The Connes Embedding Problem, MIP* = RE, and the Completeness Theorem

Isaac Goldbring

University of California, Irvine

University of Colorado, Colorado Springs
Math Department Colloquium

Overview

- The Connes Embedding Problem (CEP) is an old and famous problem in the field of **von Neumann algebras.**
- Earlier this year, an amazing result in **complexity theory** called $\text{MIP}^* = \text{RE}$ was proven.
- Through very nontrivial detours through the fields of **C*-algebras** and **quantum information theory**, the complexity theory result yields a negative solution to CEP.
- Using some basic model theory, Bradd Hart and I showed how to go directly from $\text{MIP}^* = \text{RE}$ to the failure of CEP (while adding some other interesting results).
- In this talk, I will try to give an overview of this story.

Overview

- The Connes Embedding Problem (CEP) is an old and famous problem in the field of **von Neumann algebras**.
- Earlier this year, an amazing result in **complexity theory** called $\mathrm{MIP}^* = \mathrm{RE}$ was proven.
- Through very nontrivial detours through the fields of **C*-algebras** and **quantum information theory**, the complexity theory result yields a negative solution to CEP.
- Using some basic model theory, Bradd Hart and I showed how to go directly from $\mathrm{MIP}^* = \mathrm{RE}$ to the failure of CEP (while adding some other interesting results).
- In this talk, I will try to give an overview of this story.

Overview

- The Connes Embedding Problem (CEP) is an old and famous problem in the field of **von Neumann algebras**.
- Earlier this year, an amazing result in **complexity theory** called $\mathrm{MIP}^* = \mathrm{RE}$ was proven.
- Through very nontrivial detours through the fields of **C*-algebras** and **quantum information theory**, the complexity theory result yields a negative solution to CEP.
- Using some basic model theory, Bradd Hart and I showed how to go directly from $\mathrm{MIP}^* = \mathrm{RE}$ to the failure of CEP (while adding some other interesting results).
- In this talk, I will try to give an overview of this story.

Overview

- The Connes Embedding Problem (CEP) is an old and famous problem in the field of **von Neumann algebras**.
- Earlier this year, an amazing result in **complexity theory** called $\mathrm{MIP}^* = \mathrm{RE}$ was proven.
- Through very nontrivial detours through the fields of **C*-algebras** and **quantum information theory**, the complexity theory result yields a negative solution to CEP.
- Using some basic model theory, Bradd Hart and I showed how to go directly from $\mathrm{MIP}^* = \mathrm{RE}$ to the failure of CEP (while adding some other interesting results).
- In this talk, I will try to give an overview of this story.

Overview

- The Connes Embedding Problem (CEP) is an old and famous problem in the field of **von Neumann algebras**.
- Earlier this year, an amazing result in **complexity theory** called $\mathrm{MIP}^* = \mathrm{RE}$ was proven.
- Through very nontrivial detours through the fields of **C*-algebras** and **quantum information theory**, the complexity theory result yields a negative solution to CEP.
- Using some basic model theory, Bradd Hart and I showed how to go directly from $\mathrm{MIP}^* = \mathrm{RE}$ to the failure of CEP (while adding some other interesting results).
- In this talk, I will try to give an overview of this story.

The hyperfinite II_1 factor

- Consider the map $A \mapsto \begin{pmatrix} A & 0 \\ 0 & A \end{pmatrix}$ from $M_{2^n}(\mathbb{C})$ to $M_{2^{n+1}}(\mathbb{C})$.
- This map is a $*$-homomorphism that preserves the normalized trace on $M_{2^n}(\mathbb{C})$.
- A suitable completion of the limit of this directed system is called the **hyperfinite II_1 factor**, denoted $\mathcal{R}$.
- In general, a **von Neumann algebra** is a unital $*$-algebra of $\mathcal{B}(H)$, the set of bounded operators on a Hilbert space, closed in the strong operator topology.
- A **factor** is a von Neumann algebra with trivial center.
- A **II_1 factor** is an infinite-dimensional factor that admits a **trace**.
- $\mathcal{R}$ embeds into any II_1 factor.

The hyperfinite II_1 factor

- Consider the map $A \mapsto \begin{pmatrix} A & 0 \\ 0 & A \end{pmatrix}$ from $M_{2^n}(\mathbb{C})$ to $M_{2^{n+1}}(\mathbb{C})$.
- This map is a $*$-homomorphism that preserves the normalized trace on $M_{2^n}(\mathbb{C})$.
- A suitable completion of the limit of this directed system is called the **hyperfinite II_1 factor**, denoted $\mathcal{R}$.
- In general, a **von Neumann algebra** is a unital $*$-algebra of $\mathcal{B}(H)$, the set of bounded operators on a Hilbert space, closed in the strong operator topology.
- A **factor** is a von Neumann algebra with trivial center.
- A **II_1 factor** is an infinite-dimensional factor that admits a **trace**.
- $\mathcal{R}$ embeds into any II_1 factor.

The hyperfinite II_1 factor

- Consider the map $A \mapsto \begin{pmatrix} A & 0 \\ 0 & A \end{pmatrix}$ from $M_{2^n}(\mathbb{C})$ to $M_{2^{n+1}}(\mathbb{C})$.
- This map is a $*$-homomorphism that preserves the normalized trace on $M_{2^n}(\mathbb{C})$.
- A suitable completion of the limit of this directed system is called the **hyperfinite II_1 factor**, denoted $\mathcal{R}$.
- In general, a **von Neumann algebra** is a unital $*$-algebra of $\mathcal{B}(H)$, the set of bounded operators on a Hilbert space, closed in the strong operator topology.
- A **factor** is a von Neumann algebra with trivial center.
- A **II_1 factor** is an infinite-dimensional factor that admits a **trace**.
- $\mathcal{R}$ embeds into any II_1 factor.

The hyperfinite II_1 factor

- Consider the map $A \mapsto \begin{pmatrix} A & 0 \\ 0 & A \end{pmatrix}$ from $M_{2^n}(\mathbb{C})$ to $M_{2^{n+1}}(\mathbb{C})$.
- This map is a $*$-homomorphism that preserves the normalized trace on $M_{2^n}(\mathbb{C})$.
- A suitable completion of the limit of this directed system is called the **hyperfinite II_1 factor**, denoted $\mathcal{R}$.
- In general, a **von Neumann algebra** is a unital $*$-algebra of $\mathcal{B}(H)$, the set of bounded operators on a Hilbert space, closed in the strong operator topology.
- A **factor** is a von Neumann algebra with trivial center.
- A **II_1 factor** is an infinite-dimensional factor that admits a **trace**.
- $\mathcal{R}$ embeds into any II_1 factor.

The hyperfinite II_1 factor

- Consider the map $A \mapsto \begin{pmatrix} A & 0 \\ 0 & A \end{pmatrix}$ from $M_{2^n}(\mathbb{C})$ to $M_{2^{n+1}}(\mathbb{C})$.
- This map is a $*$-homomorphism that preserves the normalized trace on $M_{2^n}(\mathbb{C})$.
- A suitable completion of the limit of this directed system is called the **hyperfinite II_1 factor**, denoted $\mathcal{R}$.
- In general, a **von Neumann algebra** is a unital $*$-algebra of $\mathcal{B}(H)$, the set of bounded operators on a Hilbert space, closed in the strong operator topology.
- A **factor** is a von Neumann algebra with trivial center.
- A **II_1 factor** is an infinite-dimensional factor that admits a **trace**.
- $\mathcal{R}$ embeds into any II_1 factor.

The hyperfinite II_1 factor

- Consider the map $A \mapsto \begin{pmatrix} A & 0 \\ 0 & A \end{pmatrix}$ from $M_{2^n}(\mathbb{C})$ to $M_{2^{n+1}}(\mathbb{C})$.
- This map is a $*$-homomorphism that preserves the normalized trace on $M_{2^n}(\mathbb{C})$.
- A suitable completion of the limit of this directed system is called the **hyperfinite II_1 factor**, denoted $\mathcal{R}$.
- In general, a **von Neumann algebra** is a unital $*$-algebra of $\mathcal{B}(H)$, the set of bounded operators on a Hilbert space, closed in the strong operator topology.
- A **factor** is a von Neumann algebra with trivial center.
- A **II_1 factor** is an infinite-dimensional factor that admits a **trace**.
- $\mathcal{R}$ embeds into any II_1 factor.

The hyperfinite II_1 factor

- Consider the map $A \mapsto \begin{pmatrix} A & 0 \\ 0 & A \end{pmatrix}$ from $M_{2^n}(\mathbb{C})$ to $M_{2^{n+1}}(\mathbb{C})$.
- This map is a $*$-homomorphism that preserves the normalized trace on $M_{2^n}(\mathbb{C})$.
- A suitable completion of the limit of this directed system is called the **hyperfinite II_1 factor**, denoted $\mathcal{R}$.
- In general, a **von Neumann algebra** is a unital $*$-algebra of $\mathcal{B}(H)$, the set of bounded operators on a Hilbert space, closed in the strong operator topology.
- A **factor** is a von Neumann algebra with trivial center.
- A **II_1 factor** is an infinite-dimensional factor that admits a **trace**.
- $\mathcal{R}$ embeds into any II_1 factor.

The origins of the CEP

Quote (Connes, 1976)

"We now construct an approximate imbedding of N in $\mathcal{R}$. Apparently such an imbedding ought to exist for all II_1 factors because it does for the regular representation of free groups. However, the construction below relies on condition 6."

On the next page, Connes points out that an approximate imbedding of N in $\mathcal{R}$ is the same as an exact embedding of N into an **ultrapower** of $\mathcal{R}$.

The Connes Embedding Problem (CEP)

Does every II_1 factor embed into an ultrapower of $\mathcal{R}$?

The origins of the CEP

Quote (Connes, 1976)

"We now construct an approximate imbedding of N in $\mathcal{R}$. Apparently such an imbedding ought to exist for all II_1 factors because it does for the regular representation of free groups. However, the construction below relies on condition 6."

On the next page, Connes points out that an approximate imbedding of N in $\mathcal{R}$ is the same as an exact embedding of N into an **ultrapower** of $\mathcal{R}$.

The Connes Embedding Problem (CEP)

Does every II_1 factor embed into an ultrapower of $\mathcal{R}$?

The origins of the CEP

Quote (Connes, 1976)

"We now construct an approximate imbedding of N in $\mathcal{R}$. Apparently such an imbedding ought to exist for all II_1 factors because it does for the regular representation of free groups. However, the construction below relies on condition 6."

On the next page, Connes points out that an approximate imbedding of N in $\mathcal{R}$ is the same as an exact embedding of N into an **ultrapower** of $\mathcal{R}$.

The Connes Embedding Problem (CEP)

Does every II_1 factor embed into an ultrapower of $\mathcal{R}$?

Complexity theory roughly defined

Definition

A **language** is a subset L of $\{0, 1\}^{<\omega}$.

- We think of languages as encoding a collection of *problem instances* to which the answer should be "yes."

Example

There is a way of encoding finite graphs as finite strings of 0's and 1's. One could then, for example, set L to be those finite graphs (encoded as strings) that are 3-colorable.

- Complexity theory studies and compares "complexities" of languages.

Complexity theory roughly defined

Definition

A **language** is a subset L of $\{0,1\}^{<\omega}$.

- We think of languages as encoding a collection of *problem instances* to which the answer should be "yes."

Example

There is a way of encoding finite graphs as finite strings of 0's and 1's. One could then, for example, set L to be those finite graphs (encoded as strings) that are 3-colorable.

- Complexity theory studies and compares "complexities" of languages.

Complexity theory roughly defined

Definition

A **language** is a subset L of $\{0,1\}^{<\omega}$.

- We think of languages as encoding a collection of *problem instances* to which the answer should be "yes."

Example

There is a way of encoding finite graphs as finite strings of 0's and 1's. One could then, for example, set L to be those finite graphs (encoded as strings) that are 3-colorable.

- Complexity theory studies and compares "complexities" of languages.

Complexity theory roughly defined

Definition

A **language** is a subset L of $\{0, 1\}^{<\omega}$.

- We think of languages as encoding a collection of *problem instances* to which the answer should be "yes."

Example

There is a way of encoding finite graphs as finite strings of 0's and 1's. One could then, for example, set L to be those finite graphs (encoded as strings) that are 3-colorable.

- Complexity theory studies and compares "complexities" of languages.

The complexity classes P and NP

- Best-case scenario: there is an algorithm that decides, in polynomial time based on the size of the input z, whether or not $z \in L$. Such languages lie in the class P.
- For example, determining if a number is the gcd of two other numbers lies in P.
- Alternatively, instead of trying to “solve” the problem, one can just try to verify that a “purported proof” is in fact a proof.
- L lies in NP if there is an algorithm that runs in polynomial time such that:
  - If $z \in L$, there is a proof π such that the algorithm accepts (z, π).
  - If $z \notin L$, then there is no π for which (z, π) is accepted.
- For example, graph isomorphism is in NP.
- Very famous open question: $\mathrm{P} = \mathrm{NP}$?

The complexity classes P and NP

- Best-case scenario: there is an algorithm that decides, in polynomial time based on the size of the input z, whether or not $z \in L$. Such languages lie in the class P.
- For example, determining if a number is the gcd of two other numbers lies in P.
- Alternatively, instead of trying to "solve" the problem, one can just try to verify that a "purported proof" is in fact a proof.
- L lies in NP if there is an algorithm that runs in polynomial time such that:
  - If $z \in L$, there is a proof π such that the algorithm accepts (z, π).
  - If $z \notin L$, then there is no π for which (z, π) is accepted.
- For example, graph isomorphism is in NP.
- Very famous open question: $\mathrm{P} = \mathrm{NP}$?

The complexity classes P and NP

- Best-case scenario: there is an algorithm that decides, in polynomial time based on the size of the input z, whether or not $z \in L$. Such languages lie in the class P.
- For example, determining if a number is the gcd of two other numbers lies in P.
- Alternatively, instead of trying to “solve” the problem, one can just try to verify that a “purported proof” is in fact a proof.
- L lies in NP if there is an algorithm that runs in polynomial time such that:
  - If $z \in L$, there is a proof π such that the algorithm accepts (z, π).
  - If $z \notin L$, then there is no π for which (z, π) is accepted.
- For example, graph isomorphism is in NP.
- Very famous open question: $\mathrm{P} = \mathrm{NP}$?

The complexity classes P and NP

- Best-case scenario: there is an algorithm that decides, in polynomial time based on the size of the input z, whether or not $z \in L$. Such languages lie in the class P.
- For example, determining if a number is the gcd of two other numbers lies in P.
- Alternatively, instead of trying to "solve" the problem, one can just try to verify that a "purported proof" is in fact a proof.
- L lies in NP if there is an algorithm that runs in polynomial time such that:
  - If $z \in L$, there is a proof π such that the algorithm accepts (z, π).
  - If $z \notin L$, then there is no π for which (z, π) is accepted.
- For example, graph isomorphism is in NP.
- Very famous open question: $P = NP$?

The complexity classes P and NP

- Best-case scenario: there is an algorithm that decides, in polynomial time based on the size of the input z, whether or not $z \in L$. Such languages lie in the class P.
- For example, determining if a number is the gcd of two other numbers lies in P.
- Alternatively, instead of trying to "solve" the problem, one can just try to verify that a "purported proof" is in fact a proof.
- L lies in NP if there is an algorithm that runs in polynomial time such that:
  - If $z \in L$, there is a proof π such that the algorithm accepts (z, π).
  - If $z \notin L$, then there is no π for which (z, π) is accepted.
- For example, graph isomorphism is in NP.
- Very famous open question: P $=$ NP?

The complexity classes P and NP

- Best-case scenario: there is an algorithm that decides, in polynomial time based on the size of the input z, whether or not $z \in L$. Such languages lie in the class P.
- For example, determining if a number is the gcd of two other numbers lies in P.
- Alternatively, instead of trying to "solve" the problem, one can just try to verify that a "purported proof" is in fact a proof.
- L lies in NP if there is an algorithm that runs in polynomial time such that:
  - If $z \in L$, there is a proof π such that the algorithm accepts (z, π).
  - If $z \notin L$, then there is no π for which (z, π) is accepted.
- For example, graph isomorphism is in NP.
- Very famous open question: $\mathrm{P} = \mathrm{NP}$?

The complexity class IP

- What about graph *non*-isomorphism? Too many possible isomorphisms to just check in polynomial time.
- The complexity class IP is the class of languages for which there is a *randomized, interactive* verification procedure for L.
- There is a "verifier" and a "prover." The verifier randomly chooses a question to ask the prover, the prover then responds (no limitations on this computation), and based on the answer the verifier chooses to accept or reject (in polynomial time).
- If $z \in L$, then there is a strategy for the prover for which the verifier accepts with high probability, e.g. $\geq \frac{2}{3}$.
- If $z \notin L$, then no strategy results with acceptance with probability $\geq \frac{1}{3}$.
- Graph non-isomorphism is in IP.
- Theorem (Shamir): IP $=$ PSPACE.

The complexity class IP

- What about graph *non*-isomorphism? Too many possible isomorphisms to just check in polynomial time.
- The complexity class IP is the class of languages for which there is a *randomized, interactive* verification procedure for L.
- There is a "verifier" and a "prover." The verifier randomly chooses a question to ask the prover, the prover then responds (no limitations on this computation), and based on the answer the verifier chooses to accept or reject (in polynomial time).
- If $z \in L$, then there is a strategy for the prover for which the verifier accepts with high probability, e.g. $\geq \frac{2}{3}$.
- If $z \notin L$, then no strategy results with acceptance with probability $\geq \frac{1}{3}$.
- Graph non-isomorphism is in IP.
- Theorem (Shamir): IP $=$ PSPACE.

The complexity class IP

- What about graph *non*-isomorphism? Too many possible isomorphisms to just check in polynomial time.
- The complexity class IP is the class of languages for which there is a *randomized, interactive* verification procedure for L.
- There is a "verifier" and a "prover." The verifier randomly chooses a question to ask the prover, the prover then responds (no limitations on this computation), and based on the answer the verifier chooses to accept or reject (in polynomial time).
- If $z \in L$, then there is a strategy for the prover for which the verifier accepts with high probability, e.g. $\geq \frac{2}{3}$.
- If $z \notin L$, then no strategy results with acceptance with probability $\geq \frac{1}{3}$.
- Graph non-isomorphism is in IP.
- Theorem (Shamir): IP $=$ PSPACE.

The complexity class IP

- What about graph *non*-isomorphism? Too many possible isomorphisms to just check in polynomial time.
- The complexity class IP is the class of languages for which there is a *randomized, interactive* verification procedure for L.
- There is a "verifier" and a "prover." The verifier randomly chooses a question to ask the prover, the prover then responds (no limitations on this computation), and based on the answer the verifier chooses to accept or reject (in polynomial time).
- If $z \in L$, then there is a strategy for the prover for which the verifier accepts with high probability, e.g. $\geq \frac{2}{3}$.
- If $z \notin L$, then no strategy results with acceptance with probability $\geq \frac{1}{3}$.
- Graph non-isomorphism is in IP.
- Theorem (Shamir): IP $=$ PSPACE.

The complexity class IP

- What about graph *non*-isomorphism? Too many possible isomorphisms to just check in polynomial time.
- The complexity class IP is the class of languages for which there is a *randomized, interactive* verification procedure for L.
- There is a "verifier" and a "prover." The verifier randomly chooses a question to ask the prover, the prover then responds (no limitations on this computation), and based on the answer the verifier chooses to accept or reject (in polynomial time).
- If $z \in L$, then there is a strategy for the prover for which the verifier accepts with high probability, e.g. $\geq \frac{2}{3}$.
- If $z \notin L$, then no strategy results with acceptance with probability $\geq \frac{1}{3}$.
- Graph non-isomorphism is in IP.
- Theorem (Shamir): IP $=$ PSPACE.

The complexity class IP

- What about graph *non*-isomorphism? Too many possible isomorphisms to just check in polynomial time.
- The complexity class IP is the class of languages for which there is a *randomized, interactive* verification procedure for L.
- There is a “verifier” and a “prover.” The verifier randomly chooses a question to ask the prover, the prover then responds (no limitations on this computation), and based on the answer the verifier chooses to accept or reject (in polynomial time).
- If $z \in L$, then there is a strategy for the prover for which the verifier accepts with high probability, e.g. $\geq \frac{2}{3}$.
- If $z \notin L$, then no strategy results with acceptance with probability $\geq \frac{1}{3}$.
- Graph non-isomorphism is in IP.
- Theorem (Shamir): IP $=$ PSPACE.

The complexity class IP

- What about graph *non*-isomorphism? Too many possible isomorphisms to just check in polynomial time.
- The complexity class IP is the class of languages for which there is a *randomized, interactive* verification procedure for L.
- There is a "verifier" and a "prover." The verifier randomly chooses a question to ask the prover, the prover then responds (no limitations on this computation), and based on the answer the verifier chooses to accept or reject (in polynomial time).
- If $z \in L$, then there is a strategy for the prover for which the verifier accepts with high probability, e.g. $\geq \frac{2}{3}$.
- If $z \notin L$, then no strategy results with acceptance with probability $\geq \frac{1}{3}$.
- Graph non-isomorphism is in IP.
- Theorem (Shamir): $\mathrm{IP} = \mathrm{PSPACE}$.

The complexity class MIP

- One can increase computational power if one allows *multiple provers*, for then one can run “police-style interrogation techniques” to see if the provers are telling the truth, allowing one to examine “exponentially long proofs” in polynomial time.
- MIP is the class of languages for which there is a *multiprover, interactive proof* that accepts with high probability those strings that are in L and rejects with high probability those strings that are not in L.
- Theorem (Babai, Fortnow, Lund): MIP $=$ NEXP, the version of NP that allows the program to run for exponential time.

The complexity class MIP

- One can increase computational power if one allows *multiple provers*, for then one can run “police-style interrogation techniques” to see if the provers are telling the truth, allowing one to examine “exponentially long proofs” in polynomial time.
- MIP is the class of languages for which there is a *multiprover, interactive proof* that accepts with high probability those strings that are in L and rejects with high probability those strings that are not in L.
- Theorem (Babai, Fortnow, Lund): MIP $=$ NEXP, the version of NP that allows the program to run for exponential time.

The complexity class MIP

- One can increase computational power if one allows *multiple provers*, for then one can run “police-style interrogation techniques” to see if the provers are telling the truth, allowing one to examine “exponentially long proofs” in polynomial time.
- MIP is the class of languages for which there is a *multiprover, interactive proof* that accepts with high probability those strings that are in L and rejects with high probability those strings that are not in L.
- Theorem (Babai, Fortnow, Lund): $\mathrm{MIP} = \mathrm{NEXP}$, the version of NP that allows the program to run for exponential time.

Nonlocal games

Definition

A **nonlocal game with n questions and k answers** consists of:

- A probability distribution μ on $[n]^2$, and
- A *decision predicate* $D : [n]^2 \times [k]^2 \to \{0, 1\}$.

So Alice and Bob get asked questions x and y respectively from $[n]$ (randomly according to μ), they somehow return answers a and b from $[k]$, and then D decides if they "win" or not. How should they decide how to answer?

Definition

A **classical correlation** (for n and k) is a tuple $p(a, b|x, y)$ such that there is a probability space (Λ, ν) and functions $A^\lambda, B^\lambda : [n] \to [k]$ such that $p(a, b|x, y) = \nu(\{\lambda \in \Lambda \ : \ A^\lambda(x) = a \text{ and } B^\lambda(y) = b\})$. $C_c(n, k)$ denotes the set of classical correlations.

Nonlocal games

Definition

A **nonlocal game with n questions and k answers** consists of:

- A probability distribution μ on $[n]^2$, and
- A *decision predicate* $D : [n]^2 \times [k]^2 \to \{0, 1\}$.

So Alice and Bob get asked questions x and y respectively from $[n]$ (randomly according to μ), they somehow return answers a and b from $[k]$, and then D decides if they "win" or not. How should they decide how to answer?

Definition

A **classical correlation** (for n and k) is a tuple $p(a, b|x, y)$ such that there is a probability space (Λ, ν) and functions $A^\lambda, B^\lambda : [n] \to [k]$ such that $p(a, b|x, y) = \nu(\{\lambda \in \Lambda \ : \ A^\lambda(x) = a \text{ and } B^\lambda(y) = b\})$. $C_c(n, k)$ denotes the set of classical correlations.

Nonlocal games

Definition

A **nonlocal game with n questions and k answers** consists of:

- A probability distribution μ on $[n]^2$, and
- A *decision predicate* $D : [n]^2 \times [k]^2 \to \{0, 1\}$.

So Alice and Bob get asked questions x and y respectively from $[n]$ (randomly according to μ), they somehow return answers a and b from $[k]$, and then D decides if they "win" or not. How should they decide how to answer?

Definition

A **classical correlation** (for n and k) is a tuple $p(a, b|x, y)$ such that there is a probability space (Λ, ν) and functions $A^\lambda, B^\lambda : [n] \to [k]$ such that $p(a, b|x, y) = \nu(\{\lambda \in \Lambda \ : \ A^\lambda(x) = a \text{ and } B^\lambda(y) = b\})$.
$C_c(n, k)$ denotes the set of classical correlations.

MIP reformulated

Definition

If $\mathfrak{G}$ is a nonlocal game as above, and $p \in C_c(n,k)$, then the players' expected value of winning if they play according to p is

$$\mathrm{val}(\mathfrak{G}, p) := \sum_{x,y} \mu(x,y) \sum_{a,b} D(a,b,x,y) p(a,b|x,y).$$

The **classical value of** $\mathfrak{G}$ is $\mathrm{val}(\mathfrak{G}) := \sup_{p \in C_c(n,k)} \mathrm{val}(\mathfrak{G}, p)$.

Proposition

L belongs to MIP if and only if there is an "efficient" mapping $z \mapsto \mathfrak{G}_z$ from sequence of bits to nonlocal games such that:

- $z \in L \Rightarrow \mathrm{val}(\mathfrak{G}_z) \geq \frac{2}{3}$
- $z \notin L \Rightarrow \mathrm{val}(\mathfrak{G}_z) \leq \frac{1}{3}$.

MIP reformulated

Definition

If $\mathfrak{G}$ is a nonlocal game as above, and $p \in C_c(n,k)$, then the players' expected value of winning if they play according to p is

$$\mathrm{val}(\mathfrak{G}, p) := \sum_{x,y} \mu(x,y) \sum_{a,b} D(a,b,x,y)p(a,b|x,y).$$

The **classical value of** $\mathfrak{G}$ is $\mathrm{val}(\mathfrak{G}) := \sup_{p \in C_c(n,k)} \mathrm{val}(\mathfrak{G}, p)$.

Proposition

L belongs to MIP if and only if there is an "efficient" mapping $z \mapsto \mathfrak{G}_z$ from sequence of bits to nonlocal games such that:

- $z \in L \Rightarrow \mathrm{val}(\mathfrak{G}_z) \geq \frac{2}{3}$
- $z \notin L \Rightarrow \mathrm{val}(\mathfrak{G}_z) \leq \frac{1}{3}$.

Quantum correlations

We now consider *quantum strategies*:

Definition

$C_{qs}(n,k)$ denotes those correlations $p(a,b|x,y)$ for which there are:

- finite-dimensional Hilbert spaces H_A and H_B,
- for each $x \in [n]$, positive operators $A_1^x, \ldots, A_k^x$ on H_A so that $\sum_{a=1}^k A_a^x = I_{H_A}$ (quantum measurement)
- for each $y \in [n]$, positive operators $B_1^y, \ldots, B_k^y$ on H_B so that $\sum_{b=1}^n B_b^y = I_{H_B}$, and
- a unit vector $\xi \in H_A \otimes H_B$ (state of the composite system)

so that $p(a,b|x,y) = \langle (A_a^x \otimes B_b^y)\xi, \xi\rangle$.

Tsirelson's Weaker Problem: Is $C_{qs}(n,k)$ a closed set?
Answer: No! (Slofstra, 2019)

Quantum correlations

We now consider *quantum strategies*:

Definition

$C_{qs}(n,k)$ denotes those correlations $p(a,b|x,y)$ for which there are:

- finite-dimensional Hilbert spaces H_A and H_B,
- for each $x \in [n]$, positive operators $A_1^x, \ldots, A_k^x$ on H_A so that $\sum_{a=1}^k A_a^x = I_{H_A}$ (quantum measurement)
- for each $y \in [n]$, positive operators $B_1^y, \ldots, B_k^y$ on H_B so that $\sum_{b=1}^n B_b^y = I_{H_B}$, and
- a unit vector $\xi \in H_A \otimes H_B$ (state of the composite system)

so that $p(a,b|x,y) = \langle (A_a^x \otimes B_b^y)\xi, \xi\rangle$.

Tsirelson's Weaker Problem: Is $C_{qs}(n,k)$ a closed set?
Answer: No! (Slofstra, 2019)

The complexity class MIP^*

Definition

Given a nonlocal game $\mathfrak{G}$, its **quantum entangled value** is

$$\mathrm{val}^*(\mathfrak{G}) := \sup_{p \in C_{qs}(n,k)} \sum_{x,y} \mu(x,y) \sum_{a,b} D(a,b,x,y)p(a,b|x,y).$$

Definition

L belongs to MIP^* if and only if there is an "efficient" mapping $z \mapsto \mathfrak{G}_z$ from bits to nonlocal games such that:

- $z \in L \Rightarrow \mathrm{val}^*(\mathfrak{G}_z) \geq \frac{2}{3}$
- $z \notin L \Rightarrow \mathrm{val}^*(\mathfrak{G}_z) \leq \frac{1}{3}$.

The complexity class MIP*

Definition

Given a nonlocal game $\mathfrak{G}$, its **quantum entangled value** is

$$\mathrm{val}^*(\mathfrak{G}) := \sup_{p \in C_{qs}(n,k)} \sum_{x,y} \mu(x,y) \sum_{a,b} D(a,b,x,y) p(a,b|x,y).$$

Definition

L belongs to MIP* if and only if there is an "efficient" mapping $z \mapsto \mathfrak{G}_z$ from bits to nonlocal games such that:

- $z \in L \Rightarrow \mathrm{val}^*(\mathfrak{G}_z) \geq \frac{2}{3}$
- $z \notin L \Rightarrow \mathrm{val}^*(\mathfrak{G}_z) \leq \frac{1}{3}$.

Some facts about MIP^*

Theorem (Ito and Vidick (2012))

$\mathrm{MIP} \subseteq \mathrm{MIP}^*$.

Not obvious; maybe entanglement allows the provers to cheat.

Theorem (Natarajan and Wright (2019))

$\mathrm{NEEXP} \subseteq \mathrm{MIP}^*$. *Consequently,* $\mathrm{MIP} \neq \mathrm{MIP}^*$.

Definition

RE denotes the **recursively enumerable** languages: L belongs to RE if there is some algorithm (no time/space considerations) such that, if $z \in L$, then the algorithm lets us know.

Fairly easy to see that $\mathrm{MIP}^* \subseteq \mathrm{RE}$ using brute force search.

Some facts about MIP*

Theorem (Ito and Vidick (2012))

MIP $\subseteq$ MIP*.

Not obvious; maybe entanglement allows the provers to cheat.

Theorem (Natarajan and Wright (2019))

NEEXP $\subseteq$ MIP*. *Consequently,* MIP $\neq$ MIP**.*

Definition

RE denotes the **recursively enumerable** languages: L belongs to RE if there is some algorithm (no time/space considerations) such that, if $z \in L$, then the algorithm lets us know.

Fairly easy to see that MIP* $\subseteq$ RE using brute force search.

Some facts about MIP^*

Theorem (Ito and Vidick (2012))

$\text{MIP} \subseteq \text{MIP}^*$.

Not obvious; maybe entanglement allows the provers to cheat.

Theorem (Natarajan and Wright (2019))

$\text{NEEXP} \subseteq \text{MIP}^*$. *Consequently,* $\text{MIP} \neq \text{MIP}^*$.

Definition

RE denotes the **recursively enumerable** languages: L belongs to RE if there is some algorithm (no time/space considerations) such that, if $z \in L$, then the algorithm lets us know.

Fairly easy to see that $\text{MIP}^* \subseteq \text{RE}$ using brute force search.

Some facts about MIP*

Theorem (Ito and Vidick (2012))

$\mathrm{MIP} \subseteq \mathrm{MIP}^*$.

Not obvious; maybe entanglement allows the provers to cheat.

Theorem (Natarajan and Wright (2019))

$\mathrm{NEEXP} \subseteq \mathrm{MIP}^*$. *Consequently,* $\mathrm{MIP} \neq \mathrm{MIP}^*$.

Definition

RE denotes the **recursively enumerable** languages: L belongs to RE if there is some algorithm (no time/space considerations) such that, if $z \in L$, then the algorithm lets us know.

Fairly easy to see that $\mathrm{MIP}^* \subseteq \mathrm{RE}$ using brute force search.

$\mathrm{MIP}^* = \mathrm{RE}$

Theorem (Ji, Natarajan, Vidick, Wright, and Yuen (2020))

$\mathrm{MIP}^* = \mathrm{RE}$. *More precisely, there is an efficient mapping* $\mathcal{M} \mapsto \mathfrak{G}_{\mathcal{M}}$ *from Turing machines to nonlocal games such that:*

- *If* $\mathcal{M}$ *halts, then* $\mathrm{val}^*(\mathfrak{G}_{\mathcal{M}}) = 1$.
- *If* $\mathcal{M}$ *does not halt, then* $\mathrm{val}^*(\mathfrak{G}_{\mathcal{M}}) \leq \frac{1}{2}$.

Quantum computers can actually reliably verify unsolvable problems!

The spookiness of entanglement!

$\mathrm{MIP}^* = \mathrm{RE}$

Theorem (Ji, Natarajan, Vidick, Wright, and Yuen (2020))

$\mathrm{MIP}^* = \mathrm{RE}$. *More precisely, there is an efficient mapping $\mathcal{M} \mapsto \mathfrak{G}_{\mathcal{M}}$ from Turing machines to nonlocal games such that:*

- *If $\mathcal{M}$ halts, then $\mathrm{val}^*(\mathfrak{G}_{\mathcal{M}}) = 1$.*
- *If $\mathcal{M}$ does not halt, then $\mathrm{val}^*(\mathfrak{G}_{\mathcal{M}}) \leq \frac{1}{2}$.*

Quantum computers can actually reliably verify unsolvable problems!

The spookiness of entanglement!

$\mathrm{MIP}^* = \mathrm{RE}$

Theorem (Ji, Natarajan, Vidick, Wright, and Yuen (2020))

$\mathrm{MIP}^* = \mathrm{RE}$. *More precisely, there is an efficient mapping* $\mathcal{M} \mapsto \mathfrak{G}_{\mathcal{M}}$ *from Turing machines to nonlocal games such that:*

- *If* $\mathcal{M}$ *halts, then* $\mathrm{val}^*(\mathfrak{G}_{\mathcal{M}}) = 1$.
- *If* $\mathcal{M}$ *does not halt, then* $\mathrm{val}^*(\mathfrak{G}_{\mathcal{M}}) \leq \frac{1}{2}$.

Quantum computers can actually reliably verify unsolvable problems!
The spookiness of entanglement!

Quantum commuting correlations

The tensor product model is good for non-relativisitic quantum mechanics (slow movement, low energy), but not so good for more "extreme" scenarios, where one uses *quantum field theory*, where it is not clear how to assign Alice and Bob their own systems.

Definition

$C_{qc}(n,k)$ denotes those $p(a,b|x,y)$ for which there are:

- a single separable (possibly infinite-dimensional) Hilbert space H
- for each $x \in [n]$, positive operators $A^x_1, \ldots, A^x_k$ on H so that $\sum_{a=1}^n A^x_a = I_H$, and likewise (B^y_b)...
- a unit vector (state) $\xi \in H$

so that A^x_a commutes with B^y_b for each x, y, a, b (simultaneous measurability condition) and such that $p(a,b|x,y) = \langle A^x_a B^y_b \xi, \xi \rangle$.

Note $C_{qs}(n,k) \subseteq C_{qc}(n,k)$ and that $C_{qc}(n,k)$ is closed.

Quantum commuting correlations

The tensor product model is good for non-relativisitic quantum mechanics (slow movement, low energy), but not so good for more "extreme" scenarios, where one uses *quantum field theory*, where it is not clear how to assign Alice and Bob their own systems.

Definition

$C_{qc}(n,k)$ denotes those $p(a,b|x,y)$ for which there are:

- a single separable (possibly infinite-dimensional) Hilbert space H
- for each $x \in [n]$, positive operators $A_1^x, \ldots, A_k^x$ on H so that $\sum_{a=1}^n A_a^x = I_H$, and likewise (B_b^y)...
- a unit vector (state) $\xi \in H$

so that A_a^x commutes with B_b^y for each x, y, a, b (simultaneous measurability condition) and such that $p(a,b|x,y) = \langle A_a^x B_b^y \xi, \xi \rangle$.

Note $C_{qs}(n,k) \subseteq C_{qc}(n,k)$ and that $C_{qc}(n,k)$ is closed.

Quantum commuting correlations

The tensor product model is good for non-relativisitic quantum mechanics (slow movement, low energy), but not so good for more "extreme" scenarios, where one uses *quantum field theory*, where it is not clear how to assign Alice and Bob their own systems.

Definition

$C_{qc}(n,k)$ denotes those $p(a,b|x,y)$ for which there are:

- a single separable (possibly infinite-dimensional) Hilbert space H
- for each $x \in [n]$, positive operators $A_1^x, \ldots, A_k^x$ on H so that $\sum_{a=1}^n A_a^x = I_H$, and likewise (B_b^y)...
- a unit vector (state) $\xi \in H$

so that A_a^x commutes with B_b^y for each x, y, a, b (simultaneous measurability condition) and such that $p(a,b|x,y) = \langle A_a^x B_b^y \xi, \xi \rangle$.

Note $C_{qs}(n,k) \subseteq C_{qc}(n,k)$ and that $C_{qc}(n,k)$ is closed.

A negative solution to Tsirelson's Problem

Tsirelson's Problem (1993)

Is $C_{qc}(n,k) = \overline{C_{qs}(n,k)}$ for all n, k?

- A brute-force search yields effective lower bound approximations to $\mathrm{val}^*(\mathfrak{G})$.
- A *semidefinite programming/noncommutative Positivstellenzats* argument shows that one can give an effective upper bound approximation to $\mathrm{val}^{co}(\mathfrak{G}) := \sup_{p \in C_{qc}} \mathrm{val}(\mathfrak{G}, p)$. (Model theory gives a simpler argument for this fact.)
- If Tsirelson's problem had a positive answer, then $\mathrm{val}^*(\mathfrak{G}) = \mathrm{val}^{co}(\mathfrak{G})$ and we could effectively approximate the (common) quantum value of the game.
- Consequently, every language in MIP* would be decidable, a contradiction.

A negative solution to Tsirelson's Problem

Tsirelson's Problem (1993)

Is $C_{qc}(n,k) = \overline{C_{qs}(n,k)}$ for all n,k?

- A brute-force search yields effective lower bound approximations to $\mathrm{val}^*(\mathfrak{G})$.
- A *semidefinite programming/noncommutative Positivstellenzats* argument shows that one can give an effective upper bound approximation to $\mathrm{val}^{co}(\mathfrak{G}) := \sup_{p \in C_{qc}} \mathrm{val}(\mathfrak{G}, p)$. (Model theory gives a simpler argument for this fact.)
- If Tsirelson's problem had a positive answer, then $\mathrm{val}^*(\mathfrak{G}) = \mathrm{val}^{co}(\mathfrak{G})$ and we could effectively approximate the (common) quantum value of the game.
- Consequently, every language in MIP* would be decidable, a contradiction.

A negative solution to Tsirelson's Problem

Tsirelson's Problem (1993)

Is $C_{qc}(n,k) = \overline{C_{qs}(n,k)}$ for all n, k?

- A brute-force search yields effective lower bound approximations to $\mathrm{val}^*(\mathfrak{G})$.
- A *semidefinite programming/noncommutative Positivstellenzats* argument shows that one can give an effective upper bound approximation to $\mathrm{val}^{co}(\mathfrak{G}) := \sup_{p \in C_{qc}} \mathrm{val}(\mathfrak{G}, p)$. (Model theory gives a simpler argument for this fact.)
- If Tsirelson's problem had a positive answer, then $\mathrm{val}^*(\mathfrak{G}) = \mathrm{val}^{co}(\mathfrak{G})$ and we could effectively approximate the (common) quantum value of the game.
- Consequently, every language in MIP* would be decidable, a contradiction.

A negative solution to Tsirelson's Problem

Tsirelson's Problem (1993)

Is $C_{qc}(n,k) = \overline{C_{qs}(n,k)}$ for all n, k?

- A brute-force search yields effective lower bound approximations to $\mathsf{val}^*(\mathfrak{G})$.
- A *semidefinite programming/noncommutative Positivstellenzats* argument shows that one can give an effective upper bound approximation to $\mathsf{val}^{co}(\mathfrak{G}) := \sup_{p \in C_{qc}} \mathsf{val}(\mathfrak{G}, p)$. (Model theory gives a simpler argument for this fact.)
- If Tsirelson's problem had a positive answer, then $\mathsf{val}^*(\mathfrak{G}) = \mathsf{val}^{co}(\mathfrak{G})$ and we could effectively approximate the (common) quantum value of the game.
- Consequently, every language in MIP* would be decidable, a contradiction.

A negative solution to Tsirelson's Problem

Tsirelson's Problem (1993)

Is $C_{qc}(n,k) = \overline{C_{qs}(n,k)}$ for all n, k?

- A brute-force search yields effective lower bound approximations to $\mathrm{val}^*(\mathfrak{G})$.
- A *semidefinite programming/noncommutative Positivstellenzats* argument shows that one can give an effective upper bound approximation to $\mathrm{val}^{co}(\mathfrak{G}) := \sup_{p \in C_{qc}} \mathrm{val}(\mathfrak{G}, p)$. (Model theory gives a simpler argument for this fact.)
- If Tsirelson's problem had a positive answer, then $\mathrm{val}^*(\mathfrak{G}) = \mathrm{val}^{co}(\mathfrak{G})$ and we could effectively approximate the (common) quantum value of the game.
- Consequently, every language in MIP* would be decidable, a contradiction.

CEP and Kirchberg's QWEP problem

Kirchberg's QWEP Problem

$C^*(\mathbb{F}_\infty) \odot C^*(\mathbb{F}_\infty)$ possesses a unique norm whose completion is a C*-algebra.

Theorem

1. *(Kirchberg (1993)) CEP is equivalent to the QWEP problem.*
2. *(Fritz/Junge et.al (2010); Ozawa (2013)) Tsirelson's problem is equivalent to the QWEP problem.*

Corollary

CEP fails!

Huh?

CEP and Kirchberg's QWEP problem

Kirchberg's QWEP Problem

$C^*(\mathbb{F}_\infty) \odot C^*(\mathbb{F}_\infty)$ possesses a unique norm whose completion is a C*-algebra.

Theorem

1. *(Kirchberg (1993)) CEP is equivalent to the QWEP problem.*
2. *(Fritz/Junge et.al (2010); Ozawa (2013)) Tsirelson's problem is equivalent to the QWEP problem.*

Corollary

CEP fails!

Huh?

CEP and Kirchberg's QWEP problem

Kirchberg's QWEP Problem

$C^*(\mathbb{F}_\infty) \odot C^*(\mathbb{F}_\infty)$ possesses a unique norm whose completion is a C*-algebra.

Theorem

1. *(Kirchberg (1993)) CEP is equivalent to the QWEP problem.*
2. *(Fritz/Junge et.al (2010); Ozawa (2013)) Tsirelson's problem is equivalent to the QWEP problem.*

Corollary

CEP fails!

Huh?

CEP and Kirchberg's QWEP problem

Kirchberg's QWEP Problem

$C^*(\mathbb{F}_\infty) \odot C^*(\mathbb{F}_\infty)$ possesses a unique norm whose completion is a C*-algebra.

Theorem

1. *(Kirchberg (1993)) CEP is equivalent to the QWEP problem.*
2. *(Fritz/Junge et.al (2010); Ozawa (2013)) Tsirelson's problem is equivalent to the QWEP problem.*

Corollary

CEP fails!

Huh?

CEP and computability

Theorem (G. and Hart (2016))

If CEP holds, then the universal theory of $\mathcal{R}$ is computable.

- The conclusion means that for any *formal expression* $\sigma = \sup_{\|x\| \leq 1} \varphi(x)$ in the (model-theoretic) language of tracial von Neumann algebras, where φ is a continuous combination of traces of $*$-polynomials, we can effectively approximate its value $\sigma^{\mathcal{R}}$ in $\mathcal{R}$ up to any (rational) error.
- Lower bounds: brute force.
- Upper bounds: if $\sigma^{\mathcal{R}} \leq r$, then CEP tells us that $\sigma \leq r$ is a *logical consequence* of the theory of II_1 factors.
- By running formal proofs from the axioms of II_1 factors, the Completeness Theorem tells us we will eventually see that $\sigma \leq r$ is a *theorem*. (Soundness tells us no mistakes are made.)

CEP and computability

Theorem (G. and Hart (2016))

If CEP holds, then the universal theory of $\mathcal{R}$ is computable.

- The conclusion means that for any *formal expression* $\sigma = \sup_{\|x\| \leq 1} \varphi(x)$ in the (model-theoretic) language of tracial von Neumann algebras, where φ is a continuous combination of traces of $*$-polynomials, we can effectively approximate its value $\sigma^{\mathcal{R}}$ in $\mathcal{R}$ up to any (rational) error.
- Lower bounds: brute force.
- Upper bounds: if $\sigma^{\mathcal{R}} \leq r$, then CEP tells us that $\sigma \leq r$ is a *logical consequence* of the theory of II_1 factors.
- By running formal proofs from the axioms of II_1 factors, the Completeness Theorem tells us we will eventually see that $\sigma \leq r$ is a *theorem*. (Soundness tells us no mistakes are made.)

CEP and computability

Theorem (G. and Hart (2016))

If CEP holds, then the universal theory of $\mathcal{R}$ is computable.

- The conclusion means that for any *formal expression* $\sigma = \sup_{\|x\| \leq 1} \varphi(x)$ in the (model-theoretic) language of tracial von Neumann algebras, where φ is a continuous combination of traces of $*$-polynomials, we can effectively approximate its value $\sigma^{\mathcal{R}}$ in $\mathcal{R}$ up to any (rational) error.
- Lower bounds: brute force.
- Upper bounds: if $\sigma^{\mathcal{R}} \leq r$, then CEP tells us that $\sigma \leq r$ is a *logical consequence* of the theory of II_1 factors.
- By running formal proofs from the axioms of II_1 factors, the Completeness Theorem tells us we will eventually see that $\sigma \leq r$ is a *theorem*. (Soundness tells us no mistakes are made.)

CEP and computability

Theorem (G. and Hart (2016))

If CEP holds, then the universal theory of $\mathcal{R}$ is computable.

- The conclusion means that for any *formal expression* $\sigma = \sup_{\|x\| \leq 1} \varphi(x)$ in the (model-theoretic) language of tracial von Neumann algebras, where φ is a continuous combination of traces of $*$-polynomials, we can effectively approximate its value $\sigma^{\mathcal{R}}$ in $\mathcal{R}$ up to any (rational) error.
- Lower bounds: brute force.
- Upper bounds: if $\sigma^{\mathcal{R}} \leq r$, then CEP tells us that $\sigma \leq r$ is a *logical consequence* of the theory of II_1 factors.
- By running formal proofs from the axioms of II_1 factors, the Completeness Theorem tells us we will eventually see that $\sigma \leq r$ is a *theorem*. (Soundness tells us no mistakes are made.)

On the other hand

Theorem (G. and Hart (2020))

The universal theory of $\mathcal{R}$ is not computable

- Of course we use $\mathrm{MIP}^* = \mathrm{RE}$, but how?
- We show that if $\mathrm{Th}_\forall(\mathcal{R})$ is computable, then we can effectively find upper bounds for $\mathrm{val}^*(\mathfrak{G})$, uniformly in the description of $\mathfrak{G}$, contradicting $\mathrm{MIP}^* = \mathrm{RE}$.
- But how? While $\mathrm{val}^*(\mathfrak{G}, p)$ is part of the formal language for a fixed p, we then sup over $C_{qs}(n, k)$, which is not a priori part of the formal language.
- Notation: $C_{qa}(n, k) = \overline{C_{qs}(n, k)}$.

On the other hand

Theorem (G. and Hart (2020))

The universal theory of $\mathcal{R}$ is not computable

- Of course we use $\mathrm{MIP}^* = \mathrm{RE}$, but how?
- We show that if $\mathrm{Th}_\forall(\mathcal{R})$ is computable, then we can effectively find upper bounds for $\mathrm{val}^*(\mathfrak{G})$, uniformly in the description of $\mathfrak{G}$, contradicting $\mathrm{MIP}^* = \mathrm{RE}$.
- But how? While $\mathrm{val}^*(\mathfrak{G}, p)$ is part of the formal language for a fixed p, we then sup over $C_{qs}(n, k)$, which is not a priori part of the formal language.
- Notation: $C_{qa}(n, k) = \overline{C_{qs}(n, k)}$.

On the other hand

Theorem (G. and Hart (2020))

The universal theory of $\mathcal{R}$ is not computable

- Of course we use $\mathrm{MIP}^* = \mathrm{RE}$, but how?
- We show that if $\mathrm{Th}_\forall(\mathcal{R})$ is computable, then we can effectively find upper bounds for $\mathrm{val}^*(\mathfrak{G})$, uniformly in the description of $\mathfrak{G}$, contradicting $\mathrm{MIP}^* = \mathrm{RE}$.
- But how? While $\mathrm{val}^*(\mathfrak{G}, p)$ is part of the formal language for a fixed p, we then sup over $C_{qs}(n,k)$, which is not a priori part of the formal language.
- Notation: $C_{qa}(n,k) = \overline{C_{qs}(n,k)}$.

On the other hand

Theorem (G. and Hart (2020))

The universal theory of $\mathcal{R}$ is not computable

- Of course we use $\mathrm{MIP}^* = \mathrm{RE}$, but how?
- We show that if $\mathrm{Th}_\forall(\mathcal{R})$ is computable, then we can effectively find upper bounds for $\mathrm{val}^*(\mathfrak{G})$, uniformly in the description of $\mathfrak{G}$, contradicting $\mathrm{MIP}^* = \mathrm{RE}$.
- But how? While $\mathrm{val}^*(\mathfrak{G}, p)$ is part of the formal language for a fixed p, we then sup over $C_{qs}(n, k)$, which is not a priori part of the formal language.
- Notation: $C_{qa}(n, k) = \overline{C_{qs}(n, k)}$.

On the other hand

Theorem (G. and Hart (2020))

The universal theory of $\mathcal{R}$ is not computable

- Of course we use $\mathrm{MIP}^* = \mathrm{RE}$, but how?
- We show that if $\mathrm{Th}_\forall(\mathcal{R})$ is computable, then we can effectively find upper bounds for $\mathrm{val}^*(\mathfrak{G})$, uniformly in the description of $\mathfrak{G}$, contradicting $\mathrm{MIP}^* = \mathrm{RE}$.
- But how? While $\mathrm{val}^*(\mathfrak{G}, p)$ is part of the formal language for a fixed p, we then sup over $C_{qs}(n, k)$, which is not a priori part of the formal language.
- Notation: $C_{qa}(n, k) = \overline{C_{qs}(n, k)}$.

Synchronous correlations and synchronous games

Definition

A correlation $p(a, b|x, y)$ is **synchronous** if $p(a, b|x, x) = 0$ whenever $a \neq b$. $C^s_{qa}(n, k)$ denotes the synchronous elements of $C_{qa}(n, k)$. $\text{s-val}^*(\mathfrak{G}) = \sup_{p \in C^s_{qa}(n,k)} \text{val}^*(\mathfrak{G}, p)$.

- Clearly $\text{s-val}^*(\mathfrak{G}) \leq \text{val}^*(\mathfrak{G})$.

Remark

The games in $\text{MIP}^* = \text{RE}$ are such that, if $\text{val}^*(\mathfrak{G}_{\mathcal{M}}) = 1$, then $\text{s-val}^*(\mathfrak{G}_{\mathcal{M}}) = 1$.

Synchronous correlations and synchronous games

Definition

A correlation $p(a, b|x, y)$ is **synchronous** if $p(a, b|x, x) = 0$ whenever $a \neq b$. $C_{qa}^s(n, k)$ denotes the synchronous elements of $C_{qa}(n, k)$. $\text{s-val}^*(\mathfrak{G}) = \sup_{p \in C_{qa}^s(n,k)} \text{val}^*(\mathfrak{G}, p)$.

- Clearly $\text{s-val}^*(\mathfrak{G}) \leq \text{val}^*(\mathfrak{G})$.

Remark

The games in $\text{MIP}^* = \text{RE}$ are such that, if $\text{val}^*(\mathfrak{G}_\mathcal{M}) = 1$, then $\text{s-val}^*(\mathfrak{G}_\mathcal{M}) = 1$.

Synchronous correlations and synchronous games

Definition

A correlation $p(a, b|x, y)$ is **synchronous** if $p(a, b|x, x) = 0$ whenever $a \neq b$. $C^s_{qa}(n, k)$ denotes the synchronous elements of $C_{qa}(n, k)$. $\text{s-val}^*(\mathfrak{G}) = \sup_{p \in C^s_{qa}(n,k)} \text{val}^*(\mathfrak{G}, p)$.

- Clearly $\text{s-val}^*(\mathfrak{G}) \leq \text{val}^*(\mathfrak{G})$.

Remark

The games in $\text{MIP}^* = \text{RE}$ are such that, if $\text{val}^*(\mathfrak{G}_{\mathcal{M}}) = 1$, then $\text{s-val}^*(\mathfrak{G}_{\mathcal{M}}) = 1$.

This looks a little better...

Theorem (Kim, Paulsen, and Schaufhauser)

$p \in C_{qa}^{s}(n,k)$ if and only if: for each $x \in [n]$, there are projections $e_1^x, \ldots, e_k^x \in \mathcal{R}^{\mathcal{U}}$ such that $\sum_{a=1}^{k} e_a^x = 1$ (and ditto for $y \in [n]$) such that $p(a,b|x,y) = \mathrm{tr}(e_a^x e_b^y)$.

Corollary

For any nonlocal game $\mathfrak{G}$,

$$\text{s-val}^*(\mathfrak{G}) = \left(\sup_{e_a^x} \sum_{x,y} \lambda(x,y) \sum_{a,b} D(a,b,x,y)\,\mathrm{tr}(e_a^x e_b^y) \right)^{\mathcal{R}}.$$

This looks a lot more like a formula in our langugage.

This looks a little better...

Theorem (Kim, Paulsen, and Schaufhauser)

$p \in C^s_{qa}(n,k)$ if and only if: for each $x \in [n]$, there are projections $e^x_1, \ldots, e^x_k \in \mathcal{R}^{\mathcal{U}}$ such that $\sum_{a=1}^k e^x_a = 1$ (and ditto for $y \in [n]$) such that $p(a,b|x,y) = \mathrm{tr}(e^x_a e^y_b)$.

Corollary

For any nonlocal game $\mathfrak{G}$,

$$\text{s-val}^*(\mathfrak{G}) = \left(\sup_{e^x_a} \sum_{x,y} \lambda(x,y) \sum_{a,b} D(a,b,x,y)\,\mathrm{tr}(e^x_a e^y_b)\right)^{\mathcal{R}}.$$

This looks a lot more like a formula in our langugage.

Some technical wrinkles

- The remaining issue to discuss is that we are not just taking the supremum over all elements in $\mathcal{R}$, but only those satisfying a particular property.
- This is only "allowable" if the set of elements that satisfies that property is a **definable set**.
- Fortunately for us, this is the case, and Kim, Paulsen, and Schaufhauser themselves proved it!
- Then the translation from the expression using the definable set to an approximating family of "legitimate" sentences needs to be done effectively and the resulting sentences need to be universal...

Some technical wrinkles

- The remaining issue to discuss is that we are not just taking the supremum over all elements in $\mathcal{R}$, but only those satisfying a particular property.
- This is only "allowable" if the set of elements that satisfies that property is a **definable set**.
- Fortunately for us, this is the case, and Kim, Paulsen, and Schaufhauser themselves proved it!
- Then the translation from the expression using the definable set to an approximating family of "legitimate" sentences needs to be done effectively and the resulting sentences need to be universal...

Some technical wrinkles

- The remaining issue to discuss is that we are not just taking the supremum over all elements in $\mathcal{R}$, but only those satisfying a particular property.
- This is only "allowable" if the set of elements that satisfies that property is a **definable set**.
- Fortunately for us, this is the case, and Kim, Paulsen, and Schaufhauser themselves proved it!
- Then the translation from the expression using the definable set to an approximating family of "legitimate" sentences needs to be done effectively and the resulting sentences need to be universal...

Some technical wrinkles

- The remaining issue to discuss is that we are not just taking the supremum over all elements in $\mathcal{R}$, but only those satisfying a particular property.
- This is only “allowable” if the set of elements that satisfies that property is a **definable set**.
- Fortunately for us, this is the case, and Kim, Paulsen, and Schaufhauser themselves proved it!
- Then the translation from the expression using the definable set to an approximating family of “legitimate” sentences needs to be done effectively and the resulting sentences need to be universal...

A Gödelian style refutation of CEP

- Perhaps it is too arrogant to simply expect all tracial von Neumann algebras to embed into $\mathcal{R}^{\mathcal{U}}$, but maybe by adding some "reasonable" set of extra conditions, we can ensure $\mathcal{R}^{\mathcal{U}}$-embeddability.
- Nope!

Theorem (G. and Hart)

Suppose that T is any "effective" satisfiable set of (first-order) conditions extending the axioms for being a II_1 factor. Then there is a II_1 factor satisfying T that does not embed in $\mathcal{R}^{\mathcal{U}}$.

One can make similar statements for any unital, simple, nuclear C^*-algebra with the uniform Dixmier property.

A Gödelian style refutation of CEP

- Perhaps it is too arrogant to simply expect all tracial von Neumann algebras to embed into $\mathcal{R}^{\mathcal{U}}$, but maybe by adding some "reasonable" set of extra conditions, we can ensure $\mathcal{R}^{\mathcal{U}}$-embeddability.
- Nope!

Theorem (G. and Hart)

Suppose that T is any "effective" satisfiable set of (first-order) conditions extending the axioms for being a II_1 factor. Then there is a II_1 factor satisfying T that does not embed in $\mathcal{R}^{\mathcal{U}}$.

One can make similar statements for any unital, simple, nuclear C^*-algebra with the uniform Dixmier property.

A Gödelian style refutation of CEP

- Perhaps it is too arrogant to simply expect all tracial von Neumann algebras to embed into $\mathcal{R}^{\mathcal{U}}$, but maybe by adding some "reasonable" set of extra conditions, we can ensure $\mathcal{R}^{\mathcal{U}}$-embeddability.
- Nope!

Theorem (G. and Hart)

Suppose that T is any "effective" satisfiable set of (first-order) conditions extending the axioms for being a II_1 factor. Then there is a II_1 factor satisfying T that does not embed in $\mathcal{R}^{\mathcal{U}}$.

One can make similar statements for any unital, simple, nuclear C^*-algebra with the uniform Dixmier property.

$\mathcal{Z}$EP

- Using the failure of CEP, one can derive a failure of the well-known MF problem, which asks if every unital **stably finite** C^*-algebra embeds into an ultrapower of the universal UHF algebra $\mathcal{Q}$.
- One particular consequence of our Gödelian-style results for C^*-algebras is the following purely operator-algebraic result, which shows that the **stably projectionless** version of the MF problem also has a negative solution:

Theorem

There is a unital stably projectionless C^*-algebra that does not embed into an ultrapower of the Jiang-Su algebra $\mathcal{Z}$.

As far as we know, this theorem has no purely operator algebraic proof.

$\mathcal{Z}$EP

- Using the failure of CEP, one can derive a failure of the well-known MF problem, which asks if every unital **stably finite** C^*-algebra embeds into an ultrapower of the universal UHF algebra $\mathcal{Q}$.
- One particular consequence of our Gödelian-style results for C^*-algebras is the following purely operator-algebraic result, which shows that the **stably projectionless** version of the MF problem also has a negative solution:

Theorem

There is a unital stably projectionless C^*-algebra that does not embed into an ultrapower of the Jiang-Su algebra $\mathcal{Z}$.

As far as we know, this theorem has no purely operator algebraic proof.

$\mathcal{Z}$EP

- Using the failure of CEP, one can derive a failure of the well-known MF problem, which asks if every unital **stably finite** C^*-algebra embeds into an ultrapower of the universal UHF algebra $\mathcal{Q}$.
- One particular consequence of our Gödelian-style results for C^*-algebras is the following purely operator-algebraic result, which shows that the **stably projectionless** version of the MF problem also has a negative solution:

Theorem

There is a unital stably projectionless C^*-algebra that does not embed into an ultrapower of the Jiang-Su algebra $\mathcal{Z}$.

As far as we know, this theorem has no purely operator algebraic proof.

$\mathcal{Z}$EP

- Using the failure of CEP, one can derive a failure of the well-known MF problem, which asks if every unital **stably finite** C^*-algebra embeds into an ultrapower of the universal UHF algebra $\mathcal{Q}$.
- One particular consequence of our Gödelian-style results for C^*-algebras is the following purely operator-algebraic result, which shows that the **stably projectionless** version of the MF problem also has a negative solution:

Theorem

There is a unital stably projectionless C^*-algebra that does not embed into an ultrapower of the Jiang-Su algebra $\mathcal{Z}$.

As far as we know, this theorem has no purely operator algebraic proof.

A reformulation of our main theorem

- Let $m_1, \ldots, m_L$ enumerate all *-monomials in the variables $x_1, \ldots, x_n$ of total degree at most d.
- We consider the map $\mu_{n,d} : \mathcal{R}_1^n \to \mathbb{D}^L$ given by $\mu_{n,d}(\vec{a}) = (\mathrm{tr}(m_i(\vec{a})) \ : \ i = 1, \ldots, L)$.
- We let $X(n,d)$ denote the range of $\mu_{n,d}$ and $X(n,d,p)$ be the image of $(M_p(\mathbb{C}))_1$ under $\mu_{n,d}$.
- Notice that $\bigcup_{p \in \mathbb{N}} X(n,d,p)$ is dense in $X(n,d)$.

Theorem (G. and Hart)

The following statements are equivalent:

1. *The universal theory of $\mathcal{R}$ is computable.*
2. *There is a computable function $F : \mathbb{N}^3 \to \mathbb{N}$ such that, for every $n, d, k \in \mathbb{N}$, $X(n,d,F(n,d,k))$ is $\frac{1}{k}$-dense in $X(n,d)$.*

A reformulation of our main theorem

- Let $m_1, \ldots, m_L$ enumerate all *-monomials in the variables $x_1, \ldots, x_n$ of total degree at most d.
- We consider the map $\mu_{n,d} : \mathcal{R}_1^n \to \mathbb{D}^L$ given by $\mu_{n,d}(\vec{a}) = (\mathrm{tr}(m_i(\vec{a})) \ : \ i = 1, \ldots, L)$.
- We let $X(n,d)$ denote the range of $\mu_{n,d}$ and $X(n,d,p)$ be the image of $(M_p(\mathbb{C}))_1$ under $\mu_{n,d}$.
- Notice that $\bigcup_{p\in\mathbb{N}} X(n,d,p)$ is dense in $X(n,d)$.

Theorem (G. and Hart)

The following statements are equivalent:

1. *The universal theory of $\mathcal{R}$ is computable.*
2. *There is a computable function $F : \mathbb{N}^3 \to \mathbb{N}$ such that, for every $n, d, k \in \mathbb{N}$, $X(n,d,F(n,d,k))$ is $\frac{1}{k}$-dense in $X(n,d)$.*

References

- Isaac Goldbring and Bradd Hart, *A computability-theoretic reformulation of the Connes Embedding Problem*, Bulletin of Symbolic Logic, **22** (2016), 238-248.
- Isaac Goldbring and Bradd Hart, *The universal theory of the hyperfinite II_1 factor is not comutable*, arXiv 2006.05629.
- Zhengfeng Ji, Anand Natarajan, Thomas Vidick, John Wright, and Henry Yuen, $MIP^* = RE$, arXiv 2001.04383.
- Thomas Vidick, *From operator algebras to complexity theory and back*, Notices of the AMS, November 2019.