Spatial Isolation ⇒ Zero Knowledge Even in a Quantum World

TOM GUR

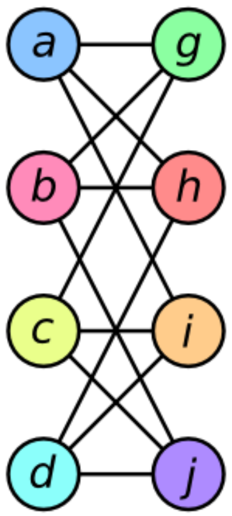
Berkeley
UNIVERSITY OF CALIFORNIA

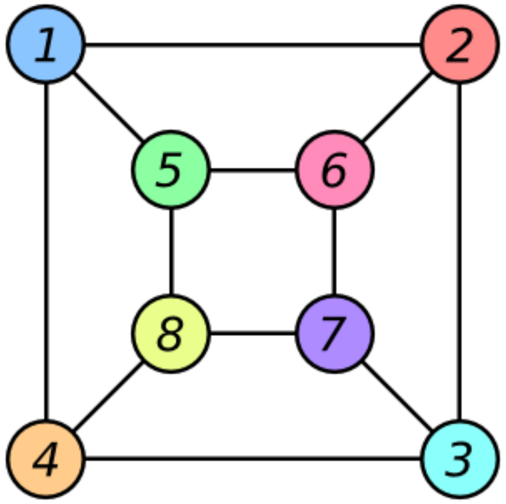
Joint work with Alessandro Chiesa, Michael Forbes, and Nicholas Spooner

The problem

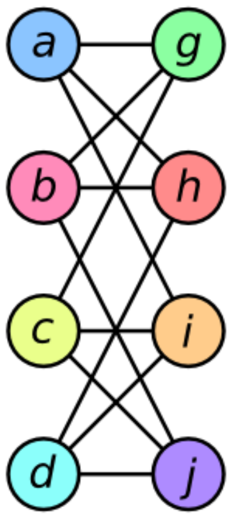
Zero Knowledge

Zero Knowledge

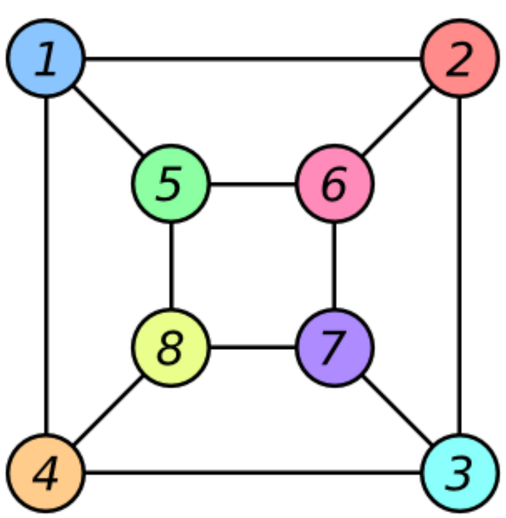




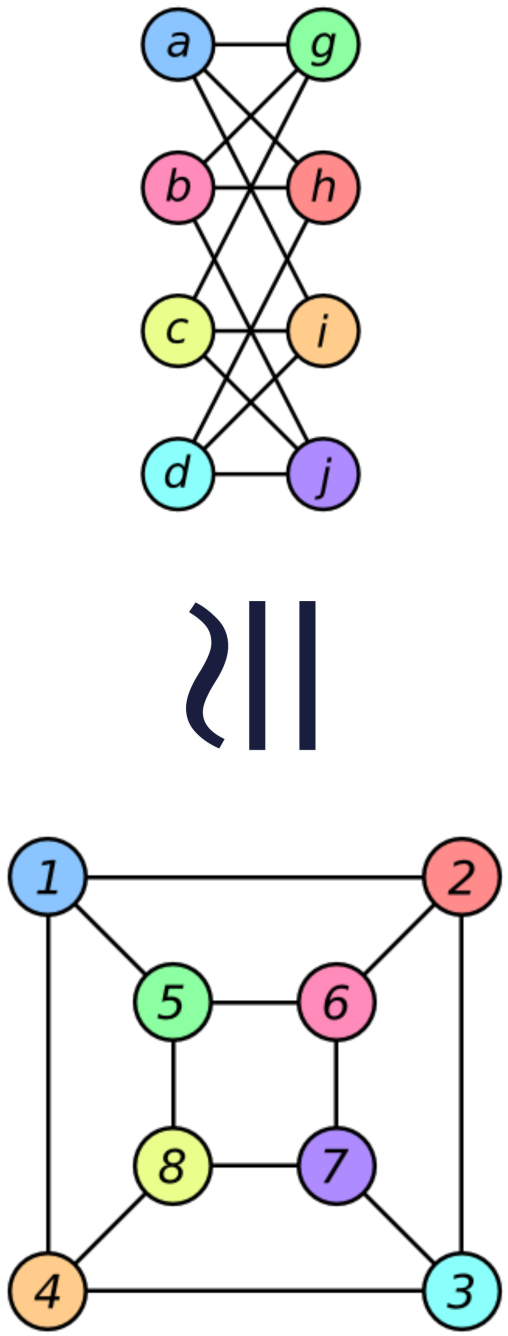
Zero Knowledge

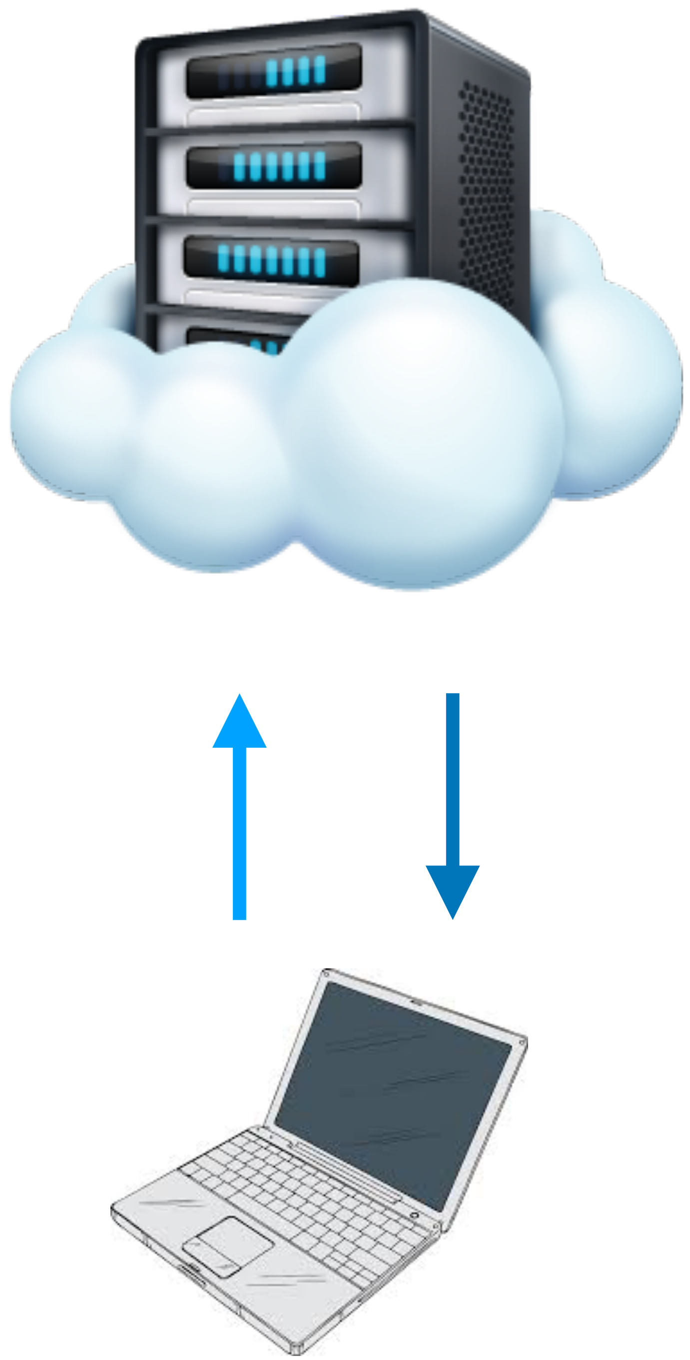


≅



Zero Knowledge

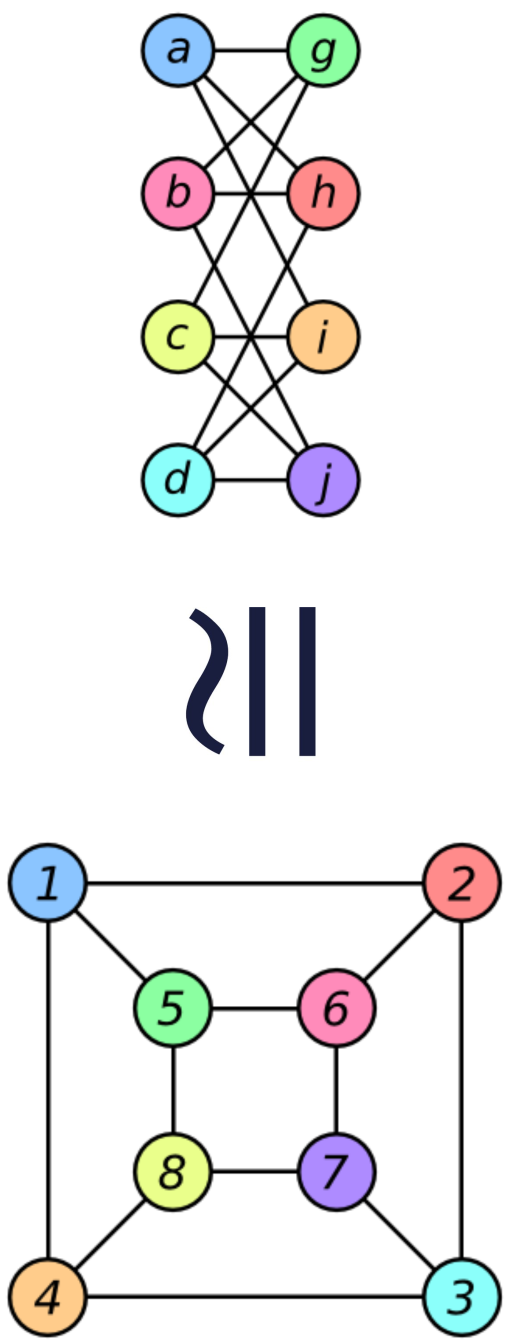


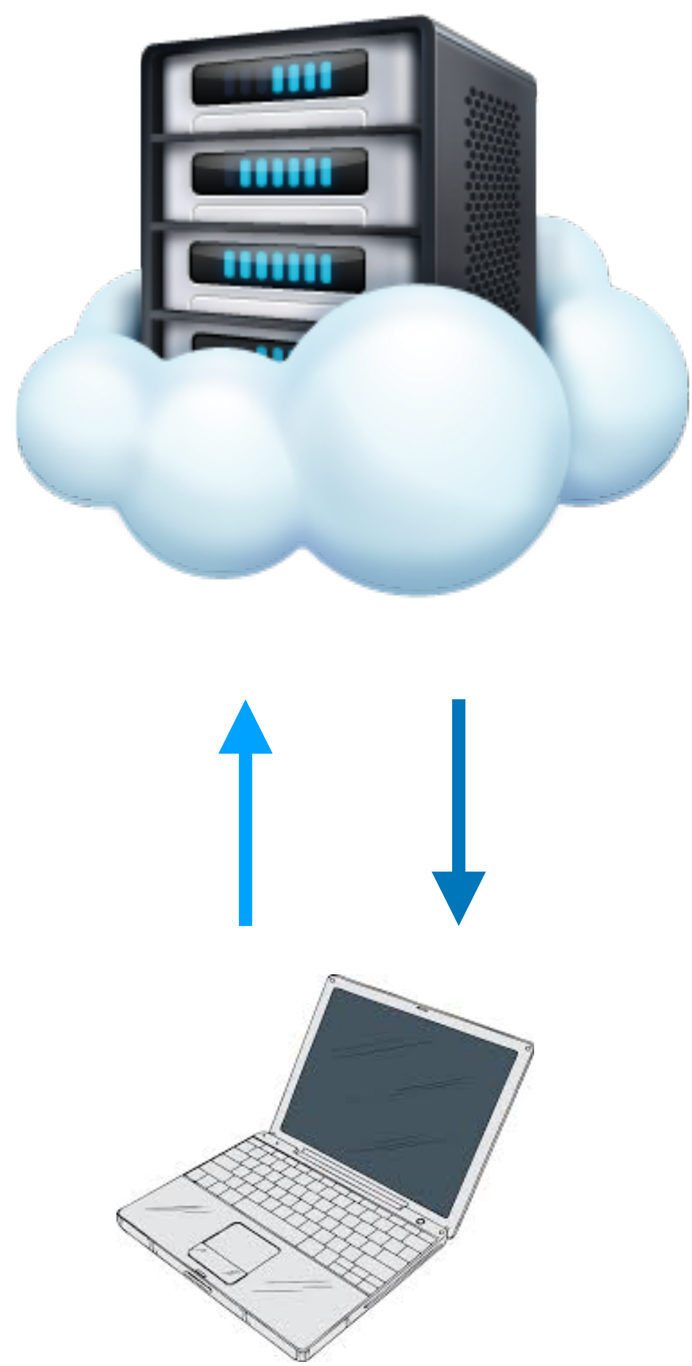


Zero Knowledge

Zero-Knowledge Proofs

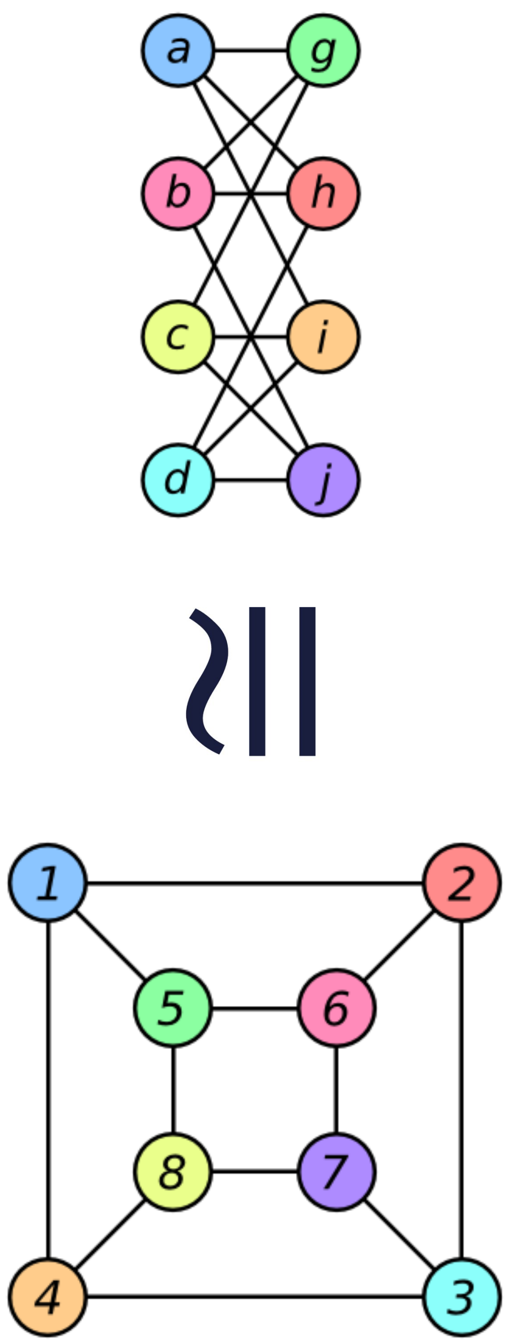
[Goldwasser-Micali-Rackoff 89]

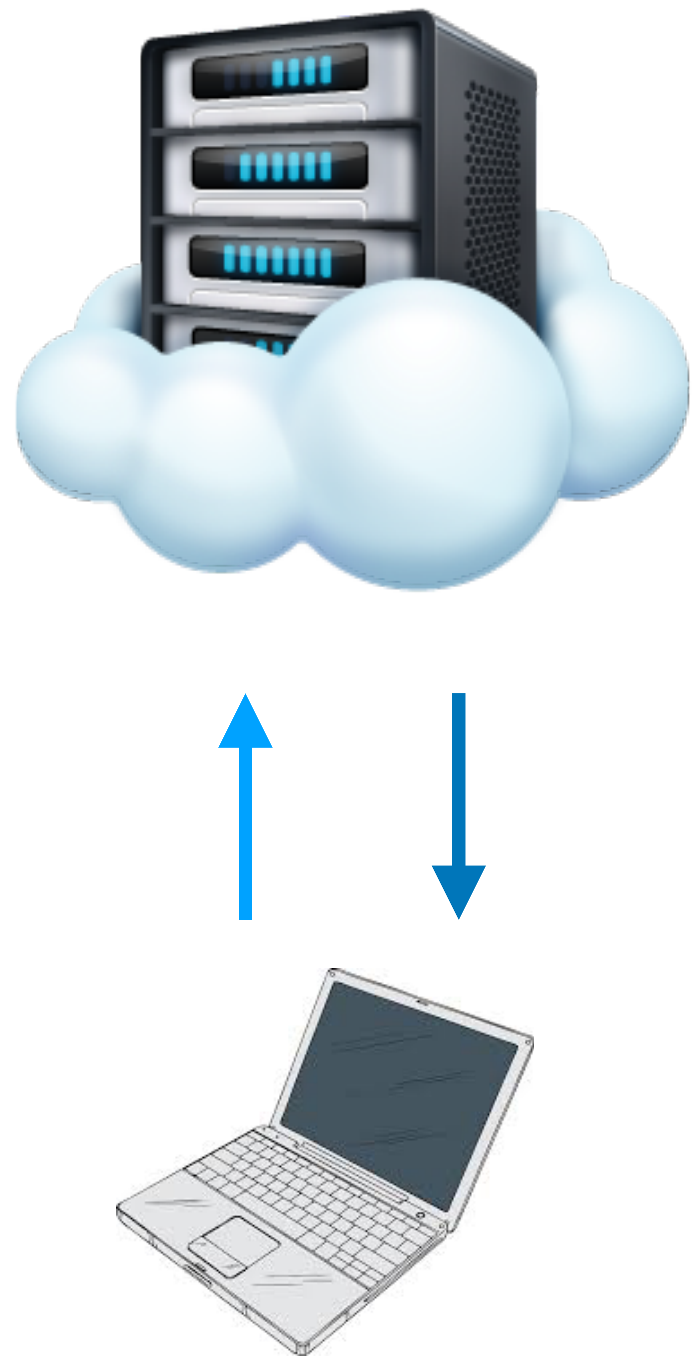




Zero-Knowledge Proofs

[Goldwasser-Micali-Rackoff 89]





Cryptographic assumptions (OWF)

↓

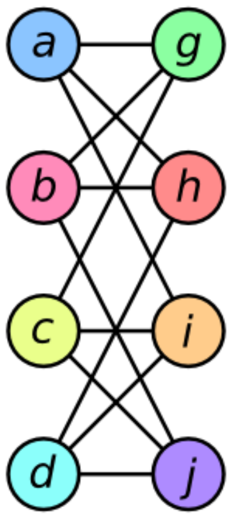
ZK for NP

[Goldreich-Micali-Wigderson 91]

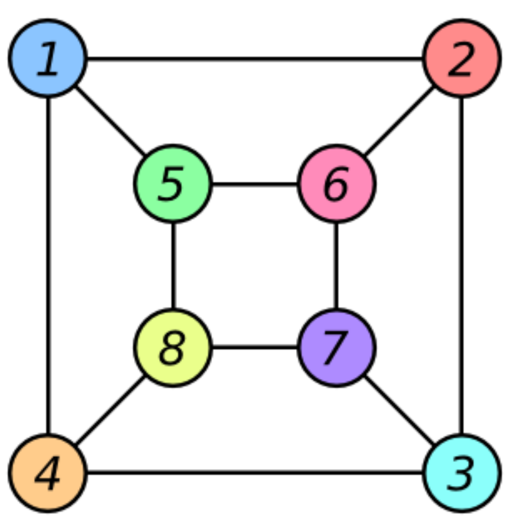
Zero Knowledge

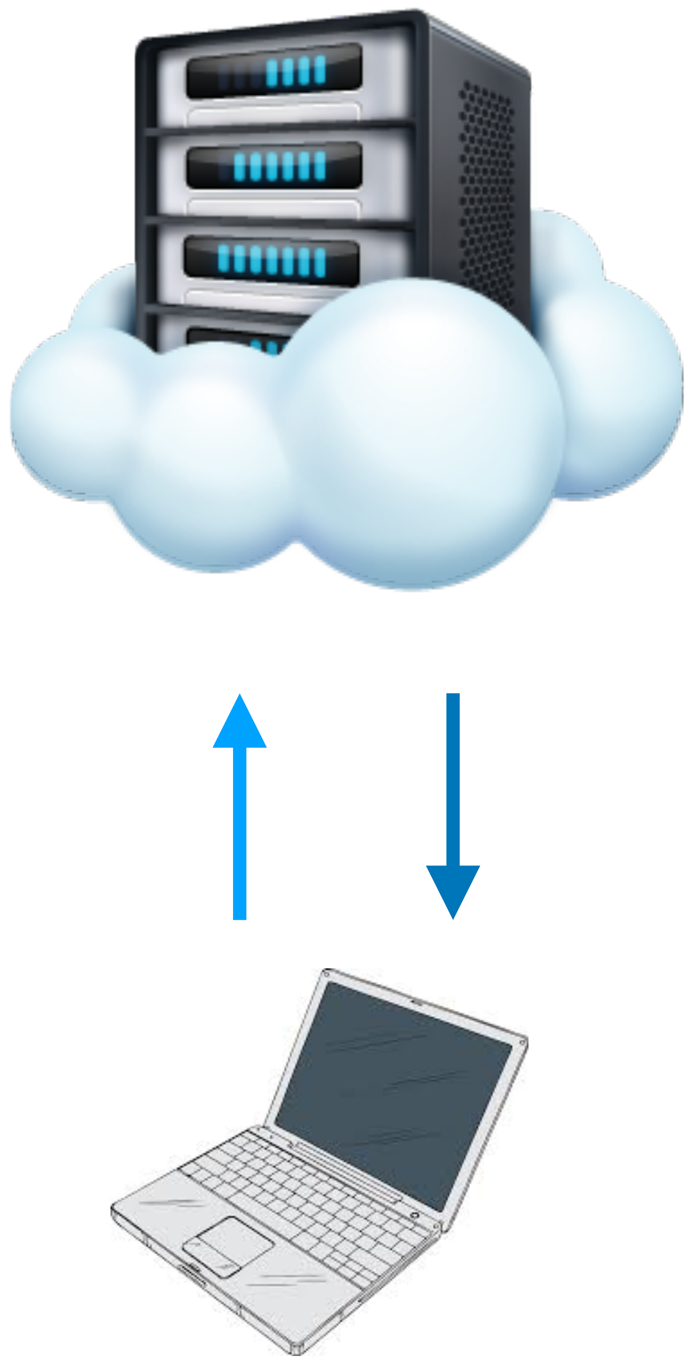
Zero-Knowledge Proofs

[Goldwasser-Micali-Rackoff 89]



$\cong$





Cryptographic assumptions (OWF)

⇓

ZK for NP

[Goldreich-Micali-Wigderson 91]

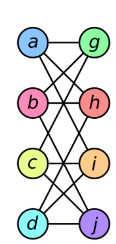
Cryptographic assumptions are **necessary**

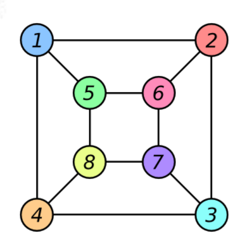
[Ostrovsky-Wigderson 93]

Spatial Isolation ⇒ Zero Knowledge

Multi-prover Interactive Proofs (MIP)

[BenOr-Goldwasser-Kilian-Wigderson 88]

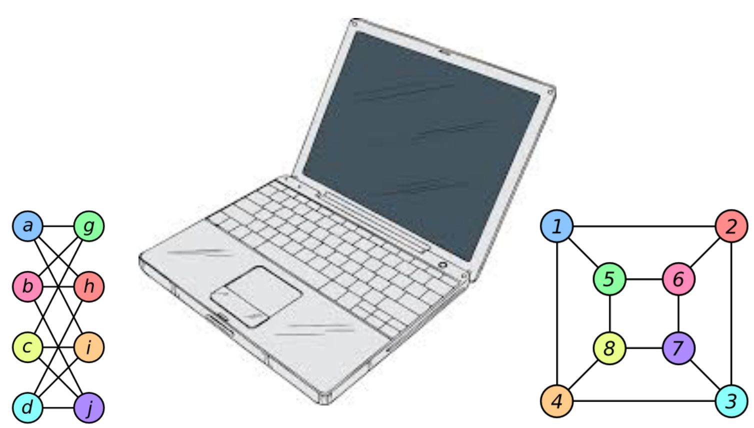




Spatial Isolation ⇒ Zero Knowledge

Multi-prover Interactive Proofs (MIP)

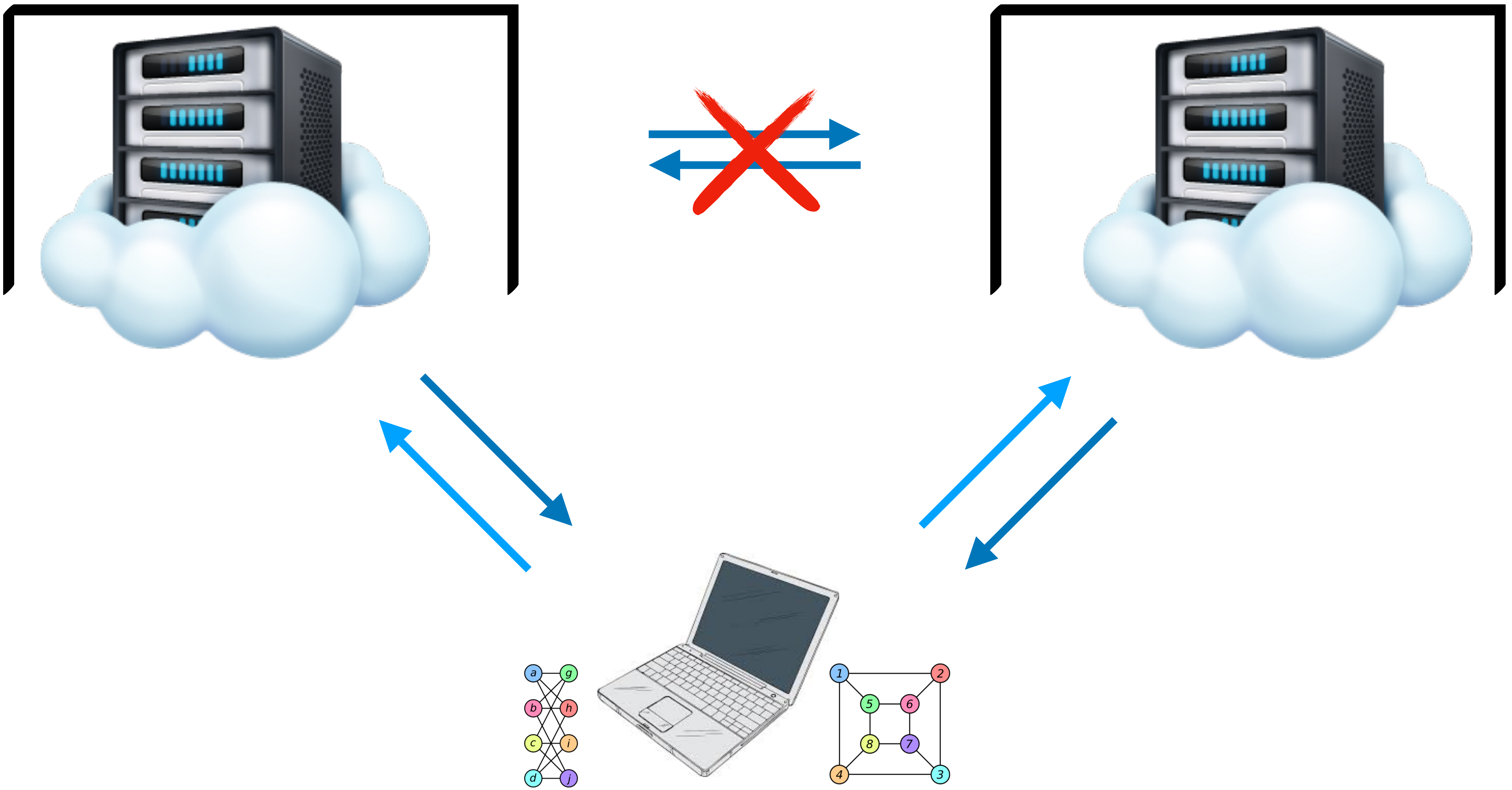
[BenOr-Goldwasser-Kilian-Wigderson 88]



Spatial Isolation ⇒ Zero Knowledge

Multi-prover Interactive Proofs (MIP)

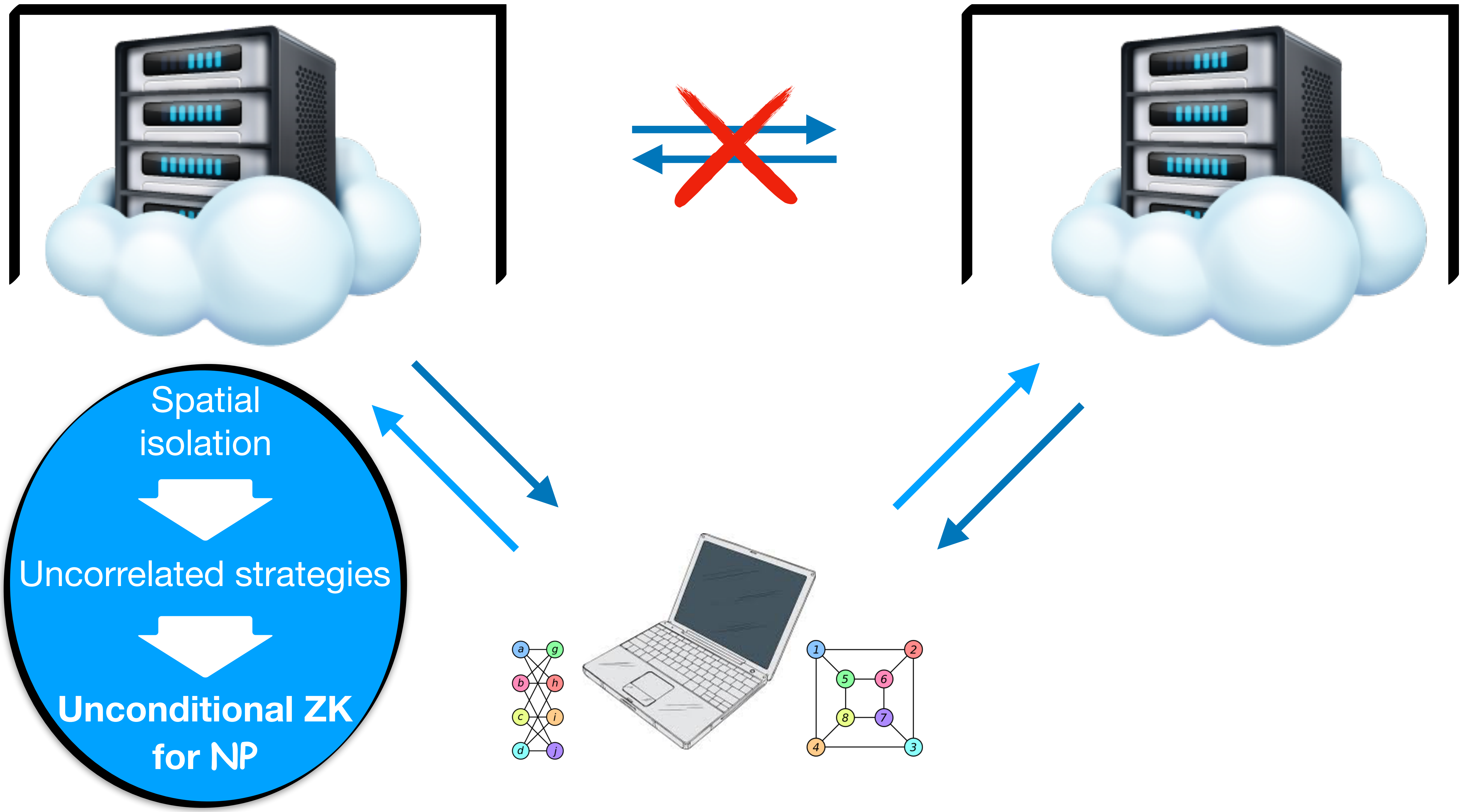
[BenOr-Goldwasser-Kilian-Wigderson 88]



Spatial Isolation ⇒ Zero Knowledge

Multi-prover Interactive Proofs (MIP)

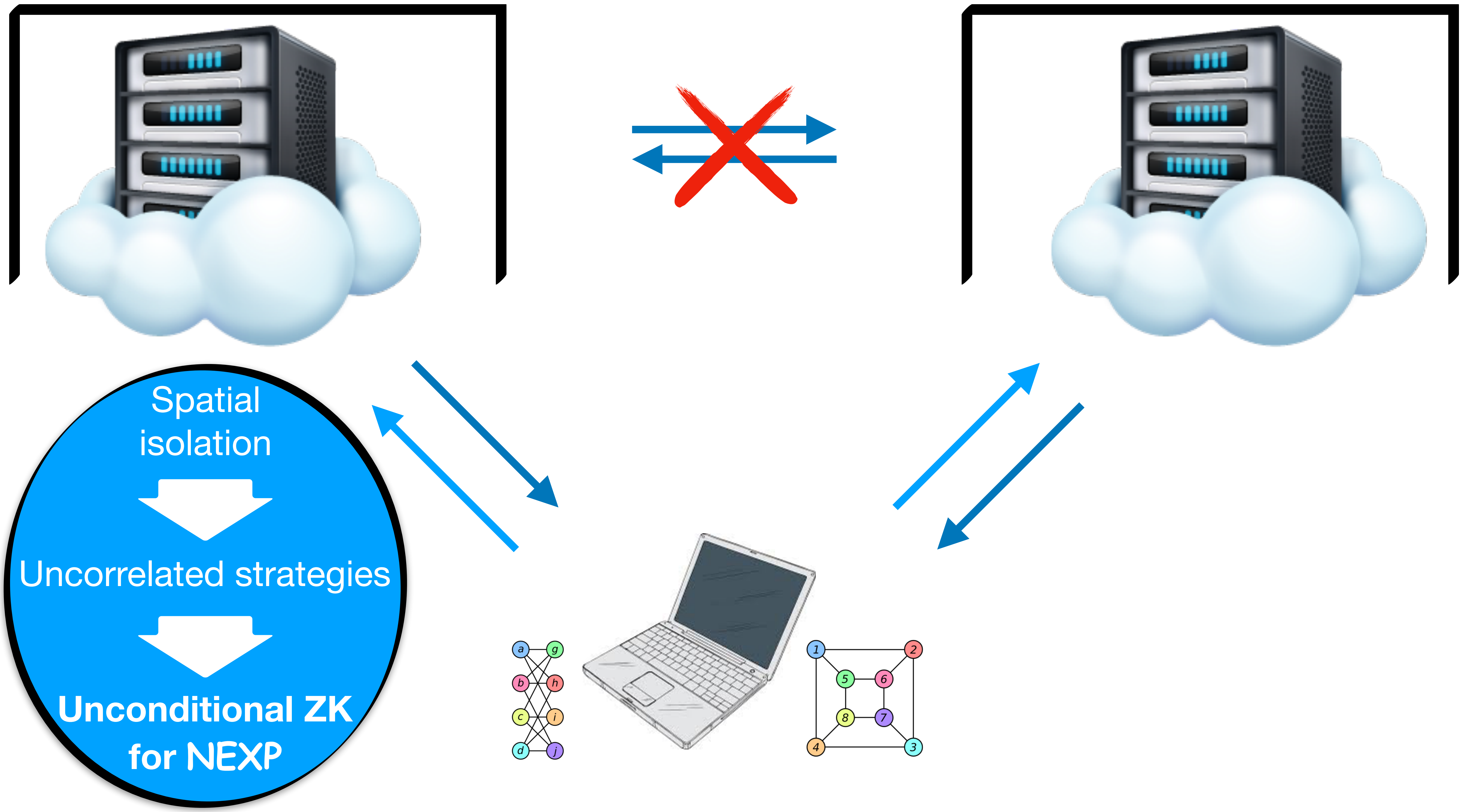
[BenOr-Goldwasser-Kilian-Wigderson 88]



Spatial Isolation ⇒ Zero Knowledge

Multi-prover Interactive Proofs (MIP)

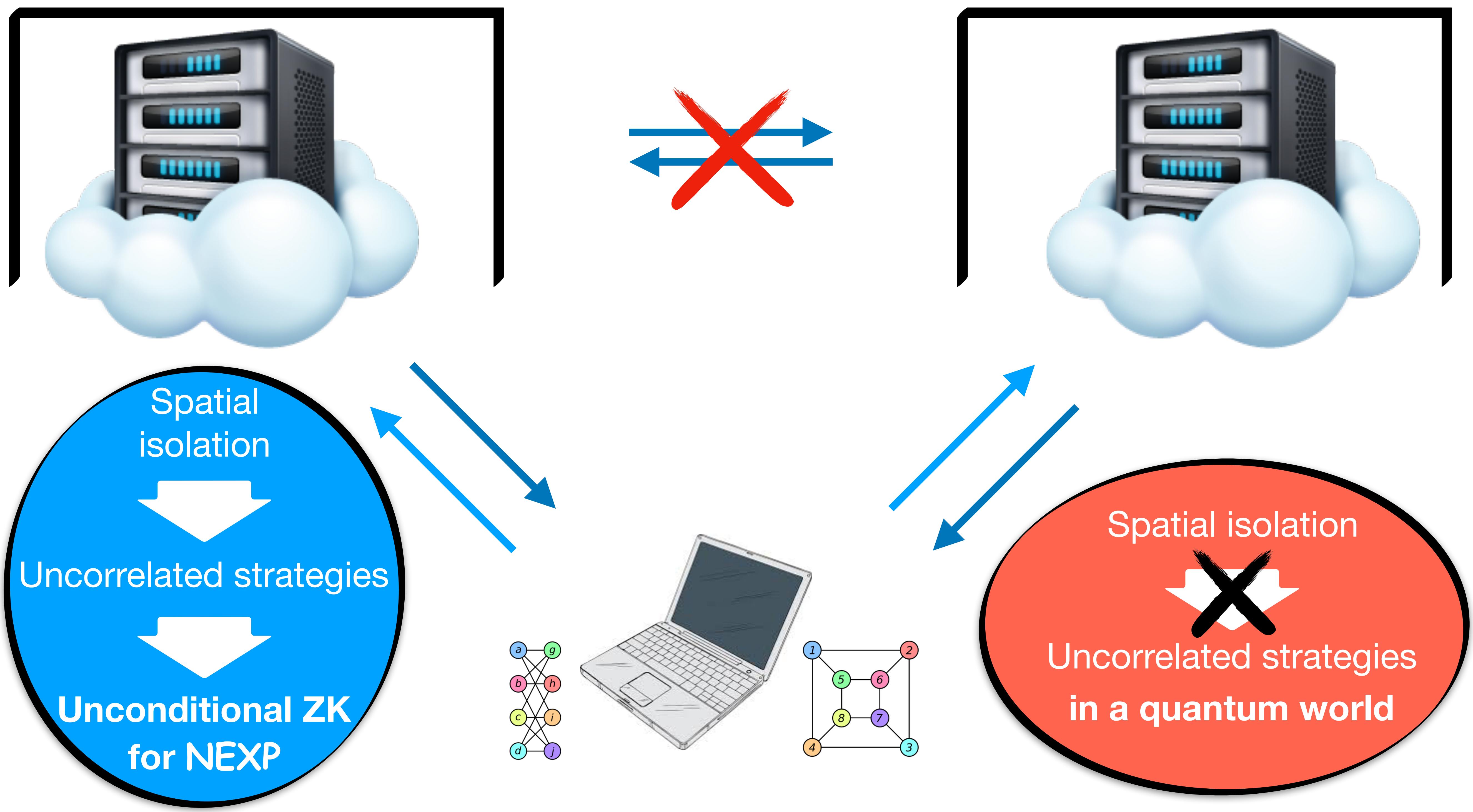
[BenOr-Goldwasser-Kilian-Wigderson 88]



Spatial Isolation ⇒ Zero Knowledge

Multi-prover Interactive Proofs (MIP)

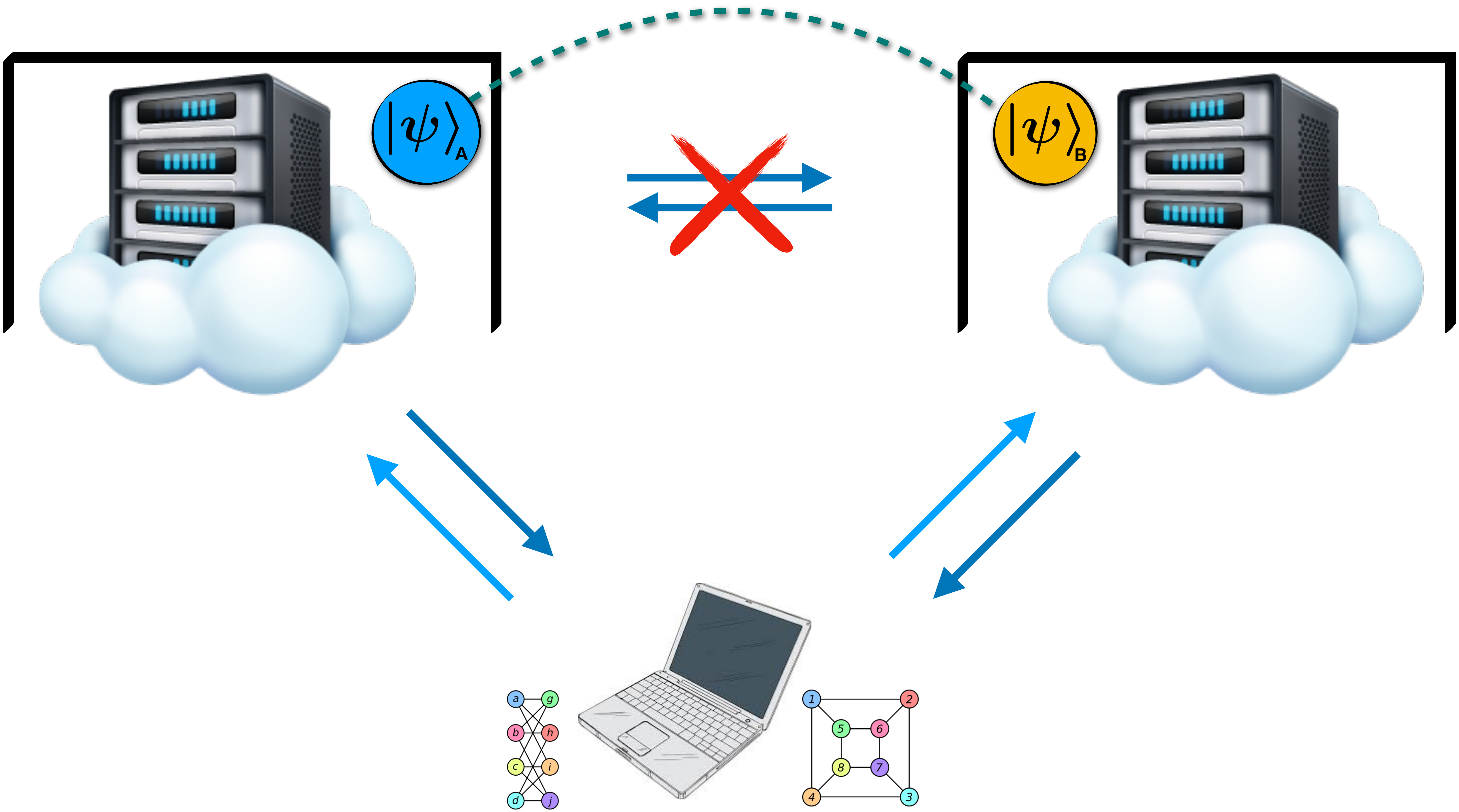
[BenOr-Goldwasser-Kilian-Wigderson 88]



Quantum Entanglement

MIP*

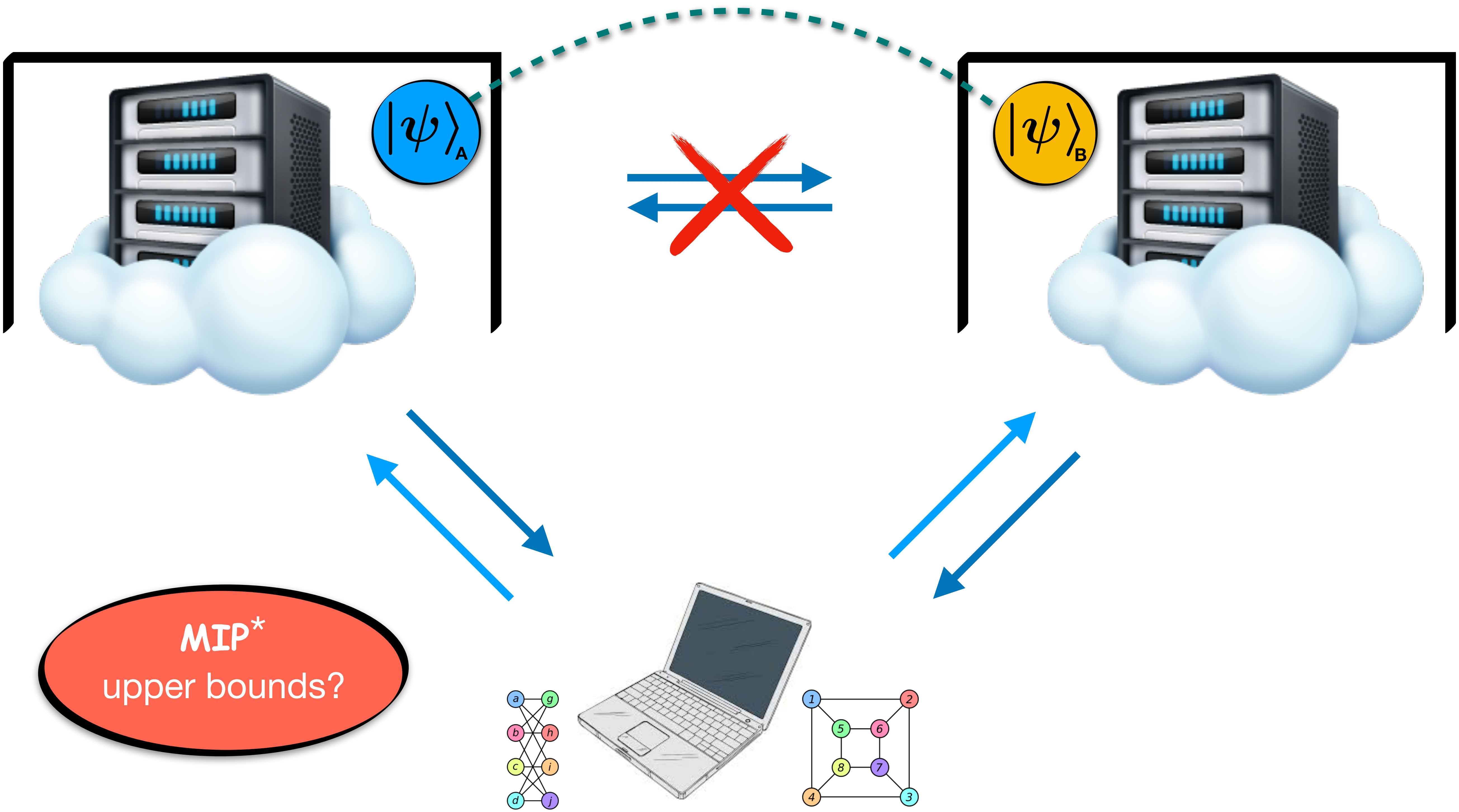
[Cleve-Hoyer-Toner-Watrous 04]



Quantum Entanglement

MIP*

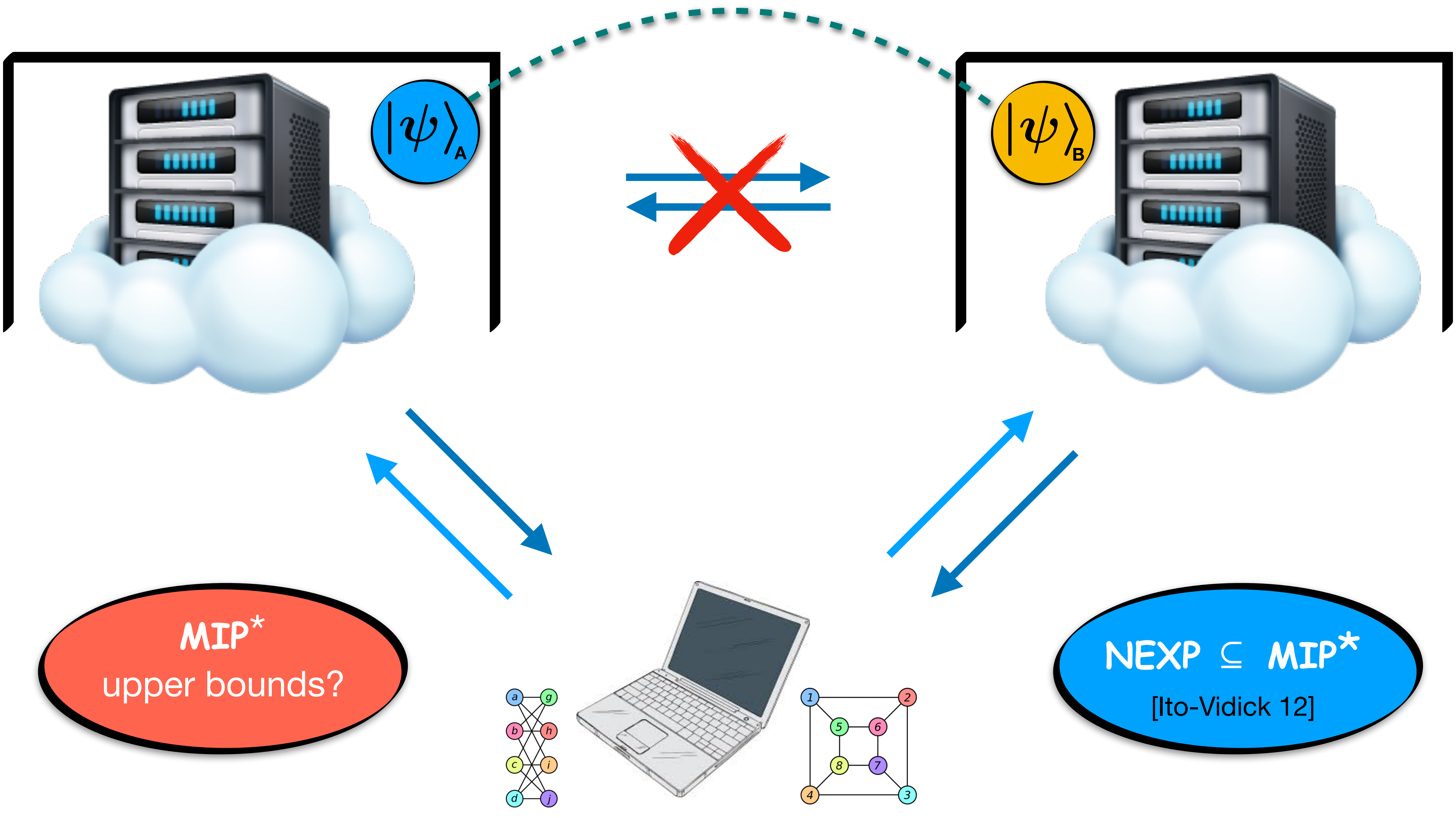
[Cleve-Hoyer-Toner-Watrous 04]



Quantum Entanglement

MIP*

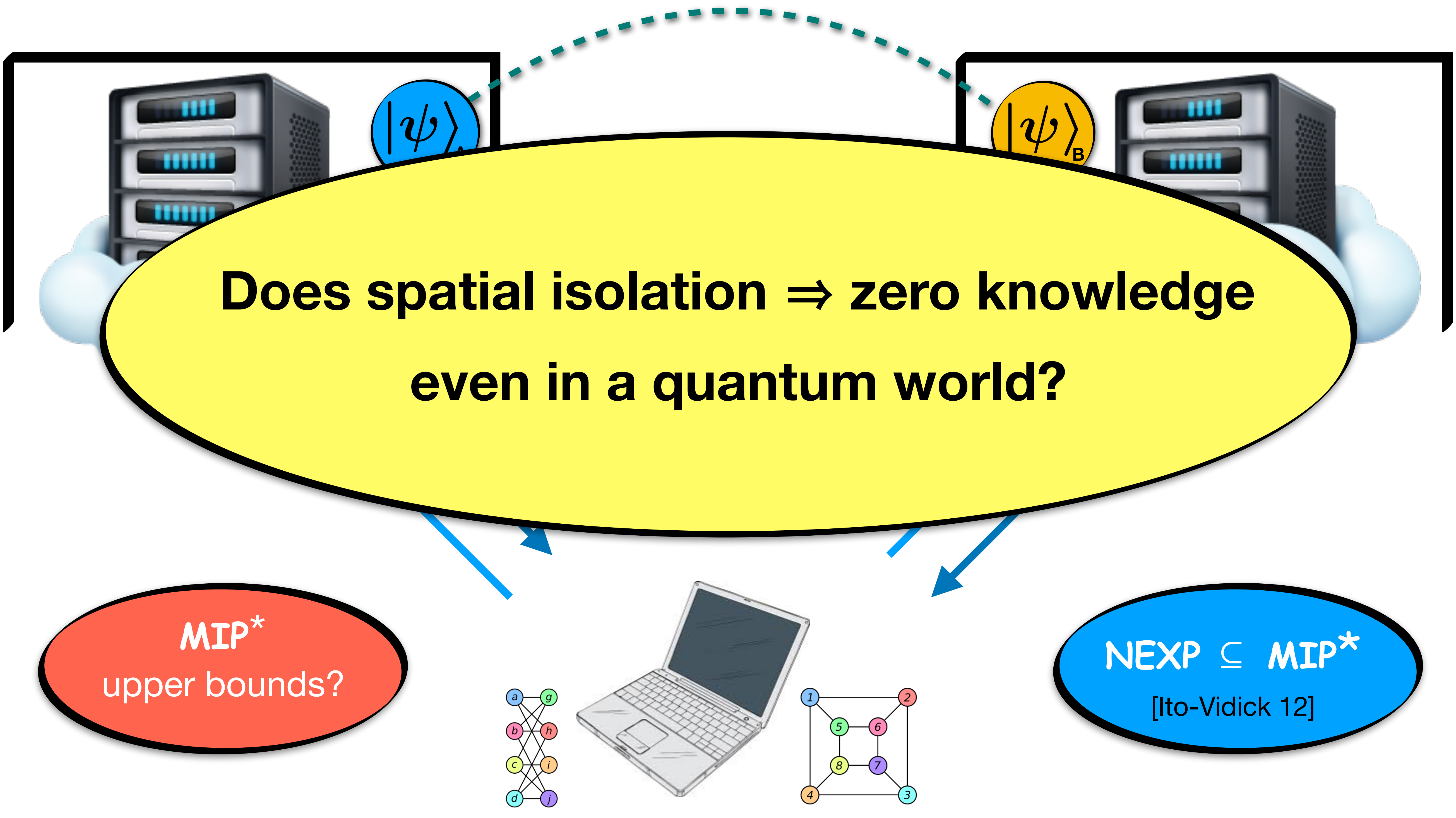
[Cleve-Hoyer-Toner-Watrous 04]



Quantum Entanglement

MIP*

[Cleve-Hoyer-Toner-Watrous 04]



Spatial isolation => zero knowledge
even in a quantum world

Yes!

Spatial isolation => zero knowledge **even in a quantum world**

Theorem: $NEXP \subseteq ZK\text{-}MIP^*$

Spatial isolation => zero knowledge
even in a quantum world

Theorem: NEXP ⊆ ZK-MIP*

The challenge

Spatial isolation => zero knowledge
even in a quantum world

Theorem: **NEXP ⊆ ZK-MIP***

The challenge

We know that: **NEXP ⊆ MIP*** [IV12]

Spatial isolation => zero knowledge
even in a quantum world

Theorem: $\mathbf{NEXP} \subseteq \mathbf{ZK\text{-}MIP^*}$

The challenge

We know that: $\mathbf{NEXP} \subseteq \mathbf{MIP^*}$ [IV12]

$\mathbf{NEXP} \subseteq \mathbf{ZK\text{-}MIP}$ [BGKW88]

Spatial isolation => zero knowledge **even in a quantum world**

Theorem: NEXP ⊆ ZK-MIP*

The challenge

We know that: NEXP ⊆ MIP* [IV12]

NEXP ⊆ ZK-MIP [BGKW88]

Why not combine them?

Spatial isolation => zero knowledge **even in a quantum world**

Theorem: NEXP ⊆ ZK-MIP*

The challenge

We know that: NEXP ⊆ MIP* [IV12]

NEXP ⊆ ZK-MIP [BGKW88]

Why not combine them?

Current MIP techniques are **ALGEBRAIC**

Spatial isolation => zero knowledge
even in a quantum world

Theorem: $NEXP \subseteq ZK\text{-}MIP^*$

The challenge

We know that: $NEXP \subseteq MIP^*$ [IV12]

$NEXP \subseteq ZK\text{-}MIP$ [BGKW88]

Why not combine them?

Current MIP techniques are **ALGEBRAIC**

(Previous) zero-knowledge techniques were **COMBINATORIAL**

Spatial isolation => zero knowledge **even in a quantum world**

Theorem: NEXP $\subseteq$ ZK-MIP*

The challenge

We know that: NEXP $\subseteq$ MIP* [IV12]

NEXP $\subseteq$ ZK-MIP [BGKW88]

Why not combine them?

Current MIP techniques are **ALGEBRAIC**

(Previous) zero-knowledge techniques were **COMBINATORIAL**

Technique Incompatibility

Spatial isolation => zero knowledge **even in a quantum world**

Spatial isolation => zero knowledge
even in a quantum world

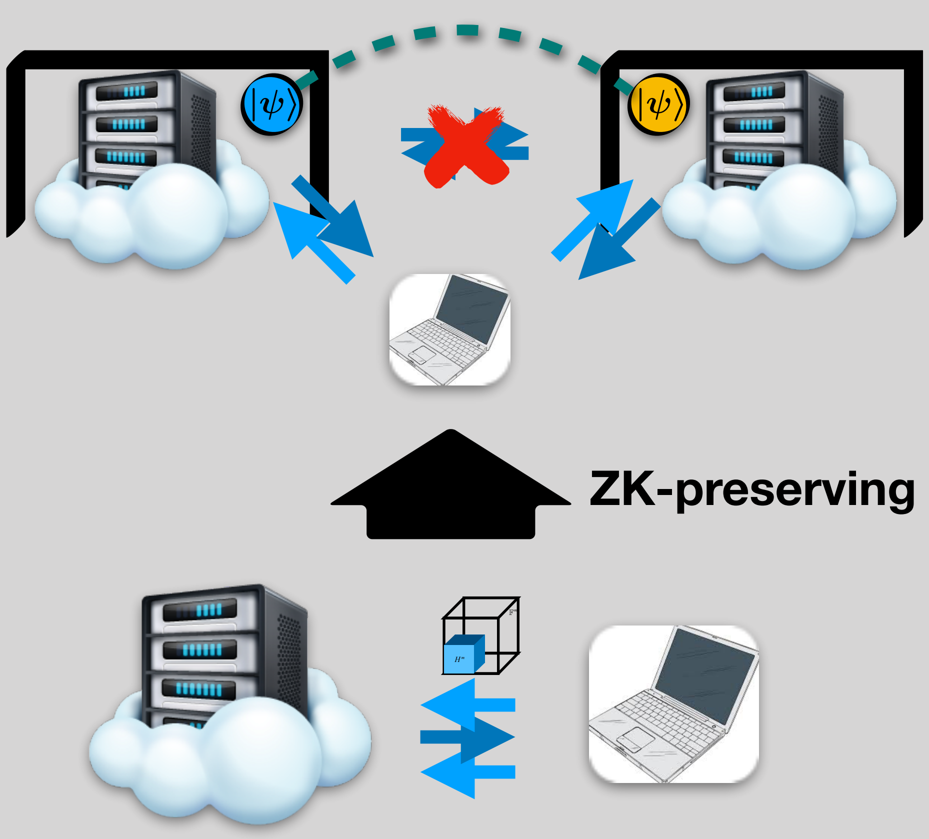
Theorem: $\mathrm{NEXP} \subseteq \mathrm{ZK\text{-}MIP}^*$

Spatial isolation => zero knowledge **even in a quantum world**

Theorem: NEXP ⊆ ZK-MIP*

Proof in 2 steps:

Lifting lemma

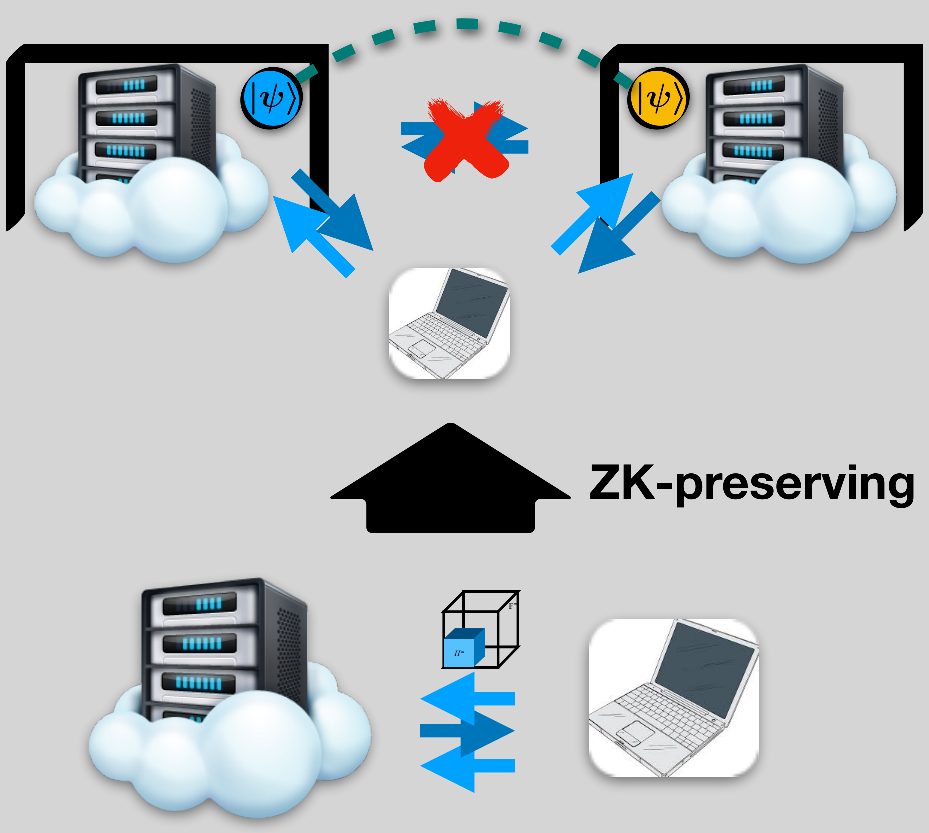


Spatial isolation => zero knowledge
even in a quantum world

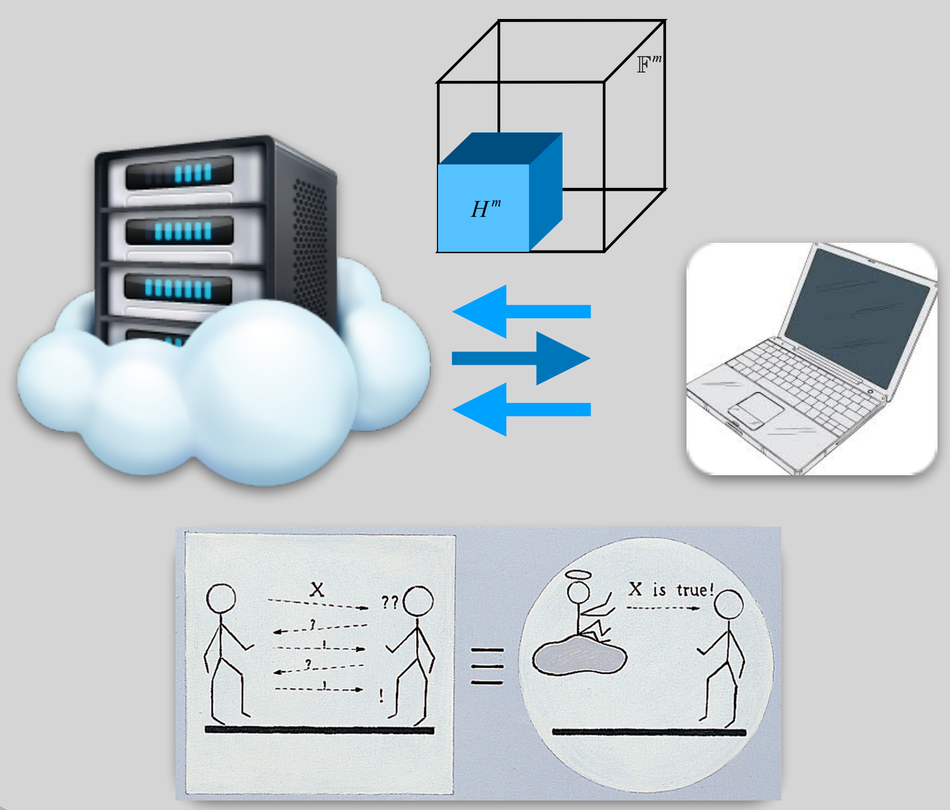
Theorem: NEXP ⊆ ZK-MIP*

Proof in 2 steps:

Lifting lemma



Algebraic ZK



Interactive PCP

$\Downarrow$

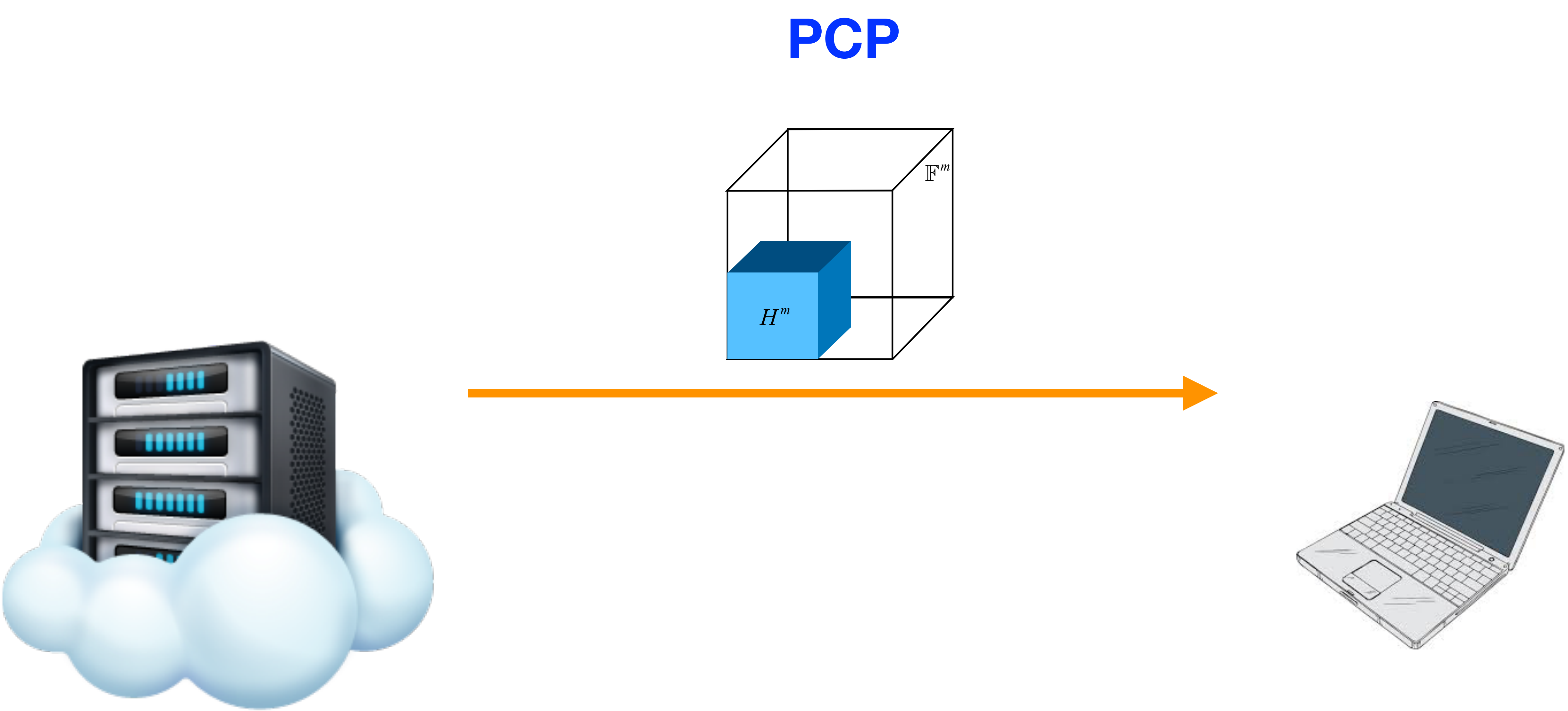
MIP*

From Classical to Quantum

Lifting Lemma: **Any PCP** ➡ **MIP* with similar parameters**

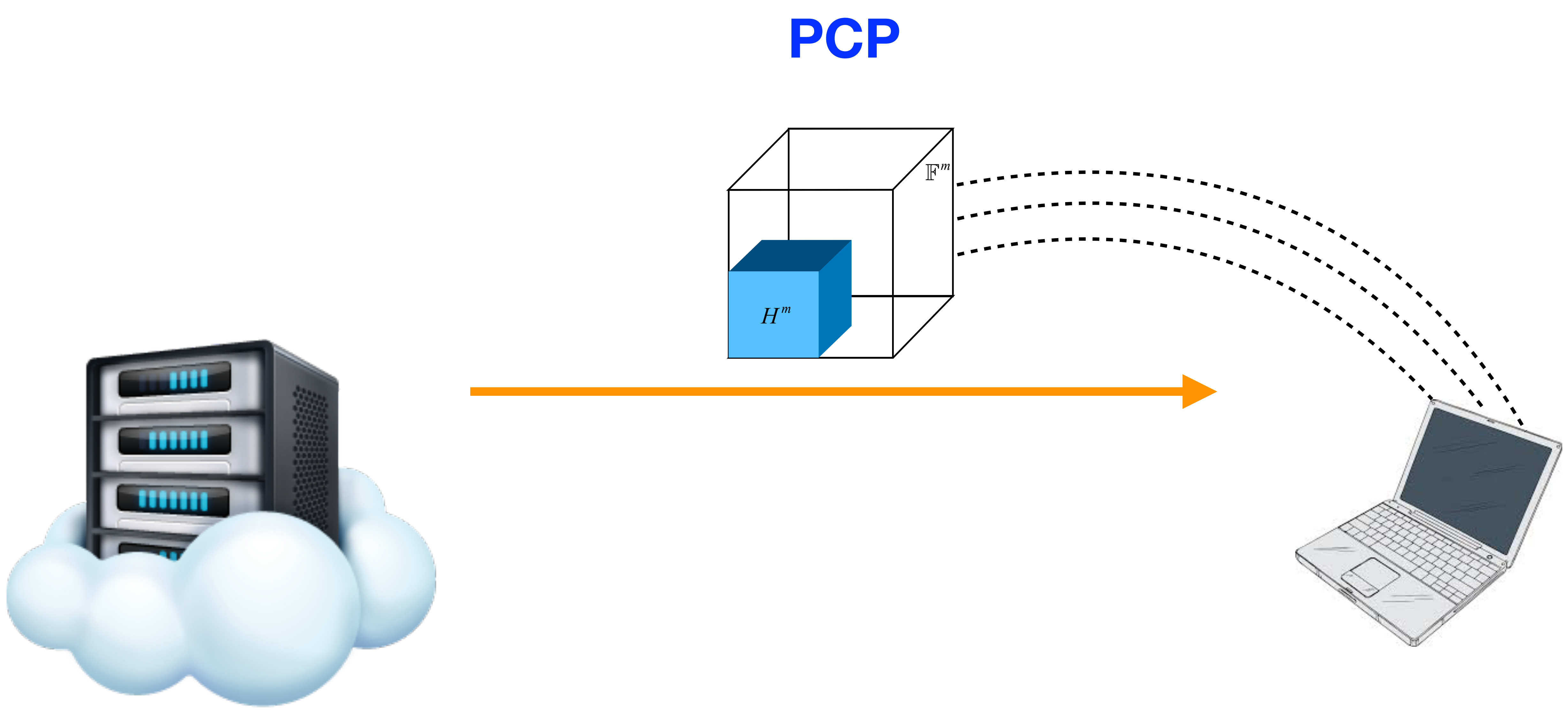
From Classical to Quantum

Lifting Lemma: **Any PCP** ➡ **MIP* with similar parameters**



From Classical to Quantum

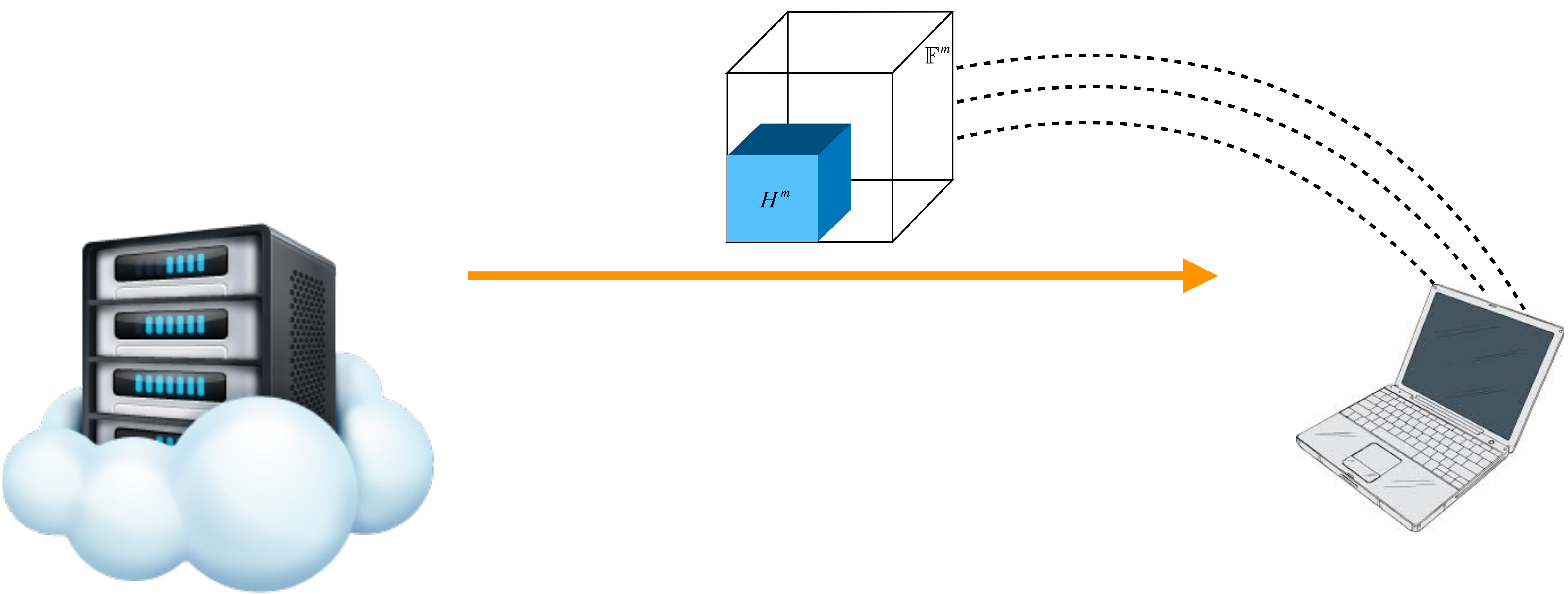
Lifting Lemma: **Any PCP** ➧ **MIP* with similar parameters**



From Classical to Quantum

Lifting Lemma: **Any PCP ➡ MIP* with similar parameters**

PCP



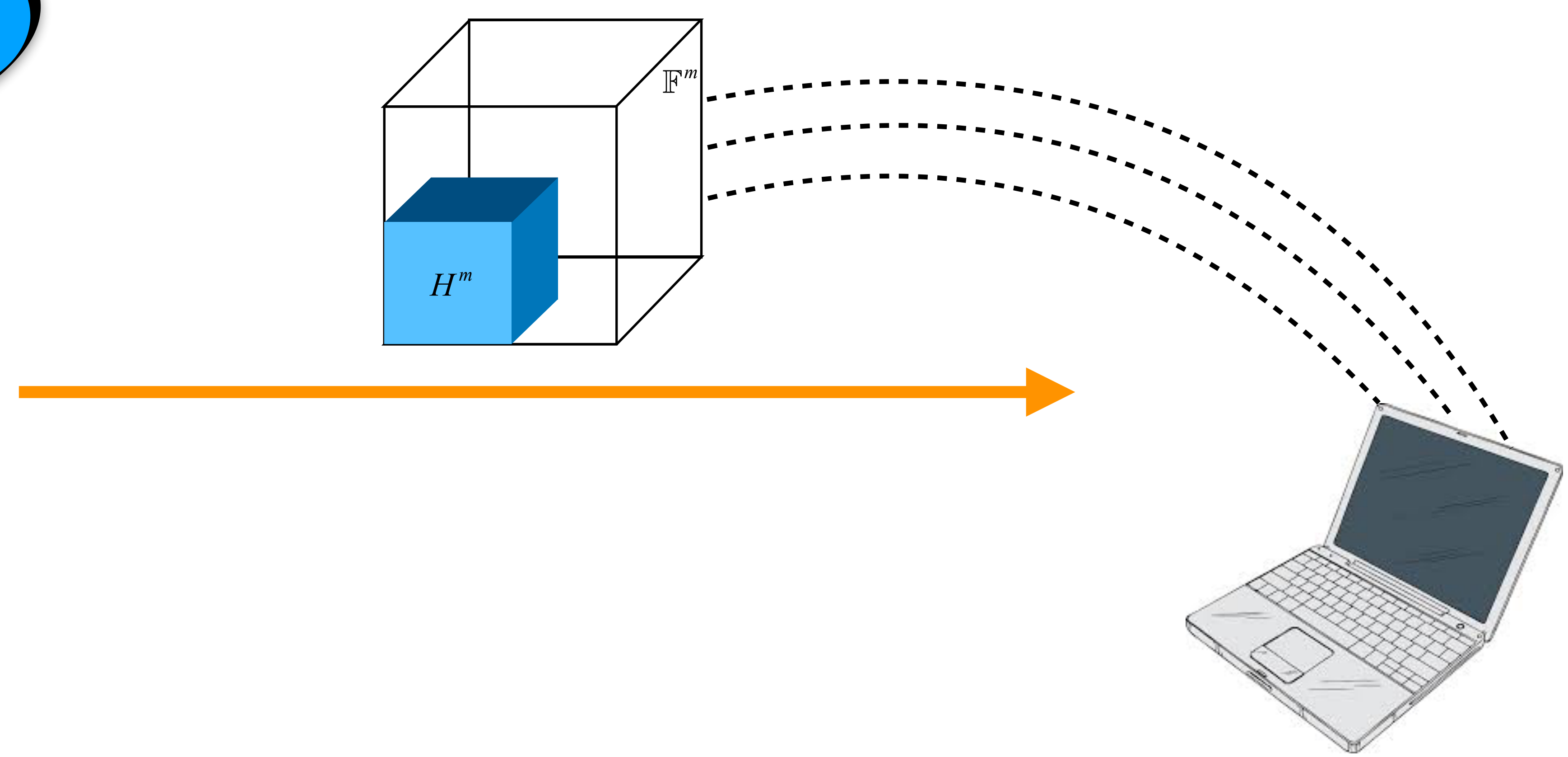
All machines are CLASSICAL

From Classical to Quantum

Lifting Lemma: **Any PCP** ➡ **MIP* with similar parameters**

Abstraction of IV12's NEXP ⊆ MIP*

PCP



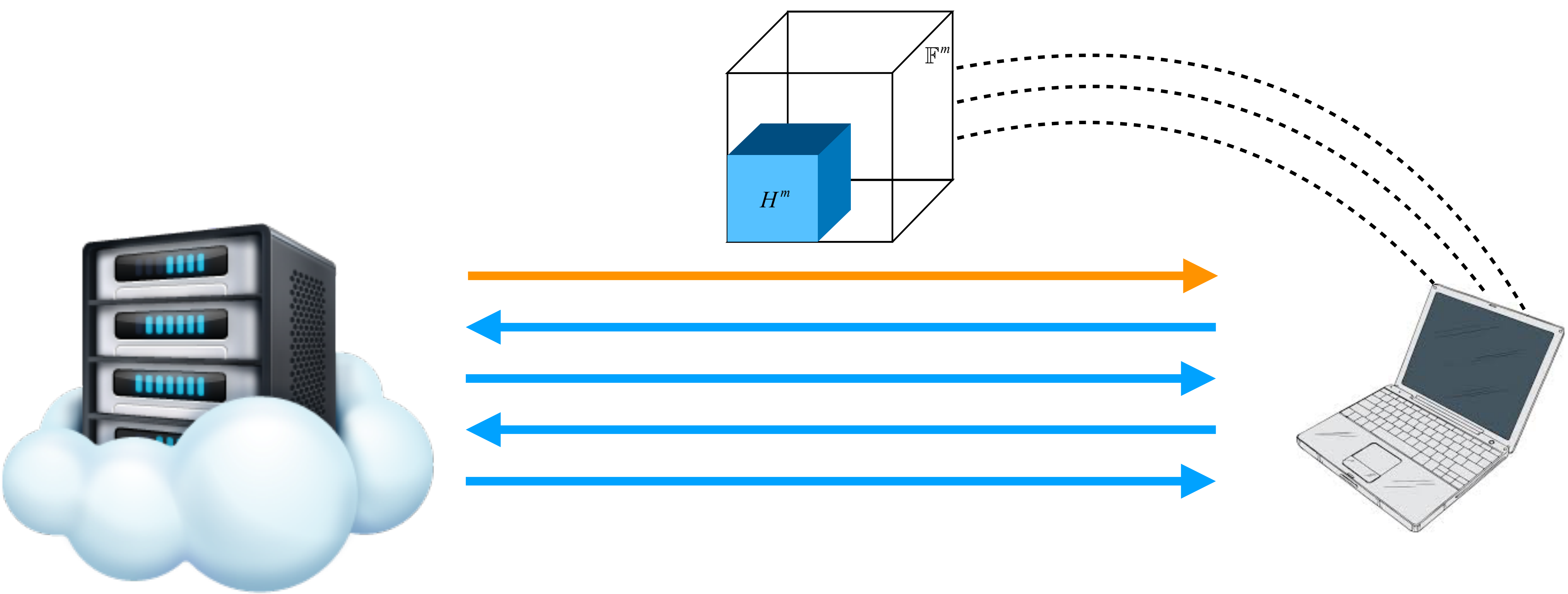
All machines are CLASSICAL

From Classical to Quantum

Lifting Lemma: **Any interactive PCP** ➡ **MIP* with similar parameters**

Interactive PCP

[Kalai-Raz 08]



All machines are CLASSICAL

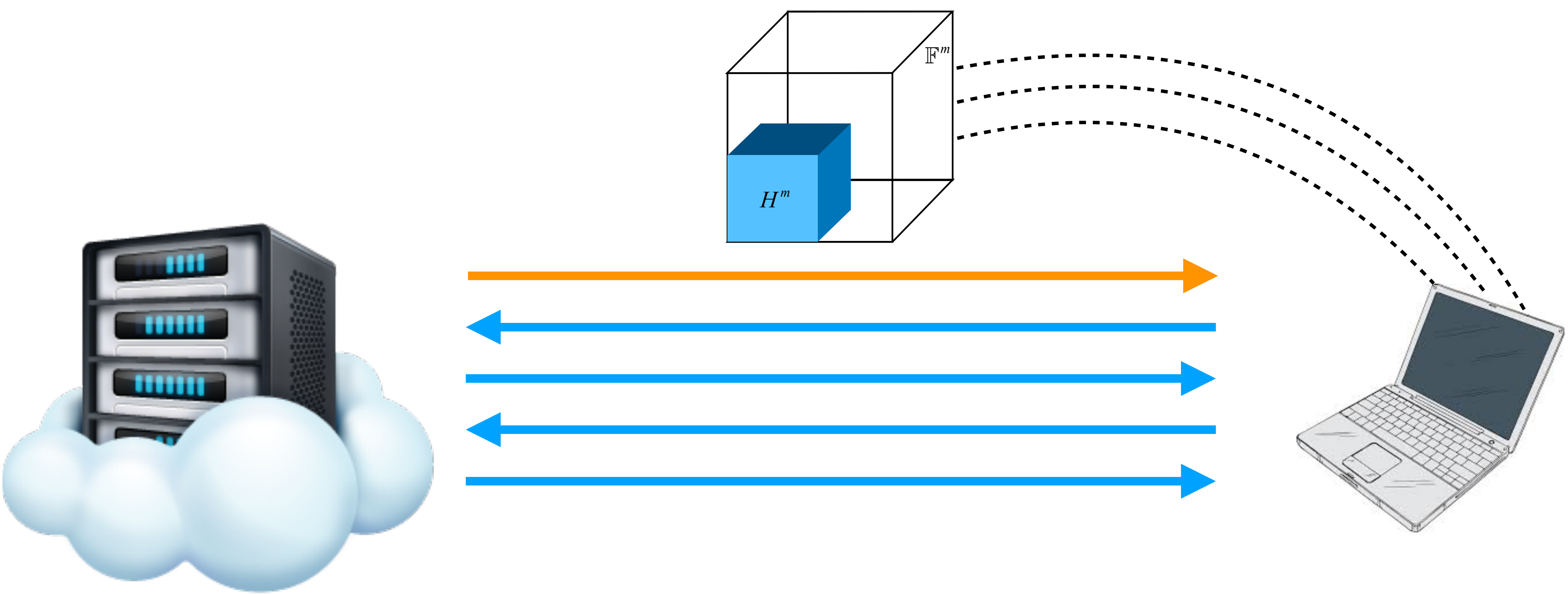
From Classical to Quantum

Lifting Lemma: Any "low-degree" interactive PCP ➡ MIP* with similar parameters

PRESERVING ZK

Low-degree Interactive PCP

[Kalai-Raz 08]



All machines are CLASSICAL

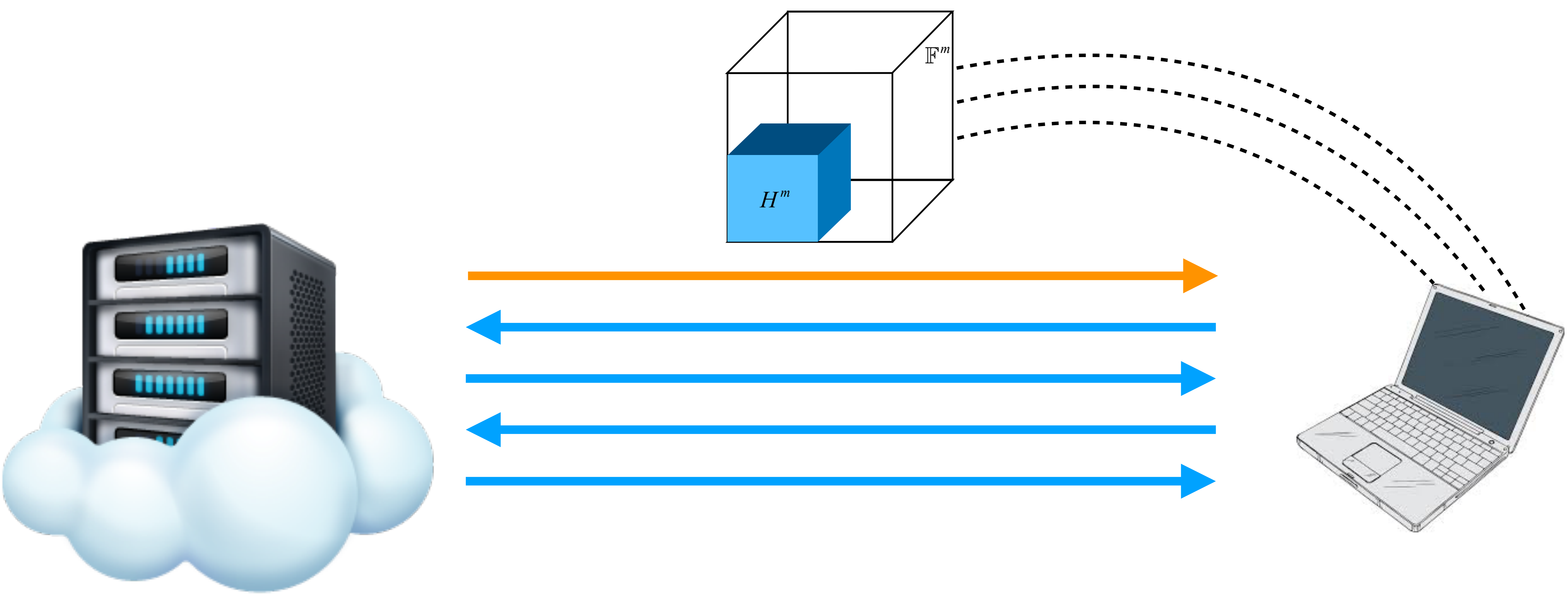
From Classical to Quantum

Lifting Lemma: Any "low-degree" interactive PCP ➡ MIP* with similar parameters

PRESERVING ZK

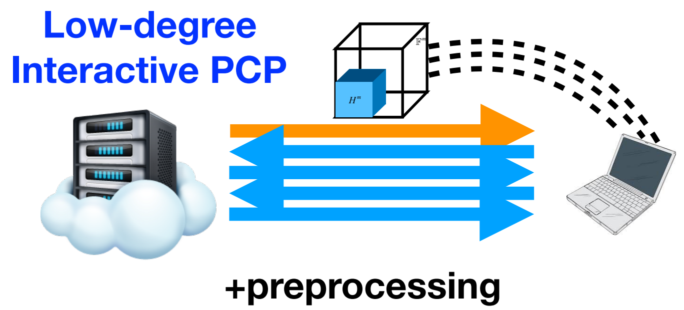
Low-degree Interactive PCP

[Kalai-Raz 08]



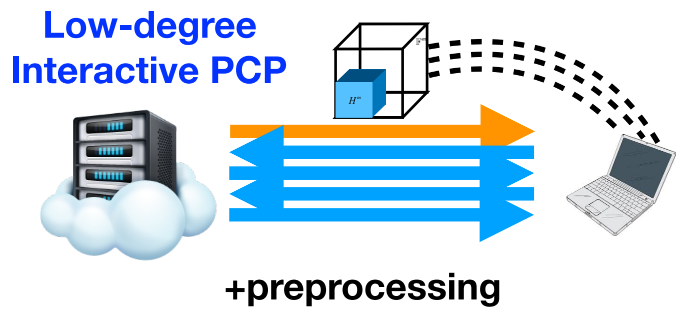
All machines are CLASSICAL

Overview of the Lifting Lemma



Overview of the Lifting Lemma

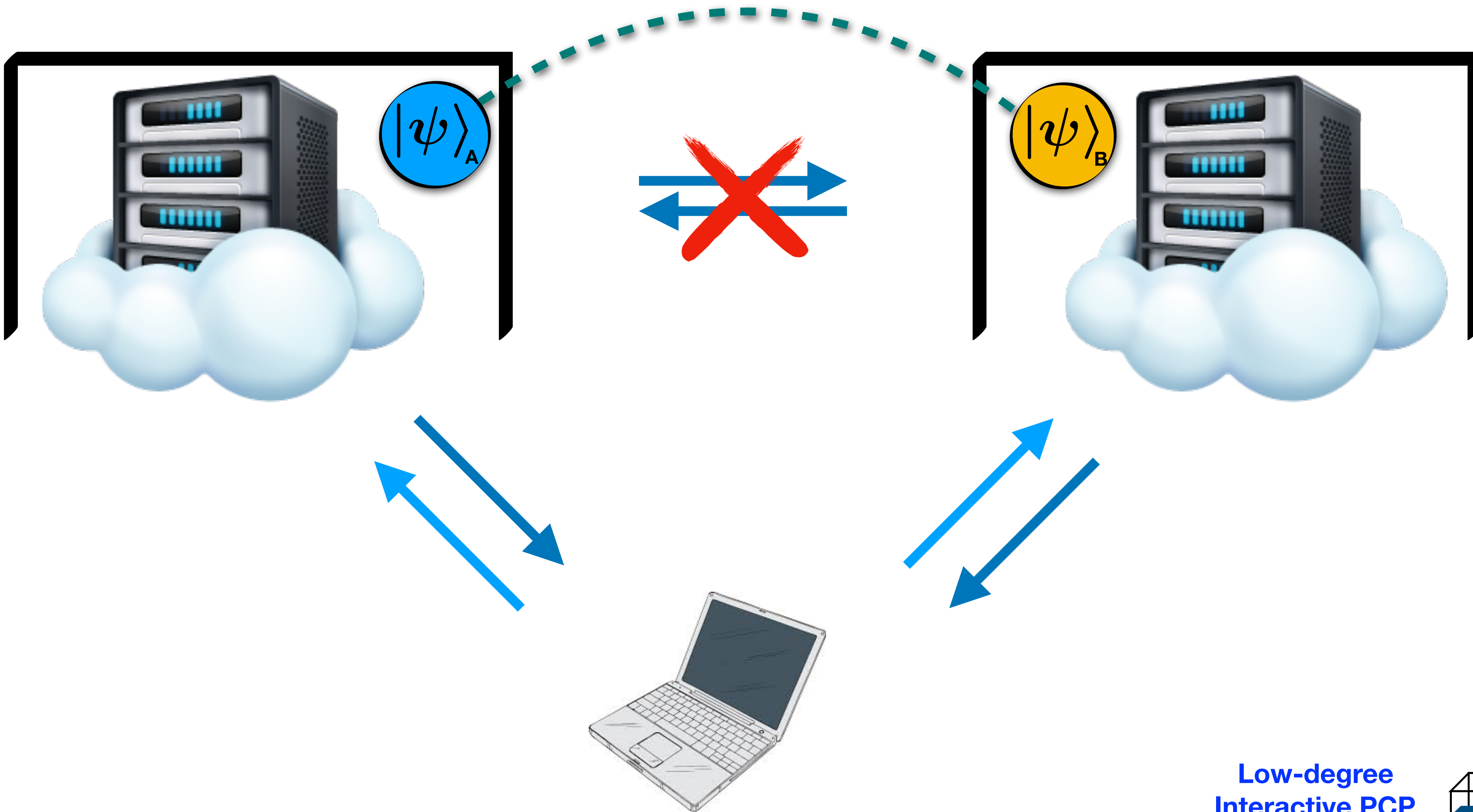
w.p. 1/2: **MIP* point-vs-plane Low-degree test**
[Natarajan-Vidick 18]

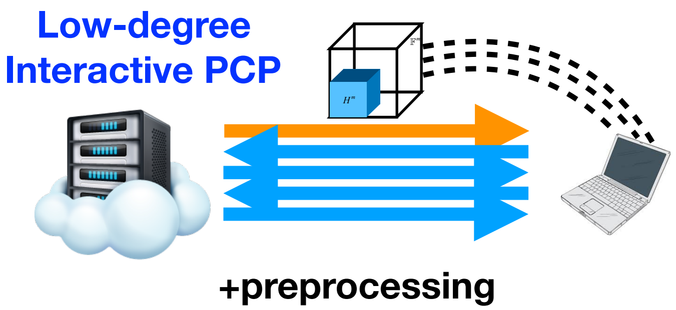


Overview of the Lifting Lemma

w.p. 1/2: **MIP* point-vs-plane**
Low-degree test
[Natarajan-Vidick 18]

MIP*

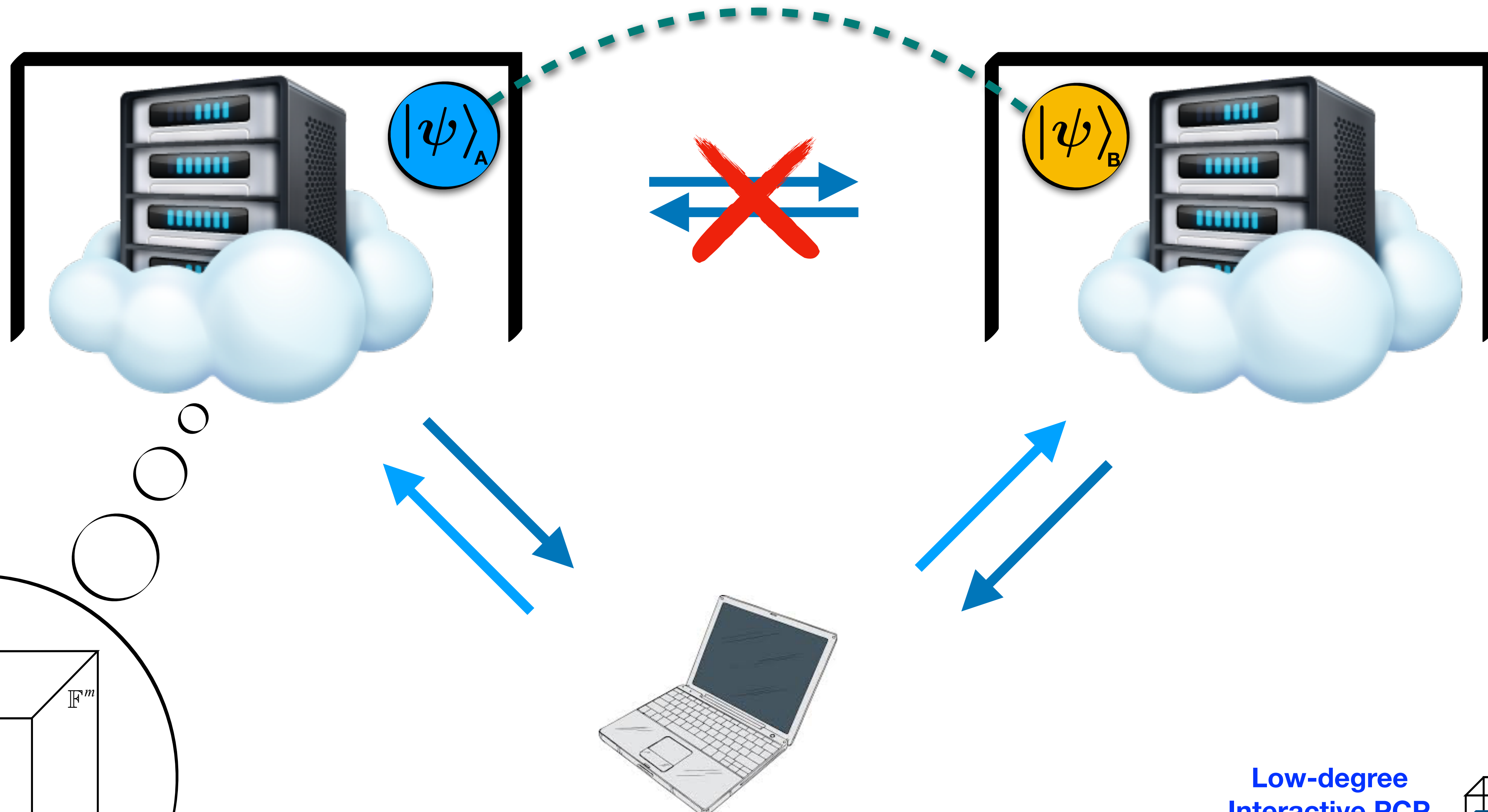


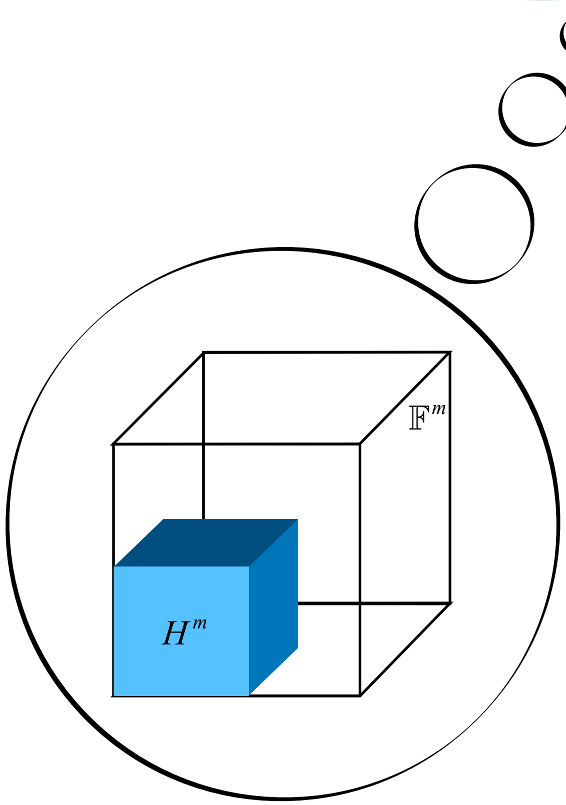


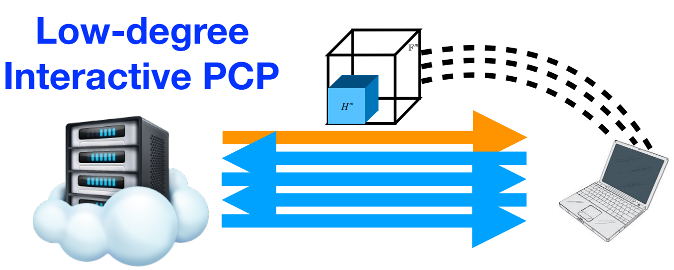
Overview of the Lifting Lemma

w.p. 1/2: **MIP* point-vs-plane Low-degree test**
[Natarajan-Vidick 18]

MIP*



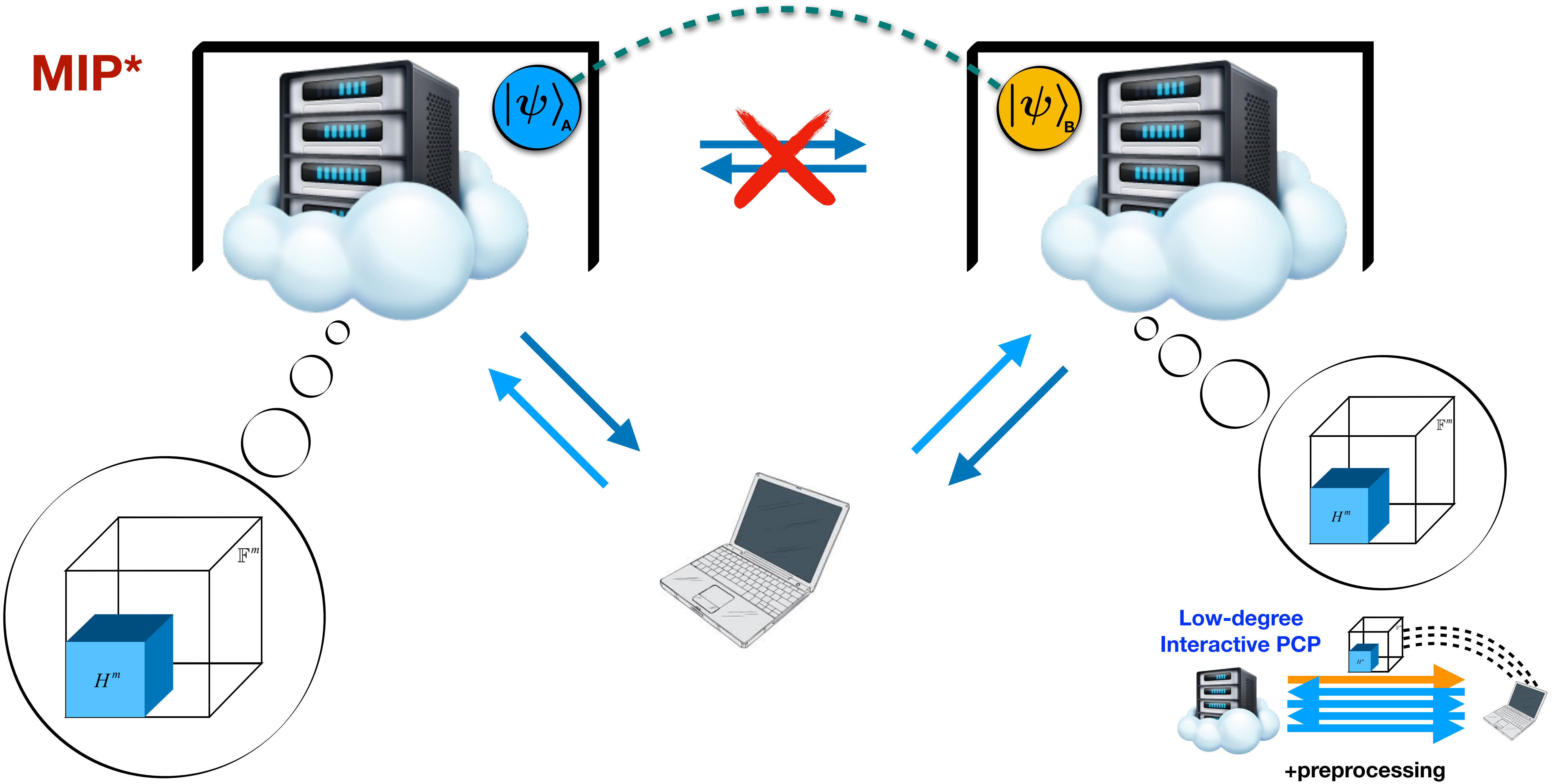




Overview of the Lifting Lemma

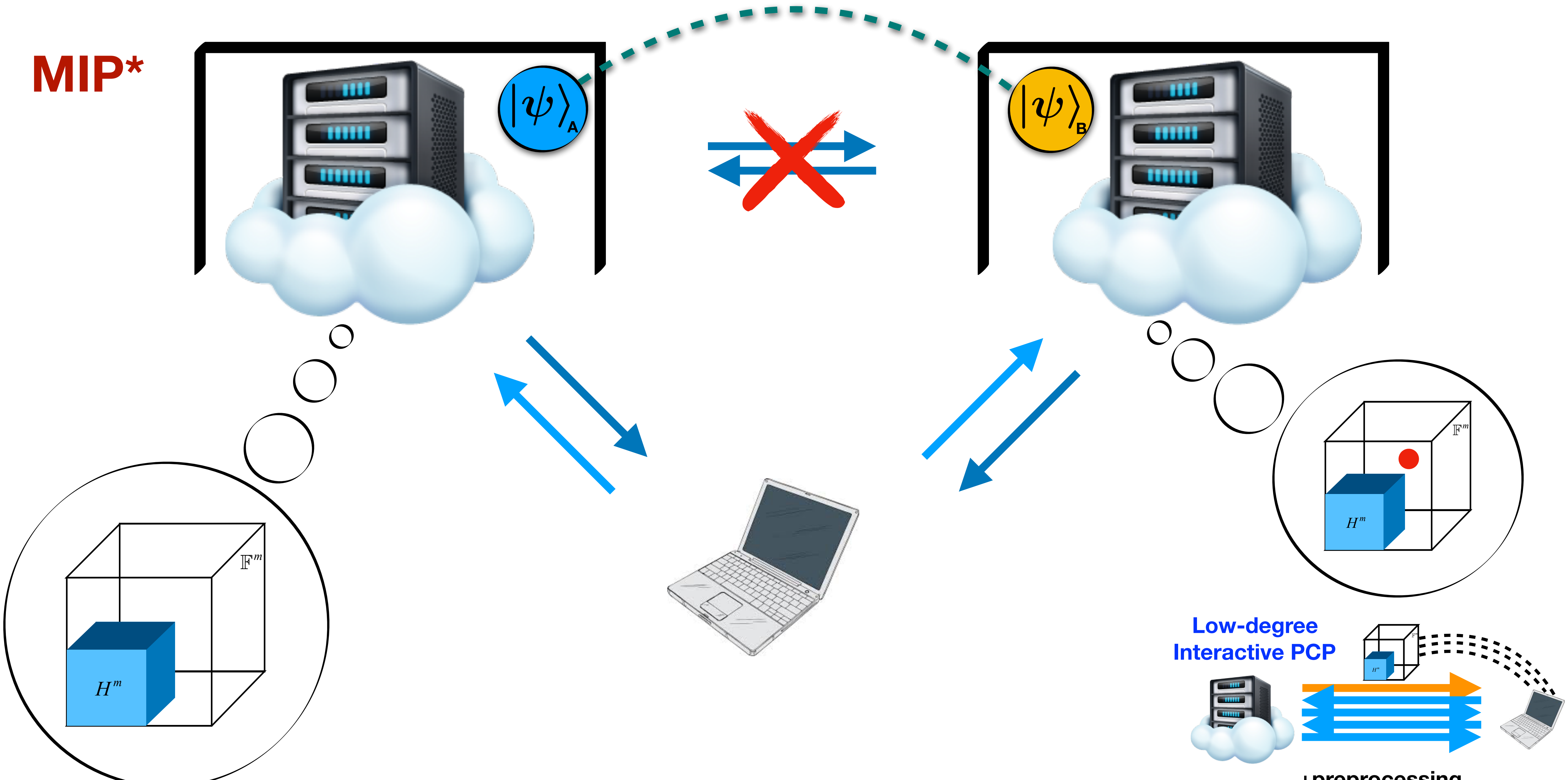
w.p. 1/2: **MIP* point-vs-plane Low-degree test**
[Natarajan-Vidick 18]

MIP*



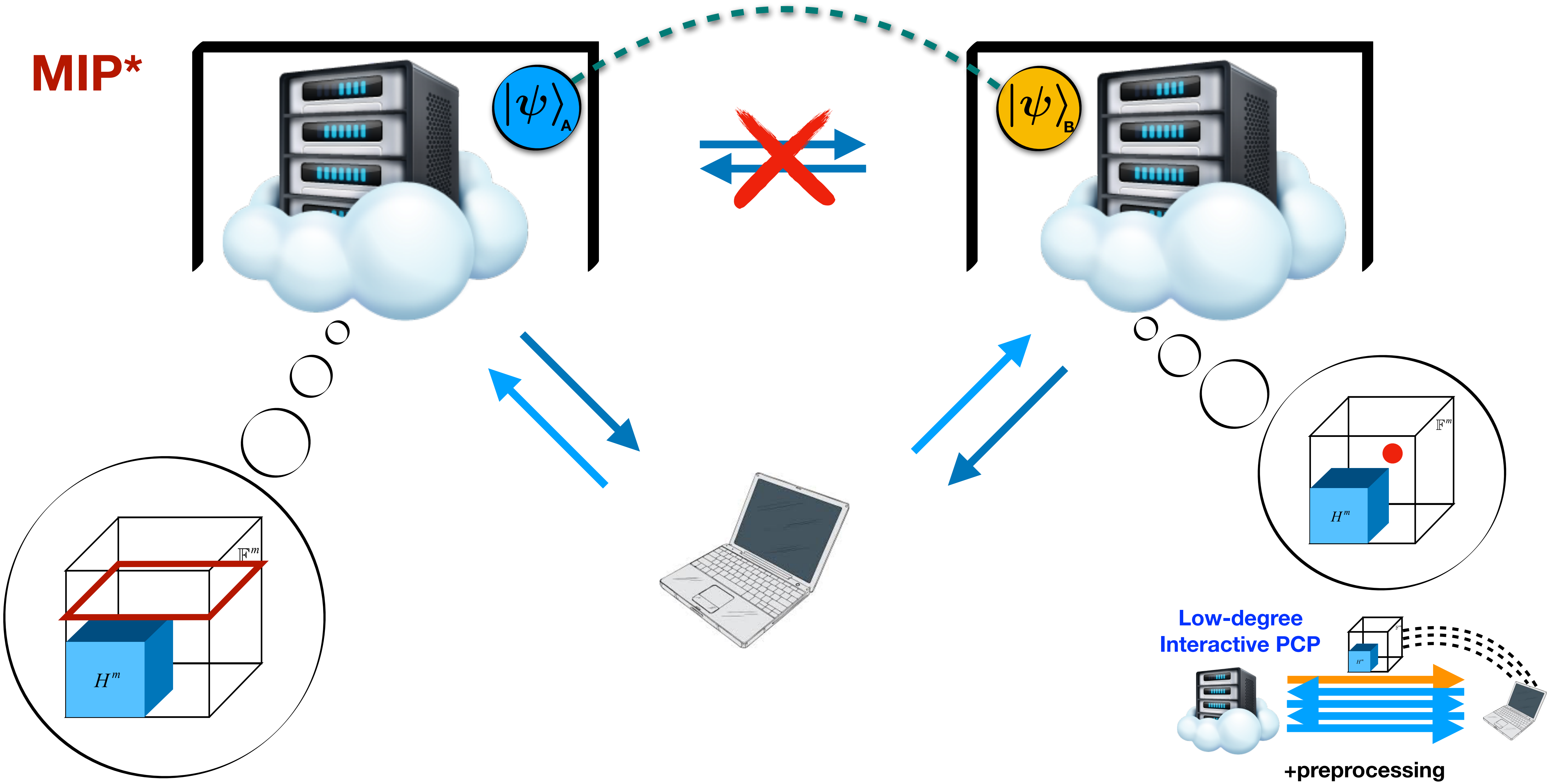
Overview of the Lifting Lemma

w.p. 1/2: **MIP* point-vs-plane Low-degree test**
[Natarajan-Vidick 18]



Overview of the Lifting Lemma

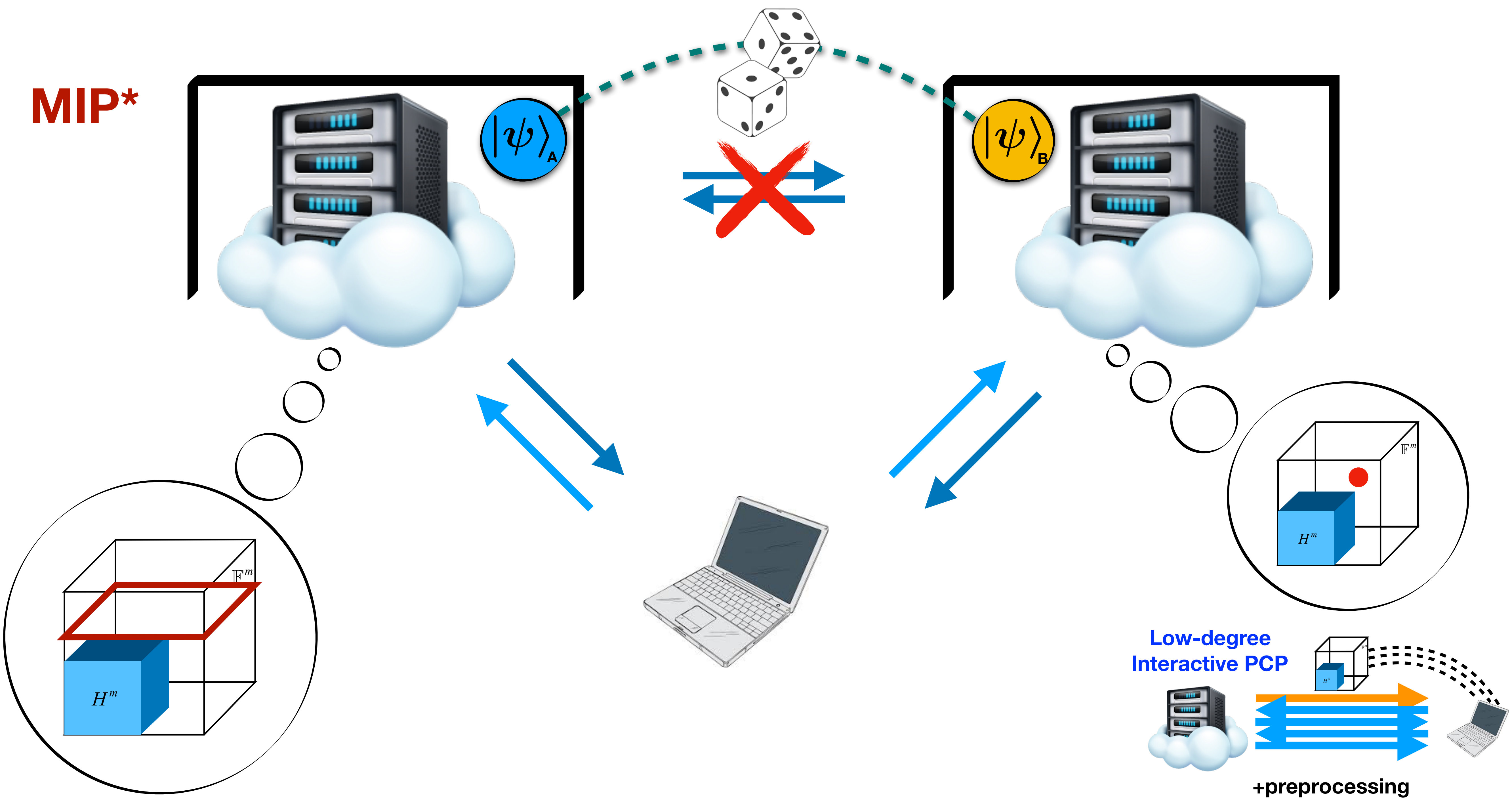
w.p. 1/2: **MIP* point-vs-plane Low-degree test**
[Natarajan-Vidick 18]



Overview of the Lifting Lemma

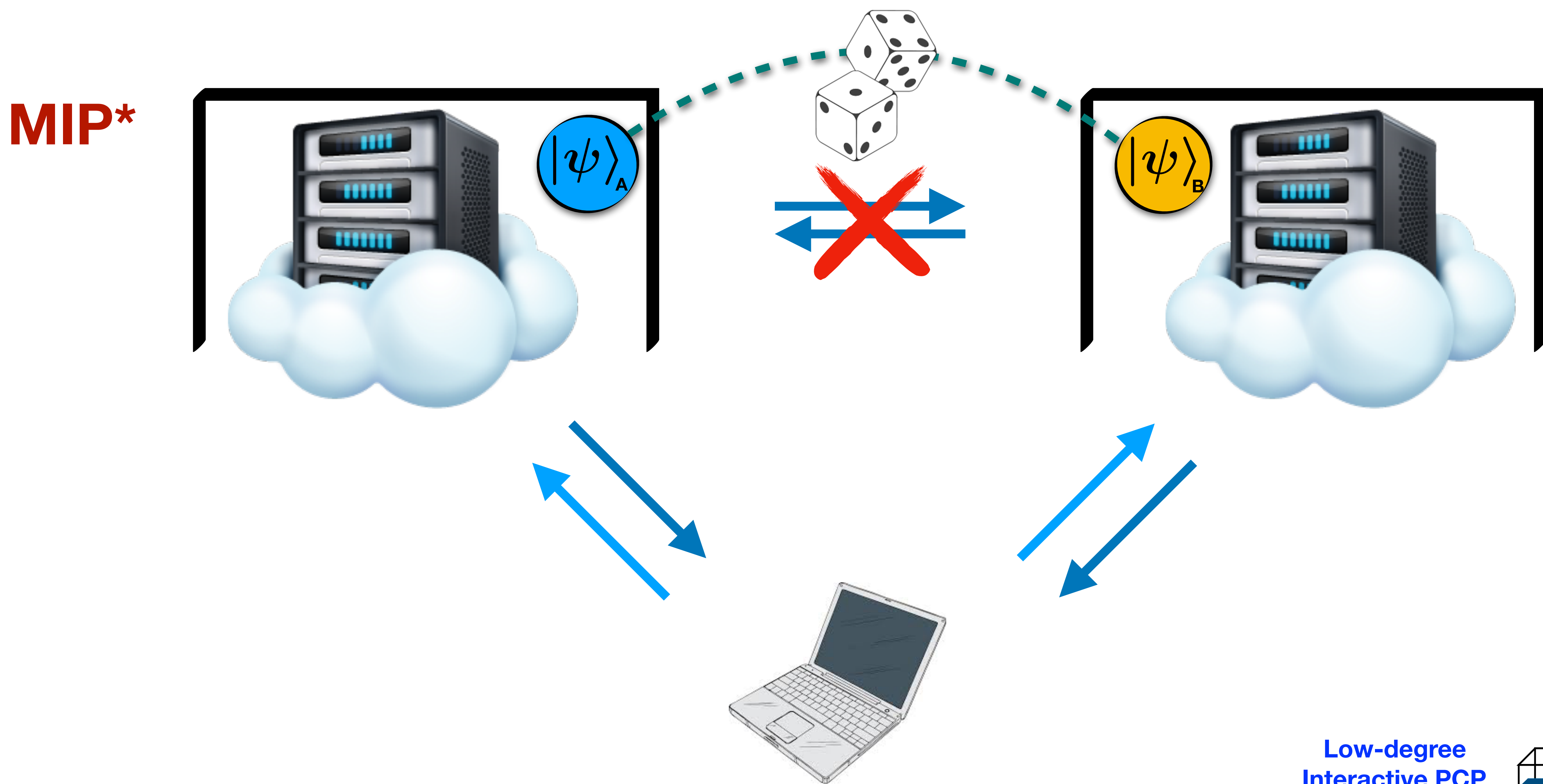
w.p. 1/2: **MIP* point-vs-plane Low-degree test**
[Natarajan-Vidick 18]

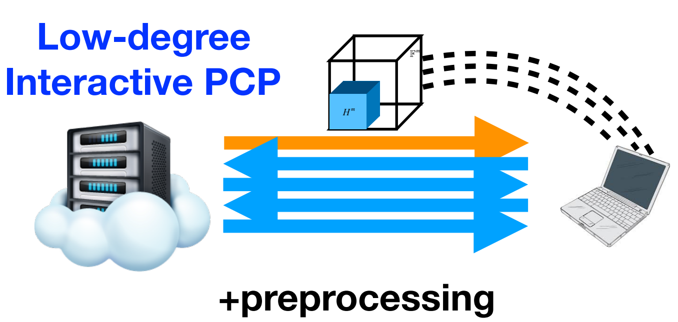
MIP*



Overview of the Lifting Lemma

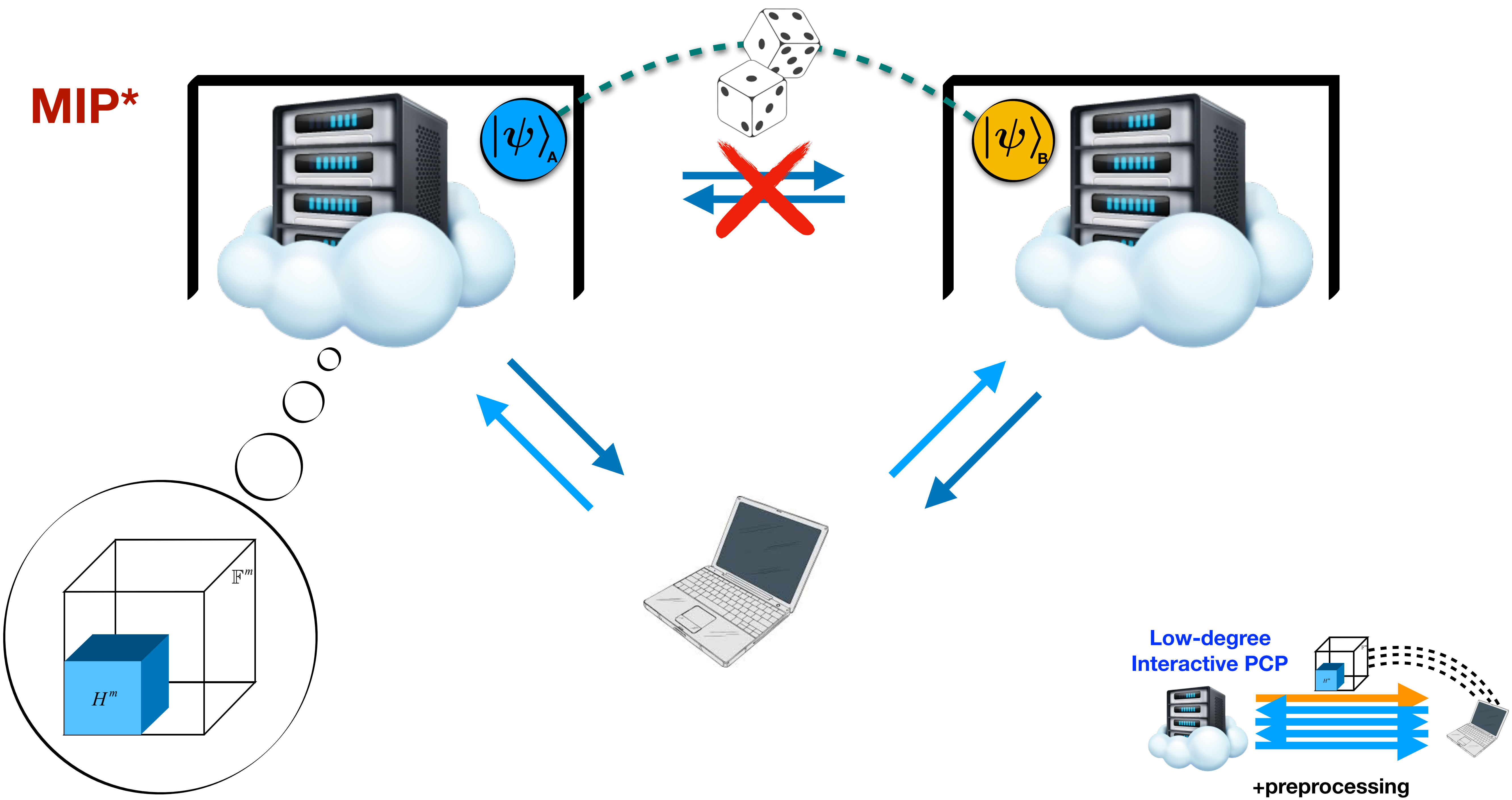
w.p. 1/2: **Interactive PCP emulation**





Overview of the Lifting Lemma

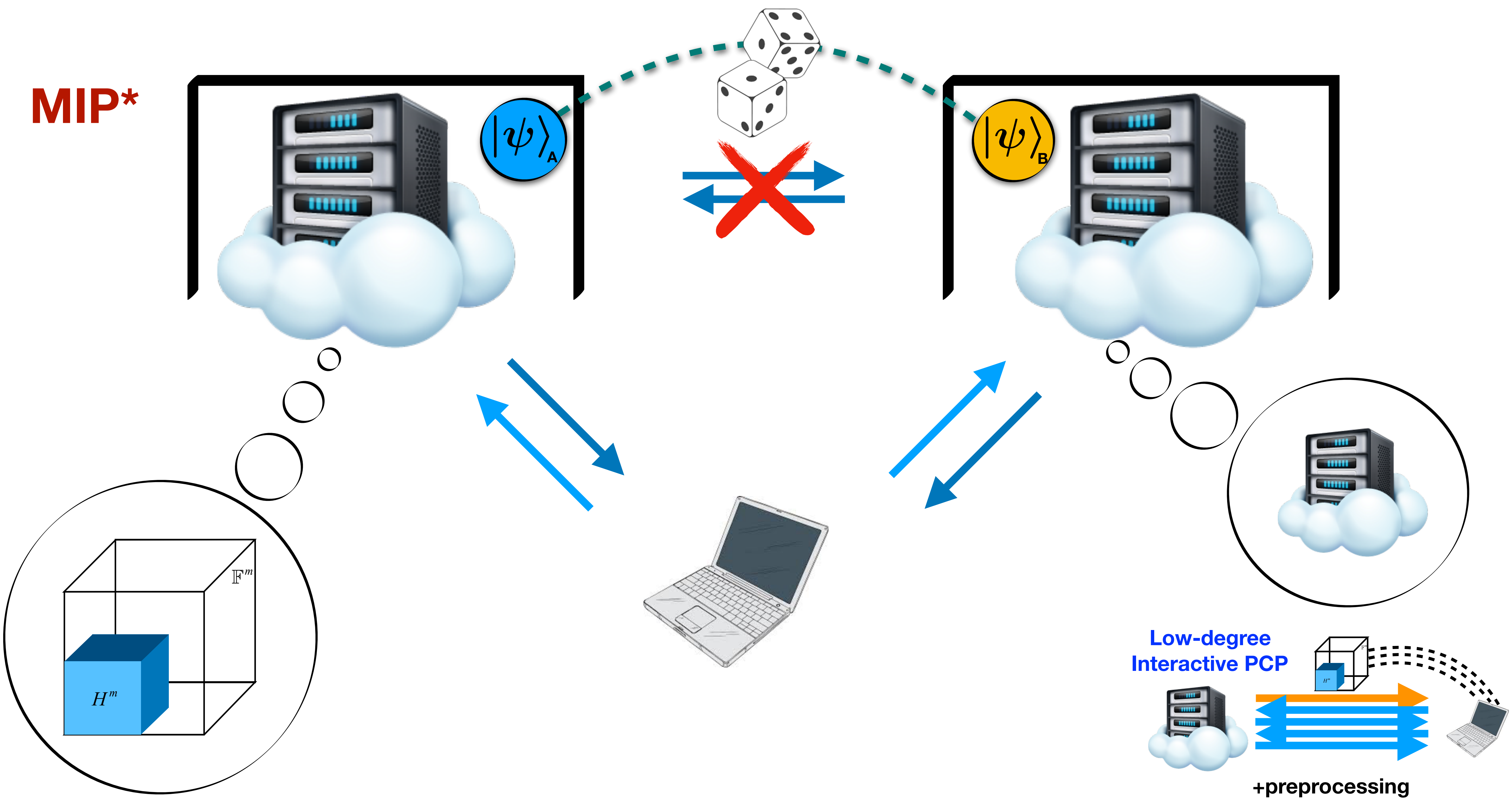
w.p. 1/2: **Interactive PCP emulation**



Overview of the Lifting Lemma

w.p. 1/2: **Interactive PCP emulation**

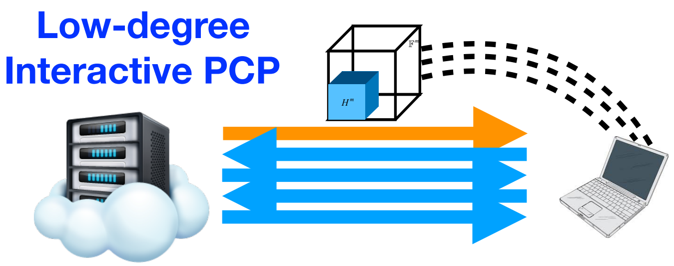
MIP*

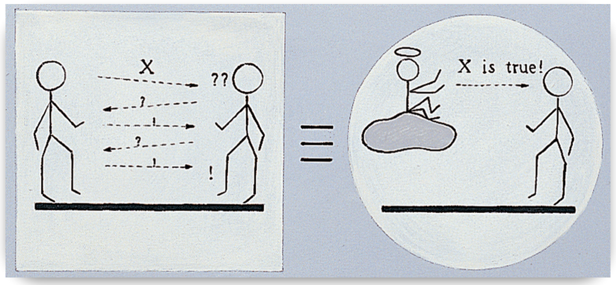


Algebraic Zero Knowledge

Algebraic Zero Knowledge

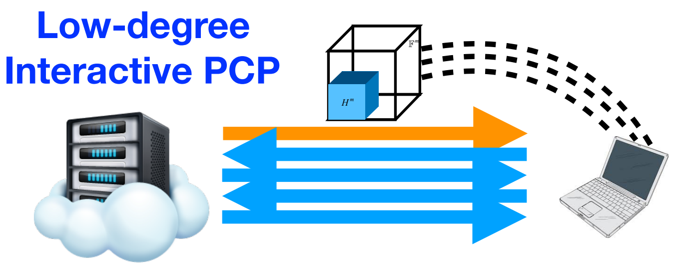
Theorem: There exists a ZERO KNOWLEDGE low-degree interactive PCP for NEXP

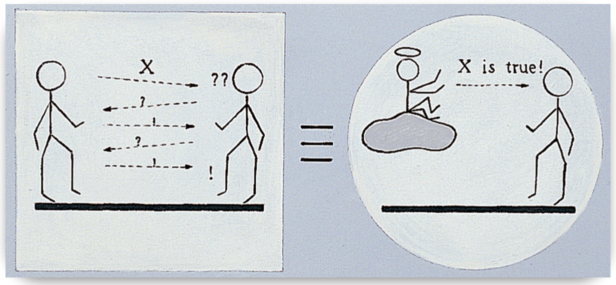




Algebraic Zero Knowledge

Theorem: There exists a ZERO KNOWLEDGE low-degree interactive PCP for NEXP

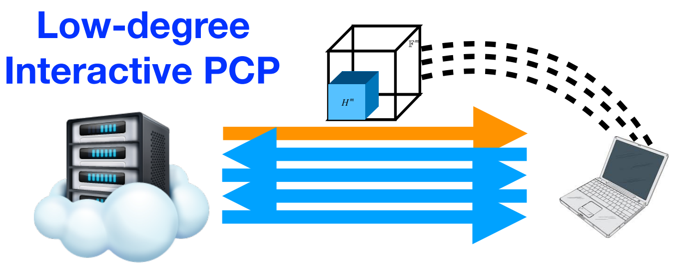


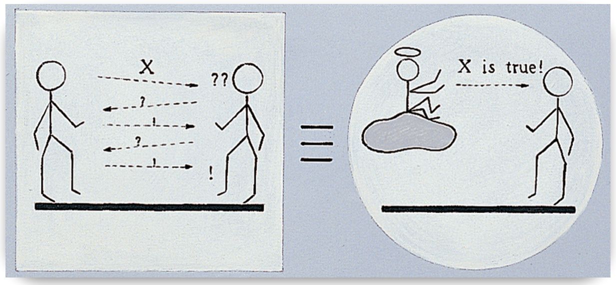


Previous ZK techniques are **Incompatible** with algebraic lifting

Algebraic Zero Knowledge

Theorem: There exists a ZERO KNOWLEDGE low-degree interactive PCP for NEXP





Strong ZK sumcheck

Algebraic Commitment scheme

Weak ZK sumcheck [BCFGRS17]

Previous ZK techniques are **Incompatible** with algebraic lifting

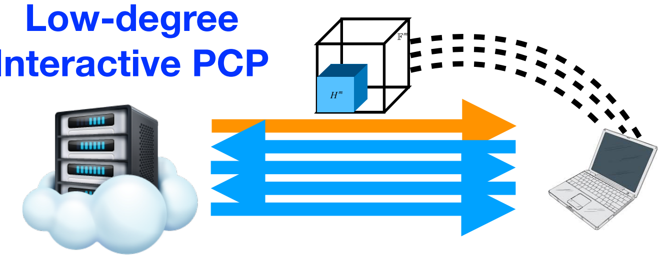
Structural results on Reed-Muller subcube sums

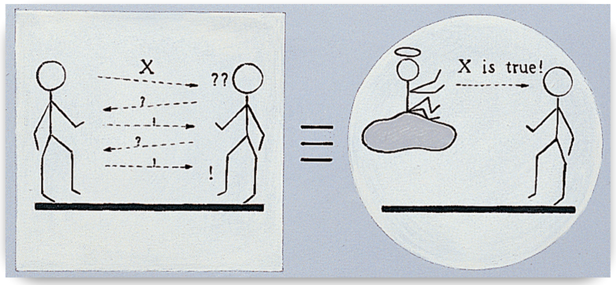
Succinct constraint detection for Reed-Muller [BCFGRS17]

Derandomized PIT for sums of products of Reed-Solomon [RS05]

Algebraic Zero Knowledge

Theorem: There exists a ZERO KNOWLEDGE low-degree interactive PCP for NEXP





Strong ZK sumcheck

Algebraic Commitment scheme

Weak ZK sumcheck [BCFGRS17]

Previous ZK techniques are **Incompatible** with algebraic lifting

Structural results on Reed-Muller subcube sums

Succinct constraint detection for Reed-Muller [BCFGRS17]

Derandomized PIT for sums of products of Reed-Solomon [RS05]

Algebraic Commitment Scheme

First some high-level motivation

Algebraic Commitment Scheme

First some high-level motivation

IPCP model

Algebraic Commitment Scheme

First some high-level motivation

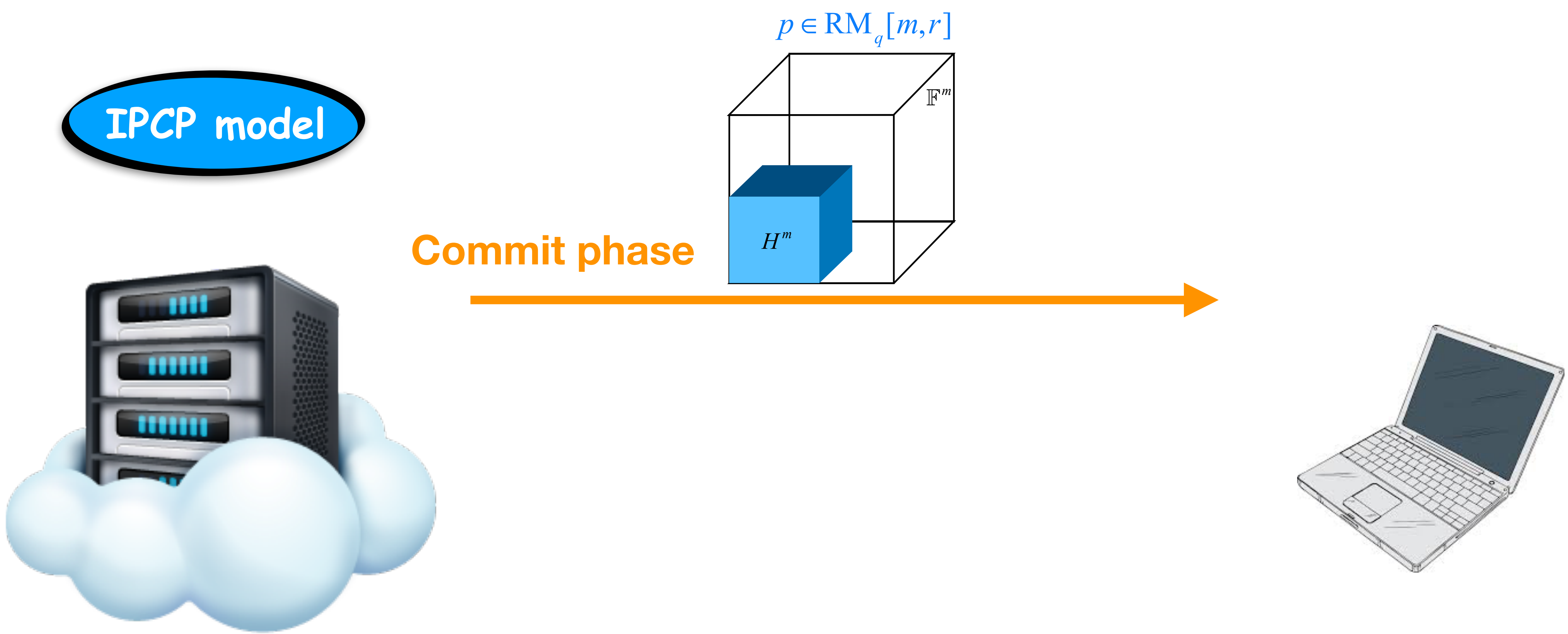
Goal: commit to a message $\beta \in \mathbb{F}$

perfectly **HIDING** the message

in a statistically **BINDING** way

Algebraic Commitment Scheme

First some high-level motivation

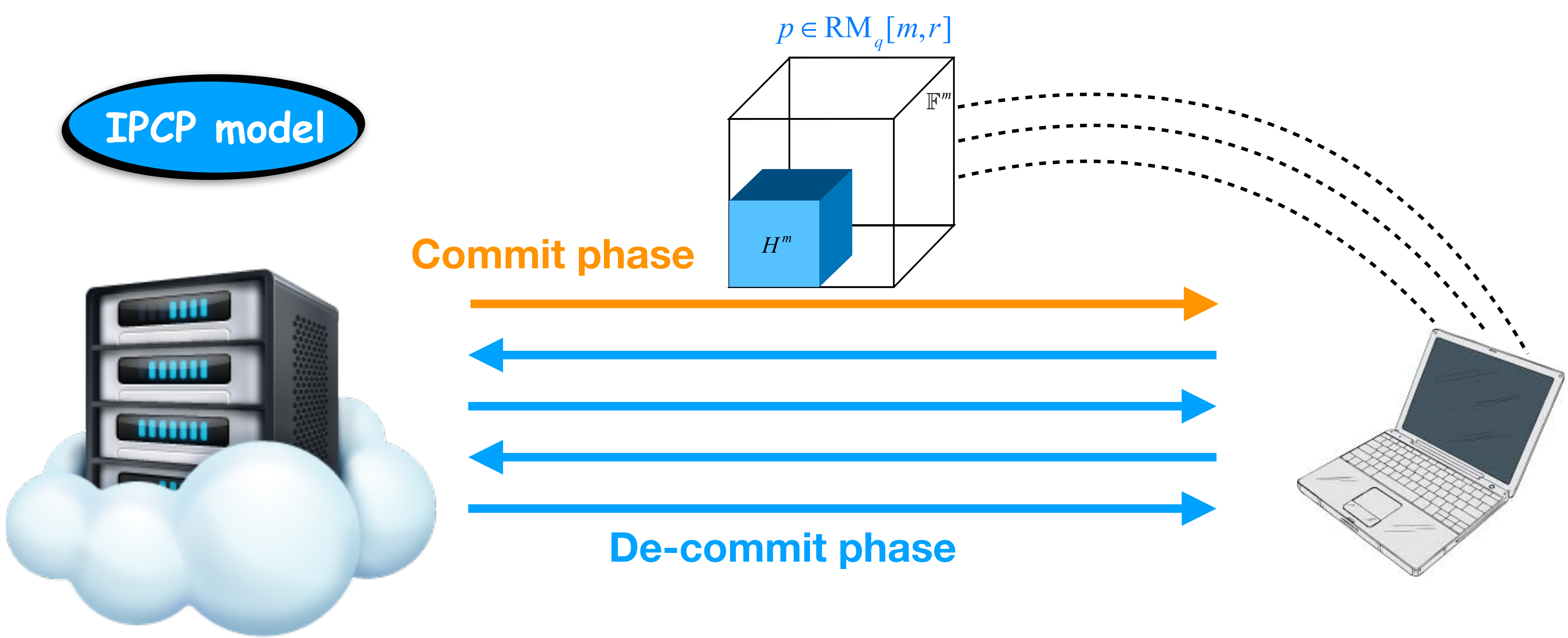


Goal: commit to a message $\beta \in \mathbb{F}$
perfectly **HIDING** the message
in a statistically **BINDING** way

How: send a random polynomial
p s.t. $\sum_{\alpha \in H^m} p(\alpha) = \beta$

Algebraic Commitment Scheme

First some high-level motivation



Goal: commit to a message $\beta \in \mathbb{F}$
perfectly **HIDING** the message
in a statistically **BINDING** way

How: send a random polynomial
p s.t. $\sum_{\alpha \in H^m} p(\alpha) = \beta$
de-commit via interaction

Warmup: Subcube Sums of Reed-Muller

$$\mathrm{RM}_q[m,r] = \{\langle p(\alpha)\rangle \mid p \in \mathbb{F}_q^{\leq r}[X_1,\ldots,X_m]\}$$

Warmup: Subcube Sums of Reed-Muller

$$\mathrm{RM}_q[m,r] = \{\langle p(\alpha) \rangle \mid p \in \mathbb{F}_q^{\leq r}[X_1,\ldots,X_m]\}$$

Low-degree extension perspective

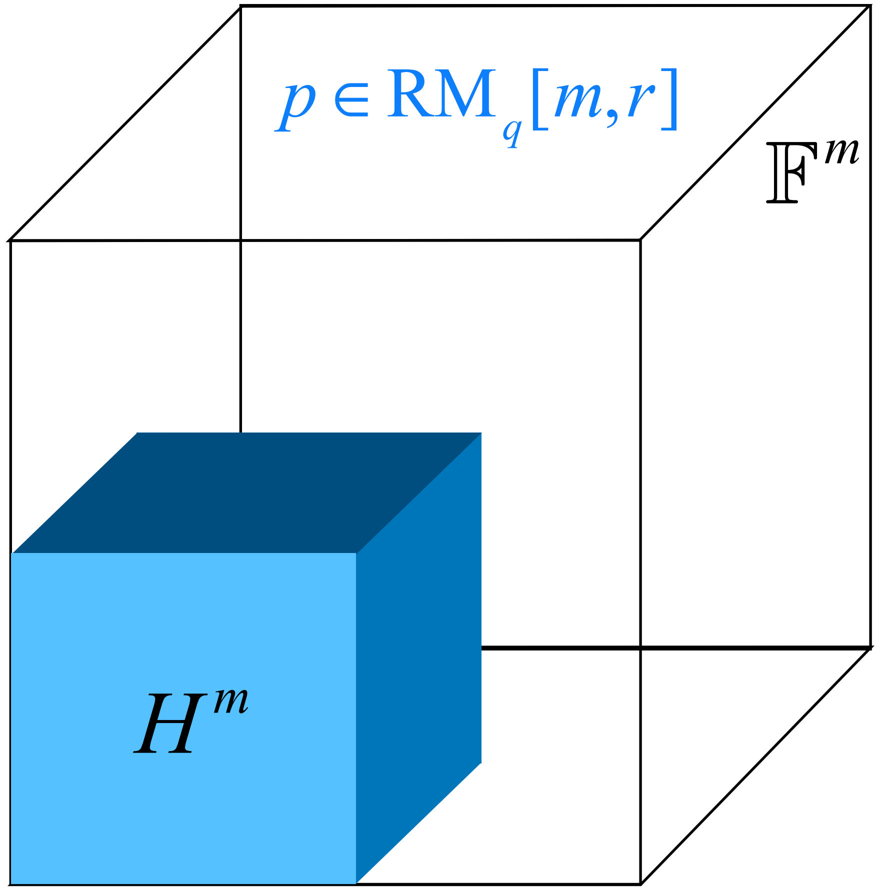
$H \subseteq \mathbb{F}$ $\quad |H| < r$

Warmup: Subcube Sums of Reed-Muller

$$\mathrm{RM}_q[m,r] = \{\langle p(\alpha)\rangle \mid p \in \mathbb{F}_q^{\leq r}[X_1,\ldots,X_m]\}$$

Low-degree extension perspective

$H \subseteq \mathbb{F}$ $\qquad |H| < r$



Warmup: Subcube Sums of Reed-Muller

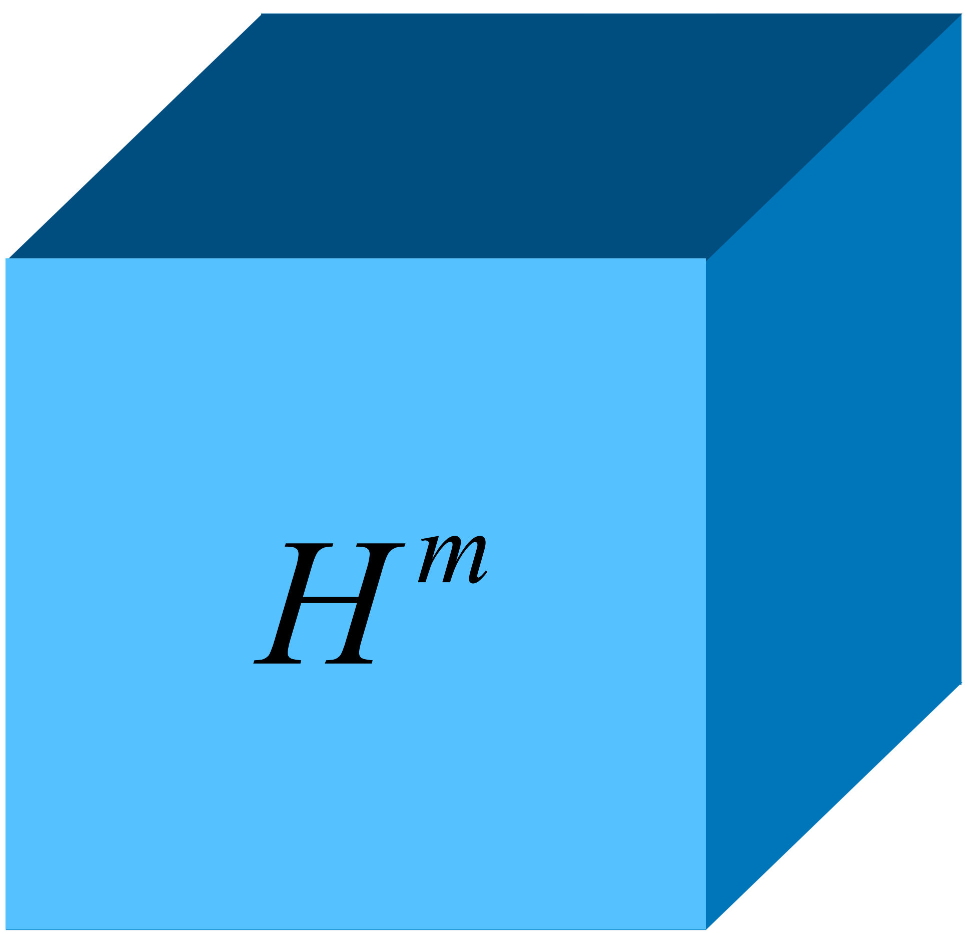
$$\mathrm{RM}_q[m,r] = \{\langle p(\alpha)\rangle \mid p \in \mathbb{F}_q^{\le r}[X_1,\ldots,X_m]\}$$

For $f : H^m \to \mathbb{F}$

$\sum_{\alpha \in H^m} f(a)$ **is #P-hard to compute**

Low-degree extension perspective

$H \subseteq \mathbb{F}$ $\qquad |H| < r$

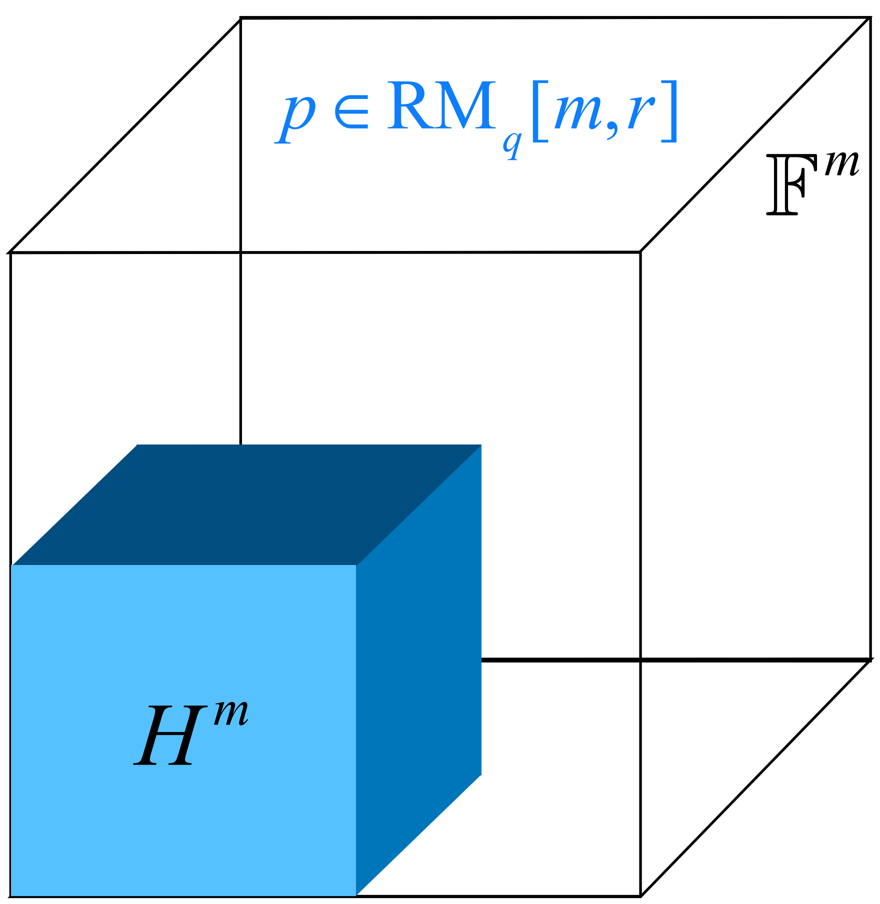


Warmup: Subcube Sums of Reed-Muller

$$\mathrm{RM}_q[m,r] = \{\langle p(\alpha)\rangle \mid p \in \mathbb{F}_q^{\le r}[X_1,\ldots,X_m]\}$$

Low-degree extension perspective

$H \subseteq \mathbb{F}$ $\qquad |H| < r$



For $f : H^m \to \mathbb{F}$

$\sum_{\alpha \in H^m} f(a)$ **is #P-hard to compute**

The problem: a simple case

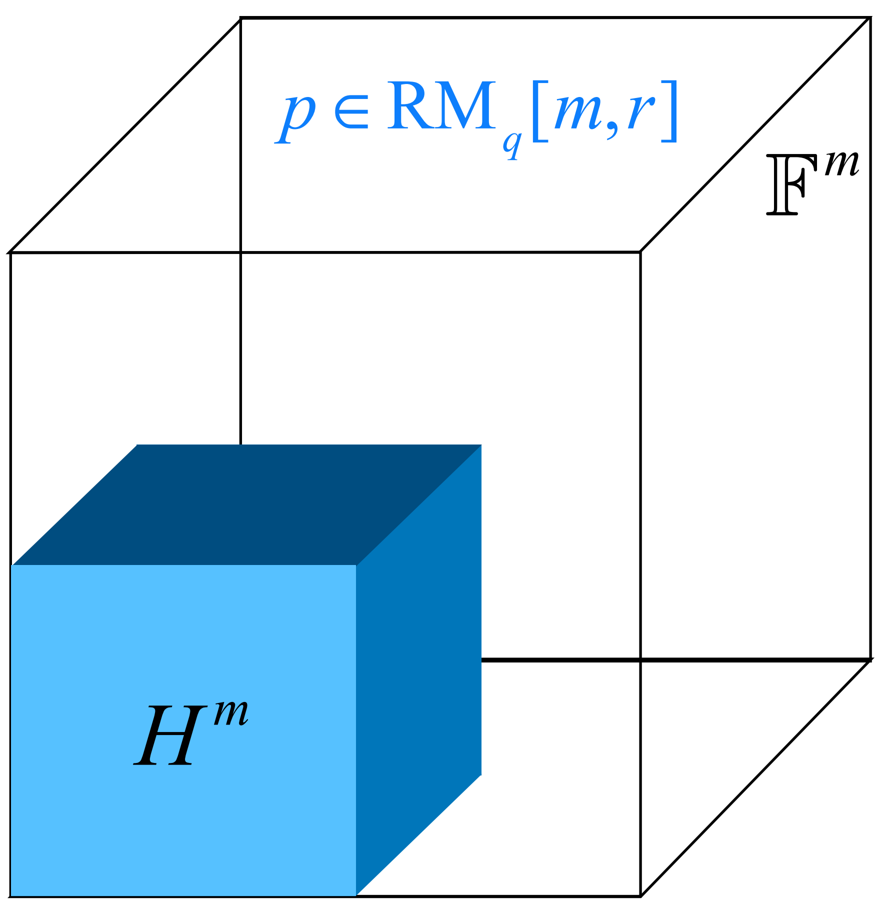
Given $\begin{bmatrix} p \in \mathrm{RM}_q[m,r] \\ p(\alpha) = f(\alpha) \quad \forall \alpha \in H^m \end{bmatrix}$

Warmup: Subcube Sums of Reed-Muller

$$\mathrm{RM}_q[m,r] = \{\langle p(\alpha)\rangle \mid p \in \mathbb{F}_q^{\leq r}[X_1,\ldots,X_m]\}$$

Low-degree extension perspective

$H \subseteq \mathbb{F}$ $\quad |H| < r$



For $f : H^m \to \mathbb{F}$

$\sum_{\alpha \in H^m} f(a)$ **is #P-hard to compute**

The problem: a simple case

Given $\begin{cases} p \in \mathrm{RM}_q[m,r] \\ p(\alpha) = f(\alpha) \quad \forall \alpha \in H^m \end{cases}$

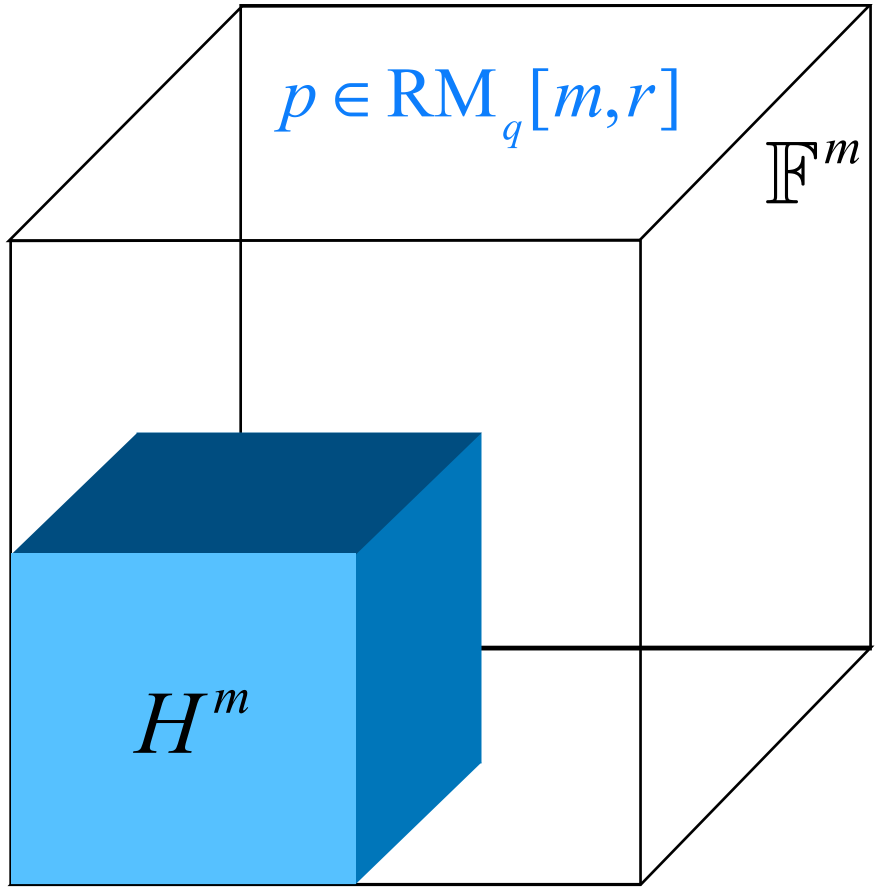
Is $\sum_{\alpha \in H^m} p(\alpha)$ **still hard to compute?**

Warmup: Subcube Sums of Reed-Muller

$$\mathrm{RM}_q[m,r] = \{\langle p(\alpha)\rangle \mid p \in \mathbb{F}_q^{\le r}[X_1,\ldots,X_m]\}$$

Low-degree extension perspective

$H \subseteq \mathbb{F}$ $\qquad |H| < r$



For $f : H^m \to \mathbb{F}$

$\sum_{\alpha \in H^m} f(a)$ **is #P-hard to compute**

The problem: a simple case

Given $\begin{cases} p \in \mathrm{RM}_q[m,r] \\ p(\alpha) = f(\alpha) \quad \forall \alpha \in H^m \end{cases}$

Is $\sum_{\alpha \in H^m} p(\alpha)$ **still hard to compute?**

Algebrization framework

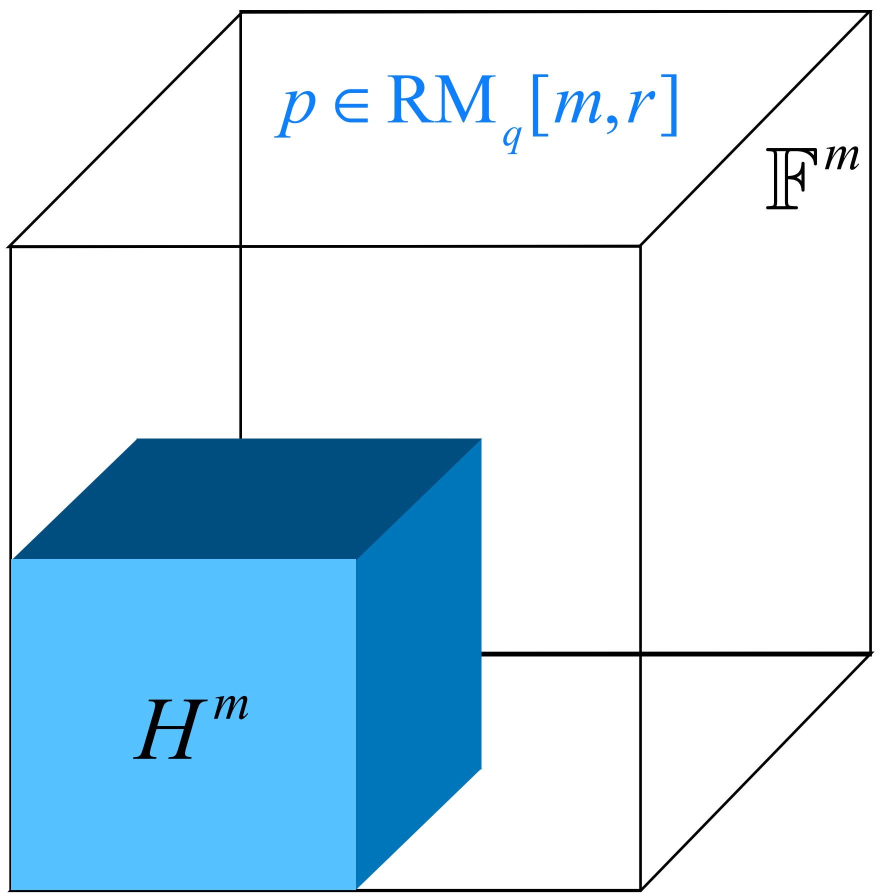
[Aaronson-Wigderson 09]

Warmup: Subcube Sums of Reed-Muller

$$\mathrm{RM}_q[m,r] = \{\langle p(\alpha)\rangle \mid p \in \mathbb{F}_q^{\leq r}[X_1,\ldots,X_m]\}$$

Low-degree extension perspective

$H \subseteq \mathbb{F}$ $\quad |H| < r$



For $f : H^m \to \mathbb{F}$

$\sum_{\alpha \in H^m} f(a)$ **is #P-hard to compute**

The problem: a simple case

Given $\begin{cases} p \in \mathrm{RM}_q[m,r] \\ p(\alpha) = f(\alpha) \quad \forall \alpha \in H^m \end{cases}$

Is $\sum_{\alpha \in H^m} p(\alpha)$ **still hard to compute?**

For r=1, H={0,1}
(multilinear extension)

$$p(2^{-1},\ldots,2^{-1}) = 2^{-k} \sum_{\alpha \in H^m} p(a)$$

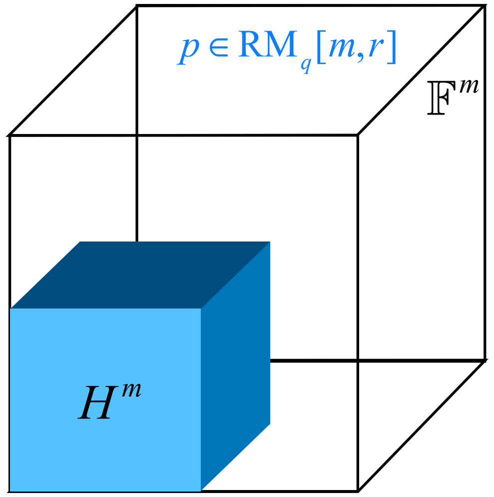
NO!

[JKRS09]

Warmup: Subcube Sums of Reed-Muller

Warmup: Let $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➡ Computing $\sum_{\alpha \in H^m} p(\alpha)$ takes $\tilde{\Omega}(|H^m|)$ queries

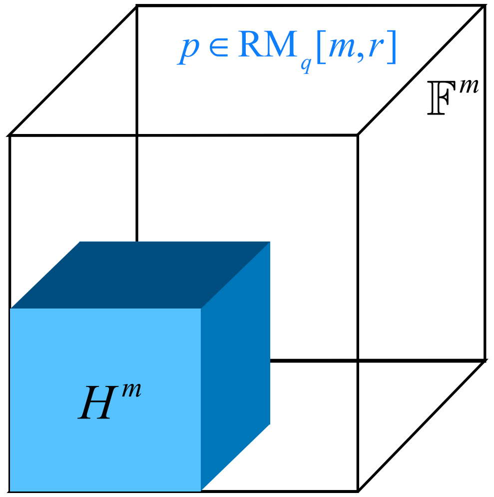


Warmup: Subcube Sums of Reed-Muller

Warmup: **Let** $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➡ **Computing** $\sum_{\alpha \in H^m} p(\alpha)$ **takes** $\tilde{\Omega}(|H^m|)$ **queries**

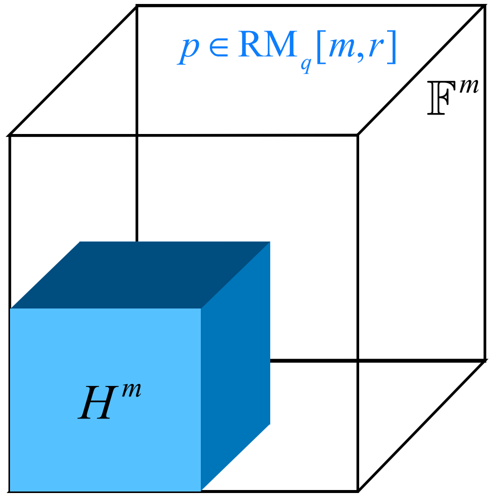
Suppose H={0,1}



Warmup: Subcube Sums of Reed-Muller

Warmup: **Let** $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➡ **Computing** $\sum_{\alpha \in H^m} p(\alpha)$ **takes** $\tilde{\Omega}(|H^m|)$ **queries**



Suppose H={0,1}

Approach: Reduction from **communication complexity**

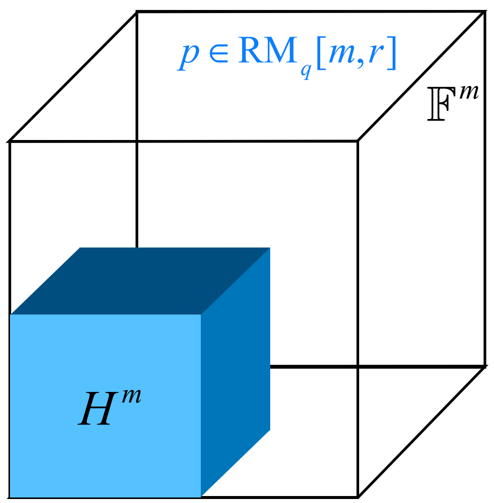
Warmup: Subcube Sums of Reed-Muller

Warmup: **Let** $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➧ **Computing** $\sum_{\alpha \in H^m} p(\alpha)$ **takes** $\tilde{\Omega}(|H^m|)$ **queries**

Suppose H={0,1}

Approach: Reduction from **communication complexity**



$x \in \{0,1\}^n$

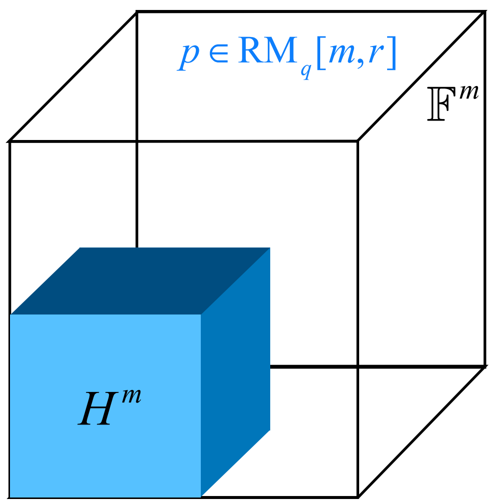
Warmup: Subcube Sums of Reed-Muller

Warmup: **Let** $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➧ **Computing** $\sum_{\alpha \in H^m} p(\alpha)$ **takes** $\tilde{\Omega}(|H^m|)$ **queries**

Suppose H={0,1}

Approach: Reduction from **communication complexity**



$x \in \{0,1\}^n$

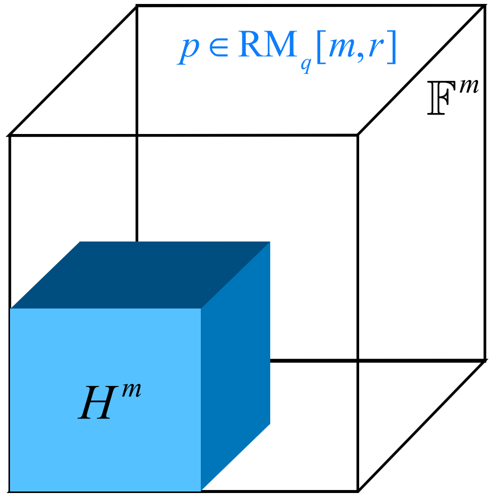
$y \in \{0,1\}^n$

Warmup: Subcube Sums of Reed-Muller

Warmup: **Let** $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➡ **Computing** $\sum_{\alpha \in H^m} p(\alpha)$ **takes** $\tilde{\Omega}(|H^m|)$ **queries**

Suppose H={0,1}



Approach: Reduction from **communication complexity**

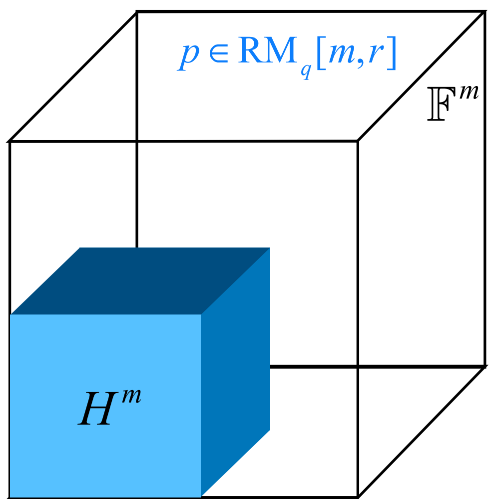
$x \in \{0,1\}^n$

$y \in \{0,1\}^n$

Warmup: Subcube Sums of Reed-Muller

Warmup: **Let** $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➧ **Computing** $\sum_{\alpha \in H^m} p(\alpha)$ **takes** $\tilde{\Omega}(|H^m|)$ **queries**



Suppose H={0,1}

Approach: Reduction from **communication complexity**

$x \in \{0,1\}^n$ → $y \in \{0,1\}^n$

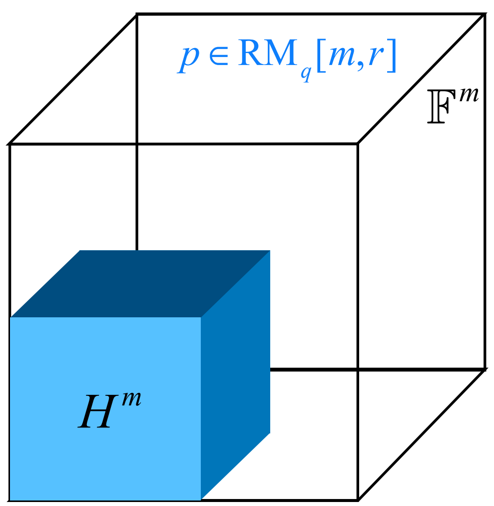
$\Omega(n)$ **communication required to decide unique-disjointness:** $\exists! \quad x_i = y_i = 1$

Warmup: Subcube Sums of Reed-Muller

Warmup: **Let** $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➧ **Computing** $\sum_{\alpha \in H^m} p(\alpha)$ **takes** $\tilde{\Omega}(|H^m|)$ **queries**

Suppose H={0,1}



Approach: Reduction from **communication complexity**

$x \in \{0,1\}^n$

$y \in \{0,1\}^n$

$\Omega(n)$ **communication required to decide unique-disjointness:** $\exists! \quad x_i = y_i = 1$

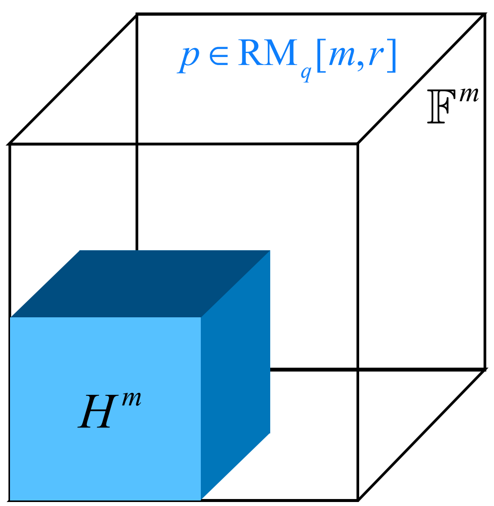
Towards contradiction: suppose $\sum_{\alpha \in H^m} p(\alpha)$ computable with $\tilde{o}(|H^m|)$ queries

Warmup: Subcube Sums of Reed-Muller

Warmup: **Let** $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➡ **Computing** $\sum_{\alpha \in H^m} p(\alpha)$ **takes** $\tilde{\Omega}(|H^m|)$ **queries**

Suppose H={0,1}



Approach: Reduction from **communication complexity**

$x \in \{0,1\}^n$

$y \in \{0,1\}^n$

$\Omega(n)$ **communication required to decide unique-disjointness:** $\exists! \quad x_i = y_i = 1$

Towards contradiction: suppose $\sum_{\alpha \in H^m} p(\alpha)$ computable with $\tilde{o}(|H^m|)$ queries

Construct a protocol for unique disjointness!

Warmup: Subcube Sums of Reed-Muller

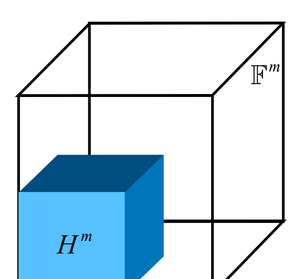
Warmup: **Let** $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➡ **Computing** $\sum_{\alpha \in H^m} p(\alpha)$ **takes** $\tilde{\Omega}(|H^m|)$ **queries**

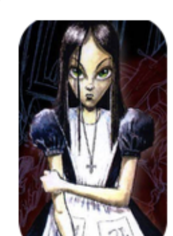
The protocol

Towards contradiction:

$\sum_{\alpha \in H^m} p(\alpha)$ computable with $\tilde{o}(|H^m|)$ queries



Reduction from communication complexity



$x \in \{0,1\}^n$ $y \in \{0,1\}^n$

$\Omega(n)$ communication required to decide if $\exists! \quad x_i = y_i = 1$

Warmup: Subcube Sums of Reed-Muller

Warmup: **Let** $p \in \mathrm{RM}_q[m,r]$

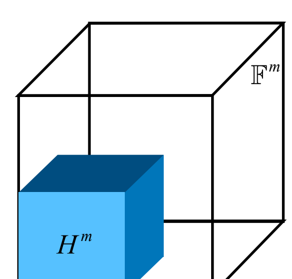
If $r \geq 2$ ➡ **Computing** $\sum_{\alpha \in H^m} p(\alpha)$ **takes** $\tilde{\Omega}(|H^m|)$ **queries**

The protocol

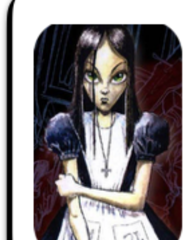
$x \in \{0,1\}^n$

Towards contradiction:

$\sum_{\alpha \in H^m} p(\alpha)$ computable with $\tilde{o}(|H^m|)$ queries



Reduction from communication complexity



$x \in \{0,1\}^n$ $y \in \{0,1\}^n$

$\Omega(n)$ **communication required to decide if** $\exists! \quad x_i = y_i = 1$

Warmup: Subcube Sums of Reed-Muller

Warmup: **Let** $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➧ **Computing** $\sum_{\alpha \in H^m} p(\alpha)$ **takes** $\tilde{\Omega}(|H^m|)$ **queries**

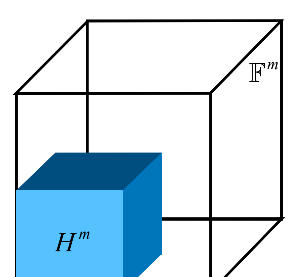
The protocol

$x \in \{0,1\}^n$

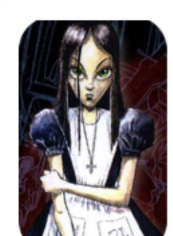
$f_x : H^m \to \{0,1\}$

Towards contradiction:

$\sum_{\alpha \in H^m} p(\alpha)$ computable with $\tilde{o}(|H^m|)$ queries



Reduction from communication complexity



$x \in \{0,1\}^n$ $y \in \{0,1\}^n$

$\Omega(n)$ **communication required to decide if** $\exists!\ \ x_i = y_i = 1$

Warmup: Subcube Sums of Reed-Muller

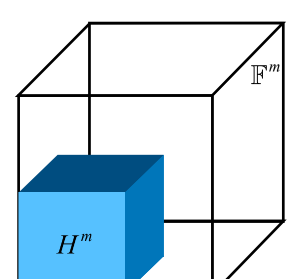
Warmup: Let $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➡ Computing $\sum_{\alpha \in H^m} p(\alpha)$ takes $\tilde{\Omega}(|H^m|)$ queries

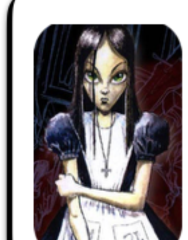
The protocol

$$x \in \{0,1\}^n$$
$$f_x : H^m \to \{0,1\}$$
$$p_x : \mathbb{F}^m \to \mathbb{F}$$

Towards contradiction:
$\sum_{\alpha \in H^m} p(\alpha)$ computable with $\tilde{o}(|H^m|)$ queries



Reduction from communication complexity



$x \in \{0,1\}^n$ $y \in \{0,1\}^n$

$\Omega(n)$ communication required to decide if $\exists! \quad x_i = y_i = 1$

Warmup: Subcube Sums of Reed-Muller

Warmup: **Let** $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➡ **Computing** $\sum_{\alpha \in H^m} p(\alpha)$ **takes** $\tilde{\Omega}(|H^m|)$ **queries**

The protocol

$x \in \{0,1\}^n$

$f_x : H^m \to \{0,1\}$

$p_x : \mathbb{F}^m \to \mathbb{F}$

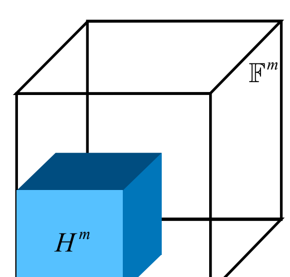
$y \in \{0,1\}^n$

$f_y : H^m \to \{0,1\}$

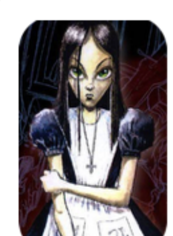
$p_y : \mathbb{F}^m \to \mathbb{F}$

Towards contradiction:

$\sum_{\alpha \in H^m} p(\alpha)$ computable with $\tilde{o}(|H^m|)$ queries



Reduction from communication complexity



$x \in \{0,1\}^n$ $y \in \{0,1\}^n$

$\Omega(n)$ **communication required to decide if**

$\exists! \quad x_i = y_i = 1$

Warmup: Subcube Sums of Reed-Muller

Warmup: Let $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➡ Computing $\sum_{\alpha \in H^m} p(\alpha)$ takes $\tilde{\Omega}(|H^m|)$ queries

The protocol

$x \in \{0,1\}^n$

$f_x : H^m \to \{0,1\}$

$p_x : \mathbb{F}^m \to \mathbb{F}$

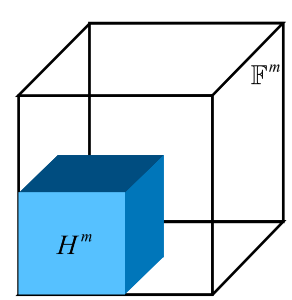
$y \in \{0,1\}^n$

$f_y : H^m \to \{0,1\}$

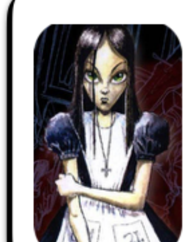
$p_y : \mathbb{F}^m \to \mathbb{F}$

$p(\alpha) = p_x(\alpha) \cdot p_y(\alpha)$

Towards contradiction: $\sum_{\alpha \in H^m} p(\alpha)$ computable with $\tilde{o}(|H^m|)$ queries



Reduction from communication complexity



$x \in \{0,1\}^n$ $y \in \{0,1\}^n$

$\Omega(n)$ communication required to decide if $\exists! \quad x_i = y_i = 1$

Warmup: Subcube Sums of Reed-Muller

Warmup: Let $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➡ Computing $\sum_{\alpha \in H^m} p(\alpha)$ takes $\tilde{\Omega}(|H^m|)$ queries

The protocol

$x \in \{0,1\}^n$

$f_x : H^m \to \{0,1\}$

$p_x : \mathbb{F}^m \to \mathbb{F}$

$y \in \{0,1\}^n$

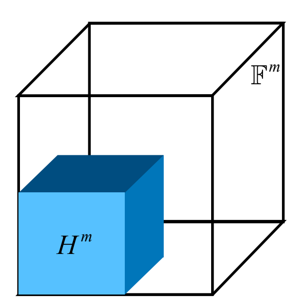
$f_y : H^m \to \{0,1\}$

$p_y : \mathbb{F}^m \to \mathbb{F}$

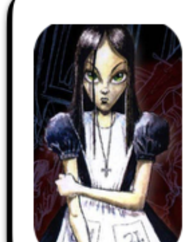
$p(\alpha) = p_x(\alpha) \cdot p_y(\alpha)$

Towards contradiction:

$\sum_{\alpha \in H^m} p(\alpha)$ computable with $\tilde{o}(|H^m|)$ queries



Reduction from communication complexity



$x \in \{0,1\}^n$ $y \in \{0,1\}^n$

$\Omega(n)$ communication required to decide if $\exists! \quad x_i = y_i = 1$

Warmup: Subcube Sums of Reed-Muller

Warmup: **Let** $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➡ **Computing** $\sum_{\alpha \in H^m} p(\alpha)$ **takes** $\tilde{\Omega}(|H^m|)$ **queries**

The protocol

$x \in \{0,1\}^n$

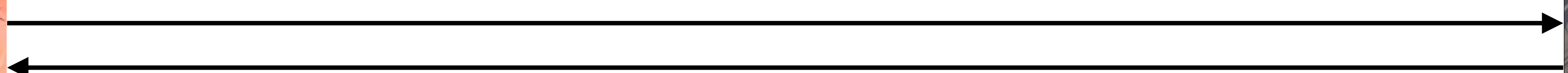
$f_x : H^m \to \{0,1\}$

$p_x : \mathbb{F}^m \to \mathbb{F}$

$y \in \{0,1\}^n$

$f_y : H^m \to \{0,1\}$

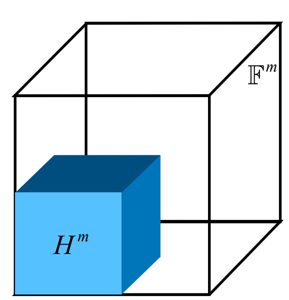
$p_y : \mathbb{F}^m \to \mathbb{F}$



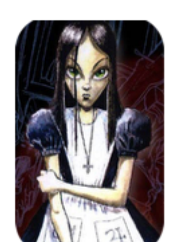
$$p(\alpha) = p_x(\alpha) \cdot p_y(\alpha)$$

$(x,y) \in \mathrm{DISJ}$

Towards contradiction:
$\sum_{\alpha \in H^m} p(\alpha)$ computable with $\tilde{o}(|H^m|)$ queries



Reduction from communication complexity



$x \in \{0,1\}^n$

$y \in \{0,1\}^n$

$\Omega(n)$ communication required to decide if $\exists! \; x_i = y_i = 1$

Warmup: Subcube Sums of Reed-Muller

Warmup: **Let** $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➧ **Computing** $\sum_{\alpha \in H^m} p(\alpha)$ **takes** $\tilde{\Omega}(|H^m|)$ **queries**

The protocol

$x \in \{0,1\}^n$

$f_x : H^m \to \{0,1\}$

$p_x : \mathbb{F}^m \to \mathbb{F}$

$y \in \{0,1\}^n$

$f_y : H^m \to \{0,1\}$

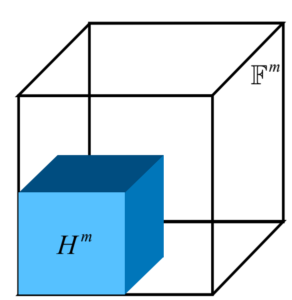
$p_y : \mathbb{F}^m \to \mathbb{F}$

$$p(\alpha) = p_x(\alpha) \cdot p_y(\alpha)$$

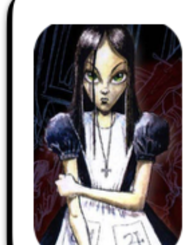
$(x,y) \in \mathrm{DISJ}$ ➧ $\sum_{\alpha \in H^m} f_x(\alpha) \cdot f_y(\alpha) = 0$

Towards contradiction:

$\sum_{\alpha \in H^m} p(\alpha)$ computable with $\tilde{o}(|H^m|)$ queries



Reduction from communication complexity



$x \in \{0,1\}^n$ $y \in \{0,1\}^n$

$\Omega(n)$ **communication required to decide if** $\exists! \quad x_i = y_i = 1$

Warmup: Subcube Sums of Reed-Muller

Warmup: Let $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➧ Computing $\sum_{\alpha \in H^m} p(\alpha)$ takes $\tilde{\Omega}(|H^m|)$ queries

The protocol

$x \in \{0,1\}^n$

$f_x : H^m \to \{0,1\}$

$p_x : \mathbb{F}^m \to \mathbb{F}$

$y \in \{0,1\}^n$

$f_y : H^m \to \{0,1\}$

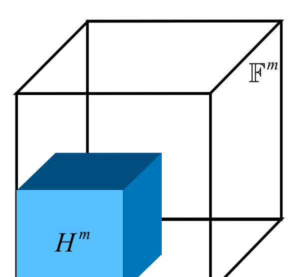
$p_y : \mathbb{F}^m \to \mathbb{F}$

$p(\alpha) = p_x(\alpha) \cdot p_y(\alpha)$

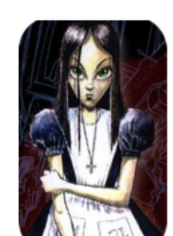
$(x,y) \in \mathrm{DISJ}$ ➧ $\sum_{\alpha \in H^m} f_x(\alpha) \cdot f_y(\alpha) = 0$ ➧ $\sum_{\alpha \in H^m} p(\alpha) = 0$

Towards contradiction:

$\sum_{\alpha \in H^m} p(\alpha)$ computable with $\tilde{o}(|H^m|)$ queries



Reduction from communication complexity



$x \in \{0,1\}^n$

$y \in \{0,1\}^n$

$\Omega(n)$ communication required to decide if $\exists!\ x_i = y_i = 1$

Warmup: Subcube Sums of Reed-Muller

Warmup: **Let** $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➧ **Computing** $\sum_{\alpha \in H^m} p(\alpha)$ **takes** $\tilde{\Omega}(|H^m|)$ **queries**

The protocol

$x \in \{0,1\}^n$

$f_x : H^m \to \{0,1\}$

$p_x : \mathbb{F}^m \to \mathbb{F}$

$y \in \{0,1\}^n$

$f_y : H^m \to \{0,1\}$

$p_y : \mathbb{F}^m \to \mathbb{F}$

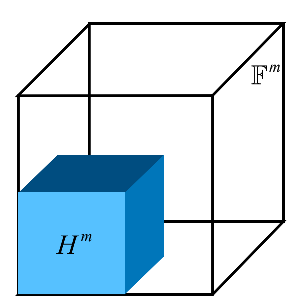
$p(\alpha) = p_x(\alpha) \cdot p_y(\alpha)$

$(x,y) \in \mathrm{DISJ}$ ➧ $\sum_{\alpha \in H^m} f_x(\alpha) \cdot f_y(\alpha) = 0$ ➧ $\sum_{\alpha \in H^m} p(\alpha) = 0$

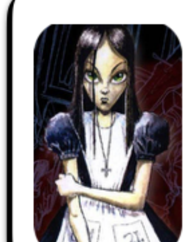
$(x,y) \notin \mathrm{DISJ}$

Towards contradiction:

$\sum_{\alpha \in H^m} p(\alpha)$ computable with $\tilde{o}(|H^m|)$ queries



Reduction from communication complexity



$x \in \{0,1\}^n$

$y \in \{0,1\}^n$

$\Omega(n)$ communication required to decide if

$\exists! \quad x_i = y_i = 1$

Warmup: Subcube Sums of Reed-Muller

Warmup: Let $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➧ Computing $\sum_{\alpha \in H^m} p(\alpha)$ takes $\tilde{\Omega}(|H^m|)$ queries

The protocol

$x \in \{0,1\}^n$

$f_x : H^m \to \{0,1\}$

$p_x : \mathbb{F}^m \to \mathbb{F}$

$y \in \{0,1\}^n$

$f_y : H^m \to \{0,1\}$

$p_y : \mathbb{F}^m \to \mathbb{F}$

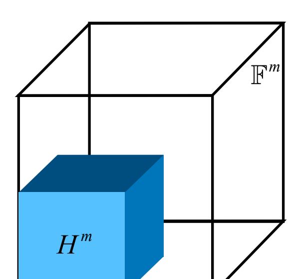
$p(\alpha) = p_x(\alpha) \cdot p_y(\alpha)$

$(x,y) \in \mathrm{DISJ}$ ➧ $\sum_{\alpha \in H^m} f_x(\alpha) \cdot f_y(\alpha) = 0$ ➧ $\sum_{\alpha \in H^m} p(\alpha) = 0$

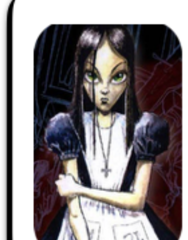
$(x,y) \notin \mathrm{DISJ}$ ➧ $\sum_{\alpha \in H^m} f_x(\alpha) \cdot f_y(\alpha) = 1$

Towards contradiction:

$\sum_{\alpha \in H^m} p(\alpha)$ computable with $\tilde{o}(|H^m|)$ queries



Reduction from communication complexity



$x \in \{0,1\}^n$ $y \in \{0,1\}^n$

$\Omega(n)$ communication required to decide if $\exists!\ x_i = y_i = 1$

Warmup: Subcube Sums of Reed-Muller

Warmup: Let $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➧ Computing $\sum_{\alpha \in H^m} p(\alpha)$ takes $\tilde{\Omega}(|H^m|)$ queries

The protocol

$x \in \{0,1\}^n$

$f_x : H^m \to \{0,1\}$

$p_x : \mathbb{F}^m \to \mathbb{F}$

$y \in \{0,1\}^n$

$f_y : H^m \to \{0,1\}$

$p_y : \mathbb{F}^m \to \mathbb{F}$

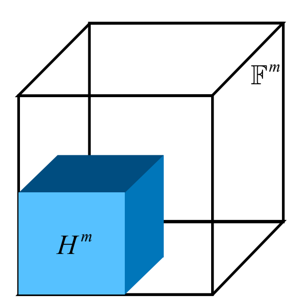
$p(\alpha) = p_x(\alpha) \cdot p_y(\alpha)$

$(x,y) \in \mathrm{DISJ}$ ➧ $\sum_{\alpha \in H^m} f_x(\alpha) \cdot f_y(\alpha) = 0$ ➧ $\sum_{\alpha \in H^m} p(\alpha) = 0$

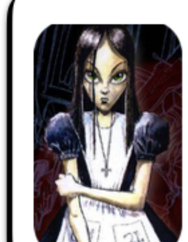
$(x,y) \notin \mathrm{DISJ}$ ➧ $\sum_{\alpha \in H^m} f_x(\alpha) \cdot f_y(\alpha) = 1$ ➧ $\sum_{\alpha \in H^m} p(\alpha) = 1$

Towards contradiction:

$\sum_{\alpha \in H^m} p(\alpha)$ computable with $\tilde{o}(|H^m|)$ queries



Reduction from communication complexity



$x \in \{0,1\}^n$ $y \in \{0,1\}^n$

$\Omega(n)$ communication required to decide if $\exists!\ x_i = y_i = 1$

Warmup: Subcube Sums of Reed-Muller

Warmup: Let $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➧ Computing $\sum_{\alpha \in H^m} p(\alpha)$ takes $\tilde{\Omega}(|H^m|)$ queries

The protocol

$x \in \{0,1\}^n$

$f_x : H^m \to \{0,1\}$

$p_x : \mathbb{F}^m \to \mathbb{F}$

$y \in \{0,1\}^n$

$f_y : H^m \to \{0,1\}$

$p_y : \mathbb{F}^m \to \mathbb{F}$

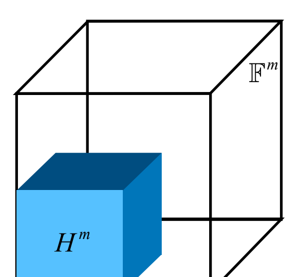
$p(\alpha) = p_x(\alpha) \cdot p_y(\alpha)$

$(x,y) \in \mathrm{DISJ}$ ➧ $\sum_{\alpha \in H^m} f_x(\alpha) \cdot f_y(\alpha) = 0$ ➧ $\sum_{\alpha \in H^m} p(\alpha) = 0$

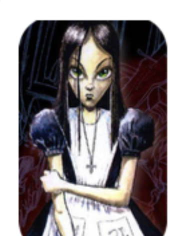
$(x,y) \notin \mathrm{DISJ}$ ➧ $\sum_{\alpha \in H^m} f_x(\alpha) \cdot f_y(\alpha) = 1$ ➧ $\sum_{\alpha \in H^m} p(\alpha) = 1$

Towards contradiction:

$\sum_{\alpha \in H^m} p(\alpha)$ computable with $\tilde{o}(|H^m|)$ queries



Reduction from communication complexity



$x \in \{0,1\}^n$ $y \in \{0,1\}^n$

$\Omega(n)$ communication required to decide if

$\exists! \quad x_i = y_i = 1$

What is missing?

So far, we showed:

Warmup: **Let** $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➡ **Computing** $\sum_{\alpha \in H^m} p(\alpha)$ **takes** $\tilde{\Omega}(|H^m|)$ **queries**

What is missing?

So far, we showed:

Warmup: **Let** $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➡ **Computing** $\sum_{\alpha \in H^m} p(\alpha)$ **takes** $\tilde{\Omega}(|H^m|)$ **queries**

This suffices for committing to an ELEMENT

What is missing?

So far, we showed:

Warmup: **Let** $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➡ **Computing** $\sum_{\alpha \in H^m} p(\alpha)$ **takes** $\tilde{\Omega}(|H^m|)$ **queries**

This suffices for committing to an ELEMENT

We need to commit to a POLYNOMIAL!

What is missing?

So far, we showed:

Warmup: Let $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➧ Computing $\sum_{\alpha \in H^m} p(\alpha)$ takes $\tilde{\Omega}(|H^m|)$ queries

This suffices for committing to an ELEMENT

We need to commit to a POLYNOMIAL!

Now, we wish to de-commit w.r.t. a single point

What is missing?

So far, we showed:

Warmup: **Let** $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➡ **Computing** $\sum_{\alpha \in H^m} p(\alpha)$ **takes** $\tilde{\Omega}(|H^m|)$ **queries**

This suffices for committing to an ELEMENT

We need to commit to a POLYNOMIAL!

Now, we wish to de-commit w.r.t. a single point

WITHOUT LEAKING information about other points

What is missing?

So far, we showed:

Warmup: Let $p \in \mathrm{RM}_q[m,r]$

If $r \geq 2$ ➡ Computing $\sum_{\alpha \in H^m} p(\alpha)$ takes $\tilde{\Omega}(|H^m|)$ queries

This suffices for committing to an ELEMENT

We need to commit to a POLYNOMIAL!

Now, we wish to de-commit w.r.t. a single point

WITHOUT LEAKING information about other points

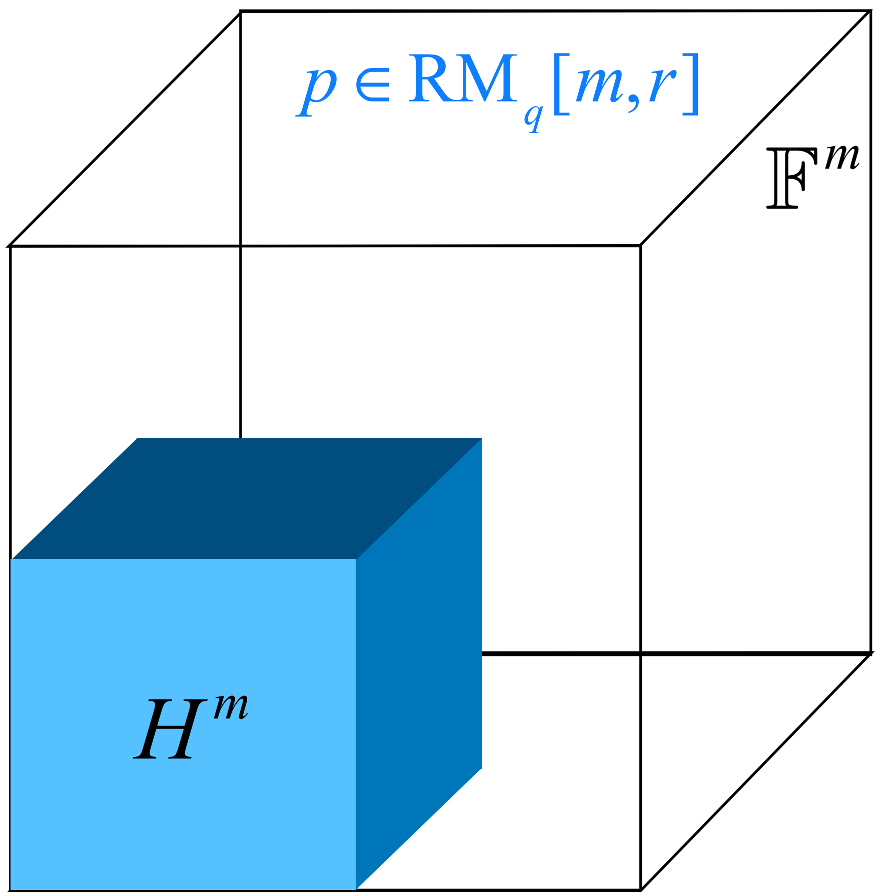
Requires new algebraic complexity lower bounds!

The General Case: Reed-Muller Subcube Sums

$$\mathrm{RM}_q[m,r] = \{\langle p(\alpha)\rangle \mid p \in \mathbb{F}_q^{\le r}[X_1,\ldots,X_m]\}$$

The General Case: Reed-Muller Subcube Sums

$$\mathrm{RM}_q[m,r] = \{\langle p(\alpha)\rangle \mid p \in \mathbb{F}_q^{\le r}[X_1,\ldots,X_m]\}$$

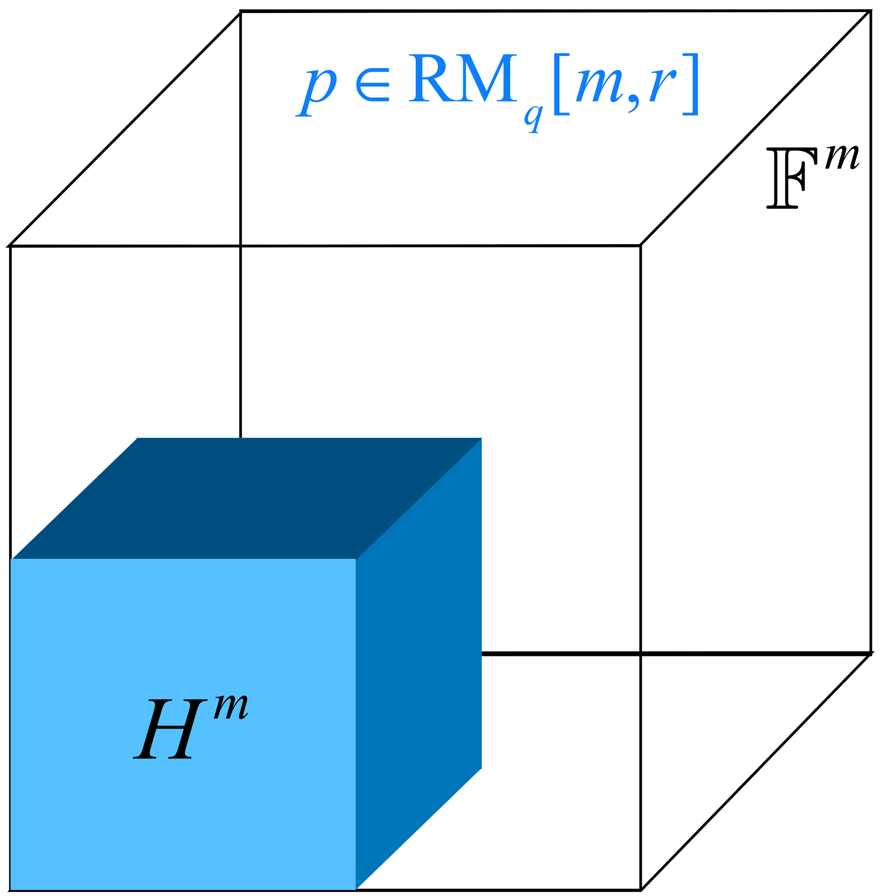


Given p, not only the sum over the whole cube

$\sum_{\alpha_1,\ldots,\alpha_m \in H} p(\alpha_1,\ldots,\alpha_m)$ **is hard to compute**

The General Case: Reed-Muller Subcube Sums

$$\mathrm{RM}_q[m,r] = \{\langle p(\alpha)\rangle \mid p \in \mathbb{F}_q^{\leq r}[X_1,\ldots,X_m]\}$$



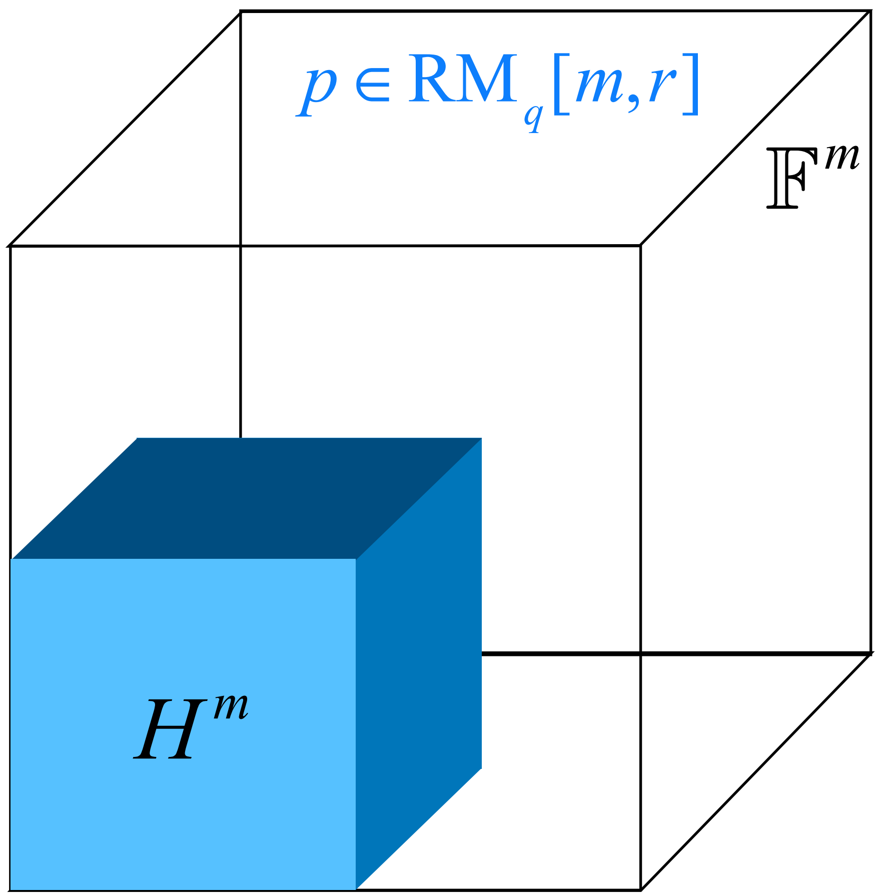
Given p, not only the sum over the whole cube

$\sum_{\alpha_1,\ldots,\alpha_m \in H} p(\alpha_1,\ldots,\alpha_m)$ **is hard to compute**

But also partial (subcube) sums!

The General Case: Reed-Muller Subcube Sums

$$\mathrm{RM}_q[m,r] = \{\langle p(\alpha)\rangle \mid p \in \mathbb{F}_q^{\le r}[X_1,\ldots,X_m]\}$$



Given p, not only the sum over the whole cube

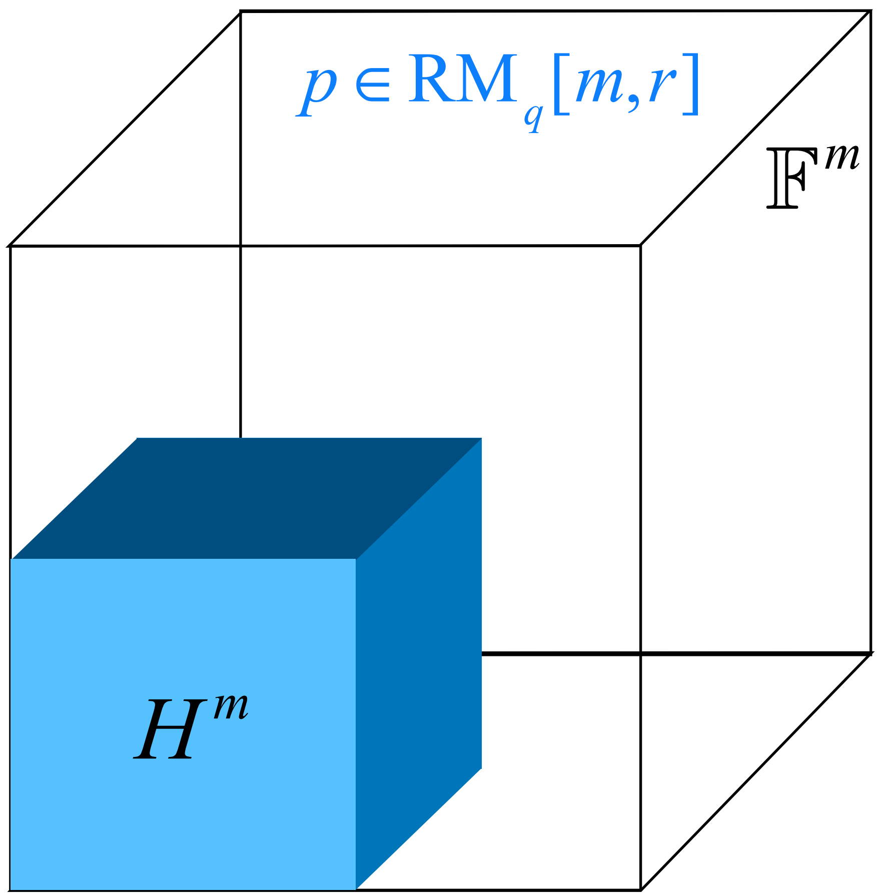
$\sum_{\alpha_1,\ldots,\alpha_m \in H} p(\alpha_1,\ldots,\alpha_m)$ **is hard to compute**

But also partial (subcube) sums!

$$\langle \sum p(z_1,\alpha_2\ldots,\alpha_m)\rangle_{z_1\in\mathbb{F}}$$

The General Case: Reed-Muller Subcube Sums

$$\mathrm{RM}_q[m,r] = \{\langle p(\alpha)\rangle \mid p \in \mathbb{F}_q^{\le r}[X_1,\ldots,X_m]\}$$



Given p**, not only the sum over the whole cube**

$\sum_{\alpha_1,\ldots,\alpha_m \in H} p(\alpha_1,\ldots,\alpha_m)$ **is hard to compute**

But also partial (subcube) sums!

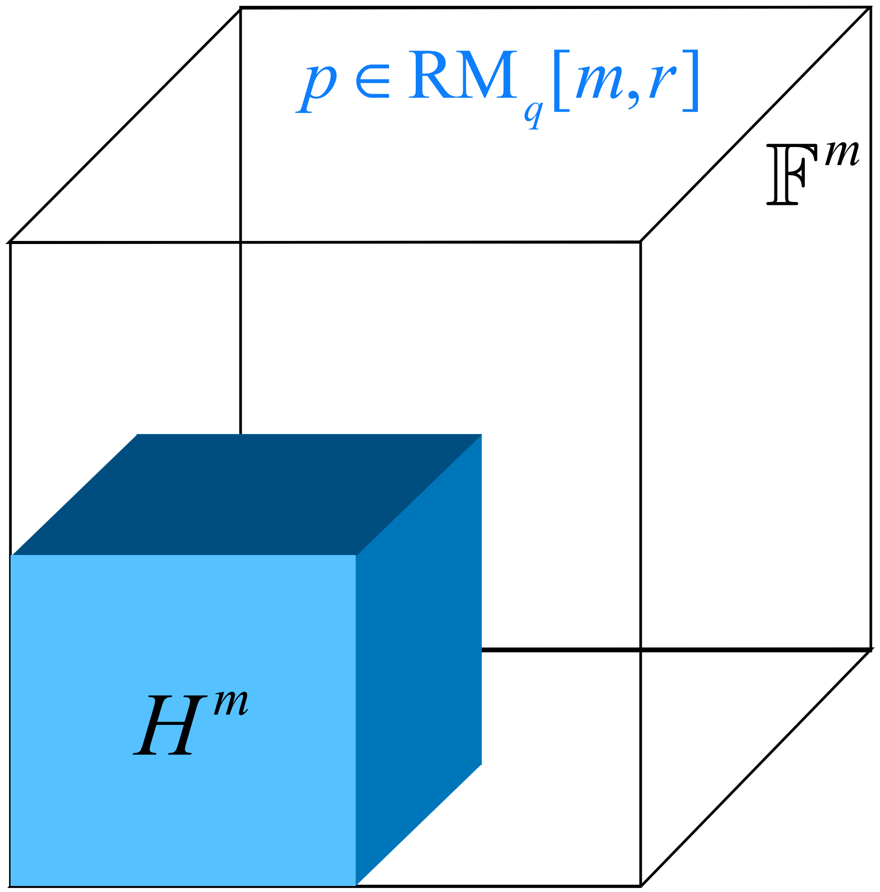
$$\langle \sum p(z_1,\alpha_2\ldots,\alpha_m)\rangle_{z_1\in\mathbb{F}}$$

$$\langle \sum p(z_1,z_2,\alpha_3\ldots,\alpha_m)\rangle_{z_1,z_2\in\mathbb{F}}$$

$$\langle \sum p(z_1,\ldots,z_{m-1},\alpha_m)\rangle_{z_1,\ldots,z_{m-1}\in\mathbb{F}}$$

The General Case: Reed-Muller Subcube Sums

$$\mathrm{RM}_q[m,r] = \{\langle p(\alpha)\rangle \mid p \in \mathbb{F}_q^{\le r}[X_1,\ldots,X_m]\}$$



Given p, not only the sum over the whole cube

$$\sum_{\alpha_1,\ldots,\alpha_m \in H} p(\alpha_1,\ldots,\alpha_m)$$ **is hard to compute**

But also partial (subcube) sums!

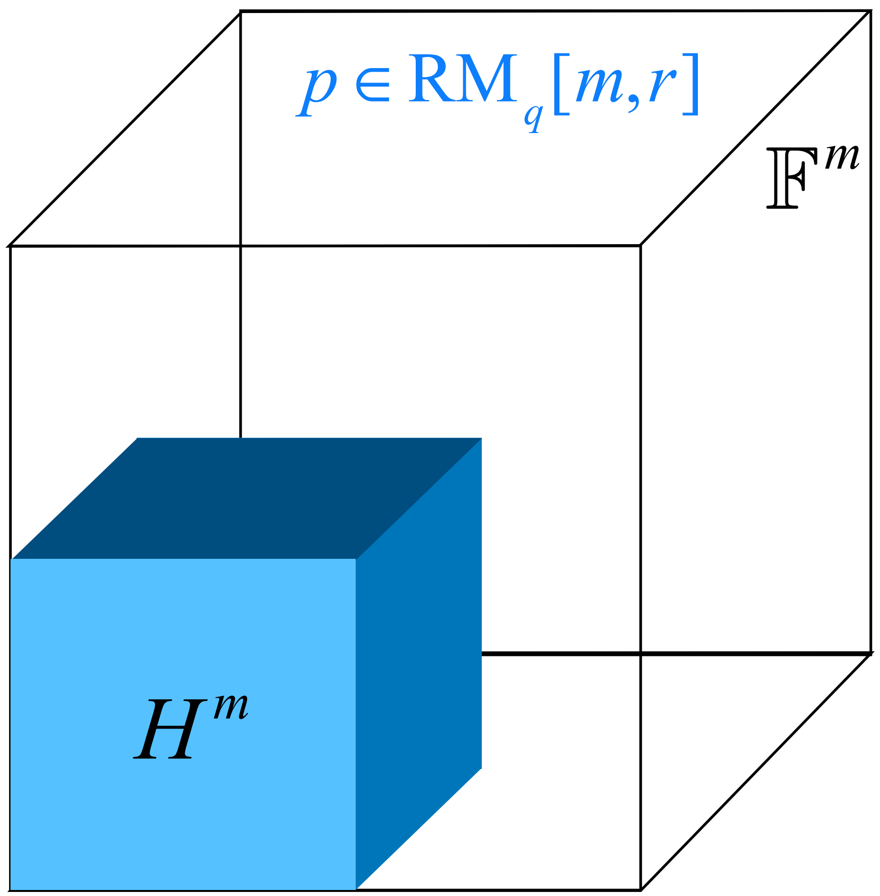
$$\langle \sum p(z_1,\alpha_2\ldots,\alpha_m)\rangle_{z_1\in\mathbb{F}}$$
$$\langle \sum p(z_1,z_2,\alpha_3\ldots,\alpha_m)\rangle_{z_1,z_2\in\mathbb{F}}$$
$$\langle \sum p(z_1,\ldots,z_{m-1},\alpha_m)\rangle_{z_1,\ldots,z_{m-1}\in\mathbb{F}}$$

and their linear combinations!

The General Case: Reed-Muller Subcube Sums

Extending the low-degree extension!

$$\Sigma\mathrm{RM}_q[m,r] = \left\{ \left\langle \sum_{\substack{\vec{z}\in\mathbb{F}^{\leq m} \\ \alpha\in H^{m-|\vec{z}|}}} p(\vec{z},\vec{\alpha}) \right\rangle \;\middle|\; p \in \mathbb{F}_q^{\leq r}[X_1,\ldots,X_m] \right\}$$



Given p**, not only the sum over the whole cube**

$\sum_{\alpha_1,\ldots,\alpha_m\in H} p(\alpha_1,\ldots,\alpha_m)$ **is hard to compute**

But also partial (subcube) sums!

$$\left\langle \sum p(z_1,\alpha_2\ldots,\alpha_m) \right\rangle_{z_1\in\mathbb{F}}$$

$$\left\langle \sum p(z_1,z_2,\alpha_3\ldots,\alpha_m) \right\rangle_{z_1,z_2\in\mathbb{F}}$$

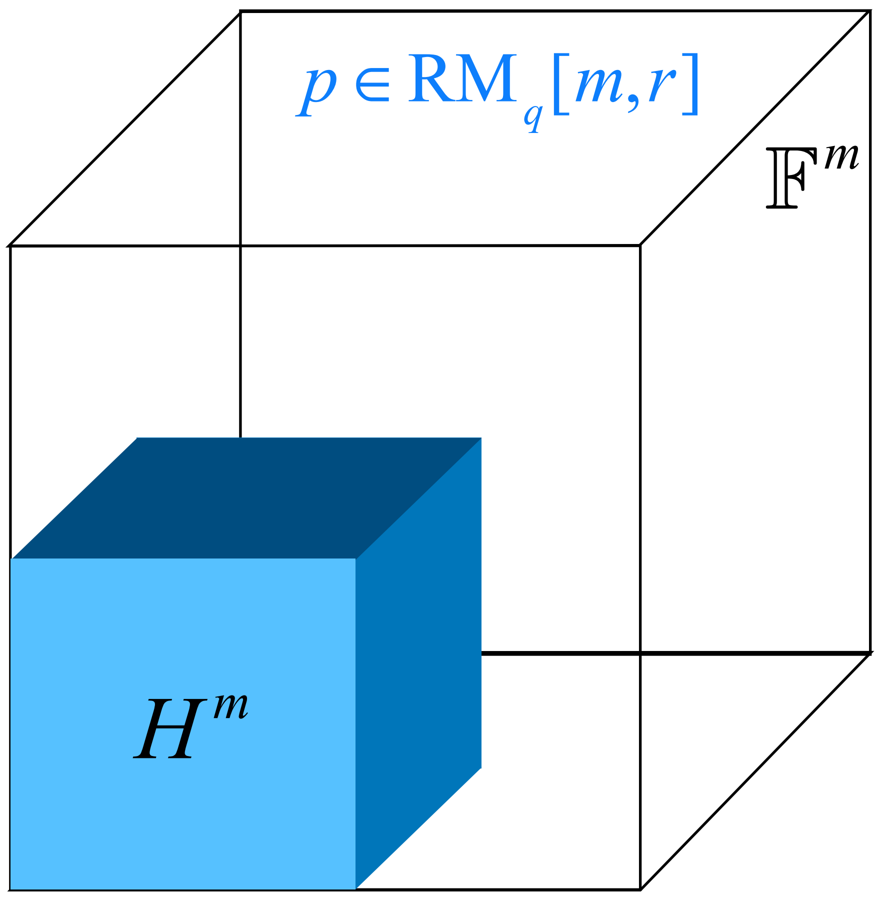
$$\left\langle \sum p(z_1,\ldots,z_{m-1},\alpha_m) \right\rangle_{z_1,\ldots,z_{m-1}\in\mathbb{F}}$$

and their linear combinations!

The General Case: Reed-Muller Subcube Sums

Extending the low-degree extension!

$$\Sigma\mathrm{RM}_q[m,r] = \left\{ \left\langle \sum_{\substack{\vec{z}\in\mathbb{F}^{\leq m} \\ \alpha\in H^{m-|\vec{z}|}}} p(\vec{z},\vec{\alpha}) \right\rangle \;\middle|\; p\in\mathbb{F}_q^{\leq r}[X_1,\ldots,X_m] \right\}$$



Given p, not only the sum over the whole cube

$$\sum_{\alpha_1,\ldots,\alpha_m\in H} p(\alpha_1,\ldots,\alpha_m)$$ **is hard to compute**

But also partial (subcube) sums!

$$\left\langle \sum p(z_1,\alpha_2,\ldots,\alpha_m) \right\rangle_{z_1\in\mathbb{F}}$$

$$\left\langle \sum p(z_1,z_2,\alpha_3,\ldots,\alpha_m) \right\rangle_{z_1,z_2\in\mathbb{F}}$$

$$\left\langle \sum p(z_1,\ldots,z_{m-1},\alpha_m) \right\rangle_{z_1,\ldots,z_{m-1}\in\mathbb{F}}$$

and their linear combinations!

Theorem: **Let** $p\in\mathrm{RM}_q[m,r]$

If $r\geq 2$ ➡ $\forall\alpha\in H^\ell$ **computing** $\sum_{\alpha\in H^\ell} p(z,\alpha)$ **takes** $\tilde{\Omega}(|H^\ell|)$ **queries**

Open Questions

Open Questions

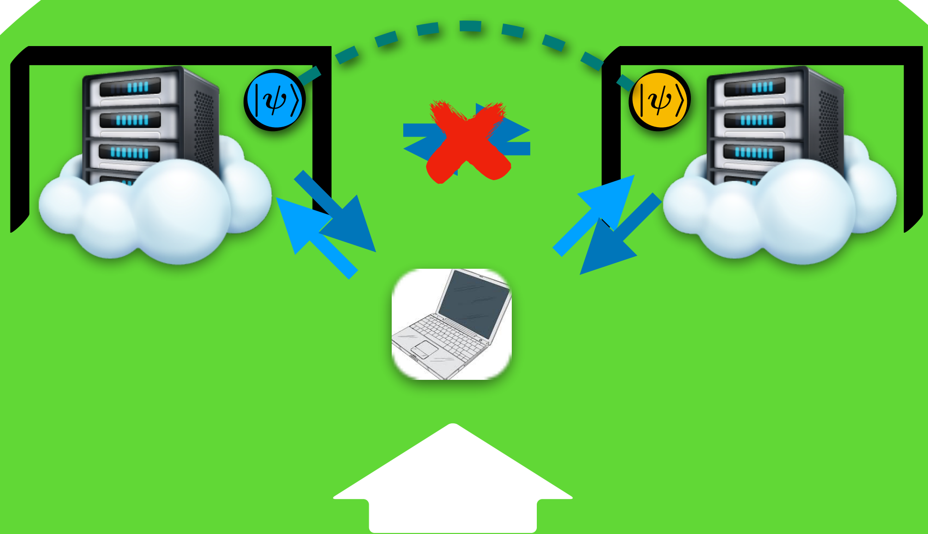
$NEXP \subseteq ZK\text{-}MIP^*$

with O(1) rounds

Open Questions

$\text{NEXP} \subseteq \text{ZK-MIP}^*$

with O(1) rounds

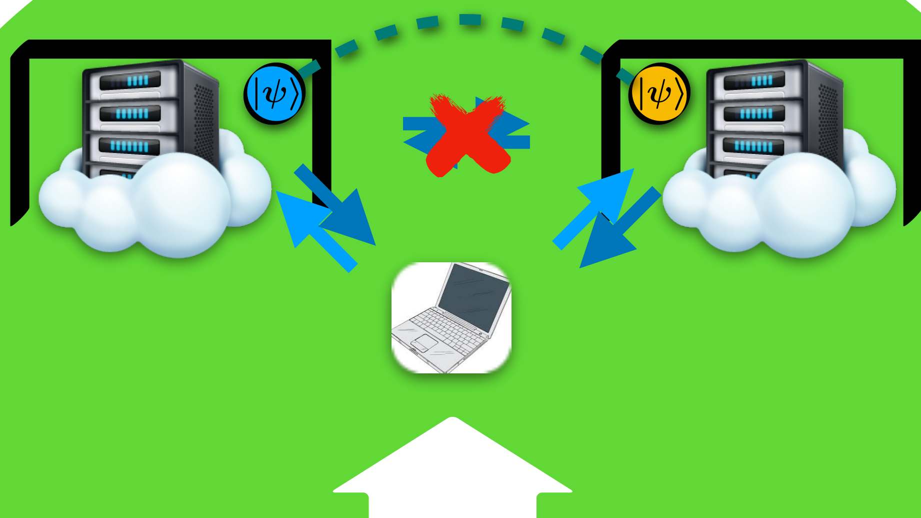


Lifting a richer class of protocols

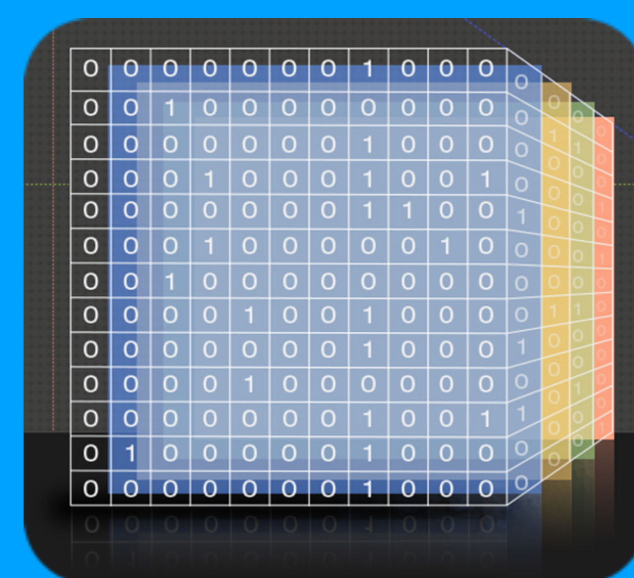
Open Questions

NEXP $\subseteq$ ZK-MIP*

with O(1) rounds



Lifting a richer
class of protocols



Entanglement-resistant
Tensor code testing

Thank you!