ dist. $\{p_i\}$,

$$\rho = \sum p_i \sigma_X^i \otimes \sigma_Y^i, \text{ s.t. } \sigma_X^i \in \mathrm{D}(\mathcal{X}), \sigma_Y^i \in \mathrm{D}(\mathcal{Y}).$$

Otherwise, ρ is *entangled*. Let Sep $\stackrel{\text{def}}{=}$ { separable states }.

Definition (Entanglement Detection)

A KEY problem: given the description of $\rho \in \mathrm{D}(\mathcal{X} \otimes \mathcal{Y})$, decide

Either $\rho \in$ Sep, or ρ is far away from Sep.

Entanglement Detection

Definition (Separable and Entangled States)

A bi-partitie state $\rho \in \mathrm{D}(\mathcal{X} \otimes \mathcal{Y})$ is *separable* if $\exists$ dist. $\{p_i\}$,

$$\rho = \sum p_i \sigma_X^i \otimes \sigma_Y^i, \text{ s.t. } \sigma_X^i \in \mathrm{D}(\mathcal{X}), \sigma_Y^i \in \mathrm{D}(\mathcal{Y}).$$

Otherwise, ρ is *entangled*. Let Sep $\stackrel{\text{def}}{=}$ { separable states }.

Definition (Entanglement Detection)

A KEY problem: given the description of $\rho \in \mathrm{D}(\mathcal{X} \otimes \mathcal{Y})$, decide

Either $\rho \in$ Sep, or ρ is far away from Sep.

Alternative Formulation

Definition (Weak Membership)

WMem($\epsilon, \|\cdot\|$) : for any $\rho \in \mathrm{D}(\mathcal{X} \otimes \mathcal{Y})$, decide either $\rho \in$ Sep or $\|\rho - \mathrm{Sep}\| \geq \epsilon$.

Via standard techniques in convex optimization, equivalent to

Definition (Weak Optimization)

WOpt(M, ϵ) : for any $M \in \mathrm{Herm}(\mathcal{X} \otimes \mathcal{Y})$, estimate the value of

$$\max_{\rho \in \mathrm{Sep}} \langle M, \rho \rangle ,$$

with additive error ϵ.

From now on, we focus on WOpt(M, ϵ).

Alternative Formulation

Definition (Weak Membership)

WMem$(\epsilon, \|\cdot\|)$: for any $\rho \in \mathrm{D}(\mathcal{X} \otimes \mathcal{Y})$, decide either $\rho \in$ Sep or $\|\rho - \text{Sep}\| \geq \epsilon$.

Via standard techniques in convex optimization, equivalent to

Definition (Weak Optimization)

WOpt(M, ϵ) : for any $M \in \mathrm{Herm}(\mathcal{X} \otimes \mathcal{Y})$, estimate the value of

$$\max_{\rho \in \text{Sep}} \langle M, \rho \rangle ,$$

with additive error ϵ.

From now on, we focus on WOpt(M, ϵ).

Alternative Formulation

Definition (Weak Membership)

WMem($\epsilon, \|\cdot\|$) : for any $\rho \in \mathrm{D}(\mathcal{X} \otimes \mathcal{Y})$, decide either $\rho \in \text{Sep}$ or $\|\rho - \text{Sep}\| \geq \epsilon$.

Via standard techniques in convex optimization, equivalent to

Definition (Weak Optimization)

WOpt(M, ϵ) : for any $M \in \mathrm{Herm}(\mathcal{X} \otimes \mathcal{Y})$, estimate the value of

$$\max_{\rho \in \text{Sep}} \langle M, \rho \rangle ,$$

with additive error ϵ.

From now on, we focus on WOpt(M, ϵ).

The Problem: alternative formulation

Recall that $h_{\mathrm{Sep}(d)}(M)$ refers to

$$\max \langle \mathbf{M}, \rho \rangle \text{ s.t. } \rho \in \mathrm{Sep}(\mathcal{X} \otimes \mathcal{Y}).$$

For any $M \in \mathbb{C}^{d \times d}$, there exists $M' \in \mathbb{C}^{2d \times 2d}$ s.t.

$$h_{\mathrm{ProdSym}(2d)}(M') = \frac{1}{4} h_{\mathrm{Sep}(d)}(M),$$

where $\mathrm{ProdSym}(d, k) := \mathrm{conv}\{(|\psi\rangle\langle\psi|)^{\otimes 2} : |\psi\rangle \in B(\mathbb{C}^d)\}$. [HM]

REDUCE our problem to the mathematically simpler $h_{\mathrm{ProdSym}(d)}$.

The Problem: alternative formulation

Recall that $h_{\mathrm{Sep}(d)}(M)$ refers to

$$\max \langle \mathbf{M}, \rho \rangle \text{ s.t. } \rho \in \mathrm{Sep}(\mathcal{X} \otimes \mathcal{Y}).$$

For any $M \in \mathbb{C}^{d\times d}$, there exists $M' \in \mathbb{C}^{2d\times 2d}$ s.t.

$$h_{\mathrm{ProdSym}(2d)}(M') = \frac{1}{4} h_{\mathrm{Sep}(d)}(M),$$

where $\mathrm{ProdSym}(d,k) := \mathrm{conv}\{(|\psi\rangle\langle\psi|)^{\otimes 2} : |\psi\rangle \in B(\mathbb{C}^d)\}$. [HM]

REDUCE our problem to the mathematically simpler $h_{\mathrm{ProdSym}(d)}$.

Reduce $h_{\mathrm{ProdSym}(d)}$ further

Let $|\psi\rangle = \sum_{i=1}^{d} a_i |i\rangle$ such that $\forall i, a_i \in \mathbb{C}$ and $\sum_i |a_i|^2 = 1$. It is easy to see that $h_{\mathrm{ProdSym}(d)}$ is equivalent to

$$\begin{aligned} \max_{a \in \mathbb{C}^d} \quad & \sum_{i_1, i_2, j_1, j_2} M_{(i_1,i_2),(j_1,j_2)} a^*_{i_1} a^*_{i_2} a_{j_1} a_{j_2} \\ \text{subject to} \quad & ||a||^2 = 1. \end{aligned} \tag{1}$$

Now reduce from $\mathbb{C}$ to $\mathbb{R}$ by observing:

- M is a Hermitian so the objective function is real.
- Decomposing the complex number into real and imaginary parts.

Reduce $h_{\mathrm{ProdSym}(d)}$ further

Let $|\psi\rangle = \sum_{i=1}^{d} a_i |i\rangle$ such that $\forall i, a_i \in \mathbb{C}$ and $\sum_i |a_i|^2 = 1$. It is easy to see that $h_{\mathrm{ProdSym}(d)}$ is equivalent to

$$\begin{aligned} \max_{a \in \mathbb{C}^d} \quad & \sum_{i_1, i_2, j_1, j_2} M_{(i_1,i_2),(j_1,j_2)} a_{i_1}^* a_{i_2}^* a_{j_1} a_{j_2} \\ \text{subject to} \quad & ||a||^2 = 1. \end{aligned} \tag{1}$$

Now reduce from $\mathbb{C}$ to $\mathbb{R}$ by observing:

- M is a Hermitian so the objective function is real.
- Decomposing the complex number into real and imaginary parts.

$h_{\mathrm{ProdSym}(n)}$ with real variables

By renaming, we arrive at the $h_{\mathrm{ProdSym}(n)}$ with real variables:

$$\begin{aligned} \max_{x \in \mathbb{R}^n} \quad & f_0(x) = \sum_{i_1, i_2, j_1, j_2} M_{(i_1,i_2),(j_1,j_2)} x_{i_1} x_{i_2} x_{j_1} x_{j_2} \\ \text{subject to} \quad & f_1(x) = \|x\|^2 - 1 = 0. \end{aligned} \tag{2}$$

REMARK: this is an instance of *polynomial optimization* problems with a homogenous degree 4 objective polynomial and a degree 2 constraint polynomial.

Connections

Quantum Information:

- Ground energy that is achieved by *non-entangled* states.
- *Mean-field* approximation in statistical quantum mechanics.
- *Positivity* test of quantum channels.
- *17 more examples* in quantum information in [HM10].

Quantum Complexity:

- Quantum Merlin-Arthur Game with Two-Provers (QMA(2)).

Classical Complexity:

- Unique Game Conjecture and Small-set Expansion. ($\ell_2 \to \ell_4$ norm)

Connections

Quantum Information:

- Ground energy that is achieved by *non-entangled* states.
- *Mean-field* approximation in statistical quantum mechanics.
- *Positivity* test of quantum channels.
- *17 more examples* in quantum information in [HM10].

Quantum Complexity:

- Quantum Merlin-Arthur Game with Two-Provers (QMA(2)).

Classical Complexity:

- Unique Game Conjecture and Small-set Expansion. ($\ell_2 \to \ell_4$ norm)

Connections

Quantum Information:

- Ground energy that is achieved by *non-entangled* states.
- *Mean-field* approximation in statistical quantum mechanics.
- *Positivity* test of quantum channels.
- *17 more examples* in quantum information in [HM10].

Quantum Complexity:

- Quantum Merlin-Arthur Game with Two-Provers (QMA(2)).

Classical Complexity:

- Unique Game Conjecture and Small-set Expansion. ($\ell_2 \to \ell_4$ norm)

Connections

Quantum Information:

- Ground energy that is achieved by *non-entangled* states.
- *Mean-field* approximation in statistical quantum mechanics.
- *Positivity* test of quantum channels.
- *17 more examples* in quantum information in [HM10].

Quantum Complexity:

- Quantum Merlin-Arthur Game with Two-Provers (QMA(2)).

Classical Complexity:

- Unique Game Conjecture and Small-set Expansion. ($\ell_2 \to \ell_4$ norm)

Connections

Quantum Information:

- Ground energy that is achieved by *non-entangled* states.
- *Mean-field* approximation in statistical quantum mechanics.
- *Positivity* test of quantum channels.
- *17 more examples* in quantum information in [HM10].

Quantum Complexity:

- Quantum Merlin-Arthur Game with Two-Provers (QMA(2)).

Classical Complexity:

- Unique Game Conjecture and Small-set Expansion. ($\ell_2 \to \ell_4$ norm)

Connections

Quantum Information:

- Ground energy that is achieved by *non-entangled* states.
- *Mean-field* approximation in statistical quantum mechanics.
- *Positivity* test of quantum channels.
- *17 more examples* in quantum information in [HM10].

Quantum Complexity:

- Quantum Merlin-Arthur Game with Two-Provers (QMA(2)).

Classical Complexity:

- Unique Game Conjecture and Small-set Expansion. ($\ell_2 \to \ell_4$ norm)

Early Attempts

Separability Criterions:

- Positive Partial Transpose (PPT) : $\rho^{T_{\mathcal{Y}}} = \rho$? [PH]
- Reduction Criterions: $I_{\mathcal{X}} \otimes \rho_Y \geq \rho$? [HH]
- $\cdots\cdots$
- FAILURE: any such test has arbitrarily large error. [BS]

Doherty-Parrilo-Spedalieri (DPS) hierarchy:

- ρ is k-extendible if $\exists$ *symmetric* $\sigma \in \mathrm{D}(\mathcal{X} \otimes \mathcal{Y}_1 \otimes \cdots \otimes \mathcal{Y}_k)$, $\forall i, \rho = \sigma_{XY_i}$.
- $\rho \in$ Sep if and only if ρ is k-extendible for any $k < \infty$.
- Semidefinite program (SDP) size exponential in k.

Early Attempts

Separability Criterions:

- Positive Partial Transpose (PPT) : $\rho^{T_{\mathcal{Y}}} = \rho$? [PH]
- Reduction Criterions: $I_{\mathcal{X}} \otimes \rho_Y \geq \rho$? [HH]
- $\cdots\cdots$
- **FAILURE**: any such test has arbitrarily large error. [BS]

Doherty-Parrilo-Spedalieri (DPS) hierarchy:

- ρ is k-extendible if $\exists$ *symmetric* $\sigma \in \mathrm{D}(\mathcal{X} \otimes \mathcal{Y}_1 \otimes \cdots \otimes \mathcal{Y}_k)$, $\forall i, \rho = \sigma_{XY_i}$.
- $\rho \in$ Sep if and only if ρ is k-extendible for any $k > 0$.
- Semidefinite program (SDP): size exponential in k

Early Attempts

Separability Criterions:

- Positive Partial Transpose (PPT) : $\rho^{T_{\mathcal{Y}}} = \rho$? [PH]
- Reduction Criterions: $I_{\mathcal{X}} \otimes \rho_Y \geq \rho$? [HH]
- $\cdots\cdots$
- FAILURE: any such test has arbitrarily large error. [BS]

Doherty-Parrilo-Spedalieri (DPS) hierarchy:

- ρ is k-extendible if $\exists$ *symmetric* $\sigma \in \mathrm{D}(\mathcal{X} \otimes \mathcal{Y}_1 \otimes \cdots \otimes \mathcal{Y}_k)$, $\forall i, \rho = \sigma_{XY_i}$.
- $\rho \in$ Sep if and only if ρ is k-extendible for any $k > 0$.
- Semidefinite program (SDP): size exponential in k.

Early Attempts

Separability Criterions:

- Positive Partial Transpose (PPT) : $\rho^{T_{\mathcal{Y}}} = \rho$? [PH]
- Reduction Criterions: $I_{\mathcal{X}} \otimes \rho_Y \geq \rho$? [HH]
- $\cdots\cdots$
- **FAILURE**: any such test has arbitrarily large error. [BS]

Doherty-Parrilo-Spedalieri (DPS) hierarchy:

- ρ is k-extendible if $\exists$ *symmetric* $\sigma \in \mathrm{D}(\mathcal{X} \otimes \mathcal{Y}_1 \otimes \cdots \otimes \mathcal{Y}_k)$, $\forall i, \rho = \sigma_{XY_i}$.
- $\rho \in$ Sep if and only if ρ is k-extendible for any $k \geq 0$.
- Semidefinite program (SDP) size exponential in k.

Early Attempts

Separability Criterions:

- Positive Partial Transpose (PPT) : $\rho^{T_{\mathcal{Y}}} = \rho$? [PH]
- Reduction Criterions: $I_{\mathcal{X}} \otimes \rho_Y \geq \rho$? [HH]
- $\cdots\cdots$
- **FAILURE**: any such test has arbitrarily large error. [BS]

Doherty-Parrilo-Spedalieri (DPS) hierarchy:

- ρ is k-extendible if $\exists$ *symmetric* $\sigma \in \mathrm{D}(\mathcal{X} \otimes \mathcal{Y}_1 \otimes \cdots \otimes \mathcal{Y}_k)$, $\forall i, \rho = \sigma_{XY_i}$.
- $\rho \in \mathrm{Sep}$ **if and only if** ρ is k-extendible for any $k \geq 0$.
- **Semidefinite program (SDP)**: size exponential in k.

Early Attempts

Separability Criterions:

- Positive Partial Transpose (PPT) : $\rho^{T_{\mathcal{Y}}} = \rho$? [PH]
- Reduction Criterions: $I_{\mathcal{X}} \otimes \rho_Y \geq \rho$? [HH]
- $\cdots\cdots$
- **FAILURE**: any such test has arbitrarily large error. [BS]

Doherty-Parrilo-Spedalieri (DPS) hierarchy:

- ρ is k-extendible if $\exists$ *symmetric* $\sigma \in \mathrm{D}(\mathcal{X} \otimes \mathcal{Y}_1 \otimes \cdots \otimes \mathcal{Y}_k)$, $\forall i, \rho = \sigma_{XY_i}$.
- $\rho \in$ Sep **if and only if** ρ is k-extendible for any $k \geq 0$.
- **Semidefinite program (SDP)**: size exponential in k.

Early Attempts

Separability Criterions:

- Positive Partial Transpose (PPT) : $\rho^{T_{\mathcal{Y}}} = \rho$? [PH]
- Reduction Criterions: $I_{\mathcal{X}} \otimes \rho_Y \geq \rho$? [HH]
- $\cdots\cdots$
- **FAILURE**: any such test has arbitrarily large error. [BS]

Doherty-Parrilo-Spedalieri (DPS) hierarchy:

- ρ is k-extendible if $\exists$ *symmetric* $\sigma \in \mathrm{D}(\mathcal{X} \otimes \mathcal{Y}_1 \otimes \cdots \otimes \mathcal{Y}_k)$, $\forall i, \rho = \sigma_{XY_i}$.
- $\rho \in$ Sep **if and only if** ρ is k-extendible for any $k \geq 0$.
- **Semidefinite program (SDP)**: size exponential in k.

Hardness

Let $h_{\mathrm{Sep}(d)}(M)$ denote the value of

$$\max \langle \mathbf{M}, \rho \rangle \text{ s.t. } \rho \in \mathrm{D}(\mathcal{X} \otimes \mathcal{Y}) \text{ is } \textit{separable},$$

where d refers to the dimension of $\mathcal{X} \otimes \mathcal{Y}$.

Hardness

- **NP-hard** to approximate $h_{\mathrm{Sep}(d)}(M)$ with additive error $\epsilon = 1/poly(d)$. [Gur03,Ioa07,Gha10], [deK08, LQNY09].
- Assuming **Exponential Time Hypothesis (ETH)**, for constant ϵ, approximate $h_{\mathrm{Sep}(d)}(M)$ needs $d^{\Omega(\log(d))}$ time. *via the connection to QMA(2).* [HM, AB+]

Hardness

Let $h_{\mathrm{Sep}(d)}(M)$ denote the value of

$$\max \langle \mathbf{M}, \rho \rangle \text{ s.t. } \rho \in \mathrm{D}(\mathcal{X} \otimes \mathcal{Y}) \text{ is } \textit{separable},$$

where d refers to the dimension of $\mathcal{X} \otimes \mathcal{Y}$.

Hardness

- **NP-hard** to approximate $h_{\mathrm{Sep}(d)}(M)$ with additive error $\epsilon = 1/poly(d)$. [Gur03,Ioa07,Gha10], [deK08, LQNY09].
- Assuming **Exponential Time Hypothesis (ETH)**, for constant ϵ, approximate $h_{\mathrm{Sep}(d)}(M)$ needs $d^{\Omega(log(d))}$ time. *via the connection to QMA(2).* [HM, AB+]

Hardness

Let $h_{\mathrm{Sep}(d)}(M)$ denote the value of

$$\max \langle \mathbf{M}, \rho \rangle \text{ s.t. } \rho \in \mathrm{D}\left(\mathcal{X} \otimes \mathcal{Y}\right) \text{ is } \textit{separable},$$

where d refers to the dimension of $\mathcal{X} \otimes \mathcal{Y}$.

Hardness

- **NP-hard** to approximate $h_{\mathrm{Sep}(d)}(M)$ with additive error $\epsilon = 1/poly(d)$. [Gur03,Ioa07,Gha10], [deK08, LQNY09].
- Assuming **Exponential Time Hypothesis (ETH)**, for constant ϵ, approximate $h_{\mathrm{Sep}(d)}(M)$ needs $d^{\Omega(\log(d))}$ time. *via the connection to QMA(2).* [HM, AB+]

Upper bounds

When $\epsilon = 1/poly(d)$

- DPS to $O(d/\sqrt{\epsilon})$ level: time $(d/\sqrt{\epsilon})^{O(d)} \to d^{O(d)}$. [NOP]
- Epsilon-net (brute-force): time $(1/\epsilon)^{O(d)} \to d^{O(d)}$.

When $\epsilon = \text{const}$

- DPS to $O(log(d)/\epsilon^2)$ level for 1-LOCC M: time $d^{O(log(d)/\epsilon^2)} \to d^{O(log(d))}$. [BYC, BH]
- Epsilon-net for 1-LOCC M or M with small $\|M\|_2$: time similar to above. [SW, BH]

REMARK: all DPS results correspond to variants of quantum de Finetti theorem.

Upper bounds

When $\epsilon = 1/poly(d)$

- DPS to $O(d/\sqrt{\epsilon})$ level: time $(d/\sqrt{\epsilon})^{O(d)} \to d^{O(d)}$. [NOP]
- Epsilon-net (brute-force): time $(1/\epsilon)^{O(d)} \to d^{O(d)}$.

When $\epsilon = \mathrm{const}$

- DPS to $O(log(d)/\epsilon^2)$ level for **1-LOCC** M: time $d^{O(log(d)/\epsilon^2)} \to d^{O(log(d))}$. [BYC, BH]
- Epsilon-net for **1-LOCC** M or M with small $\|M\|_2$: time similar to above. [SW, BH]

REMARK: all DPS results correspond to variants of quantum de Finetti theorem.

Upper bounds

When $\epsilon = 1/poly(d)$

- DPS to $O(d/\sqrt{\epsilon})$ level: time $(d/\sqrt{\epsilon})^{O(d)} \to d^{O(d)}$. [NOP]
- Epsilon-net (brute-force): time $(1/\epsilon)^{O(d)} \to d^{O(d)}$.

When $\epsilon = \mathrm{const}$

- DPS to $O(log(d)/\epsilon^2)$ level for **1-LOCC** M: time $d^{O(log(d)/\epsilon^2)} \to d^{O(log(d))}$. [BYC, BH]
- Epsilon-net for **1-LOCC** M **or** M **with small** $\|M\|_F$: time similar to above. [SW, BH]

REMARK: all DPS results correspond to variants of quantum de Finetti theorem.

Upper bounds

When $\epsilon = 1/poly(d)$

- DPS to $O(d/\sqrt{\epsilon})$ level: time $(d/\sqrt{\epsilon})^{O(d)} \rightarrow d^{O(d)}$. [NOP]
- Epsilon-net (brute-force): time $(1/\epsilon)^{O(d)} \rightarrow d^{O(d)}$.

When $\epsilon = \text{const}$

- DPS to $O(log(d)/\epsilon^2)$ level for **1-LOCC** M: time $d^{O(log(d)/\epsilon^2)} \rightarrow d^{O(log(d))}$. [BYC, BH]
- Epsilon-net for **1-LOCC** M **or** M **with small** $\|M\|_F$: time similar to above. [SW, BH]

REMARK: all DPS results correspond to variants of quantum de Finetti theorem.

Upper bounds

When $\epsilon = 1/poly(d)$

- DPS to $O(d/\sqrt{\epsilon})$ level: time $(d/\sqrt{\epsilon})^{O(d)} \to d^{O(d)}$. [NOP]
- Epsilon-net (brute-force): time $(1/\epsilon)^{O(d)} \to d^{O(d)}$.

When $\epsilon = \mathrm{const}$

- DPS to $O(log(d)/\epsilon^2)$ level for **1-LOCC** M: time $d^{O(log(d)/\epsilon^2)} \to d^{O(log(d))}$. [BYC, BH]
- Epsilon-net for **1-LOCC** M **or** M **with small** $\|M\|_F$: time similar to above. [SW, BH]

REMARK: all DPS results correspond to variants of quantum de Finetti theorem.

Upper bounds

When $\epsilon = 1/poly(d)$

- DPS to $O(d/\sqrt{\epsilon})$ level: time $(d/\sqrt{\epsilon})^{O(d)} \to d^{O(d)}$. [NOP]
- Epsilon-net (brute-force): time $(1/\epsilon)^{O(d)} \to d^{O(d)}$.

When $\epsilon = \text{const}$

- DPS to $O(log(d)/\epsilon^2)$ level for **1-LOCC** M: time $d^{O(log(d)/\epsilon^2)} \to d^{O(log(d))}$. [BYC, BH]
- Epsilon-net for **1-LOCC M or M with small $\|M\|_F$**: time similar to above. [SW, BH]

REMARK: all DPS results correspond to variants of quantum de Finetti theorem.

Upper bounds

When $\epsilon = 1/poly(d)$

- DPS to $O(d/\sqrt{\epsilon})$ level: time $(d/\sqrt{\epsilon})^{O(d)} \to d^{O(d)}$. [NOP]
- Epsilon-net (brute-force): time $(1/\epsilon)^{O(d)} \to d^{O(d)}$.

When $\epsilon = \text{const}$

- DPS to $O(log(d)/\epsilon^2)$ level for **1-LOCC** M: time $d^{O(log(d)/\epsilon^2)} \to d^{O(log(d))}$. [BYC, BH]
- Epsilon-net for **1-LOCC** M **or** M **with small** $\|M\|_{\mathbf{F}}$: time similar to above. [SW, BH]

REMARK: all DPS results correspond to variants of quantum de Finetti theorem.

Landscape

Table: Known results about approximating $h_{\mathrm{Sep}(d)}$ to error ϵ

Error ϵ	Lower bounds	Upper b. (DPS)	Upper b. (ϵ-net)
$1/poly(d)$	NP-hard	$(d/\sqrt{\epsilon})^{O(d)}$	$(1/\epsilon)^{O(d)}$
const	$d^{O(log(d))}$ (ETH)	$d^{O(log(d)/\epsilon^2)}$ (1-LOCC)	similar to left (1-LOCC)

REMARK: previous results focus on the *dependence on d*, which is sufficient for their purpose. However, the *dependence on ϵ* could be bad.

Landscape

Table: Known results about approximating $h_{\mathrm{Sep}(d)}$ to error ϵ

Error ϵ	Lower bounds	Upper b. (DPS)	Upper b. (ϵ-net)
$1/poly(d)$	NP-hard	$(d/\sqrt{\epsilon})^{O(d)}$	$(1/\epsilon)^{O(d)}$
const	$d^{O(log(d))}$ (ETH)	$d^{O(log(d)/\epsilon^2)}$ (1-LOCC)	similar to left (1-LOCC)

REMARK: previous results focus on the *dependence on d*, which is sufficient for their purpose. However, the *dependence on ϵ* could be bad.

Landscape

Table: Known results about approximating $h_{\mathrm{Sep}(d)}$ to error ϵ

Error ϵ	Lower bounds	Upper b. (DPS)	Upper b. (ϵ-net)
$1/poly(d)$	NP-hard	$poly(1/\epsilon)$	$poly(1/\epsilon)$
const	$d^{O(log(d))}$ (ETH)	$\exp(1/\epsilon)$ (1-LOCC)	similar to left (1-LOCC)

REMARK: previous results focus on the *dependence on d*, which is sufficient for their purpose. However, the *dependence on* ϵ could be bad. Is such dependence necessary?

Landscape

Table: Known results about approximating $h_{\mathrm{Sep}(d)}$ to error ϵ

Error ϵ	Lower bounds	Upper b. (DPS)	Upper b. (ϵ-net)
$1/poly(d)$	NP-hard	$poly(1/\epsilon)$	$poly(1/\epsilon)$
const	$d^{O(log(d))}$ (ETH)	$\exp(1/\epsilon)$ (1-LOCC)	similar to left (1-LOCC)

REMARK: previous results focus on the *dependence on d*, which is sufficient for their purpose. However, the *dependence on ϵ* could be bad. Is such dependence necessary?

Angle I: Error MATTERs!

Complexity could grow with $1/\epsilon$

- **Infinite translationally invariant Hamiltonian**: the complexity grows rapidly with $1/\epsilon$ even with fixed local dimension. [CPW]
- **Quantum Interactive Proof**: the complexity jumps from **PSPACE** to **EXP** with smaller ϵ. [IKW]

Will approximating $h_{\mathrm{Sep}(d)}$ be such a case?

REMARK: It is not clear how to improve the error dependence for either DPS or epsilon-net approach.

- DPS hard due to tightness of de Finetti and k-extendibility

Angle I: Error MATTERs!

Complexity could grow with $1/\epsilon$

- **Infinite translationally invariant Hamiltonian**: the complexity grows rapidly with $1/\epsilon$ even with fixed local dimension. [CPW]
- **Quantum Interactive Proof**: the complexity jumps from **PSPACE** to **EXP** with smaller ϵ. [IKW]

Will approximating $h_{\mathrm{Sep}(d)}$ be such a case?

REMARK: It is not clear how to improve the error dependence for either DPS or epsilon-net approach.

- DPS hard due to tightness of de Finetti and k-extendibility

Angle I: Error MATTERs!

Complexity could grow with $1/\epsilon$

- **Infinite translationally invariant Hamiltonian**: the complexity grows rapidly with $1/\epsilon$ even with fixed local dimension. [CPW]
- **Quantum Interactive Proof**: the complexity jumps from **PSPACE** to **EXP** with smaller ϵ. [IKW]

Will approximating $h_{\mathrm{Sep}(d)}$ be such a case?

REMARK: It is not clear how to improve the error dependence for either DPS or epsilon-net approach.

- DPS hard due to **tightness** of de Finetti and k-extendibility.

Angle I: Error MATTERs!

Complexity could grow with $1/\epsilon$

- **Infinite translationally invariant Hamiltonian**: the complexity grows rapidly with $1/\epsilon$ even with fixed local dimension. [CPW]
- **Quantum Interactive Proof**: the complexity jumps from **PSPACE** to **EXP** with smaller ϵ. [IKW]

Will approximating $h_{\mathrm{Sep}(d)}$ be such a case?

REMARK: It is not clear how to improve the error dependence for either DPS or epsilon-net approach.

- DPS hard due to **tightness** of de Finetti and k-extendibility.

Angle I: Error MATTERs!

Complexity could grow with $1/\epsilon$

- **Infinite translationally invariant Hamiltonian**: the complexity grows rapidly with $1/\epsilon$ even with fixed local dimension. [CPW]
- **Quantum Interactive Proof**: the complexity jumps from **PSPACE** to **EXP** with smaller ϵ. [IKW]

Will approximating $h_{\mathrm{Sep}(d)}$ be such a case?

REMARK: It is not clear how to improve the error dependence for either DPS or epsilon-net approach.

- DPS hard due to **tightness** of de Finetti and k-extendibility.

Main Result I:

Error dependence about $h_{\mathrm{Sep}(d)}$

- NO error dependence except *numerical* errors.
- For analytical purposes, there is *no error* at all.
- Numerically, the dependence is $\mathrm{polylog}(1/\epsilon)$, *exponential* improvement from best known $\mathrm{poly}(1/\epsilon)$, $\exp(1/\epsilon)$.

Moreover, the dependence on d **remains the same.**

Theorem (Main I)

There exist two algorithms that estimate $h_{\mathrm{Sep}(d)}(M)$ to error ϵ in time $\exp(\mathrm{poly}(d))\,\mathrm{poly}\log(1/\epsilon)$. similar for the multi-partite case.

Main Result I:

Error dependence about $h_{\mathrm{Sep}(d)}$

- NO error dependence except *numerical* errors.
- For analytical purposes, there is *no error* at all.
- Numerically, the dependence is $\mathrm{polylog}(1/\epsilon)$, *exponential* improvement from best known $\mathrm{poly}(1/\epsilon)$, $\exp(1/\epsilon)$.

Moreover, the dependence on d **remains the same.**

Theorem (Main I)

There exist two algorithms that estimate $h_{\mathrm{Sep}(d)}(M)$ to error ϵ in time $\exp(\mathrm{poly}(d))\,\mathrm{poly}\log(1/\epsilon)$. similar for the multi-partite case.

Main Result I:

Error dependence about $h_{\mathrm{Sep}(d)}$

- NO error dependence except *numerical* errors.
- For analytical purposes, there is *no error* at all.
- Numerically, the dependence is $\mathrm{polylog}(1/\epsilon)$, *exponential* improvement from best known $\mathrm{poly}(1/\epsilon)$, $\exp(1/\epsilon)$.

Moreover, the dependence on d **remains the same.**

Theorem (Main I)

There exist two algorithms that estimate $h_{\mathrm{Sep}(d)}(M)$ to error ϵ in time $\exp(\mathrm{poly}(d))\,\mathrm{poly}\log(1/\epsilon)$. similar for the multi-partite case.

Main Result I:

Error dependence about $h_{\mathrm{Sep}(d)}$

- NO error dependence except *numerical* errors.
- For analytical purposes, there is *no error* at all.
- Numerically, the dependence is $\mathrm{polylog}(1/\epsilon)$, *exponential* improvement from best known $\mathrm{poly}(1/\epsilon)$, $\exp(1/\epsilon)$.

Moreover, the dependence on d **remains the same.**

Theorem (Main I)

There exist two algorithms that estimate $h_{\mathrm{Sep}(d)}(M)$ to error ϵ in time $\exp(\mathrm{poly}(d)) \,\mathrm{poly}\log(1/\epsilon)$. similar for the multi-partite case.

Main Result I:

Error dependence about $h_{\mathrm{Sep}(d)}$

- NO error dependence except *numerical* errors.
- For analytical purposes, there is *no error* at all.
- Numerically, the dependence is $\mathrm{polylog}(1/\epsilon)$, *exponential* improvement from best known $\mathrm{poly}(1/\epsilon)$, $\exp(1/\epsilon)$.

Moreover, the dependence on d **remains the same.**

Theorem (Main I)

There exist two algorithms that estimate $h_{\mathrm{Sep}(d)}(M)$ to error ϵ in time $\exp(\mathrm{poly}(d))\,\mathrm{poly}\log(1/\epsilon)$. similar for the multi-partite case.

DPS+ hierarchy

DPS+ hierarchy level k for $h_{\mathrm{Sep}(d)}(M)$

$$
\begin{aligned}
\max_{\rho} \quad & \langle \rho_{X\mathcal{Y}_1}, M \rangle \\
\text{such that} \quad & \rho \in \mathrm{D}\left(\mathcal{X} \otimes \mathcal{Y}_1 \otimes \cdots \otimes \mathcal{Y}_k\right), \\
& \rho \text{ is symmetric on } \mathcal{Y}_1 \otimes \cdots \otimes \mathcal{Y}_k, \\
& \langle \rho, \Gamma_i \rangle = 0, \forall i. \quad \text{KKT conditions}
\end{aligned}
\tag{3}
$$

Remarks

- The new hierarchy is **exact** when $k = \exp(\mathrm{poly}(d))$.
- KKT conditions Γ_i depend on M.
- KKT conditions are written without multipliers.

DPS+ hierarchy

DPS+ hierarchy level k for $h_{\mathrm{Sep}(d)}(M)$

$$\begin{aligned} \max_{\rho} \quad & \langle \rho_{X\mathcal{Y}_1}, M \rangle \\ \text{such that} \quad & \rho \in \mathrm{D}\left(\mathcal{X} \otimes \mathcal{Y}_1 \otimes \cdots \otimes \mathcal{Y}_k\right), \\ & \rho \text{ is symmetric on } \mathcal{Y}_1 \otimes \cdots \otimes \mathcal{Y}_k, \\ & \langle \rho, \Gamma_i \rangle = 0, \forall i. \quad \text{KKT conditions} \end{aligned} \tag{3}$$

Remarks

- The new hierarchy is **exact** when $k = \exp(\mathrm{poly}(d))$.
- KKT conditions Γ_i depend on M.
- KKT conditions are written without *multipliers*.

DPS+ hierarchy

DPS+ hierarchy level k for $h_{\mathrm{Sep}(d)}(M)$

$$\begin{aligned} \max_{\rho} \quad & \langle \rho_{X\mathcal{Y}_1}, M \rangle \\ \text{such that} \quad & \rho \in \mathrm{D}(\mathcal{X} \otimes \mathcal{Y}_1 \otimes \cdots \otimes \mathcal{Y}_k), \\ & \rho \text{ is symmetric on } \mathcal{Y}_1 \otimes \cdots \otimes \mathcal{Y}_k, \\ & \langle \rho, \Gamma_i \rangle = 0, \forall i. \quad \text{KKT conditions} \end{aligned} \tag{3}$$

Remarks

- The new hierarchy is **exact** when $k = \exp(\mathrm{poly}(d))$.
- KKT conditions Γ_i depend on M.
- KKT conditions are written without *multipliers*.

DPS+ hierarchy

DPS+ hierarchy level k for $h_{\mathrm{Sep}(d)}(M)$

$$\begin{aligned} \max_{\rho} \quad & \langle \rho_{X\mathcal{Y}_1}, M \rangle \\ \text{such that} \quad & \rho \in \mathrm{D}\left(\mathcal{X} \otimes \mathcal{Y}_1 \otimes \cdots \otimes \mathcal{Y}_k\right), \\ & \rho \text{ is symmetric on } \mathcal{Y}_1 \otimes \cdots \otimes \mathcal{Y}_k, \\ & \langle \rho, \Gamma_i \rangle = 0, \forall i. \quad \text{KKT conditions} \end{aligned} \tag{3}$$

Remarks

- The new hierarchy is **exact** when $k = \exp(\mathrm{poly}(d))$.
- KKT conditions Γ_i depend on M.
- KKT conditions are written without *multipliers*.

Result II: Hardness w/o ASSUMPTIONs?

Will the hardness of $h_{\mathrm{Sep}(d)}$ for const ϵ hold w/o ETH?

Theorem (Main II.1)

DPS hierarchies (or general Sum-of-Squares SDP) require $\Omega(\log(d))$ levels to solve $h_{\mathrm{Sep}(d)}$ with constant precision.

Theorem (Main II.2)

Any SDP that estimate $h_{\mathrm{Sep}(d)}(M)$ with constant errors requires size $d^{\Omega(\log(d))}$.

Remark: Match $d^{\Omega(\log(d))}$ time bound when assuming ETH.

Result II: Hardness w/o ASSUMPTIONs?

Will the hardness of $h_{\mathrm{Sep}(d)}$ for const ϵ hold w/o ETH?

Theorem (Main II.1)

DPS hierarchies (or general Sum-of-Squares SDP) require $\Omega(\log(d))$ levels to solve $h_{\mathrm{Sep}(d)}$ with constant precision.

Theorem (Main II.2)

Any SDP that estimate $h_{\mathrm{Sep}(d)}(M)$ with constant errors requires size $d^{\Omega(\log(d))}$.

Remark: Match $d^{\Omega(\log(d))}$ time bound when assuming ETH.

Result II: Hardness w/o ASSUMPTIONs?

Will the hardness of $h_{\mathrm{Sep}(d)}$ for const ϵ hold w/o ETH?

Theorem (Main II.1)

DPS hierarchies (or general Sum-of-Squares SDP) require $\Omega(\log(d))$ levels to solve $h_{\mathrm{Sep}(d)}$ with constant precision.

Theorem (Main II.2)

Any SDP that estimate $h_{\mathrm{Sep}(d)}(M)$ with constant errors requires size $d^{\Omega(\log(d))}$.

Remark: Match $d^{\Omega(\log(d))}$ time bound when assuming ETH.

Result II: Hardness w/o ASSUMPTIONs?

Will the hardness of $h_{\mathrm{Sep}(d)}$ for const ϵ hold w/o ETH?

Theorem (Main II.1)

DPS hierarchies (or general Sum-of-Squares SDP) require $\Omega(\log(d))$ levels to solve $h_{\mathrm{Sep}(d)}$ with constant precision.

Theorem (Main II.2)

Any SDP that estimate $h_{\mathrm{Sep}(d)}(M)$ with constant errors requires size $d^{\Omega(\log(d))}$.

Remark: Match $d^{\Omega(\log(d))}$ time bound when assuming ETH.

Principle of Sum-of-Squares

One way to show that a polynomial $f(x)$ is *nonnegative* could be

$$f(x) = \sum a_i(x)^2 \geq 0.$$

Example

$$\begin{aligned} f(x) &= 2x^2 - 6x + 5 \\ &= (x^2 - 2x + 1) + (x^2 - 4x + 4) \\ &= (x-1)^2 + (x-2)^2 \geq 0. \end{aligned}$$

Such a decomposition is called a *sum of squares (SOS) certificate* for the non-negativity of f. The min degree, $\deg_{\mathrm{sos}}$.

Principle of SoS : constrained domain

Definition (Variety)

A set $V \subseteq \mathbb{C}^n$ is called an *algebraic variety* if $V = \{x \in \mathbb{C}^n : g_1(x) = \cdots = g_k(x) = 0\}$.

Non-negativity of $f(x)$ on V could be shown by

$$f(x) = \sum a_i(x)^2 + \sum b_j(x) g_j(x) \geq 0.$$

Question: whether all nonnegative polynomials on certain variety have a SOS certificate? Hilbert 17th problem!

Principle of SoS : constrained domain

Definition (Variety)

A set $V \subseteq \mathbb{C}^n$ is called an *algebraic variety* if $V = \{x \in \mathbb{C}^n : g_1(x) = \cdots = g_k(x) = 0\}$.

Non-negativity of $f(x)$ on V could be shown by

$$f(x) = \sum a_i(x)^2 + \sum b_j(x) g_j(x) \geq 0.$$

Question: whether all nonnegative polynomials on certain variety have a SOS certificate? Hilbert 17th problem!

Principle of SoS : constrained domain

Definition (Variety)

A set $V \subseteq \mathbb{C}^n$ is called an *algebraic variety* if $V = \{x \in \mathbb{C}^n : g_1(x) = \cdots = g_k(x) = 0\}$.

Non-negativity of $f(x)$ on V could be shown by

$$f(x) = \sum a_i(x)^2 + \sum b_j(x)g_j(x) \geq 0.$$

Question: whether all nonnegative polynomials on certain variety have a SOS certificate? Hilbert 17th problem!

SoS in Optimization

$$\begin{aligned} \max \quad & f(x) \\ \text{subject to} \quad & g_i(x) = 0 \quad \forall i \end{aligned} \tag{4}$$

is equivalent to (justified by *Positivstellensatz*)

$$\begin{aligned} \min \quad & \nu \\ \text{such that} \quad & \nu - f(x) = \sigma(x) + \sum_i b_i(x) g_i(x), \end{aligned} \tag{5}$$

where $\sigma(x)$ is SOS and $b_i(x)$ is any polynomial.

SoS relaxation: Lasserre/Parrilo Hierarchy

- If $\sigma(x), b_i(x)$ have *any* degrees (or $\deg_{\text{sos}}(v - f)$), then problem (5) is equivalent to problem (4).
- By bounding the degrees, we get the Lasserre/Parrilo hierarchy.

$$\begin{aligned} &\min && \nu \\ &\text{such that} && \nu - f(x) = \sigma(x) + \sum_i b_i(x) g_i(x), \end{aligned} \tag{6}$$

where $\sigma(x)$ is SOS and $b_i(x)$ is any polynomial and $\deg(\sigma(x))$, $\deg(b_i(x)g_i(x)) \leq 2D$.

SoS relaxation: Lasserre/Parrilo Hierarchy

- If $\sigma(x), b_i(x)$ have *any* degrees (or $\deg_{\text{sos}}(v - f)$), then problem (5) is equivalent to problem (4).
- By bounding the degrees, we get the Lasserre/Parrilo hierarchy.

$$\begin{aligned} \min \quad & \nu \\ \text{such that} \quad & \nu - f(x) = \sigma(x) + \sum_i b_i(x) g_i(x), \end{aligned} \tag{6}$$

where $\sigma(x)$ is SOS and $b_i(x)$ is any polynomial and $\deg(\sigma(x))$, $\deg(b_i(x)g_i(x)) \leq 2D$.

SoS relaxation: Lasserre/Parrilo Hierarchy

- If $\sigma(x), b_i(x)$ have *any* degrees (or $\deg_{\text{sos}}(v-f)$), then problem (5) is equivalent to problem (4).
- By bounding the degrees, we get the Lasserre/Parrilo hierarchy.

$$\begin{aligned} \min \quad & \nu \\ \text{such that} \quad & \nu - f(x) = \sigma(x) + \sum_i b_i(x) g_i(x), \end{aligned} \tag{6}$$

where $\sigma(x)$ is SOS and $b_i(x)$ is any polynomial and $\deg(\sigma(x))$, $\deg(b_i(x)g_i(x)) \leq 2D$.

SoS relaxation: Lasserre/Parrilo Hierarchy

- If $\sigma(x), b_i(x)$ have *any* degrees (or $\deg_{\mathrm{sos}}(v - f)$), then problem (5) is equivalent to problem (4).
- By bounding the degrees, we get the Lasserre/Parrilo hierarchy.

$$\begin{aligned} \min \quad & \nu \\ \text{such that} \quad & \nu - f(x) = \sigma(x) + \sum_i b_i(x) g_i(x), \end{aligned} \tag{6}$$

where $\sigma(x)$ is SOS and $b_i(x)$ is any polynomial and $\deg(\sigma(x))$, $\deg(b_i(x)g_i(x)) \leq 2D$.

Why it is a SDP?

Observation

- Any $p(x)$ (of degree $2D$) $= m^T Q m$, where m is the vector of monomials of degree up to $2D$ and Q is the coefficients.
- $p(x)$ is a SOS iff $Q \geq 0$.

$$\begin{aligned} \min_{\nu, b_{i\alpha} \in \mathbb{R}} \quad & \nu \\ \text{such that} \quad & \nu A_0 - F - \sum_{i\alpha} b_{i\alpha} G_{i\alpha} \geq 0. \end{aligned} \tag{7}$$

Complexity: $\mathrm{poly}(m) \, \mathrm{poly} \log(1/\epsilon)$, where $m = \binom{n+D}{D}$.

Why it is a SDP?

Observation

- Any $p(x)$ (of degree $2D$) $= m^T Qm$, where m is the vector of monomials of degree up to $2D$ and Q is the coefficients.
- $p(x)$ is a SOS iff $Q \geq 0$.

$$\begin{aligned} \min_{\nu, b_{i\alpha} \in \mathbb{R}} \quad & \nu \\ \text{such that} \quad & \nu A_0 - F - \sum_{i\alpha} b_{i\alpha} G_{i\alpha} \geq 0. \end{aligned} \tag{7}$$

Complexity: $\mathrm{poly}(m)\,\mathrm{poly}\log(1/\epsilon)$, where $m = \binom{n+D}{D}$.

Why it is a SDP?

Observation

- Any $p(x)$ (of degree $2D$) $= m^T Qm$, where m is the vector of monomials of degree up to $2D$ and Q is the coefficients.
- $p(x)$ is a SOS iff $Q \geq 0$.

$$\begin{aligned} \min_{\nu, b_{i\alpha} \in \mathbb{R}} \quad & \nu \\ \text{such that} \quad & \nu A_0 - F - \sum_{i\alpha} b_{i\alpha} G_{i\alpha} \geq 0. \end{aligned} \tag{7}$$

Complexity: $\operatorname{poly}(m) \operatorname{poly}\log(1/\epsilon)$, where $m = \binom{n+D}{D}$.

Dual of the SDP: moment

Dual of the SOS cone

- Let $\Sigma_{d,2D}$ be the cone of all PSD matrices representing SOS polynomials with degree up to $2D$.
- The dual cone $\Sigma^*_{d,2D}$ is moment $M_D(x) \geq 0$, where entry (α, β) of $M_d(x)$ is $\int x^{\alpha+\beta}\mu(dx), |\alpha|, |\beta| \leq d$.

Pseudo-expectation

- Expectation on moment $M_D(x)$ gives rise to *pseudo-expectation*.
- Behave similar to expectation for low-degree polynomials.

Dual of the SDP: moment

Dual of the SOS cone

- Let $\Sigma_{d,2D}$ be the cone of all PSD matrices representing SOS polynomials with degree up to $2D$.
- The dual cone $\Sigma^*_{d,2D}$ is moment $M_D(x) \geq 0$, where entry (α, β) of $M_d(x)$ is $\int x^{\alpha+\beta}\mu(dx), |\alpha|, |\beta| \leq d$.

Pseudo-expectation

- Expectation on moment $M_D(x)$ gives rise to *pseudo-expectation*.
- Behave similar to expectation for low-degree polynomials.

Full Symmetry $\implies$ DPS

Example

Now each entry is labelled with $((i,j),(k,l))$ for degree 4 case, i.e., $M_d(x) = \rho \in \mathrm{D}\left(\mathbb{C}^n \otimes \mathbb{C}^n\right)$.

$$\rho = \sum_{(i,j),(k,l)} x_i x_j x_k x_l \left|i\right\rangle \left|j\right\rangle \left\langle k\right| \left\langle l\right| .$$

Note that entry $((i,j),(k,l))$ and $((i,l),(k,j))$ have the same value $x_i x_j x_k x_l$. This is **PPT** condition. Similar for **DPS**.

Remark: more symmetry because in ProdSym. Flexible in choosing more or less symmetry.

Full Symmetry $\implies$ DPS

Example

Now each entry is labelled with $((i,j),(k,l))$ for degree 4 case, i.e., $M_d(x) = \rho \in \mathrm{D}\left(\mathbb{C}^n \otimes \mathbb{C}^n\right)$.

$$\rho = \sum_{(i,j),(k,l)} x_i x_j x_k x_l \left|i\right\rangle \left|j\right\rangle \left\langle k\right| \left\langle l\right| .$$

Note that entry $((i,j),(k,l))$ and $((i,l),(k,j))$ have the same value $x_i x_j x_k x_l$. This is **PPT** condition. Similar for **DPS**.

Remark: more symmetry because in ProdSym. Flexible in choosing more or less symmetry.

Karush-Kuhn-Tucker Conditions

For any optimization problem

$$\max f(x) \text{ s.t. } g_i(x) \leq 0, h_j(x) = 0, \forall i, j,$$

if x^* is a *local* optimizer, then $\exists \mu_i, \lambda_j$,

$$\begin{aligned} \nabla f(x^*) &= \sum \mu_i \nabla g_i(x^*) + \sum \lambda_j \nabla h_j(x^*) \\ g_i(x^*) &\leq 0, h_j(x^*) = 0, \\ \mu_i &\geq 0, \mu_i g_i(x^*) = 0. \end{aligned}$$

Remark: for convex optimization (our case), any global optimizer satisfies KKT.

Karush-Kuhn-Tucker Conditions

For any optimization problem

$$\max f(x) \text{ s.t. } g_i(x) \leq 0, h_j(x) = 0, \forall i, j,$$

if x^* is a *local* optimizer, then $\exists \mu_i, \lambda_j$,

$$\begin{aligned} \nabla f(x^*) &= \sum \mu_i \nabla g_i(x^*) + \sum \lambda_j \nabla h_j(x^*) \\ g_i(x^*) &\leq 0, h_j(x^*) = 0, \\ \mu_i &\geq 0, \mu_i g_i(x^*) = 0. \end{aligned}$$

Remark: for convex optimization (our case), any global optimizer satisfies KKT.

Our case

Recall our optimization problem is

$$\max f_0(x) \text{ s.t. } f_1(x) = 0.$$

The KKT condition is $\nabla f_0(x) = \lambda \nabla f_1(x)$, which is equivalent to

$$\operatorname{rank} \begin{pmatrix} \frac{\partial f_0(x)}{\partial x_1} & \frac{\partial f_1(x)}{\partial x_1} \\ \vdots & \vdots \\ \frac{\partial f_0(x)}{\partial x_{2n}} & \frac{\partial f_1(x)}{\partial x_{2n}} \end{pmatrix} < 2.$$

$$g_{ij}(x) = \frac{\partial f_0(x)}{\partial x_i}\frac{\partial f_1(x)}{\partial x_j} - \frac{\partial f_0(x)}{\partial x_j}\frac{\partial f_1(x)}{\partial x_i}, \quad \forall i, j$$

Our case

Recall our optimization problem is

$$\max f_0(x) \text{ s.t. } f_1(x) = 0.$$

The KKT condition is $\nabla f_0(x) = \lambda \nabla f_1(x)$, which is equivalent to

$$\text{rank} \begin{pmatrix} \frac{\partial f_0(x)}{\partial x_1} & \frac{\partial f_1(x)}{\partial x_1} \\ \vdots & \vdots \\ \frac{\partial f_0(x)}{\partial x_{2n}} & \frac{\partial f_1(x)}{\partial x_{2n}} \end{pmatrix} < 2.$$

$$g_{ij}(x) = \frac{\partial f_0(x)}{\partial x_i}\frac{\partial f_1(x)}{\partial x_j} - \frac{\partial f_0(x)}{\partial x_j}\frac{\partial f_1(x)}{\partial x_i}, \quad \forall i, j$$

Optimization Problem with KKT constraints

$$\begin{array}{lll} \min & \nu & \\ \text{such that} & \nu - f_0(x) \geq 0 & \\ & f_1(x) = 0 & \\ \text{KKT} & g_{ij}(x) = 0 & \forall\, 1 \leq i \neq j \leq 2d \end{array}$$

- Apply the degree bound D, we get the SoS SDP hierarchy.
- Will show finite convergence when $D = \exp(\mathrm{poly}(d))$. Then $m = \binom{d+D}{D} = \exp(\mathrm{poly}(d))$. Thus the final time is $\exp(\mathrm{poly}(d))\,\mathrm{poly}\log(1/\epsilon)$.

Optimization Problem with KKT constraints

$$\begin{array}{lll} \text{min} & \nu & \\ \text{such that} & \nu - f_0(x) \geq 0 & \\ & f_1(x) = 0 & \\ \text{KKT} & g_{ij}(x) = 0 & \forall\, 1 \leq i \neq j \leq 2d \end{array}$$

- Apply the degree bound D, we get the SoS SDP hierarchy.
- Will show finite convergence when $D = \exp(\text{poly}(d))$. Then $m = \binom{d+D}{D} = \exp(\text{poly}(d))$. Thus the final time is $\exp(\text{poly}(d)) \operatorname{poly}\log(1/\epsilon)$.

Proof Overview

- KKT conditions are necessary for *critical* points.
- KKT conditions imply **finite convergence** (tri-exponential or higher) for a **generic** optimization problem. [N, NR]
- Bring down the level for our problem to exponential.
  - KKT shrinks the feasible set to isolated points. (Bézout and Bertini)
  - Exponential level suffices. (Gröbner basis)
- Handle arbitrary inputs rather than generic ones.

Proof Overview

- KKT conditions are necessary for *critical* points.
- KKT conditions imply **finite convergence** (tri-exponential or higher) for a **generic** optimization problem. [N, NR]
- Bring down the level for our problem to exponential.
  - KKT shrinks the feasible set to **isolated** points. (Bézout and Bertini)
  - Exponential level suffices. (Gröbner basis)
- Handle arbitrary inputs rather than generic ones.

Proof Overview

- KKT conditions are necessary for *critical* points.
- KKT conditions imply **finite convergence** (tri-exponential or higher) for a **generic** optimization problem. [N, NR]
- Bring down the level for our problem to exponential.
  - KKT shrinks the feasible set to **isolated** points. (Bézout and Bertini)
  - Exponential level suffices. (Grobner basis)
- Handle arbitrary inputs rather than generic ones.

Proof Overview

- KKT conditions are necessary for *critical* points.
- KKT conditions imply **finite convergence** (tri-exponential or higher) for a **generic** optimization problem. [N, NR]
- Bring down the level for our problem to exponential.
  - KKT shrinks the feasible set to **isolated** points. (Bézout and Bertini)
  - Exponential level suffices. (Grobner basis)
- Handle arbitrary inputs rather than generic ones.

Proof Overview

- KKT conditions are necessary for *critical* points.
- KKT conditions imply **finite convergence** (tri-exponential or higher) for a **generic** optimization problem. [N, NR]
- Bring down the level for our problem to exponential.
  - KKT shrinks the feasible set to **isolated** points. (Bézout and Bertini)
  - Exponential level suffices. (Grobner basis)
- Handle arbitrary inputs rather than generic ones.

Proof Overview

- KKT conditions are necessary for *critical* points.
- KKT conditions imply **finite convergence** (tri-exponential or higher) for a **generic** optimization problem. [N, NR]
- Bring down the level for our problem to exponential.
  - KKT shrinks the feasible set to **isolated** points. (Bézout and Bertini)
  - Exponential level suffices. (Grobner basis)
- Handle arbitrary inputs rather than generic ones.

Generic input M

Theorem (Zero-dimensional of generic I_K)

For a generic M, $|V(I_K)| < \infty$ and I_K is zero-dimensional.

Theorem (Degree bound)

There exists $m = O(\exp(\mathrm{poly}(n)))$, s.t. for a generic M, $\epsilon > 0$,

$$v - f_0(x) + \epsilon = \sigma(x) + g(x),$$

where $\sigma(x)$ is SoS and $\deg(\sigma(x)) \leq m, g(x) \in I_K^m$.

Corollary (SDP solution)

Estimate $h_{\mathrm{ProdSym}(n)}(M)$ for a generic M to error ϵ needs $\exp(\mathrm{poly}(n))\mathrm{poly}\log(1/\epsilon)$.

Generic input M

Theorem (Zero-dimensional of generic I_K)

For a generic M, $|V(I_K)| < \infty$ and I_K is zero-dimensional.

Theorem (Degree bound)

There exists $m = O(\exp(\text{poly}(n)))$, s.t. for a generic M, $\epsilon > 0$,

$$v - f_0(x) + \epsilon = \sigma(x) + g(x),$$

where $\sigma(x)$ is SoS and $\deg(\sigma(x)) \leq m, g(x) \in I_K^m$.

Corollary (SDP solution)

Estimate $h_{\text{ProdSym}(n)}(M)$ for a generic M to error ϵ needs $\exp(\text{poly}(n))\text{poly}\log(1/\epsilon)$.

Generic input M

Theorem (Zero-dimensional of generic I_K)

For a generic M, $|V(I_K)| < \infty$ and I_K is zero-dimensional.

Theorem (Degree bound)

There exists $m = O(\exp(\text{poly}(n)))$, s.t. for a generic M, $\epsilon > 0$,

$$v - f_0(x) + \epsilon = \sigma(x) + g(x),$$

where $\sigma(x)$ is SoS and $\deg(\sigma(x)) \leq m, g(x) \in I_K^m$.

Corollary (SDP solution)

Estimate $h_{\text{ProdSym}(n)}(M)$ for a generic M to error ϵ needs $\exp(\text{poly}(n))\text{poly}\log(1/\epsilon)$.

Arbitrary input M

Observations

- Generic M is *dense*. The opt of SDP could be continuous.
- Issue: SOS SDP might be *infeasible* up to degree m for arbitrary input M.

Solutions

- Switch to the dual SDP (moment): satisfies Slater's condition, i.e., strictly feasible.
- For a generic M, by strong duality, $h_{\mathrm{ProdSym}(n)}(M) = OPT_{\mathrm{mom}}(M)$.
- For any input M, use the continuity of the dual SDP then.

Arbitrary input M

Observations

- Generic M is *dense*. The opt of SDP could be continuous.
- Issue: SOS SDP might be *infeasible* up to degree m for arbitrary input M.

Solutions

- Switch to the dual SDP (moment): satisfies Slater's condition, i.e, strictly feasible.
- For a generic M, by strong duality, $h_{\mathrm{ProdSym}(n)}(M) = OPT_{mom}(M)$.
- For any input M, use the continuity of the dual SDP then.

Arbitrary input M

Observations

- Generic M is *dense*. The opt of SDP could be continuous.
- Issue: SOS SDP might be *infeasible* up to degree m for arbitrary input M.

Solutions

- Switch to the dual SDP (moment): satisfies Slater's condition, i.e, strictly feasible.
- For a generic M, by strong duality, $h_{\mathrm{ProdSym}(n)}(M) = OPT_{\mathrm{mom}}(M)$.
- For any input M, use the continuity of the dual SDP then

Arbitrary input M

Observations

- Generic M is *dense*. The opt of SDP could be continuous.
- Issue: SOS SDP might be *infeasible* up to degree m for arbitrary input M.

Solutions

- Switch to the dual SDP (moment): satisfies Slater's condition, i.e, strictly feasible.
- For a generic M, by strong duality, $h_{\mathrm{ProdSym}(n)}(M) = OPT_{\mathrm{mom}}(M)$.
- For any input M, use the continuity of the dual SDP then.

Arbitrary input M

Observations

- Generic M is *dense*. The opt of SDP could be continuous.
- Issue: SOS SDP might be *infeasible* up to degree m for arbitrary input M.

Solutions

- Switch to the dual SDP (moment): satisfies Slater's condition, i.e, strictly feasible.
- For a generic M, by strong duality, $h_{\mathrm{ProdSym}(n)}(M) = OPT_{\mathrm{mom}}(M)$.
- For any input M, use the continuity of the dual SDP then.

Arbitrary input M

Observations

- Generic M is *dense*. The opt of SDP could be continuous.
- Issue: SOS SDP might be *infeasible* up to degree m for arbitrary input M.

Solutions

- Switch to the dual SDP (moment): satisfies Slater's condition, i.e, strictly feasible.
- For a generic M, by strong duality, $h_{\mathrm{ProdSym}(n)}(M) = OPT_{\mathrm{mom}}(M)$.
- For any input M, use the continuity of the dual SDP then.

Result II: Hardness w/o ASSUMPTIONs!

Theorem (Main II.1)

DPS hierarchies (or general Sum-of-Squares SDP) require $\Omega(\log(d))$ levels to solve $h_{\mathrm{Sep}(d)}$ with constant precision.

Theorem (Main II.2)

Any SDP that estimate $h_{\mathrm{Sep}(d)}(M)$ with constant errors requires size $d^{\Omega(\log(d))}$.

Remark: Theorem II.1 $\Rightarrow$ Theorem II.2 due to a recent result on psd rank (SDP) lower bound [LRS].

Proof Overview

- LB: instance w/ true value small, SoS (or SDP) value large.
- Start w/ such an instance: random 3XOR w/ true value $\sim 1/2 + \epsilon$, SoS value$= 1$ for large sos degree.
- Goal to embed such random 3XOR to an instance of $h_{\text{Sep}(d)}(M)$! How?
- Make use of a QMA(2) protocol (for 2-out-of-4 SAT) [AB+] to solve this 3XOR.
- Step 1: a random 3XOR $\Rightarrow$ a 2-out-of-4 SAT instance.
- Step 2: QMA(2) protocol as a reduction!
  - Step 2.1: Embed it further as an instance to $h_{\text{Sep}(d)}(M)$. (Theorem II.1)
  - Step 2.2: Apply LRS to the resultant problem. Then reduce it to $h_{\text{Sep}(d)}(M)$. (Theorem II.2)

Proof Overview

- LB: instance w/ true value small, SoS (or SDP) value large.
- Start w/ such an instance: random 3XOR w/ true value $\sim 1/2+\epsilon$, SoS value$=1$ for large sos degree.
- Goal to embed such random 3XOR to an instance of $h_{\mathrm{Sep}(d)}(M)$! How?
- Make use of a QMA(2) protocol (for 2-out-of-4 SAT) [AB+] to solve this 3XOR.
- Step 1: a random 3XOR $\Rightarrow$ a 2-out-of-4 SAT instance.
- Step 2: QMA(2) protocol as a reduction!
  - Step 2.1: Embed it further as an instance to $h_{\mathrm{Sep}(d)}(M)$ (Theorem II.1)
  - Step 2.2: Apply LRS to the resultant problem. Then reduce it to $h_{\mathrm{Sep}(d)}(M)$. (Theorem II.2)

Proof Overview

- LB: instance w/ true value small, SoS (or SDP) value large.
- Start w/ such an instance: random 3XOR w/ true value $\sim 1/2+\epsilon$, SoS value$=1$ for large sos degree.
- Goal to embed such random 3XOR to an instance of $h_{\mathrm{Sep}(d)}(M)$! How?
- Make use of a QMA(2) protocol (for 2-out-of-4 SAT) [AB+] to solve this 3XOR.
- Step 1: a random 3XOR $\Rightarrow$ a 2-out-of-4 SAT instance.
- Step 2: QMA(2) protocol as a reduction!

Proof Overview

- LB: instance w/ true value small, SoS (or SDP) value large.
- Start w/ such an instance: random 3XOR w/ true value $\sim 1/2 + \epsilon$, SoS value$=$ 1 for large sos degree.
- Goal to embed such random 3XOR to an instance of $h_{\mathrm{Sep}(d)}(M)$! How?
- Make use of a QMA(2) protocol (for 2-out-of-4 SAT) [AB+] to solve this 3XOR.
- Step 1: a random 3XOR $\Rightarrow$ a 2-out-of-4 SAT instance.
- Step 2: QMA(2) protocol as a reduction!

Proof Overview

- LB: instance w/ true value small, SoS (or SDP) value large.
- Start w/ such an instance: random 3XOR w/ true value $\sim 1/2 + \epsilon$, SoS value$=$ 1 for large sos degree.
- Goal to embed such random 3XOR to an instance of $h_{\mathrm{Sep}(d)}(M)$! How?
- Make use of a QMA(2) protocol (for 2-out-of-4 SAT) [AB+] to solve this 3XOR.
- Step 1: a random 3XOR $\Rightarrow$ a 2-out-of-4 SAT instance.
- Step 2: QMA(2) protocol as a reduction!
  - Step 2.1: Embed it further as an instance to $h_{\mathrm{Sep}(d)}(M)$. (Theorem II.1)
  - Step 2.2: Apply LRS to the resultant problem. Then reduce it to $h_{\mathrm{Sep}(d)}(M)$. (Theorem II.2)

Proof Overview

- LB: instance w/ true value small, SoS (or SDP) value large.
- Start w/ such an instance: random 3XOR w/ true value $\sim 1/2+\epsilon$, SoS value$=1$ for large sos degree.
- Goal to embed such random 3XOR to an instance of $h_{\mathrm{Sep}(d)}(M)$! How?
- Make use of a QMA(2) protocol (for 2-out-of-4 SAT) [AB+] to solve this 3XOR.
- Step 1: a random 3XOR $\Rightarrow$ a 2-out-of-4 SAT instance.
- Step 2: QMA(2) protocol as a reduction!
  - Step 2.1: Embed it further as an instance to $h_{\mathrm{Sep}(d)}(M)$. (Theorem II.1)
  - Step 2.2: Apply LRS to the resultant problem. Then reduce it to $h_{\mathrm{Sep}(d)}(M)$. (Theorem II.2)

Proof Overview

- LB: instance w/ true value small, SoS (or SDP) value large.
- Start w/ such an instance: random 3XOR w/ true value $\sim 1/2 + \epsilon$, SoS value$=$ 1 for large sos degree.
- Goal to embed such random 3XOR to an instance of $h_{\mathrm{Sep}(d)}(M)$! How?
- Make use of a QMA(2) protocol (for 2-out-of-4 SAT) [AB+] to solve this 3XOR.
- Step 1: a random 3XOR $\Rightarrow$ a 2-out-of-4 SAT instance.
- Step 2: QMA(2) protocol as a reduction!
  - Step 2.1: Embed it further as an instance to $h_{\mathrm{Sep}(d)}(M)$. (Theorem II.1)
  - Step 2.2: Apply LRS to the resultant problem. Then reduce it to $h_{\mathrm{Sep}(d)}(M)$. (Theorem II.2)

Proof Overview

- LB: instance w/ true value small, SoS (or SDP) value large.
- Start w/ such an instance: random 3XOR w/ true value $\sim 1/2+\epsilon$, SoS value$=$ 1 for large sos degree.
- Goal to embed such random 3XOR to an instance of $h_{\mathrm{Sep}(d)}(M)$! How?
- Make use of a QMA(2) protocol (for 2-out-of-4 SAT) [AB+] to solve this 3XOR.
- Step 1: a random 3XOR $\Rightarrow$ a 2-out-of-4 SAT instance.
- Step 2: QMA(2) protocol as a reduction!
  - Step 2.1: Embed it further as an instance to $h_{\mathrm{Sep}(d)}(M)$. (Theorem II.1)
  - Step 2.2: Apply LRS to the resultant problem. Then reduce it to $h_{\mathrm{Sep}(d)}(M)$. (Theorem II.2)

Step 1: 3XOR $\Rightarrow$ 2-out-of-4 SAT

A random 3XOR on n vars with $O(n)$ clauses: sos-deg $\Omega(n)$, true value $\sim 1/2$, pseudo-expectation value 1.

- A random 3XOR (each var appears in const clauses) has sos-deg $\Omega(n)$.
- Replace each clause $x_1 \oplus x_2 \oplus x_3 = z_c$ with $2o4(x_1, b, c, z)$, $2o4(x_2, a, c, z)$,$2o4(x_3, a, b, z)$.
- Use $2o4$ clauses to make all auxiliary z_c the same. Use expander graphs to force const appearances.
- Extending the pseudo-expectation: $\tilde{E}[y_1(x)y_2(x)] = \sum_{\alpha \in y_1 y_2} \tilde{E}[x^\alpha]$.

Step 1: 3XOR $\Rightarrow$ 2-out-of-4 SAT

A random 3XOR on n vars with $O(n)$ clauses: sos-deg $\Omega(n)$, true value $\sim 1/2$, pseudo-expectation value 1.

- A random 3XOR (each var appears in const clauses) has sos-deg $\Omega(n)$.
- Replace each clause $x_1 \oplus x_2 \oplus x_3 = z_c$ with $2o4(x_1, b, c, z)$, $2o4(x_2, a, c, z)$,$2o4(x_3, a, b, z)$.
- Use $2o4$ clauses to make all auxiliary z_c the same. Use expander graphs to force const appearances.
- Extending the pseudo-expectation: $\tilde{E}[y_1(x)y_2(x)] = \sum_{\alpha \in y_1 y_2} \tilde{E}[x^\alpha]$.

Step 1: 3XOR $\Rightarrow$ 2-out-of-4 SAT

A random 3XOR on n vars with $O(n)$ clauses: sos-deg $\Omega(n)$, true value $\sim 1/2$, pseudo-expectation value 1.

- A random 3XOR (each var appears in const clauses) has sos-deg $\Omega(n)$.
- Replace each clause $x_1 \oplus x_2 \oplus x_3 = z_c$ with $2o4(x_1, b, c, z)$, $2o4(x_2, a, c, z)$,$2o4(x_3, a, b, z)$.
- Use $2o4$ clauses to make all auxiliary z_c the same. Use expander graphs to force const appearances.
- Extending the pseudo-expectation: $\tilde{E}[y_1(x)y_2(x)] = \sum_{\alpha \in y_1 y_2} \tilde{E}[x^\alpha]$.

Step 1: 3XOR $\Rightarrow$ 2-out-of-4 SAT

A random 3XOR on n vars with $O(n)$ clauses: sos-deg $\Omega(n)$, true value $\sim 1/2$, pseudo-expectation value 1.

- A random 3XOR (each var appears in const clauses) has sos-deg $\Omega(n)$.
- Replace each clause $x_1 \oplus x_2 \oplus x_3 = z_c$ with $2o4(x_1, b, c, z)$, $2o4(x_2, a, c, z)$,$2o4(x_3, a, b, z)$.
- Use $2o4$ clauses to make all auxiliary z_c the same. Use expander graphs to force const appearances.
- Extending the pseudo-expectation: $\tilde{E}[y_1(x)y_2(x)] = \sum_{\alpha \in y_1 y_2} \tilde{E}[x^\alpha]$.

Step 2: QMA(2) protocol as a reduction

A QMA(2) protocol solves this 2-out-of-4 SAT w/ completeness 1, soundness 1/2. [AB+]

- The acceptance probability of this QMA(2) protocol as the output function.
- By soundness, the true value should be at most 1/2.
- This QMA(2) protocol has three tests. One is testing whether any $2o4$ clause is satisfied.
- The other two have "low-degree" test-measures.
- By natural extension of pseudo-expectation $\Rightarrow$ pseudo-value 1.
- Caveat: the function domain is still on $\{0,1\}^n$.

Step 2: QMA(2) protocol as a reduction

A QMA(2) protocol solves this 2-out-of-4 SAT w/ completeness 1, soundness 1/2. [AB+]

- The acceptance probability of this QMA(2) protocol as the output function.
- By soundness, the true value should be at most 1/2.
- This QMA(2) protocol has three tests. One is testing whether any $2o4$ clause is satisfied.
- The other two have "low-degree" test-measures.
- By natural extension of pseudo-expectation $\Rightarrow$ pseudo-value 1.
- Caveat: the function domain is still on $\{0, 1\}^n$.

Step 2: QMA(2) protocol as a reduction

A QMA(2) protocol solves this 2-out-of-4 SAT w/ completeness 1, soundness 1/2. [AB+]

- The acceptance probability of this QMA(2) protocol as the output function.
- By soundness, the true value should be at most 1/2.
- This QMA(2) protocol has three tests. One is testing whether any $2o4$ clause is satisfied.
- The other two have "low-degree" test-measures.
- By natural extension of pseudo-expectation $\Rightarrow$ pseudo-value 1.
- Caveat: the function domain is still on $\{0,1\}^n$.

Step 2: QMA(2) protocol as a reduction

A QMA(2) protocol solves this 2-out-of-4 SAT w/ completeness 1, soundness 1/2. [AB+]

- The acceptance probability of this QMA(2) protocol as the output function.
- By soundness, the true value should be at most 1/2.
- This QMA(2) protocol has three tests. One is testing whether any $2o4$ clause is satisfied.
- The other two have "low-degree" test-measures.
- By natural extension of pseudo-expectation $\Rightarrow$ pseudo-value 1.
- Caveat: the function domain is still on $\{0, 1\}^n$.

Step 2: QMA(2) protocol as a reduction

A QMA(2) protocol solves this 2-out-of-4 SAT w/ completeness 1, soundness 1/2. [AB+]

- The acceptance probability of this QMA(2) protocol as the output function.
- By soundness, the true value should be at most 1/2.
- This QMA(2) protocol has three tests. One is testing whether any $2o4$ clause is satisfied.
- The other two have "low-degree" test-measures.
- By natural extension of pseudo-expectation $\Rightarrow$ pseudo-value 1.
- Caveat: the function domain is still on $\{0, 1\}^n$.

Step 2: QMA(2) protocol as a reduction

A QMA(2) protocol solves this 2-out-of-4 SAT w/ completeness 1, soundness 1/2. [AB+]

- The acceptance probability of this QMA(2) protocol as the output function.
- By soundness, the true value should be at most 1/2.
- This QMA(2) protocol has three tests. One is testing whether any $2o4$ clause is satisfied.
- The other two have "low-degree" test-measures.
- By natural extension of pseudo-expectation $\Rightarrow$ pseudo-value 1.
- Caveat: the function domain is still on $\{0,1\}^n$.

Step 2: DPS and SDP lower bounds

DPS lower bound

- Embed this pseudo-distribution on $\{0,1\}^n$ to $\mathbb{R}^d$. ($d = n^{\sqrt{n}\mathrm{polylog}(n)}$)
- Thus $h_{\mathrm{Sep}(d)}(M)$ has sos degree $\Omega(\log(d))$.

SDP lower bound

- Apply LRS to this function on $\{0,1\}^n$. Obtain SDP size lower bound $(d/\log\log(d))^{\Omega(\log(d))}$.
- By soundness, a general $h_{\mathrm{Sep}(d)}(M)$ can solve this problem, thus has the same lower bound.

Step 2: DPS and SDP lower bounds

DPS lower bound

- Embed this pseudo-distribution on $\{0,1\}^n$ to $\mathbb{R}^d$. ($d = n^{\sqrt{n}\mathrm{polylog}(n)}$)
- Thus $h_{\mathrm{Sep}(d)}(M)$ has sos degree $\Omega(\log(d))$.

SDP lower bound

- Apply LRS to this function on $\{0,1\}^n$. Obtain SDP size lower bound $(d/\log\log(d))^{\Omega(\log(d))}$.
- By soundness, a general $h_{\mathrm{Sep}(d)}(M)$ can solve this problem, thus has the same lower bound.

Open Questions

DPS+

- Analyze the low levels of DPS+.
- Advantages of adding KKT conditions other than presented here.
- Extension to the non-commutative case?

SoS, SDP lower bound

- Any hope for a better bound?
- Extension to general algorithms?
- Any other applications to quantum information?

Question And Answer

Thank you!

Q & A

Proof of Theorem 1

Let $\mathcal{U} = \{f_1(x) = 0\}, \mathcal{W} = \{\forall\, i, j, g_{ij} = 0\}$. then $V(I_K) \subseteq \mathcal{U} \cap \mathcal{W}$.

It suffices to show $|\mathcal{U} \cap \mathcal{W}| < \infty$. Construct $\mathcal{A} = \mathcal{X} \cap \mathcal{U}$ s.t.

$\mathcal{A} \cap \mathcal{W} = \emptyset$ and $\dim(\mathcal{X}) = n - 1$. Note $\mathcal{W} \cap \mathcal{A} = (\mathcal{W} \cap \mathcal{U}) \cap \mathcal{X}$.

By Bézout's theorem, two varieties with dimension sum $\geq n$ must intersect. Thus

$$\dim(\mathcal{W} \cap \mathcal{U}) + \dim(\mathcal{X}) = \dim(\mathcal{W} \cap \mathcal{U}) + n - 1 < n.$$

This implies $\dim(\mathcal{W} \cap \mathcal{U}) = 0$ and thus $|V(I_K)| < \infty$.

Proof of Theorem 1

Let $\mathcal{U} = \{f_1(x) = 0\}, \mathcal{W} = \{\forall\, i, j, g_{ij} = 0\}$. then $V(I_K) \subseteq \mathcal{U} \cap \mathcal{W}$.

It suffices to show $|\mathcal{U} \cap \mathcal{W}| < \infty$. Construct $\mathcal{A} = \mathcal{X} \cap \mathcal{U}$ s.t.

$\mathcal{A} \cap \mathcal{W} = \emptyset$ and $\dim(\mathcal{X}) = n - 1$. Note $\mathcal{W} \cap \mathcal{A} = (\mathcal{W} \cap \mathcal{U}) \cap \mathcal{X}$.

By Bézout's theorem, two varieties with dimension sum $\geq n$ must intersect. Thus

$$\dim(\mathcal{W} \cap \mathcal{U}) + \dim(\mathcal{X}) = \dim(\mathcal{W} \cap \mathcal{U}) + n - 1 < n.$$

This implies $\dim(\mathcal{W} \cap \mathcal{U}) = 0$ and thus $|V(I_K)| < \infty$.

Proof of Theorem 1

Let $\mathcal{U} = \{f_1(x) = 0\}, \mathcal{W} = \{\forall\, i, j, g_{ij} = 0\}$. then $V(I_K) \subseteq \mathcal{U} \cap \mathcal{W}$.

It suffices to show $|\mathcal{U} \cap \mathcal{W}| < \infty$. Construct $\mathcal{A} = \mathcal{X} \cap \mathcal{U}$ s.t.

$\mathcal{A} \cap \mathcal{W} = \emptyset$ and $\dim(\mathcal{X}) = n - 1$. Note $\mathcal{W} \cap \mathcal{A} = (\mathcal{W} \cap \mathcal{U}) \cap \mathcal{X}$.

By Bézout's theorem, two varieties with dimension sum $\geq n$ must intersect. Thus

$$\dim(\mathcal{W} \cap \mathcal{U}) + \dim(\mathcal{X}) = \dim(\mathcal{W} \cap \mathcal{U}) + n - 1 < n.$$

This implies $\dim(\mathcal{W} \cap \mathcal{U}) = 0$ and thus $|V(I_K)| < \infty$.

Proof of Theorem 1: construct $\mathcal{X}$

Let $\mathcal{X} = \{f_0(x) = \mu\}$ for generic (μ, M). $\dim(\mathcal{X}) = n - 1$.

By Bertini's theorem, $\dim(\mathcal{A}) = \dim(\mathcal{U} \cap \mathcal{X}) = n - 2$.

The Jacobian matrix $J_{\mathcal{A}} = \begin{pmatrix} \frac{\partial f_0}{\partial x_1} & \frac{\partial f_1}{\partial x_1} \\ \vdots & \vdots \\ \frac{\partial f_0}{\partial x_n} & \frac{\partial f_1}{\partial x_n} \end{pmatrix}$ has $\mathrm{rank}(J_{\mathcal{A}}) = 2$.

$\mathcal{W}$ by definition says $\mathrm{rank}(J_{\mathcal{A}}) = 1$. Thus no intersection!

Subtly: **genericity**; **projective space**; **homogenization**!

Proof of Theorem 1: construct $\mathcal{X}$

Let $\mathcal{X} = \{f_0(x) = \mu\}$ for generic (μ, M). $\dim(\mathcal{X}) = n - 1$.

By Bertini's theorem, $\dim(\mathcal{A}) = \dim(\mathcal{U} \cap \mathcal{X}) = n - 2$.

The Jacobian matrix $J_{\mathcal{A}} = \begin{pmatrix} \frac{\partial f_0}{\partial x_1} & \frac{\partial f_1}{\partial x_1} \\ \vdots & \vdots \\ \frac{\partial f_0}{\partial x_n} & \frac{\partial f_1}{\partial x_n} \end{pmatrix}$ has $\mathrm{rank}(J_{\mathcal{A}}) = 2$.

$\mathcal{W}$ by definition says $\mathrm{rank}(J_{\mathcal{A}}) = 1$. Thus no intersection!

Subtly: **genericity**; **projective space**; **homogenization**!

Proof of Theorem 1: construct $\mathcal{X}$

Let $\mathcal{X} = \{f_0(x) = \mu\}$ for generic (μ, M). $\dim(\mathcal{X}) = n - 1$.

By Bertini's theorem, $\dim(\mathcal{A}) = \dim(\mathcal{U} \cap \mathcal{X}) = n - 2$.

The Jacobian matrix $J_{\mathcal{A}} = \begin{pmatrix} \frac{\partial f_0}{\partial x_1} & \frac{\partial f_1}{\partial x_1} \\ \vdots & \vdots \\ \frac{\partial f_0}{\partial x_n} & \frac{\partial f_1}{\partial x_n} \end{pmatrix}$ has $\operatorname{rank}(J_{\mathcal{A}}) = 2$.

$\mathcal{W}$ by definition says $\operatorname{rank}(J_{\mathcal{A}}) = 1$. Thus no intersection!

Subtly: genericity; projective space; homogenization!

Proof of Theorem 1: construct $\mathcal{X}$

Let $\mathcal{X} = \{f_0(x) = \mu\}$ for generic (μ, M). $\dim(\mathcal{X}) = n - 1$.

By Bertini's theorem, $\dim(\mathcal{A}) = \dim(\mathcal{U} \cap \mathcal{X}) = n - 2$.

The Jacobian matrix $J_{\mathcal{A}} = \begin{pmatrix} \frac{\partial f_0}{\partial x_1} & \frac{\partial f_1}{\partial x_1} \\ \vdots & \vdots \\ \frac{\partial f_0}{\partial x_n} & \frac{\partial f_1}{\partial x_n} \end{pmatrix}$ has $\mathrm{rank}(J_{\mathcal{A}}) = 2$.

$\mathcal{W}$ by definition says $\mathrm{rank}(J_{\mathcal{A}}) = 1$. Thus no intersection!

Subtly: **genericity**; **projective space**; **homogenization**!

Proof of Theorem 2

Let $\{\gamma_i\}$ be a Grobner basis for I_K.

$$|V(I_K)| < \infty \implies \max\deg\{\gamma_i\} \leq D = \exp(\mathrm{poly}(n)).$$

Now, want to bound $\deg(\sigma(x)), \deg(g(x))$ in

$$v - f_0(x) = \sigma(x) + g(x). \text{ s.t. } \sigma(x) \text{ SOS }, g(x) \in I_K^m.$$

Let $\sigma(x) = \sum s_a(x)^2$. By property of Grobner basis

$$s_a(x) = g_a(x) + u_a(x), \text{ s.t. } g_a(x) \in I_K, \deg(u_a(x)) \leq nD.$$

Thus

$$v - f_0(x) = \sigma'(x) + g'(x), \deg(\sigma'(x)) \leq \exp(\mathrm{poly}(n)), g' \in I_K.$$

Proof of Theorem 2

Let $\{\gamma_i\}$ be a Grobner basis for I_K.

$$|V(I_K)| < \infty \implies \max \deg\{\gamma_i\} \leq D = \exp(\mathrm{poly}(n)).$$

Now, want to bound $\deg(\sigma(x)), \deg(g(x))$ in

$$v - f_0(x) = \sigma(x) + g(x). \text{ s.t. } \sigma(x) \text{ SOS }, g(x) \in I_K^m.$$

Let $\sigma(x) = \sum s_a(x)^2$. By property of Grobner basis

$$s_a(x) = g_a(x) + u_a(x), \text{ s.t. } g_a(x) \in I_K, \deg(u_a(x)) \leq nD.$$

Thus

$$v - f_0(x) = \sigma'(x) + g'(x), \deg(\sigma'(x)) \leq \exp(\mathrm{poly}(n)), g' \in I_K.$$

Proof of Theorem 2

Let $\{\gamma_i\}$ be a Grobner basis for I_K.

$$|V(I_K)| < \infty \implies \max \deg\{\gamma_i\} \leq D = \exp(\text{poly}(n)).$$

Now, want to bound $\deg(\sigma(x)), \deg(g(x))$ in

$$v - f_0(x) = \sigma(x) + g(x). \text{ s.t. } \sigma(x) \text{ SOS }, g(x) \in I_K^m.$$

Let $\sigma(x) = \sum s_a(x)^2$. By property of Grobner basis

$$s_a(x) = g_a(x) + u_a(x), \text{ s.t. } g_a(x) \in I_K, \deg(u_a(x)) \leq nD.$$

Thus

$$v - f_0(x) = \sigma'(x) + g'(x), \deg(\sigma'(x)) \leq \exp(\text{poly}(n)), g' \in I_K.$$

Proof of Theorem 2

Let $\{\gamma_i\}$ be a Grobner basis for I_K.

$$|V(I_K)| < \infty \implies \max \deg\{\gamma_i\} \leq D = \exp(\mathrm{poly}(n)).$$

Now, want to bound $\deg(\sigma(x)), \deg(g(x))$ in

$$v - f_0(x) = \sigma(x) + g(x). \text{ s.t. } \sigma(x) \text{ SOS }, g(x) \in I_K^m.$$

Let $\sigma(x) = \sum s_a(x)^2$. By property of Grobner basis

$$s_a(x) = g_a(x) + u_a(x), \text{ s.t. } g_a(x) \in I_K, \deg(u_a(x)) \leq nD.$$

Thus

$$v - f_0(x) = \sigma'(x) + g'(x), \deg(\sigma'(x)) \leq \exp(\mathrm{poly}(n)), g' \in I_K.$$

Proof of Theorem 2: $g' \in I_K^m$

All we need is to show $g' \in I_K^m, m = \exp(\text{poly}(n))$.

- $\deg(g'(x)) = \deg(\sigma'(x)) = m$.
- In Grobner basis, $g'(x) = \sum t_k \gamma_k(x)$, $\deg(t_k \gamma_k(x)) \leq m$.
- (Omitted) $\gamma_k(x) = \sum u_{ij}(x) g_{ij}(x)$, $\deg(u_{ij}) \leq m$.

Thus, $g'(x) = \sum t_k u_{ij} g_{ij}(x), \deg(t_k u_{ij}) \leq m, \implies g'(x) \in I_K^m$.

$$\begin{aligned} I_K^m &= \{v(x)f_1(x) + \sum h_{ij}(x)g_{ij}(x) : \deg(v(x)f_1(x)) \leq m, \\ &\quad \forall i, j, \deg(h_{ij}g_{ij}) \leq m\}. \end{aligned}$$

Proof of Theorem 2: $g' \in I_K^m$

All we need is to show $g' \in I_K^m, m = \exp(\text{poly}(n))$.

- $\deg(g'(x)) = \deg(\sigma'(x)) = m$.
- In Grobner basis, $g'(x) = \sum t_k \gamma_k(x)$, $\deg(t_k \gamma_k(x)) \leq m$.
- (Omitted) $\gamma_k(x) = \sum u_{ij}(x) g_{ij}(x)$, $\deg(u_{ij}) \leq m$.

Thus, $g'(x) = \sum t_k u_{ij} g_{ij}(x), \deg(t_k u_{ij}) \leq m, \implies g'(x) \in I_K^m$.

$$I_K^m = \{v(x)f_1(x) + \sum h_{ij}(x)g_{ij}(x) : \deg(v(x)f_1(x)) \leq m, \forall i,j, \deg(h_{ij}g_{ij}) \leq m\}.$$

Proof of Theorem 2: $g' \in I_K^m$

All we need is to show $g' \in I_K^m, m = \exp(\mathrm{poly}(n))$.

- $\deg(g'(x)) = \deg(\sigma'(x)) = m$.
- In Grobner basis, $g'(x) = \sum t_k \gamma_k(x)$, $\deg(t_k \gamma_k(x)) \leq m$.
- (Omitted) $\gamma_k(x) = \sum u_{ij}(x) g_{ij}(x)$, $\deg(u_{ij}) \leq m$.

Thus, $g'(x) = \sum t_k u_{ij} g_{ij}(x), \deg(t_k u_{ij}) \leq m, \implies g'(x) \in I_K^m$.

$$
\begin{aligned}
I_K^m = \; & \{v(x)f_1(x) + \sum h_{ij}(x)g_{ij}(x) : \deg(v(x)f_1(x)) \leq m, \\
& \forall i, j, \deg(h_{ij}g_{ij}) \leq m\}.
\end{aligned}
$$

Proof of Theorem 2: $g' \in I_K^m$

All we need is to show $g' \in I_K^m, m = \exp(\mathrm{poly}(n))$.

- $\deg(g'(x)) = \deg(\sigma'(x)) = m$.
- In Grobner basis, $g'(x) = \sum t_k \gamma_k(x)$, $\deg(t_k \gamma_k(x)) \leq m$.
- (Omitted) $\gamma_k(x) = \sum u_{ij}(x) g_{ij}(x)$, $\deg(u_{ij}) \leq m$.

Thus, $g'(x) = \sum t_k u_{ij} g_{ij}(x), \deg(t_k u_{ij}) \leq m, \implies g'(x) \in I_K^m$.

$$\begin{aligned} I_K^m \quad = \quad & \{v(x)f_1(x) + \sum h_{ij}(x)g_{ij}(x) : \deg(v(x)f_1(x)) \leq m, \\ & \forall i,j, \deg(h_{ij}g_{ij}) \leq m\}. \end{aligned}$$

Proof of Theorem 2: $g' \in I_K^m$

All we need is to show $g' \in I_K^m, m = \exp(\mathrm{poly}(n))$.

- $\deg(g'(x)) = \deg(\sigma'(x)) = m$.
- In Grobner basis, $g'(x) = \sum t_k \gamma_k(x)$, $\deg(t_k \gamma_k(x)) \leq m$.
- (Omitted) $\gamma_k(x) = \sum u_{ij}(x) g_{ij}(x)$, $\deg(u_{ij}) \leq m$.

$$\text{Thus, } g'(x) = \sum t_k u_{ij} g_{ij}(x), \deg(t_k u_{ij}) \leq m, \implies g'(x) \in I_K^m.$$

$$\begin{aligned} I_K^m \quad = \quad & \{v(x)f_1(x) + \sum h_{ij}(x)g_{ij}(x) : \deg(v(x)f_1(x)) \leq m, \\ & \forall i, j, \deg(h_{ij}g_{ij}) \leq m\}. \end{aligned}$$