Quantum Proofs of Proximity

Marcel Dall'Agnol
University of Warwick

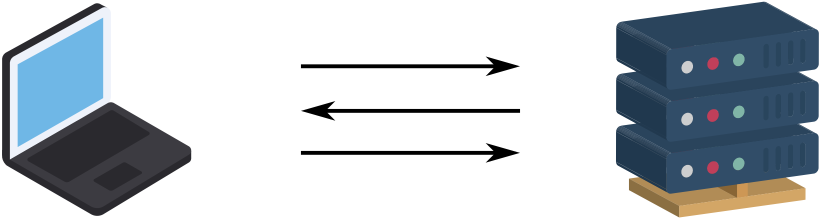
Tom Gur
University of Warwick

Subhayan Roy Moulik
University of Oxford

WPCCS 2020

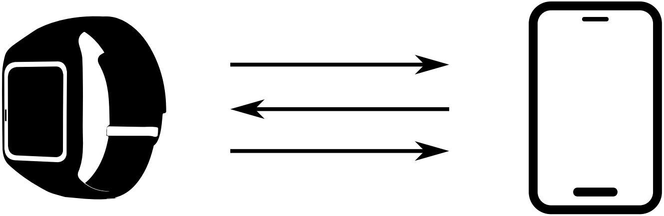
Introduction

Massive datasets, IoT: devices with *dramatically* different power.



Introduction

Massive datasets, IoT: devices with *dramatically* different power.



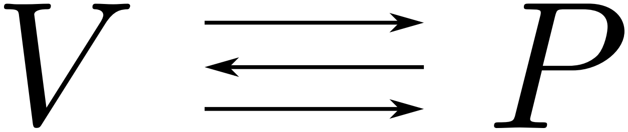
Introduction

Massive datasets, IoT: devices with *dramatically* different power.

$$V \quad \begin{array}{c} \xrightarrow{\hspace{3em}} \\ \xleftarrow{\hspace{3em}} \\ \xrightarrow{\hspace{3em}} \end{array} \quad P$$

Introduction

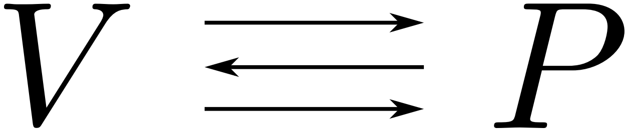
Massive datasets, IoT: devices with *dramatically* different power.



Delegation of computation: *prover* computes, *verifier* checks.

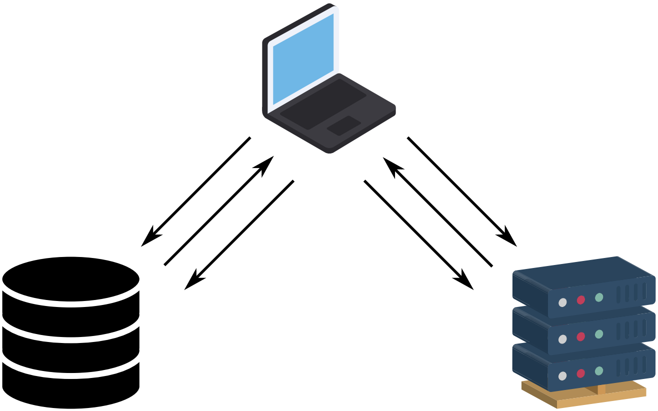
Introduction

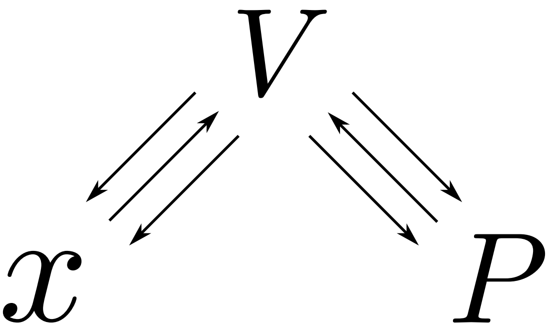
Massive datasets, IoT: devices with *dramatically* different power.



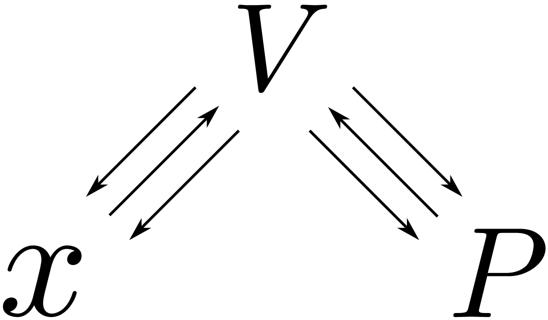
Delegation of computation: *prover* computes, *verifier* checks.
Efficient: $\tilde{O}(n)$ verifier runtime.

Proofs. . .



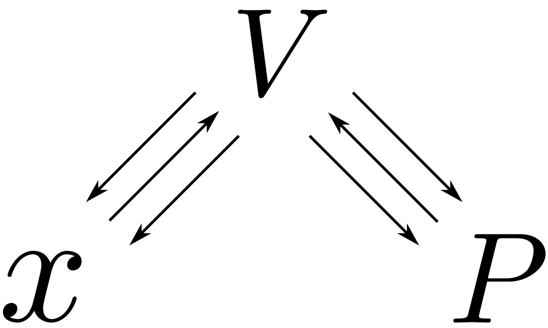


V checks if $x \in L$ by *querying* x and *communicating* with P.



V checks if $x \in L$ by *querying* x and *communicating* with P.
Query complexity q, communication complexity c.

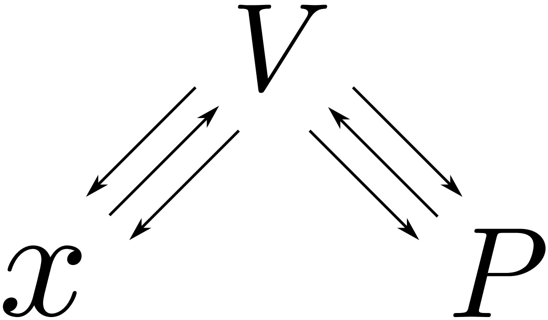
Proofs...



V checks if $x \in L$ by *querying* x and *communicating* with P.
Efficient: $o(n)$ queries and communication.

Proofs...

Example: Does x contain a 1?



Proofs…

Example: Does x contain a 1?

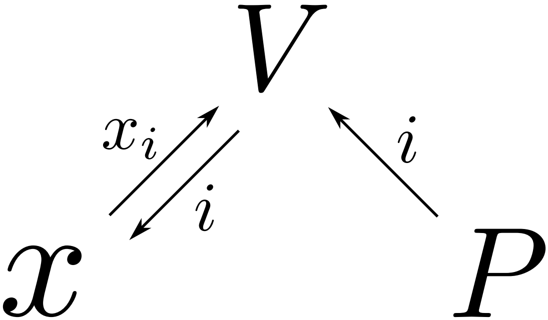
$$\begin{tikzcd} & V \arrow[dl, shift left] & \\ x \arrow[ur, shift left] & & P \arrow[ul, "i"'] \end{tikzcd}$$

Proofs...

Example: Does x contain a 1?

$$\begin{tikzcd}
 & V \arrow[dl, shift left, "i"] & \\
x \arrow[ur, shift left] & & P \arrow[ul, "i"']
\end{tikzcd}$$

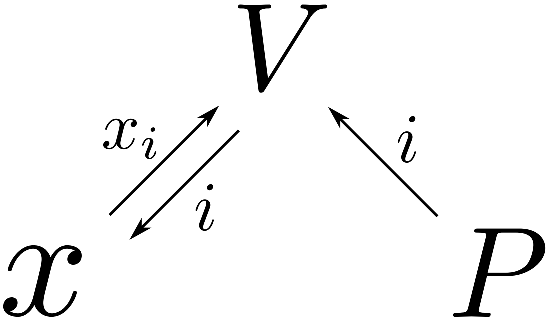
Example: Does x contain a 1?



Query complexity 1, communication complexity $\log n$.

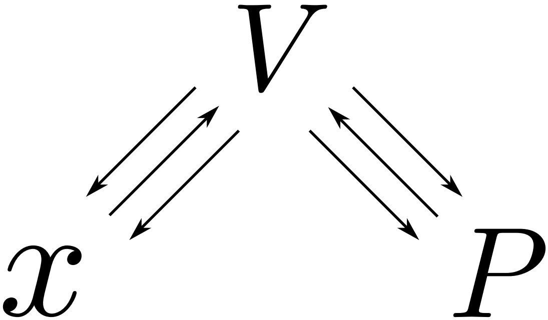
Proofs...

Example: Does x contain a 1?

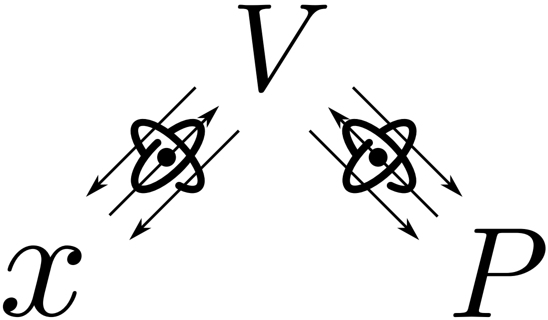


Query complexity 1, communication complexity $\log n$.
($\Omega(n)$ with no proof!)

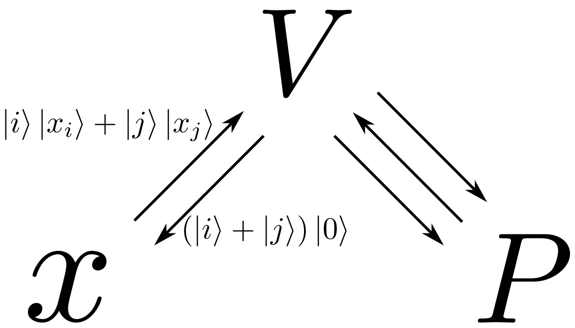
. . . of proximity. . .



V checks if $x \in \Pi$ **or x is far from** Π (property testing).
Query complexity q, communication complexity c.



V checks if $x \in \Pi$ **or x is far from** Π (property testing).
Query complexity q, communication complexity c.


V
$|i\rangle|x_i\rangle + |j\rangle|x_j\rangle$
$(|i\rangle + |j\rangle)|0\rangle$
x
P

Complexity classes

	V	$V \leftarrow P$	$V \leftrightarrow P$
Classical	$\mathcal{PT}$	$\mathcal{MAP}$	$\mathcal{IPP}$
Quantum	$\mathcal{QPT}$	$\mathcal{QMAP}$	$\mathcal{QIPP}$

Also $\mathcal{QCMAP}$: classical proof, quantum input access.

Complexity classes

	V	$V \leftarrow P$	$V \leftrightarrow P$
Classical	$\mathcal{PT}$	$\mathcal{MAP}$	$\mathcal{IPP}$
Quantum	$\mathcal{QPT}$	$\mathcal{QMAP}$	$\mathcal{QIPP}$

Also $\mathcal{QCMAP}$: classical proof, quantum input access.

$$\mathcal{C} := \mathcal{C}(\varepsilon, c, q) \text{ with } c, q = \mathrm{polylog}(n) \text{ and}$$
$$\varepsilon \text{ a small enough constant.}$$

Main result

Theorem

The following separations hold:

- *Quantum input access with a proof are more powerful in tandem than separately, i.e.,* $\mathcal{QMAP} \not\subseteq \mathcal{MAP} \cup \mathcal{QPT}$;
- *Classical proofs are weaker than quantum even with a quantum verifier, i.e.,* $\mathcal{QMAP} \not\subseteq \mathcal{QCMAP}$;
- *Quantum proofs cannot substitute for interaction, i.e.,* $\mathcal{IPP} \not\subseteq \mathcal{QMAP}$.

Main result

Theorem

The following separations hold:

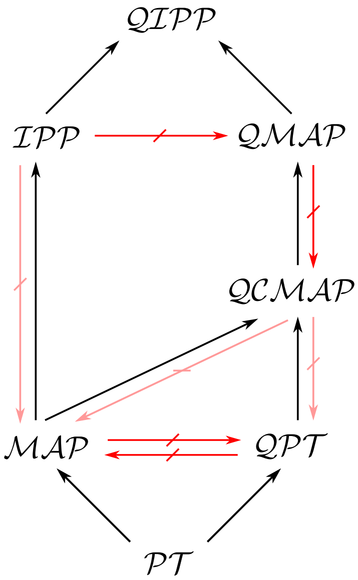
- *Quantum input access with a proof are more powerful in tandem than separately, i.e.,* $\mathcal{QMAP} \not\subseteq \mathcal{MAP} \cup \mathcal{QPT}$*;*
- *Classical proofs are weaker than quantum even with a quantum verifier, i.e.,* $\mathcal{QMAP} \not\subseteq \mathcal{QCMAP}$*;*
- *Quantum proofs cannot substitute for interaction, i.e.,* $\mathcal{IPP} \not\subseteq \mathcal{QMAP}$*.*

(Also, some inclusions in the polynomial-time setting carry over)

Main result

$$
\begin{tikzcd}
 & \mathcal{QIPP} & \\
\mathcal{IPP} \arrow[ur] \arrow[rr, "/" marking] & & \mathcal{QMAP} \arrow[ul] \arrow[d, shift left, "/" marking] \\
 & & \mathcal{QCMAP} \arrow[u, shift left] \\
\mathcal{MAP} \arrow[uu] \arrow[urr] \arrow[rr, shift left, "/" marking] & & \mathcal{QPT} \arrow[u] \arrow[ll, shift left, "/" marking] \\
 & \mathcal{PT} \arrow[ul] \arrow[ur] &
\end{tikzcd}
$$

Main result

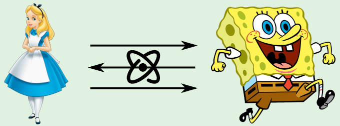


Techniques

$\mathcal{MAP} \not\subseteq \mathcal{QPT}$: disjointness + relaxed locally decodable code
$\mathcal{QPT} \not\subseteq \mathcal{MAP}$: Forrelation
$\mathcal{QMAP} \not\subseteq \mathcal{QCMAP}$: recasting $\mathcal{QMA} \not\subseteq \mathcal{QCMA}$ [AK07]
$\mathcal{IPP} \not\subseteq \mathcal{QMAP}$: permutation testing

Techniques

$\mathcal{MAP} \not\subseteq \mathcal{QPT}$: disjointness + relaxed locally decodable code



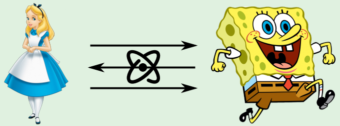
$x \in \{0,1\}^n$ $y \in \{0,1\}^n$

Given $C(x)$ and $C(y)$,
$\exists i \in [n]$ such that $x_i = y_i = 1$?

$\mathcal{QPT} \not\subseteq \mathcal{MAP}$: Forrelation
$\mathcal{QMAP} \not\subseteq \mathcal{QCMAP}$: recasting $\mathcal{QMA} \not\subseteq \mathcal{QCMA}$ [AK07]
$\mathcal{IPP} \not\subseteq \mathcal{QMAP}$: permutation testing

Techniques

$\mathcal{MAP} \not\subseteq \mathcal{QPT}$: disjointness + relaxed locally decodable code



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

Given $C(x)$ and $C(y)$,
$\exists i \in [n]$ such that $x_i = y_i = 1$?

- $\Omega(\sqrt{n})$ without proof
- $O(1)$ with $\log n$ proof

$\mathcal{QPT} \not\subseteq \mathcal{MAP}$: Forrelation
$\mathcal{QMAP} \not\subseteq \mathcal{QCMAP}$: recasting $\mathcal{QMA} \not\subseteq \mathcal{QCMA}$ [AK07]
$\mathcal{IPP} \not\subseteq \mathcal{QMAP}$: permutation testing

Techniques

$\mathcal{MAP} \not\subseteq \mathcal{QPT}$: disjointness + relaxed locally decodable code
$\mathcal{QPT} \not\subseteq \mathcal{MAP}$: Forrelation

Given $f, g : \{0,1\}^{\log n} \to \{0,1\}$, is $\langle f, \hat{g} \rangle$ small?

$\mathcal{QMAP} \not\subseteq \mathcal{QCMAP}$: recasting $\mathcal{QMA} \not\subseteq \mathcal{QCMA}$ [AK07]
$\mathcal{IPP} \not\subseteq \mathcal{QMAP}$: permutation testing

Techniques

$\mathcal{MAP} \not\subseteq \mathcal{QPT}$: disjointness + relaxed locally decodable code
$\mathcal{QPT} \not\subseteq \mathcal{MAP}$: Forrelation

Given $f, g : \{0,1\}^{\log n} \to \{0,1\}$, is $\langle f, \hat{g} \rangle$ small?

- $O(1)$ without proof
- $c \cdot q = \Omega(n^{1/4})$ with proof

$\mathcal{QMAP} \not\subseteq \mathcal{QCMAP}$: recasting $\mathcal{QMA} \not\subseteq \mathcal{QCMA}$ [AK07]
$\mathcal{IPP} \not\subseteq \mathcal{QMAP}$: permutation testing

Techniques

$\mathcal{MAP} \not\subseteq \mathcal{QPT}$: disjointness + relaxed locally decodable code
$\mathcal{QPT} \not\subseteq \mathcal{MAP}$: Forrelation
$\mathcal{QMAP} \not\subseteq \mathcal{QCMAP}$: recasting $\mathcal{QMA} \not\subseteq \mathcal{QCMA}$ [AK07]
$\mathcal{IPP} \not\subseteq \mathcal{QMAP}$: permutation testing

Techniques

$\mathcal{MAP} \not\subseteq \mathcal{QPT}$: disjointness + relaxed locally decodable code
$\mathcal{QPT} \not\subseteq \mathcal{MAP}$: Forrelation
$\mathcal{QMAP} \not\subseteq \mathcal{QCMAP}$: recasting $\mathcal{QMA} \not\subseteq \mathcal{QCMA}$ [AK07]
$\mathcal{IPP} \not\subseteq \mathcal{QMAP}$: permutation testing

Given $f : [n] \to [n]$, is f a permutation?

Techniques

$\mathcal{MAP} \not\subseteq \mathcal{QPT}$: disjointness + relaxed locally decodable code
$\mathcal{QPT} \not\subseteq \mathcal{MAP}$: Forrelation
$\mathcal{QMAP} \not\subseteq \mathcal{QCMAP}$: recasting $\mathcal{QMA} \not\subseteq \mathcal{QCMA}$ [AK07]
$\mathcal{IPP} \not\subseteq \mathcal{QMAP}$: permutation testing

Given $f : [n] \to [n]$, is f a permutation?

- $O(1)$ with (classical) interaction
- $c \cdot q = \Omega(n^{1/3})$ with (non-interactive) quantum proof

Next goal

Theorem ([RVW13], [RR20])

Every language in logspace-uniform NC admits a doubly-efficient IPP with communication and query complexities $\tilde{O}(\sqrt{n})$.

Next goal

Theorem ([RVW13], [RR20])

Every language in logspace-uniform NC admits a doubly-efficient IPP with communication and query complexities $\tilde{O}(\sqrt{n})$. (Prover runtime $\mathrm{poly}(n)$, verifier runtime $\tilde{O}(\sqrt{n})$.)

Next goal

Theorem ([RVW13], [RR20])

Every language in logspace-uniform NC admits a doubly-efficient IPP with communication and query complexities $\tilde{O}(\sqrt{n})$. (Prover runtime $\mathrm{poly}(n)$, verifier runtime $\tilde{O}(\sqrt{n})$.)

Theorem (Hopefully!)

QIPPs with complexities $O(n^{\alpha})$ for some $\alpha < 1/2$.

References

Scott Aaronson and Greg Kuperberg.
Quantum versus classical proofs and advice.
In *22nd Annual IEEE Conference on Computational Complexity (CCC 2007), 13-16 June 2007, San Diego, California, USA*, pages 115–128. IEEE Computer Society, 2007.

Guy N. Rothblum and Ron Rothblum.
Batch verification and proofs of proximity with polylog overhead.
Electron. Colloquium Comput. Complex., 27:157, 2020.

Guy N Rothblum, Salil Vadhan, and Avi Wigderson.
Interactive proofs of proximity: delegating computation in sublinear time.
In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 793–802, 2013.

Images:

Server Icon by Rank Sol on Iconscout
Mobile by Momento Design from the Noun Project
Laptop Icon by Jemis Mali from Iconscout
Smartwatch by juan manjarrez from the Noun Project
database by mardjoe from the Noun Project
atom by Fengquan Li from the Noun Project
`https://disney.fandom.com/wiki/Alice/Gallery?file=Alice_Render.png`
`https://loathsomecharacters.miraheze.org/wiki/File:SpongeBob_SquarePants.png`