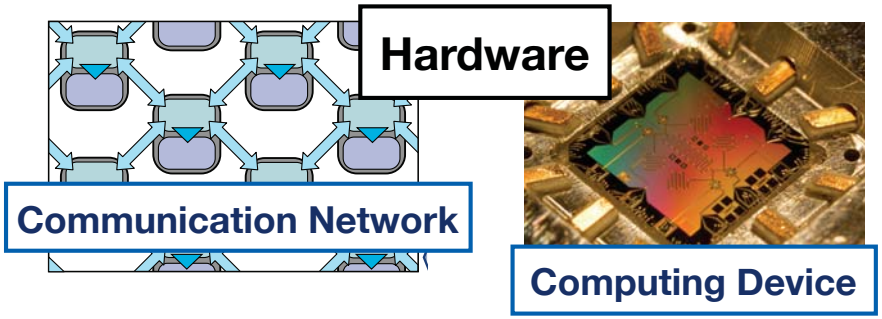
Verification of Quantum Computing

Elham Kashefi

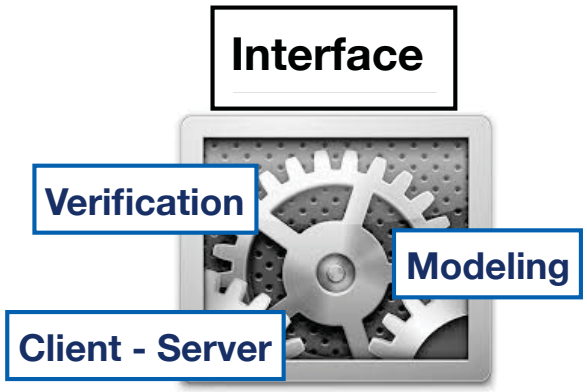
University of Edinburgh
Oxford Quantum Technology Hub
Paris Centre for Quantum Computing
Laboratoire traitement et communication de l'information

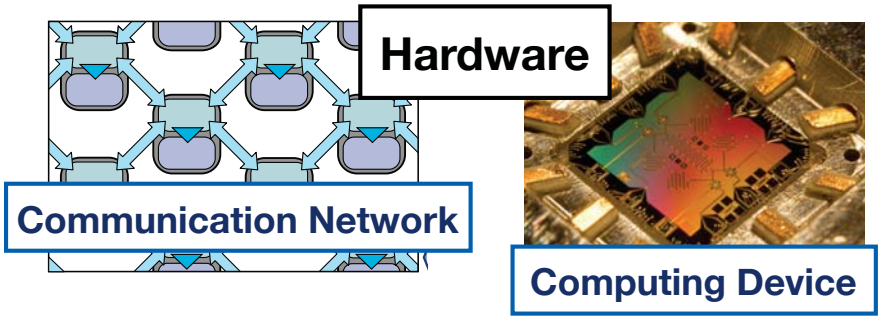
Our Vision of Quantum Technology

Our Vision of Quantum Technology

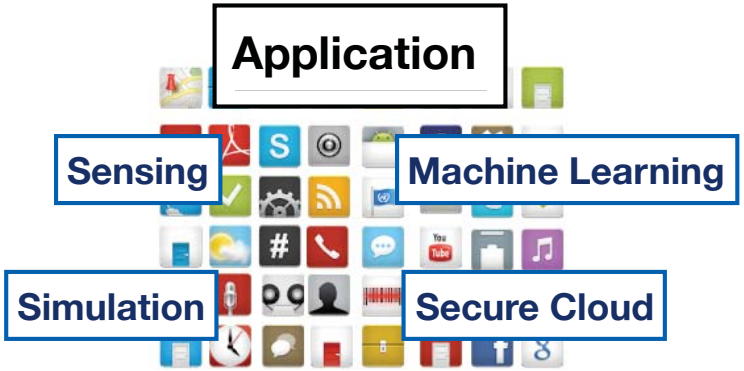


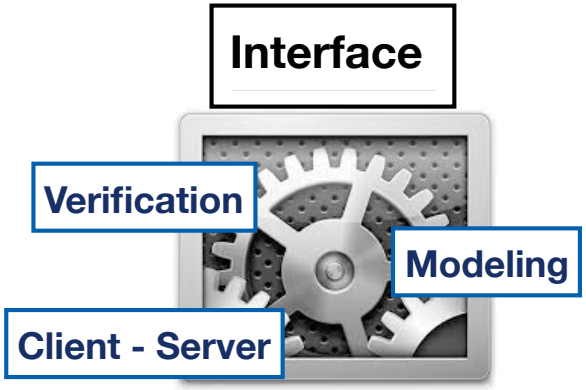
Our Vision of Quantum Technology

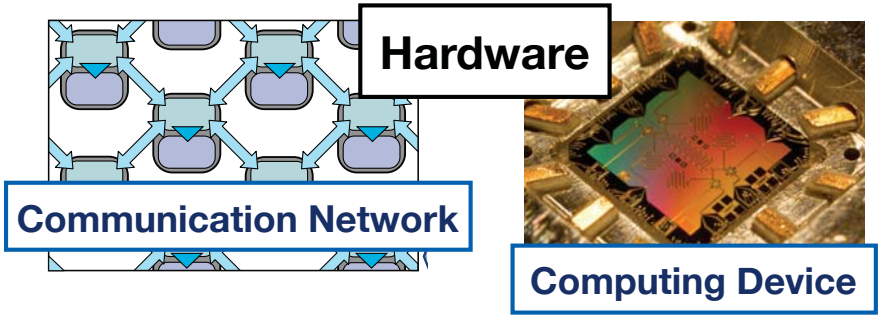




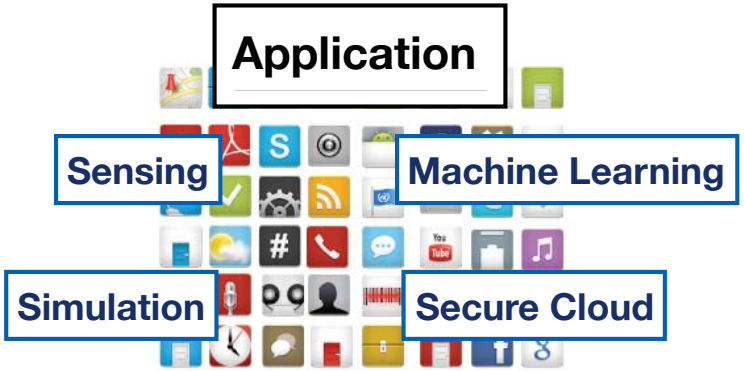
Our Vision of Quantum Technology

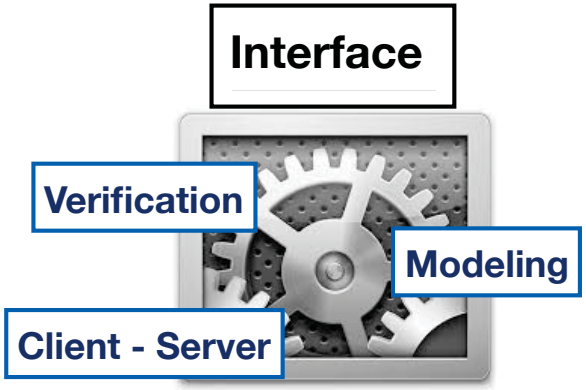




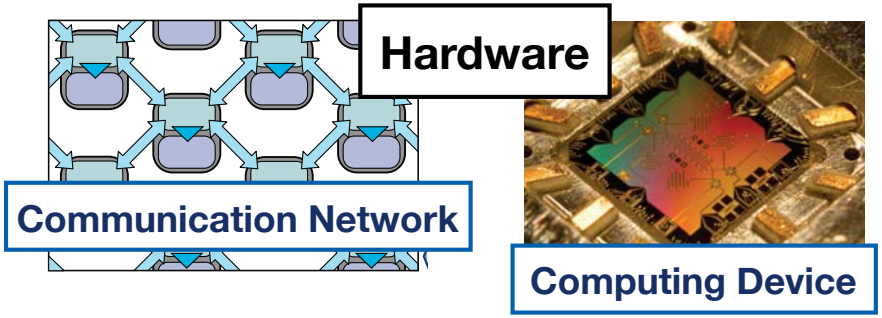


Our Vision of Quantum Technology

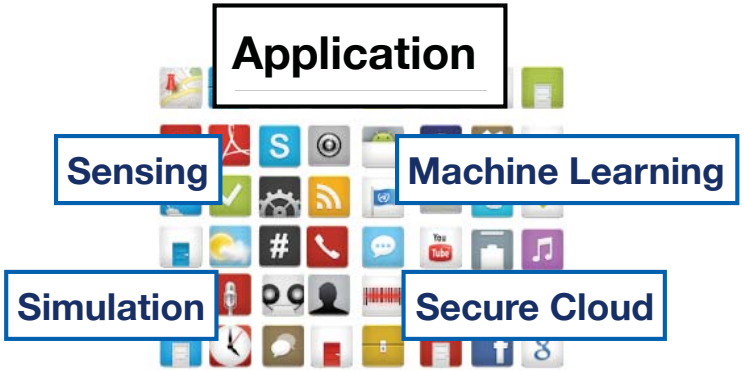


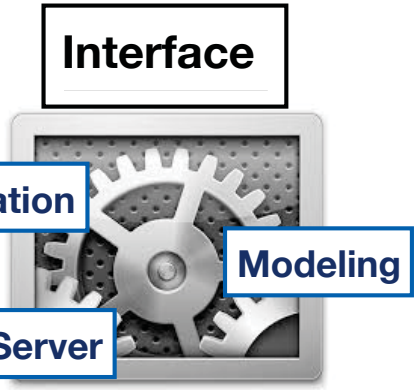


EK *et.al.* *Measurement Calculus*
ICALP 2006, J. ACM 2007
Phys. Rev. A. 2005 New J. Phys. 2007



Our Vision of Quantum Technology

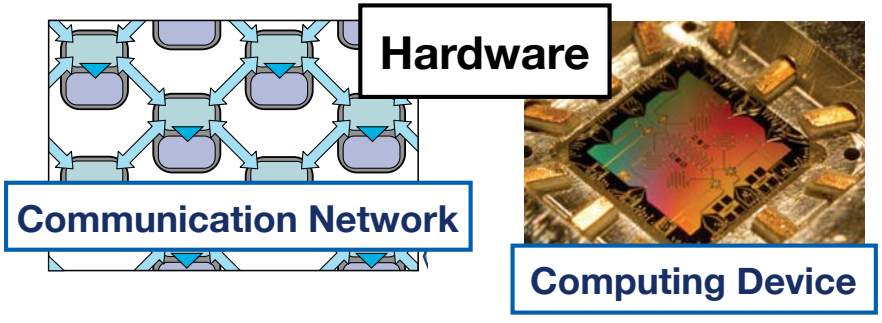




***EK** et.al. Verification*
New J. Phys. 2015, Qcrypt, 2015

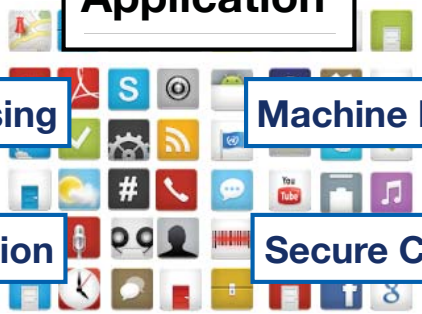
***EK** et.al. Blind QC*
FOCS, 2009 Phys. Rev. Lett. 12, TQC 14

***EK** et.al. Measurement Calculus*
ICALP 2006, J. ACM 2007
Phys. Rev. A. 2005 New J. Phys. 2007



Our Vision of Quantum Technology

Application



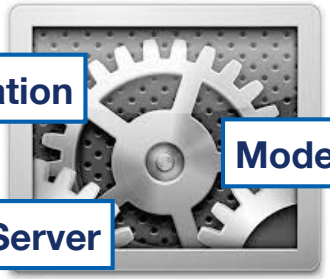
Sensing

Machine Learning

Simulation

Secure Cloud

Interface



Verification

Modeling

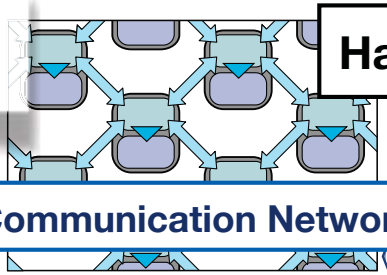
Client - Server

***EK** et.al. Verification*
New J. Phys. 2015, Qcrypt, 2015

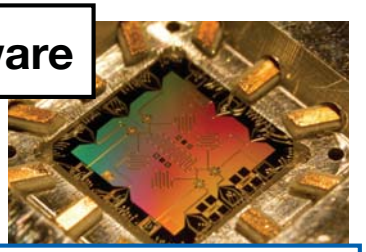
***EK** et.al. Blind QC*
FOCS, 2009 Phys. Rev. Lett. 12, TQC 14

***EK** et.al. Measurement Calculus*
ICALP 2006, J. ACM 2007
Phys. Rev. A. 2005 New J. Phys. 2007

Hardware



Communication Network



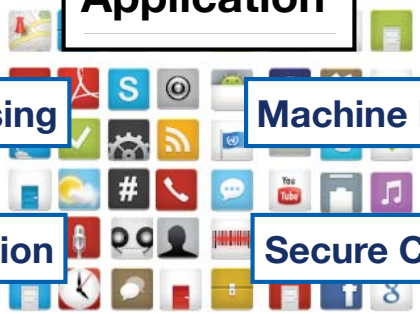
Computing Device

***EK** et.al. Verification Experiment*
Nature Physics 13

***EK** et.al. Blind QC Experiment*
Science 12

Our Vision of Quantum Technology

Application



Sensing

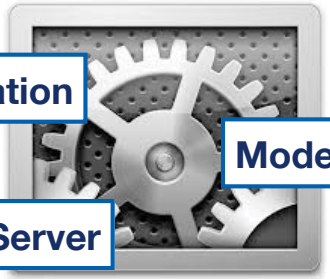
Machine Learning

Simulation

Secure Cloud

***EK** et.al. Quantum-enhanced Cloud*
J. QIC 2015, Phys. Rev. A 2016, Patent

Interface



Verification

Modeling

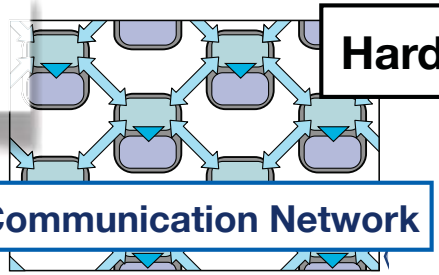
Client - Server

***EK** et.al. Verification*
New J. Phys. 2015, Qcrypt, 2015

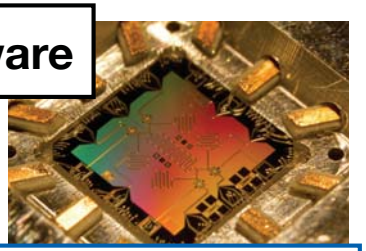
***EK** et.al. Measurement Calculus*
ICALP 2006, J. ACM 2007
Phys. Rev. A. 2005 New J. Phys. 2007

***EK** et.al. Blind QC*
FOCS, 2009 Phys. Rev. Lett. 12, TQC 14

Hardware



Communication Network



Computing Device

***EK** et.al. Verification Experiment*
Nature Physics 13

***EK** et.al. Blind QC Experiment*
Science 12

My Vision of Quantum Technology

Application

Sensing

Machine Learning

Simulation

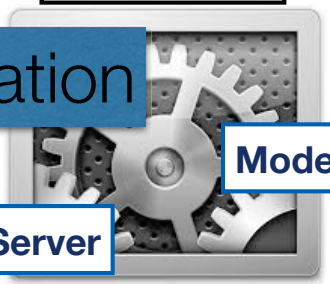
Secure Cloud

K et.al. Quantum-enhanced Cloud
. QIC 2015, Phys. Rev. A 2016, Patent

Interface

EK et.al. Verification
New J. Phys. 2015, Qcrypt, 2015

Verification



Modeling

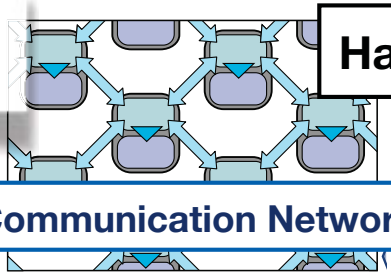
EK et.al. Measurement Calculus
ICALP 2006, J. ACM 2007
Phys. Rev. A. 2005 New J. Phys. 2007

EK et.al. Blind QC
FOCS, 2009 Phys. Rev. Lett. 12, TQC 14

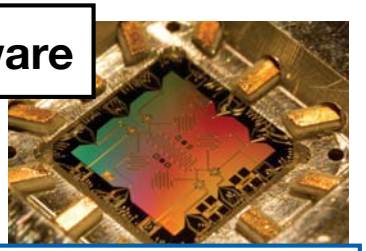
Client - Server

Hardware

EK et.al. Verification Experiment
Nature Physics 13



Communication Network



Computing Device

EK et.al. Blind QC Experiment
Science 12

Quantum Era

National Investments

Europe 1bn€
UK 270M £
Netherlands 80M $
US, Singapore,Canada

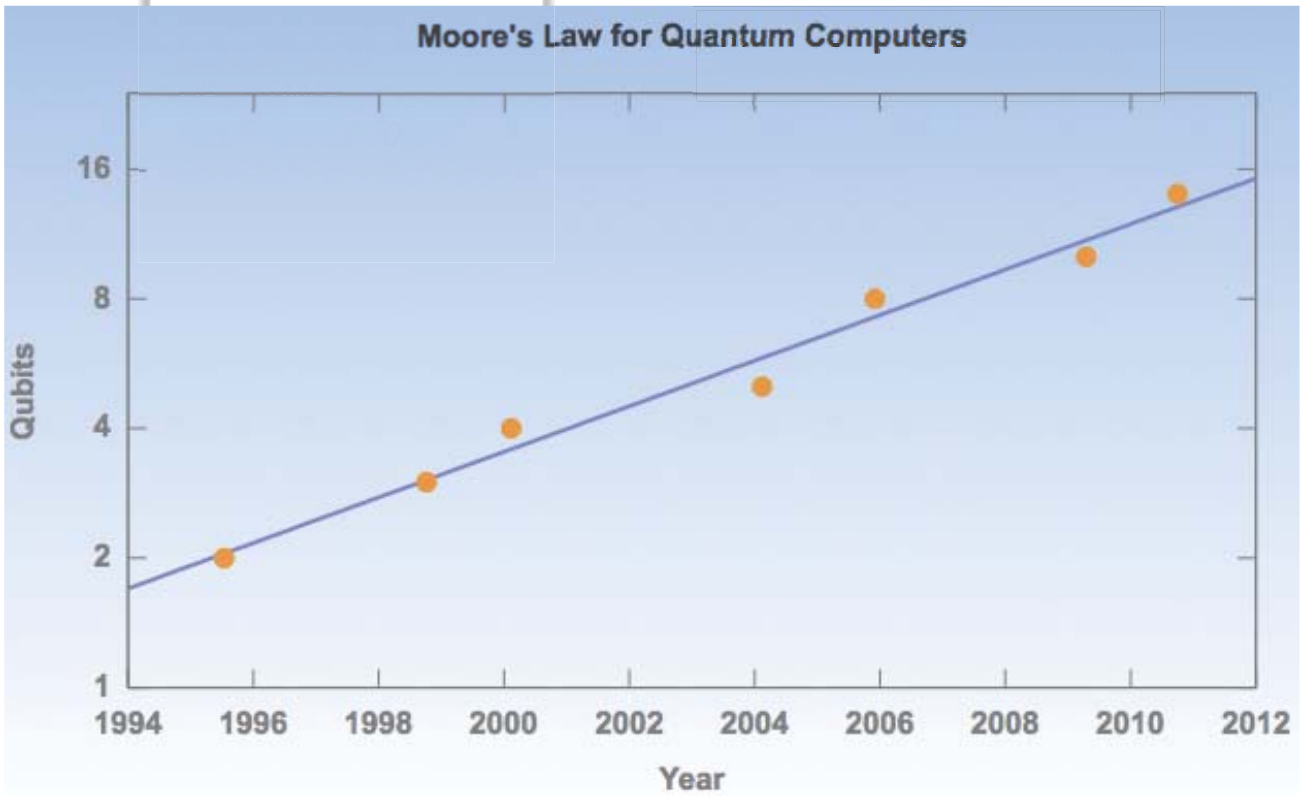
Quantum Machines

Private Investments

Google, IBM, Intel
Big VC founds
Startups Companies: D-Wave

Quantum Era

National Investments

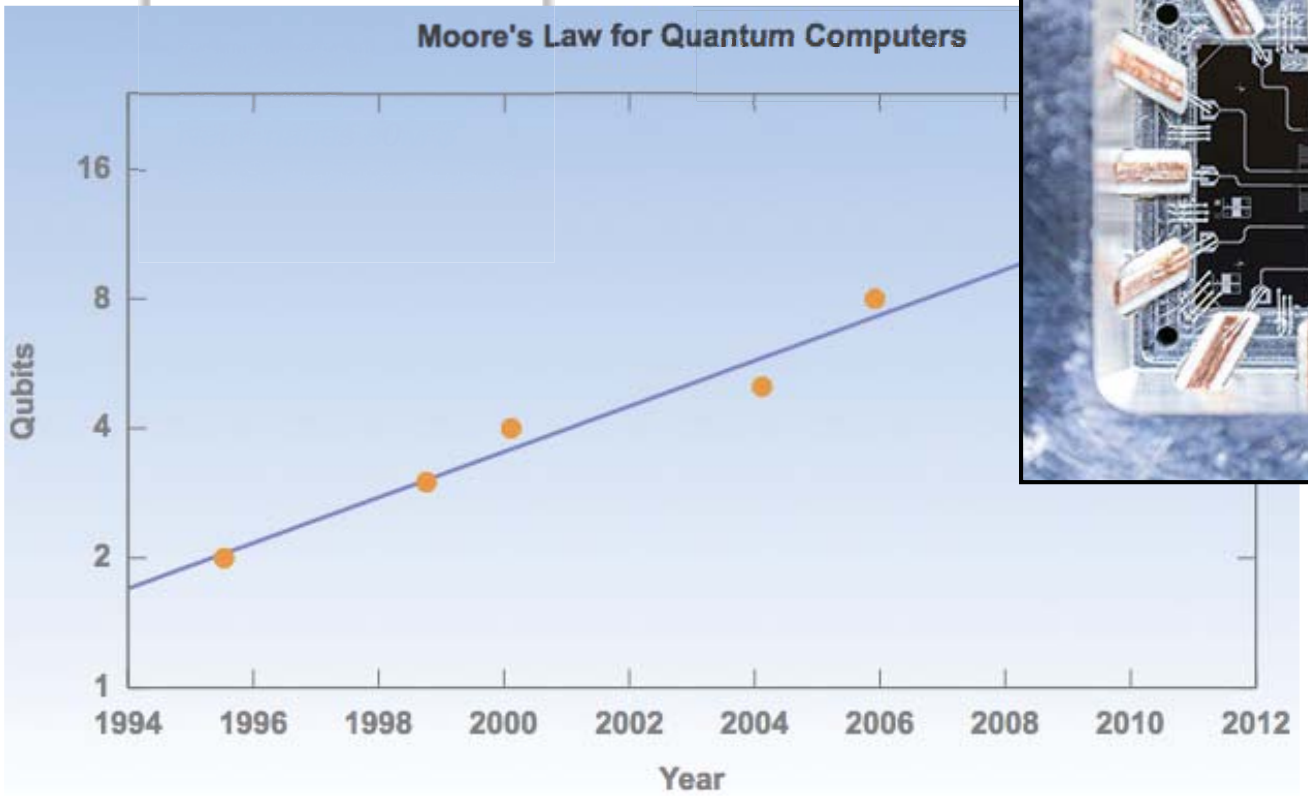


Private Investments

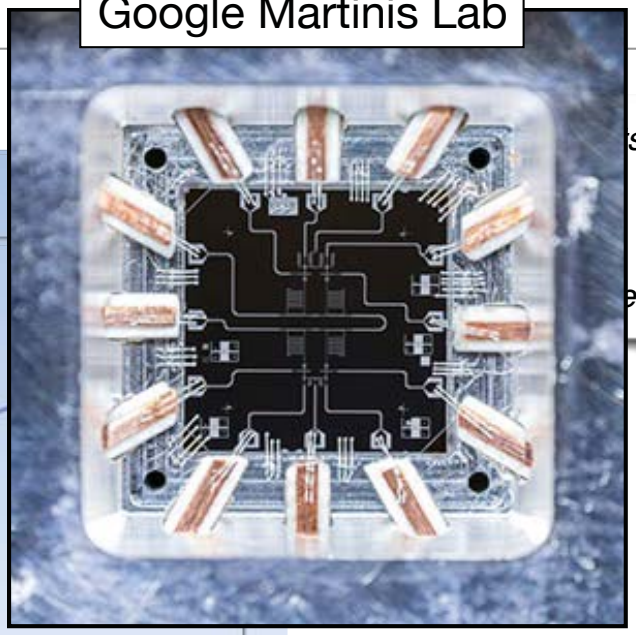
Google, IBM, Intel
Big VC founds
Startups Companies: D-Wave

Quantum Era

National Investments



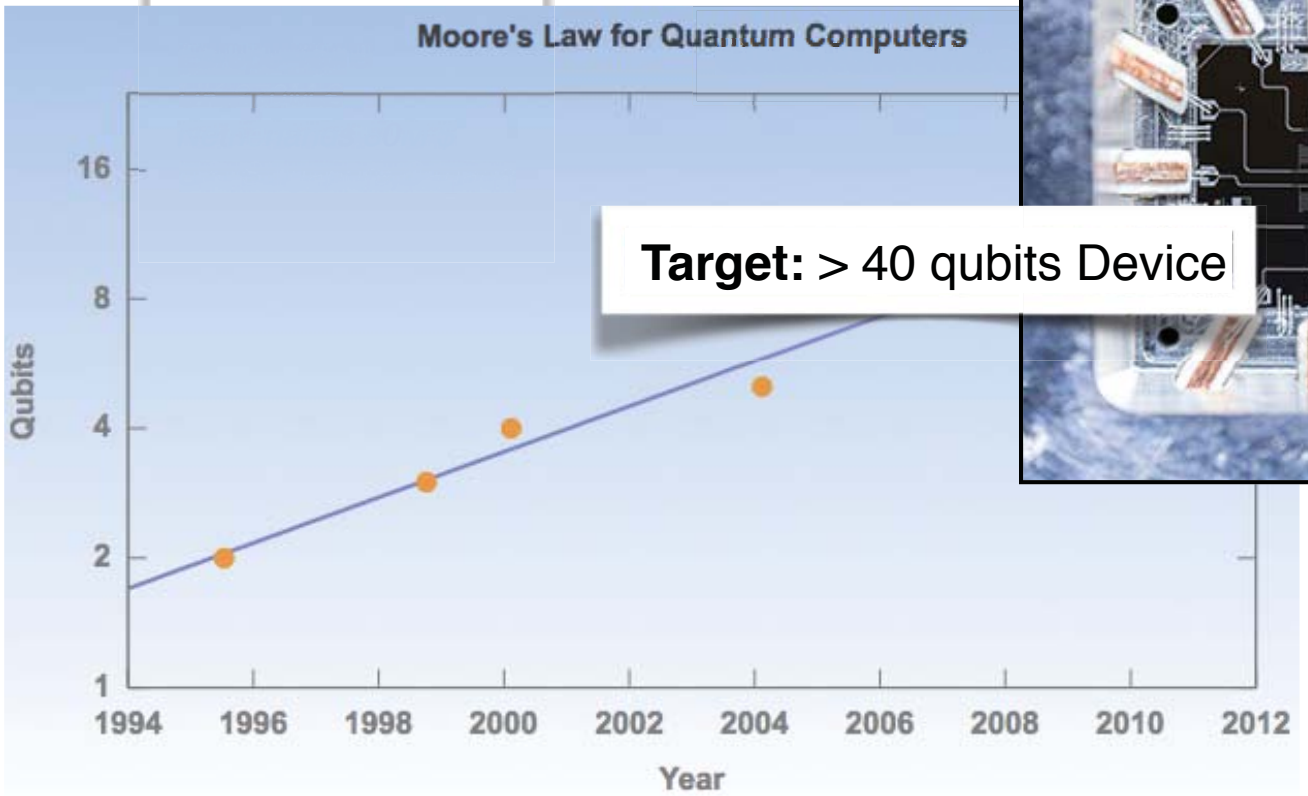
Google Martinis Lab



es: D-Wave

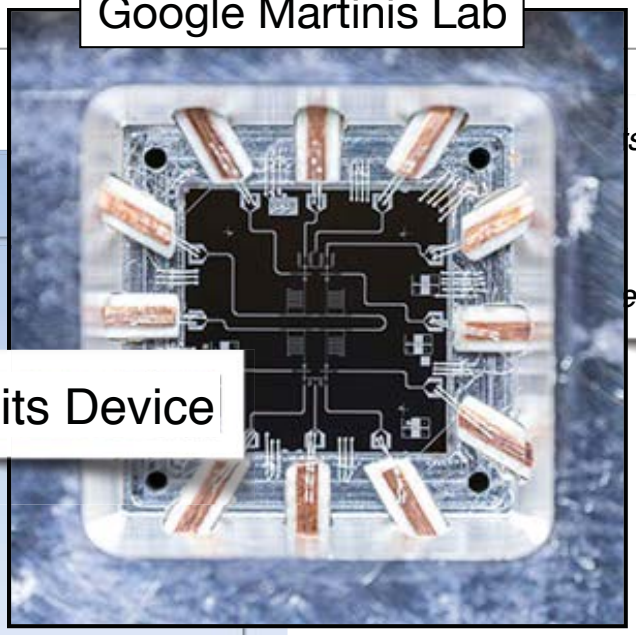
Quantum Era

National Investments



Target: > 40 qubits Device

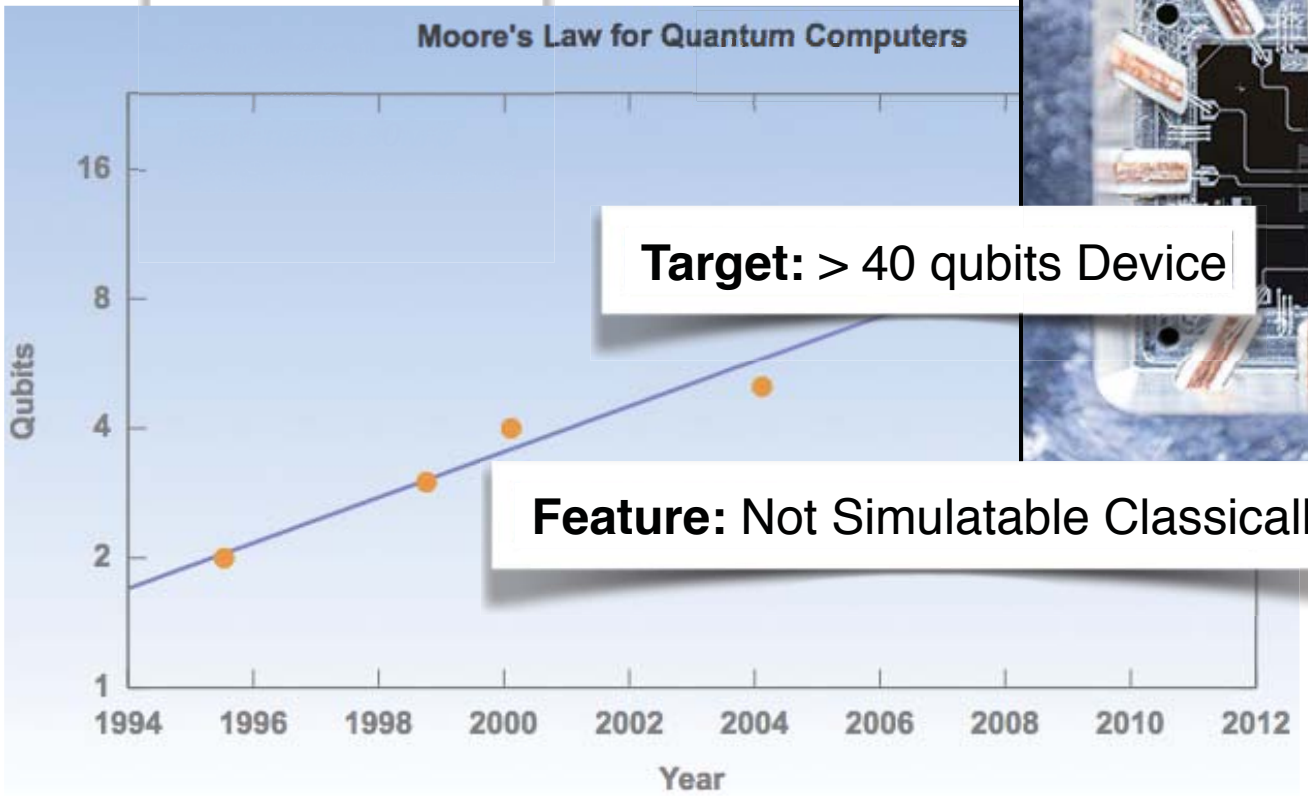
Google Martinis Lab



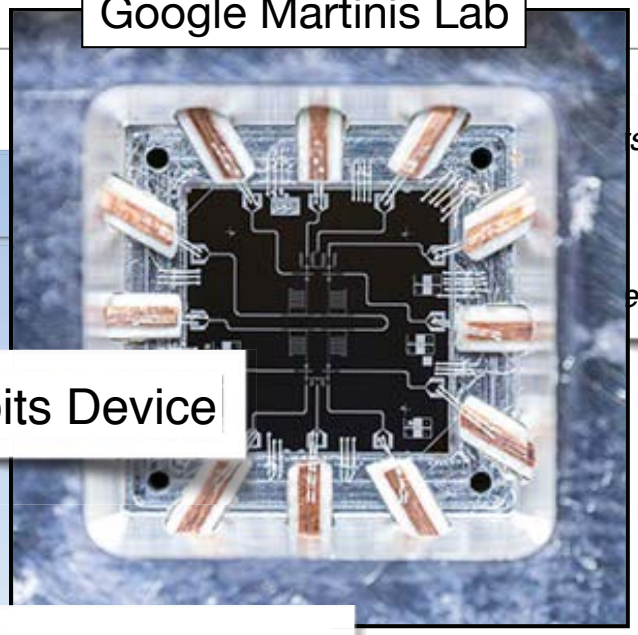
es: D-Wave

Quantum Era

National Investments



Google Martinis Lab



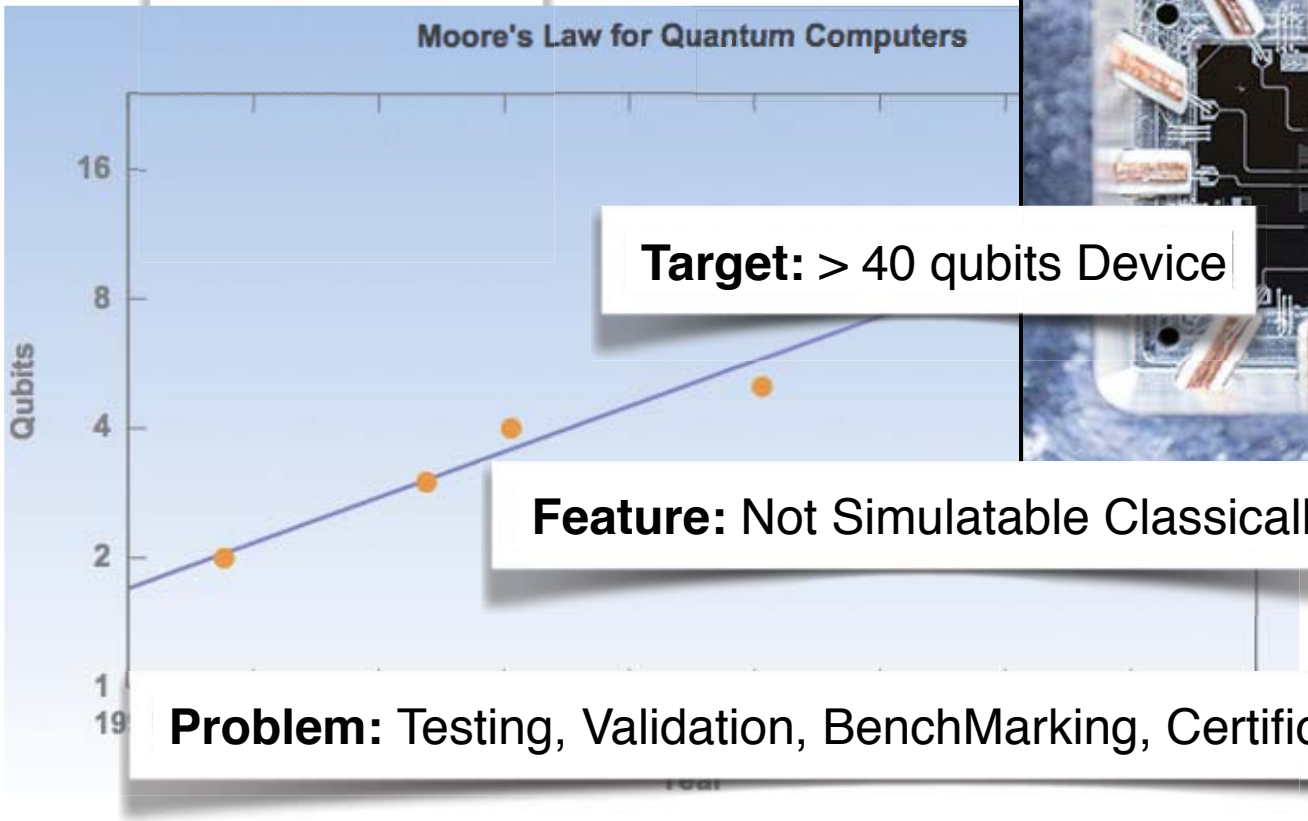
es: D-Wave

Target: > 40 qubits Device

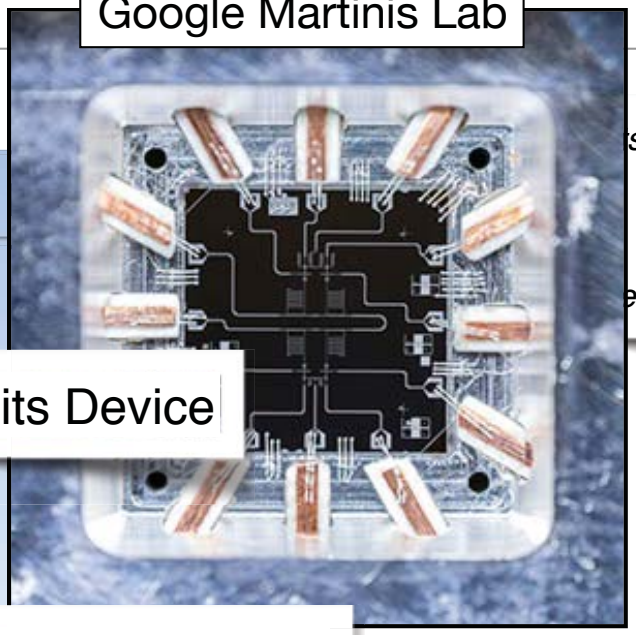
Feature: Not Simulatable Classically

Quantum Era

National Investments



Google Martinis Lab



es: D-Wave

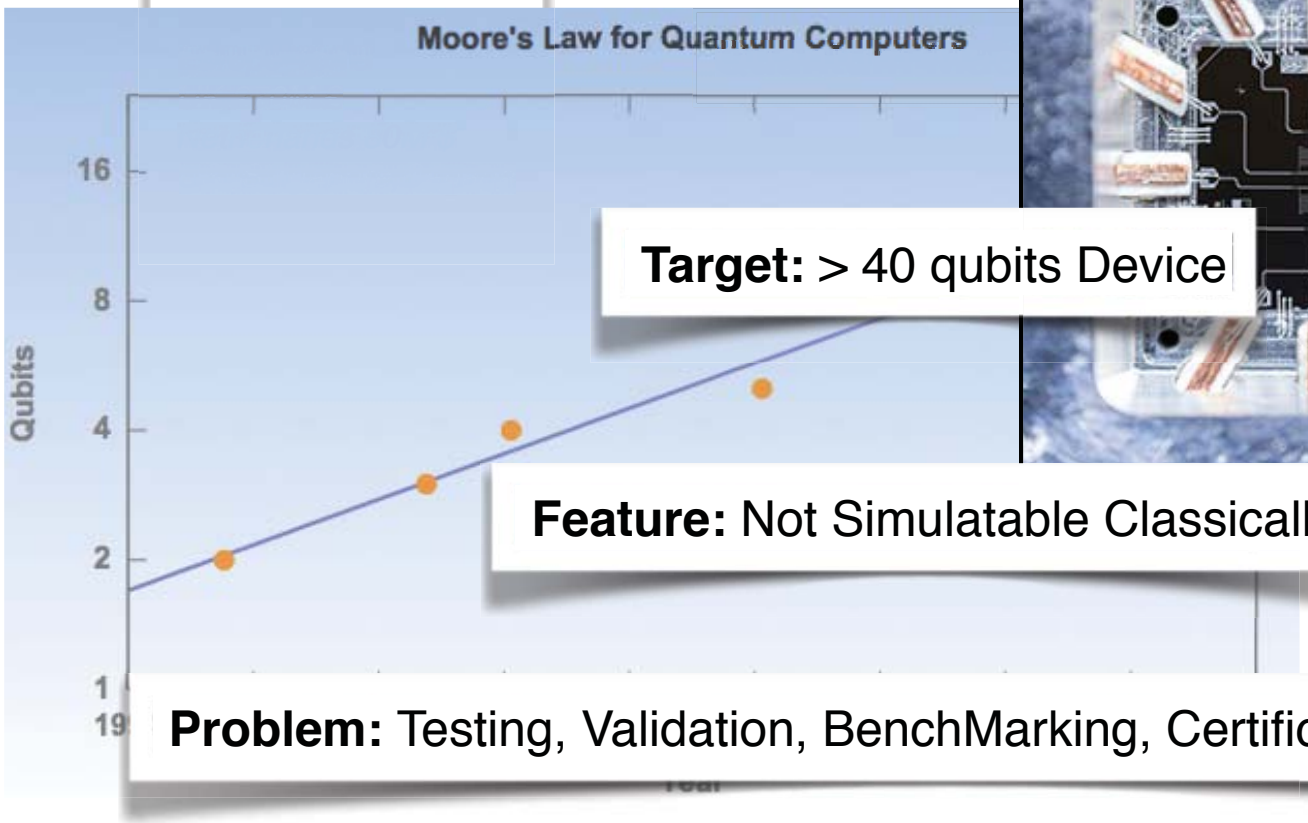
Target: > 40 qubits Device

Feature: Not Simulatable Classically

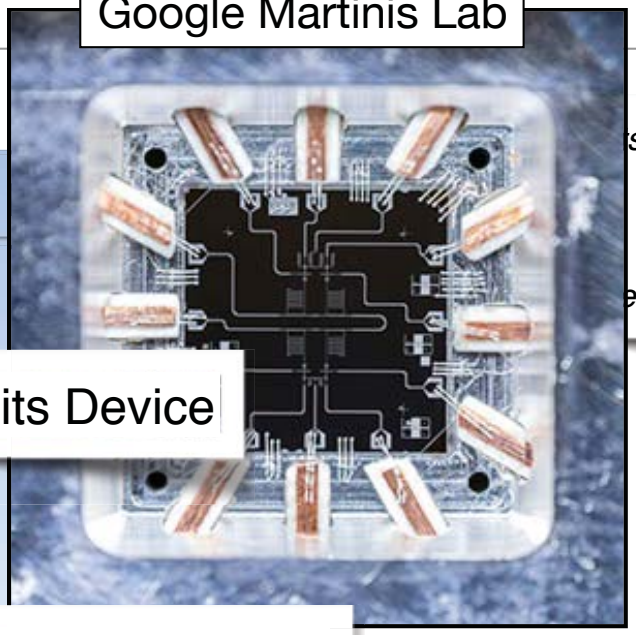
Problem: Testing, Validation, BenchMarking, Certification, Verification …

Quantum Era

National Investments



Google Martinis Lab



es: D-Wave

Target: > 40 qubits Device

Feature: Not Simulatable Classically

Problem: Testing, Validation, BenchMarking, Certification, Verification …

Do we need another quantum computer to testing a quantum computer or
can we bootstrap a mini quantum device to test a bigger one

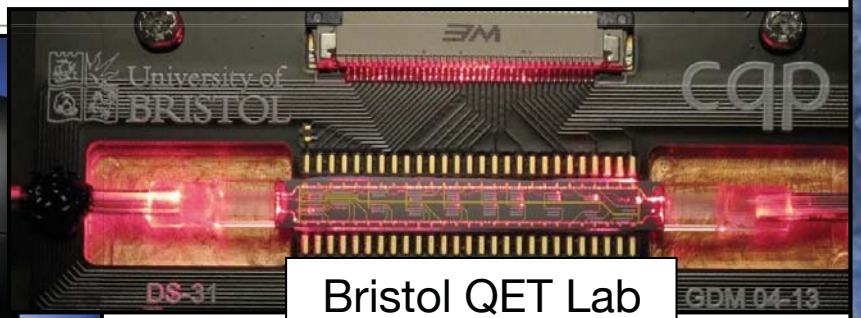
Question

Quantum Machines Era

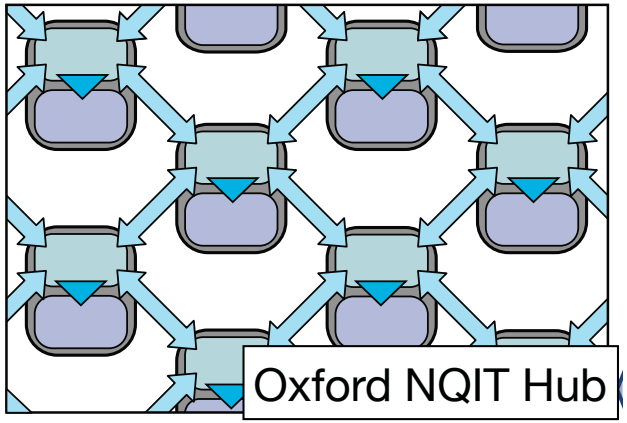
Question

Question

Lockheed Martin/NASA/Google
Artificial Intelligence lab

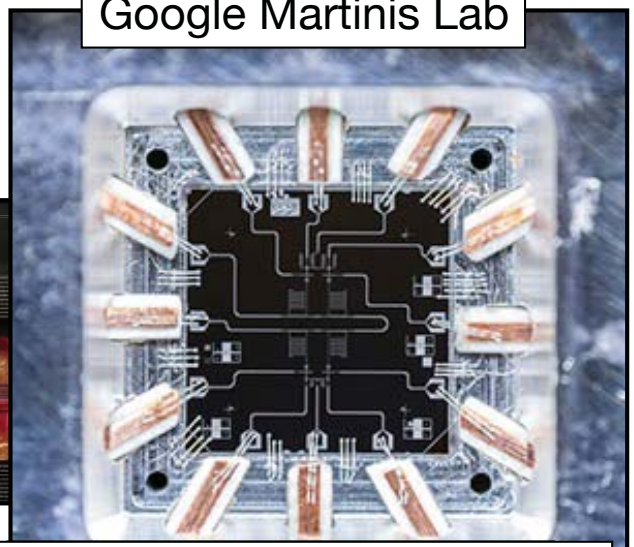


Bristol QET Lab



Oxford NQIT Hub

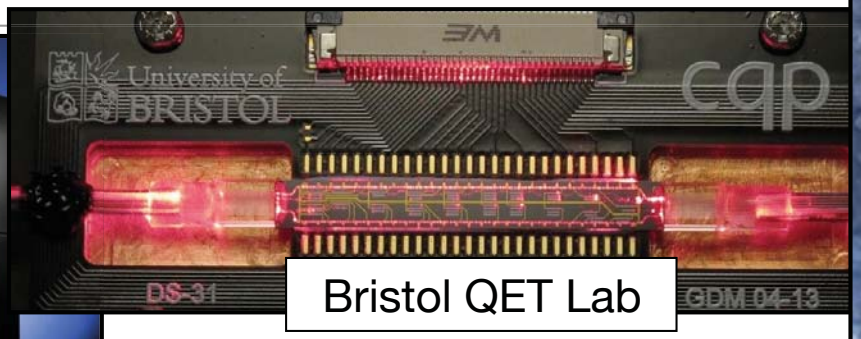
Google Martinis Lab



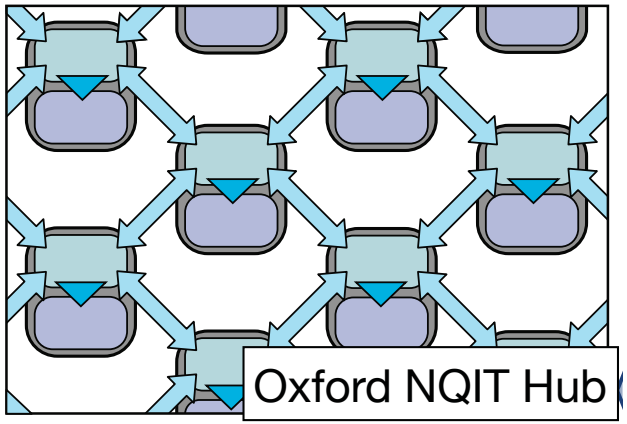
TU Delft Quantum Tech Lab

Question

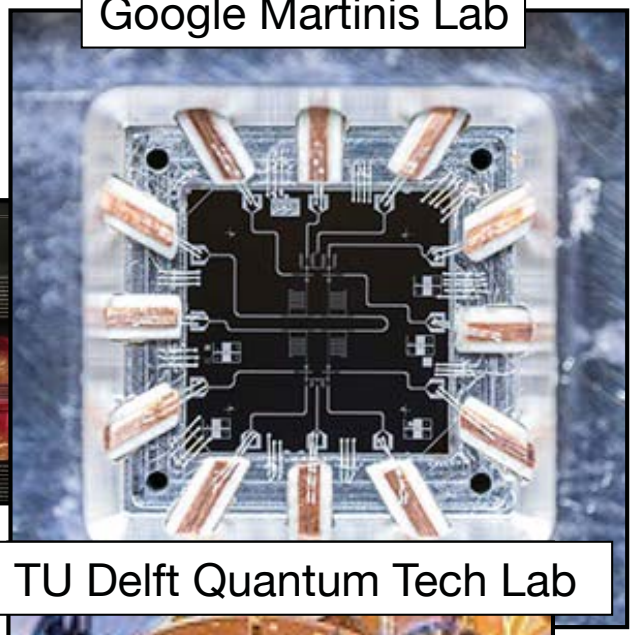
Lockheed Martin/NASA/Google
Artificial Intelligence lab



Bristol QET Lab



Oxford NQIT Hub



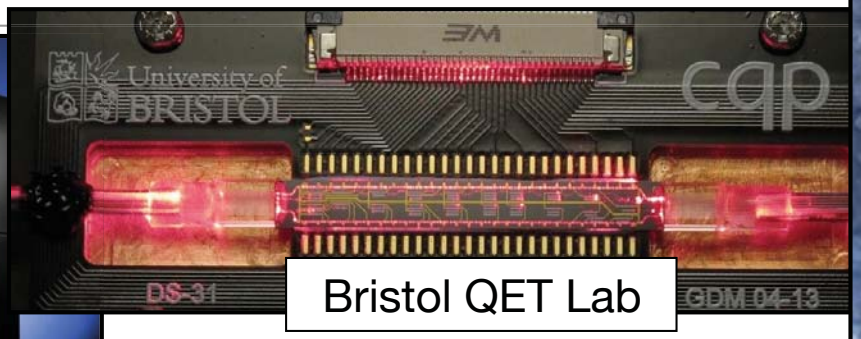
Google Martinis Lab

TU Delft Quantum Tech Lab

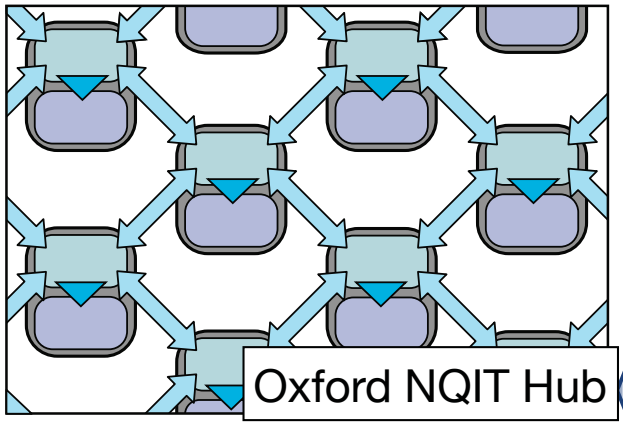
These devices become **relevant** at the moment they are no longer classically simulatable

Question

Lockheed Martin/NASA/Google
Artificial Intelligence lab

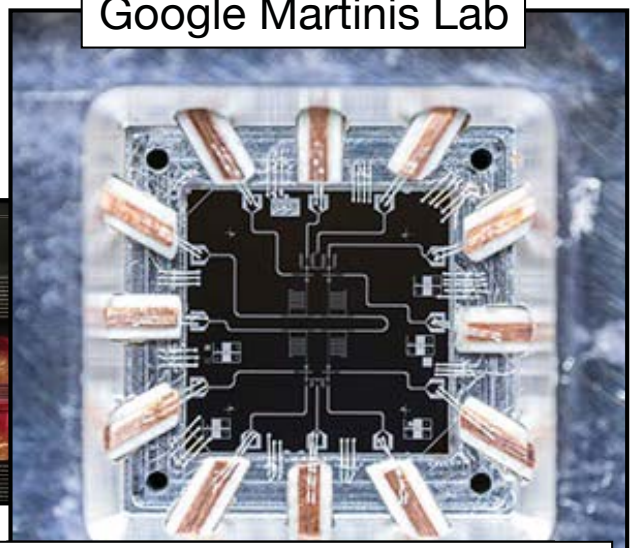


Bristol QET Lab



Oxford NQIT Hub

Google Martinis Lab



TU Delft Quantum Tech Lab

These devices become **relevant** at the moment they are no longer classically simulatable

Existing methods of Testing/Validation/Simulation/Monitoring/Tomography ... all become **IRRELEVANT**

Target

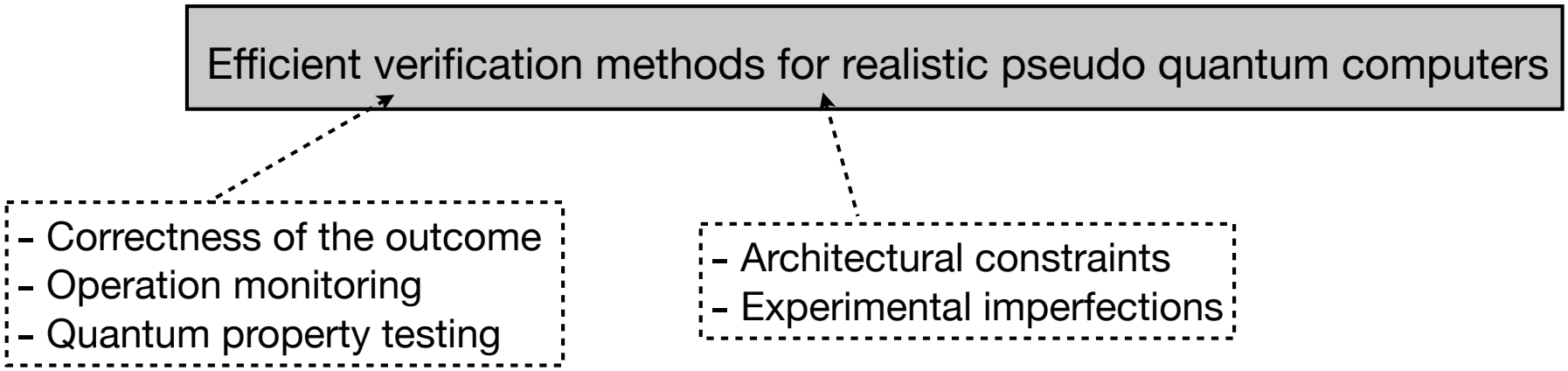
Efficient verification methods for realistic pseudo quantum computers

Target

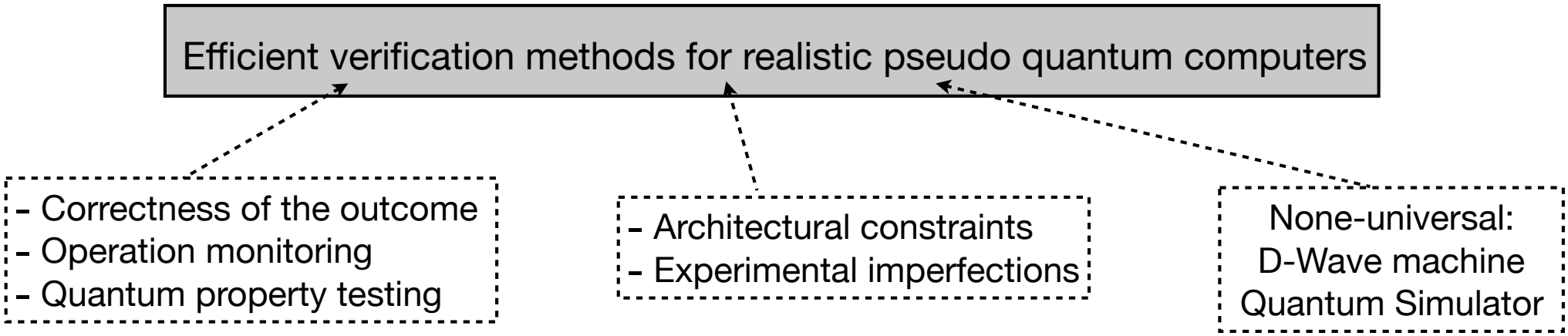
Efficient verification methods for realistic pseudo quantum computers

- Correctness of the outcome
- Operation monitoring
- Quantum property testing

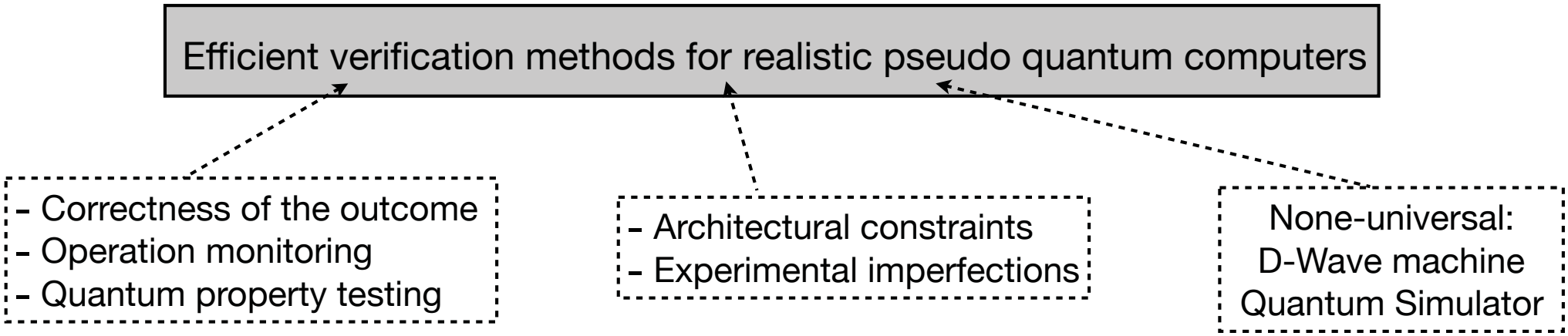
Target


Efficient verification methods for realistic pseudo quantum computers
- Correctness of the outcome
- Operation monitoring
- Quantum property testing
- Architectural constraints
- Experimental imperfections

Target


Efficient verification methods for realistic pseudo quantum computers
- Correctness of the outcome
- Operation monitoring
- Quantum property testing
- Architectural constraints
- Experimental imperfections
None-universal:
D-Wave machine
Quantum Simulator

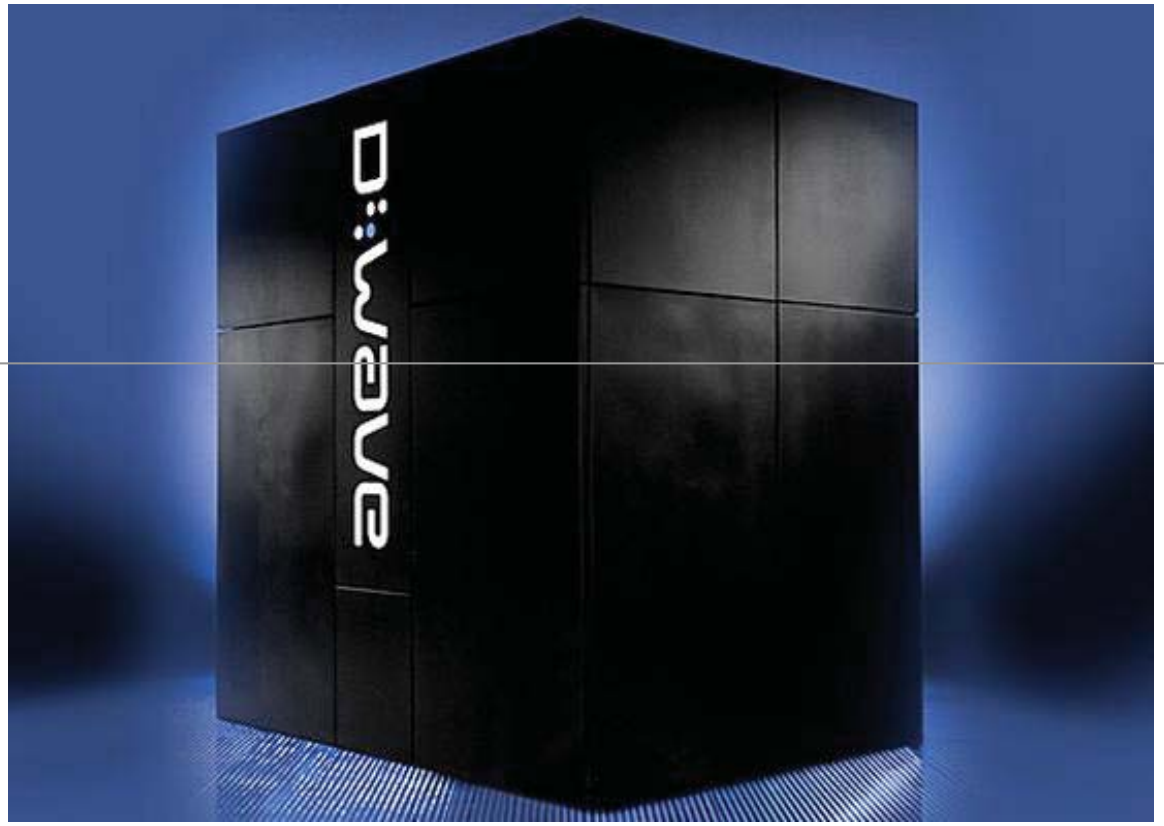
Target



Goal

Over the next decades, as quantum technology matures globally
we will provide **testing criteria** for industry adaptation

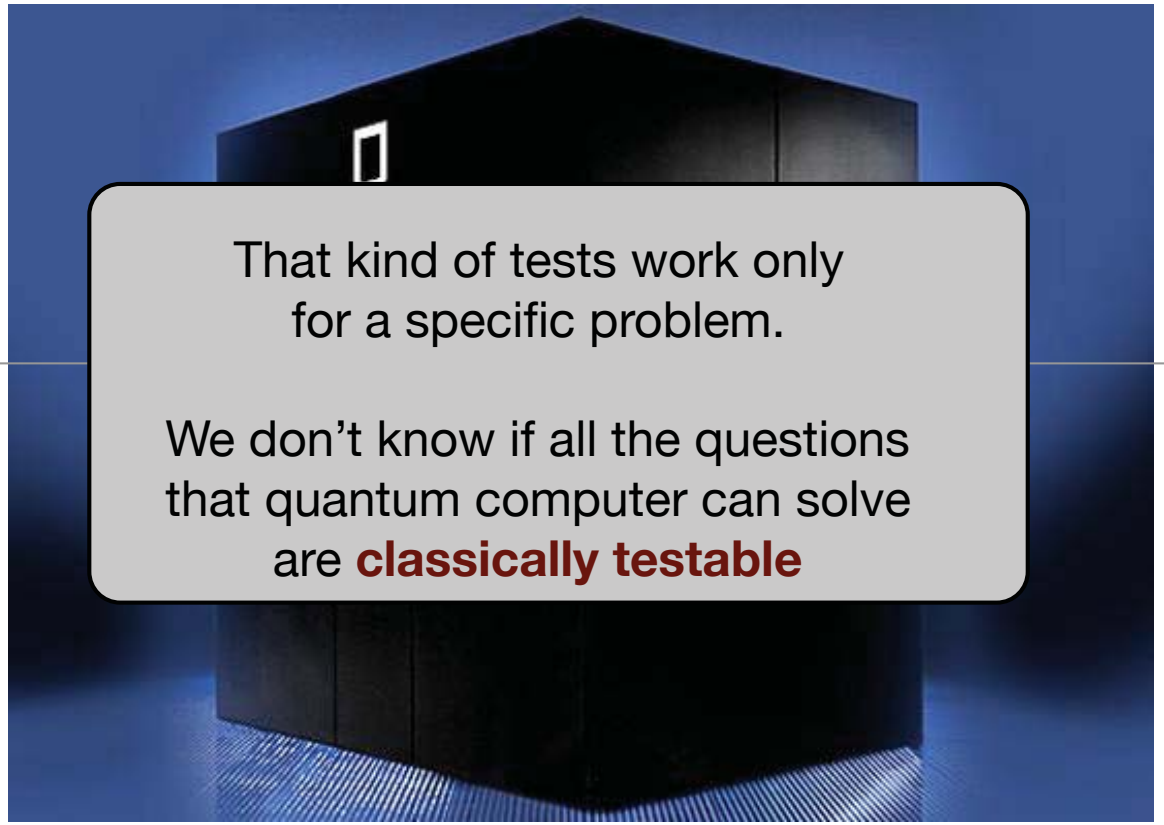
Should we pay $10000000 for a quantum computer



Should we pay $10000000 for a quantum computer

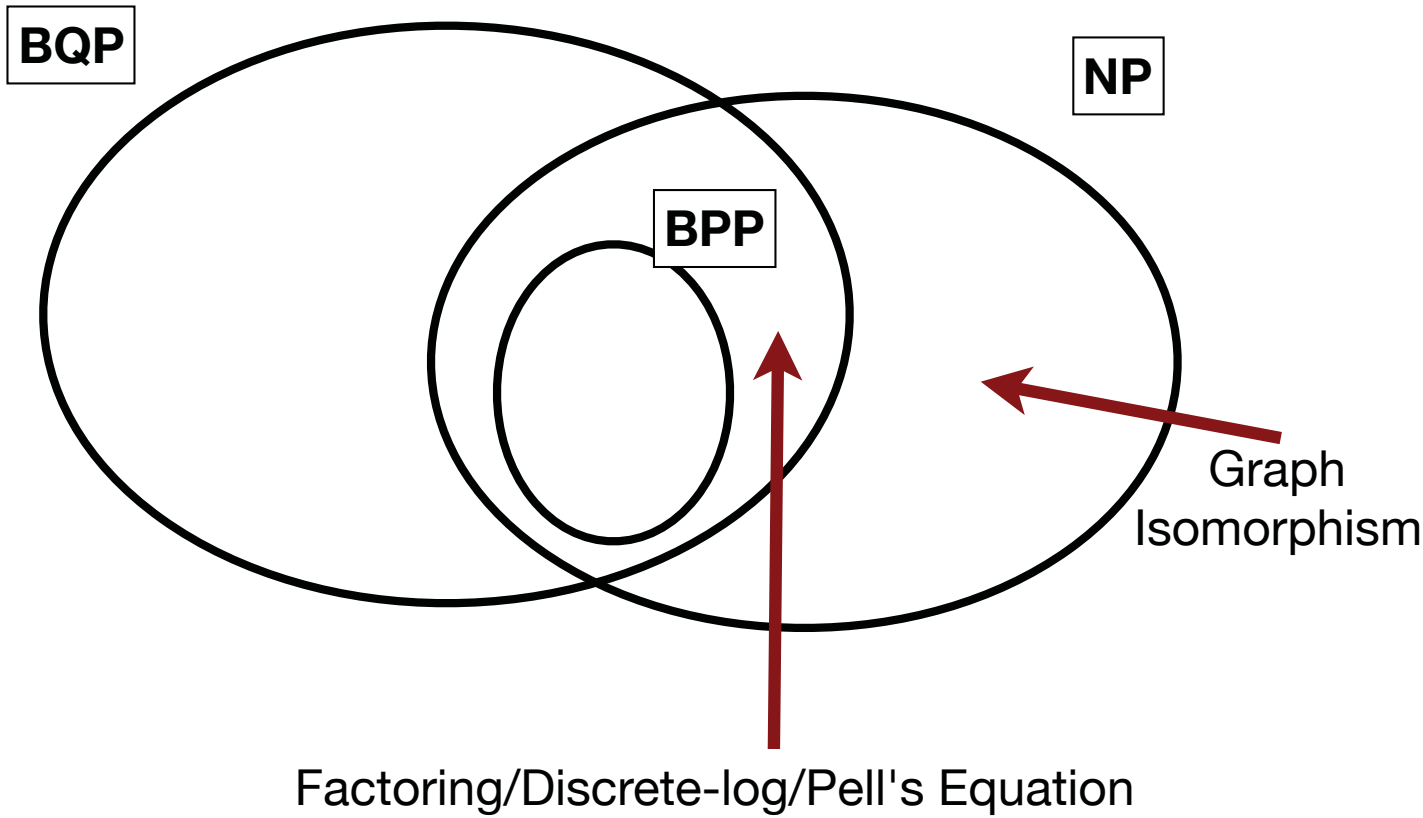
Simple test: We ask the box to factor a big number

Should we pay $10000000 for a quantum computer

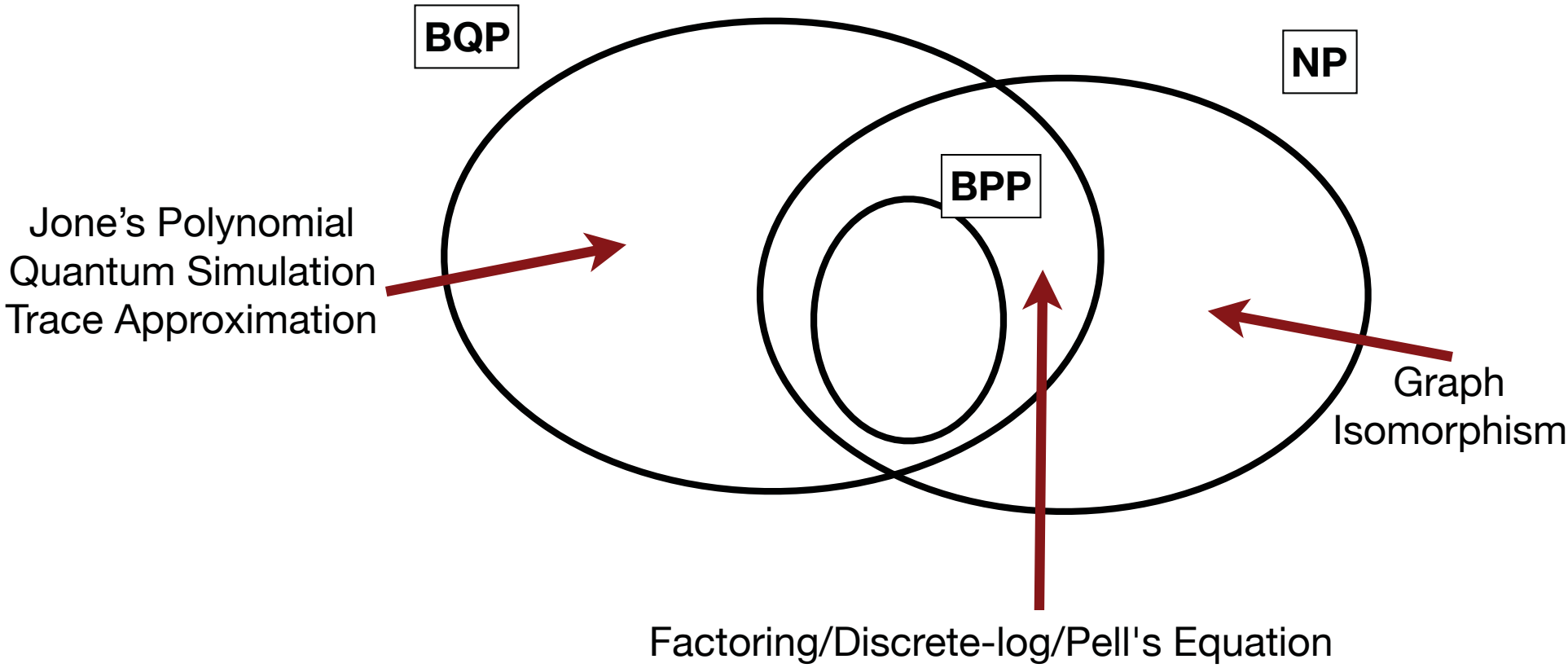


Simple test: We ask the box to factor a big number

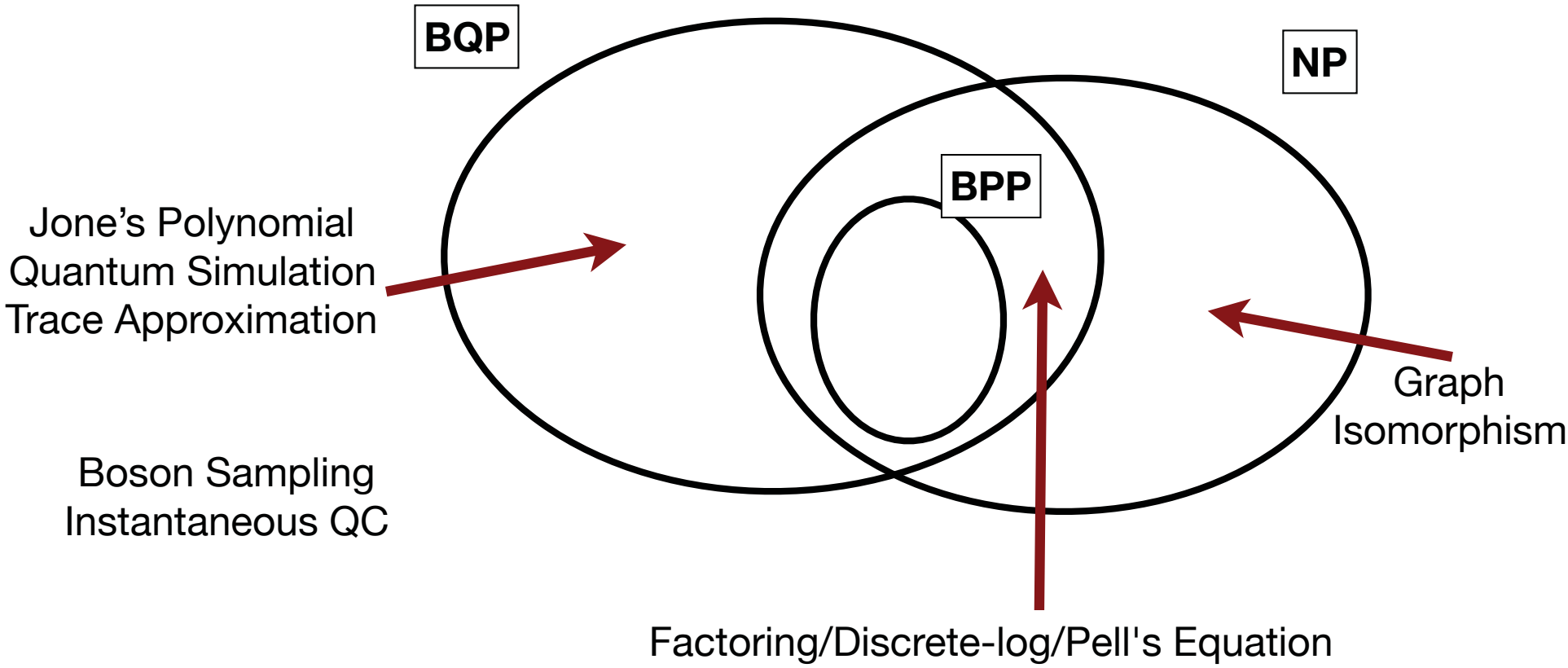
Complexity Picture



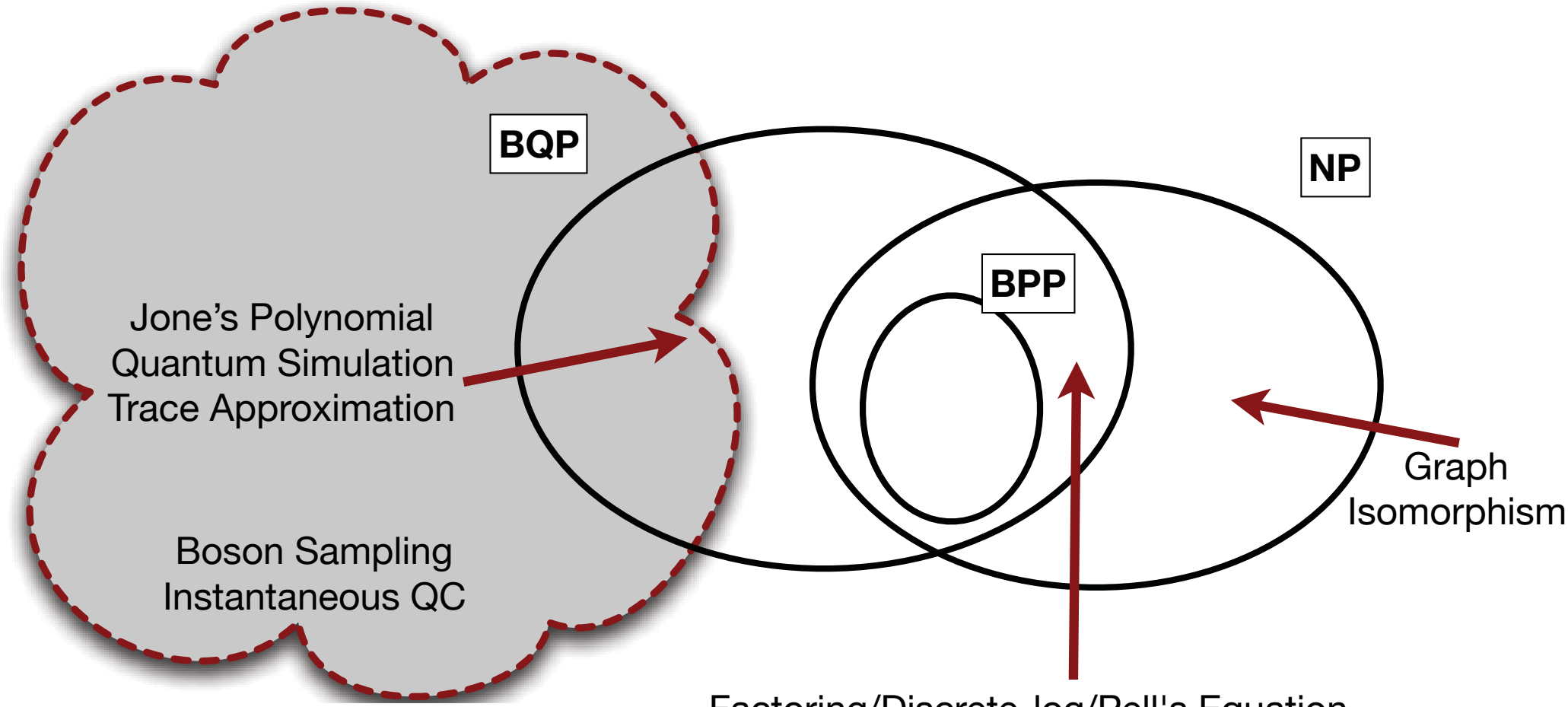
Complexity Picture



Complexity Picture

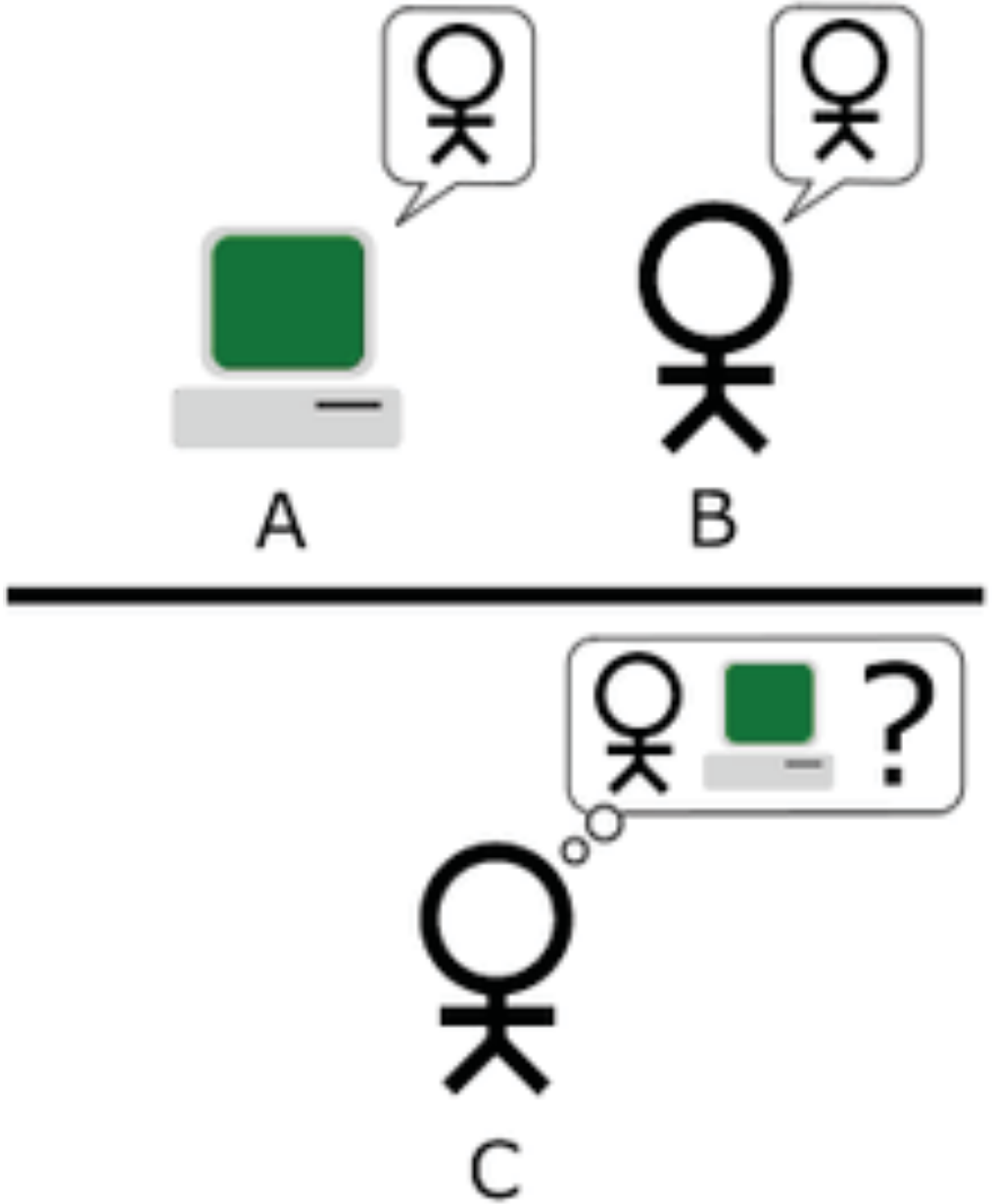


Complexity Picture

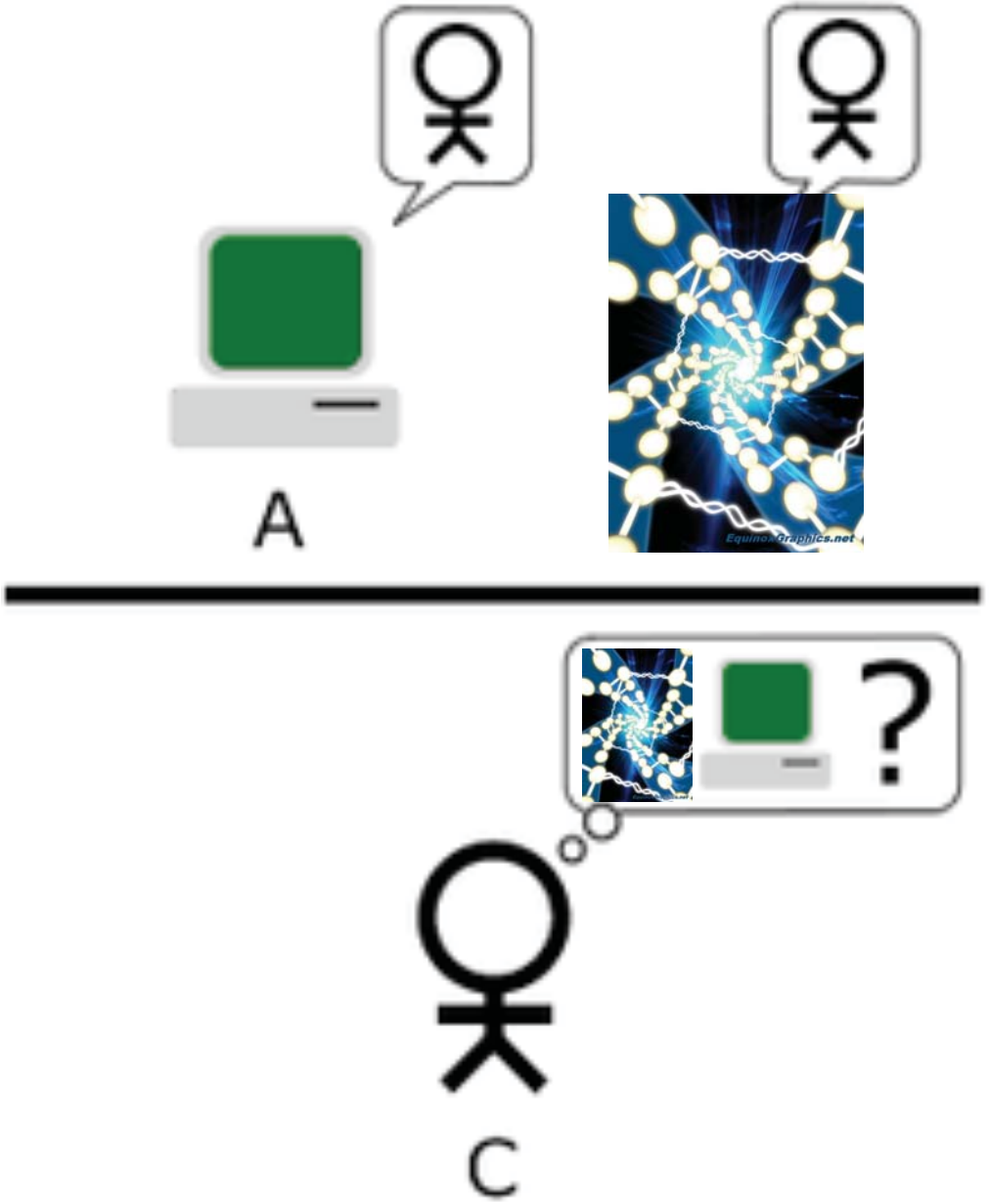


Testing outcome correctness ?

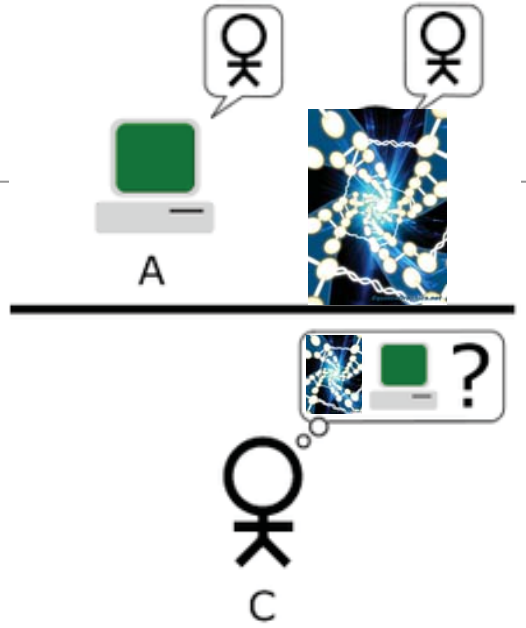
Quantum Turing Test



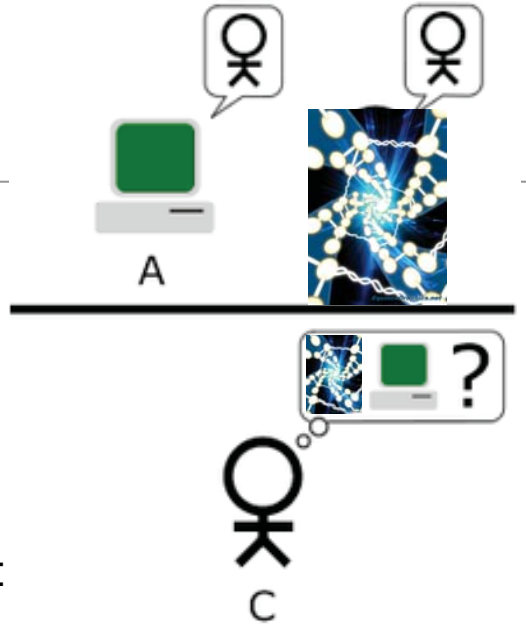
Quantum Turing Test



Quantum Turing test



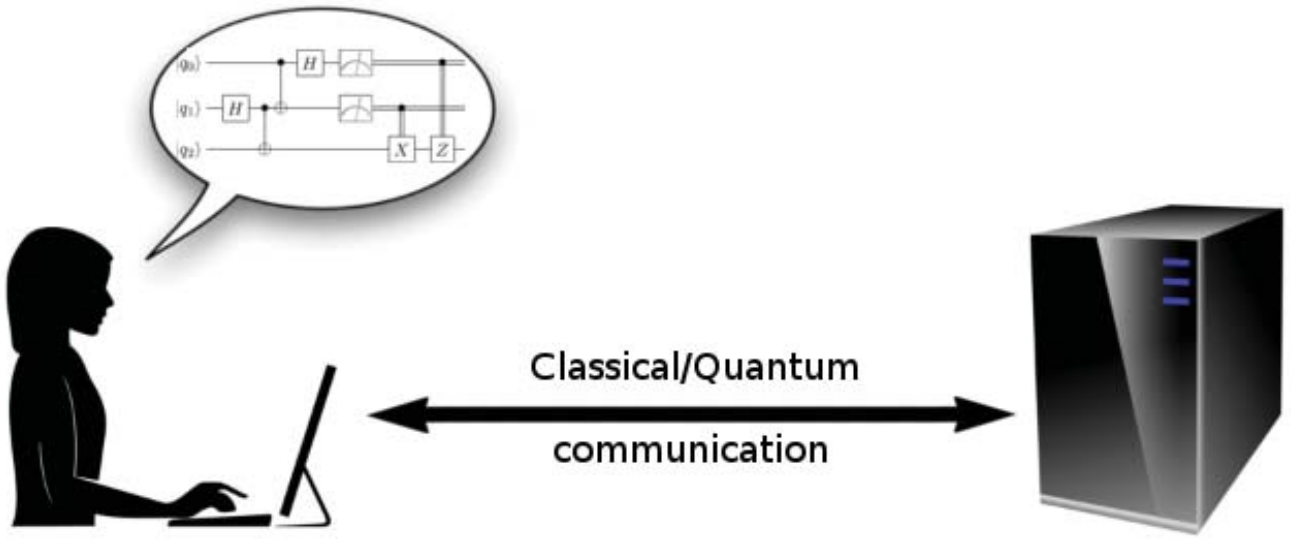
Quantum Turing test



- Can we classically communicate or do we need a bit of **quantumness** to test quantum computer

- Can we do the test **efficiently** or do we need super powerful computer to test quantum computer

Quantum Verification



- Computationally limited verifier
- Powerful quantum server(s)
- Certify the correctness of the computation

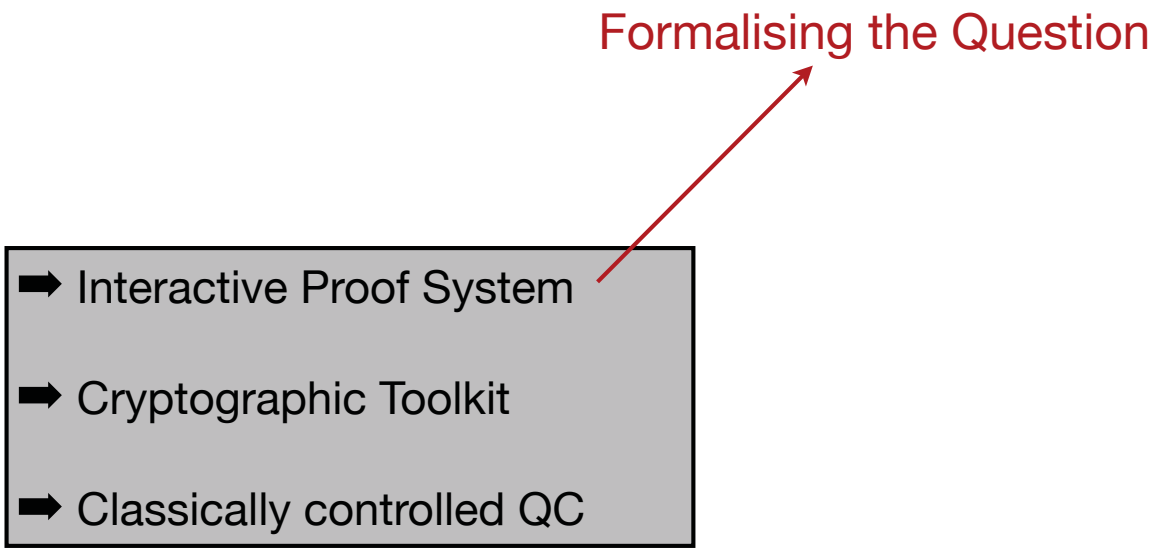
General Approaches

- State authentication-Based protocols
- Trapification based
- Measuring entanglement correlation protocols

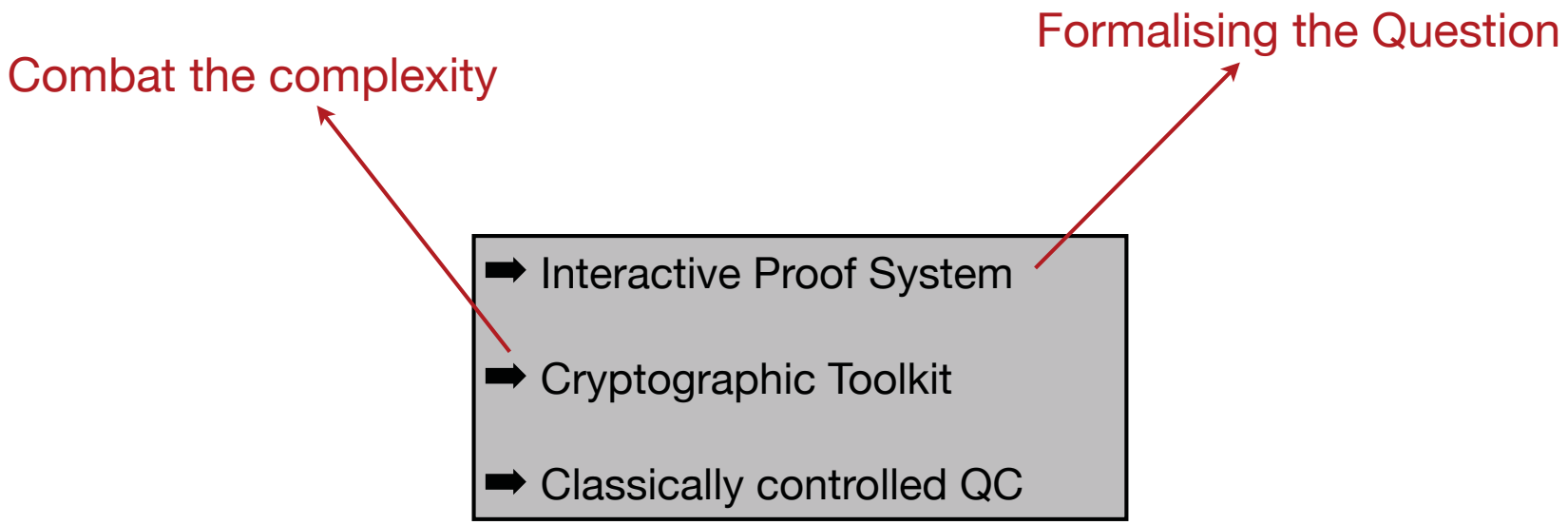
How do we do it

- ➡ Interactive Proof System
- ➡ Cryptographic Toolkit
- ➡ Classically controlled QC

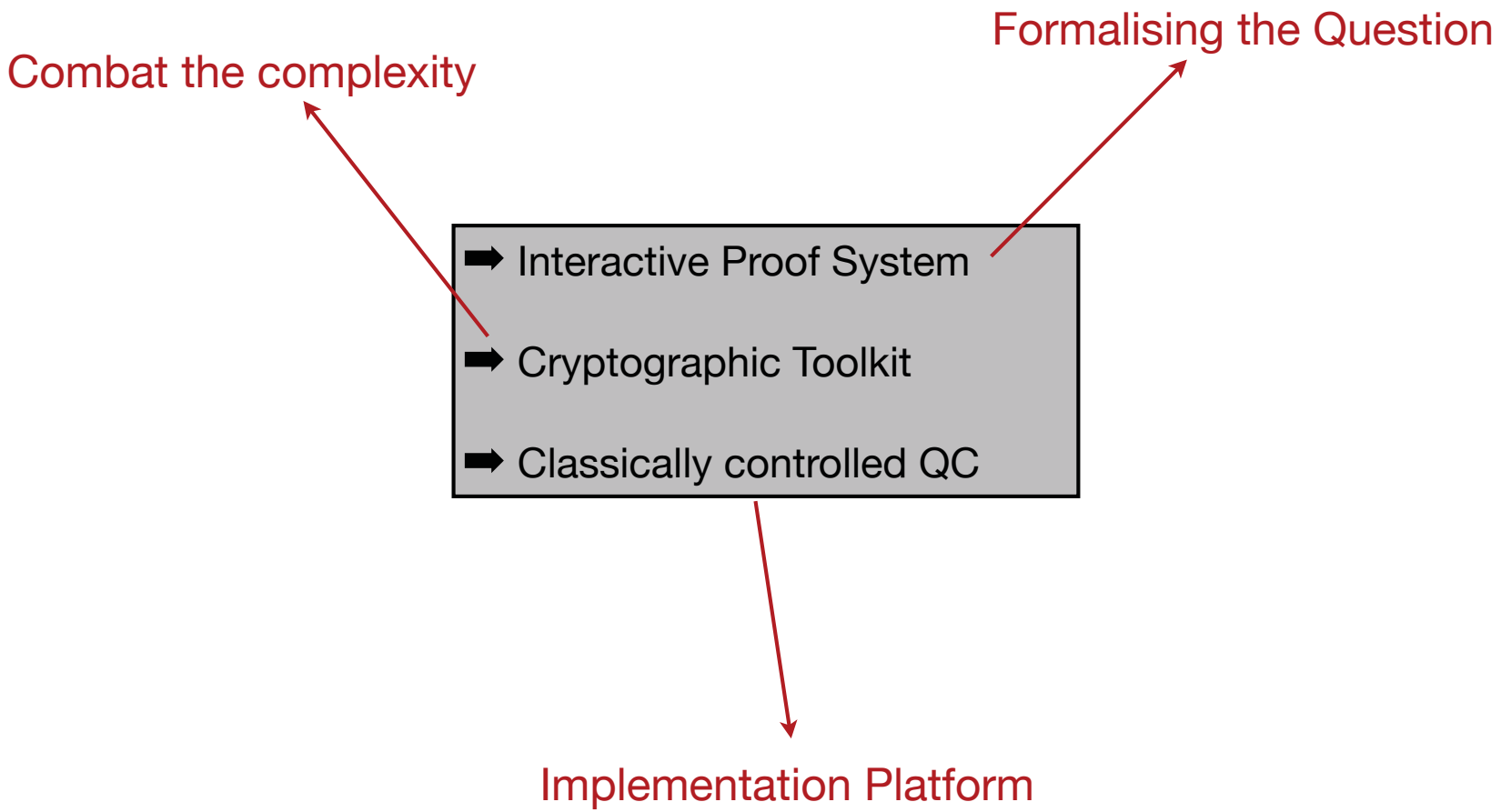
How do we do it



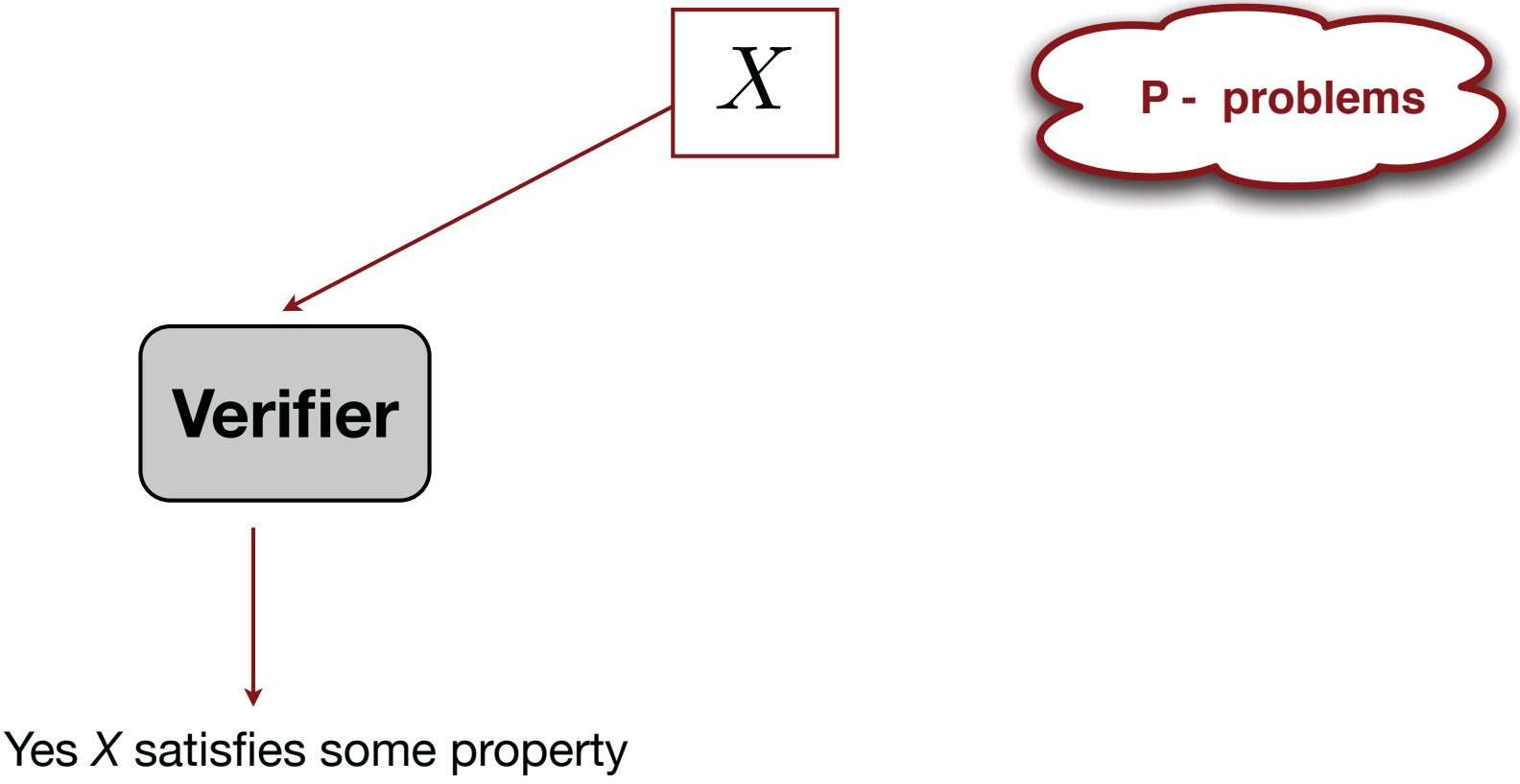
How do we do it



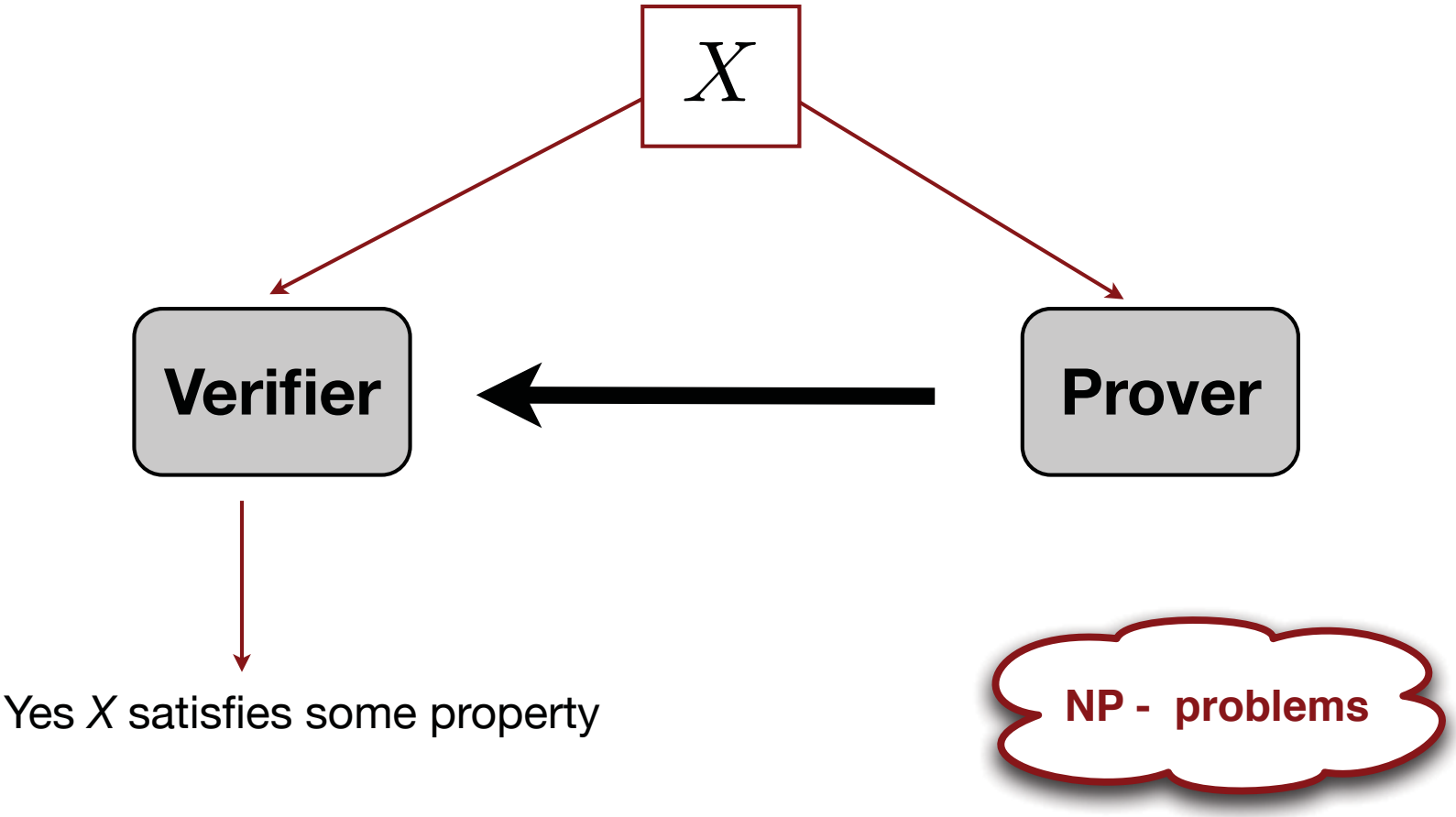
How do we do it



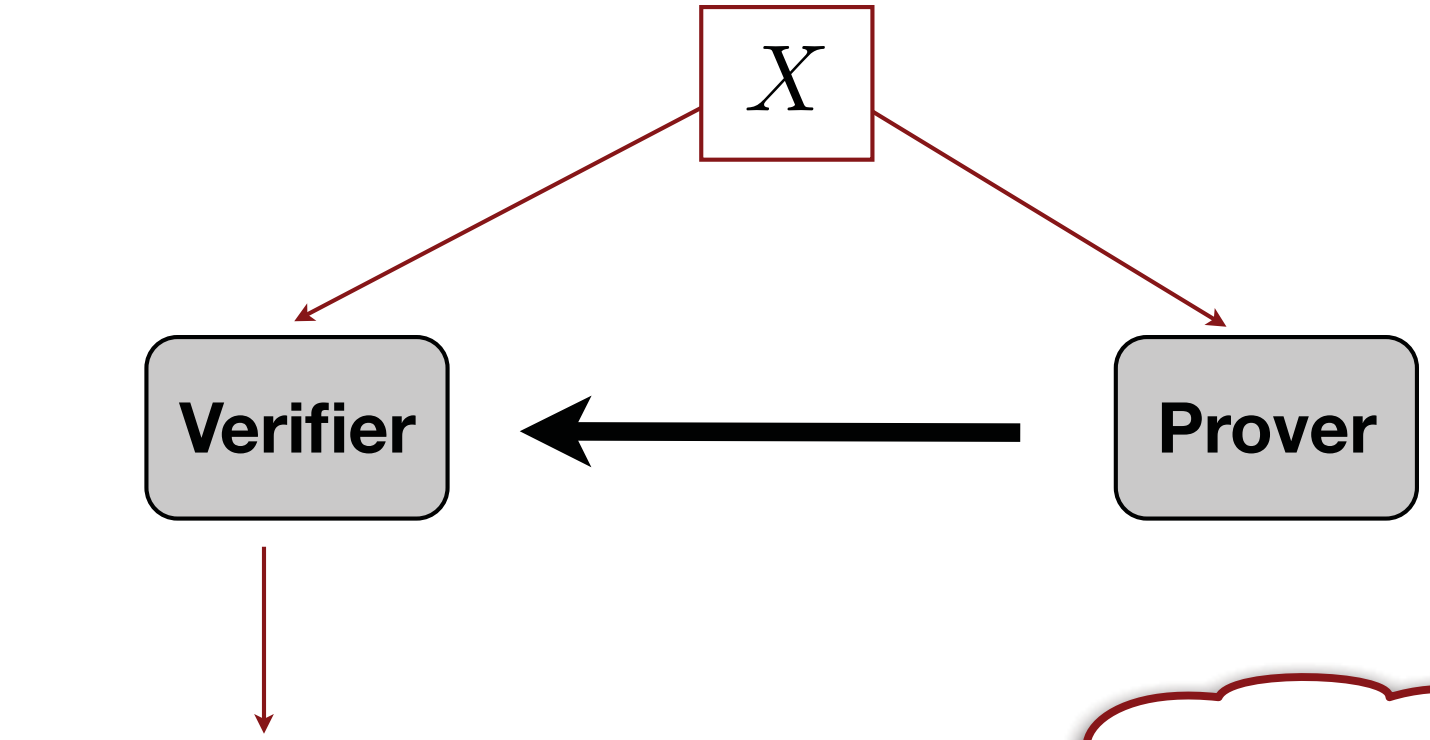
Complexity as Proof System



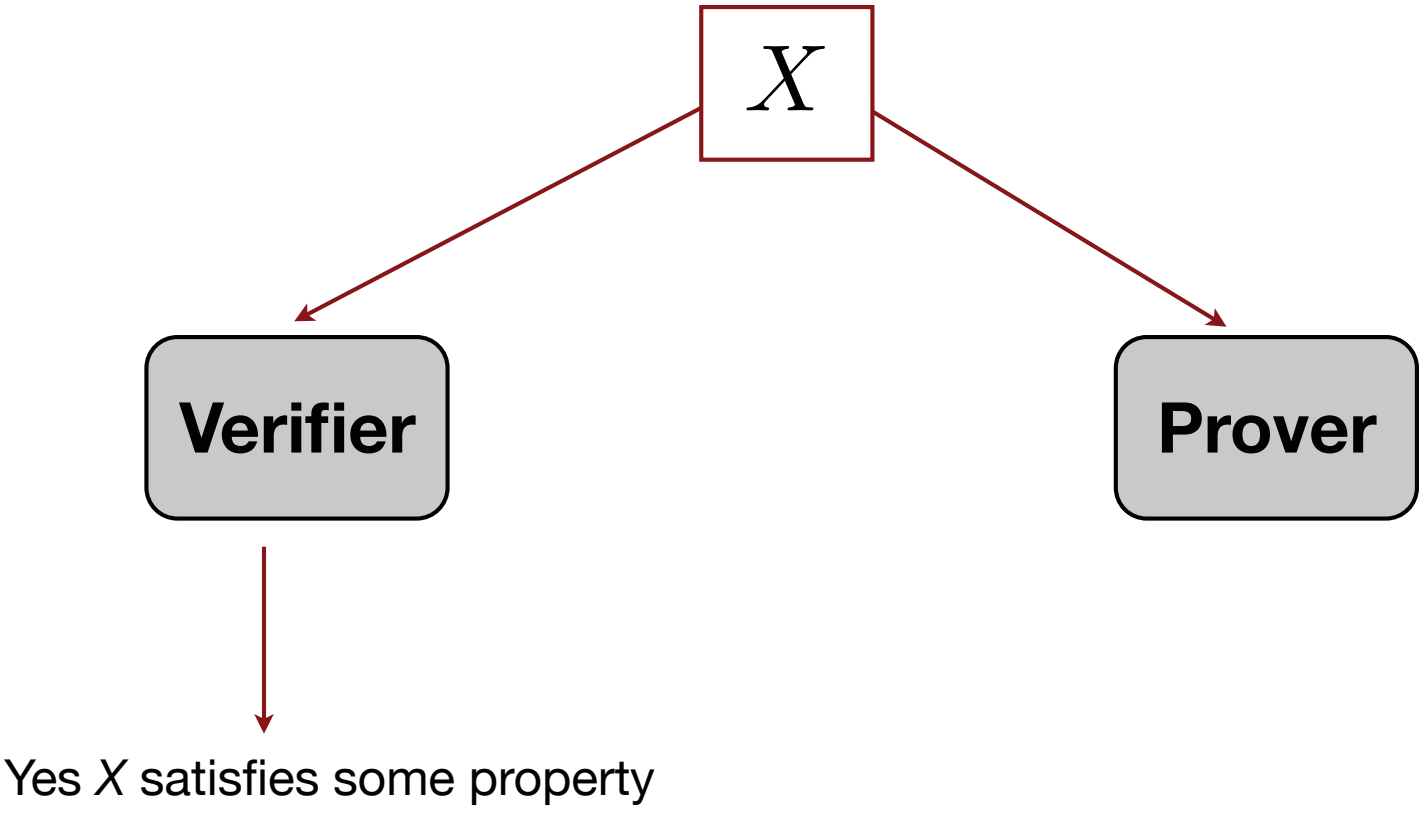
Complexity as Proof System



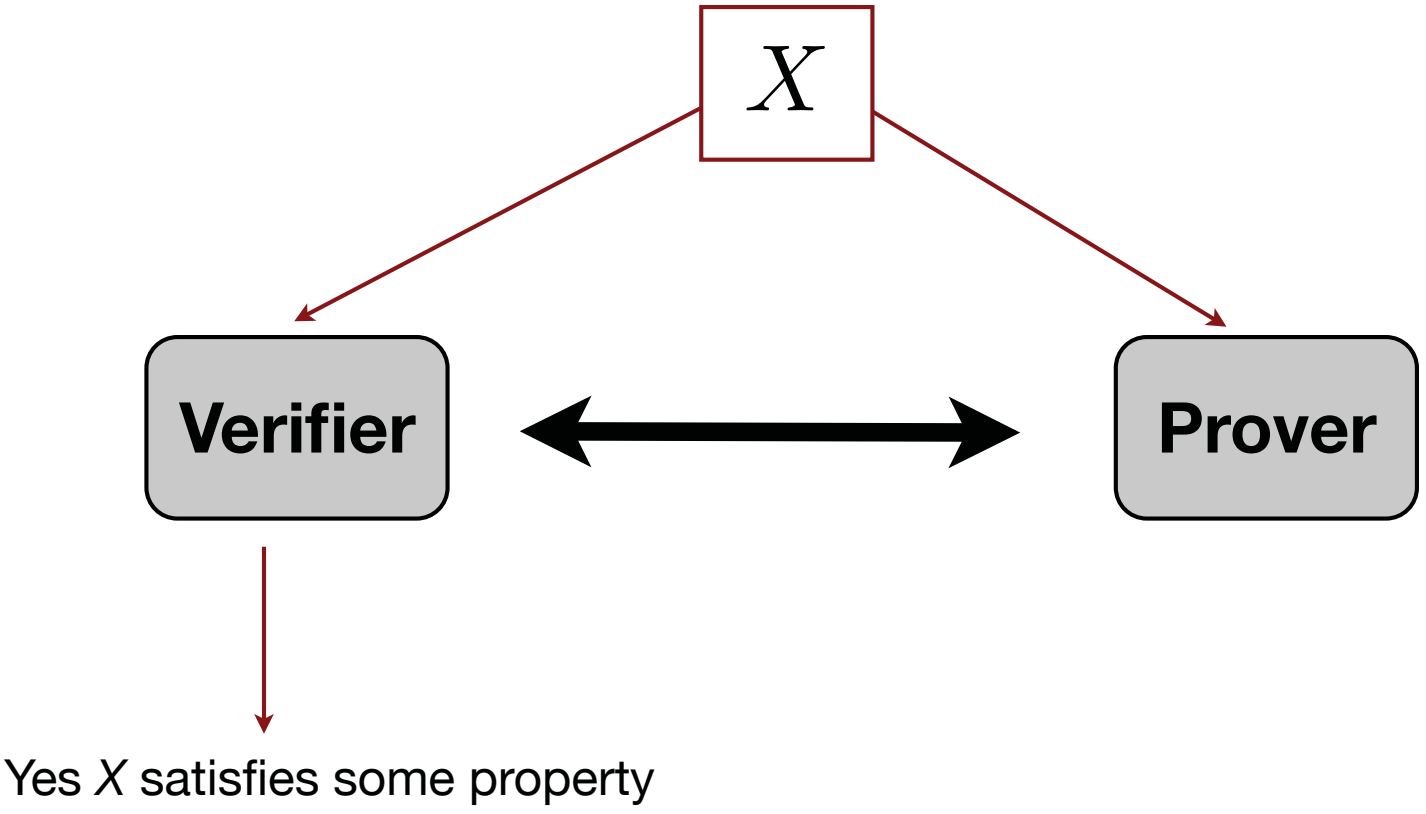
Complexity as Proof System



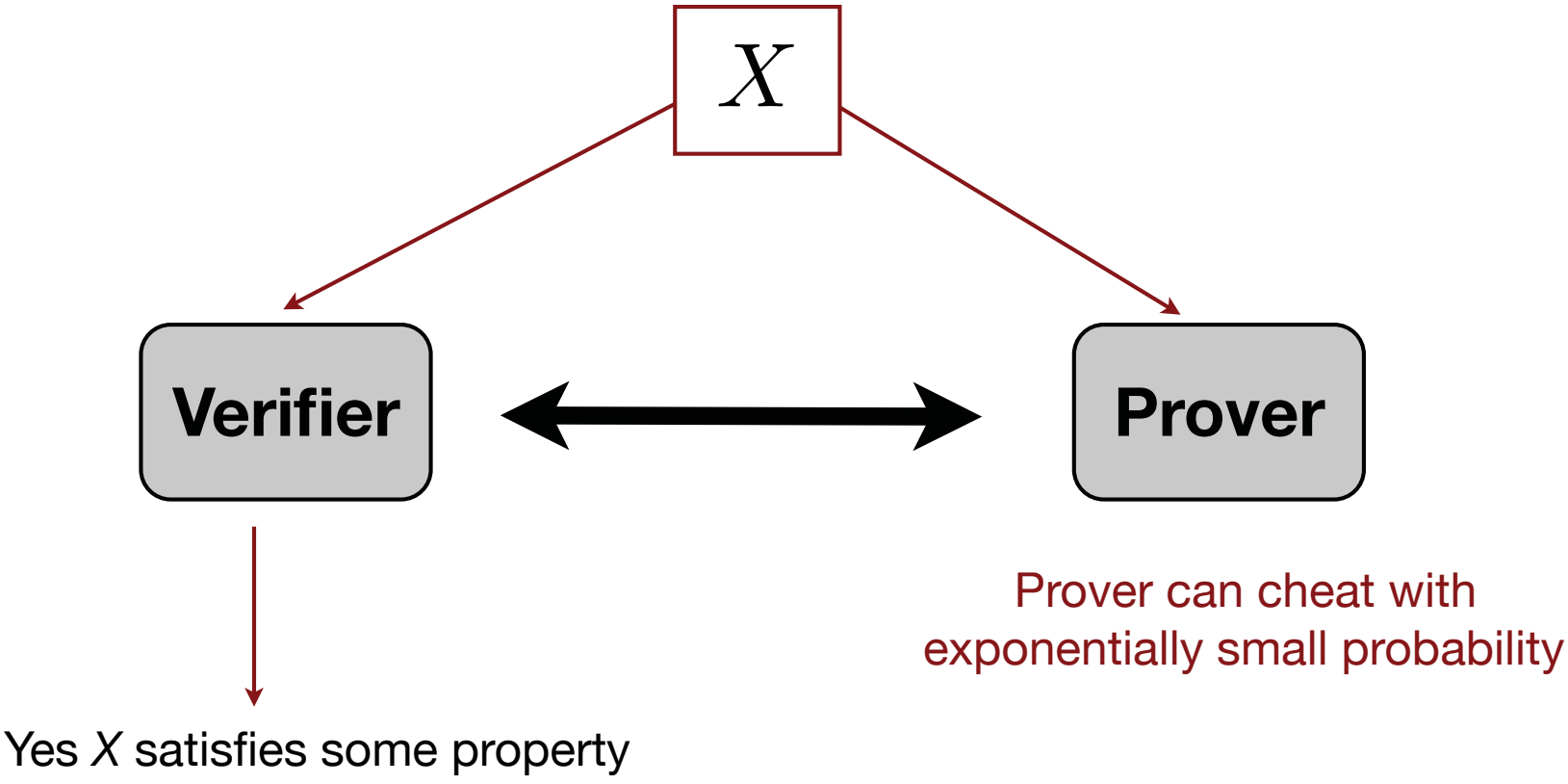
Generalisation: Interactive Proofs



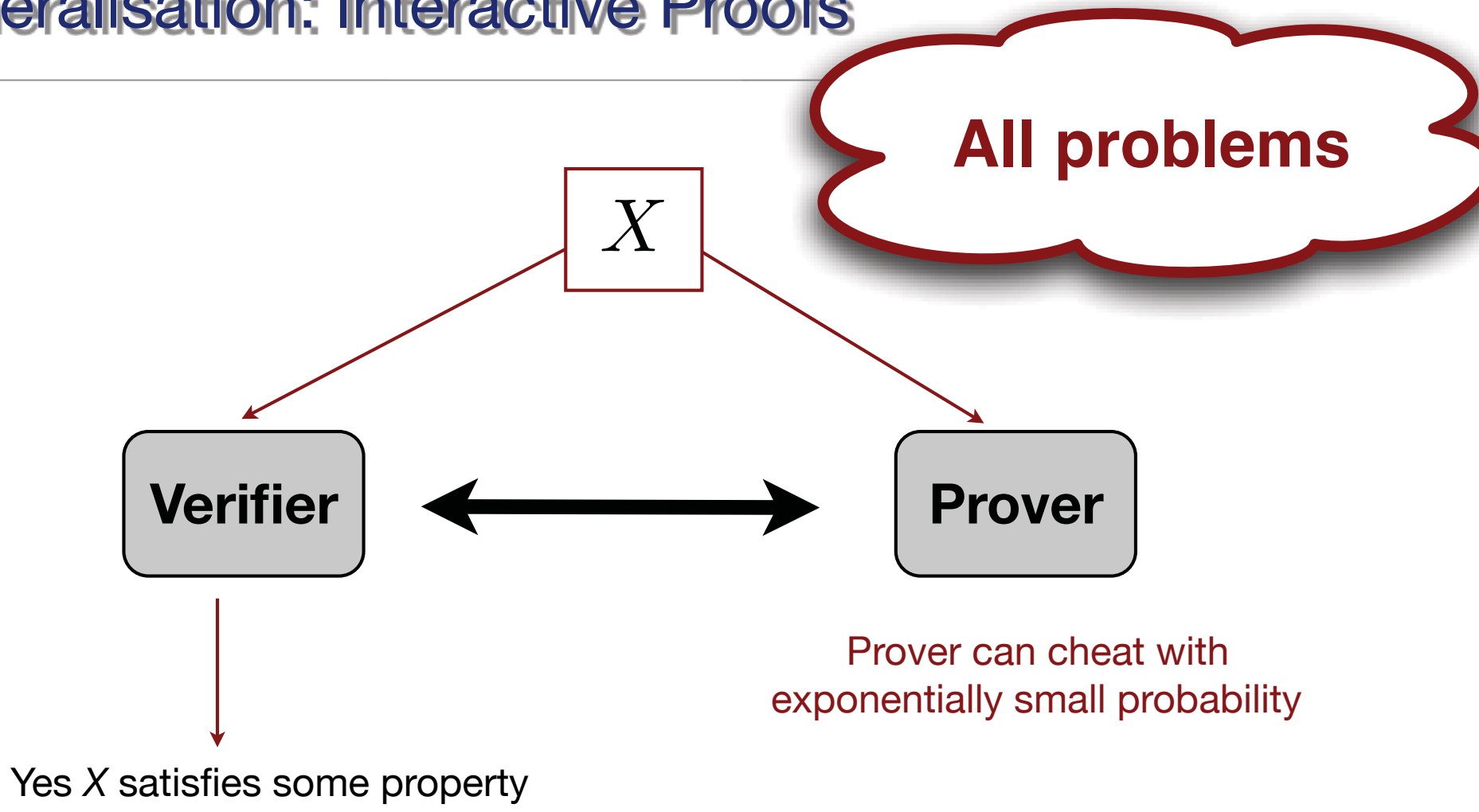
Generalisation: Interactive Proofs



Generalisation: Interactive Proofs



Generalisation: Interactive Proofs



Interactive Proofs

A **PROTOCOL** between a

computationally unrestricted prover P
and
a probabilistic polynomial-time verifier V

Interactive Proofs

A **PROTOCOL** between a

computationally unrestricted prover P
and
a probabilistic polynomial-time verifier V

Completeness: $(\forall x \in L)\ \Pr\left[(V \leftrightarrow P)(x)\ accepts\right] = 1$

Interactive Proofs

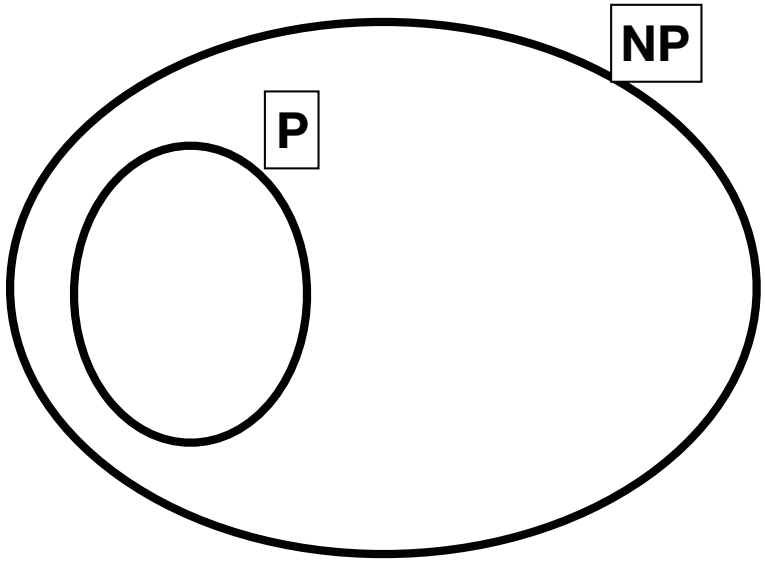
A **PROTOCOL** between a

computationally unrestricted prover P
and
a probabilistic polynomial-time verifier V

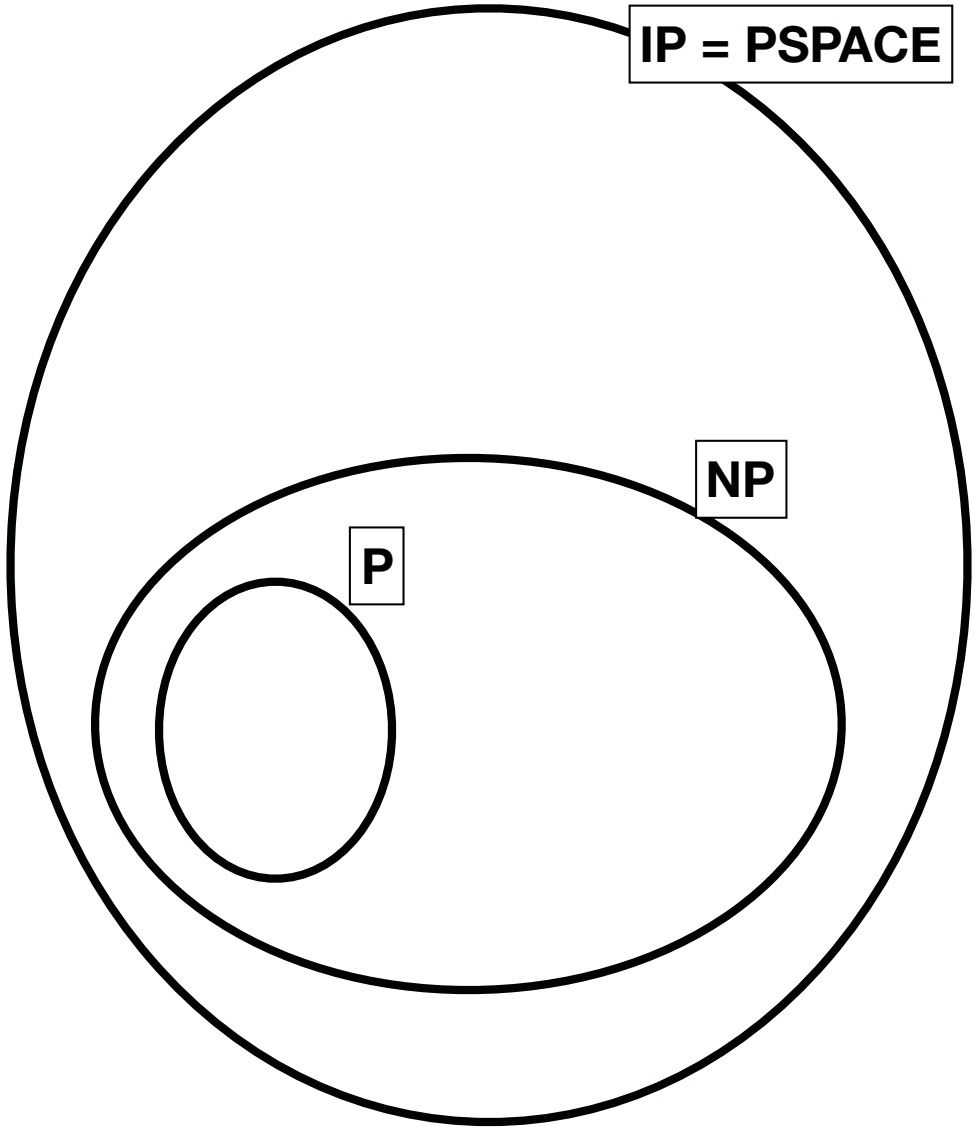
Completeness: $(\forall x \in L)\ \Pr\left[(V \leftrightarrow P)(x)\ accepts\right] = 1$

Soundness: $(\forall x \notin L)(\forall P')\ \Pr\left[(V \leftrightarrow P')(x)\ accepts\right] \leq \frac{1}{2}$

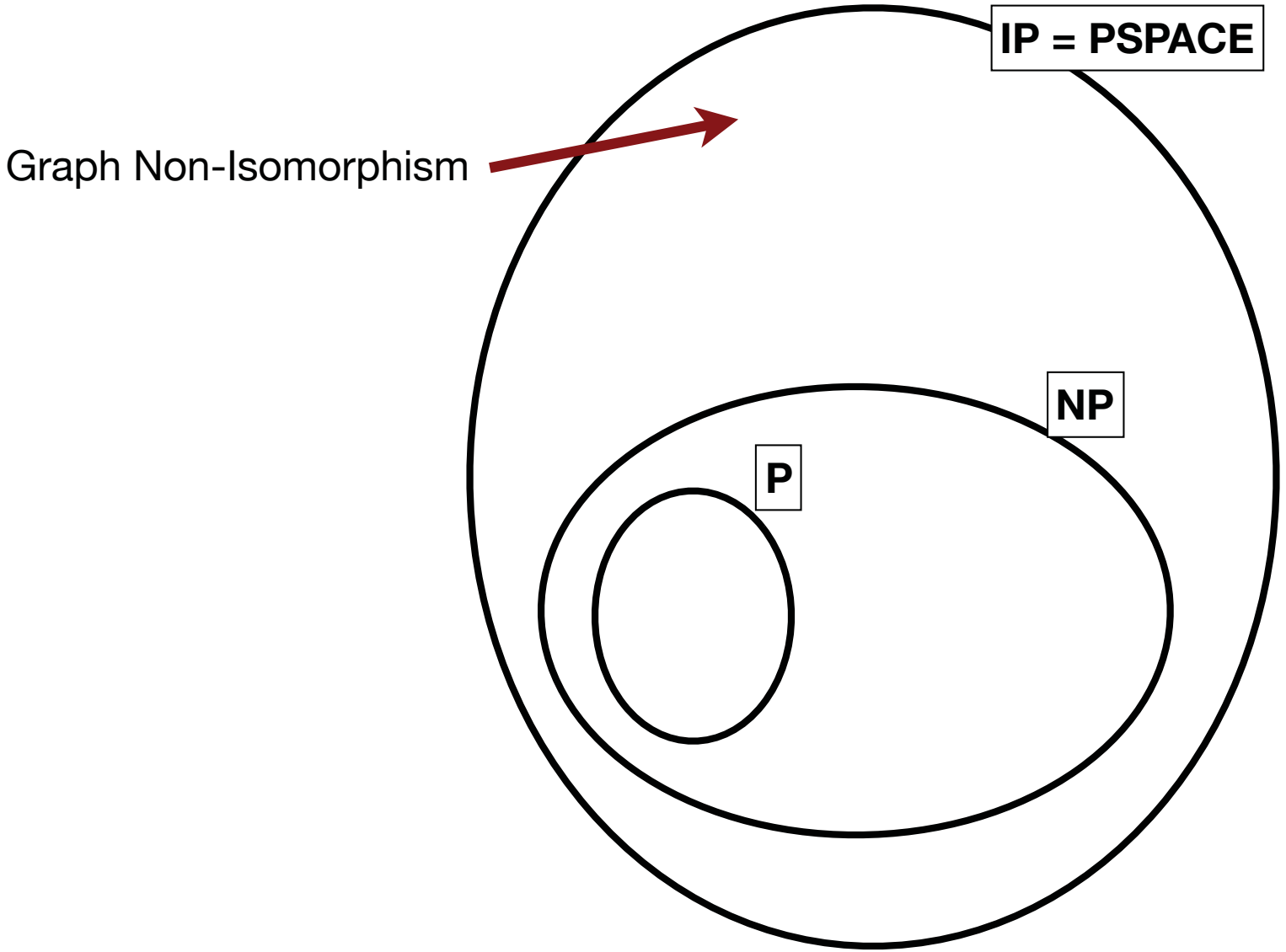
Interactive Proofs



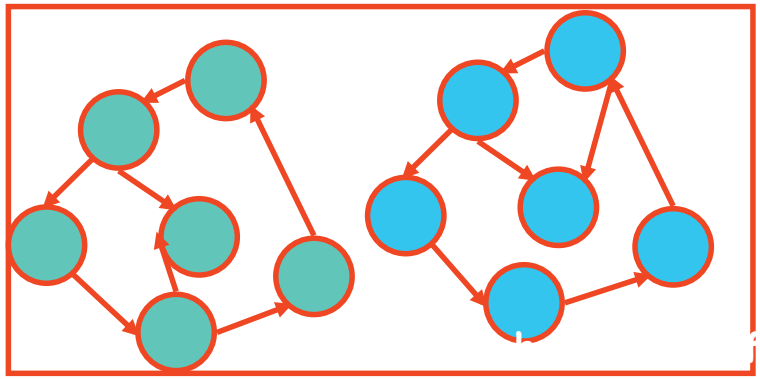
Interactive Proofs



Interactive Proofs

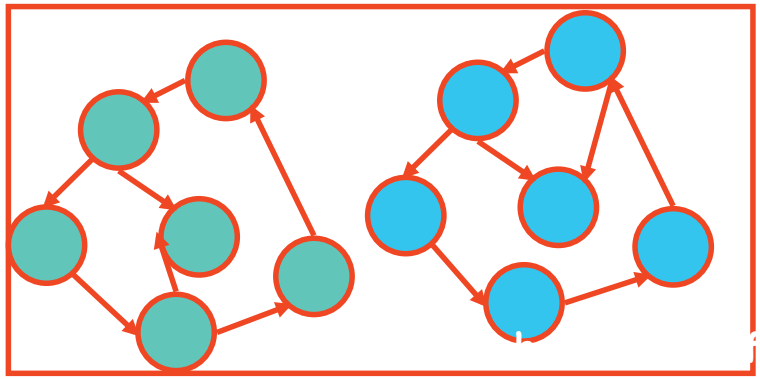


Graph Non-Isomorphism



The common input

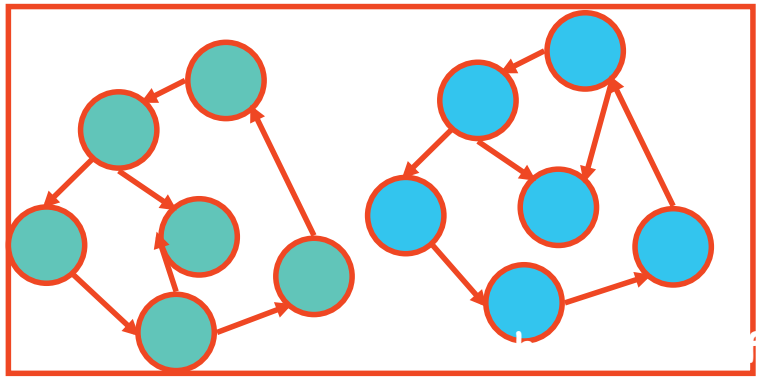
Graph Non-Isomorphism



The common input

Verifier chooses one of the two graphs randomly

Graph Non-Isomorphism

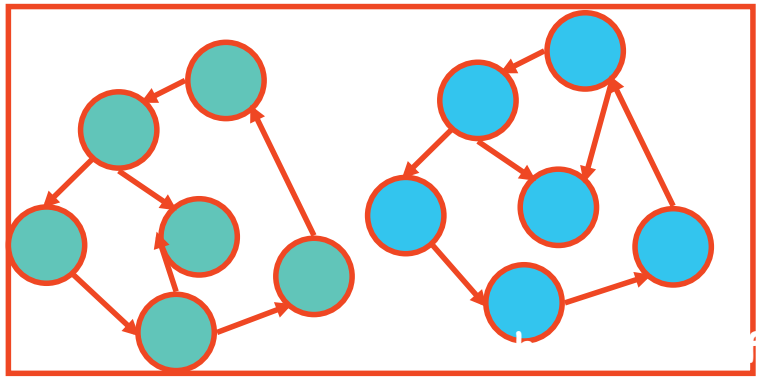


The common input

Verifier chooses one of the two graphs randomly

Verifier constructs a graph isomorphic to her choice and send it to the Prover

Graph Non-Isomorphism



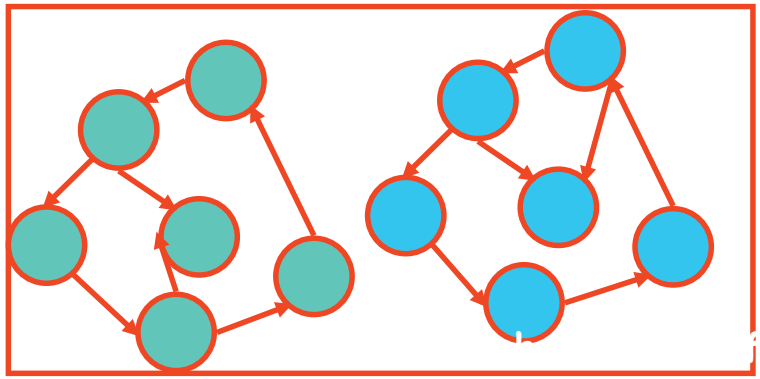
The common input

Verifier chooses one of the two graphs randomly

Verifier constructs a graph isomorphic to her choice and send it to the Prover

If $G \ncong H$ prover can find which of the two graphs was send by the verifier

Graph Non-Isomorphism



The common input

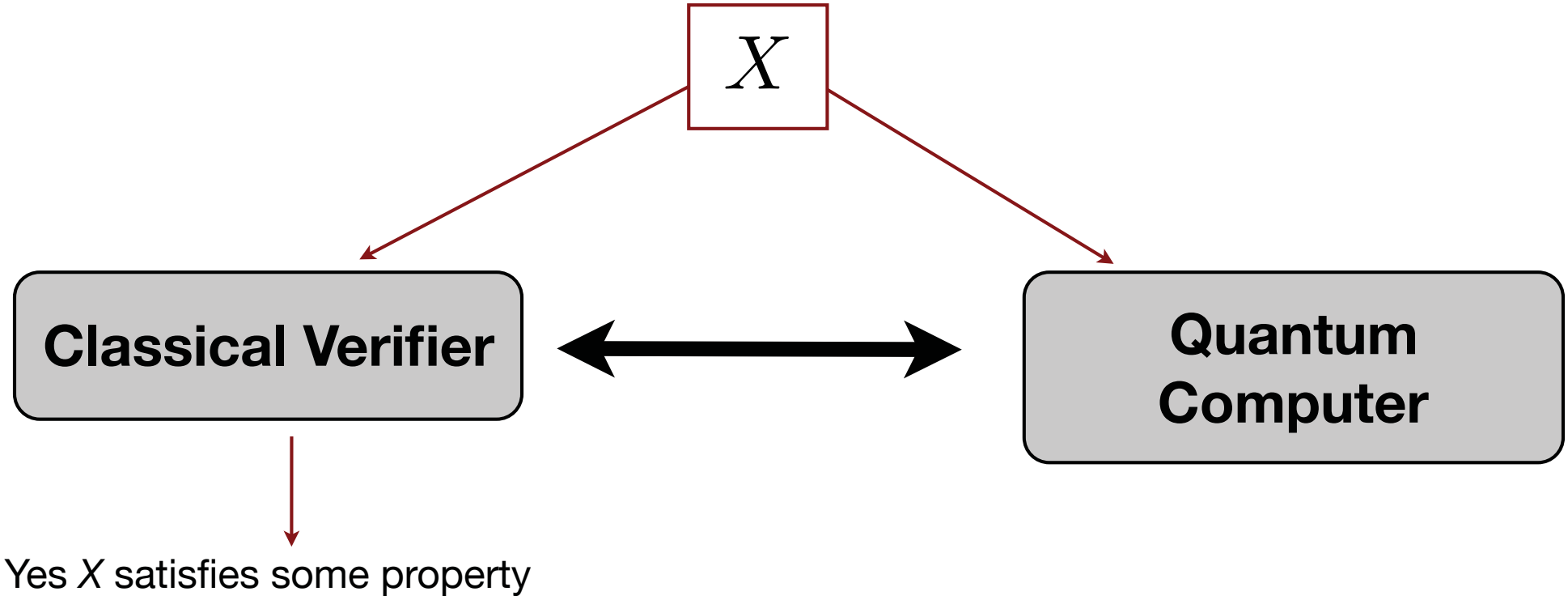
Verifier chooses one of the two graphs randomly

Verifier constructs a graph isomorphic to her choice and send it to the Prover

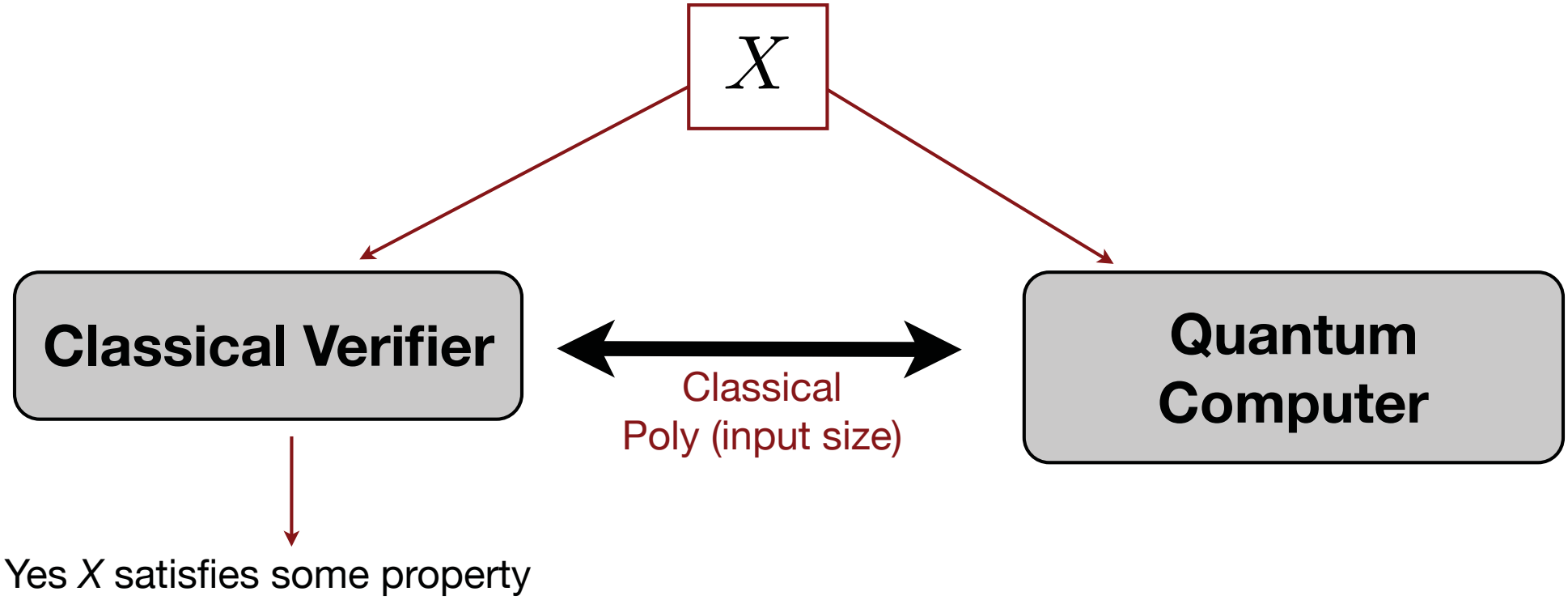
If $G \ncong H$ prover can find which of the two graphs was send by the verifier

The verifier can check the answer easily

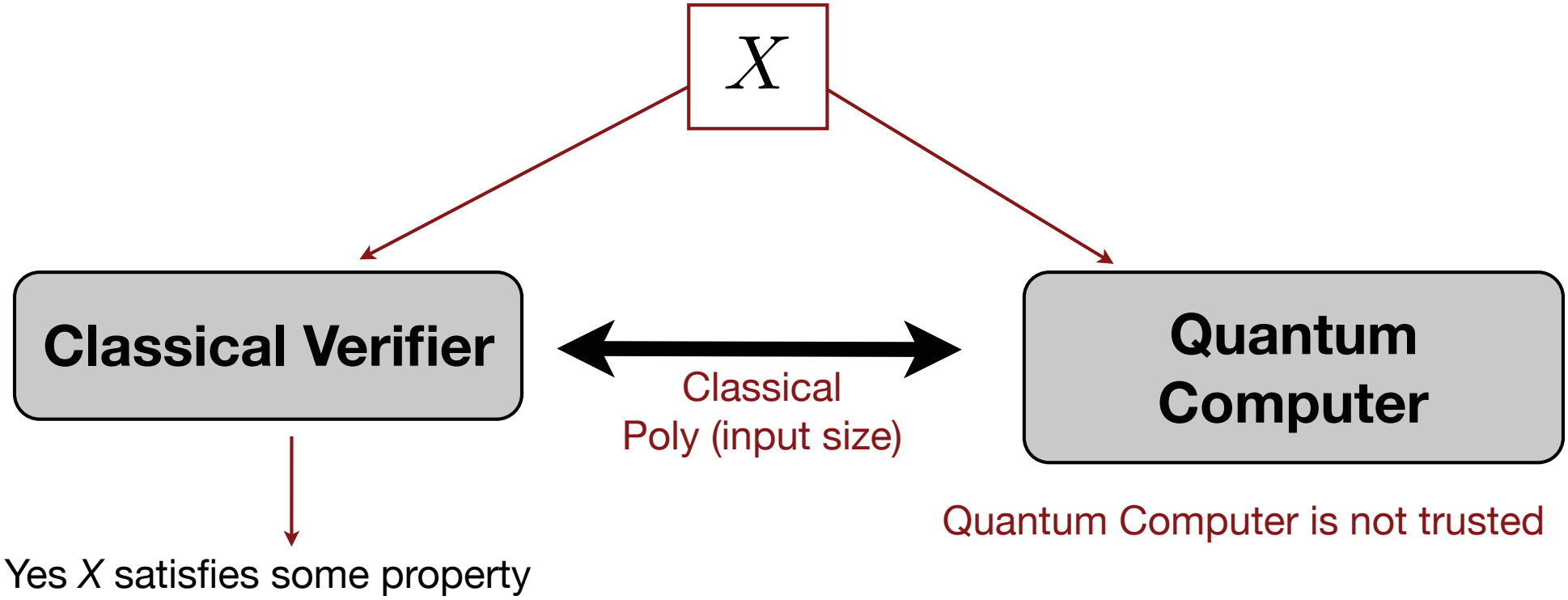
IP for Quantum Computing



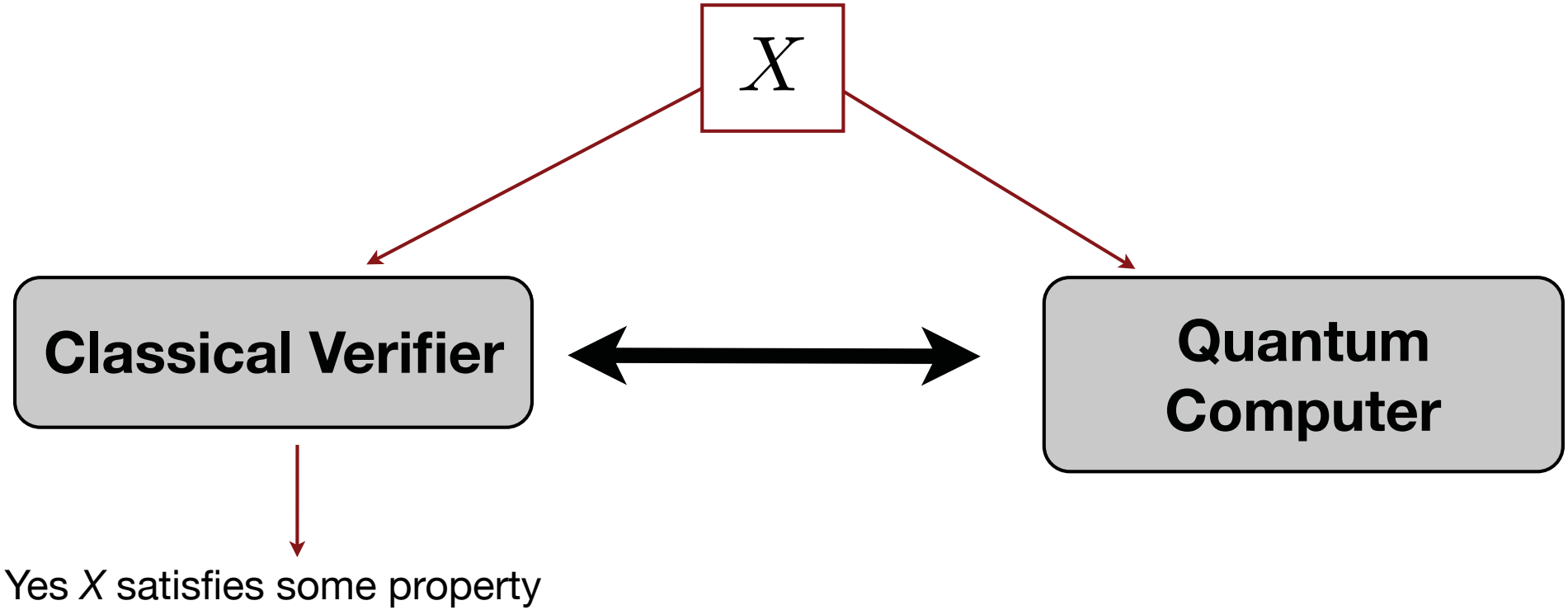
IP for Quantum Computing



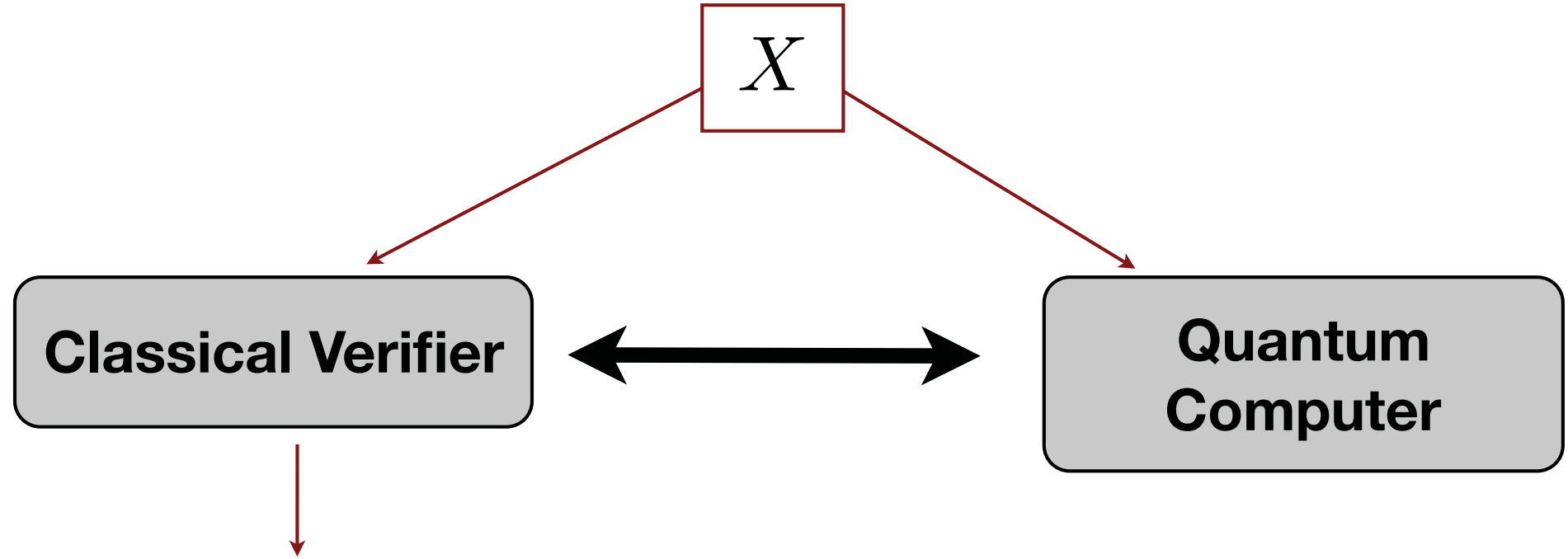
IP for Quantum Computing



IP for Quantum Computing



IP for Quantum Computing

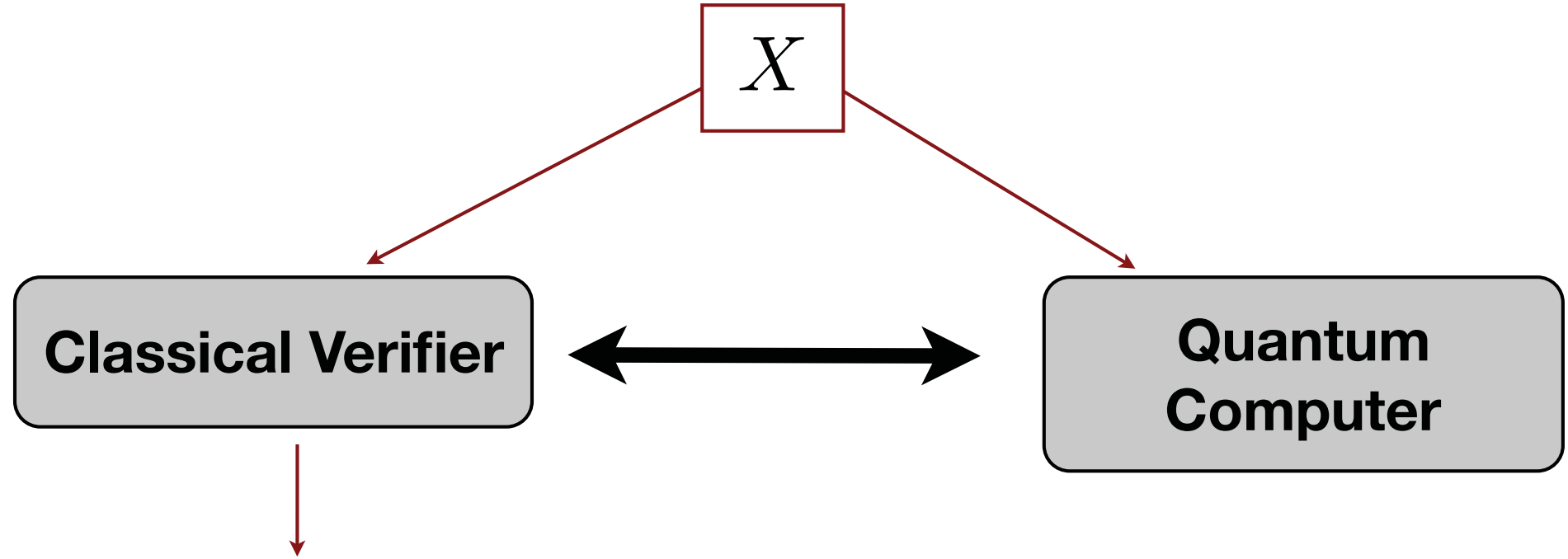


Gottesman (04) - Vazirani (07)- Aaronson **$25** *Challenge (07)*

Does BQP admit an interactive protocol

where the prover is in BQP and the verifier is in BPP?

IP for Quantum Computing



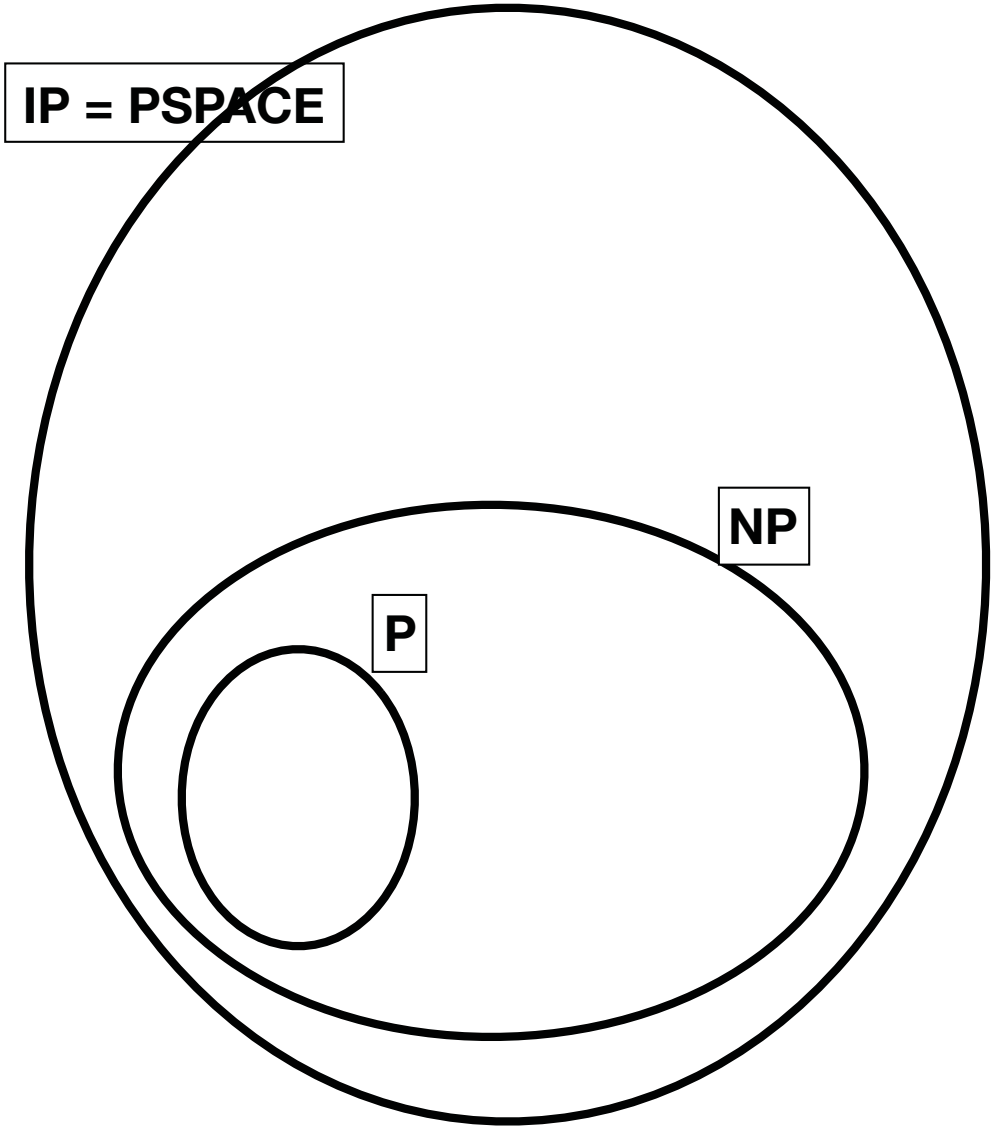
Gottesman (04) - Vazirani (07)- Aaronson ***$25*** *Challenge (07)*

Does BQP admit an interactive protocol

where the prover is in BQP and the verifier is in BPP?

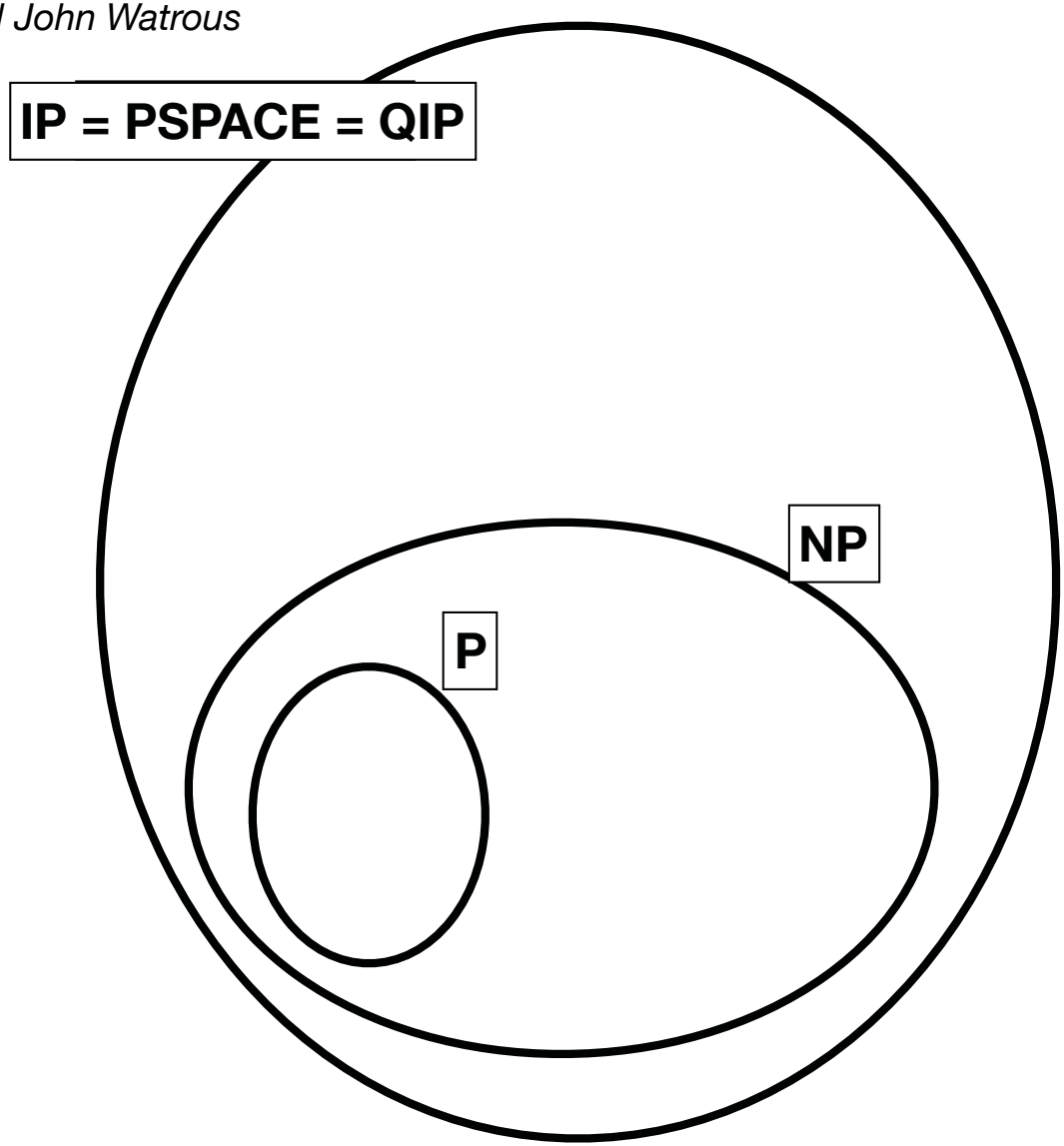
D. Aharonov and U. Vazirani, arXiv:1206.3686 (2012).

IP for Quantum Computing



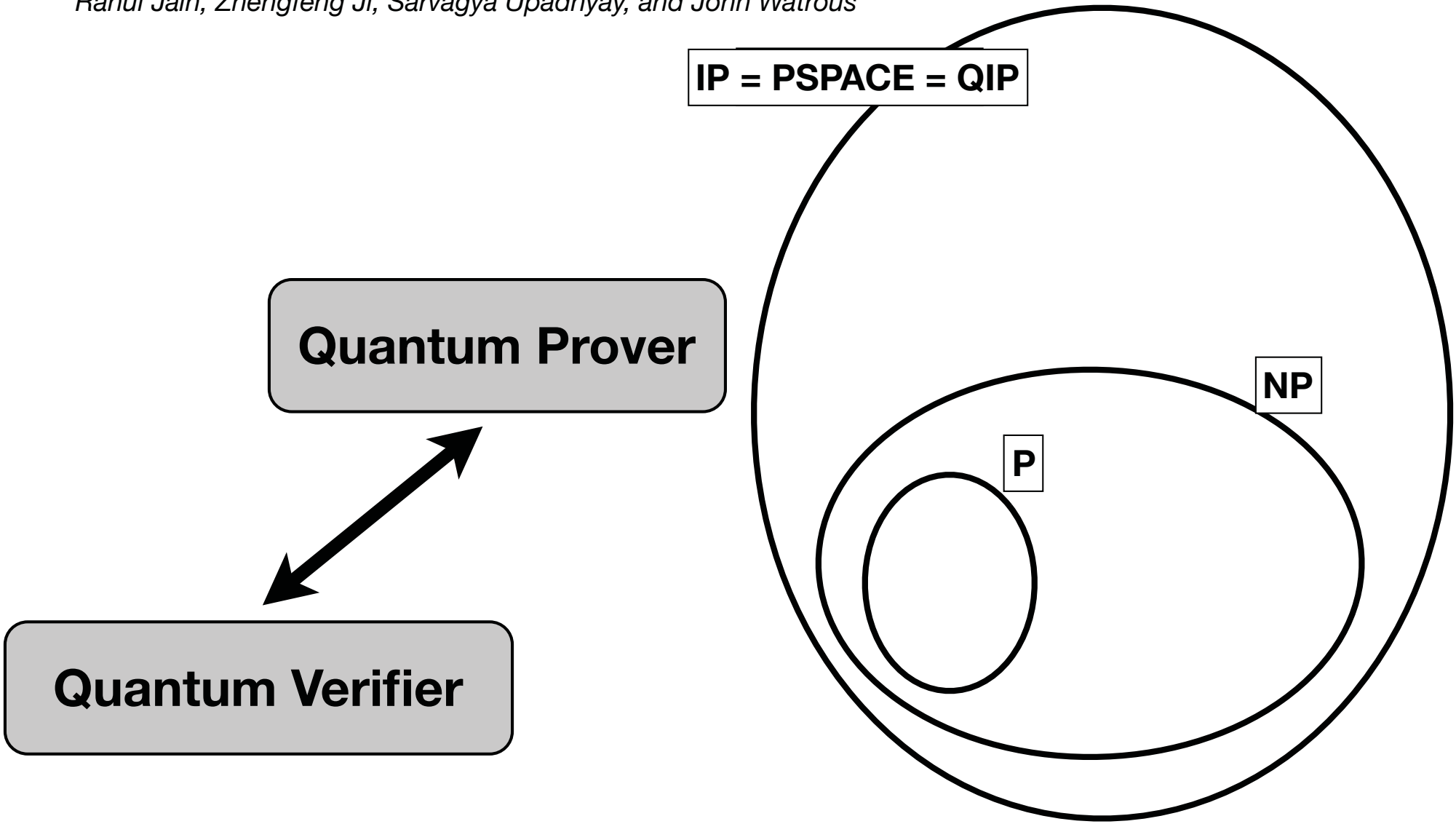
IP for Quantum Computing

Rahul Jain, Zhengfeng Ji, Sarvagya Upadhyay, and John Watrous



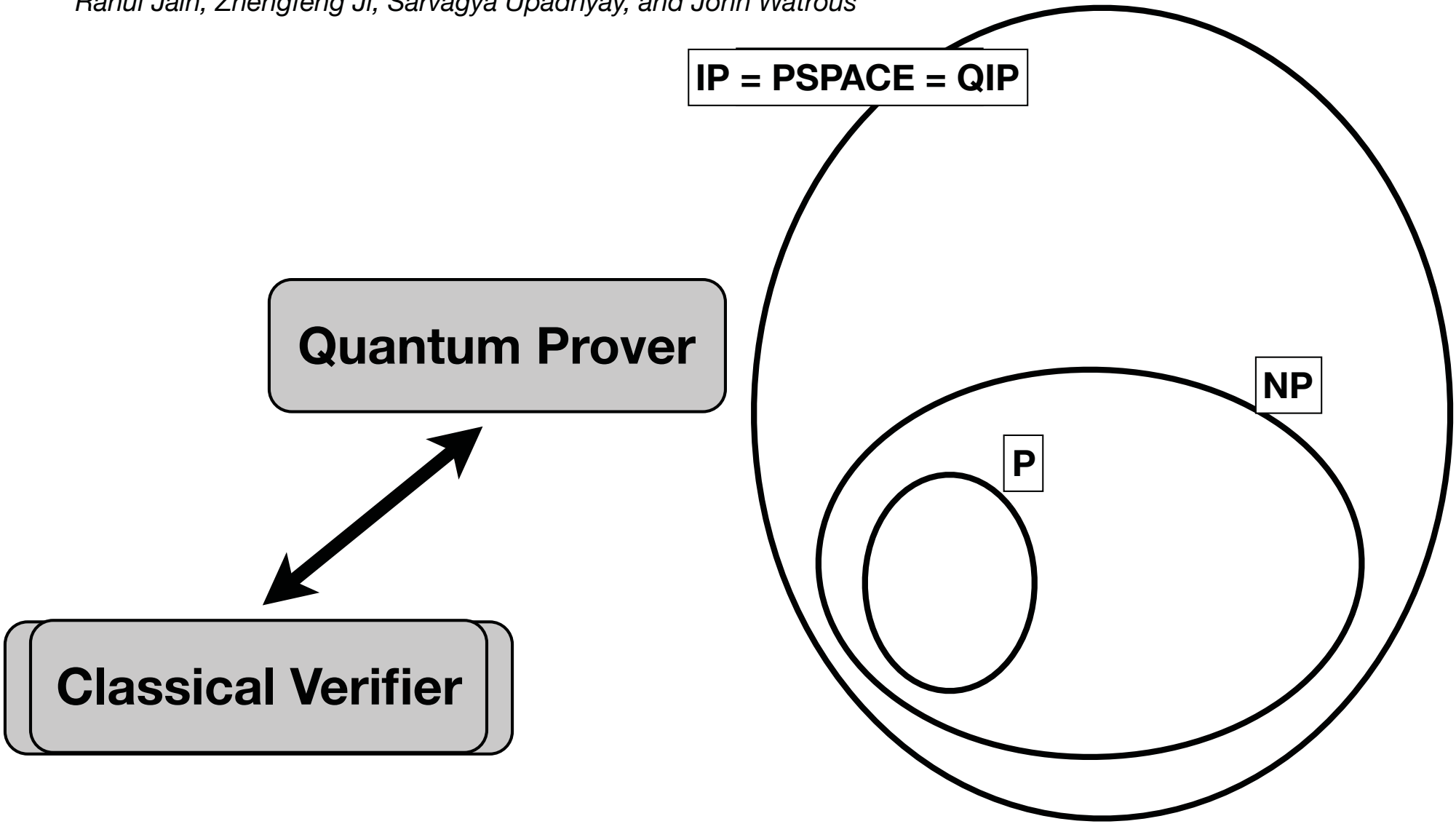
IP for Quantum Computing

Rahul Jain, Zhengfeng Ji, Sarvagya Upadhyay, and John Watrous



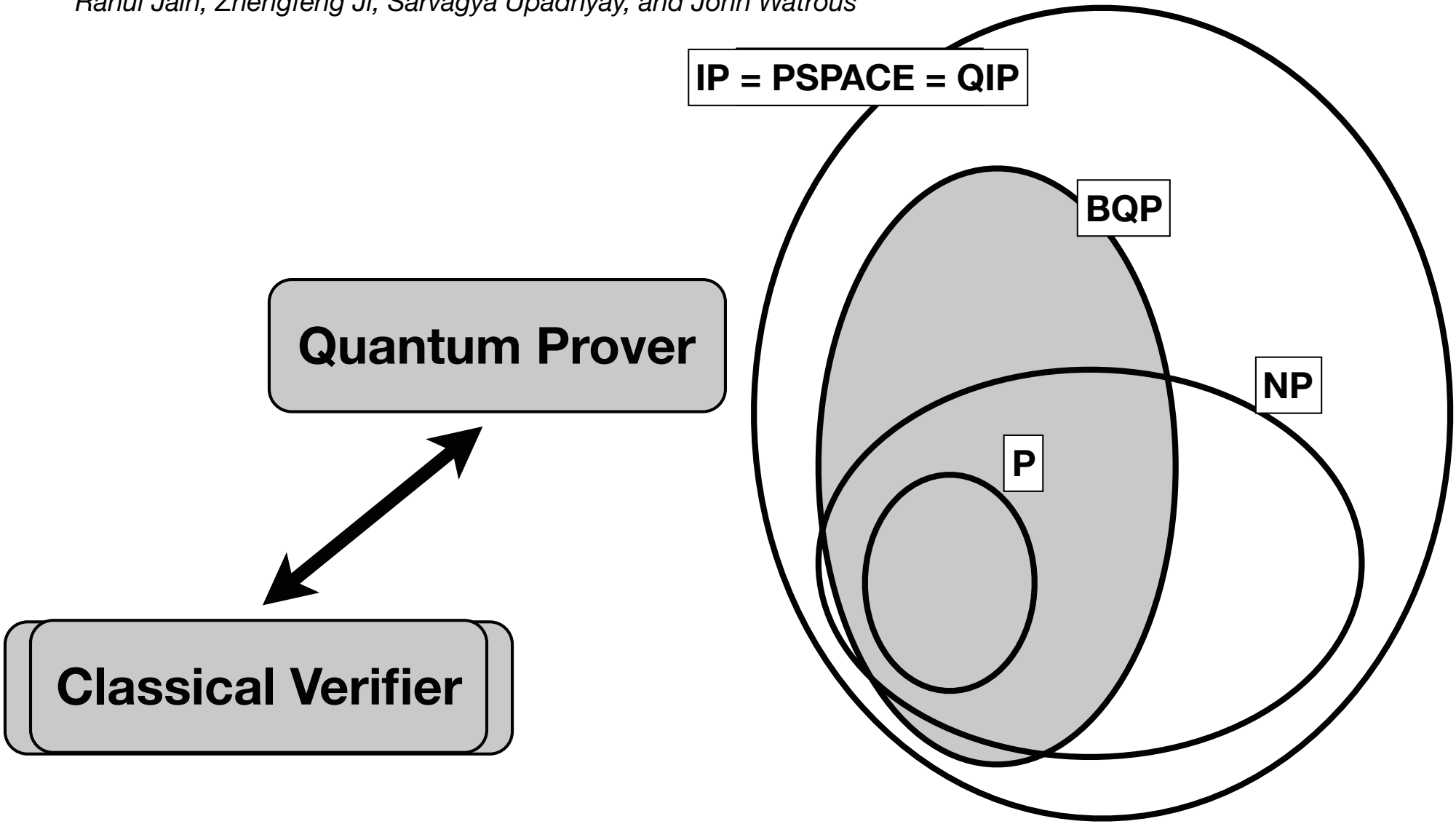
IP for Quantum Computing

Rahul Jain, Zhengfeng Ji, Sarvagya Upadhyay, and John Watrous



IP for Quantum Computing

Rahul Jain, Zhengfeng Ji, Sarvagya Upadhyay, and John Watrous



Hiding Complexity

Hiding Complexity

Encryption

Enables Secure Communication

Access to data is all or nothing

Hiding Complexity

Encryption

Enables Secure Communication

Access to data is all or nothing

Modern Encryption

Enables arbitrary computation on encrypted data ***without decrypting***

Holy Grail of Cryptography since 1987

Rivest, Adleman and Dertouzos

Can we process encrypted data without decrypting it

Holy Grail of Cryptography since 1987

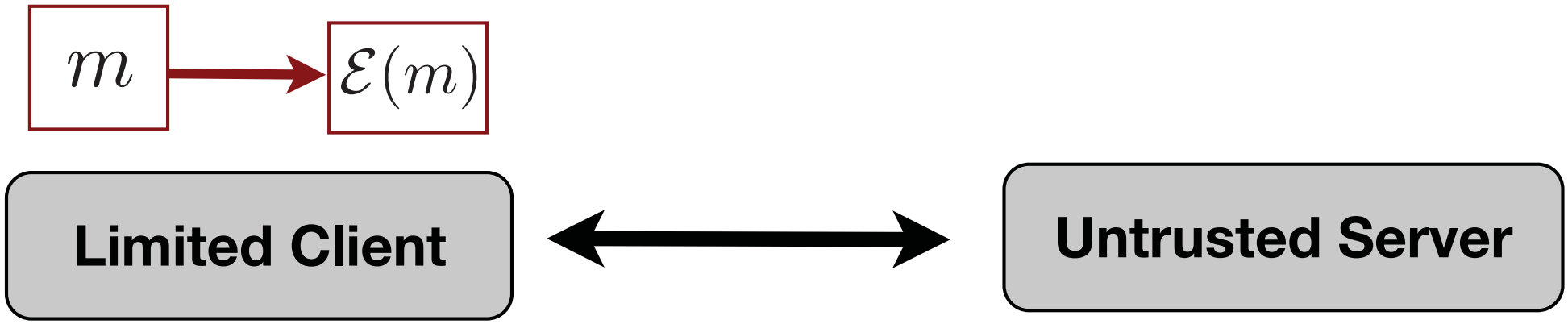
Rivest, Adleman and Dertouzos

Can we process encrypted data without decrypting it

Holy Grail of Cryptography since 1987

Rivest, Adleman and Dertouzos

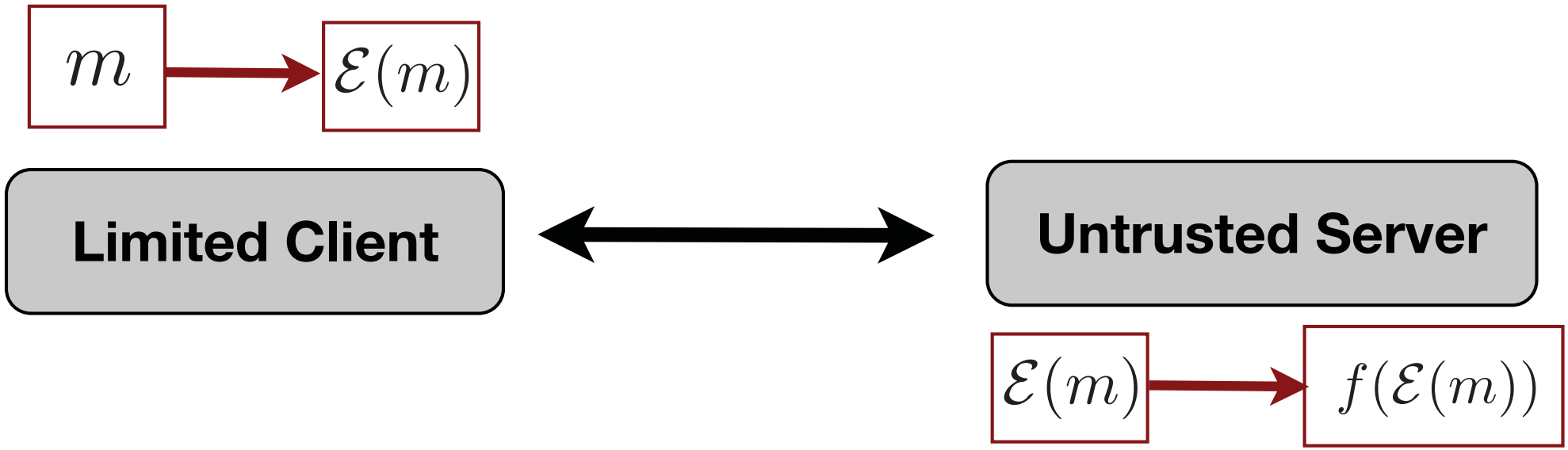
Can we process encrypted data without decrypting it



Holy Grail of Cryptography since 1987

Rivest, Adleman and Dertouzos

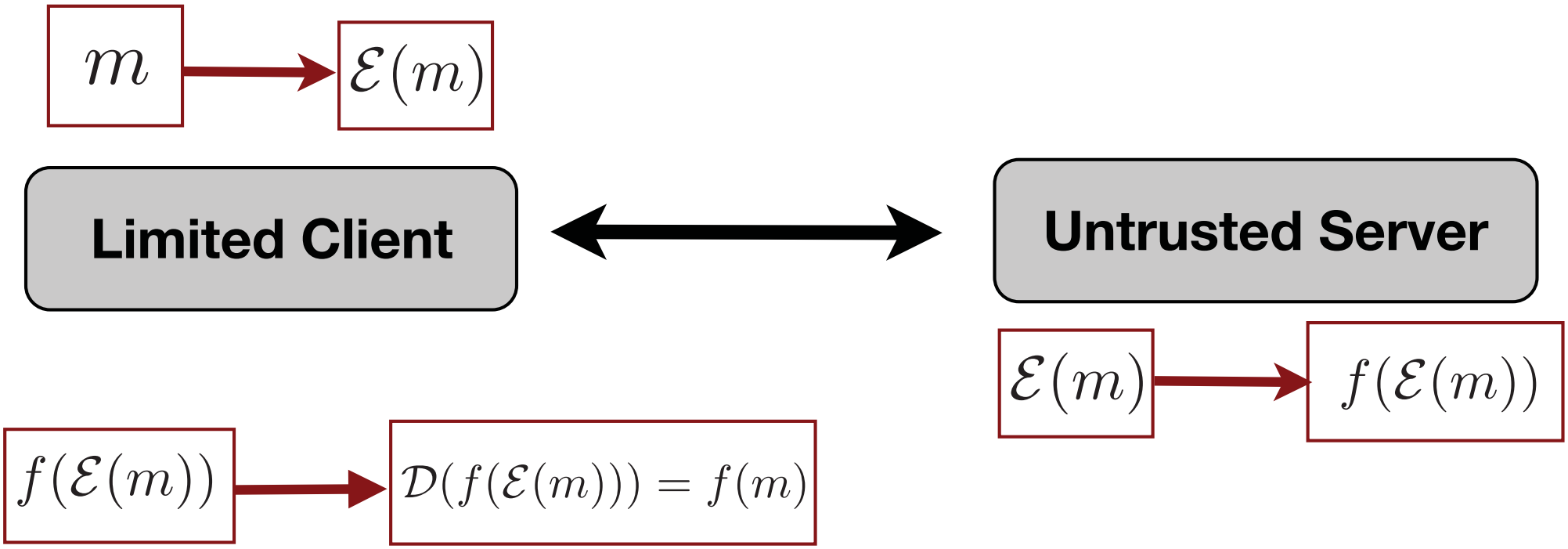
Can we process encrypted data without decrypting it



Holy Grail of Cryptography since 1987

Rivest, Adleman and Dertouzos

Can we process encrypted data without decrypting it



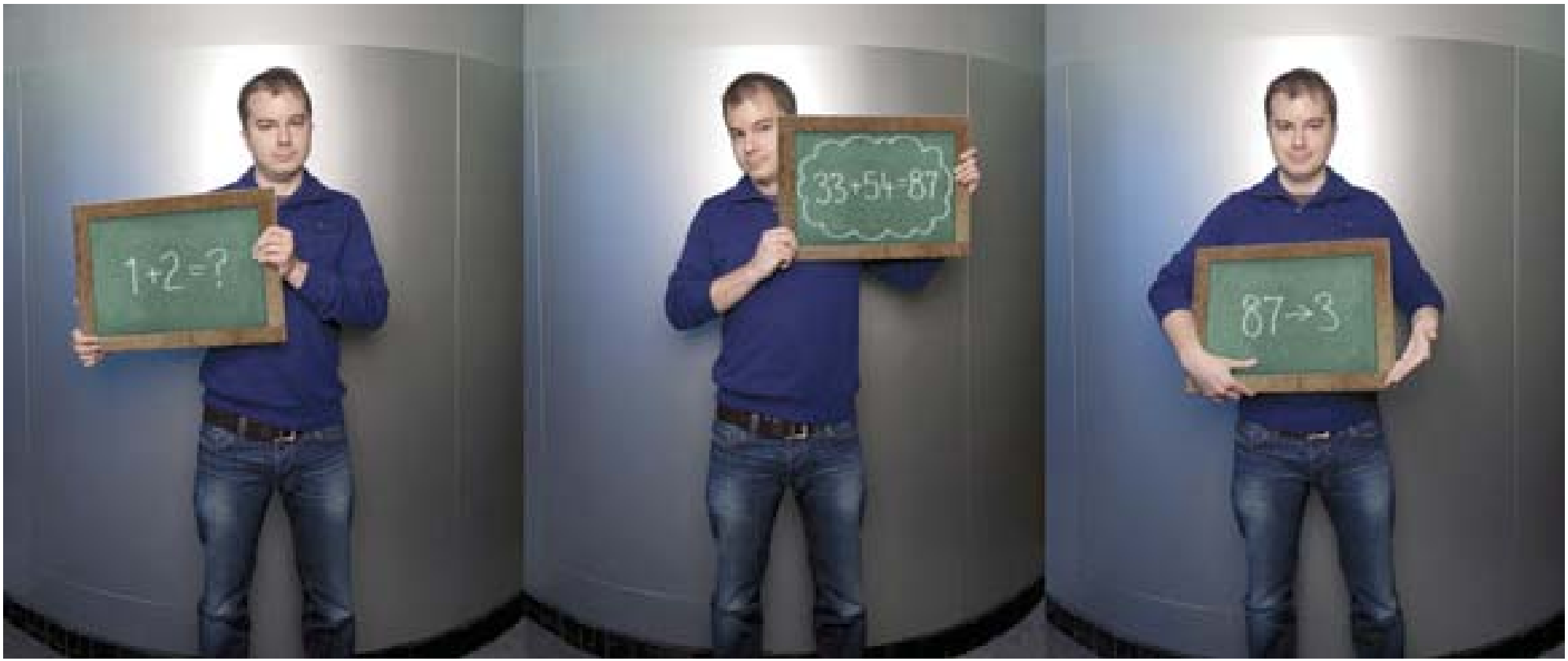
Fully Homomorphic Encryption (FHE)

Fully Homomorphic Encryption (FHE)

Gentry 09: A Lattice-based cryptosystem that is fully homomorphic

Fully Homomorphic Encryption (FHE)

Gentry 09: A Lattice-based cryptosystem that is fully homomorphic



Fully Homomorphic Encryption (FHE)

Gentry 09: A Lattice-based cryptosystem that is fully homomorphic

Fully Homomorphic Encryption (FHE)

Gentry 09: A Lattice-based cryptosystem that is fully homomorphic

3278764873692384398479478394739487234997938798370947005983095830958094850349850498487 9ut9875937493

+ %^&&£££$%

590094867-3498674-0967590674589764597 65-9067459685489765498765468978745943580487568760876508457095

Fully Homomorphic Encryption (FHE)

Gentry 09: A Lattice-based cryptosystem that is fully homomorphic

Long Key
Complicated Server Operations
Computational Security

Applications

Secure Cloud

Remote File Storage

Secure Multi Party Computation

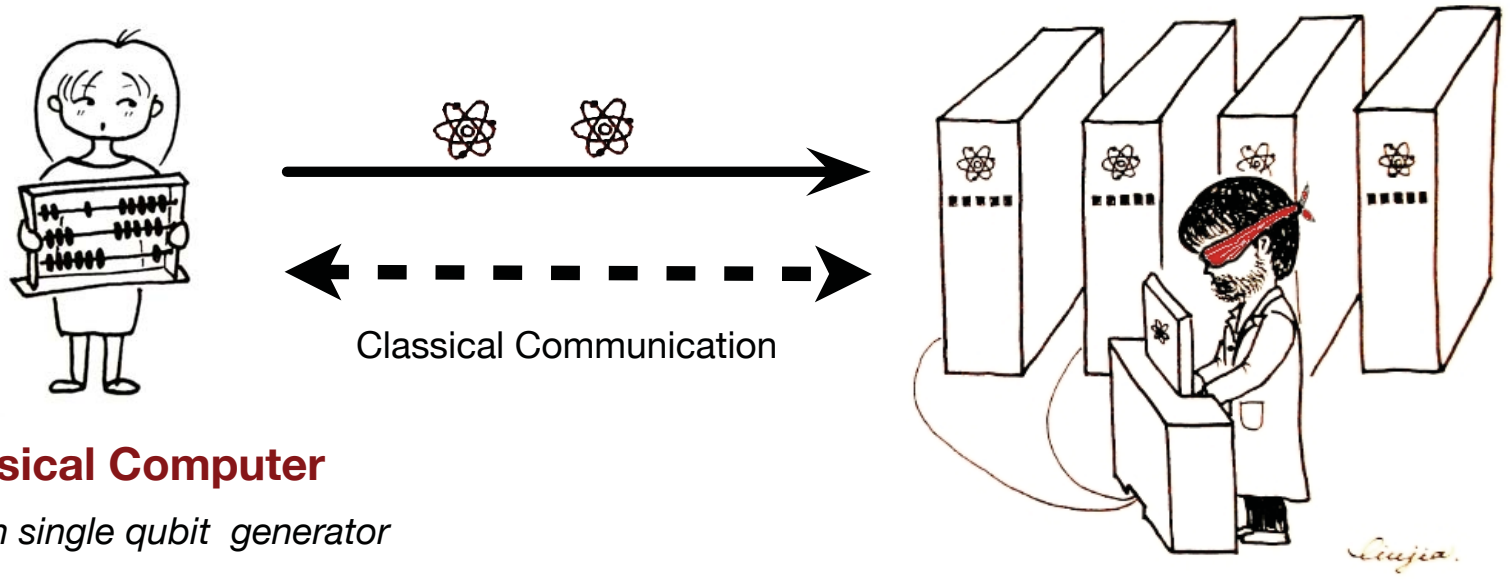
Verification of Outsourced Computation

Short Proof of Knowledge

....

Universal Blind Quantum Computing (UBQC)

Broadbent, Fitzsimons and Kashefi 09: Quantum Key Distribution + Quantum Teleportation



Classical Computer

random single qubit generator

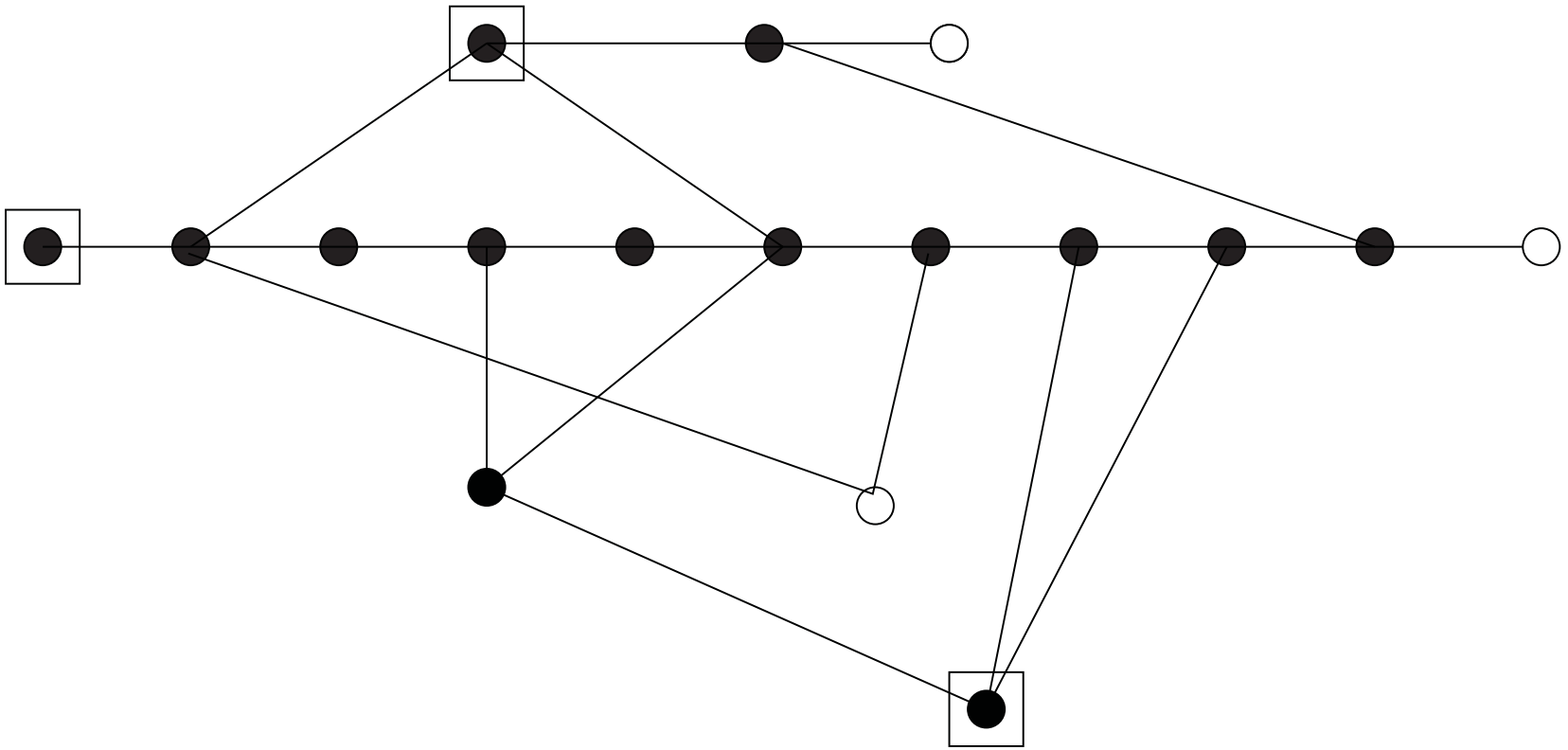
Unconditional Perfect Privacy

Server learns nothing about client's input/output/computation

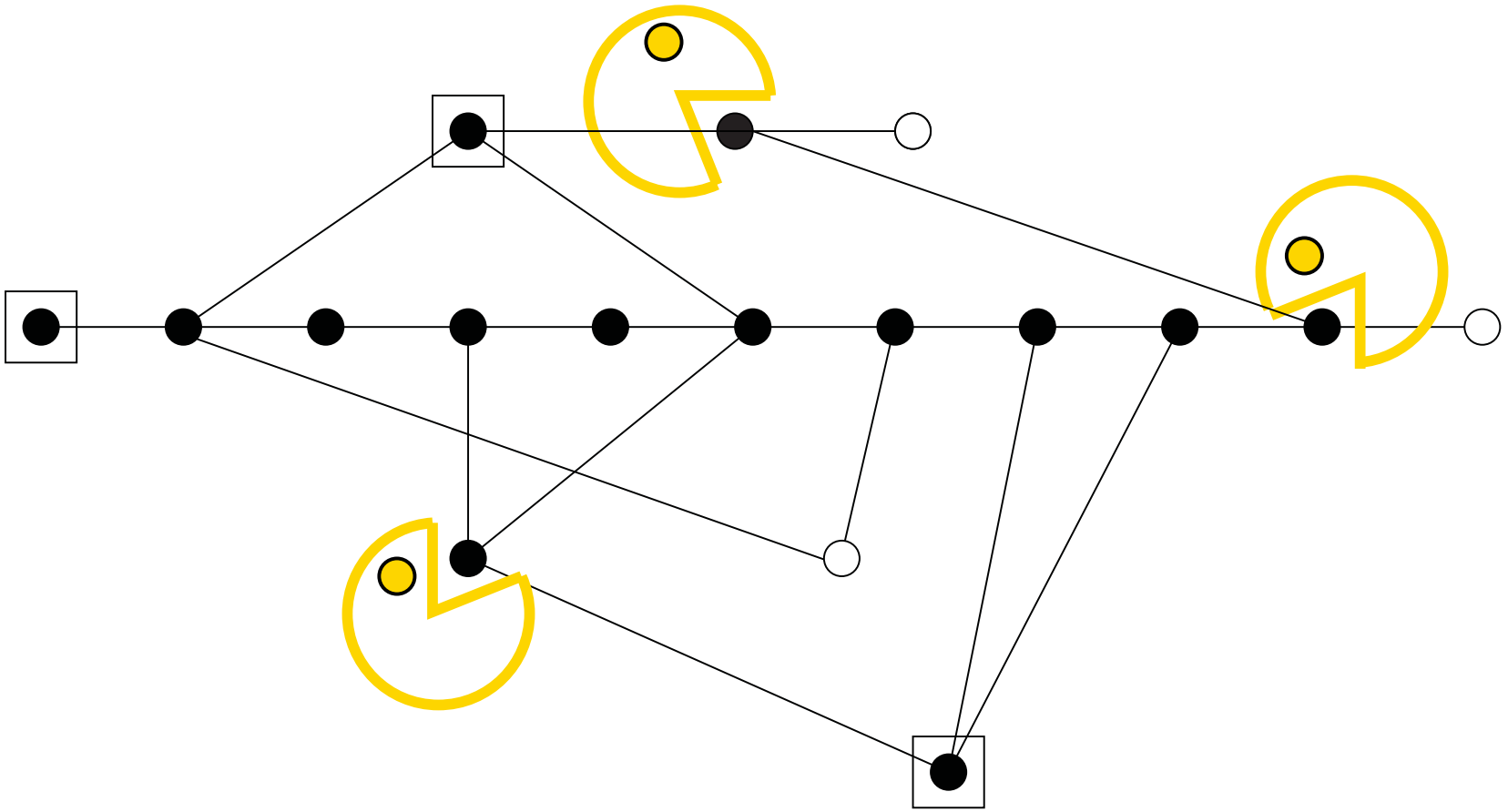
Measurement-based QC

- New qubits, to prepare the auxiliary qubits: N
- Entanglements, to build the quantum channel: E
- Measurements, to propagate (manipulate) qubits: M
- Corrections, to make the computation deterministic: C

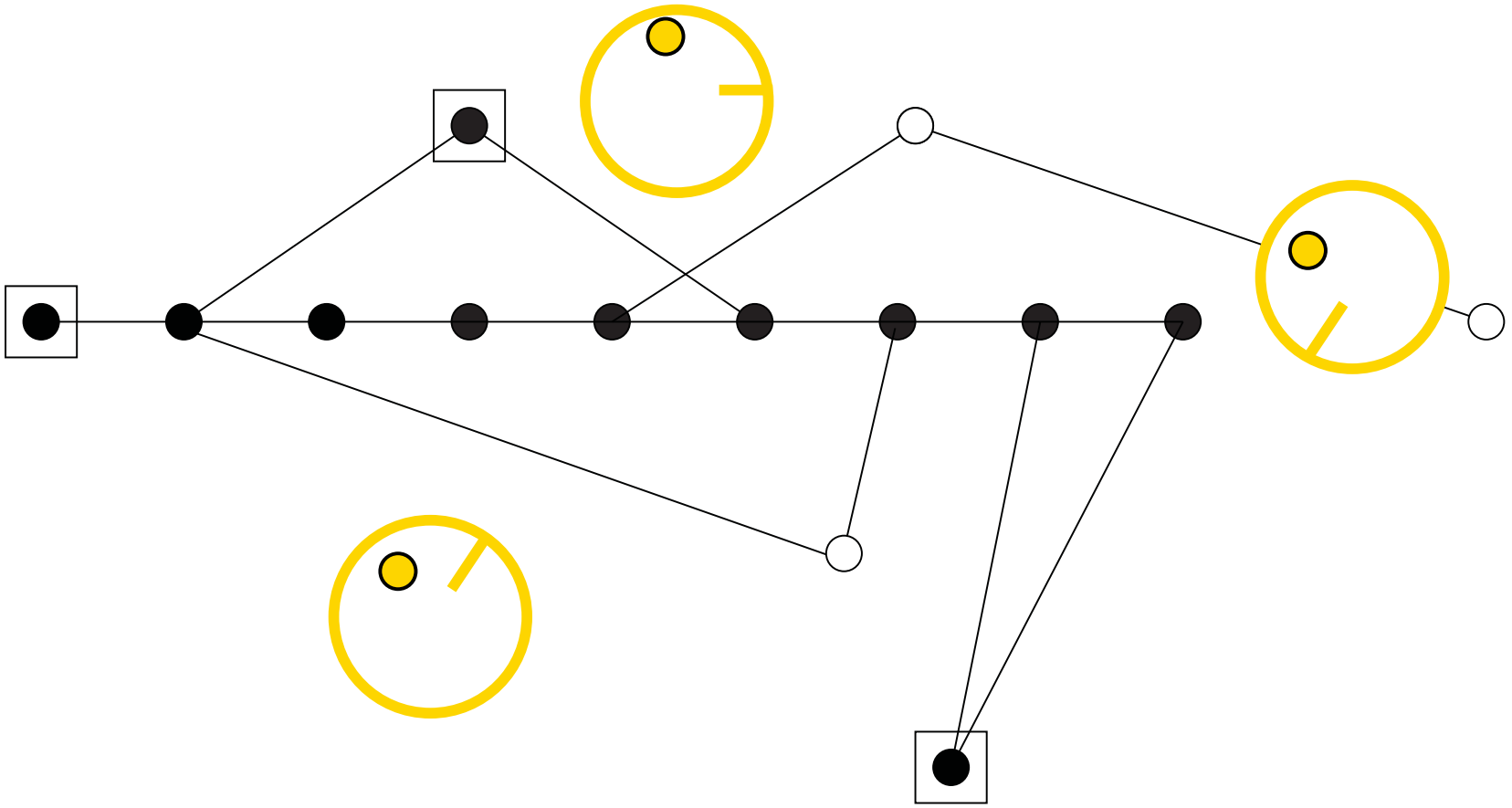
Quantum Pacman



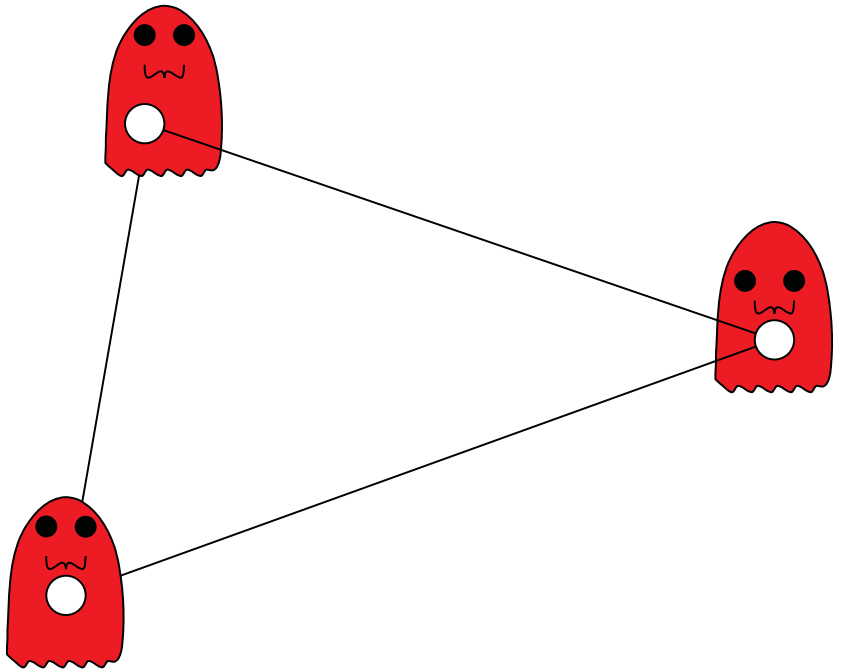
Quantum Pacman



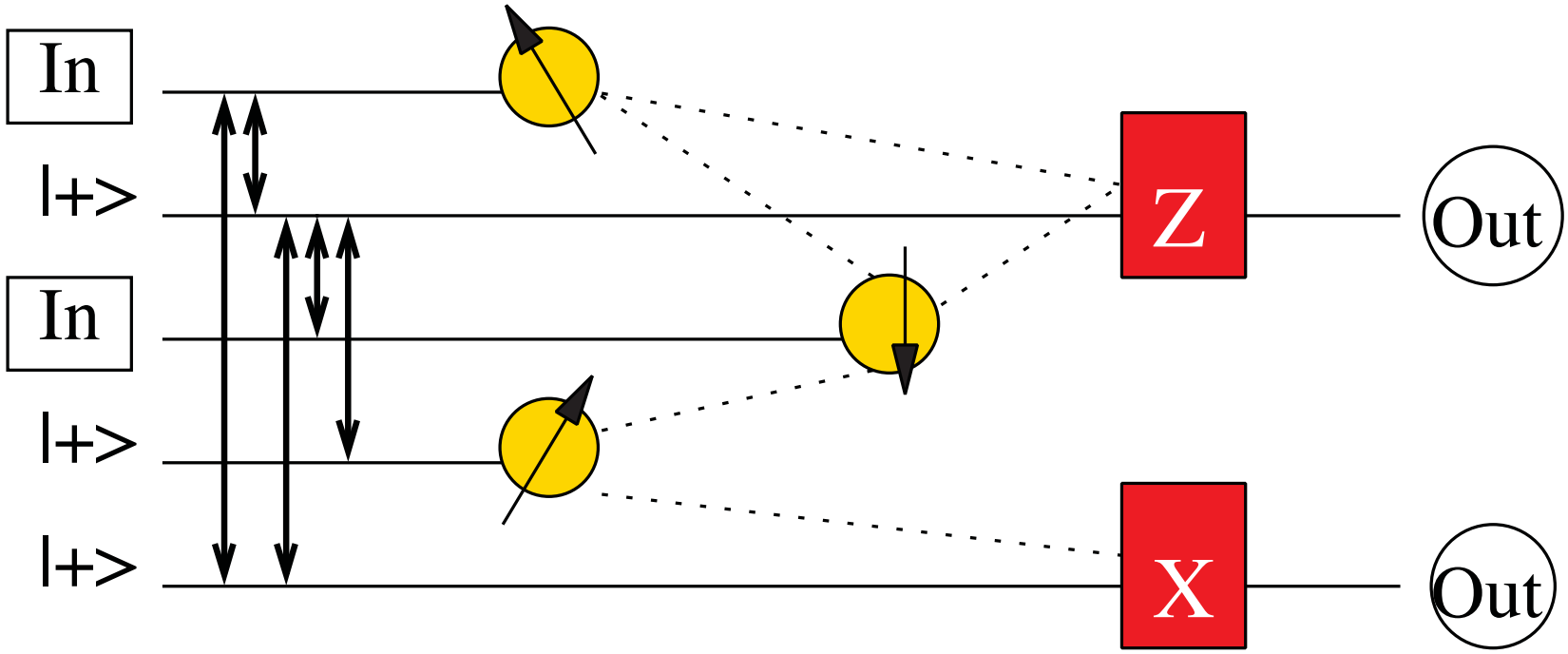
Quantum Pacman



Quantum Pacman

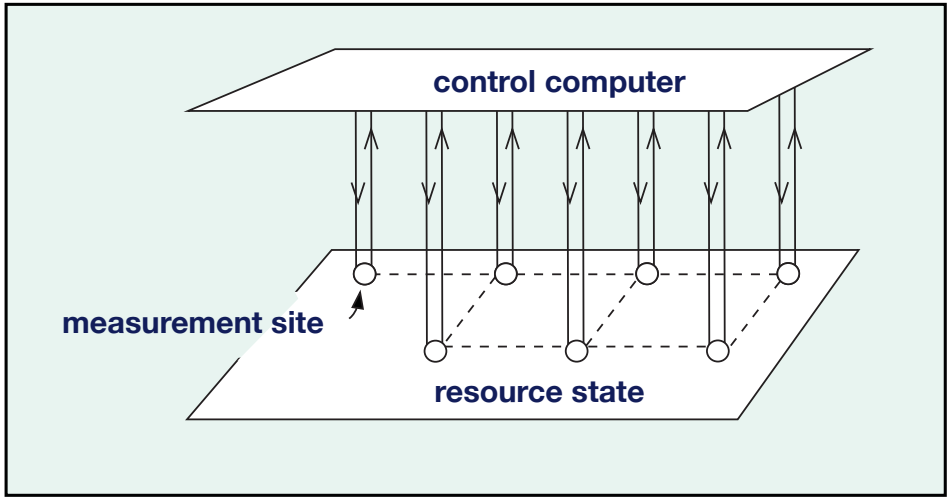


Circuit Picture



Formal Calculus

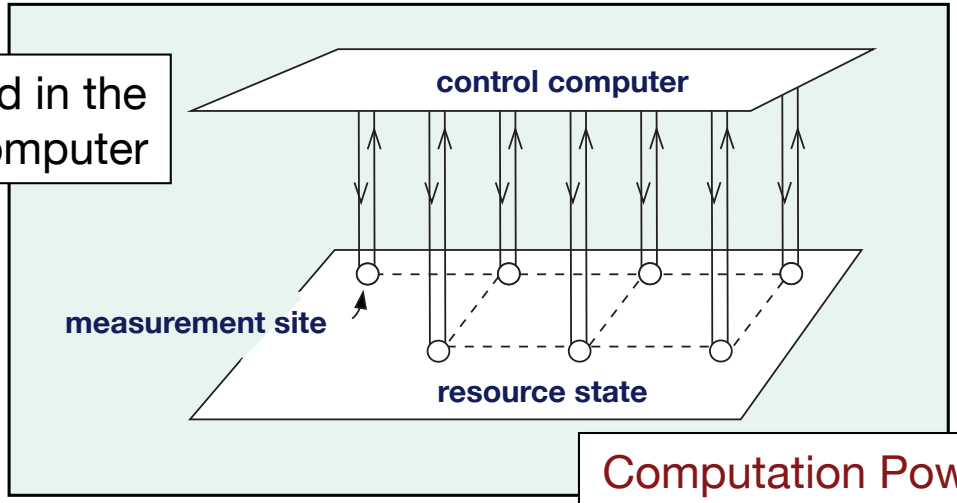
Kashefi et.al. Measurement Calculus JACM 2007



Formal Calculus

Kashefi et.al. Measurement Calculus JACM 2007

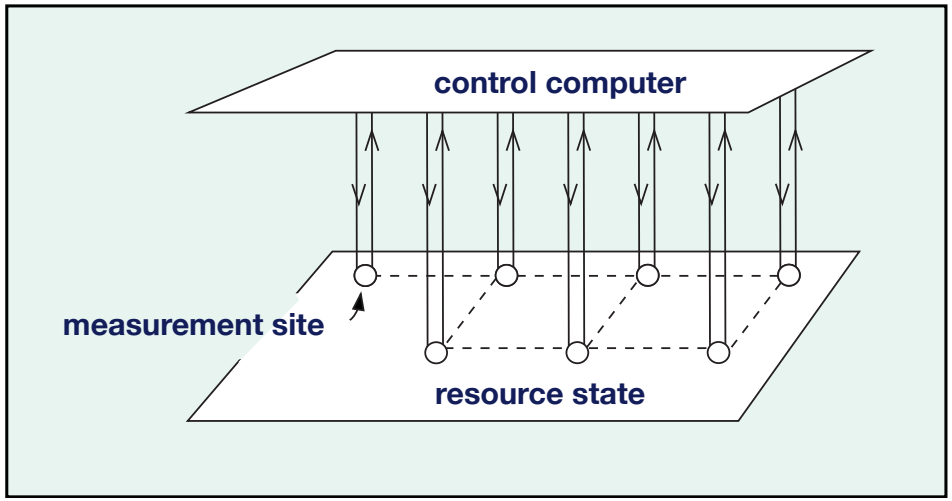
Program is encoded in the classical control computer



Computation Power is encoded in the quantum entanglement

Hiding in MBQC

Program is encoded in the classical control computer
Computation Power is encoded in the entanglement

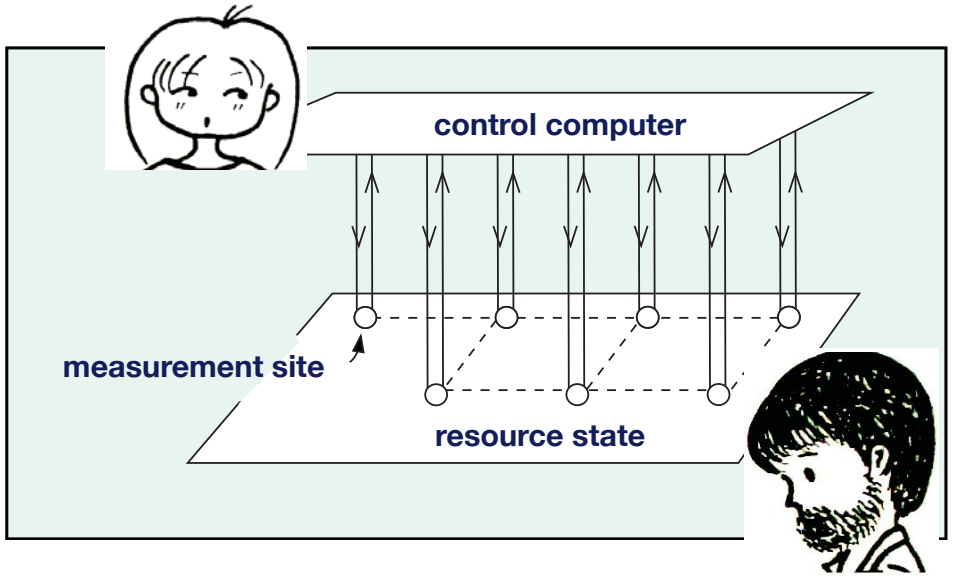


Raussendorf and Briegel, PRL, 2001

Danos, Panangaden, Kashefi, JACM, 2009

Hiding in MBQC

Program is encoded in the classical control computer
Computation Power is encoded in the entanglement



Raussendorf and Briegel, PRL, 2001

Danos, Panangaden, Kashefi, JACM, 2009

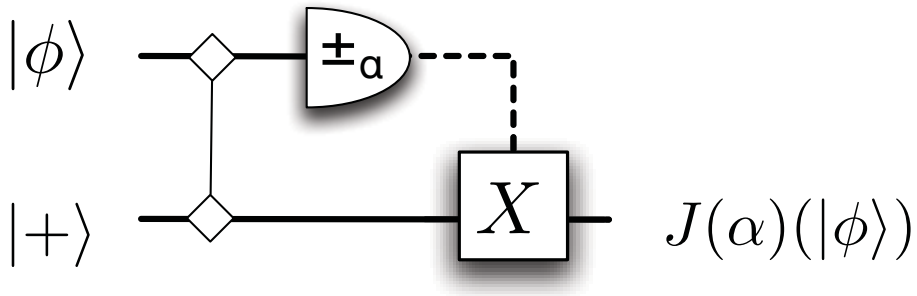
Hiding in MBQC

$$J(\alpha) \;:=\; \tfrac{1}{\sqrt{2}} \begin{pmatrix} 1 & e^{i\alpha} \\ 1 & -e^{i\alpha} \end{pmatrix}$$

Hiding in MBQC

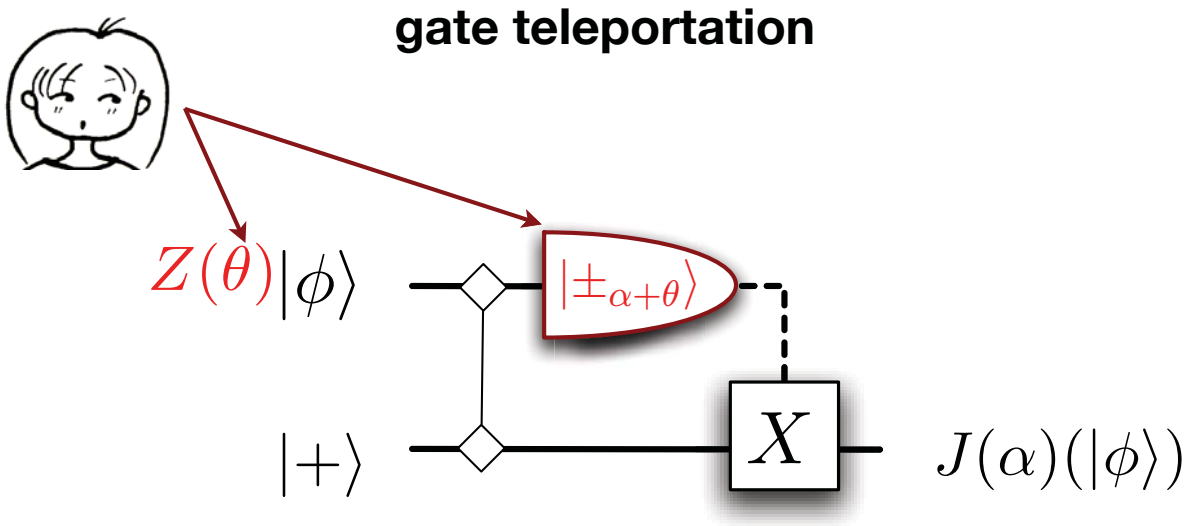
$$J(\alpha) \;:=\; \tfrac{1}{\sqrt{2}} \begin{pmatrix} 1 & e^{i\alpha} \\ 1 & -e^{i\alpha} \end{pmatrix}$$

gate teleportation



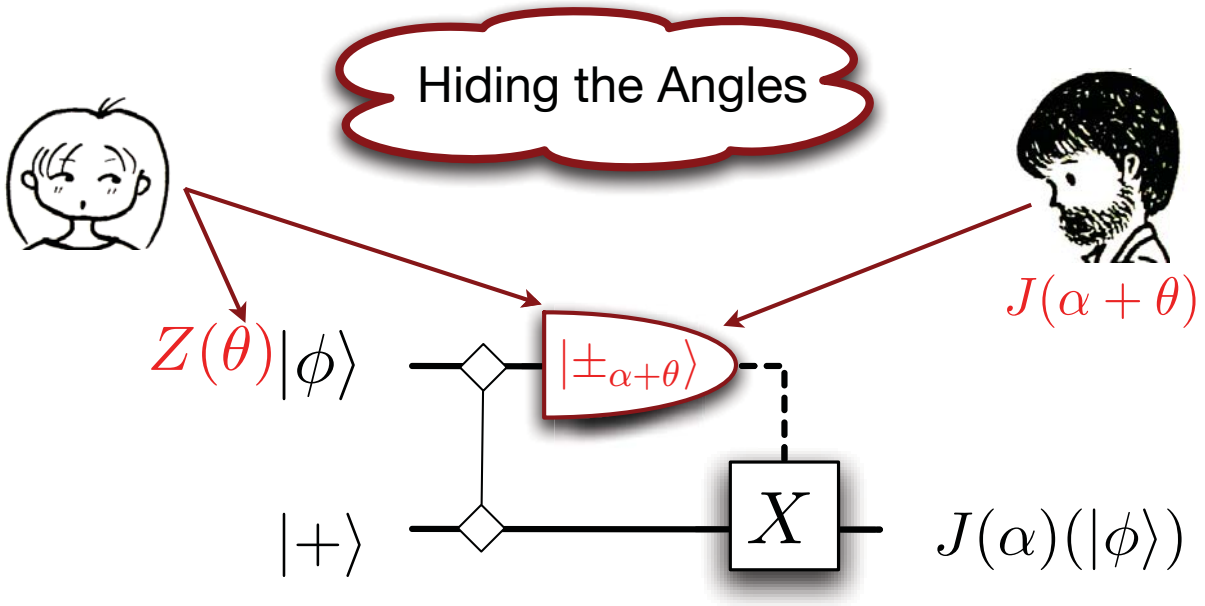
Hiding in MBQC

$$J(\alpha) := \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & e^{i\alpha} \\ 1 & -e^{i\alpha} \end{pmatrix}$$



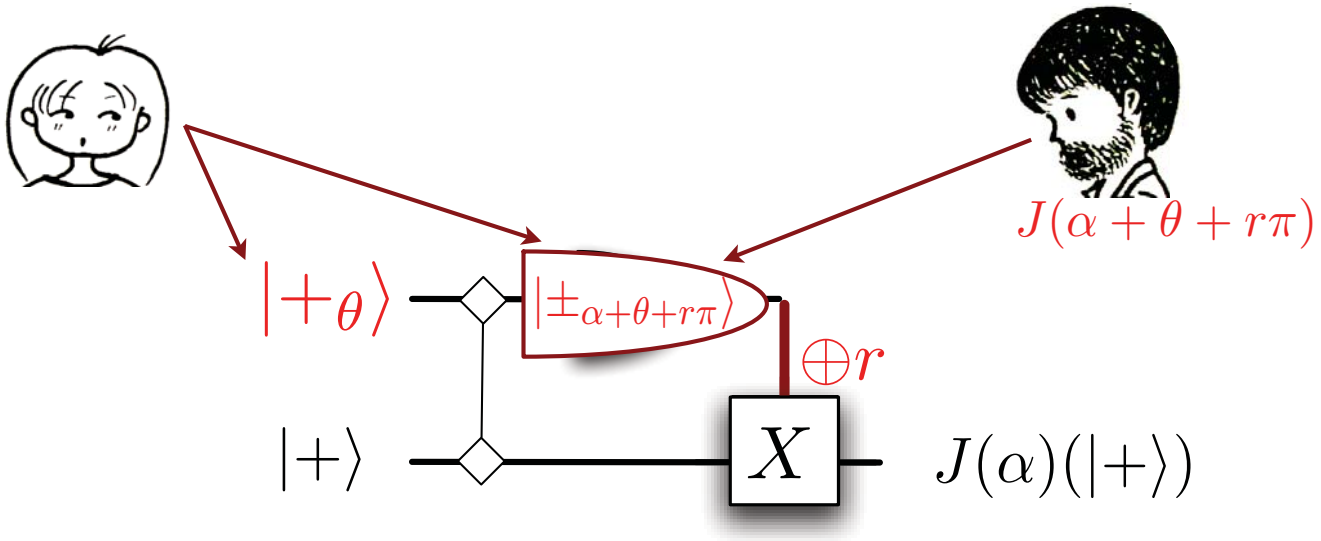
Hiding in MBQC

$$J(\alpha) := \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & e^{i\alpha} \\ 1 & -e^{i\alpha} \end{pmatrix}$$



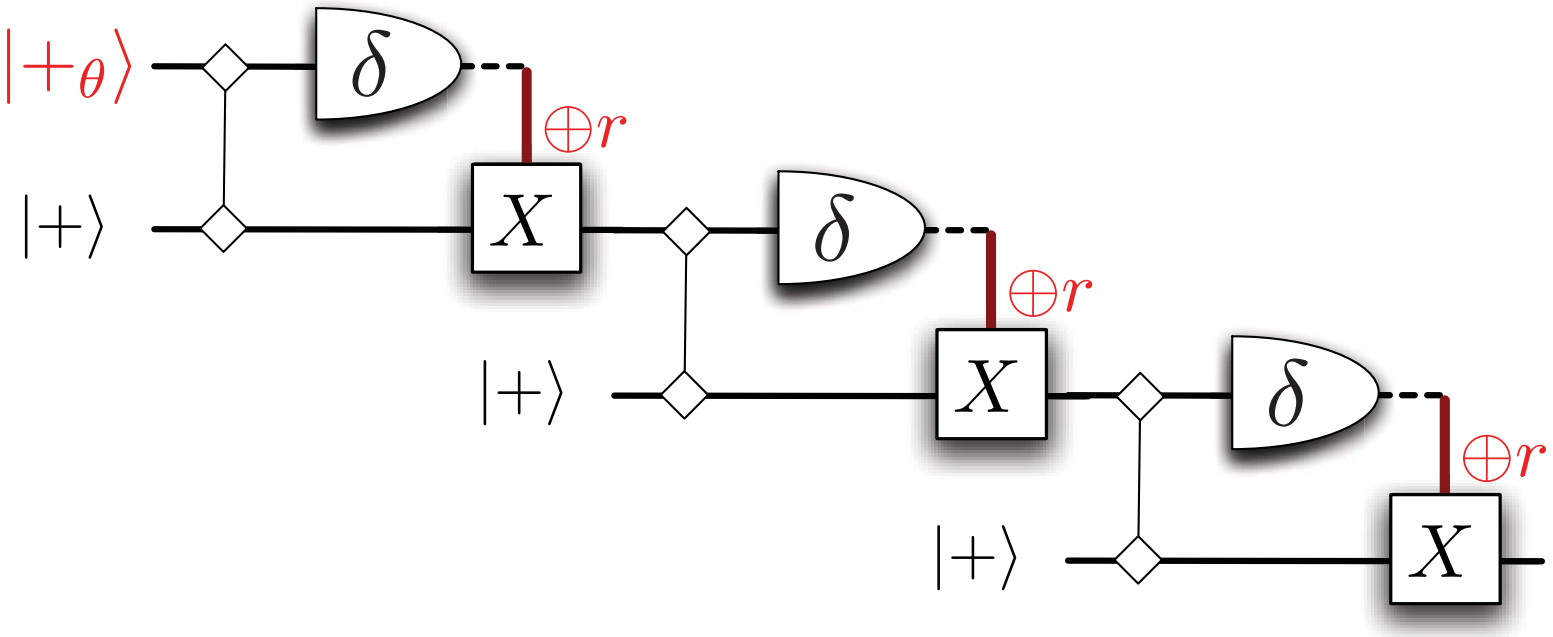
Thinking inside the box

$$J(\alpha) := \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & e^{i\alpha} \\ 1 & -e^{i\alpha} \end{pmatrix}$$



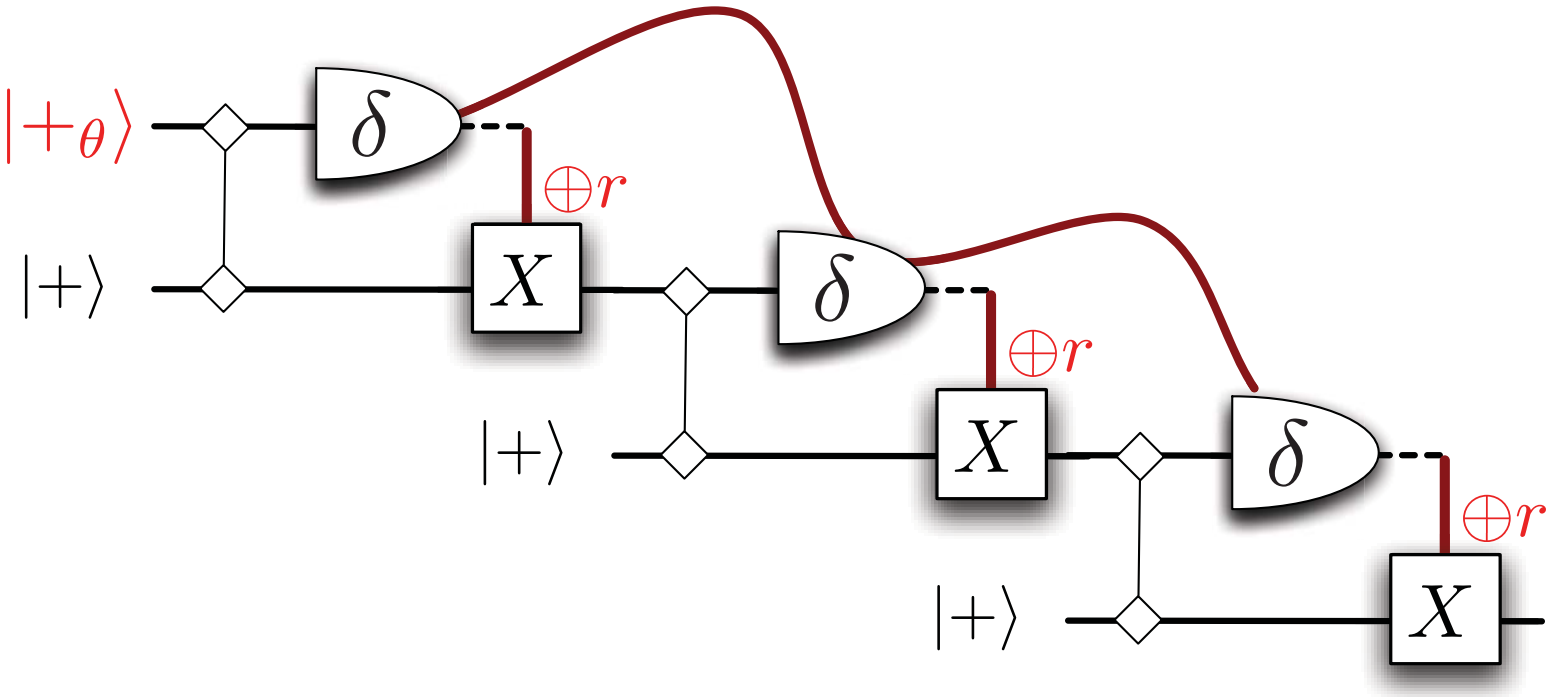
Hiding the measurement result

Gates Composition



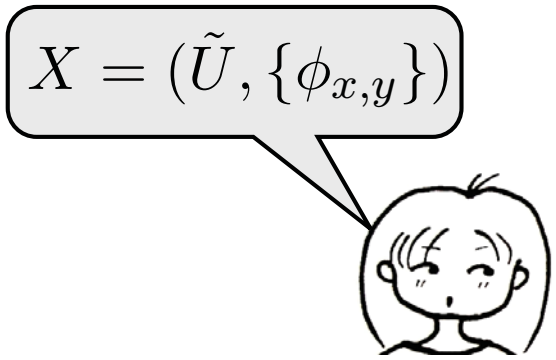
Client-Server interactions

Gates Composition

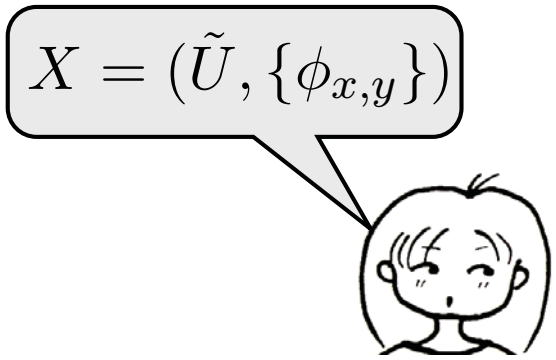


Client-Server interactions

Universal Blind Quantum Computing



Universal Blind Quantum Computing

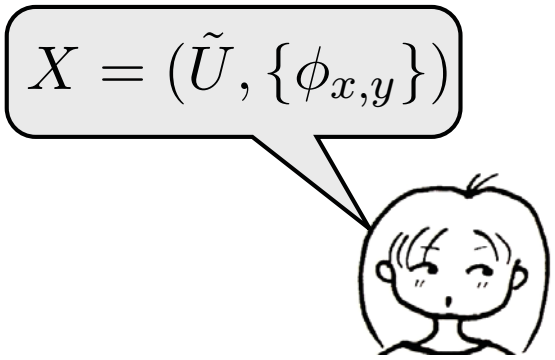


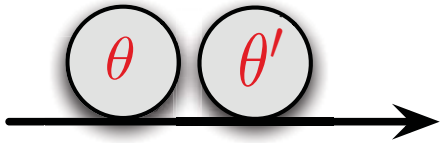
random single qubit generator

$$1/\sqrt{2}\left(|0\rangle + e^{i\theta}\,|1\rangle\right)$$

$$\theta = 0, \pi/4, 2\pi/4, \ldots, 7\pi/4$$

Universal Blind Quantum Computing



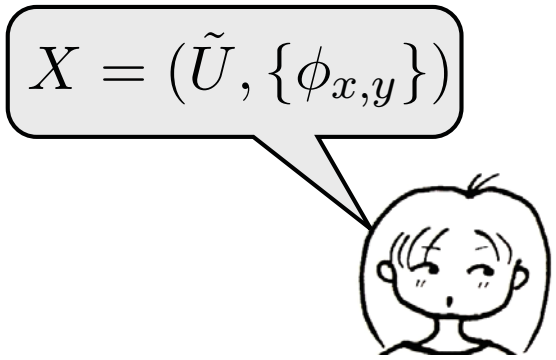


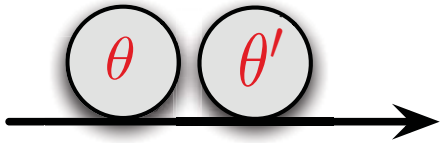
random single qubit generator

$$1/\sqrt{2}\left(|0\rangle + e^{i\theta}\,|1\rangle\right)$$

$$\theta = 0, \pi/4, 2\pi/4, \ldots, 7\pi/4$$

Universal Blind Quantum Computing

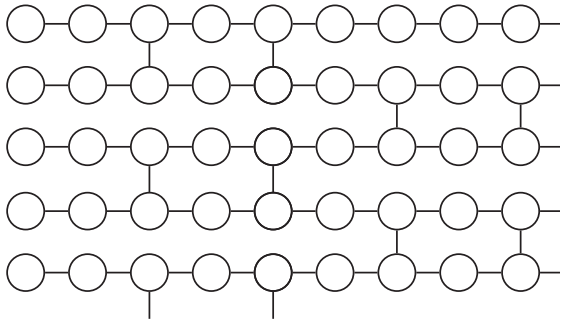




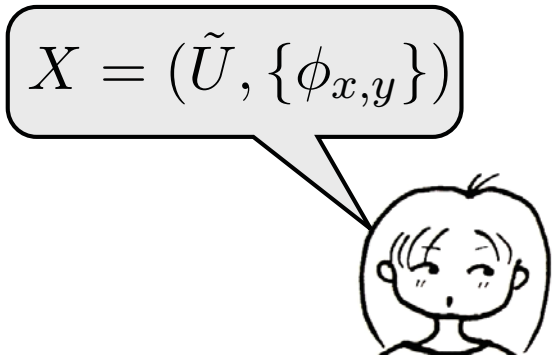
random single qubit generator

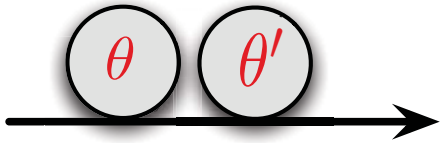
$$\left|1\right\rangle/\sqrt{2}\left(\left|0\right\rangle + e^{i\theta}\left|1\right\rangle\right)$$

$$\theta = 0, \pi/4, 2\pi/4, \ldots, 7\pi/4$$



Universal Blind Quantum Computing





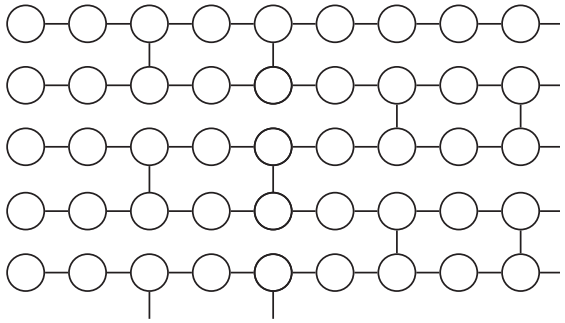
random single qubit generator

$$1/\sqrt{2}\left(|0\rangle + e^{i\theta}\,|1\rangle\right)$$

$$\theta = 0, \pi/4, 2\pi/4, \ldots, 7\pi/4$$

$$r_{x,y} \in_R \{0,1\}$$

$$\delta_{x,y} = \phi'_{x,y} + \theta_{x,y} + \pi r_{x,y}$$



Universal Blind Quantum Computing

$X = (\tilde{U}, \{\phi_{x,y}\})$

random single qubit generator

$$1/\sqrt{2}\left(|0\rangle + e^{i\theta}|1\rangle\right)$$

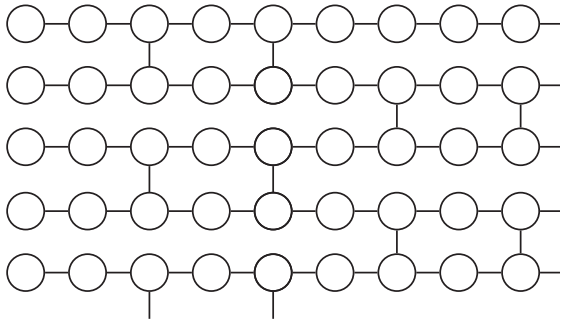
$$\theta = 0, \pi/4, 2\pi/4, \ldots, 7\pi/4$$

θ θ'

$\delta_{x,y}$

$r_{x,y} \in_R \{0,1\}$

$$\delta_{x,y} = \phi'_{x,y} + \theta_{x,y} + \pi r_{x,y}$$



Universal Blind Quantum Computing

$X = (\tilde{U}, \{\phi_{x,y}\})$

random single qubit generator

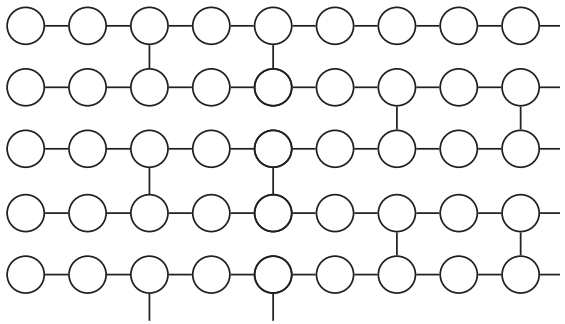
$$1/\sqrt{2}\left(|0\rangle + e^{i\theta}|1\rangle\right)$$
$$\theta = 0, \pi/4, 2\pi/4, \ldots, 7\pi/4$$

θ θ'

$\delta_{x,y}$

$r_{x,y} \in_R \{0,1\}$

$$\delta_{x,y} = \phi'_{x,y} + \theta_{x,y} + \pi r_{x,y}$$



$$\left\{\left|+\delta_{x,y}\right\rangle, \left|-\delta_{x,y}\right\rangle\right\}$$

Universal Blind Quantum Computing

$X = (\tilde{U}, \{\phi_{x,y}\})$

random single qubit generator

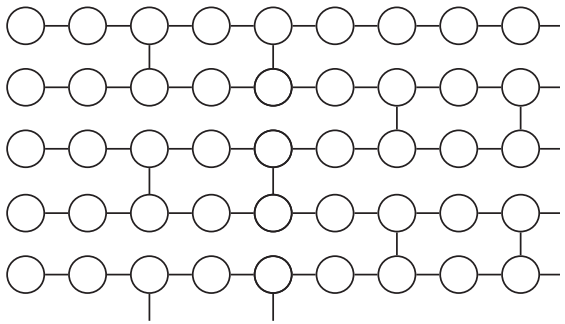
$$1/\sqrt{2}\left(|0\rangle + e^{i\theta}|1\rangle\right)$$
$$\theta = 0, \pi/4, 2\pi/4, \ldots, 7\pi/4$$

θ θ'

$\delta_{x,y}$

$r_{x,y} \in_R \{0,1\}$

$\delta_{x,y} = \phi'_{x,y} + \theta_{x,y} + \pi r_{x,y}$



$s_{x,y} \in \{0,1\}$

$\{|+_{\delta_{x,y}}\rangle, |-_{\delta_{x,y}}\rangle\}$

Universal Blind Quantum Computing

$X = (\tilde{U}, \{\phi_{x,y}\})$

random single qubit generator

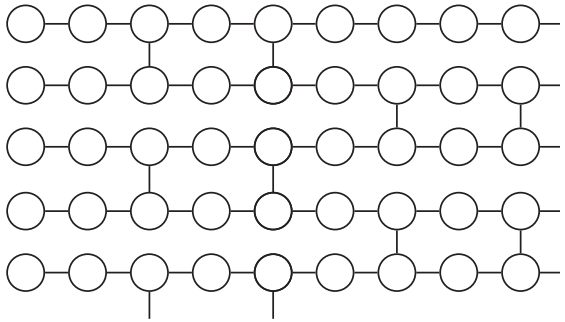
$$1/\sqrt{2}\left(|0\rangle + e^{i\theta}|1\rangle\right)$$
$$\theta = 0, \pi/4, 2\pi/4, \ldots, 7\pi/4$$

θ θ'

$\delta_{x,y}$

$r_{x,y} \in_R \{0,1\}$

$$\delta_{x,y} = \phi'_{x,y} + \theta_{x,y} + \pi r_{x,y}$$



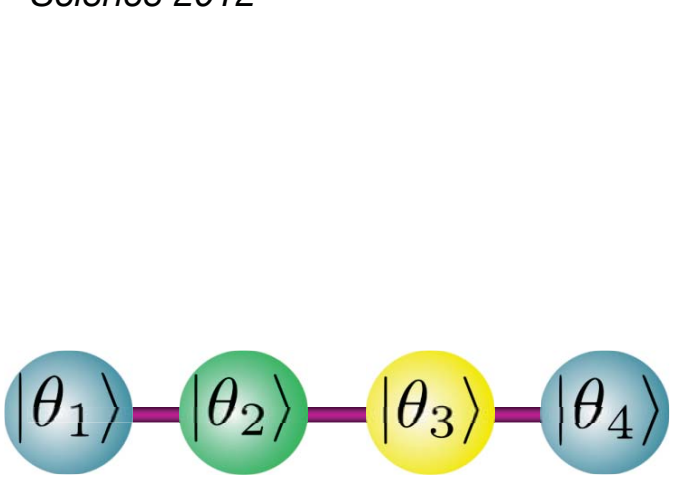
$\{|+_{\delta_{x,y}}\rangle, |-_{\delta_{x,y}}\rangle\}$

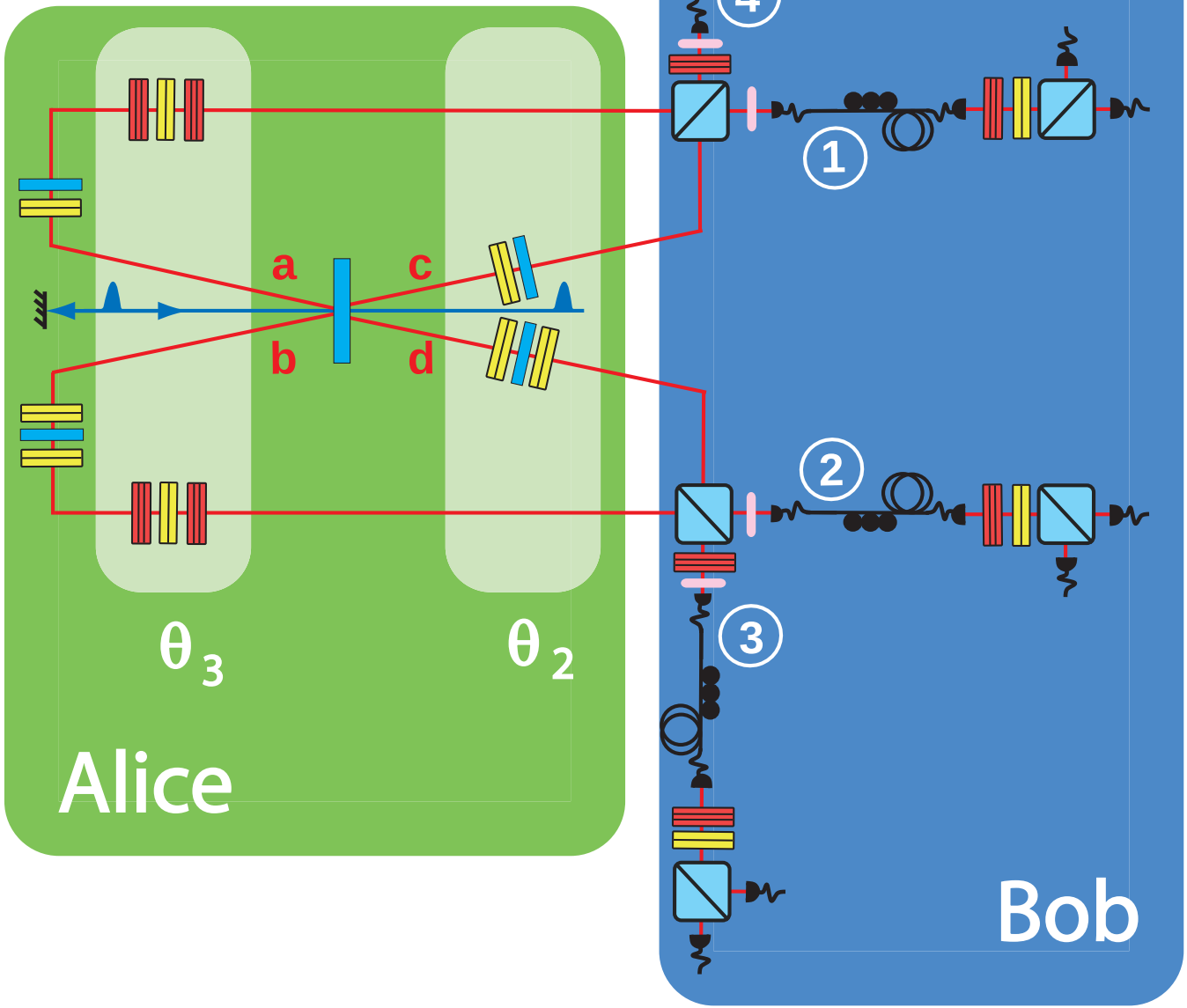
$s_{x,y} \in \{0,1\}$

$s_{x,y} := s_{x,y} + r_{x,y}$

Experimental Implementation

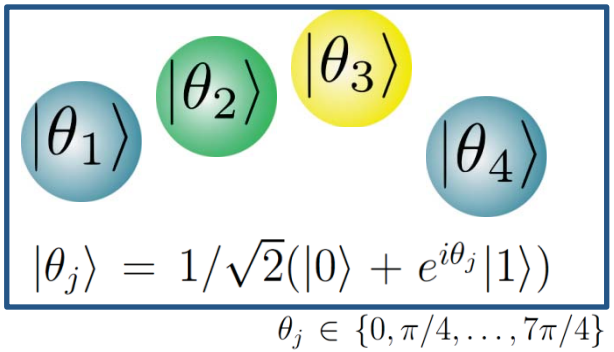
S. Barz, E. Kashefi, A. Broadbent, J. Fitzsimons, A. Zeilinger, P. Walther, Science 2012





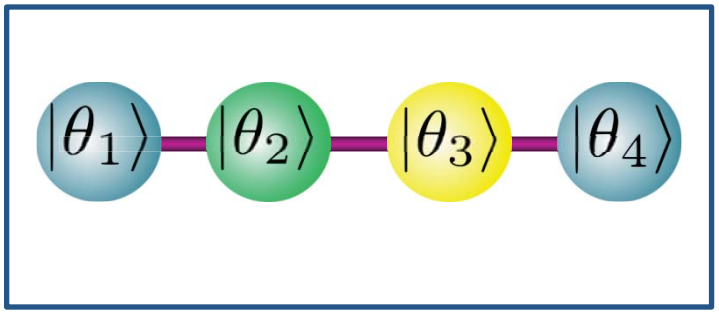
Experimental Implementation

Client:
limited computational power



Prepares random qubits

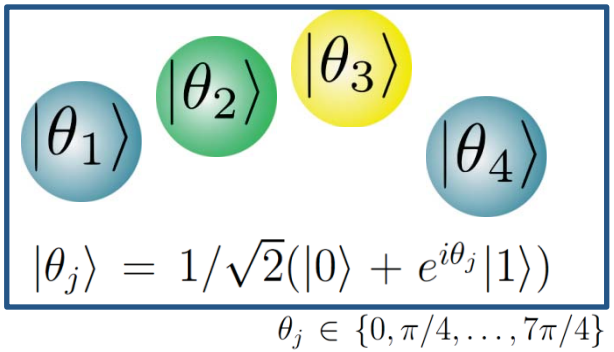
Quantum server:
full power of Quantum Computation



Entangles qubits

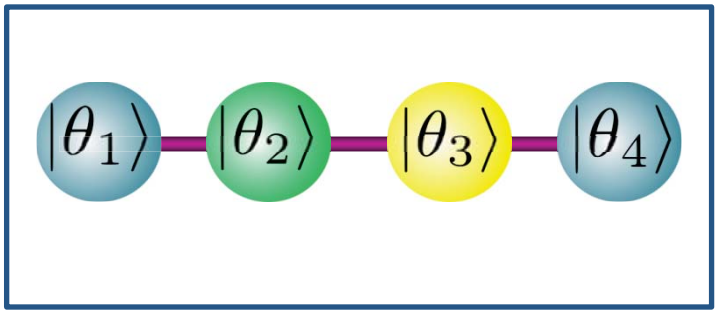
Experimental Implementation

Client:
limited computational power



Prepares random qubits

Quantum server:
full power of Quantum Computation



Entangles qubits

$$\delta_j = \theta_j + \phi_j + \pi r_j$$

Computes measurement angles

- Measurement instruction for server
- Initial rotation of the qubit
- Target rotation
- Random bit flip

Verification

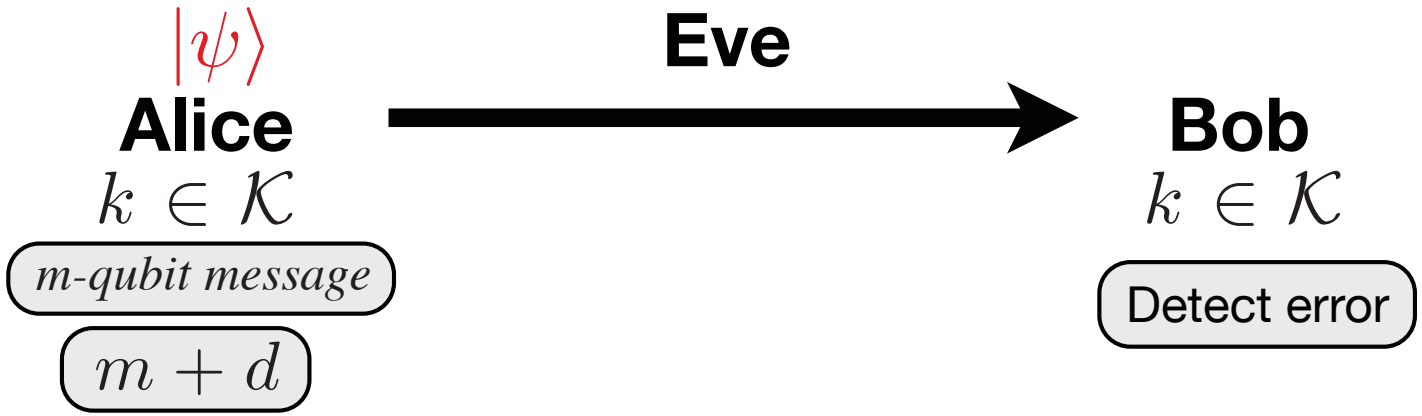
Verification

- Correctness: in the absence of any interference, client accepts and the output is correct

- Soundness: Verifier rejects an incorrect output, except with probability at most exponentially small in the security parameter

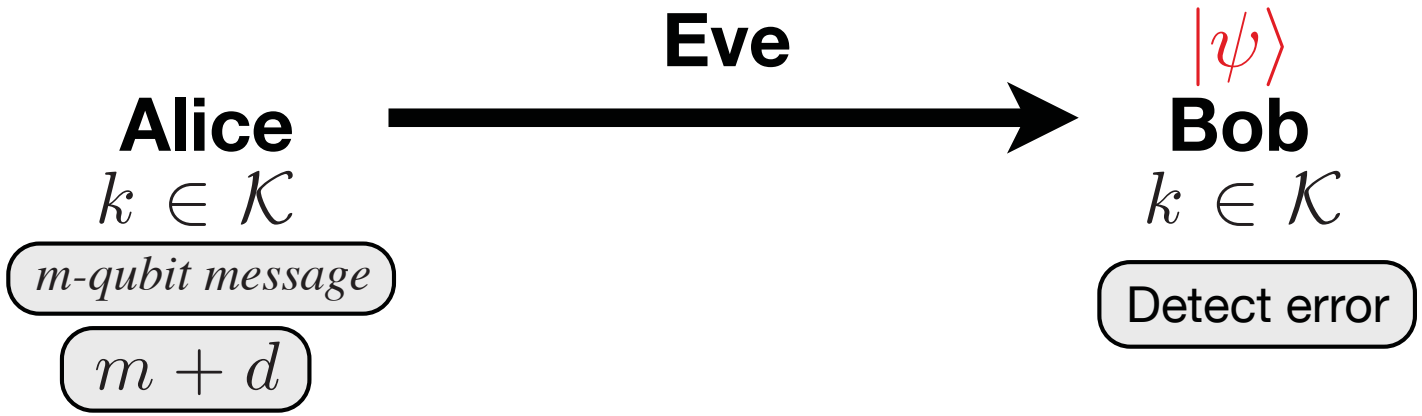
Fitzsimons and Kashefi, arXiv:1203.5217, 2012

Verification vs Authentication



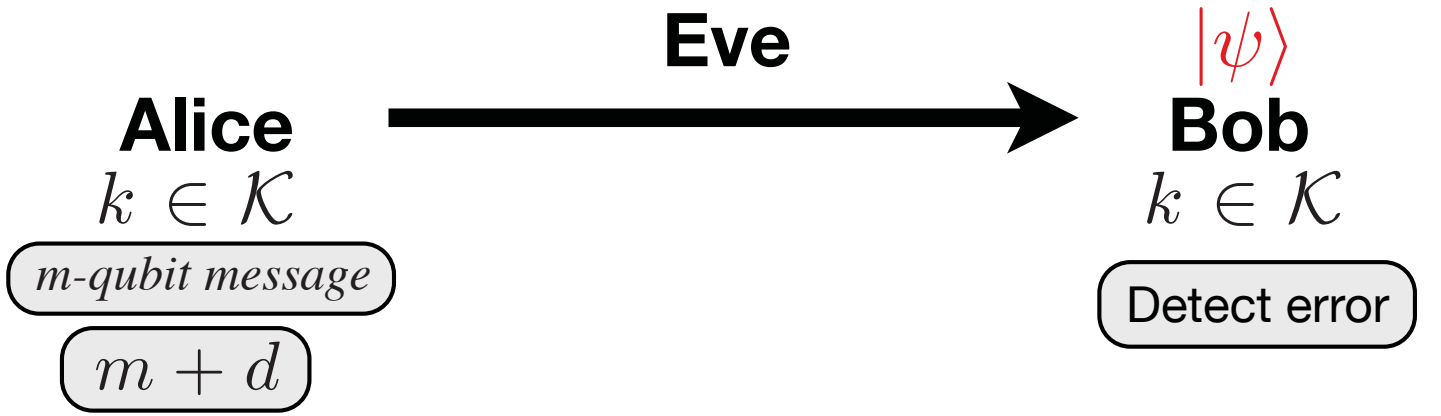
Barnum, Crepeau, Gottesman, Smith and Tapp, FOCS02

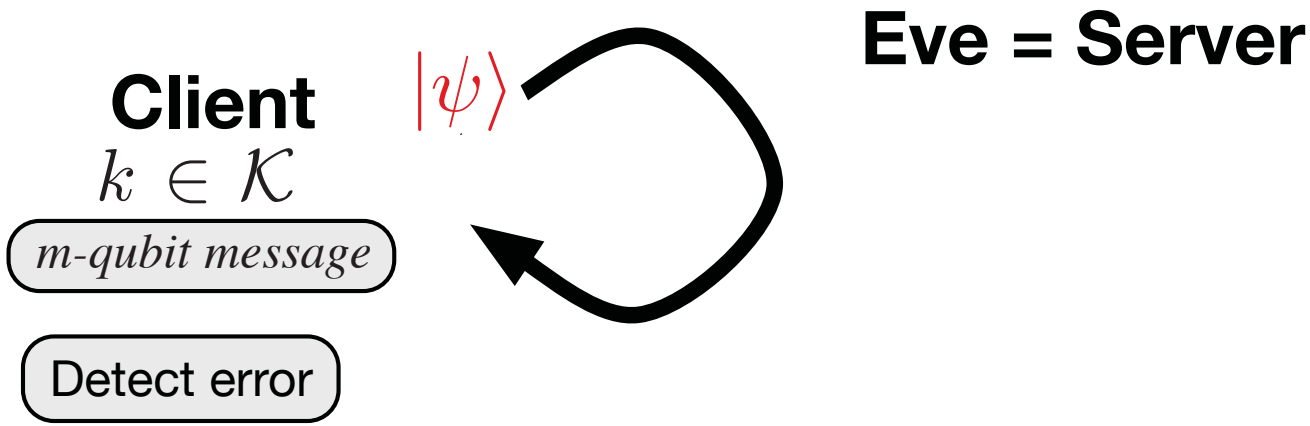
Verification vs Authentication



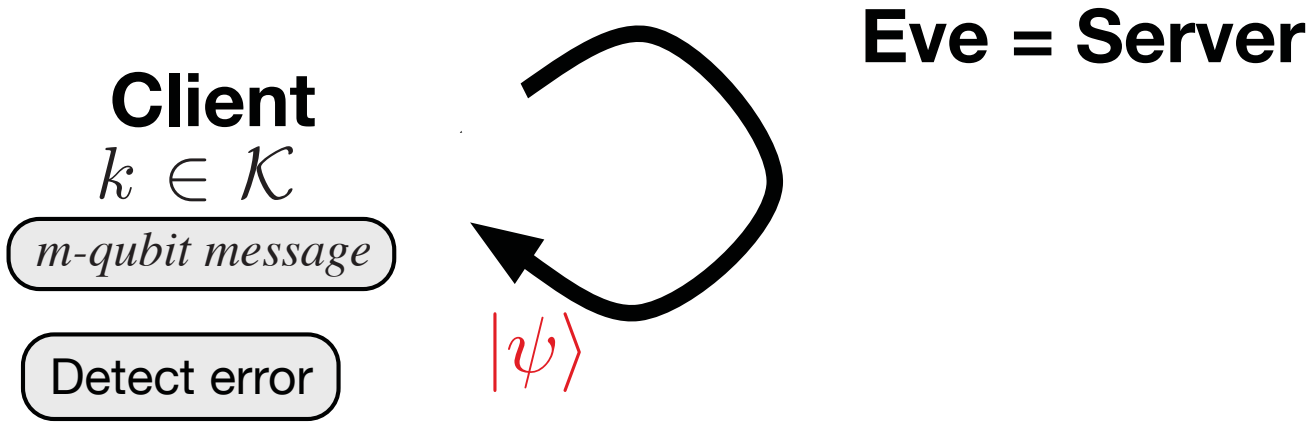
Barnum, Crepeau, Gottesman, Smith and Tapp, FOCS02

Verification vs Authentication

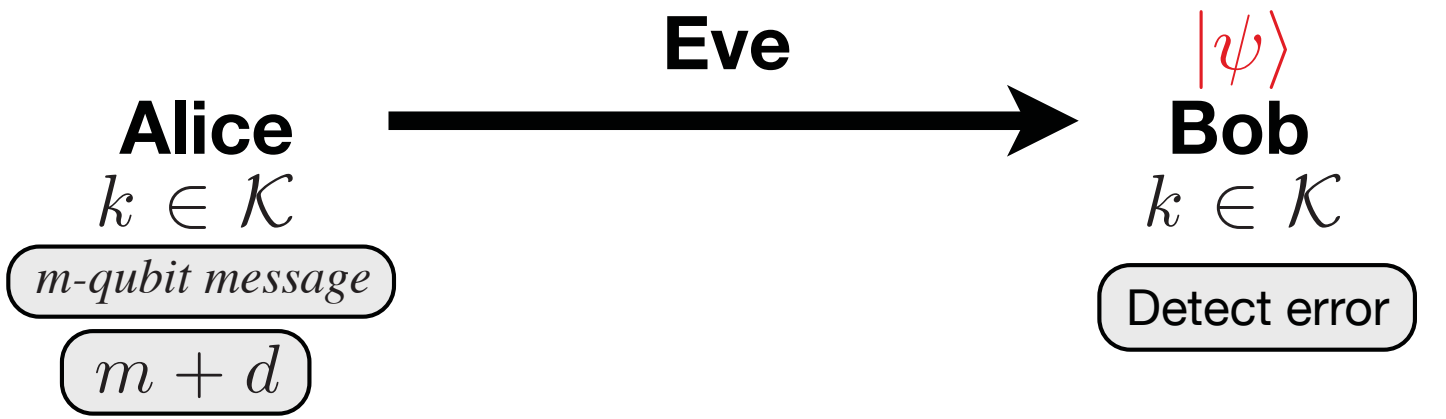


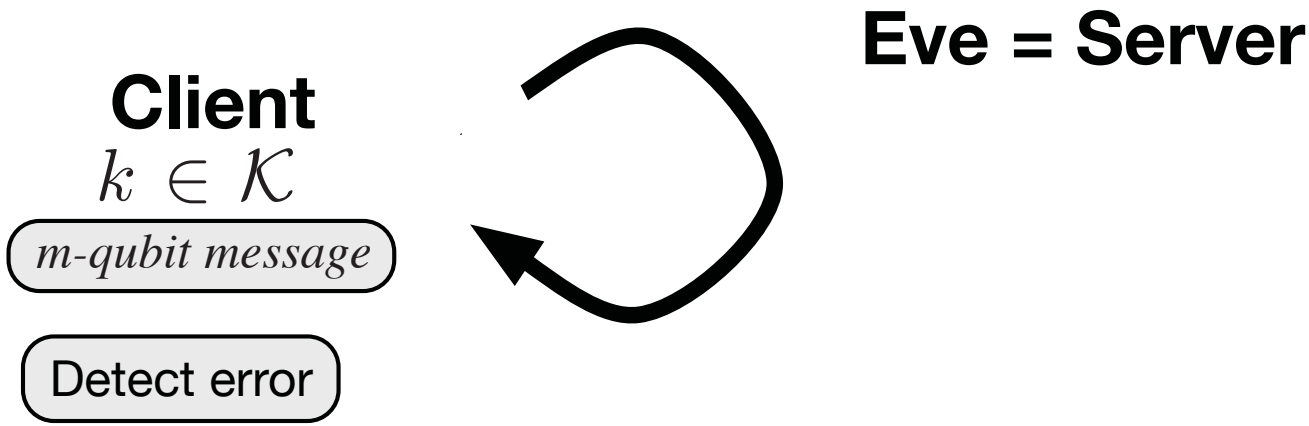


Verification vs Authentication

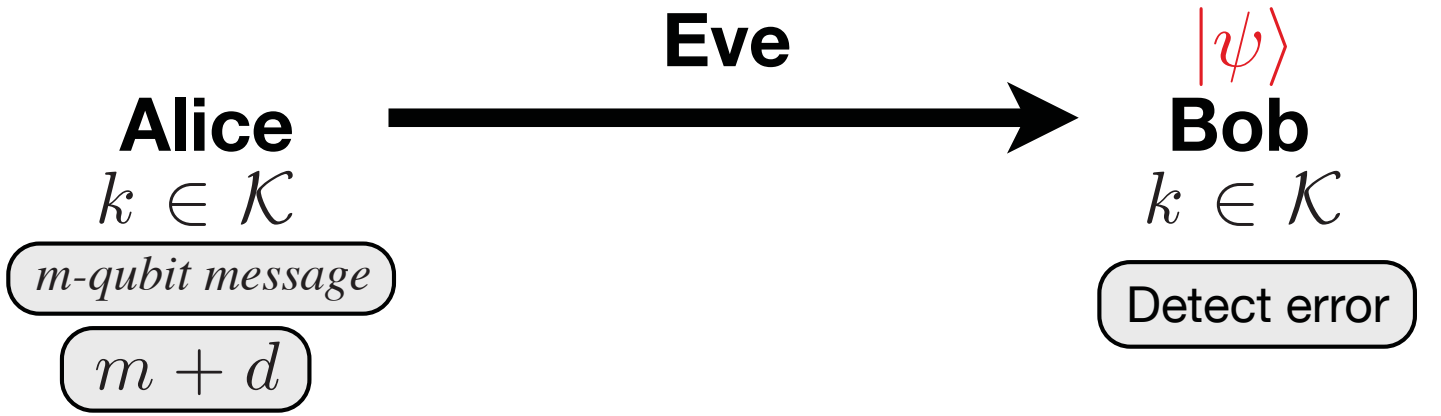


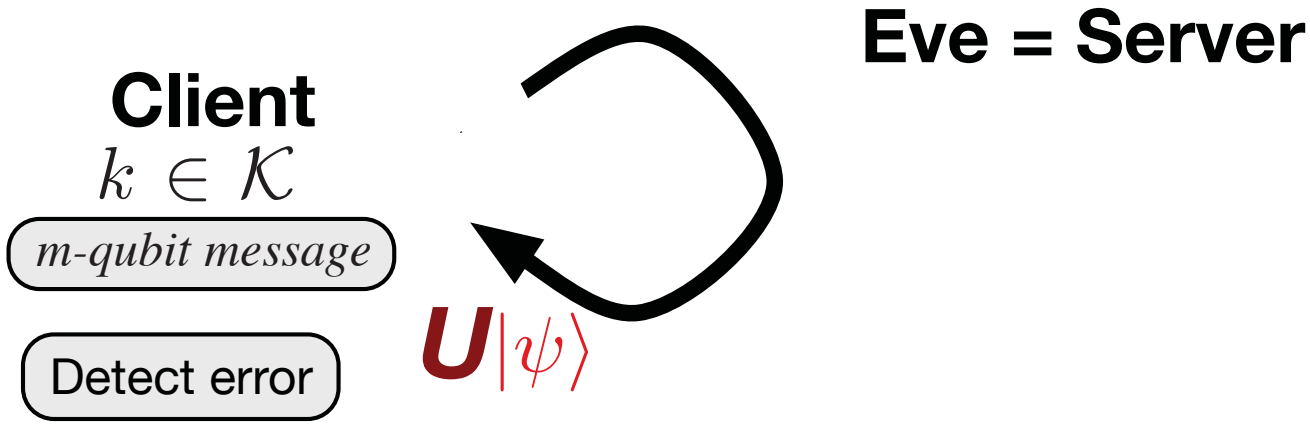
Verification vs Authentication





Verification vs Authentication





ε-Verification

Alice

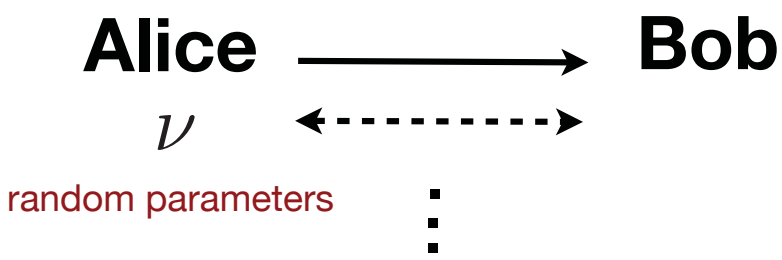
ν

random parameters

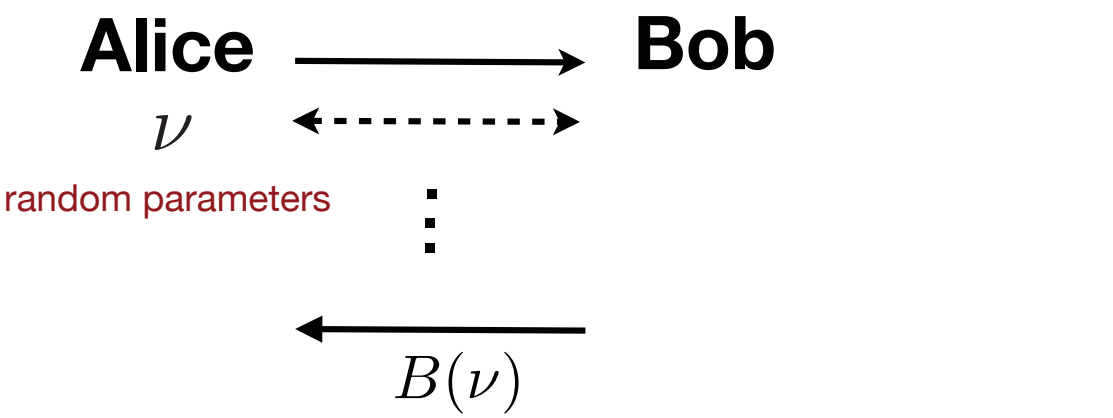
Bob

ε-Verification

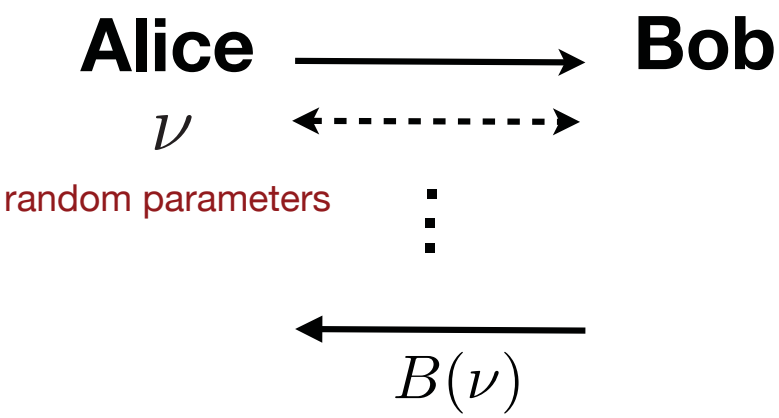
ε-Verification



ε-Verification

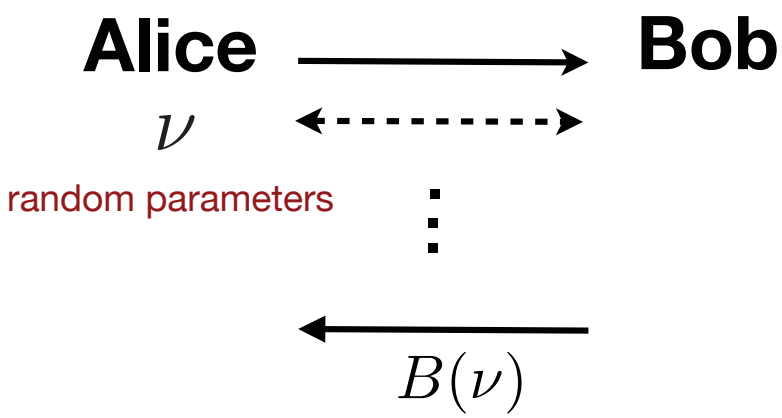


ε-Verification



For any server's strategy the probability of client accepting an incorrect outcome density operator is bounded by ε:

ε-Verification



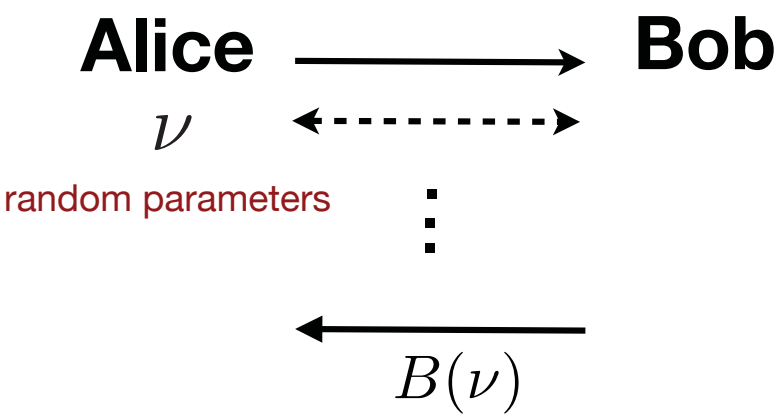
density operator of classical and quantum output

For any server's strategy the probability of client accepting an incorrect outcome density operator is bounded by ε:

$$P^{\nu}_{incorrect} = (\mathbb{I} - |\Psi^{\nu}_{ideal}\rangle\langle\Psi^{\nu}_{ideal}|) \otimes |r^{\nu}_{t}\rangle\langle r^{\nu}_{t}|$$

Accept Key

ε-Verification



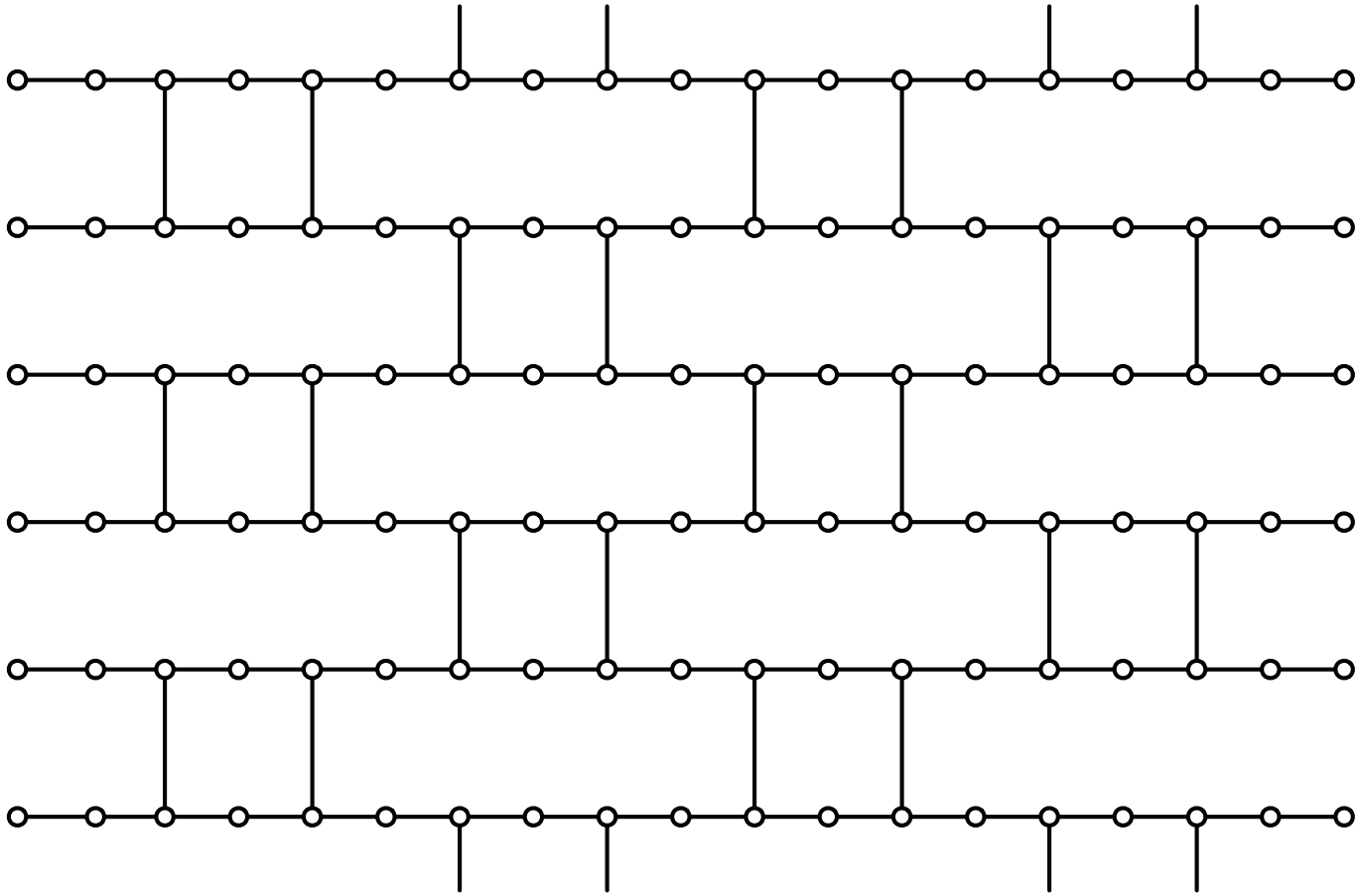
For any server's strategy the probability of client accepting an incorrect outcome density operator is bounded by ε:

$$P^{\nu}_{incorrect} = (\mathbb{I} - |\Psi^{\nu}_{ideal}\rangle \langle \Psi^{\nu}_{ideal}|) \otimes |r^{\nu}_{t}\rangle \langle r^{\nu}_{t}|$$

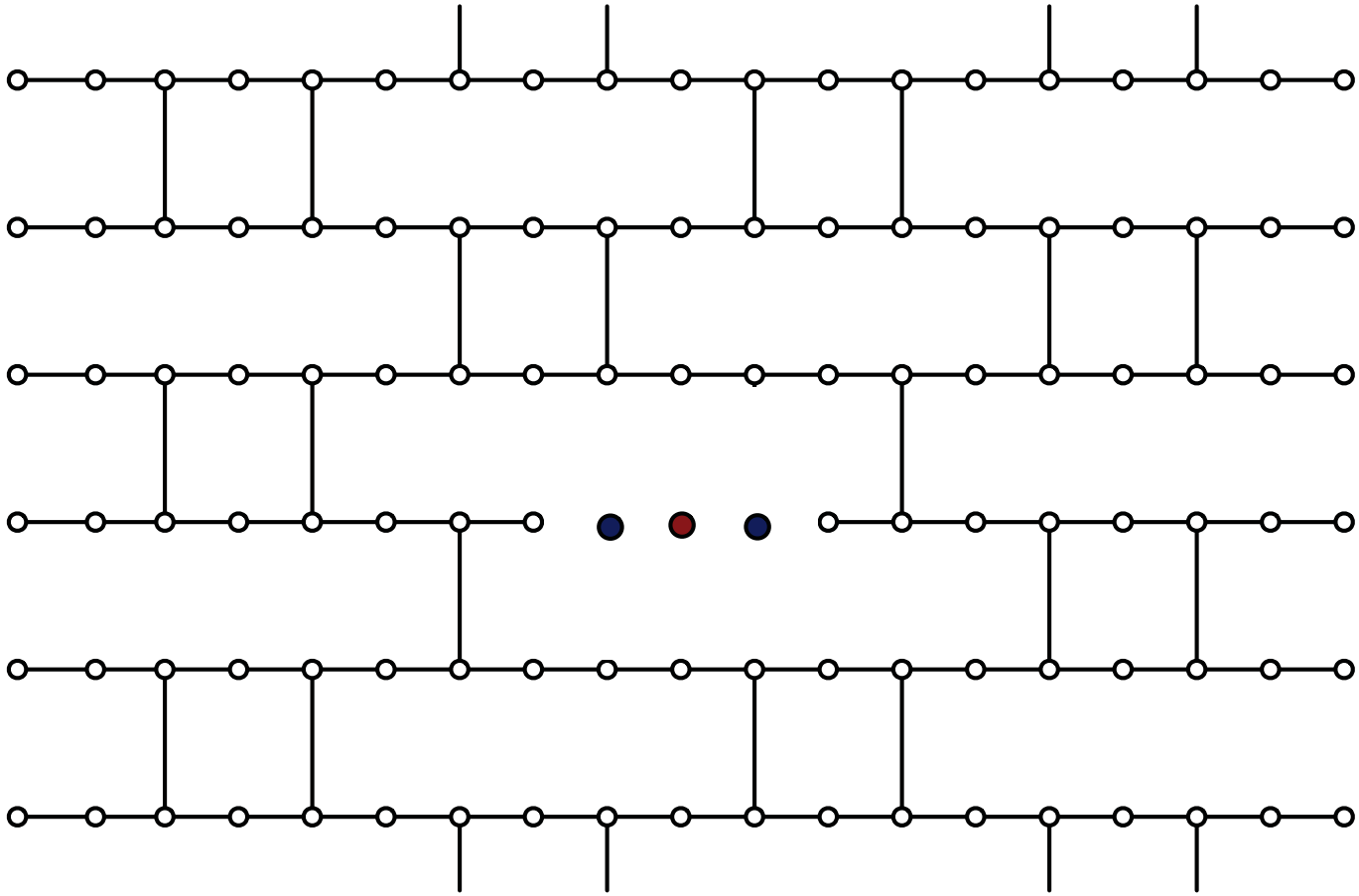
Accept Key

$$\sum_{\nu} \; p(\nu) \; Tr\left(P^{\nu}_{incorrect} \; B(\nu)\right) \leq \epsilon$$

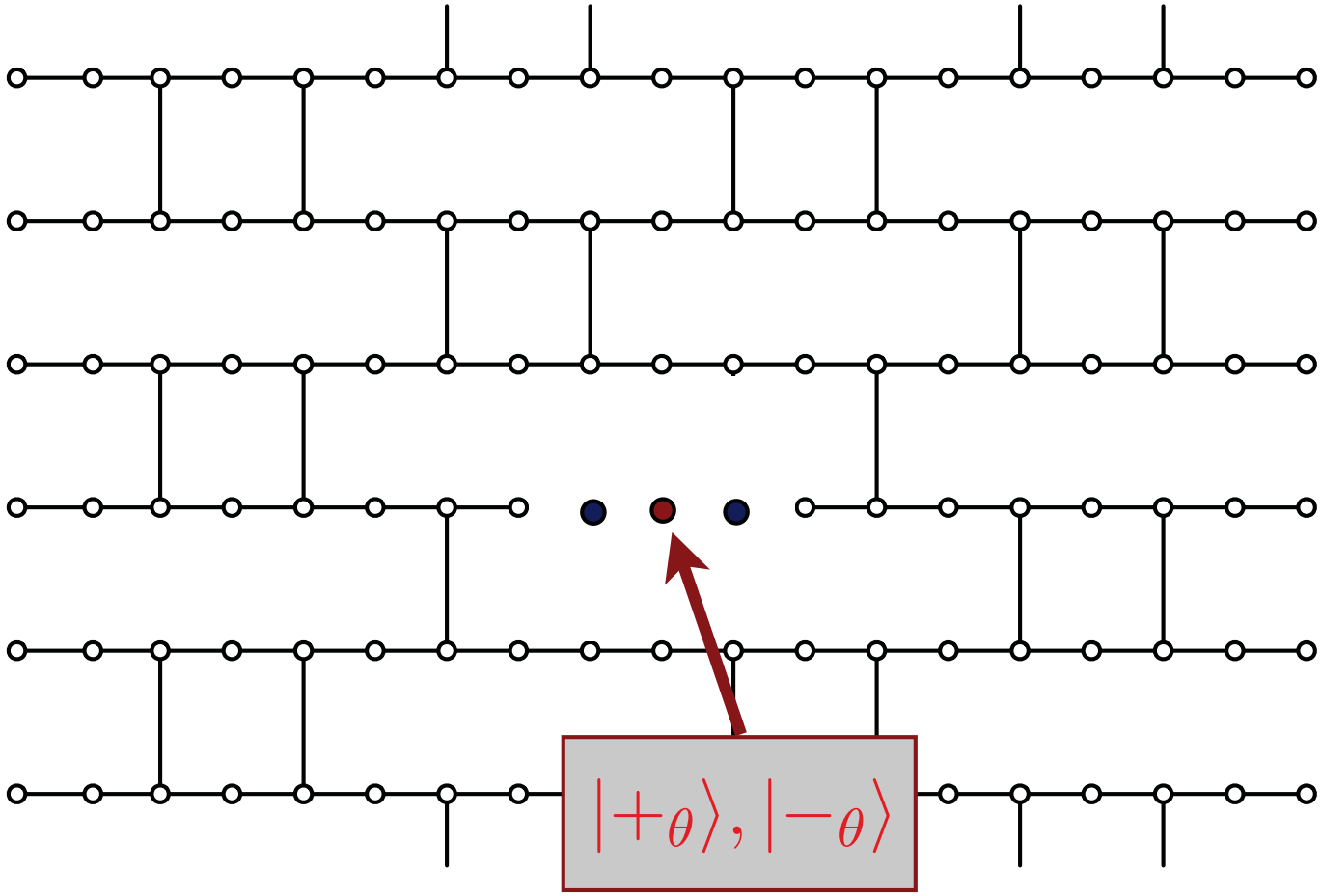
Adding Traps



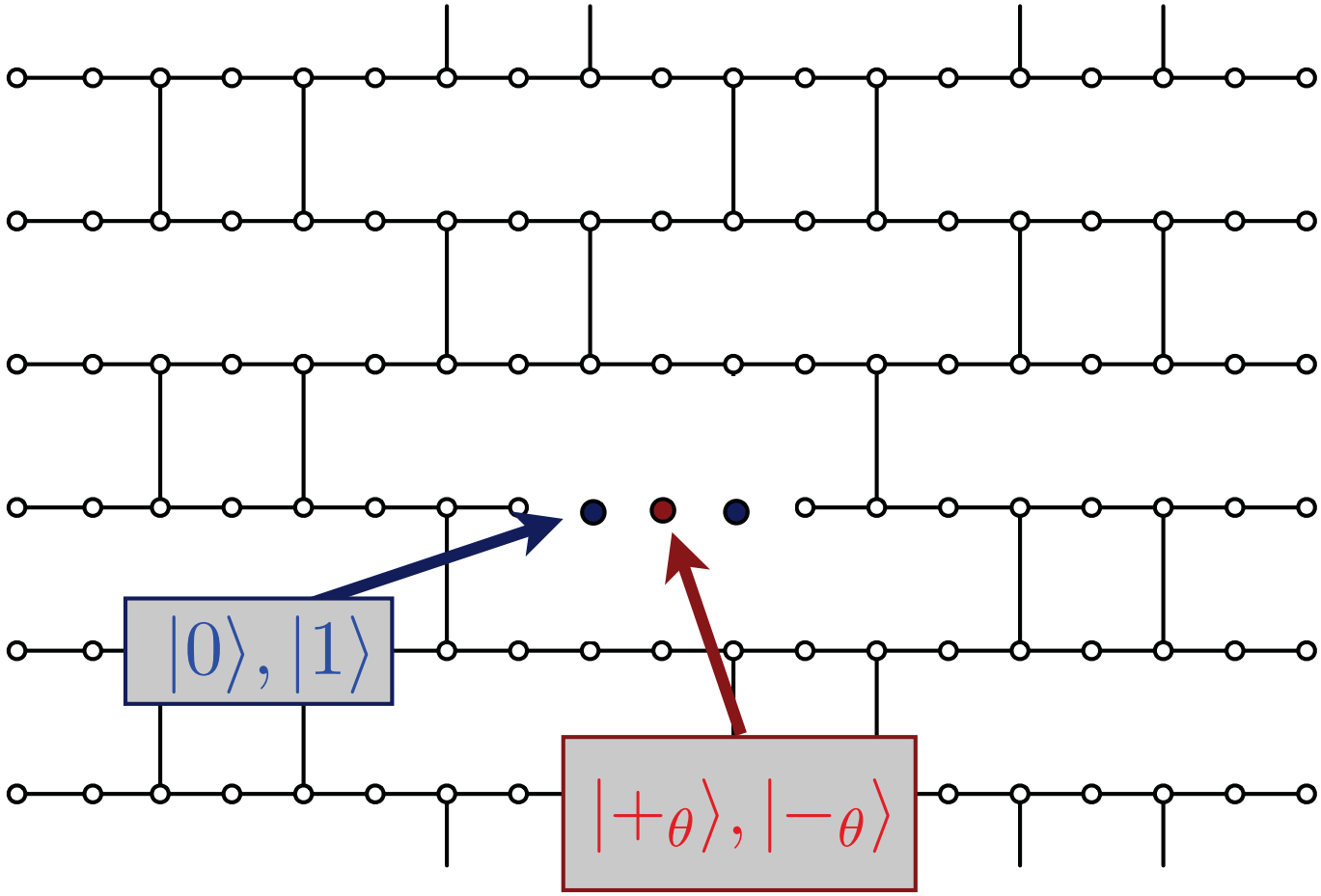
Adding Traps



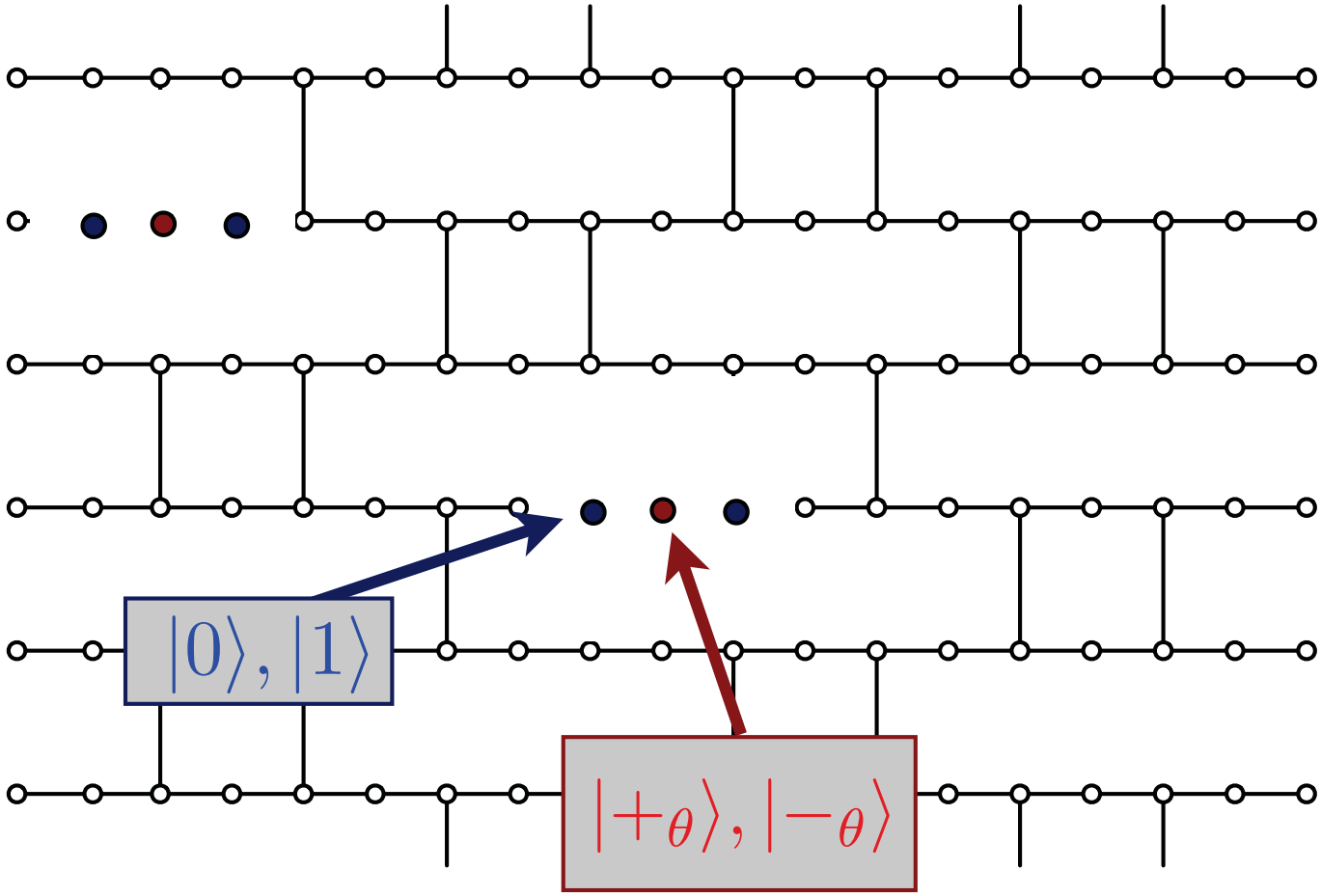
Adding Traps



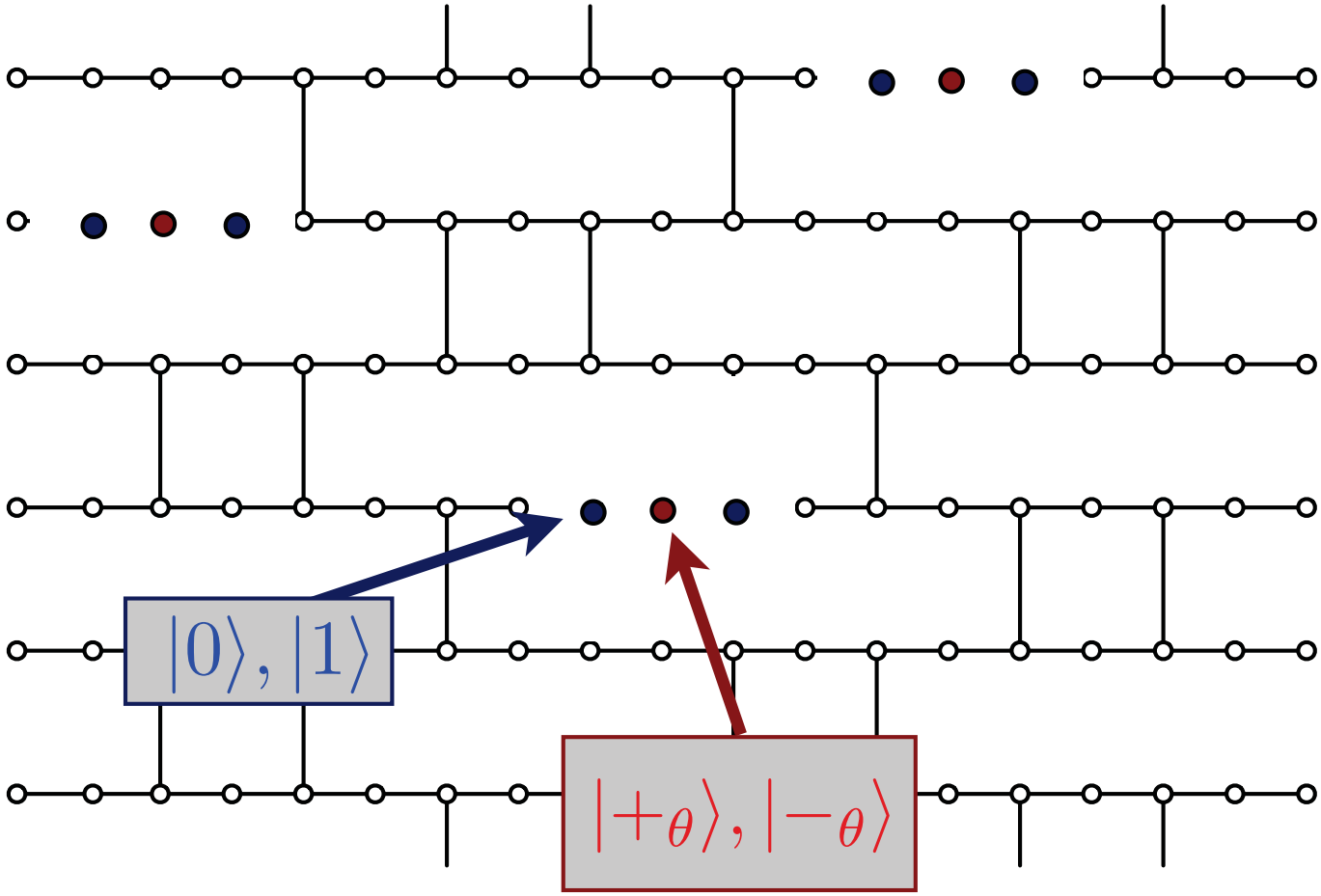
Adding Traps



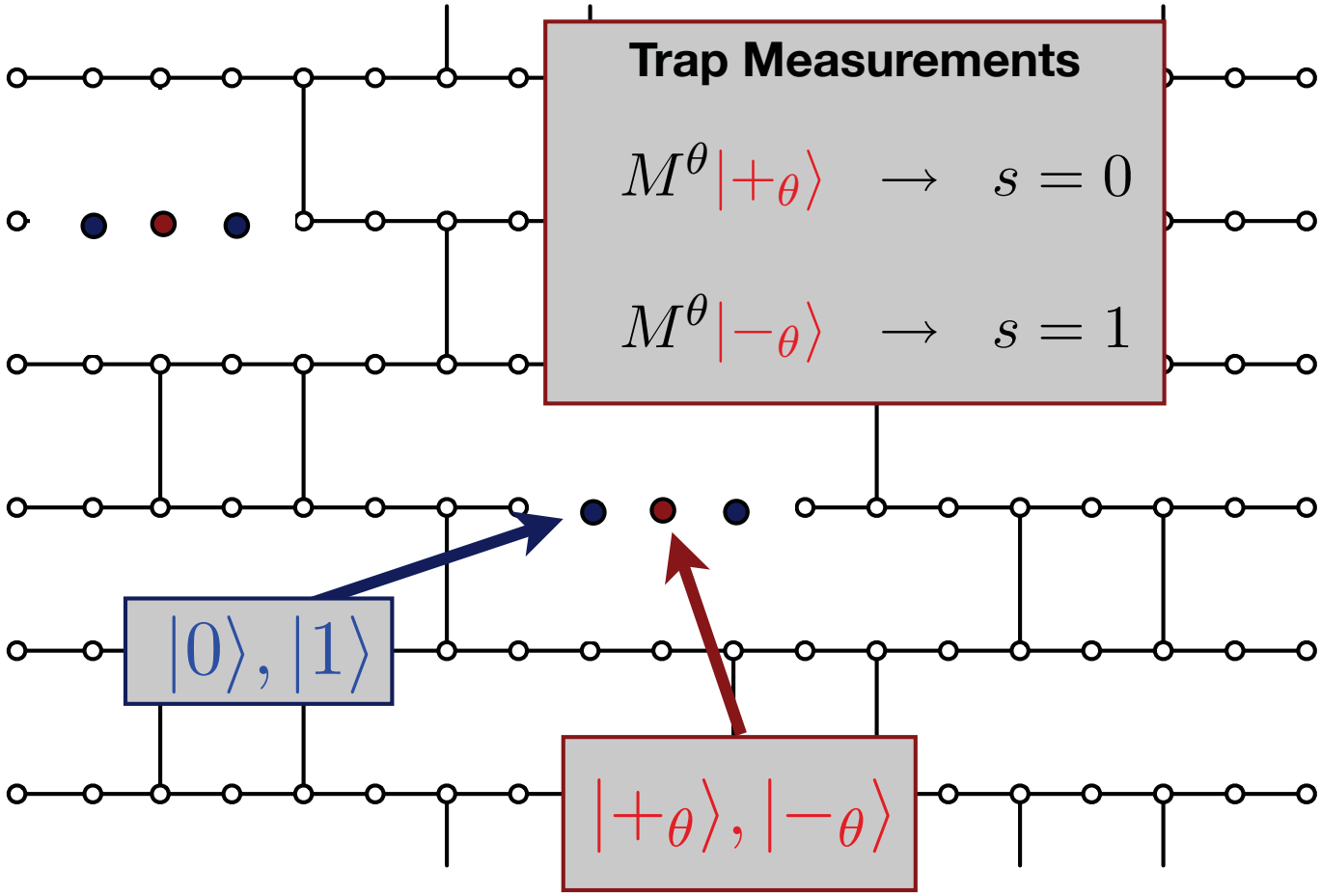
Adding Traps



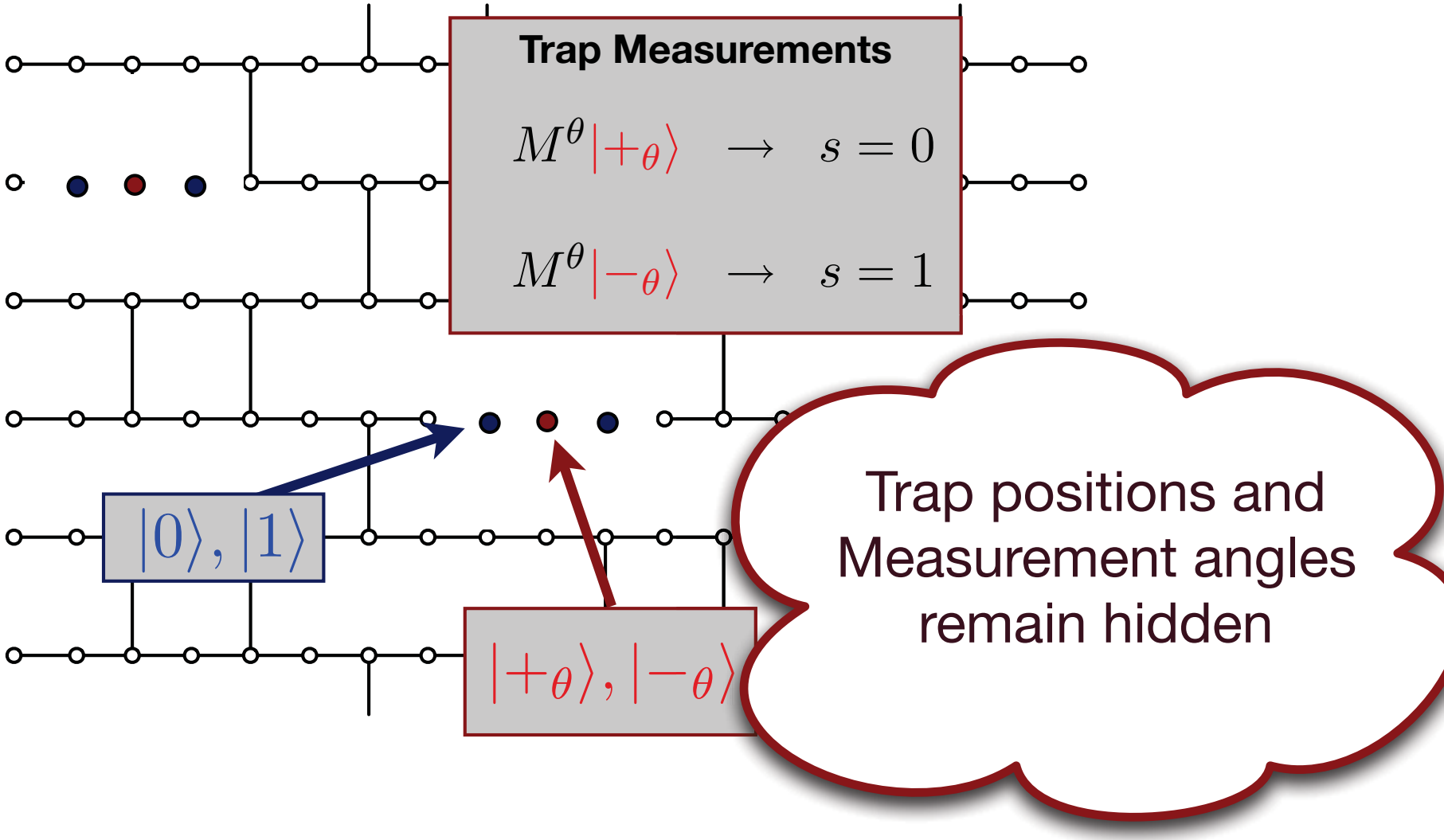
Adding Traps



Adding Traps



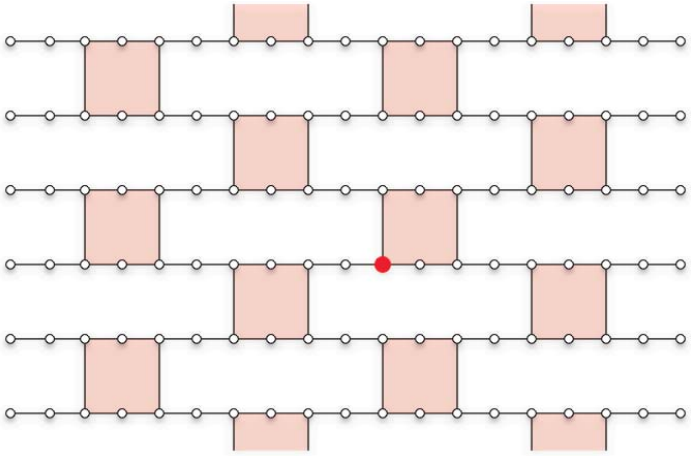
Adding Traps



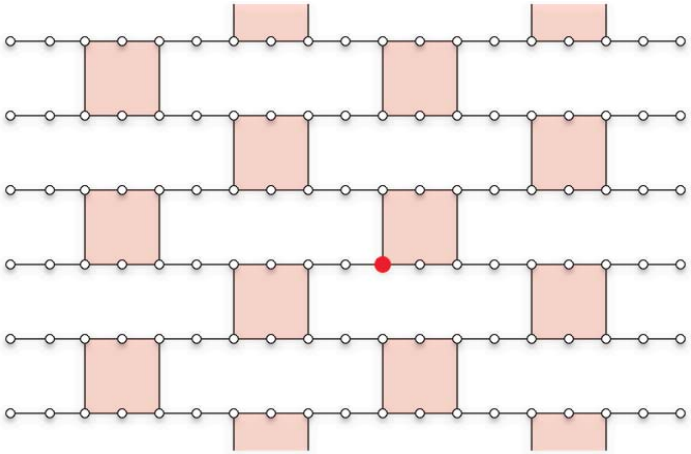
Verification with single trap

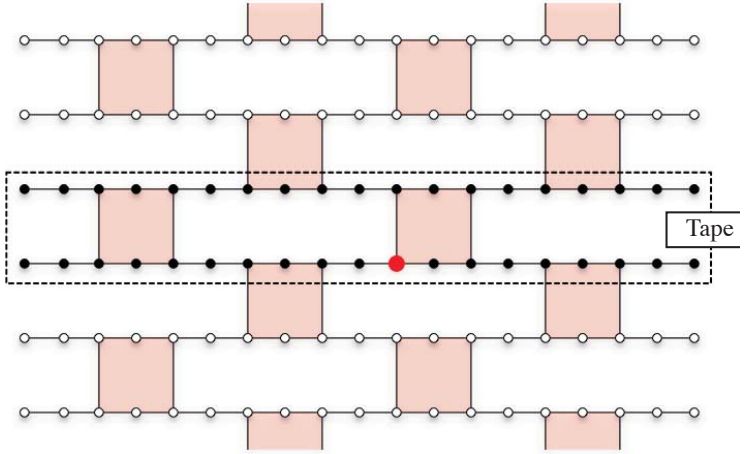
Theorem. Protocol is $(1 - 1/2N)$-verifiable in general, and in the case of purely classical output it is $(1 - 1/N)$-verifiable, where N is the total number of qubits in the protocol.

Verification

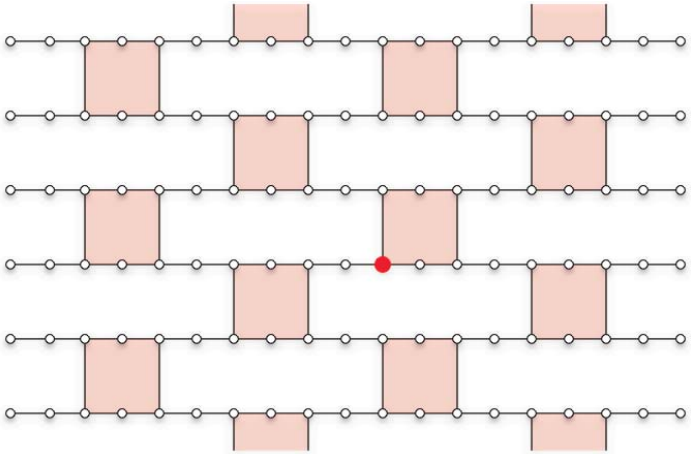


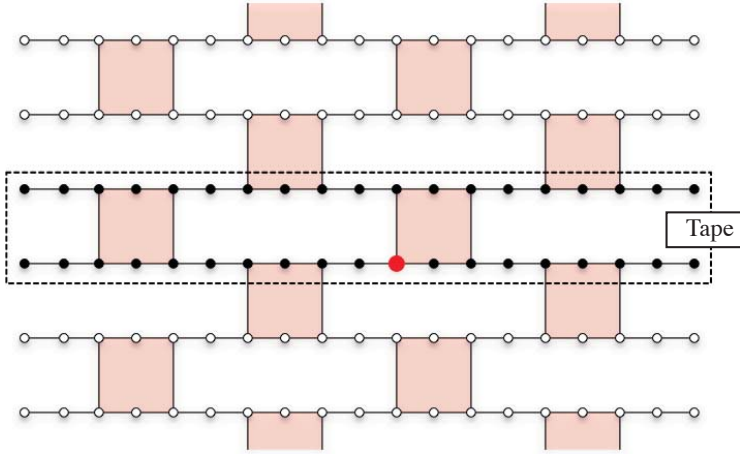
Verification

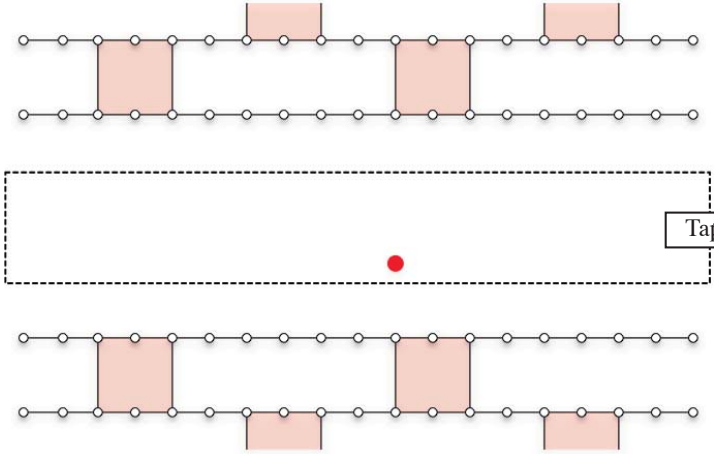




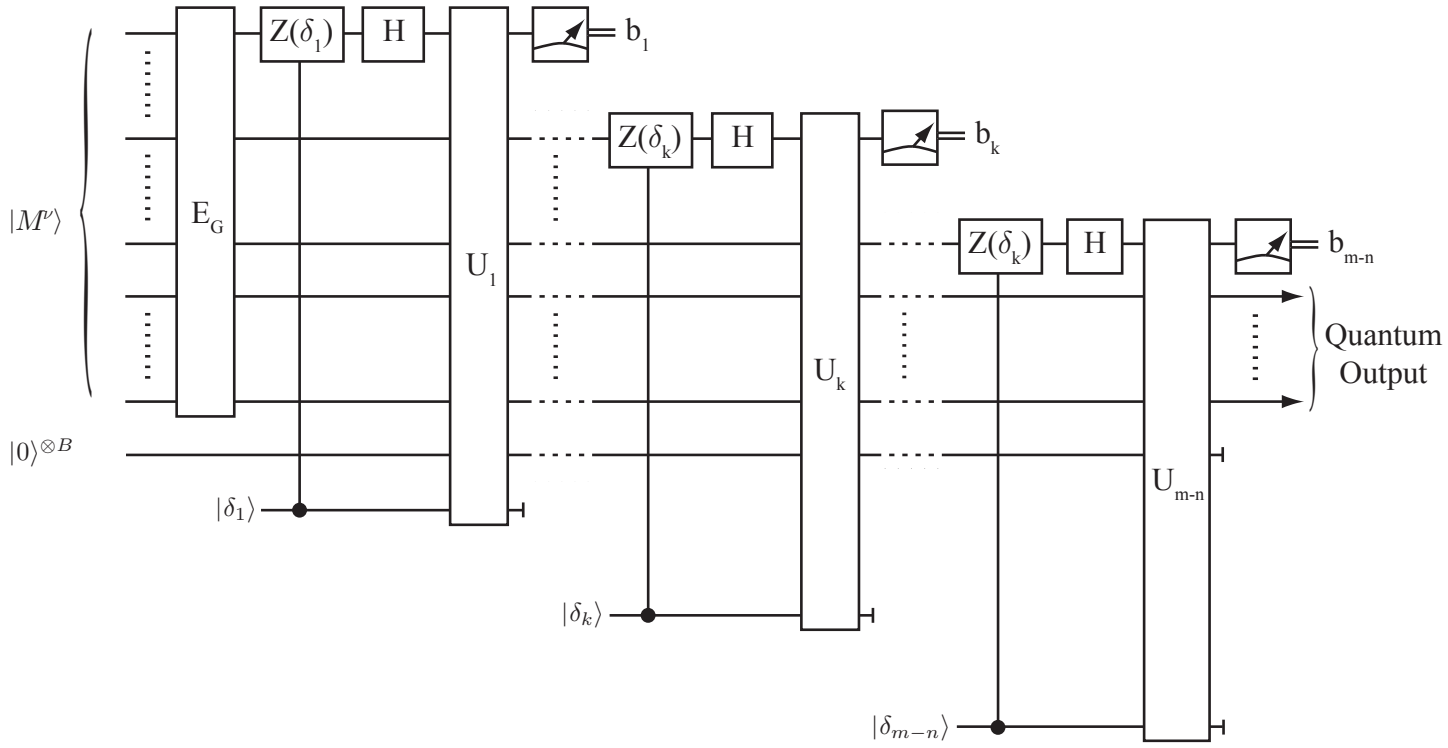
Verification





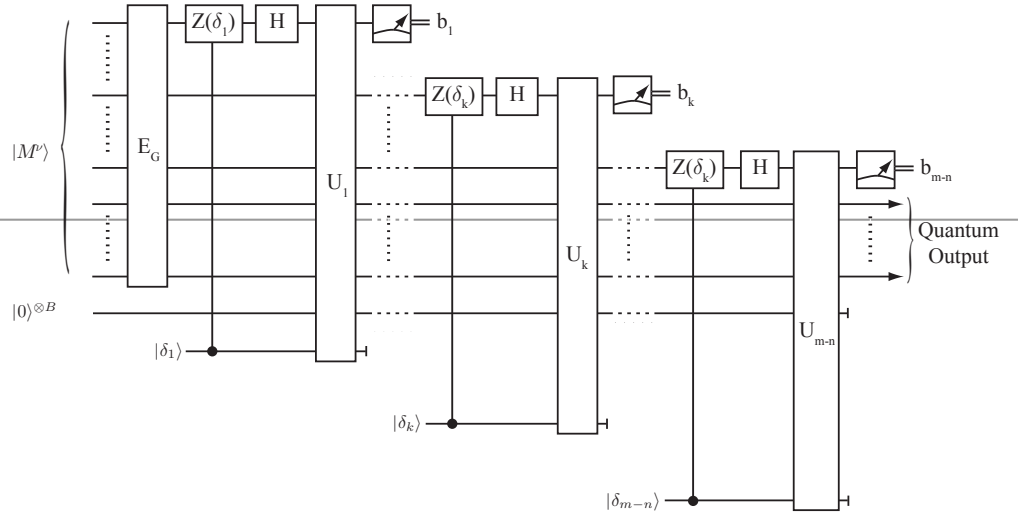


ε-Verification



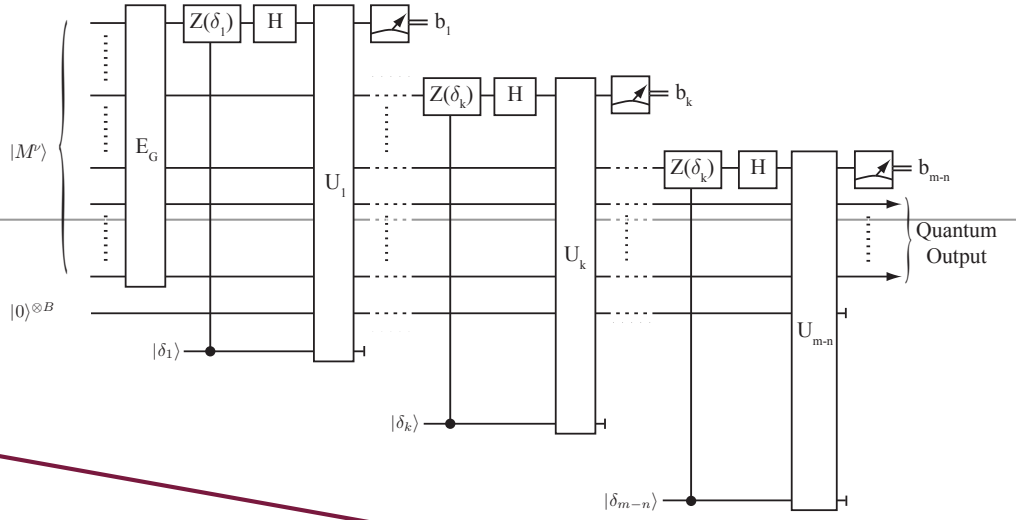
$$B_j(\nu) = \mathrm{Tr}_B \left(\sum_b |b + c_r\rangle \langle b| \, C_{\nu_C,b} \Omega \mathcal{P}((\otimes^B |0\rangle \langle 0|) \otimes \left|\Psi^{\nu,b}\right\rangle \left\langle \Psi^{\nu,b}\right|) \mathcal{P}^\dagger \Omega^\dagger C^\dagger_{\nu_C,b} |b\rangle \langle b + c_r| \right)$$

ε-Verification



$$p_{\text{incorrect}} \leq \sum_{k,\nu_T} \sum_{i \in E_i} \alpha_{ik} \alpha_{ik}^{*} p(\nu_T) \text{Tr}\left(\langle \eta_t^{\nu_T} | \, \sigma_i \left(|\eta_t^{\nu_T}\rangle \langle \eta_t^{\nu_T}| \otimes |\delta_t\rangle \langle \delta_t| \otimes \frac{I}{\text{Tr}(I)} \right) \sigma_i \, |\eta_t^{\nu_T}\rangle \right)$$

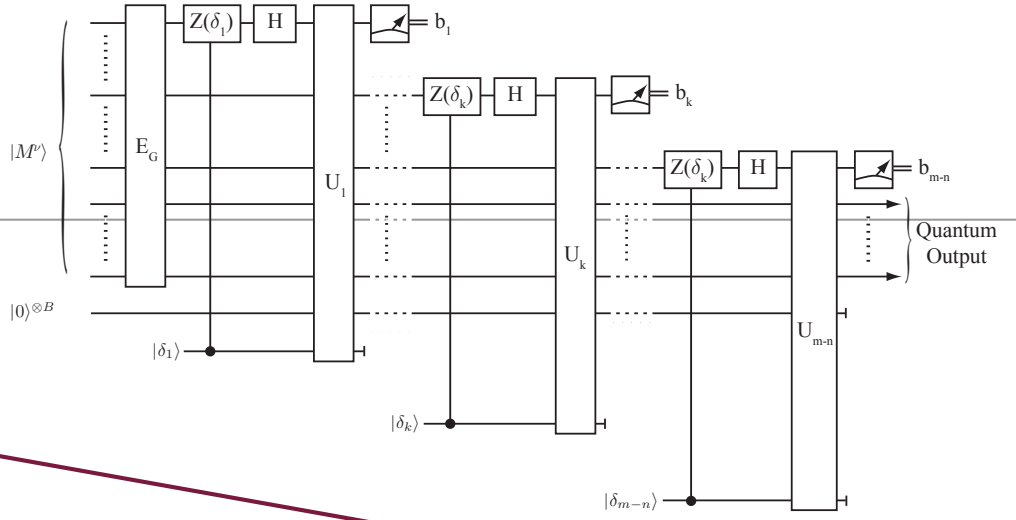
ε-Verification



Blindness

$$p_{\text{incorrect}} \leq \sum_{k,\nu_T} \sum_{i \in E_i} \alpha_{ik} \alpha_{ik}^{*} p(\nu_T) \text{Tr}\left(\langle \eta_t^{\nu_T} | \sigma_i \left(|\eta_t^{\nu_T}\rangle \langle \eta_t^{\nu_T}| \otimes |\delta_t\rangle \langle \delta_t| \otimes \frac{I}{\text{Tr}(I)} \right) \sigma_i |\eta_t^{\nu_T}\rangle \right)$$

ε-Verification



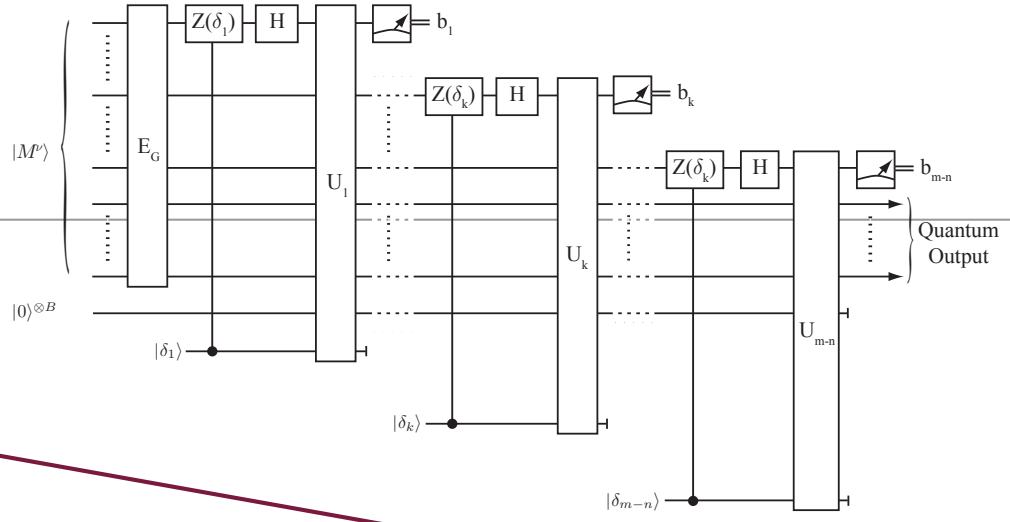
Blindness

$$p_{\text{incorrect}} \leq \sum_{k,\nu_T} \sum_{i \in E_i} \alpha_{ik} \alpha_{ik}^{*} p(\nu_T) \text{Tr}\left(\langle \eta_t^{\nu_T} | \sigma_i \left(|\eta_t^{\nu_T}\rangle \langle \eta_t^{\nu_T}| \otimes |\delta_t\rangle \langle \delta_t| \otimes \frac{I}{\text{Tr}(I)} \right) \sigma_i |\eta_t^{\nu_T}\rangle \right)$$

$$\vdots$$

$$= \frac{1}{16m} \sum_{k} \sum_{i \in E_i} |\alpha_{ki}|^2 \sum_{t, r_t, \theta_t} \left(\langle \eta_t^{\nu_T} | \sigma_{i|t} |\eta_t^{\nu_T}\rangle \right)^2$$

ε-Verification



Blindness

$$p_{\text{incorrect}} \leq \sum_{k,\nu_T} \sum_{i \in E_i} \alpha_{ik} \alpha_{ik}^* p(\nu_T) \text{Tr}\left(\langle \eta_t^{\nu_T} | \sigma_i \left(|\eta_t^{\nu_T}\rangle \langle \eta_t^{\nu_T}| \otimes |\delta_t\rangle \langle \delta_t| \otimes \frac{I}{\text{Tr}(I)} \right) \sigma_i |\eta_t^{\nu_T}\rangle \right)$$

$$\vdots$$

$$= \frac{1}{16m} \sum_k \sum_{i \in E_i} |\alpha_{ki}|^2 \sum_{t, r_t, \theta_t} \left(\langle \eta_t^{\nu_T} | \sigma_{i|t} |\eta_t^{\nu_T}\rangle \right)^2$$

$$\vdots$$

$$\leq 1 - \frac{1}{2m}$$

Probability Amplification

To increase the probability of any local error being detected

O(N) many traps in random locations

Probability Amplification

To increase the probability of any local error being detected

O(N) many traps in random locations

To increase the minimum weight of any operator which leads to an incorrect outcome

Fault-Tolerance

Probability Amplification

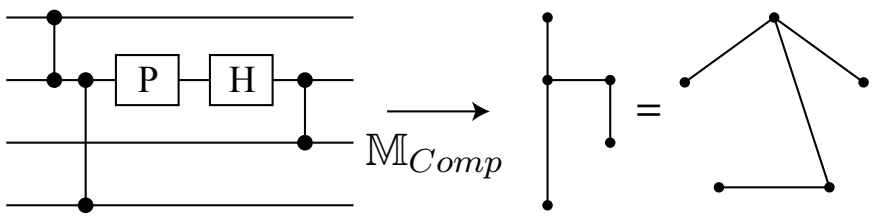
Probability Amplification

Challenge: Traps break the graph

Probability Amplification

Challenge: Traps break the graph

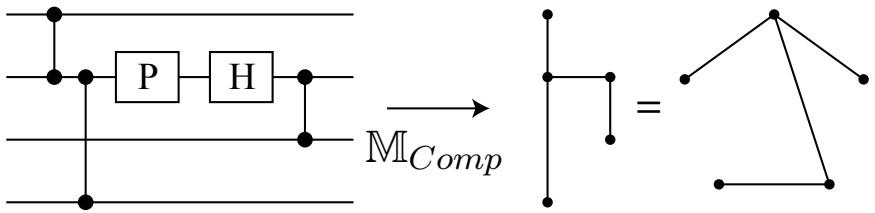
1.



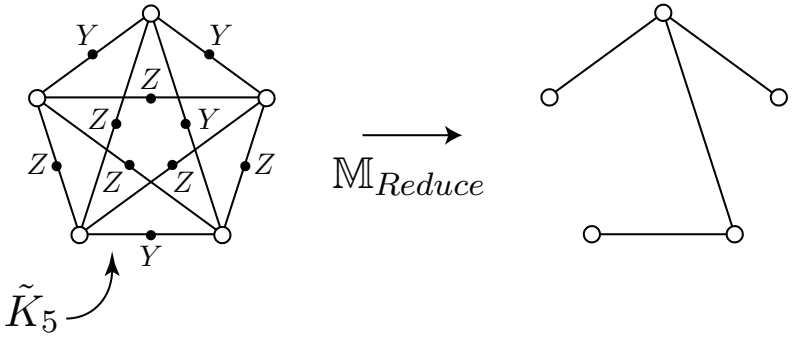
Probability Amplification

Challenge: Traps break the graph

1.



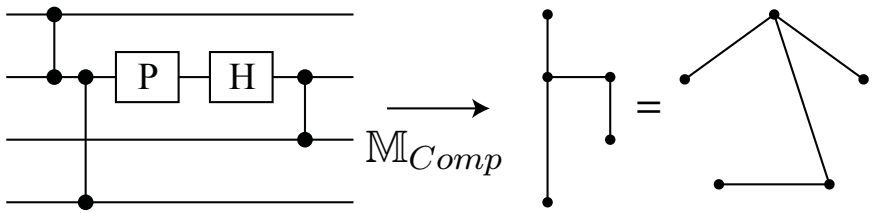
2.



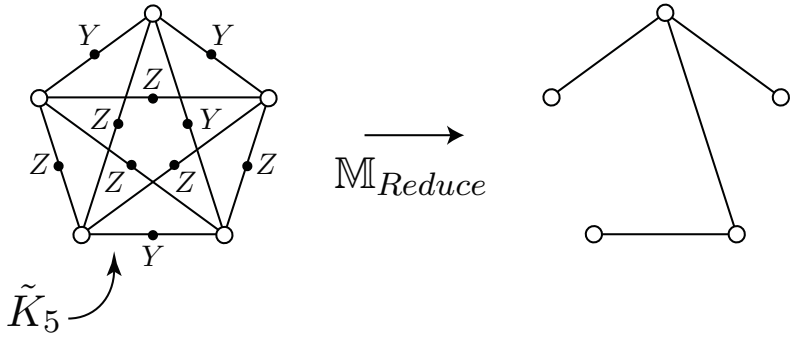
Probability Amplification

Challenge: Traps break the graph

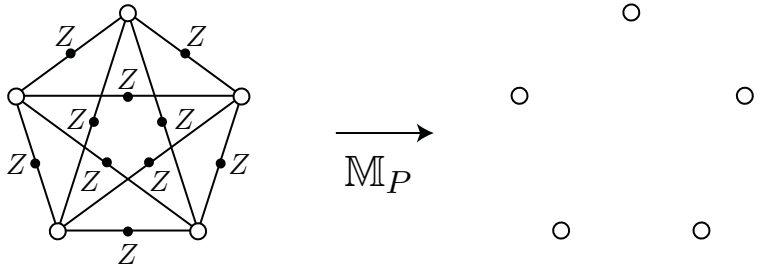
1.



2.



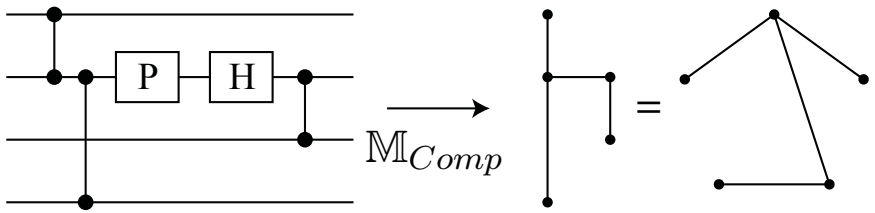
3.



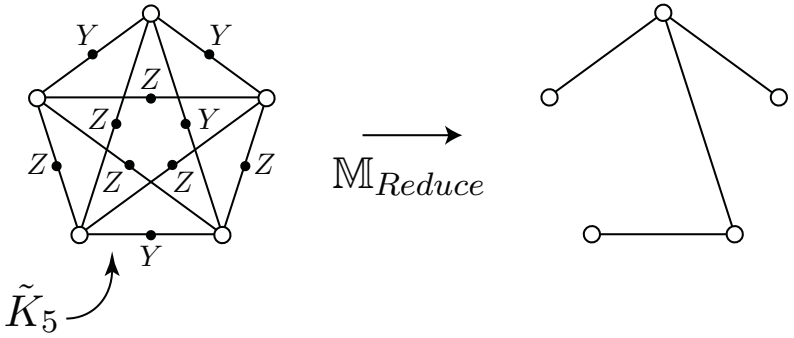
Probability Amplification

Challenge: Traps break the graph

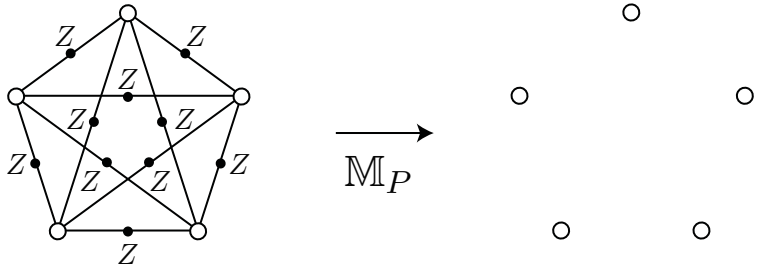
1.



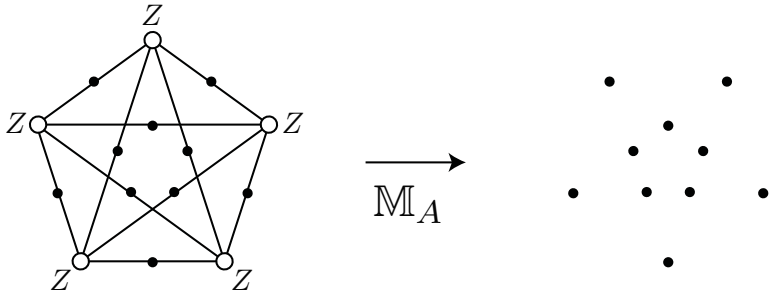
2.



3.



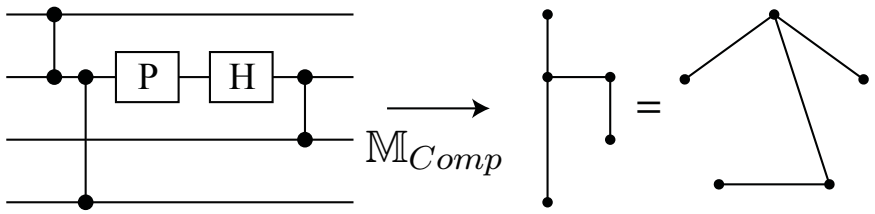
4.



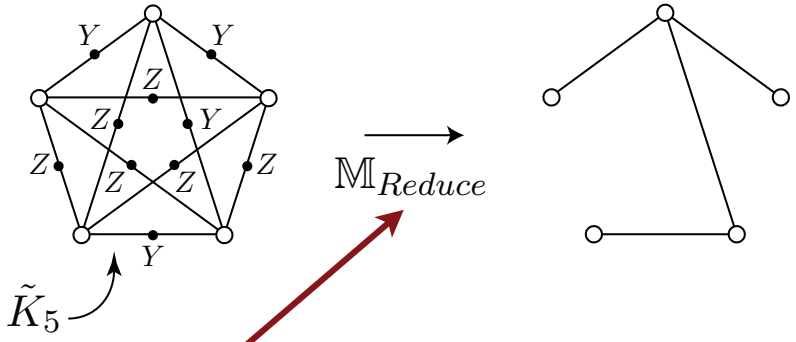
Probability Amplification

Challenge: Traps break the graph

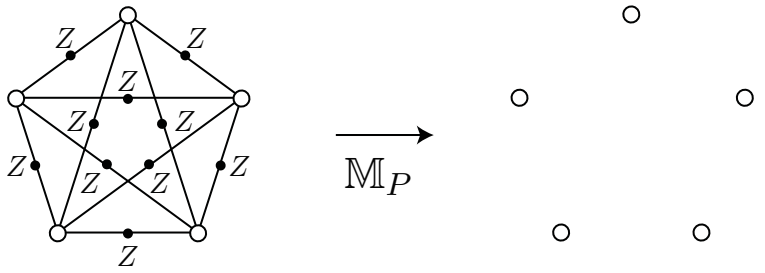
1.



2.

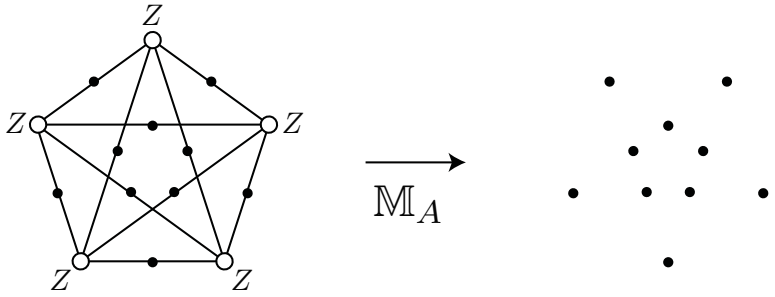


3.



Required 3D lattice for Raussendorf, Harrington and Goyal Topological error-correcting code with defect thickness *d*

4.

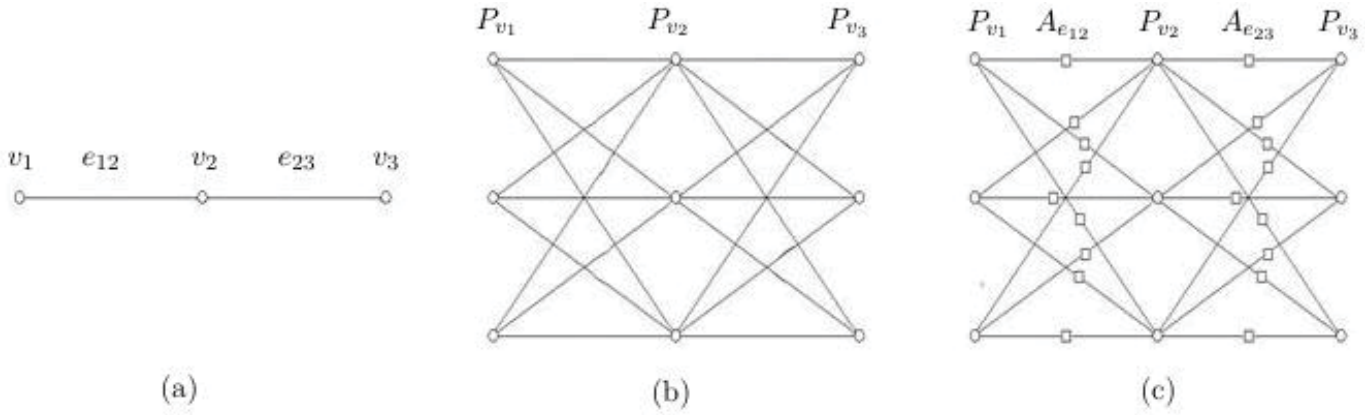


Efficient Verifiable UBQC

[Kashefi-Wallden 16]

Dotted triple-graph construction ⟶ O(N) overheads for VUBQC

generic resource state, that one can insert multiple independent traps without revealing anything or disrupting the computation

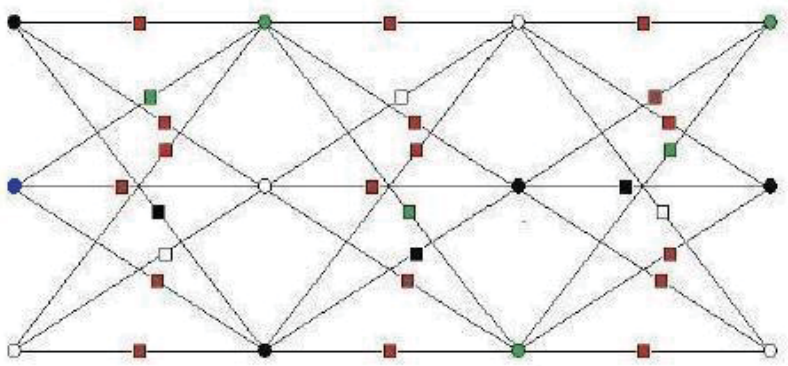


Efficient Verifiable UBQC

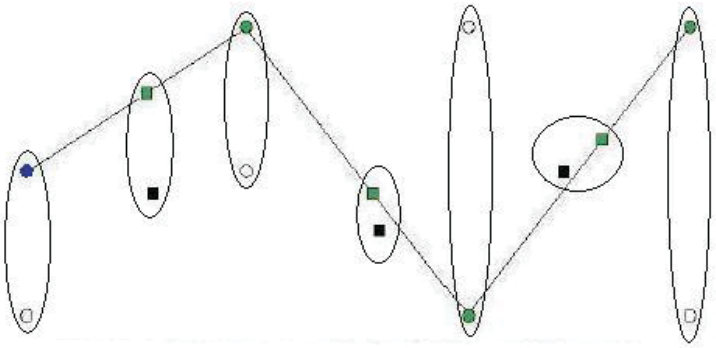
[Kashefi-Wallden 16]

Dotted triple-graph construction ⟶ O(N) overheads for VUBQC

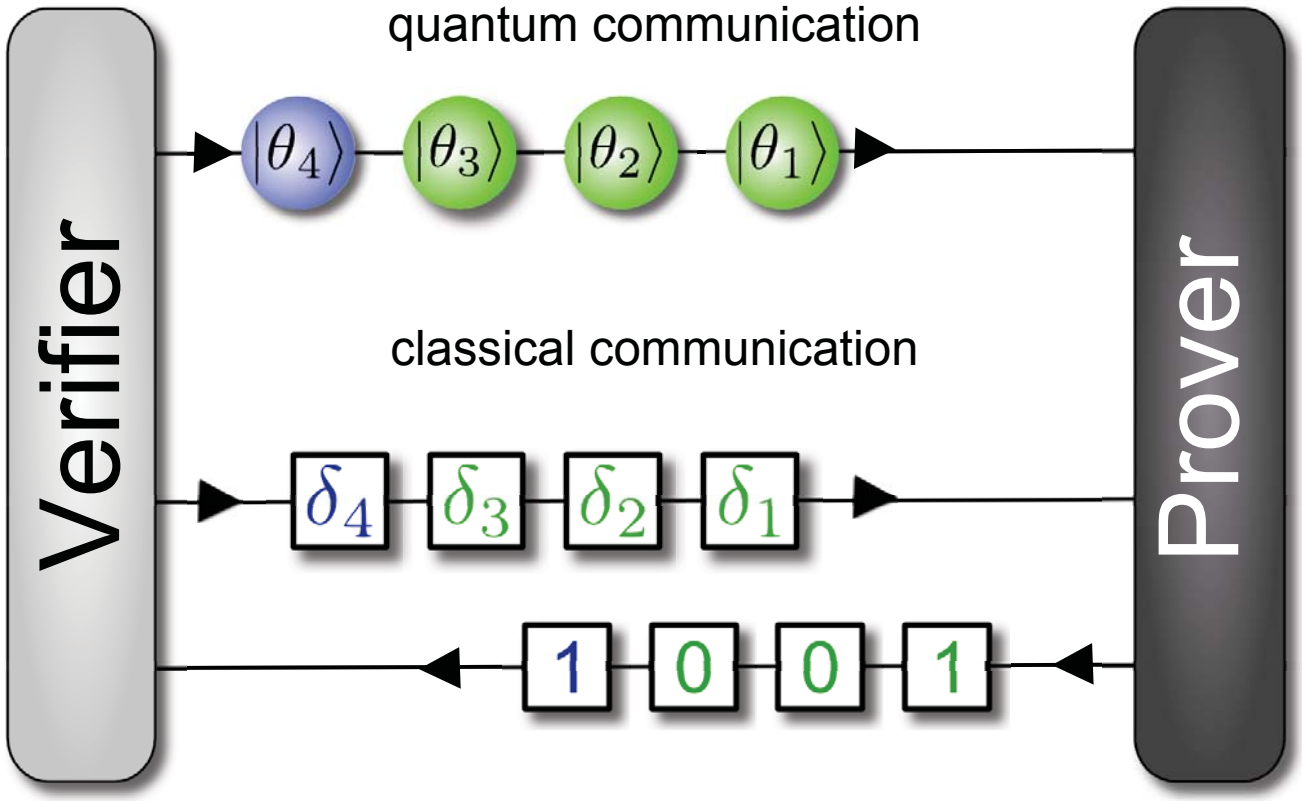
generic resource state, that one can insert multiple independent traps without revealing anything or disrupting the computation



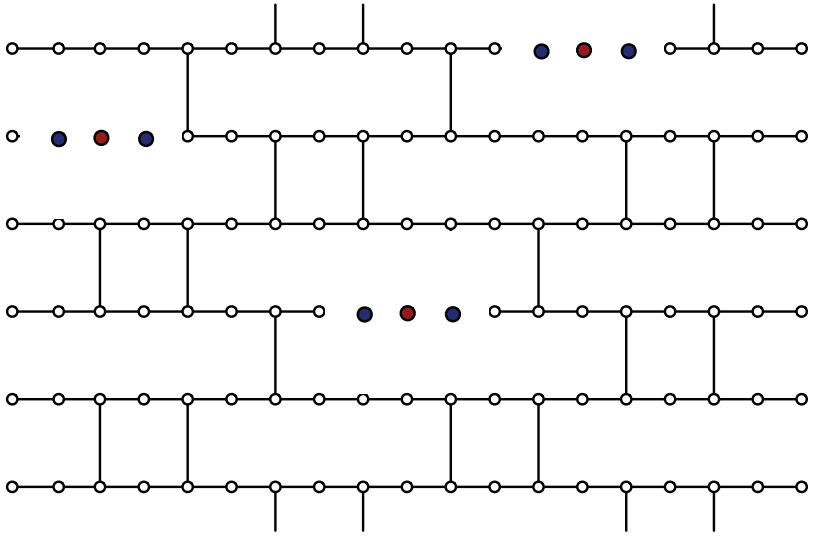
(a) Trap-colouring



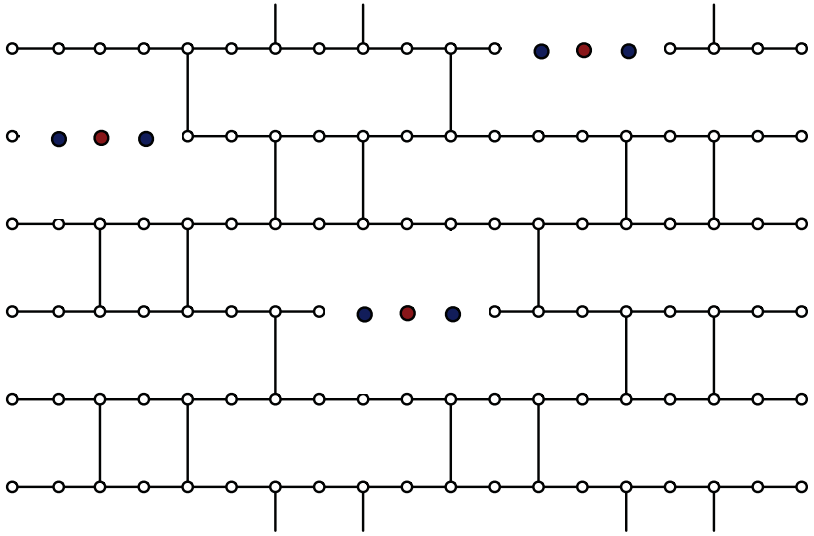
What can we do with 4-qubits

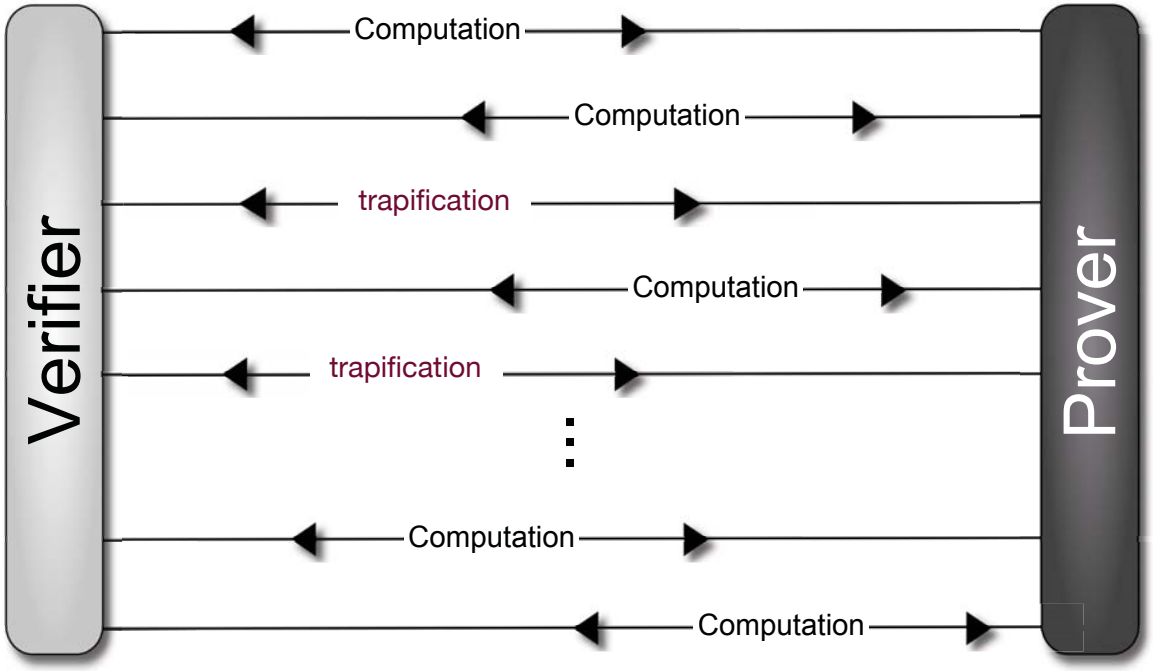


Restricting to Classical Input and Output



Restricting to Classical Input and Output





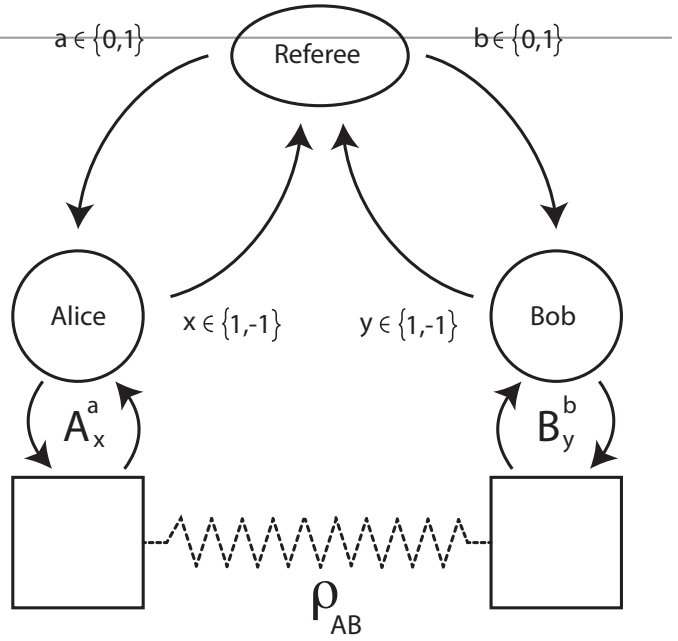
A Complete new proof of verification was required

Pauli (σ_i)	Trap Stabilizer Measurement			Overall
	$X \otimes \mathbb{I} \otimes Y \otimes Y$	$Y \otimes X \otimes X \otimes Y$	$Y \otimes Y \otimes \mathbb{I} \otimes X$	
$C \otimes C \otimes C \otimes C$	✓	✓	✓	✓
$C \otimes C \otimes C \otimes A$	✗	✗	✗	✗
$C \otimes C \otimes A \otimes C$	✗	✗	✓	✗
$C \otimes C \otimes A \otimes A$	✓	✓	✗	✗
$C \otimes A \otimes C \otimes C$	✓	✗	✗	✗
$C \otimes A \otimes C \otimes A$	✗	✓	✓	✗
$C \otimes A \otimes A \otimes C$	✗	✓	✗	✗
$C \otimes A \otimes A \otimes A$	✓	✗	✓	✗
$A \otimes C \otimes C \otimes C$	✗	✗	✗	✗
$A \otimes C \otimes C \otimes A$	✓	✓	✓	✓
$A \otimes C \otimes A \otimes C$	✓	✓	✗	✗
$A \otimes C \otimes A \otimes A$	✗	✗	✓	✗
$A \otimes A \otimes C \otimes C$	✗	✓	✓	✗
$A \otimes A \otimes C \otimes A$	✓	✗	✗	✗
$A \otimes A \otimes A \otimes C$	✓	✗	✓	✗
$A \otimes A \otimes A \otimes A$	✗	✓	✗	✗

A Complete new proof of verification was required

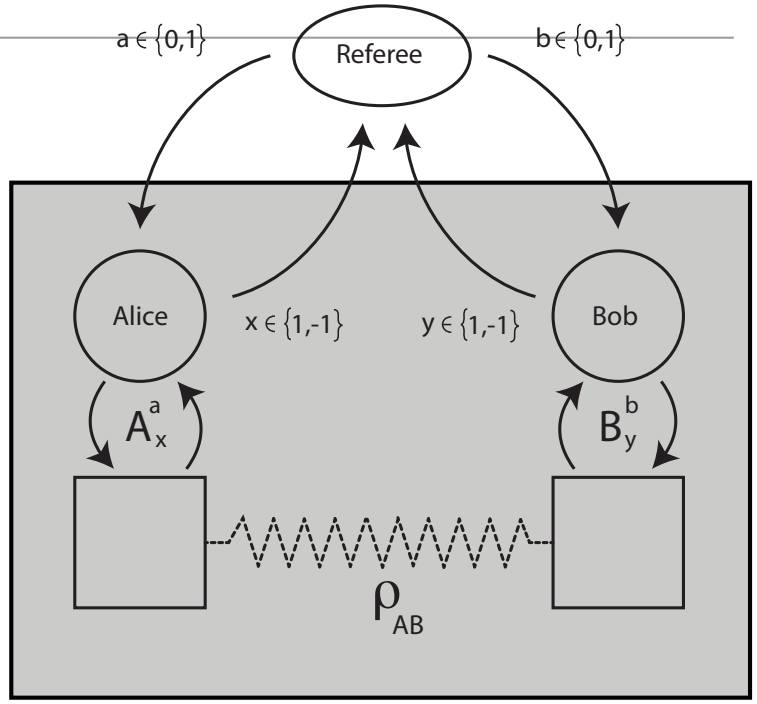
Pauli (σ_i)	Trap Stabilizer Measurement			Overall
	$X \otimes \mathbb{I} \otimes Y \otimes Y$	$Y \otimes X \otimes X \otimes Y$	$Y \otimes Y \otimes \mathbb{I} \otimes X$	
$C \otimes C \otimes C \otimes C$	✓	✓	✓	✓
$C \otimes C \otimes C \otimes A$	✗	✗	✗	✗
$C \otimes C \otimes A \otimes C$	✗	✗	✓	✗
$C \otimes C \otimes A \otimes A$	✓	✓	✗	✗
$C \otimes A \otimes C \otimes C$	✓	✗	✗	✗
$C \otimes A \otimes C \otimes A$	✗	✓	✓	✗
$C \otimes A \otimes A \otimes C$	✗	✓	✗	✗
$C \otimes A \otimes A \otimes A$	✓	✗	✓	✗
$A \otimes C \otimes C \otimes C$	✗	✗	✗	✗
$A \otimes C \otimes C \otimes A$	✓	✓	✓	✓
$A \otimes C \otimes A \otimes C$	✓	✓	✗	✗
$A \otimes C \otimes A \otimes A$	✗	✗	✓	✗
$A \otimes A \otimes C \otimes C$	✗	✓	✓	✗
$A \otimes A \otimes C \otimes A$	✓	✗	✗	✗
$A \otimes A \otimes A \otimes C$	✓	✗	✓	✗
$A \otimes A \otimes A \otimes A$	✗	✓	✗	✗

Blind Verification of Entanglement



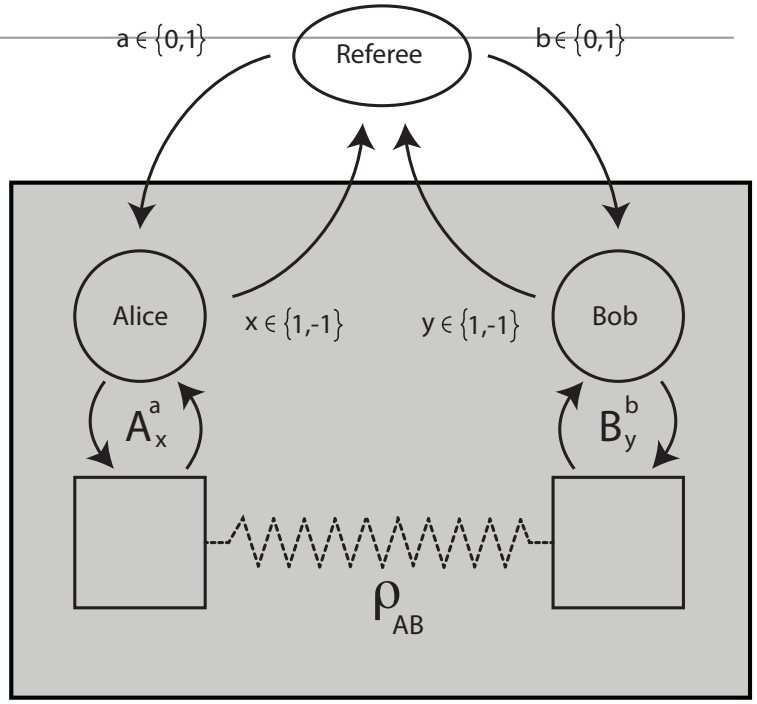
Blind Verification of Entanglement

Perform by an untrusted server



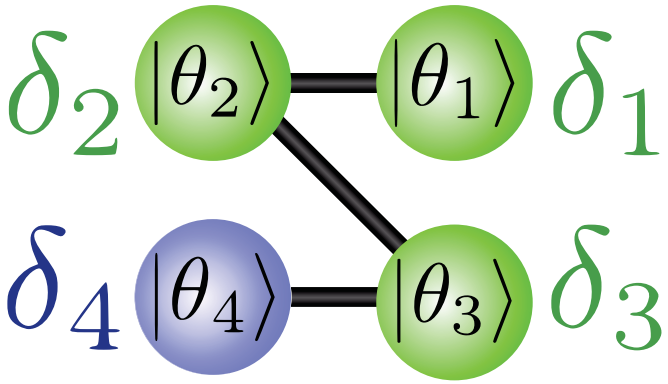
Blind Verification of Entanglement

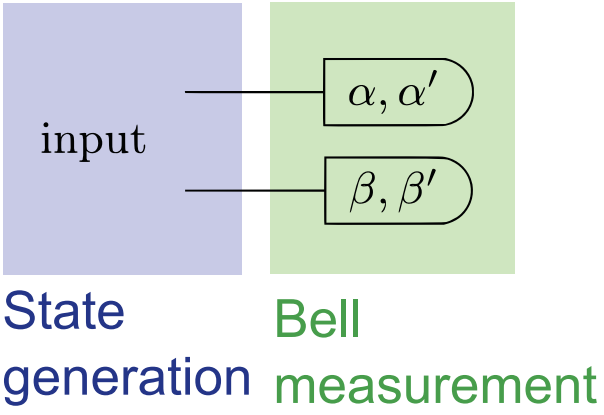
Perform by an untrusted server

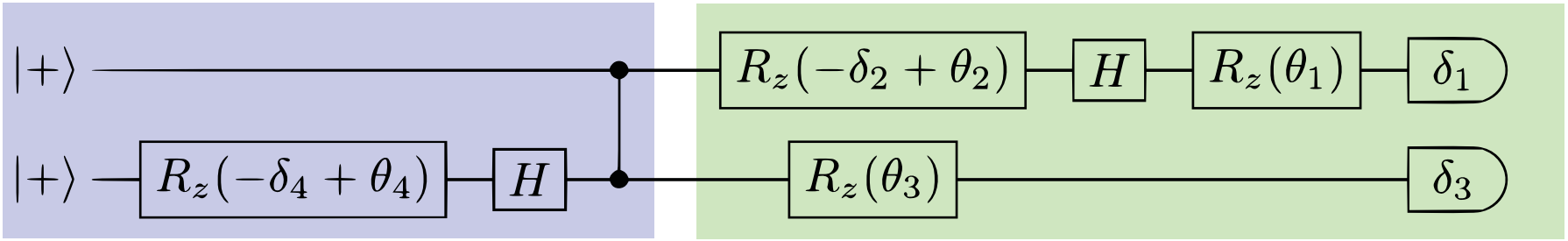


If server knows he is running Bell test,
he can create fake outcomes to violate the inequality,
the trapificaiton procedure in between prevents this to happen

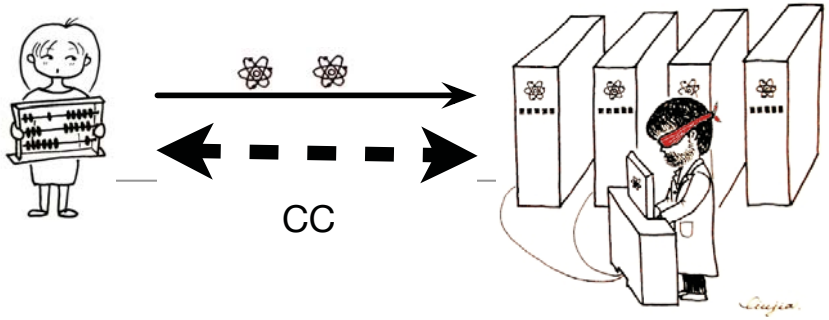
Blind Verification of Entanglement



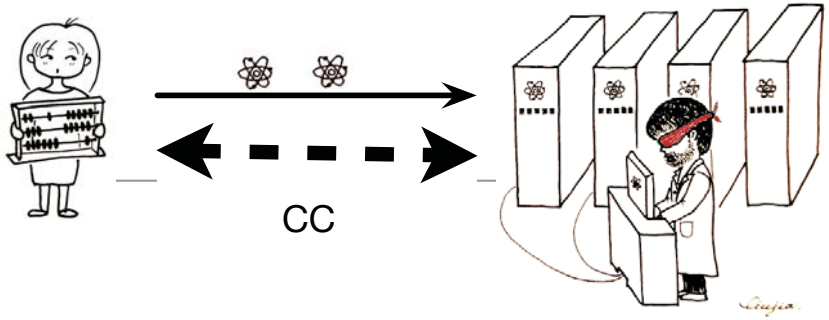




Quantum Property Testing



Quantum Property Testing



Non-classical Correlation

Contextuality

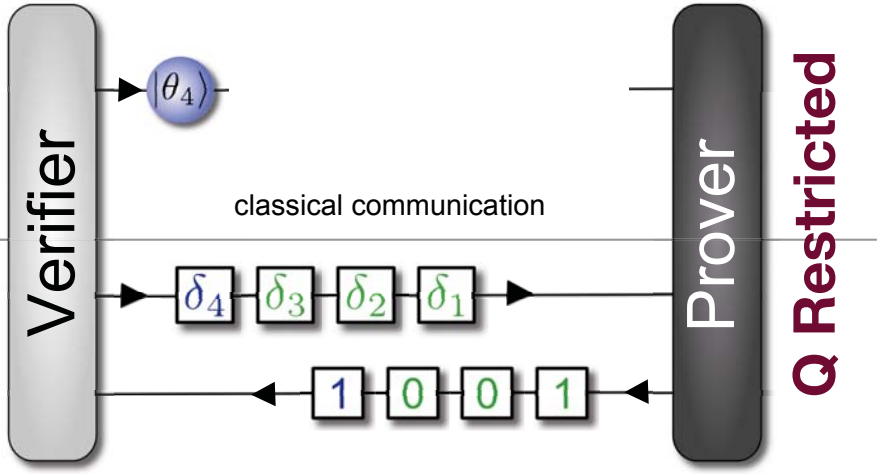
Dimentionality

Superposition

VUBQC extension

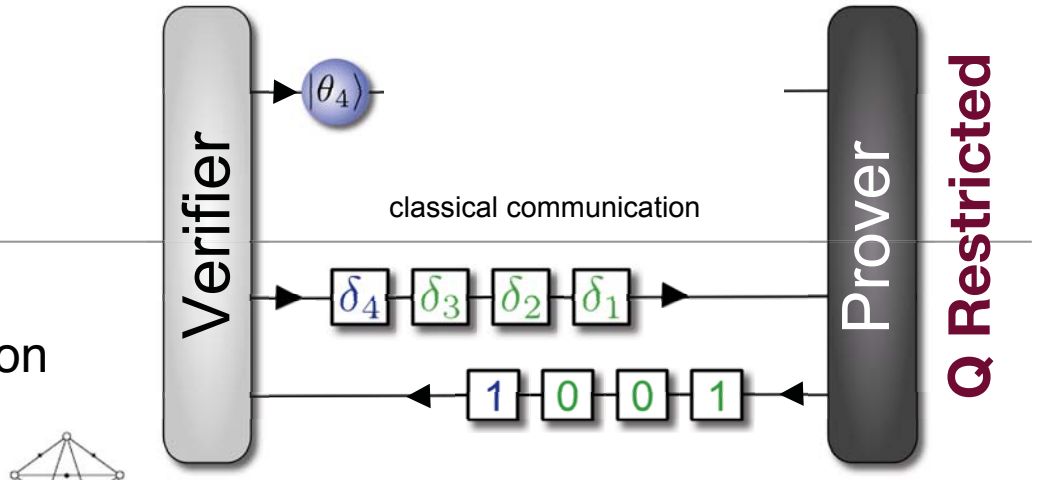
VUBQC extension

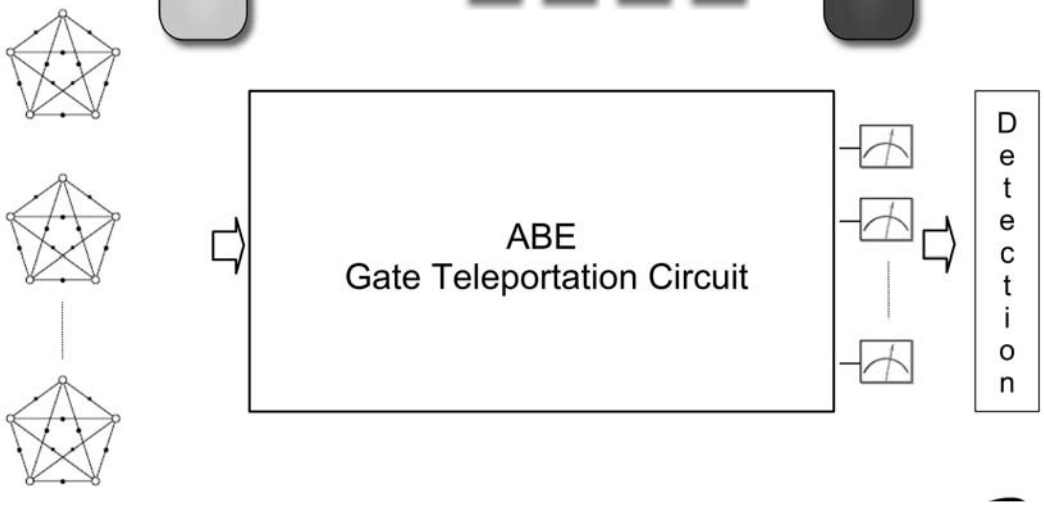
Verification of one-pure-qubit computation
Kapourniotis, Kashefi, Datta, TQC14



VUBQC extension

Verification of one-pure-qubit computation
Kapourniotis, Kashefi, Datta, TQC14

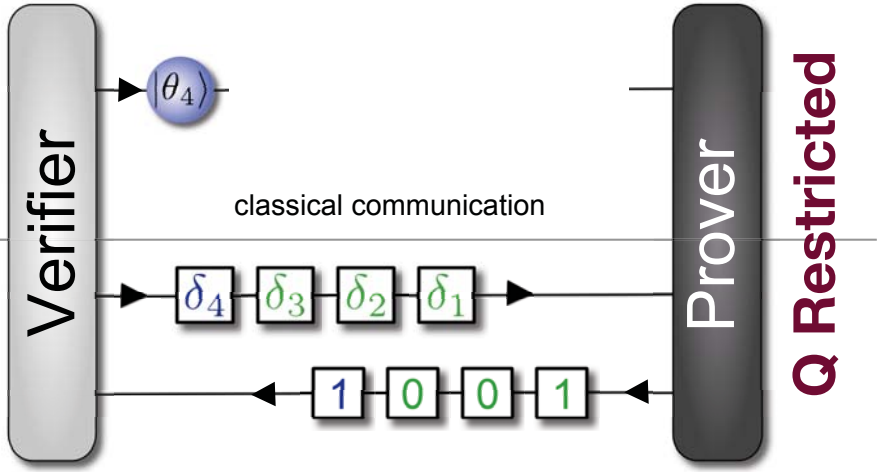


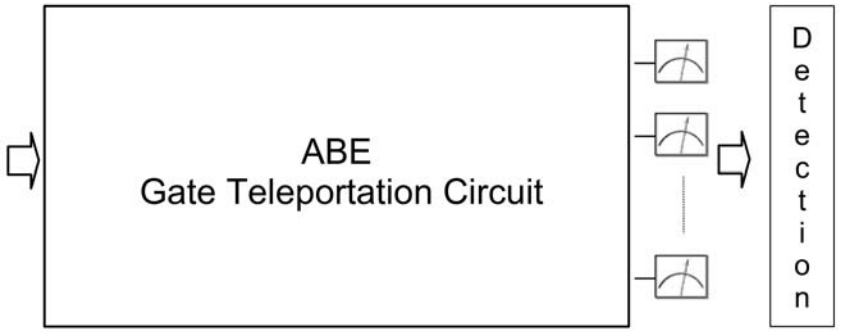


Verification with minimal communicatio
Kapourniotis, Dunjko, Kashefi ASQIC15

VUBQC extension

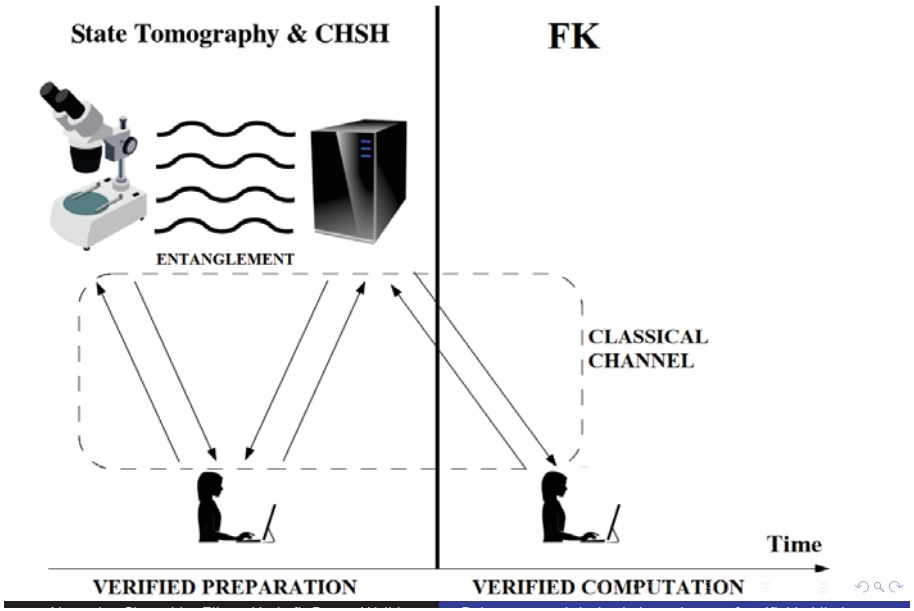
Verification of one-pure-qubit computation
Kapourniotis, Kashefi, Datta, TQC14



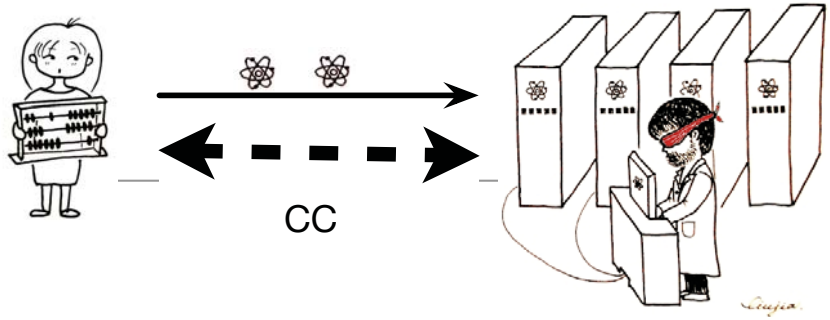


Verification with minimal communicatio
Kapourniotis, Dunjko, Kashefi ASQIC15

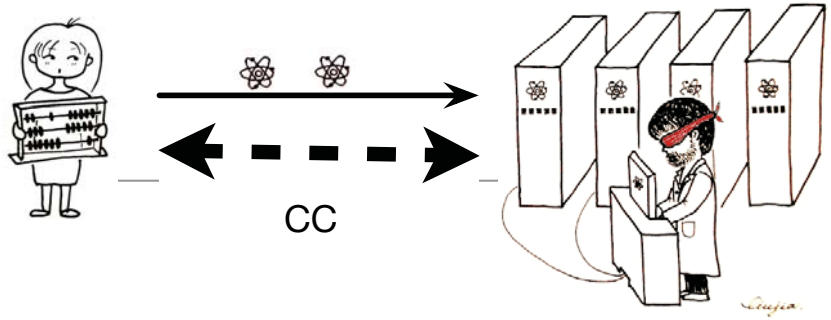
Robust and Device-independent Verification
Gheorghiu, Kashefi, Wallden NJP15



Model Dependent



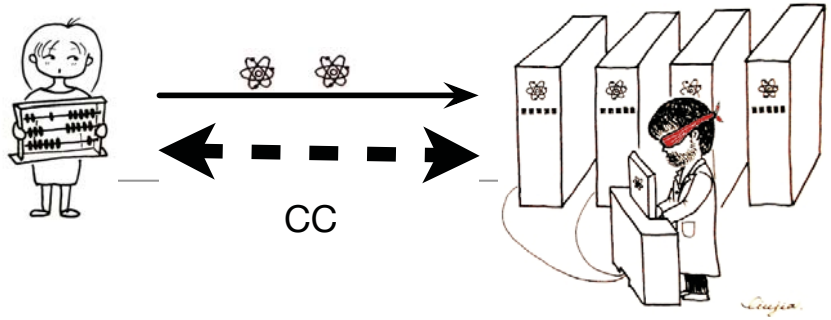
Model Dependent



Adaptation to Computational Capacity of Intermediate Models of QC

D-Wave
Boson Sampling
Instantaneous QC
Quantum Simulator

Model Dependent

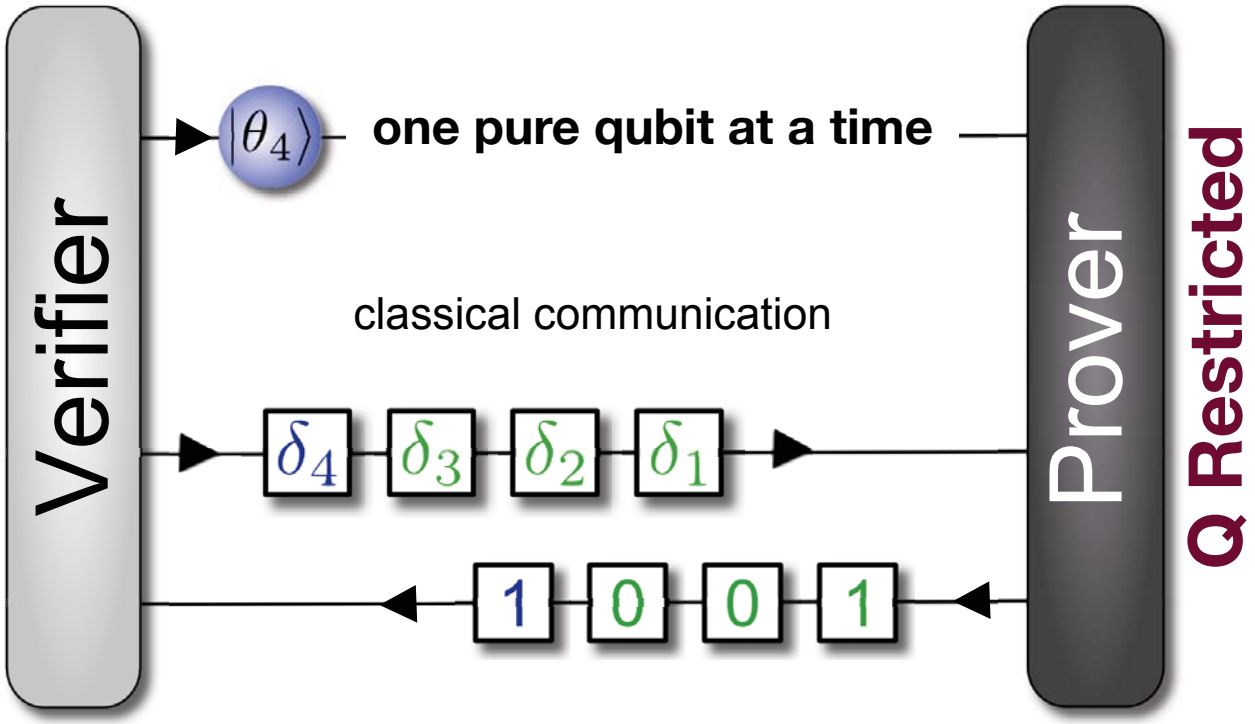


Adaptation to Computational Capacity of Intermediate Models of QC

D-Wave
Boson Sampling
Instantaneous QC
Quantum Simulator

Simplified Noise Model

Quantum Turing Test for One Pure Qubit



Kapourtionis, Kashefi, Datta, TQC, 2014

MBQC with One Pure Qubit

A runnable sequence of commands where for every elementary command the purity doesn't exceed

MBQC with One Pure Qubit

A runnable sequence of commands where for every elementary command the purity doesn't exceed

$$\pi(\rho_i) < \pi(\rho_{in}) + c$$

Input state with constant many pure qubtis

MBQC with One Pure Qubit

A runnable sequence of commands where for every elementary command the purity doesn't exceed

$$\pi(\rho_i) < \pi(\rho_{in}) + c$$

Input state with constant many pure qubtis

Constructive Def. An MBQC with surjective flow

$$P_{\boldsymbol{a}} = \prod_{i \in O} X_i^{S_i^x} Z_i^{S_i^z} \overrightarrow{\prod_{i \in O^c}} \left(S_i^z [M_i^{a_i}]^{S_i^x} \left(\prod_{\{k : k \sim i, k \succeq i\}} E_{i,k} \right) N_{f(i)}(|+\rangle) \right)$$

MBQC with One Pure Qubit

A runnable sequence of commands where for every elementary command the purity doesn't exceed

$$\pi(\rho_i) < \pi(\rho_{in}) + c$$

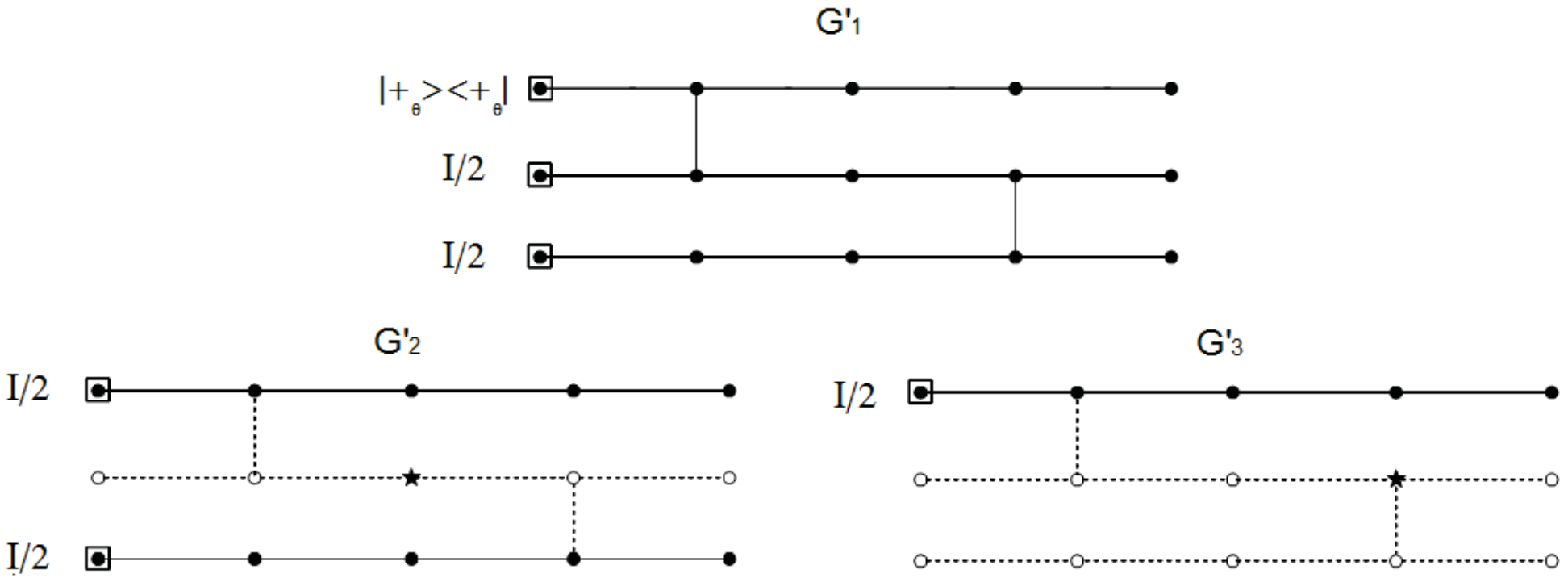
Input sta
man

UBQC comes for free in MBQC with pure qubit

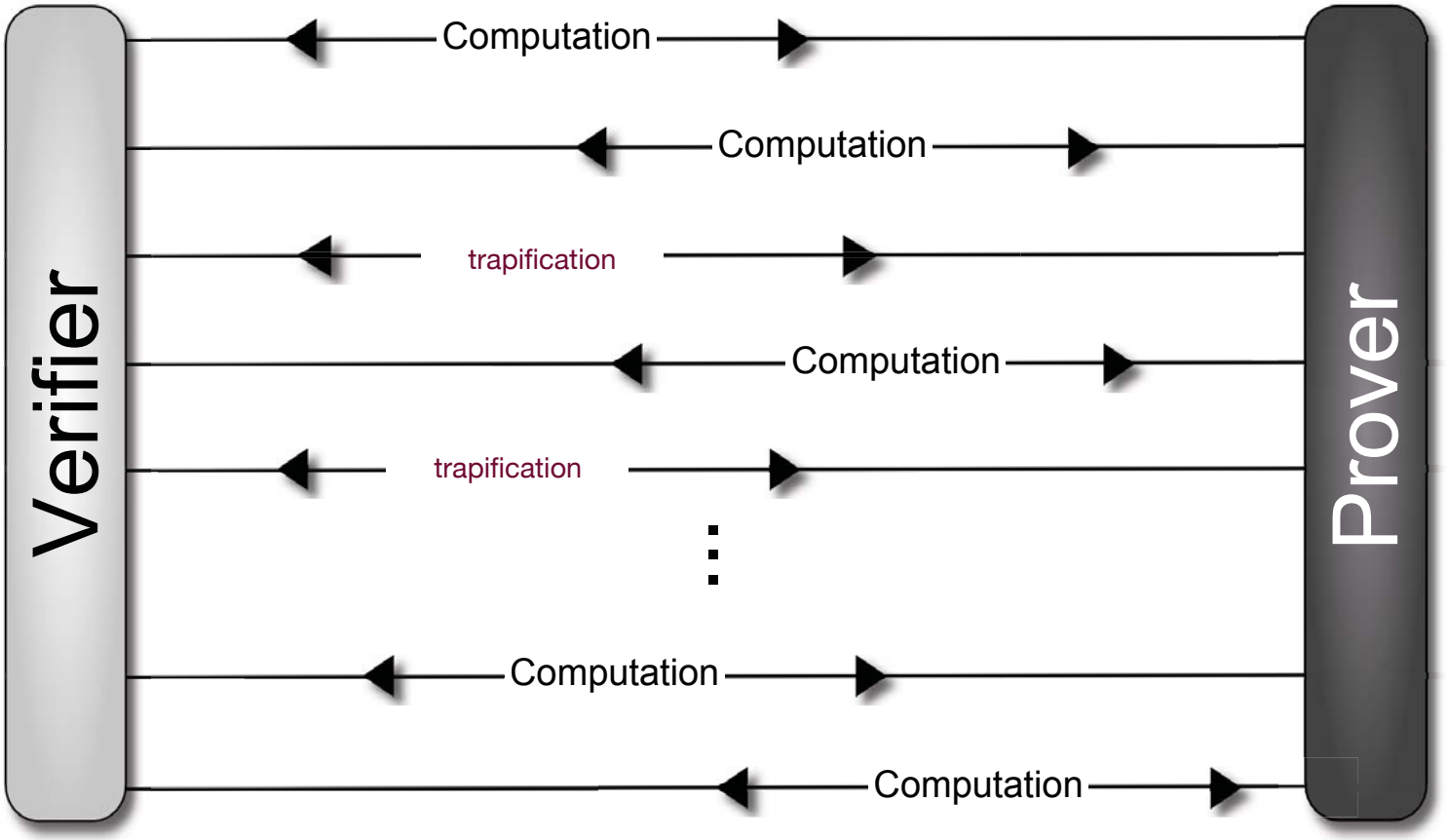
Constructive Def. An MBQC with surjective flow

$$P_{\boldsymbol{a}} = \prod_{i \in O} X_i^{S_i^x} Z_i^{S_i^z} \overrightarrow{\prod_{i \in O^c}} \left(S_i^z [M_i^{a_i}]^{S_i^x} \left(\prod_{\{k : k \sim i, k \succeq i\}} E_{i,k} \right) N_{f(i)}(|+\rangle) \right)$$

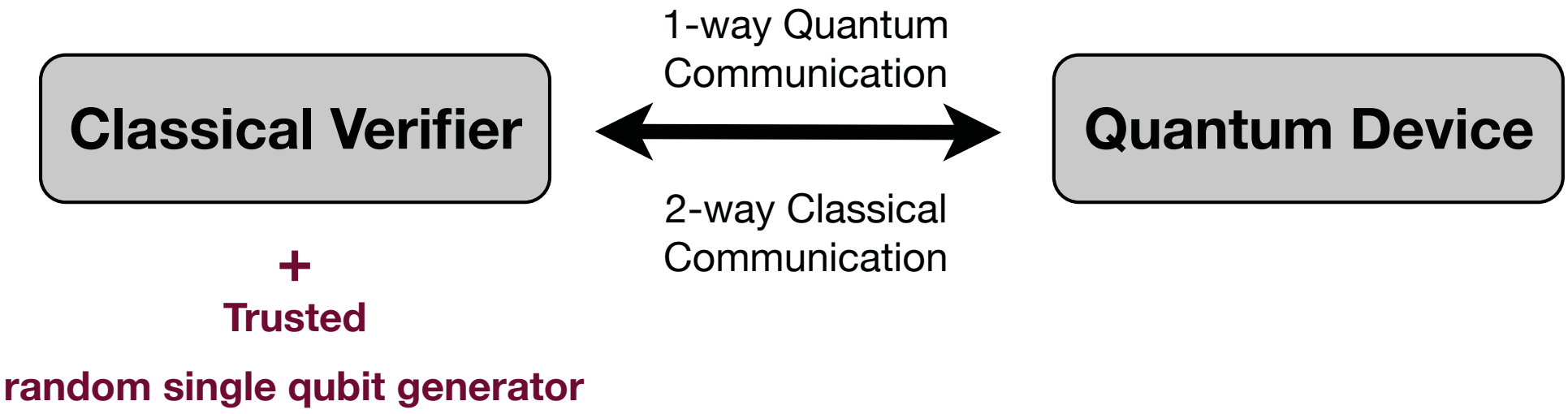
Verified Delegated QC with One Pure Qubit



Verified Delegated QC with One Pure Qubit



From QKD to verifiable quantum internet



Clients only requires (trusted) preparation of BB84 single qubits states (prepare-and-send).

[Dunjko-Kashefi 16]

From QKD to verifiable quantum internet

Classical Verifier

1-way Quantum Communication

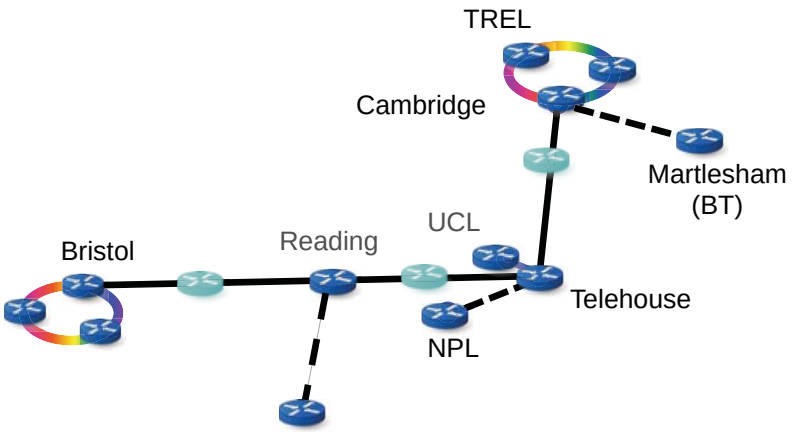
2-way Classical Communication

Quantum Device

+

Trusted

random single qubit generator



Clients only requires (trusted) preparation of BB84 single qubits states (prepare-and-send).

[Dunjko-Kashefi 16]

From QKD to verifiable quantum internet

Classical Verifier

1-way Quantum Communication

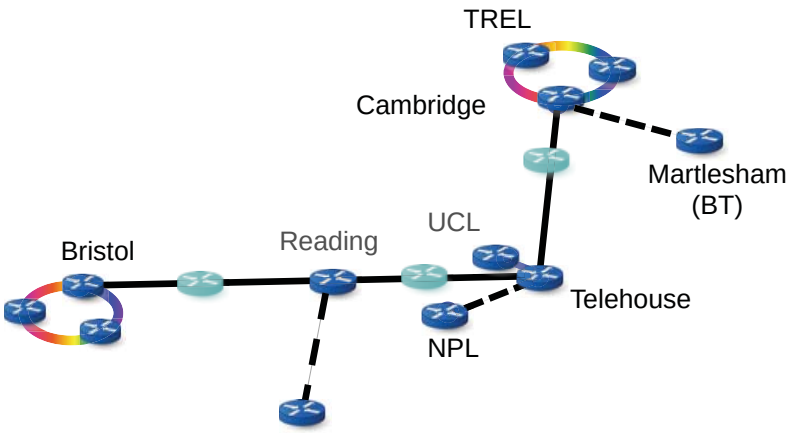
2-way Classical Communication

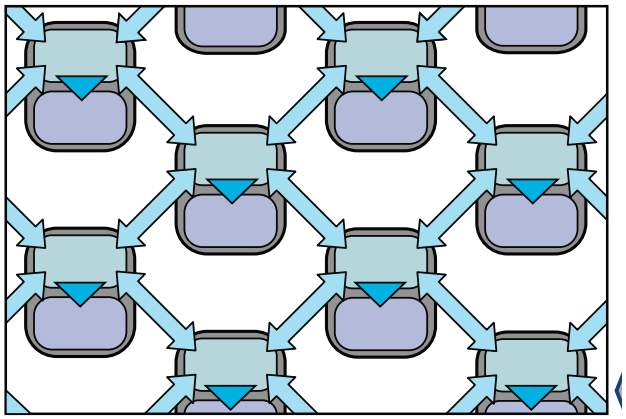
Quantum Device

+

Trusted

random single qubit generator

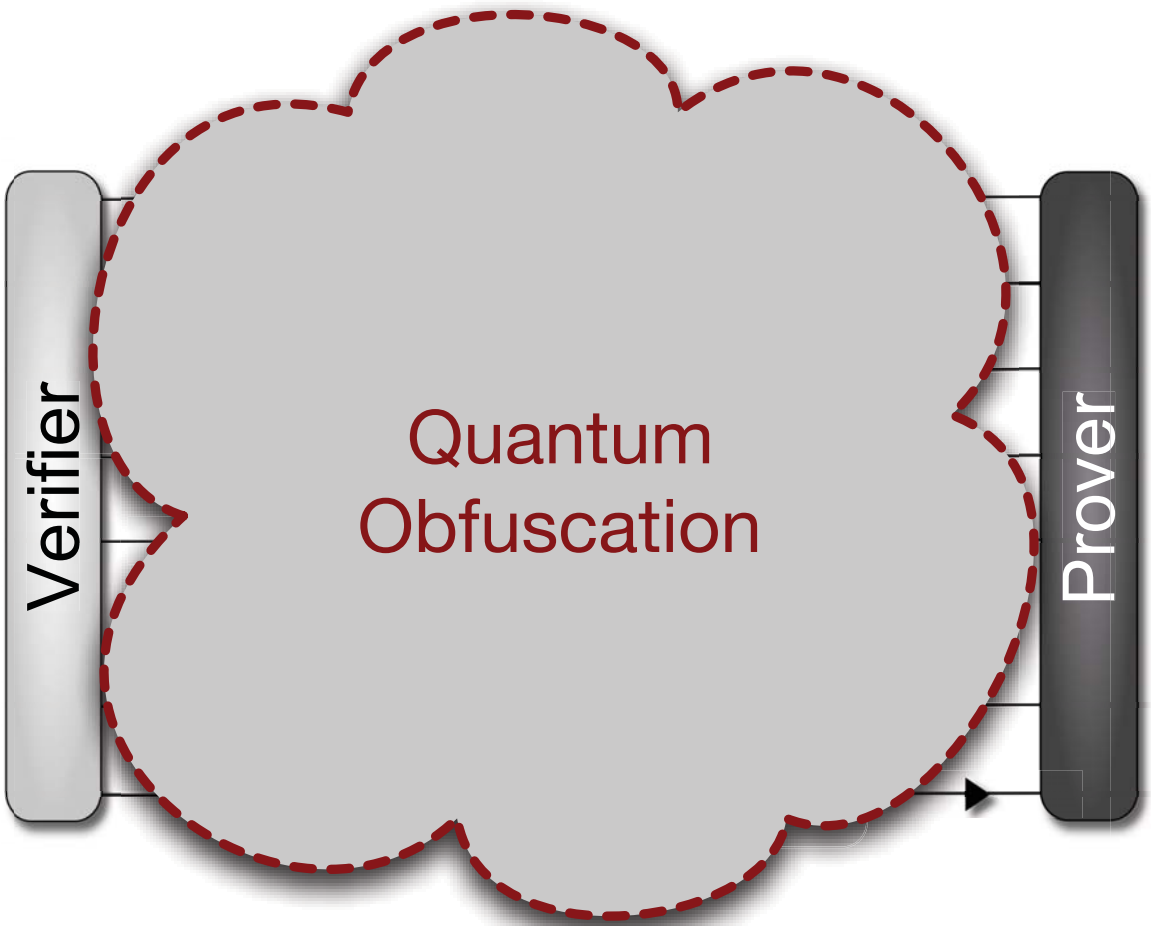




Clients only requires (trusted) preparation of BB84 single qubits states (prepare-and-send).

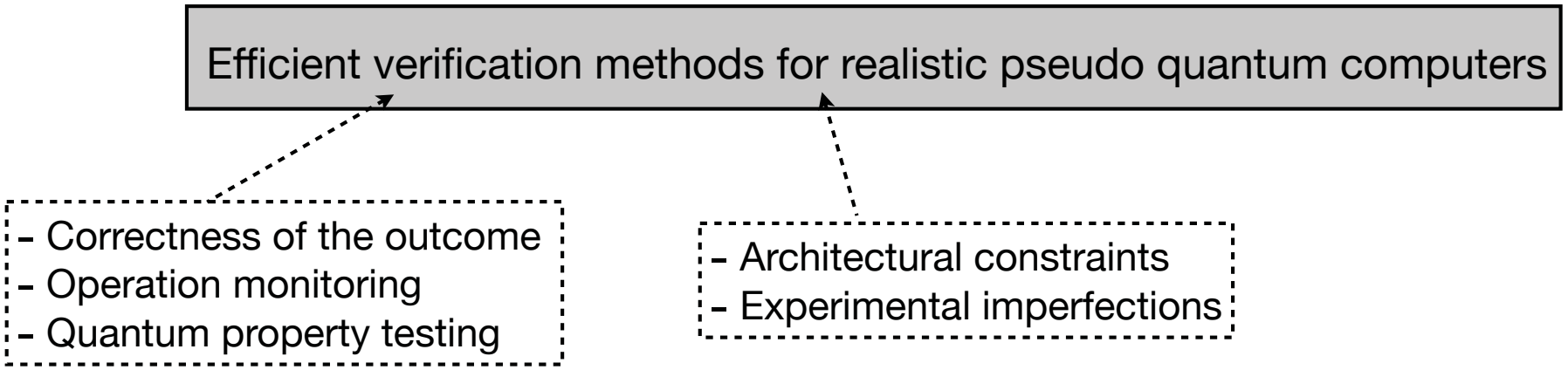
[Dunjko-Kashefi 16]

Can we get ride of qubit ?



Alagic, Fefferman, 2016

Perspective



Perspective

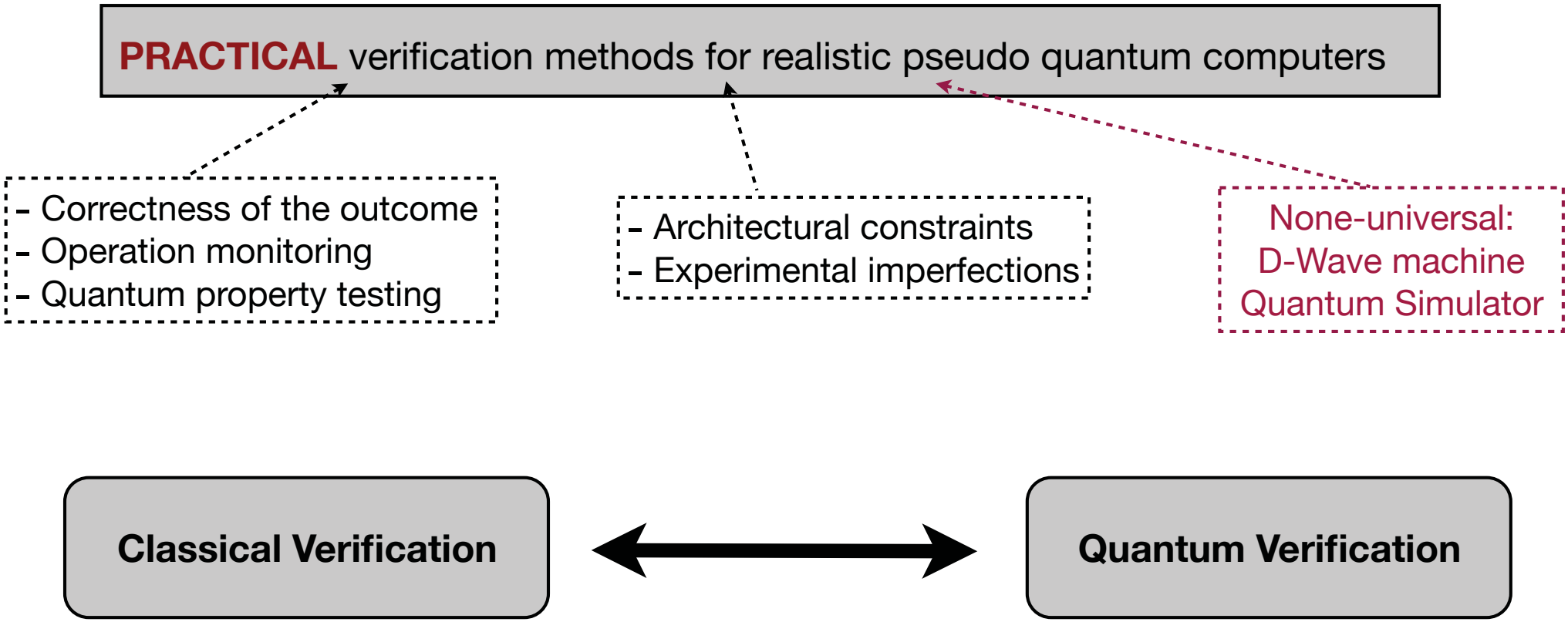
PRACTICAL verification methods for realistic pseudo quantum computers

- Correctness of the outcome
- Operation monitoring
- Quantum property testing

- Architectural constraints
- Experimental imperfections

None-universal:
D-Wave machine
Quantum Simulator

Perspective



Perspective

PRACTICAL verification methods for realistic pseudo quantum computers

- Correctness of the outcome
- Operation monitoring
- Quantum property testing

- Architectural constraints
- Experimental imperfections

None-universal:
D-Wave machine
Quantum Simulator

Classical Verification ⟷ **Quantum Verification**

Breakable Security

Server's Time

Perspective

PRACTICAL verification methods for realistic pseudo quantum computers

- Correctness of the outcome
- Operation monitoring
- Quantum property testing

- Architectural constraints
- Experimental imperfections

None-universal:
D-Wave machine
Quantum Simulator

Classical Verification ⟷ **Quantum Verification**

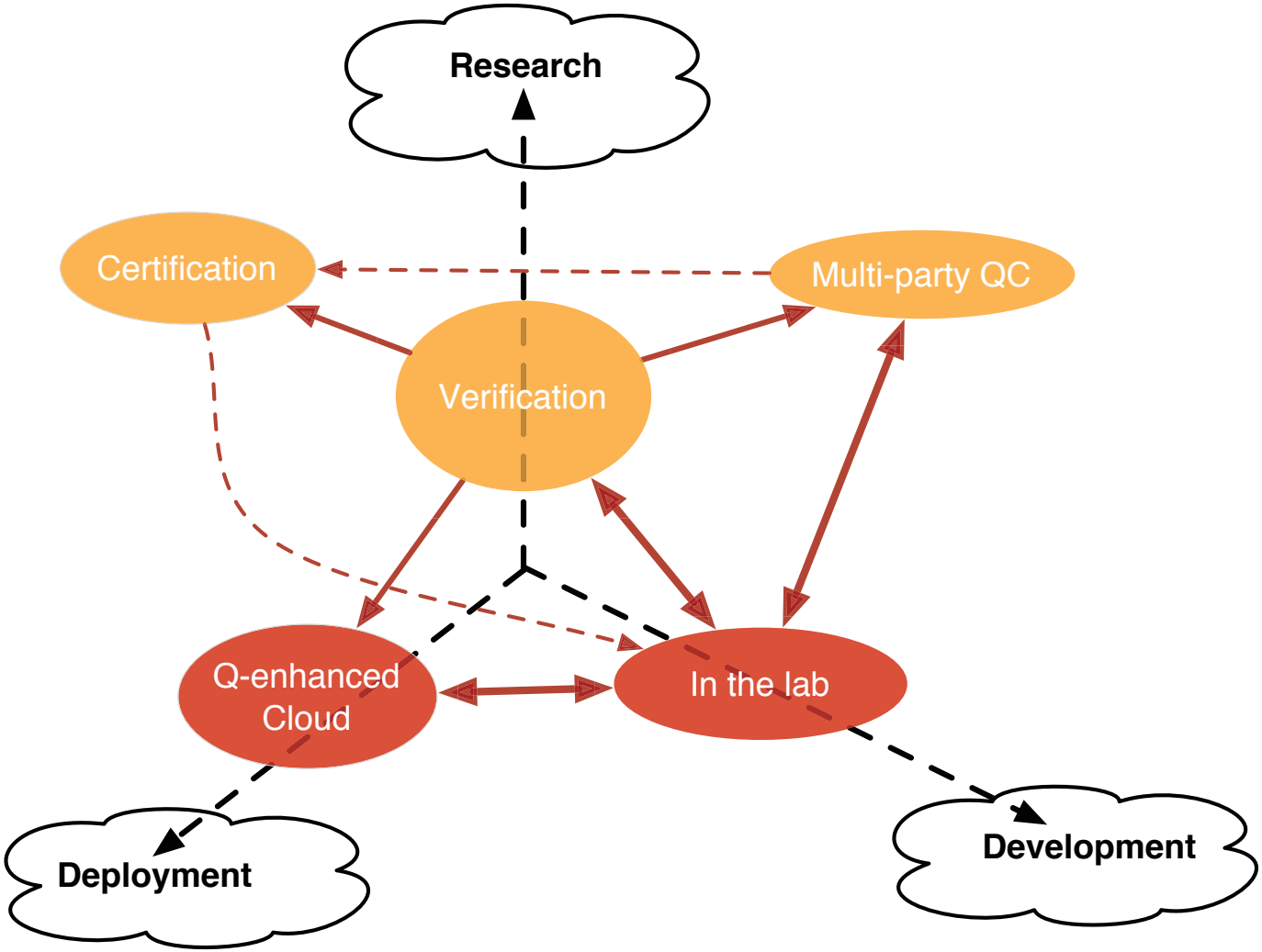
Breakable Security

Server's Time

Universal Machine

Interaction

The Edinburgh-Paris Quantum Team



The Edinburgh-Paris Quantum Team