More Distributed Quantum Merlin-Arthur Protocols: Improvement and Extension

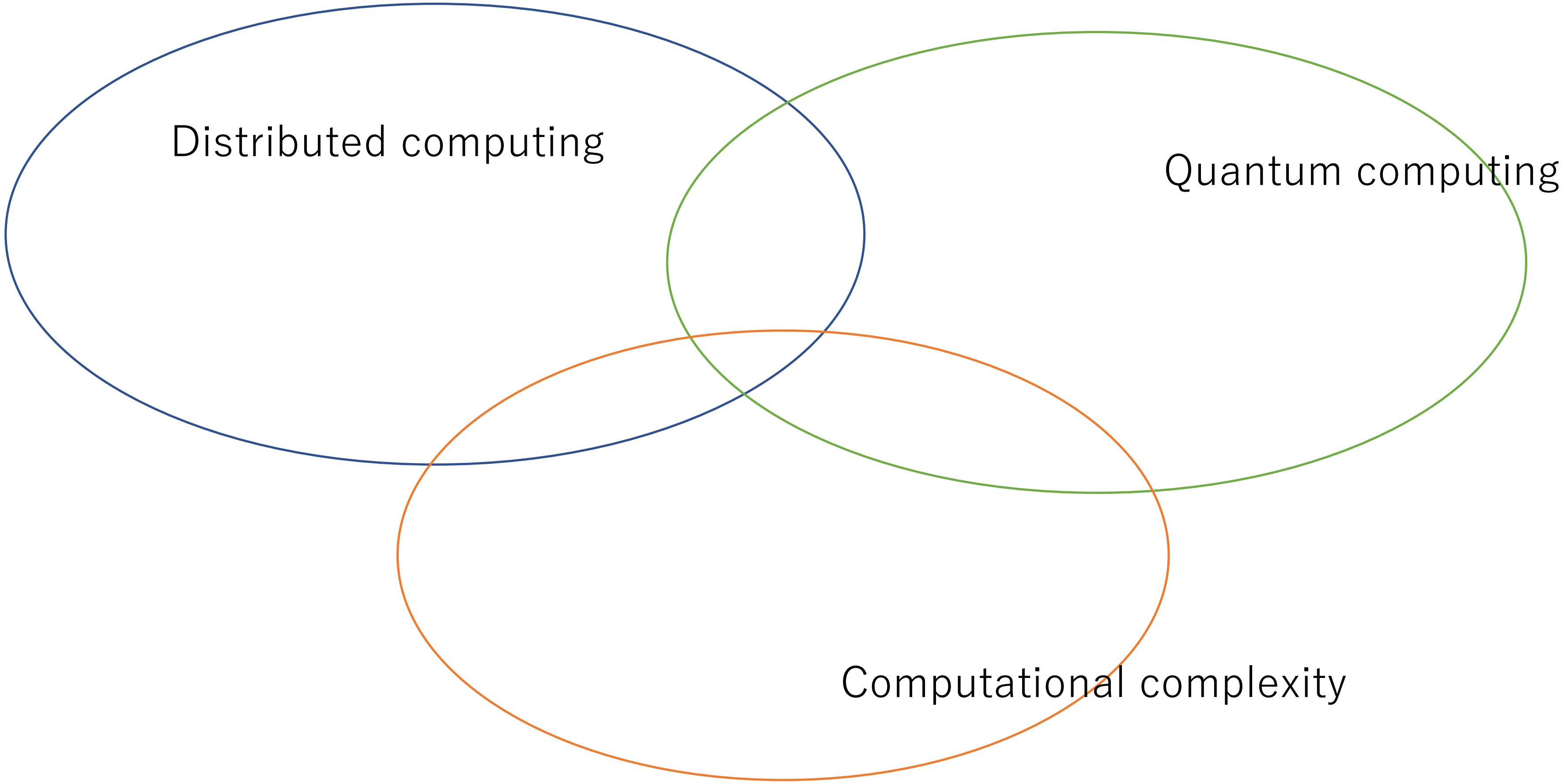
Harumichi Nishimura (Nagoya U)

Based on arXiv:2002.10018 (with P. Fraigniaud, F. Le Gall, A. Paz)
& 2210.01389 (with F. Le Gall, M. Miyamoto)

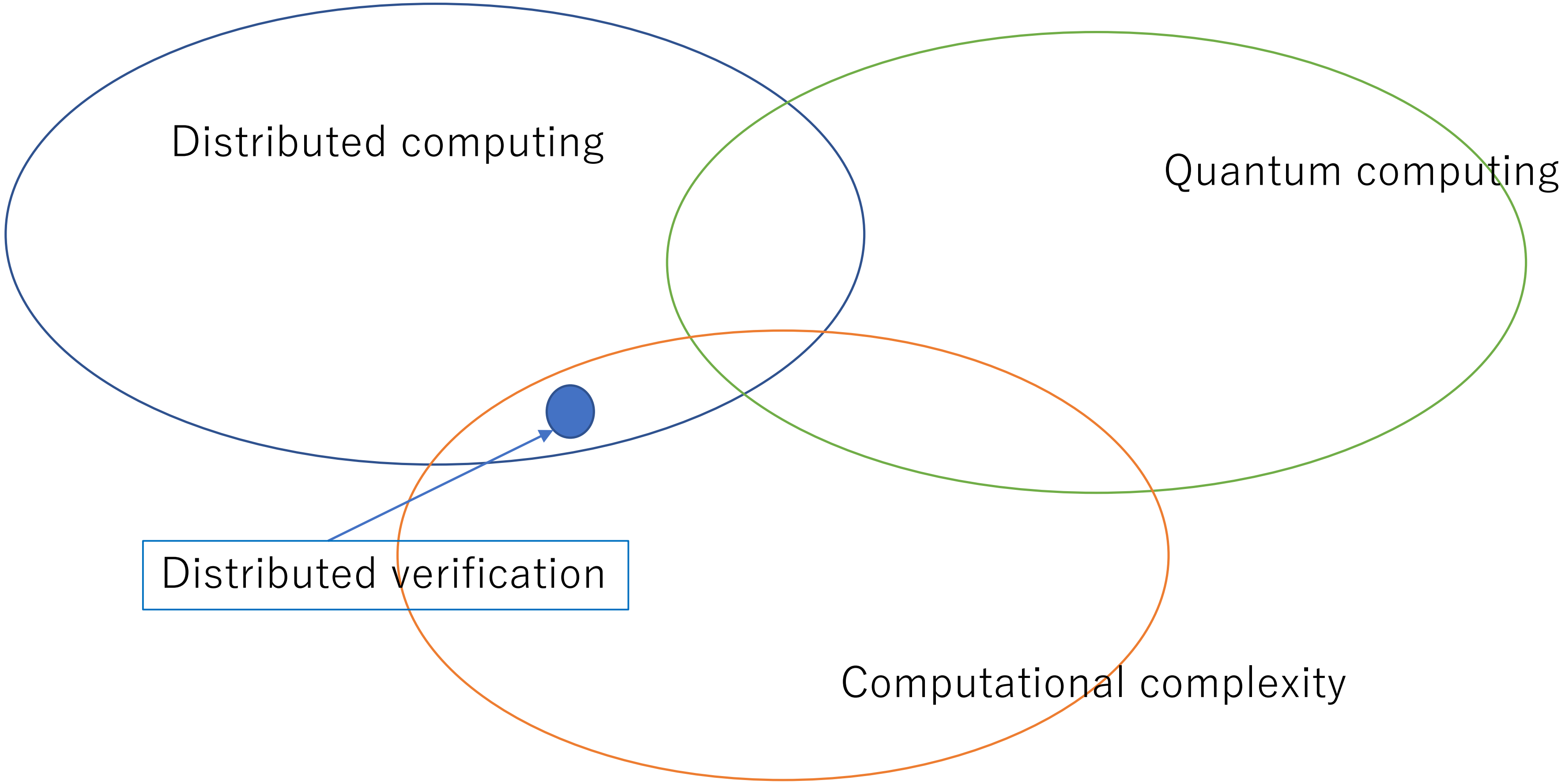
September 5, 2023

Shenzhen–Nagoya Workshop on Quantum Science 2023

Today’s talk


Distributed computing
Quantum computing
Computational complexity

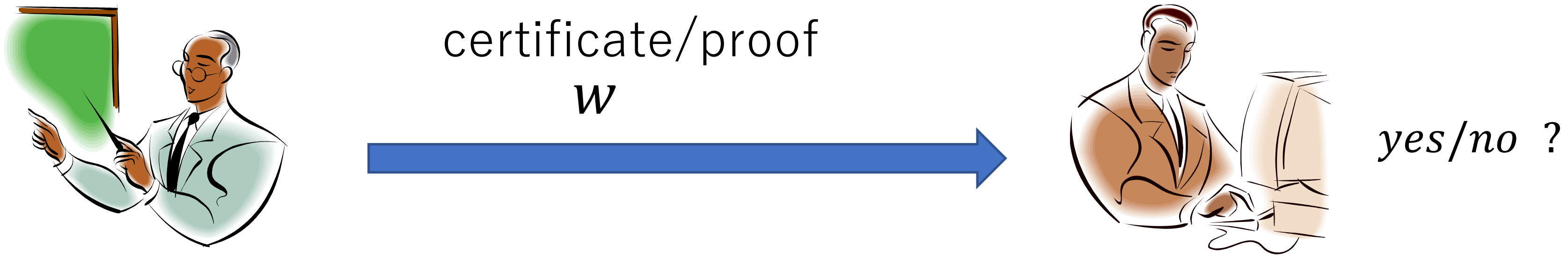
Today's talk


Distributed computing
Quantum computing
Distributed verification
Computational complexity

3 interpretations of NP

- Non-deterministic computation
  - NP:=Nondeterministic Polynomial-time
  - related classes：PP, #P
- Logical structure
  - NP=∃P, coNP=∀P, …
  - related classes: PH (polynomial-time hierarchy)
- Proof system
  - Communication protocols for verification
  - related classes：MA, AM, IP

NP as Proof Systems



Prover(Merlin)
Unlimited power

Verifier(Arthur)
Polynomial-time computation

$A = (A_{yes}, A_{no}) \in \mathrm{NP} \Leftrightarrow$

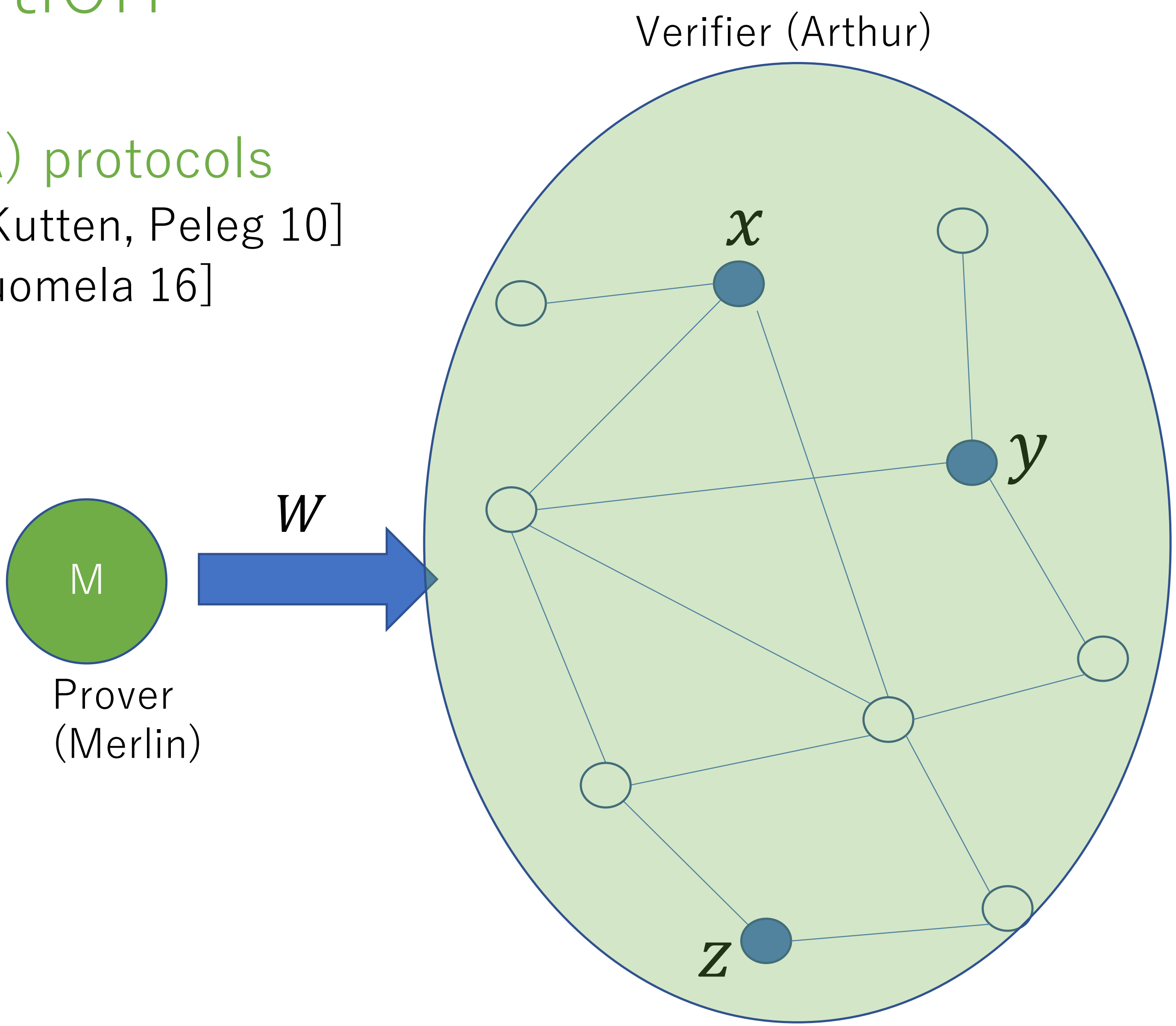
There is a polynomial-time algorithm V:

(completeness) $x \in A_{yes} \rightarrow \exists w \, [V(x, w) = \text{accept}]$
(soundness) $x \in A_{no} \rightarrow \forall w \, [V(x, w) = \text{reject}]$

Distributed certification

- Distributed Merlin-Arthur (dMA) protocols
  - Proof labeling scheme [Korman, Kutten, Peleg 10]
  - Locally checkable proof [Goos, Suomela 16]

  etc
- Input
  - Graph (structure of the network)
  - Strings for nodes



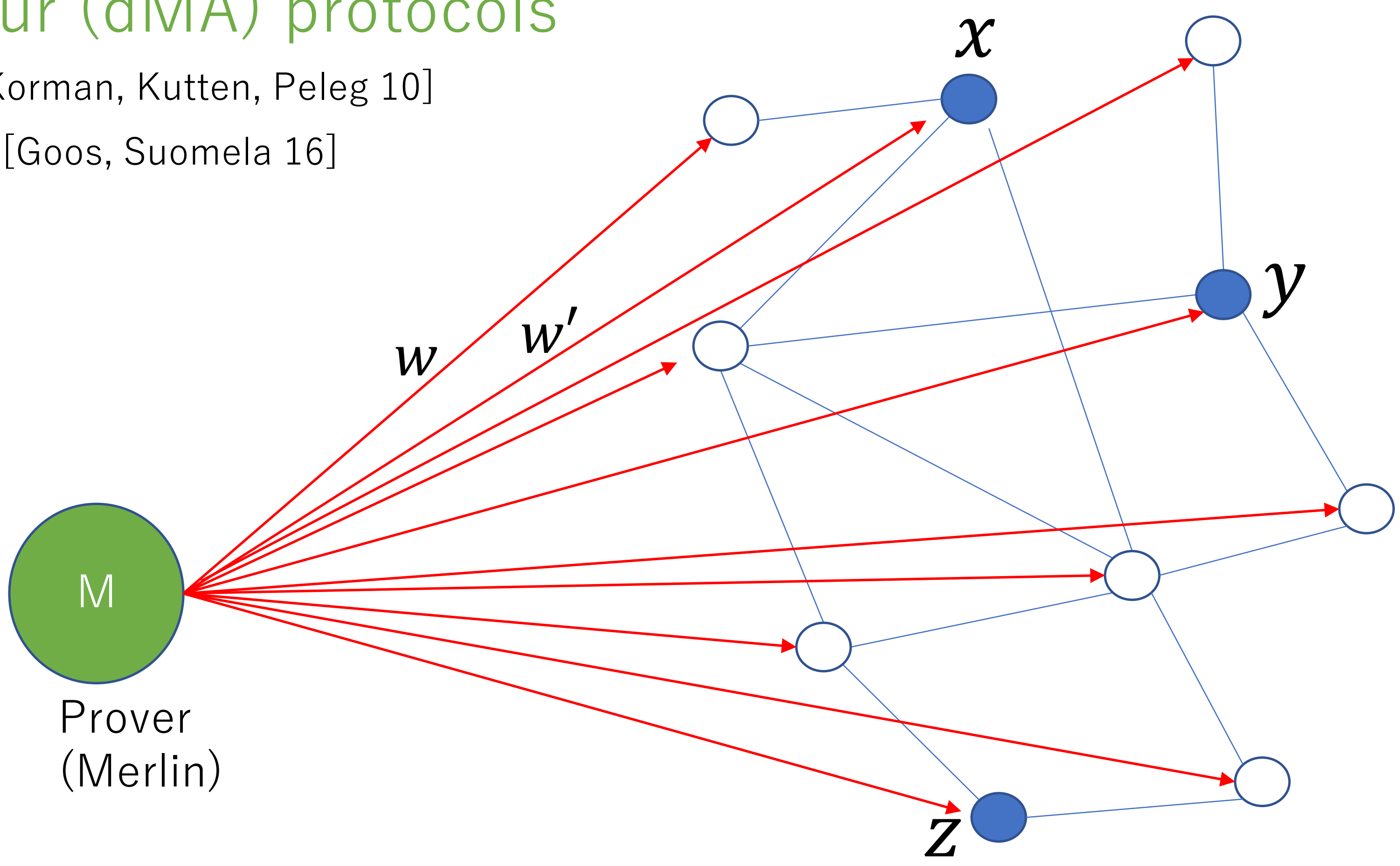
Distributed Certification

- Distributed Merlin-Arthur (dMA) protocols
  - Proof labeling scheme [Korman, Kutten, Peleg 10]
  - Locally checkable proof [Goos, Suomela 16]

  etc

Two phases:

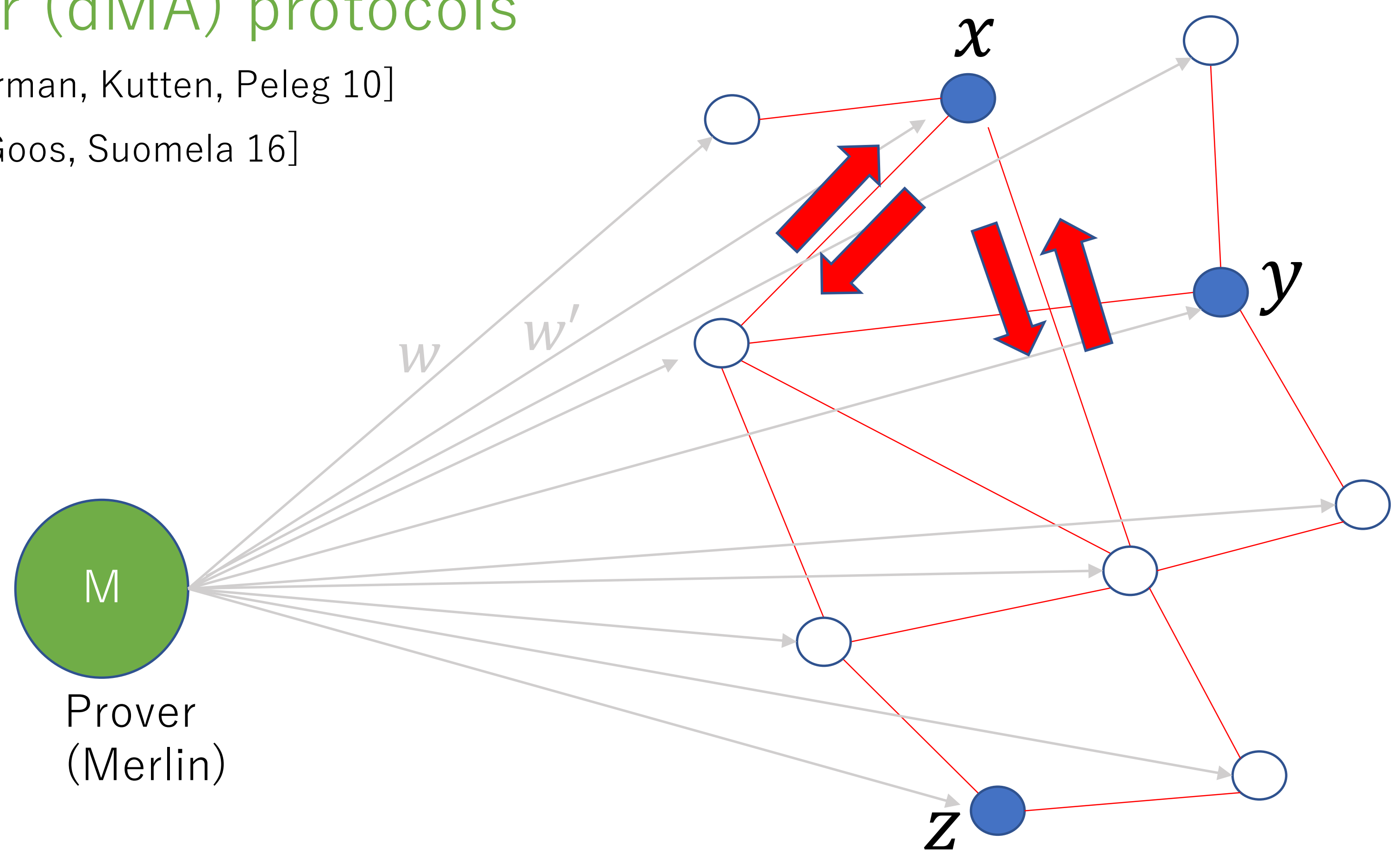
1. (Prover phase) Prover sends certificates to each node



Distributed Certification

- Distributed Merlin-Arthur (dMA) protocols
  - Proof labeling scheme [Korman, Kutten, Peleg 10]
  - Locally checkable proof [Goos, Suomela 16]

  etc



Two phases:

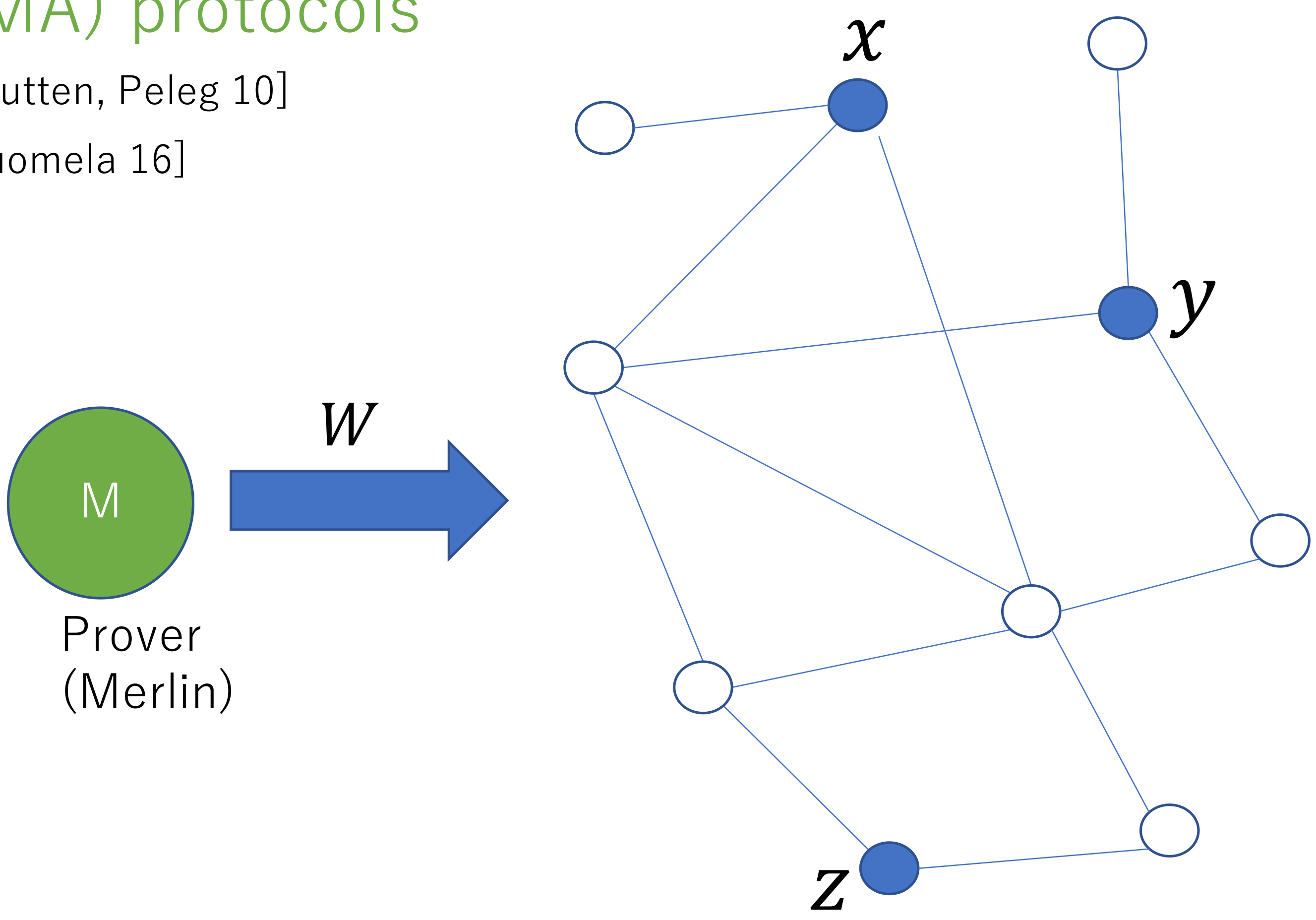
1. (Prover phase) Prover sends certificates to each node
2. (Verification phase) Each node exchanges messages with the neighbors

Distributed Certification

- Distributed Merlin-Arthur (dMA) protocols
  - Proof labeling scheme [Korman, Kutten, Peleg 10]
  - Locally checkable proof [Goos, Suomela 16]

  etc

<u>Properties</u>:
(YES case: Completeness)
$\exists w$[all nodes accept]
(w.h.p.)
(NO case: Soundness)
$\forall w$[some node rejects]
(w.h.p.)



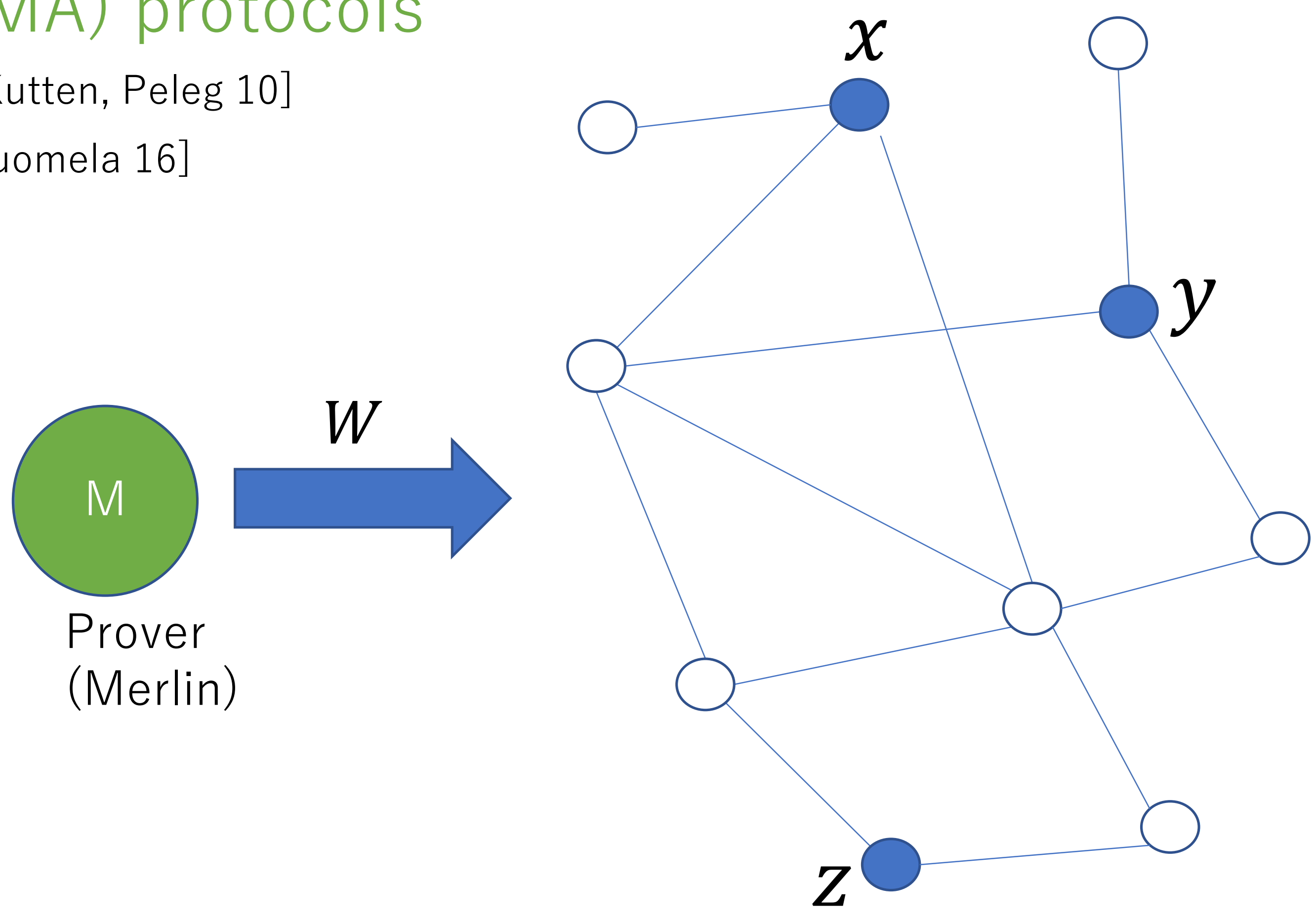
Distributed Certification

- Distributed Merlin-Arthur (dMA) protocols
  - Proof labeling scheme [Korman, Kutten, Peleg 10]
  - Locally checkable proof [Goos, Suomela 16]

  etc

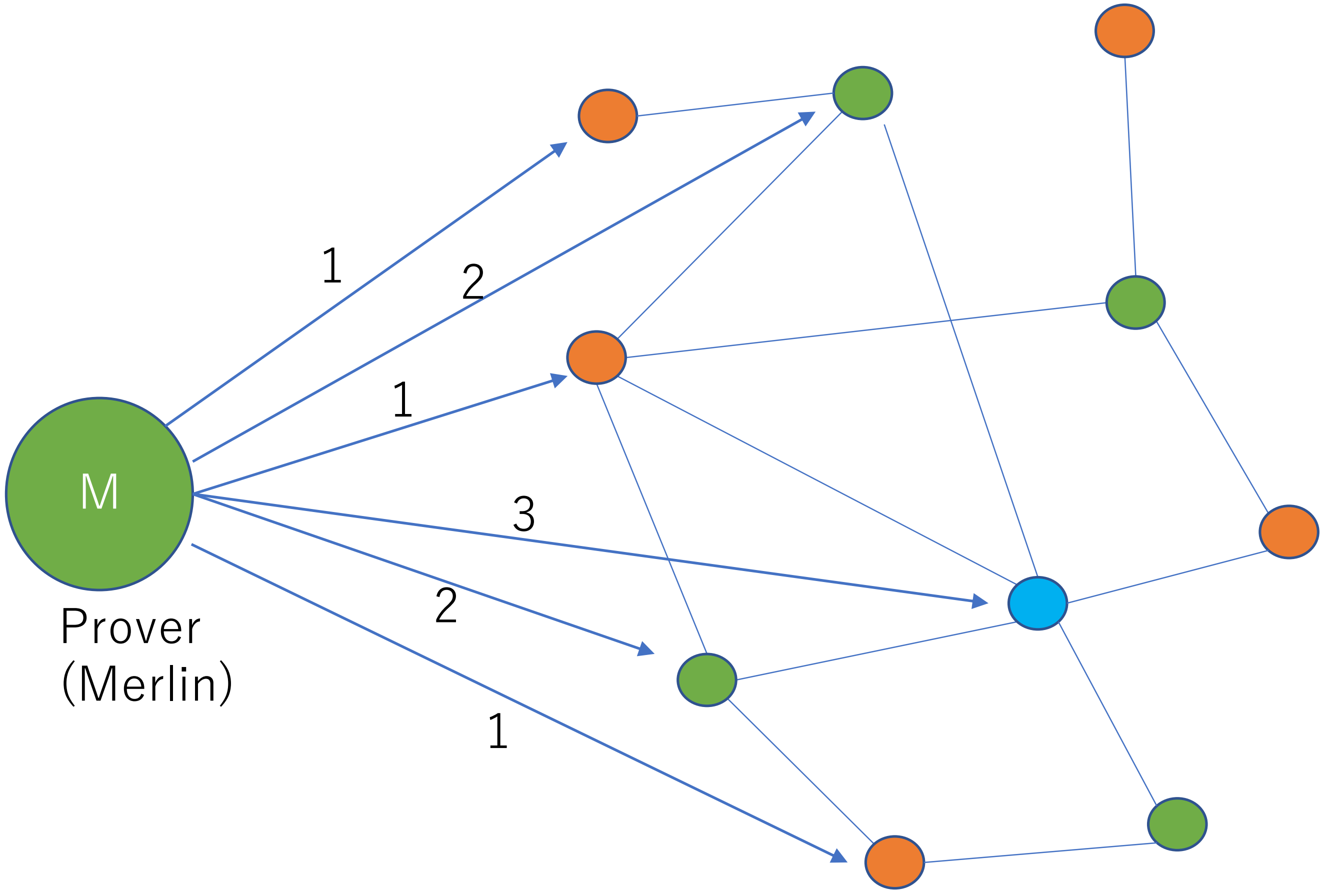
<u>Complexity parameters</u>:

- Certificate size
  - Length of a message which the prover sends to <u>each node</u>
- Message size
  - Length of messages sent on <u>each edge</u>

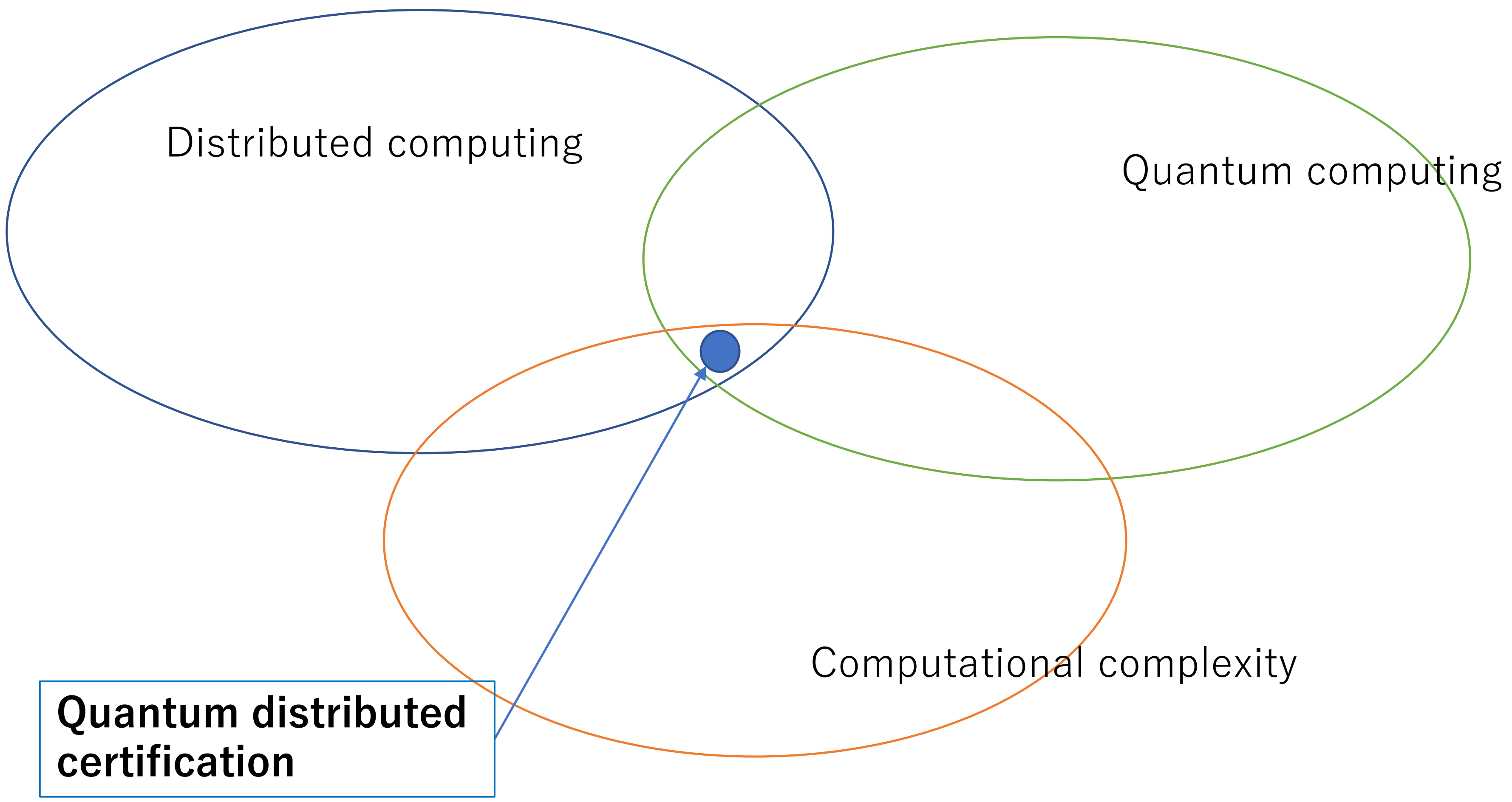


Ex: 3-colorability

- Input
  - Graph $G = (V, E)$
- Output
  - Is G 3-colorable?
- Protocol
  - Honest prover sends a color to each node such that their colors make 3-coloring of G
  - Each node checks whether the color is different from that of the neighbors
  - Certificate size O(1)
  - Message size O(1)



Today's talk



Distributed computing
Quantum computing
Computational complexity
Quantum distributed certification

QMA: Quantum NP [Knill, Kitaev, Watrous]

Prover (Merlin)
Unlimited power
= Any quantum operations

Quantum proof

$|\varphi\rangle$

yes/no ?

Verifier (Arthur)
Polynomial-time
Quantum algorithm

$A \in$ QMA $\Leftrightarrow$

There is a polynomial-time quantum algorithm V:

(completeness) $x \in A_{yes} \to \exists|\varphi\rangle$: $\Pr[V(x, |\varphi\rangle) = \text{accept}] \geq 2/3$

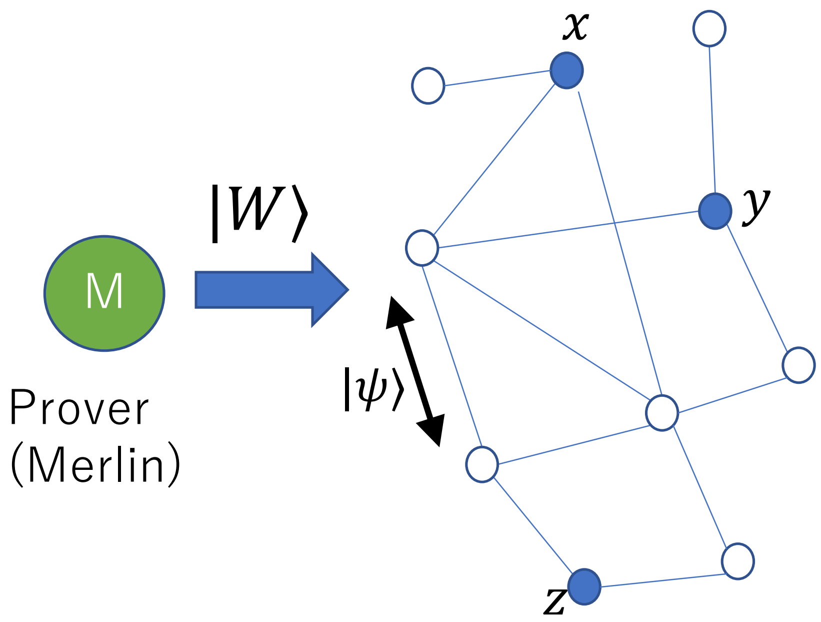
(soundness) $x \in A_{no} \to \forall|\varphi\rangle$: $\Pr[V(x, |\varphi\rangle) = \text{reject}] \geq 2/3$

Distributed Quantum Merlin-Arthur (dQMA)

[FLNP20]

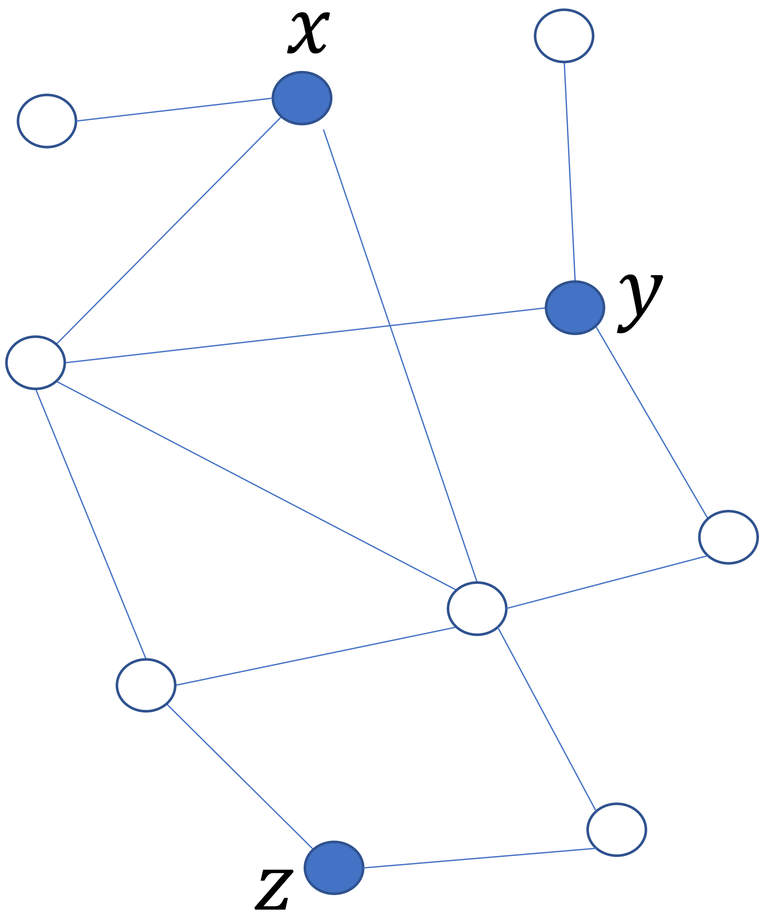
- Distributed Quantum Merlin-Arthur (dQMA) protocols on the network
  - Quantum certificates from the prover
  - Quantum messages among nodes

Q. Which problems are efficient for dQMA protocols?



EQ: Equality of Data

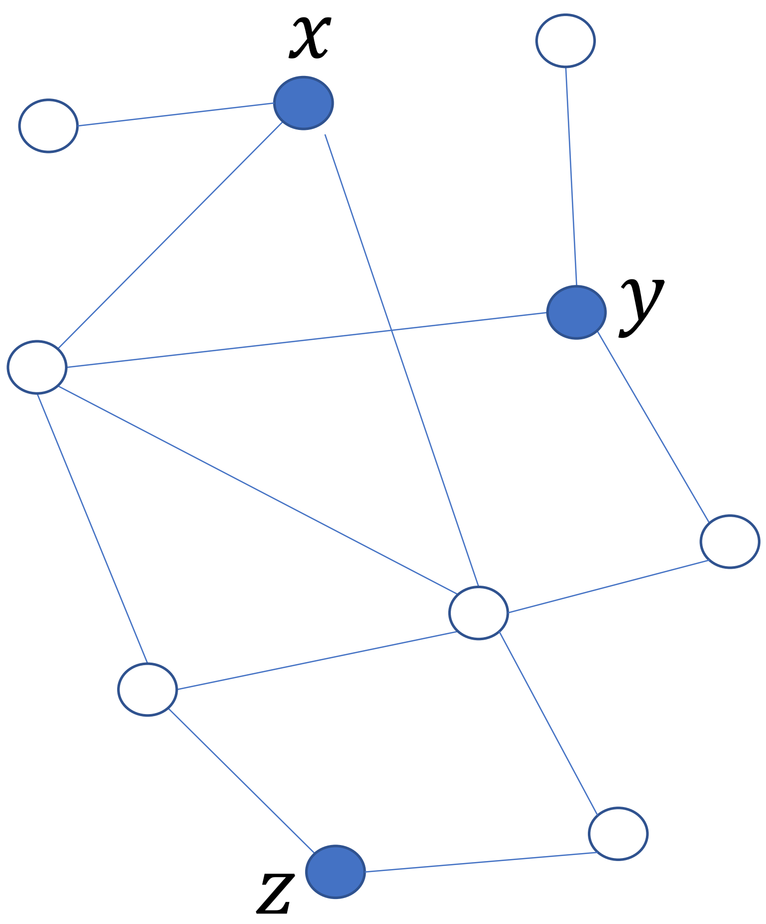
- Replicated data on a network
- Are all data identical?



 terminals (nodes who have data)

EQ: Equality of Data

- Replicated data on a network
- Are all data identical?
- No O(1) round protocol
  - $\Omega(r)$ rounds are needed (r : diameter of the network)
  - We assume **the nodes do not share prior randomness** (& entanglement)
- ∃ 1 round "NP-like" protocol (distributed certification)

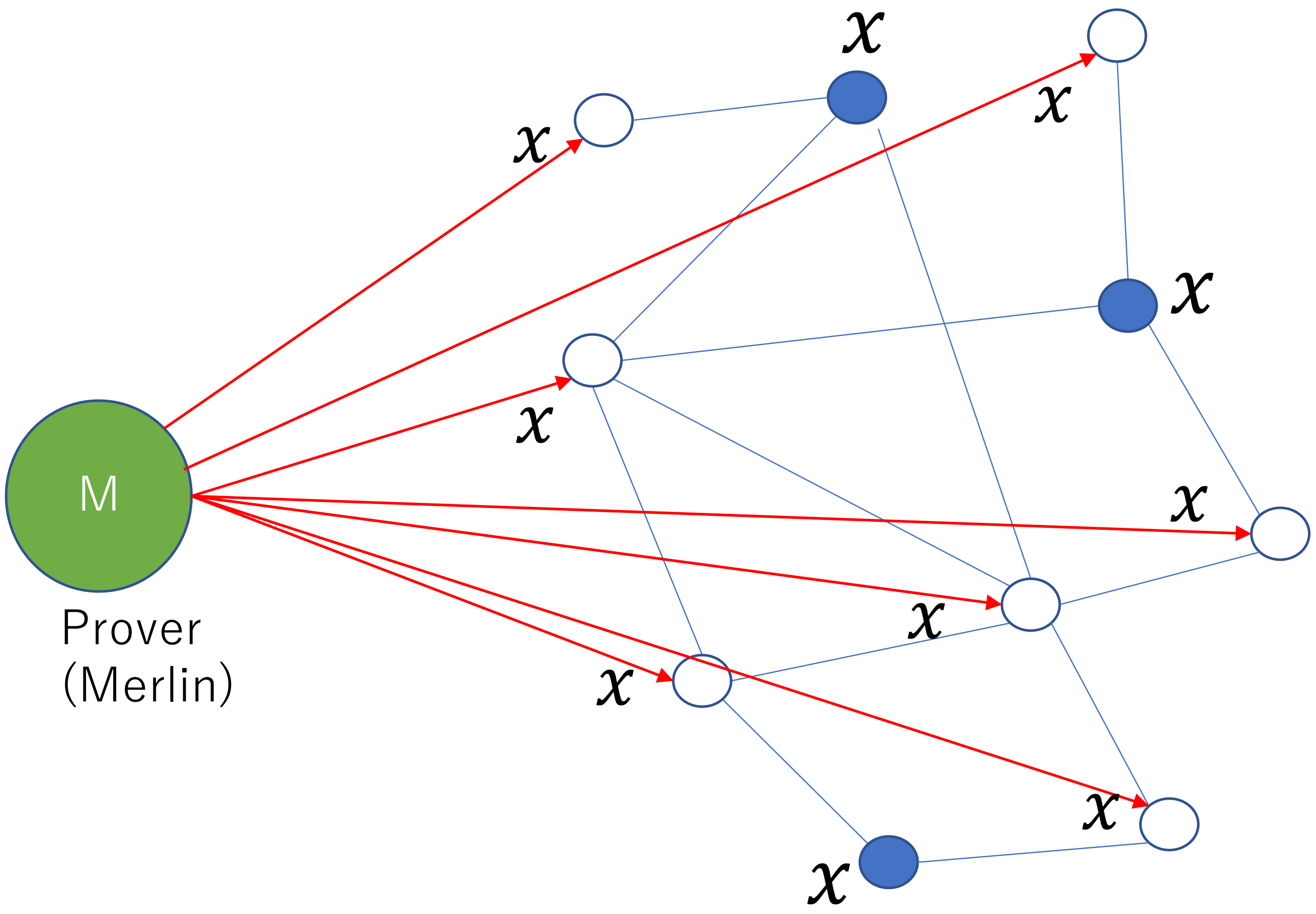


terminals (nodes who have data)

dMA Protocol for EQ

Trivial protocol:
(P) Prover M sends x when all data are x
(V) Each node checks if it is same as the neighbor's one

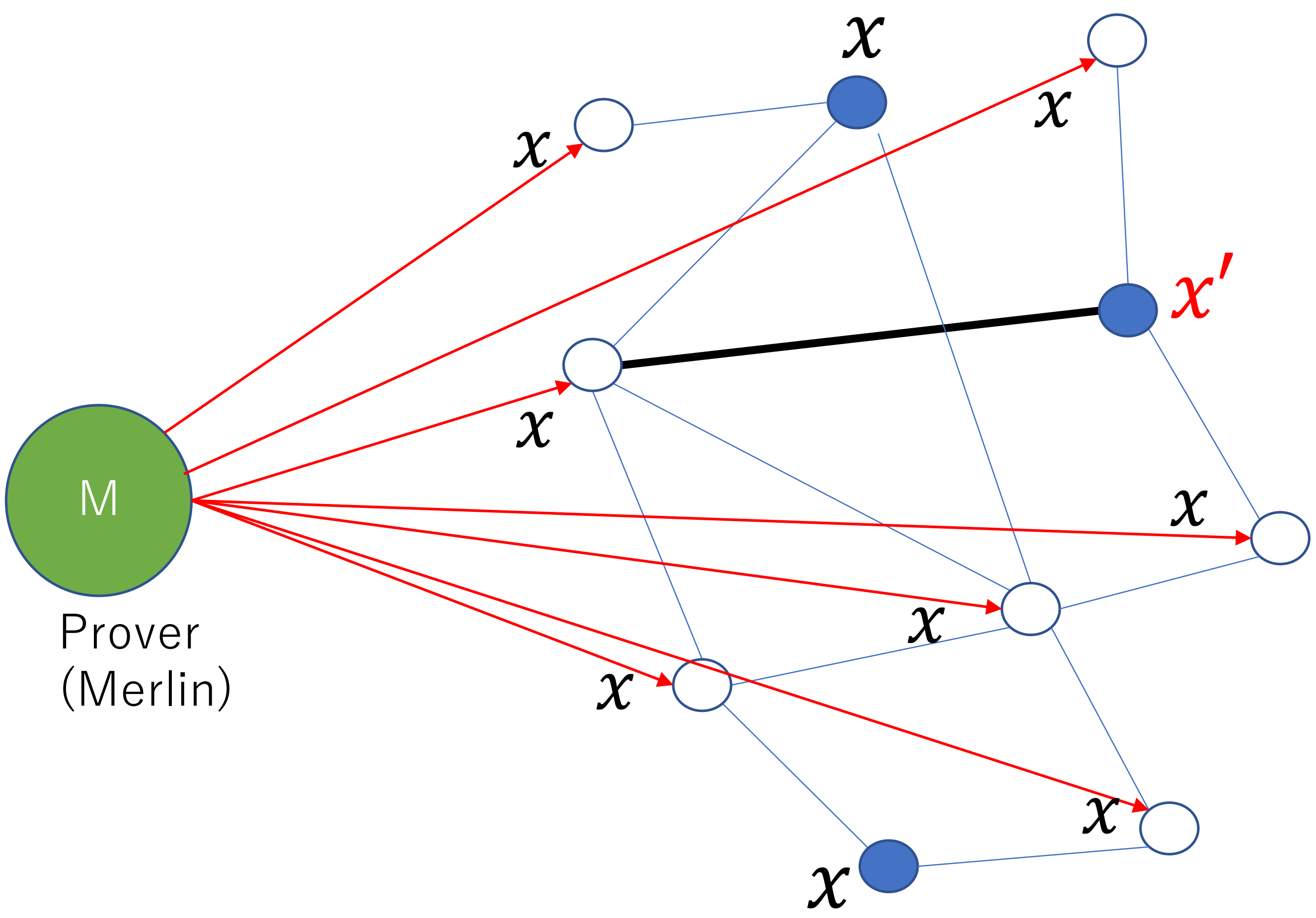
(YES case: Completeness)
$\exists W$[all nodes accept]



dMA Protocol for EQ

Trivial protocol:
(P) Prover M sends x when all data are x
(V) Each node checks if it is same as the neighbor's one

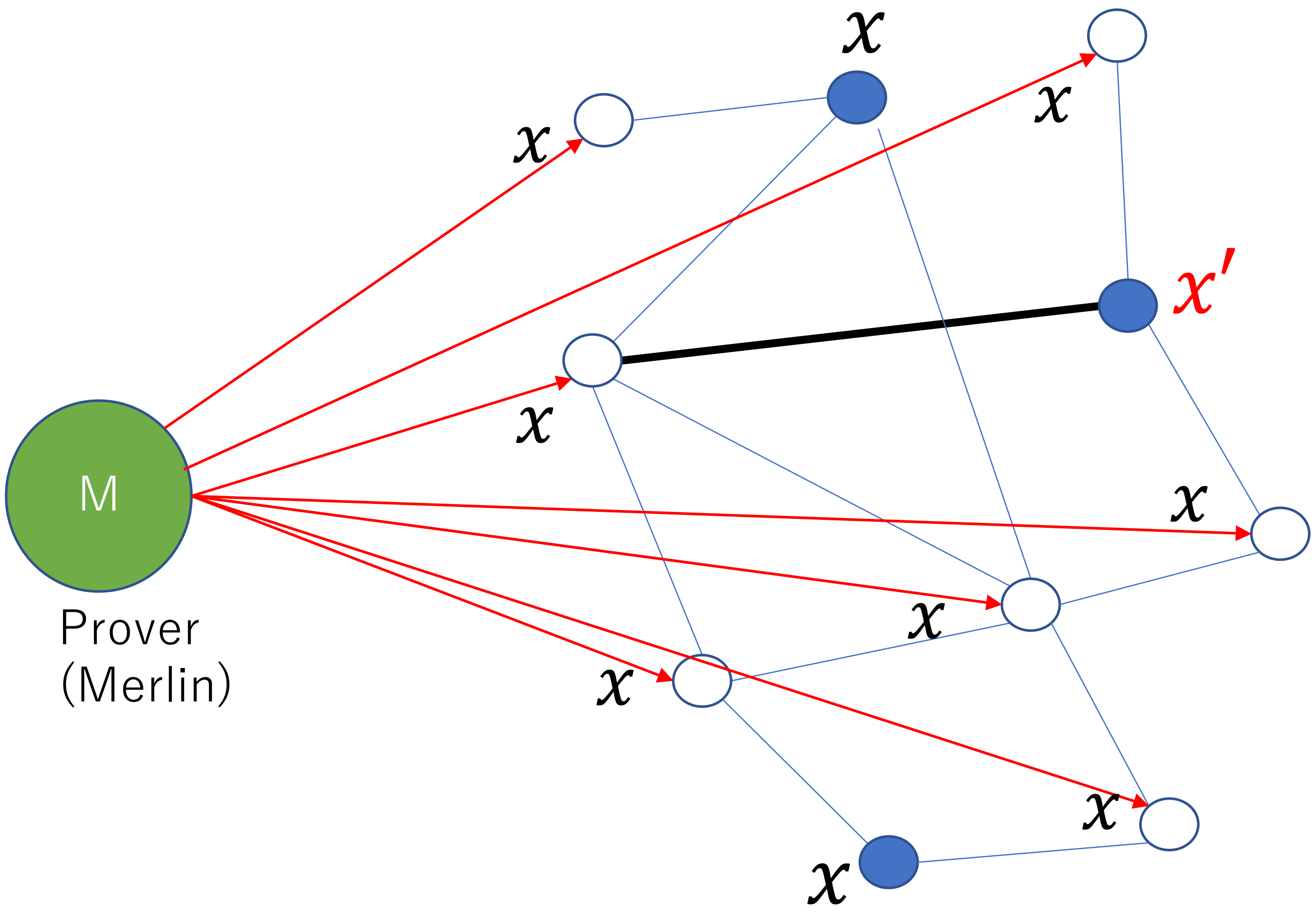
(NO case: Soundness)
$\forall w$[some node rejects]



dMA Protocol for EQ

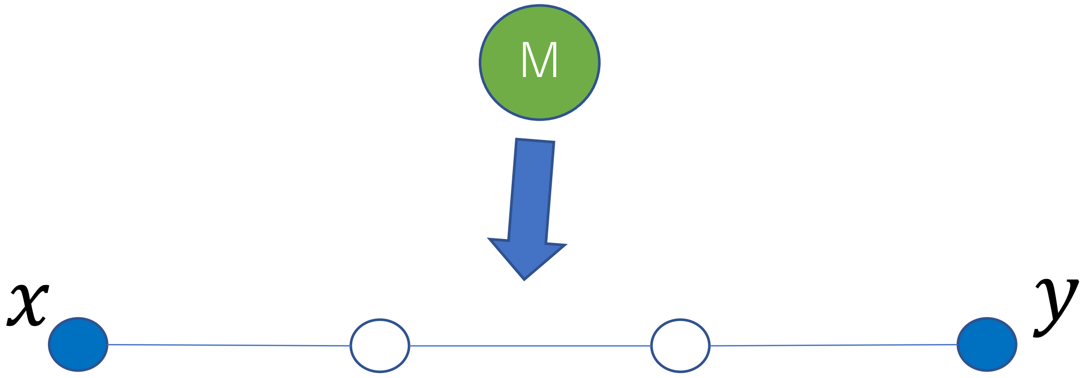
<u>Trivial Protocol is communication inefficient</u>

- Prover M sends n bits for each node ($n :=$ length of x)
- Each node sends n bits to the neighbors



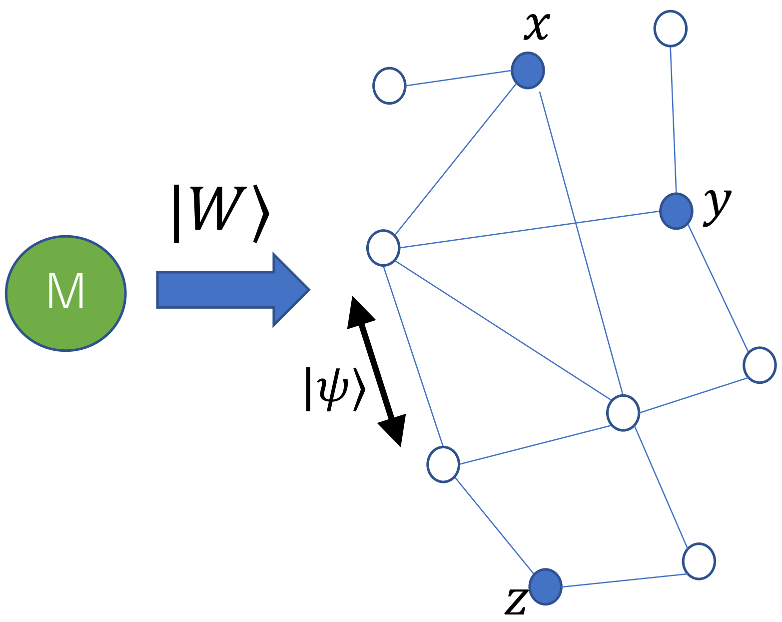
Results for EQ [FLNP20]

- Distributed Quantum Merlin-Arthur (dQMA) protocols on the network
  - Quantum certificates from the prover
  - Quantum messages among nodes
- Classical lower bound for EQ
  - Any dMA protocol requires $\Omega(n)$-bit certificates if error probability is reasonably small (say, 1/4)



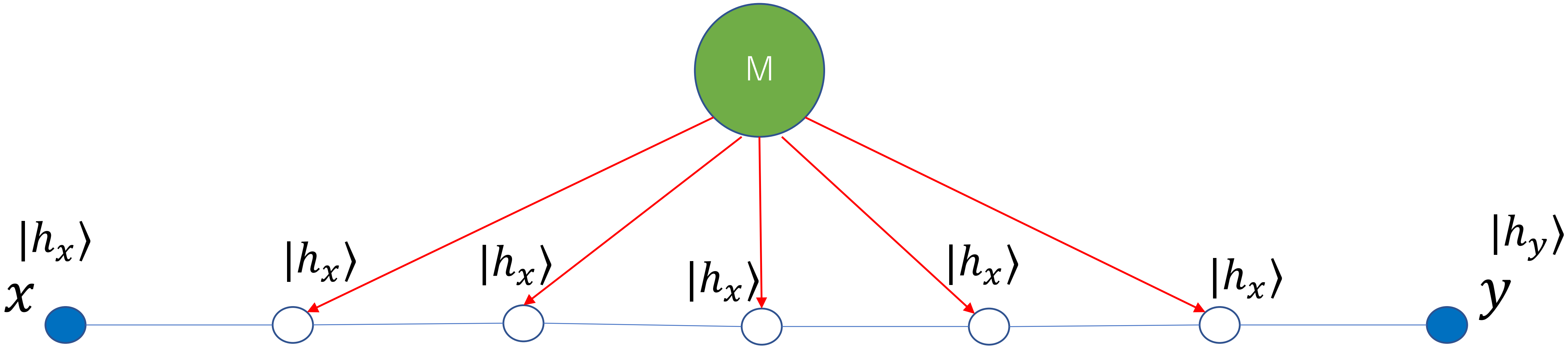
Results for EQ [FLNP20]

- Distributed Quantum Merlin-Arthur (dQMA) protocols on the network
  - Quantum certificates from the prover
  - Quantum messages among nodes
- Classical lower bound for EQ
  - Any dMA protocol requires $\Omega(n)$-bit certificates if error probability is reasonably small (say, 1/4)
- Quantum upper bound for EQ
  - ∃ dQMA protocol for equality of replicated data with $O(tr^2 \log(n+r))$-qubit certificates & messages
    - $t :=$ number of the terminals (= nodes who have data)
    - $r :=$ diameter of the network
    - **t and r are typically much smaller than n**



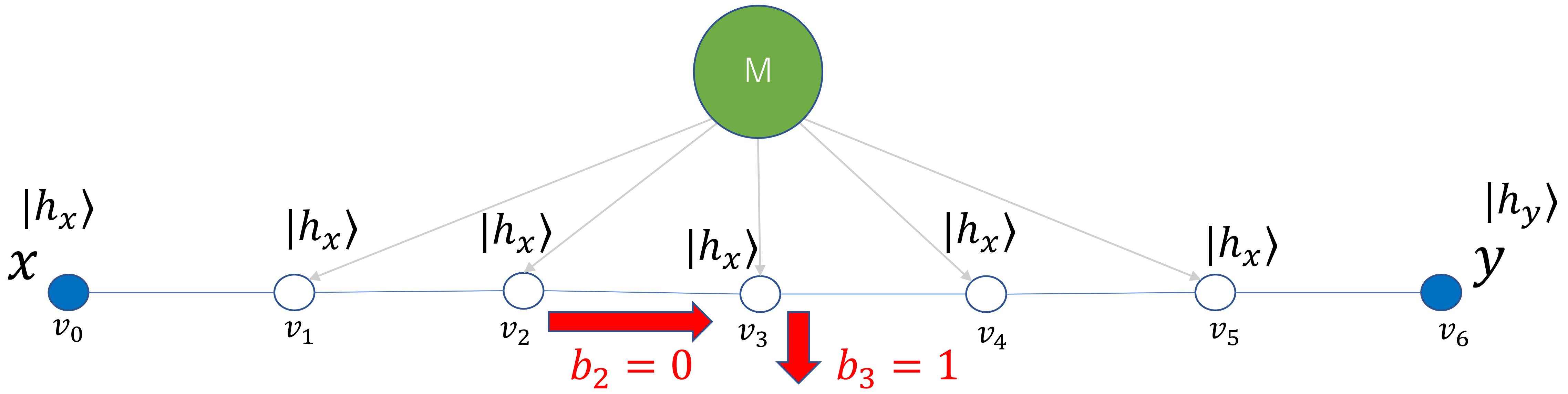
KLNP20 Protocol for a line (Prover phase)

- Honest prover (when $x = y$) sends certificate $|h_x\rangle$ (quantum fingerprint of x [BCWW01]) to each of the intermediate nodes
  - $|h_x\rangle$ is almost orthogonal to $|h_y\rangle$ if $x \neq y$
  - Length of $|h_x\rangle$ is $O(\log n)$
- The left node creates $|h_x\rangle$ and the right node creates $|h_y\rangle$



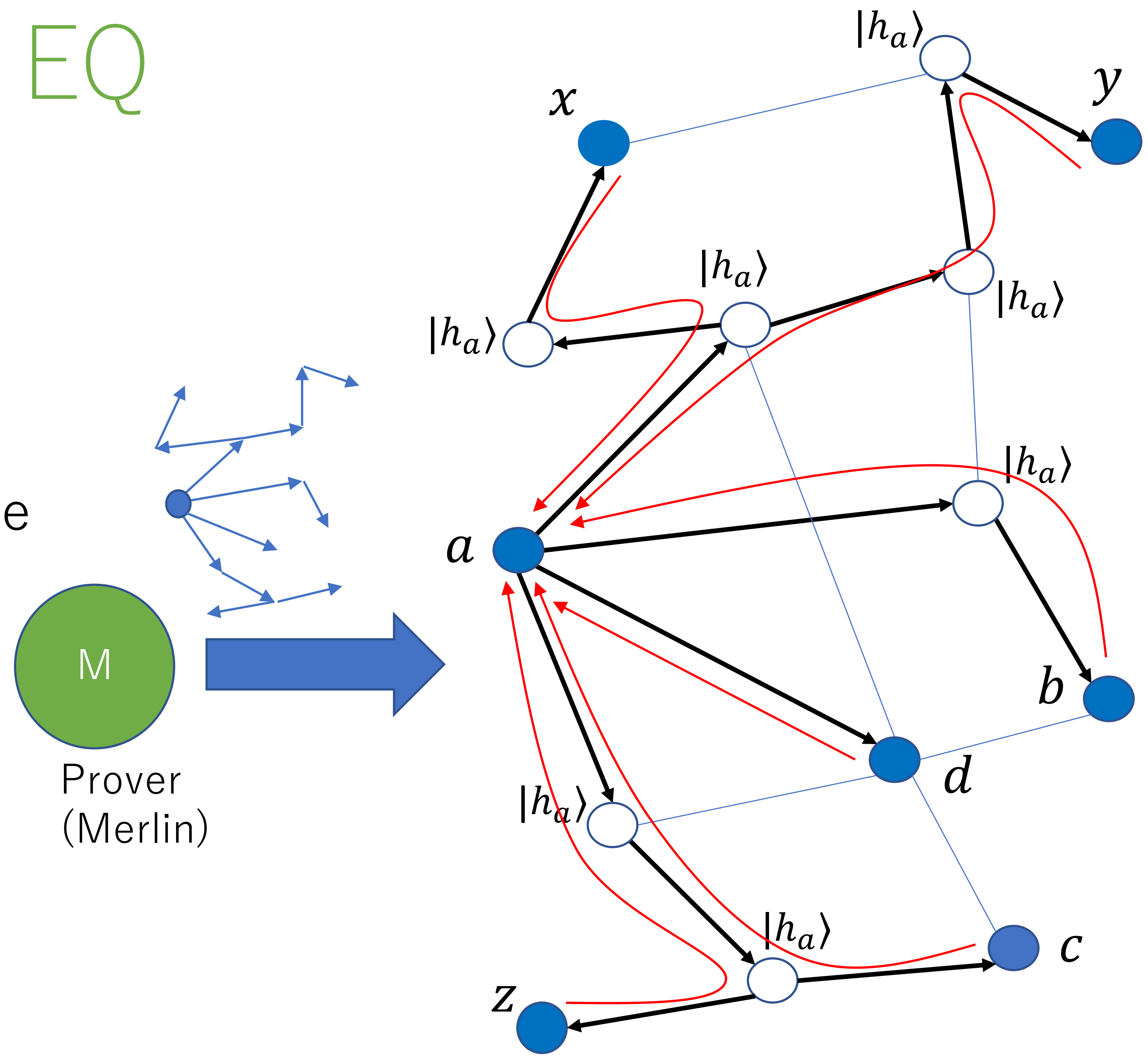
KLNP20 Protocol for a line (Verification phase)

1. Each node v_j (except right node) chooses $b_j \in \{0,1\}$ uniformly at random: if $b_j = 0$, v_j sends the state to the right neighbor; otherwise, keep it by itself.
2. Each node (except left node) does SWAP test if it has two states, and outputs its result (accept/reject), and accepts otherwise



General Graphs for EQ

- Merlin sends a rooted tree with quantum certificates:
  - Root is a terminal
  - Leaves are the other terminals
- Run the protocols on lines from the root to terminals in parallel



More Problems on a line graph

- EQ
- SetEQ
- State generation

SetEQ (2-parties P_1 & P_2)

- Input
  - Each party P_j has two lists of l elements in a finite set U
    - $a_j = (a_{j,1}, a_{j,2}, \dots, a_{j,l})$
    - $b_j = (b_{j,1}, b_{j,2}, \dots, b_{j,l})$
- Output
  - 1 (yes) iff $A \coloneqq \{a_{j,i} | j \in \{1,2\}, i \in [l]\}$ and $B \coloneqq \{b_{j,i} | j \in \{1,2\}, i \in [l]\}$ are the same as multisets

SetEQ (2-parties P_1 & P_2)

- Input
  - Each party P_j has two lists of l elements in a finite set U
    - $a_j = (a_{j,1}, a_{j,2}, \dots, a_{j,l})$
    - $b_j = (b_{j,1}, b_{j,2}, \dots, b_{j,l})$
- Output
  - 1 (yes) iff $A \coloneqq \{a_{j,i} | j \in \{1,2\}, i \in [l]\}$ and $B \coloneqq \{b_{j,i} | j \in \{1,2\}, i \in [l]\}$ are the same as multisets

Example

$a_1 = (1,2,2,4,5)$
$b_1 = (4,1,3,1,1)$

$a_2 = (5,3,1,1,4)$
$b_2 = (4,2,2,5,5)$

SetEQ (distributed comp. version) [NPY20]

SetEQ$_{l,U}$

- Input
  - Graph $G = (V, E)$
  - Each node u has two lists of l elements in a finite set U
    - $a_u = (a_{u,1}, a_{u,2}, \ldots, a_{u,l})$
    - $b_u = (b_{u,1}, b_{u,2}, \ldots, b_{u,l})$
- Output
  - 1 (yes) iff $A \coloneqq \{a_{u,i} | u \in V, i \in [l]\}$ and $B \coloneqq \{b_{u,i} | u \in V, i \in [l]\}$ are the same as multisets

Result on SetEQ

[LMN22-1, Thm2] For any small enough $\varepsilon > \mathbf{0}$, there is a dQMA protocol for $\text{SetEQ}_{l,U}$ on the line of length r with completeness $1 - \varepsilon$ and soundness ε that has

- certificate size $O(r^5 \log^2(lr) \log^2|U|)$
- message size $O(r^2 \log(lr) \log|U|)$

Result on SetEQ

[LMN22-1, Thm2] For any small enough $\varepsilon > 0$, there is a dQMA protocol for $\mathrm{SetEQ}_{l,U}$ on the line of length r with completeness $1-\varepsilon$ and soundness ε that has

- certificate size $O(r^5 \log^2(lr) \log^2|U|)$
- message size $O(r^2 \log(lr) \log|U|)$

Cf. dMA protocol

[LMN22-1, Thm3] For any dQMA protocol for $\mathrm{SetEQ}_{l,U}$ on a line graph of length r with certificate size s_c, completeness $\frac{3}{4}$ and soundness $\frac{1}{4}$,

If $|U| < l$, then $s_c = \Omega(|U| \log(l/|U|))$;

If $|U| = \Omega(l)$, then $s_c = \Omega(l)$;

If $|U| = \Omega(rl)$, then $s_c = \Omega(rl)$

More Problems on a line graph

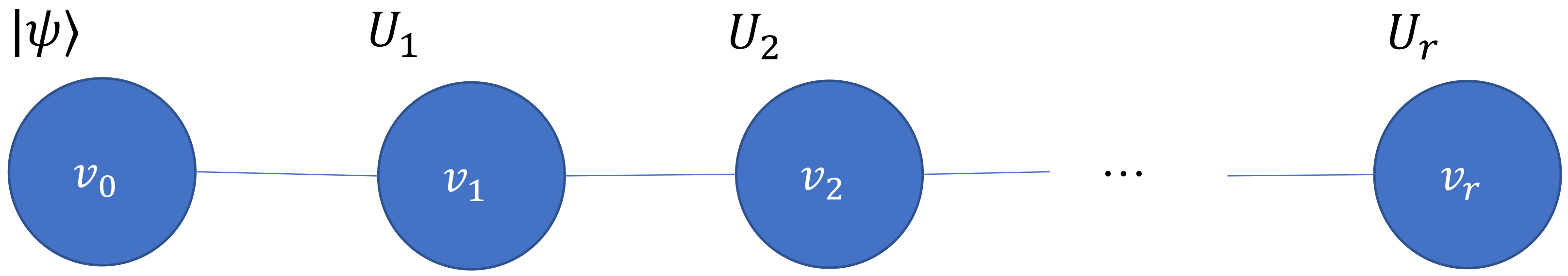
- EQ
- SetEQ
- State generation (SGDI)

Classical problems ⇒ Quantum problems

- State & Unitary synthesis [Aaronson 16]
  - State ≒ Quantum version of bit strings
  - Unitary ≒ Quantum version of Boolean circuits
- Interactive proof for State & Unitary synthesis [RY21]
- Complexity of generating a QMA certificate (search-to-decision reduction of QMA) [INNRY22]
- Pseudorandom states [JLS18,Kre21]

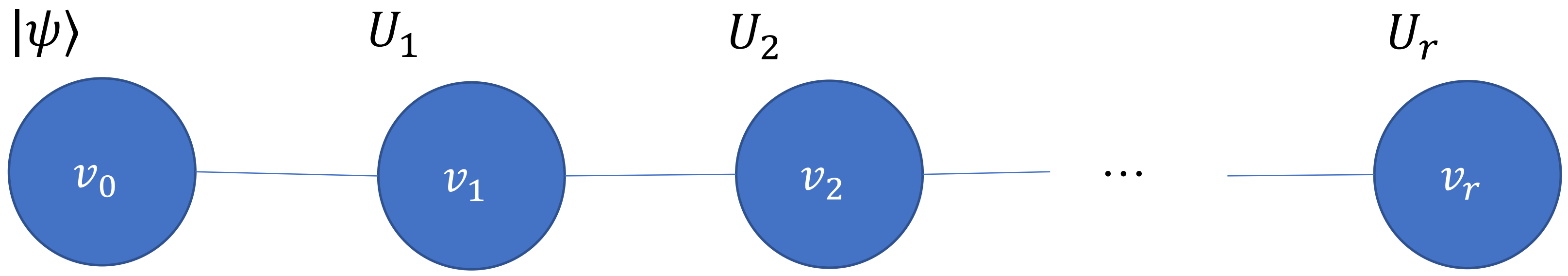
SGDI: State generation on distributed inputs

- Line $v_0 - v_1 - \cdots - v_r$
- v_0 has a classical description of an n-qubit state $|\psi\rangle$
- v_j $(j = 1,2,\ldots,r)$ has a unitary transform U_j
- Goal: Generate $|\varphi_r\rangle \coloneqq U_r \cdots U_1 |\psi\rangle$ at v_r



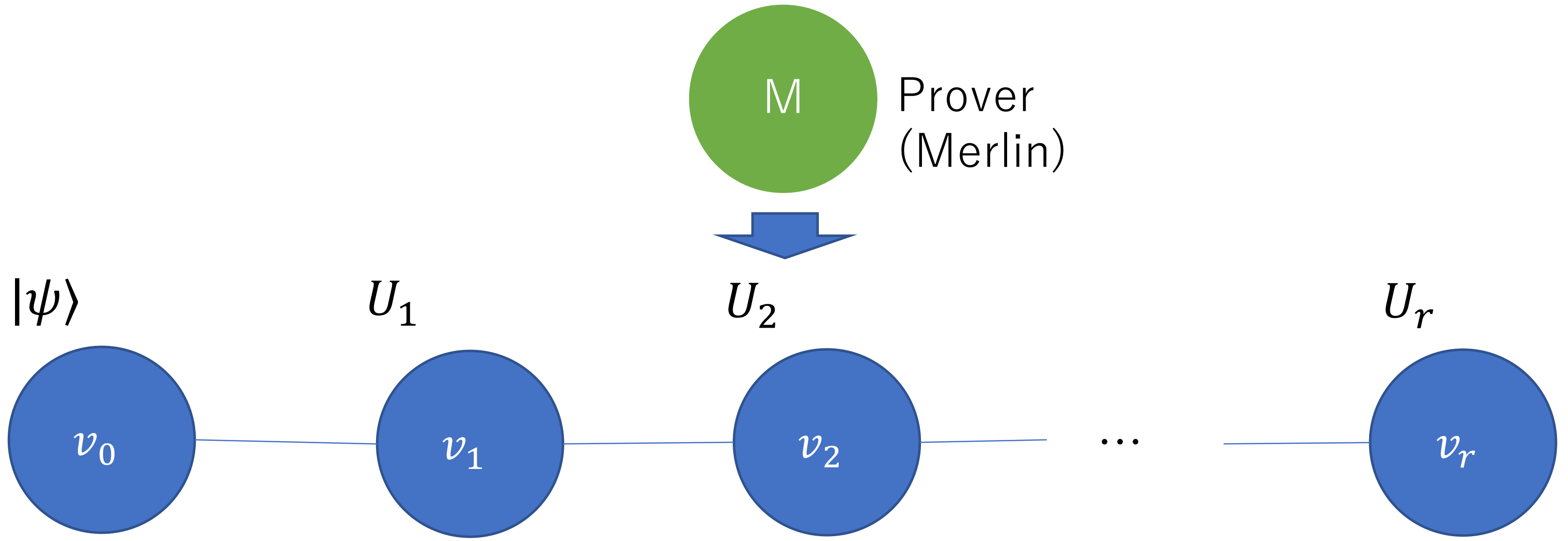
SGDI: State generation on distributed inputs

- Line $v_0 - v_1 - \cdots - v_r$
- v_0 has a classical description of an n-qubit state $|\psi\rangle$
- v_j $(j = 1,2, \dots, r)$ has a unitary transform U_j
- Goal: Generate $|\varphi_r\rangle := U_r \cdots U_1 |\psi\rangle$ at v_r
- Impossible by 1-round



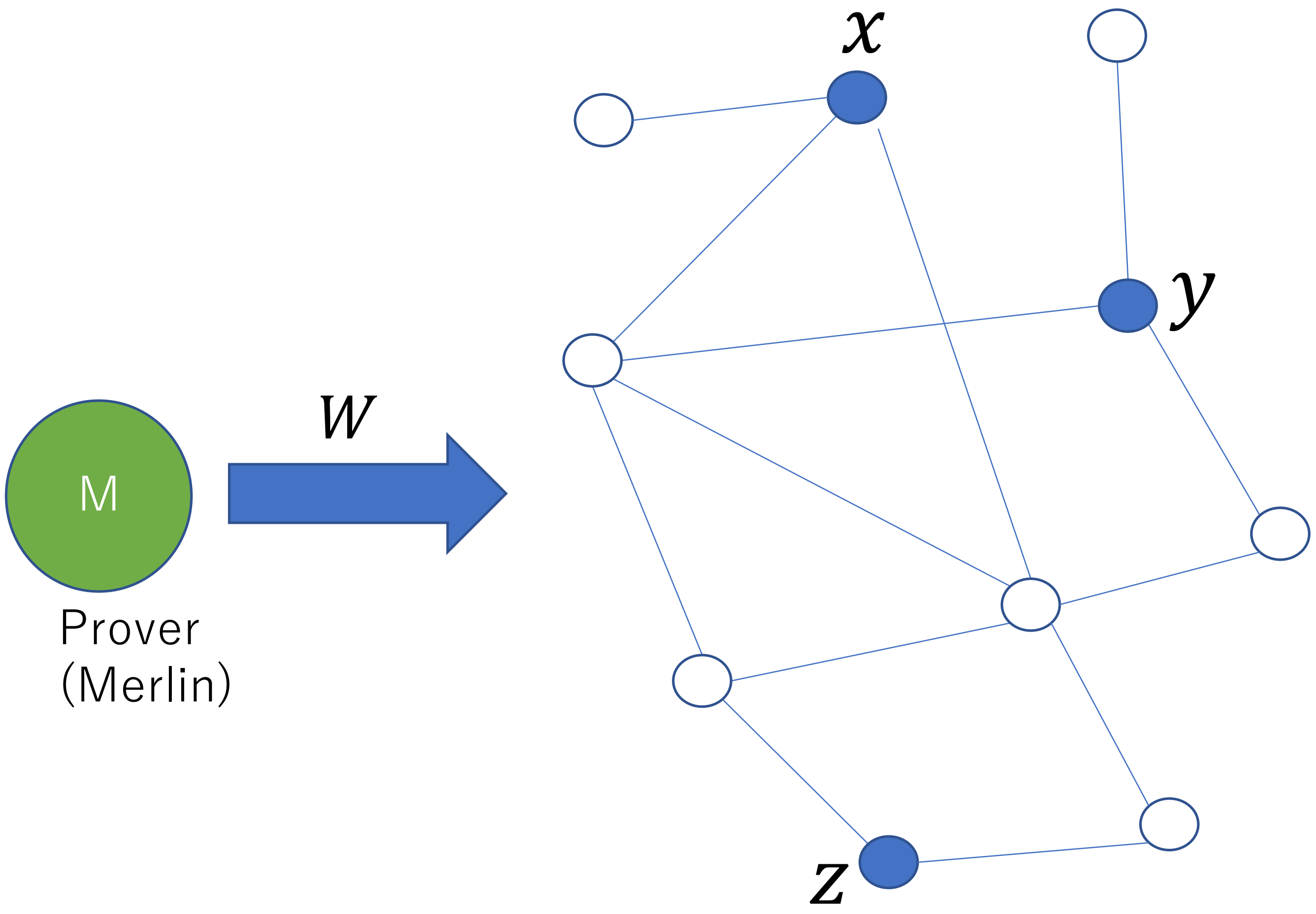
Verifying SGDI

- Line $v_0 - v_1 - \cdots - v_r$
- v_0 has a classical description of an n-qubit state $|\psi\rangle$
- v_j $(j = 1,2, \dots, r)$ has a unitary transform U_j
- Goal: Verify $|\varphi_r\rangle \coloneqq U_r \cdots U_1 |\psi\rangle$ at v_r with the help of the prover



Properties of Distributed Certification

(YES case: Completeness)
$\exists w$[all nodes accept]
(w.h.p.)
(NO case: Soundness)
$\forall w$[some node rejects]
(w.h.p.)



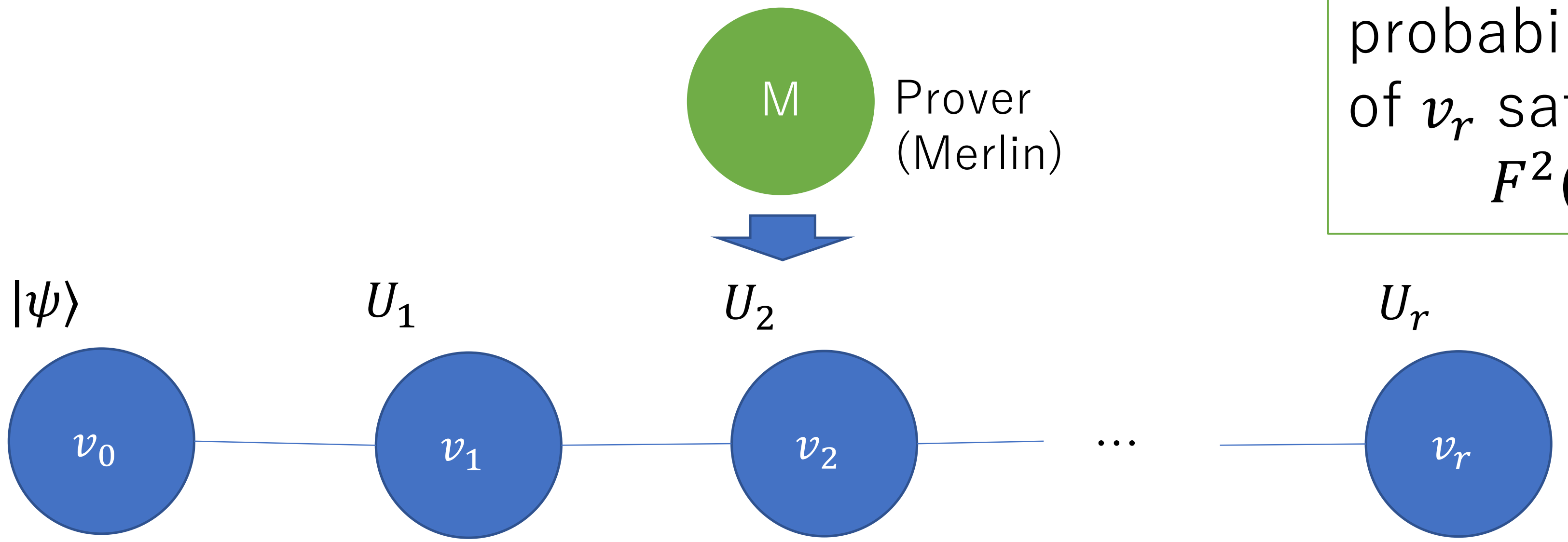
Verifying SGDI

- Line $v_0 - v_1 - \cdots - v_r$
- v_0 has a classical description of an n-qubit state $|\psi\rangle$
- v_j $(j = 1,2, \dots, r)$ has a unitary transform U_j
- Goal: Verify $|\varphi_r\rangle \coloneqq U_r \cdots U_1 |\psi\rangle$ at v_r with the help of the prover

(Completeness)
$\exists |W\rangle$[all nodes accept & v_r outputs $|\varphi_r\rangle$]

(Soundness)
If all nodes accept with probability $\geq \varepsilon$, the output of v_r satisfies

$$F^2(\rho, |\varphi_r\rangle) \geq 1 - \varepsilon$$



Result on SGDI

- Line $v_0 - v_1 - \cdots - v_r$
- v_0 has a classical description of an n-qubit state $|\psi\rangle$
- v_j $(j = 1,2,\ldots,r)$ has a unitary transform U_j
- Goal: Verify $|\varphi_r\rangle \coloneqq U_r \cdots U_1 |\psi\rangle$ at v_r with the prover

(Completeness)
$\exists |W\rangle$[all nodes accept & v_r outputs $|\varphi_r\rangle$]
(Soundness)
If all nodes accept with probability $\geq \varepsilon$, the output of v_r satisfies

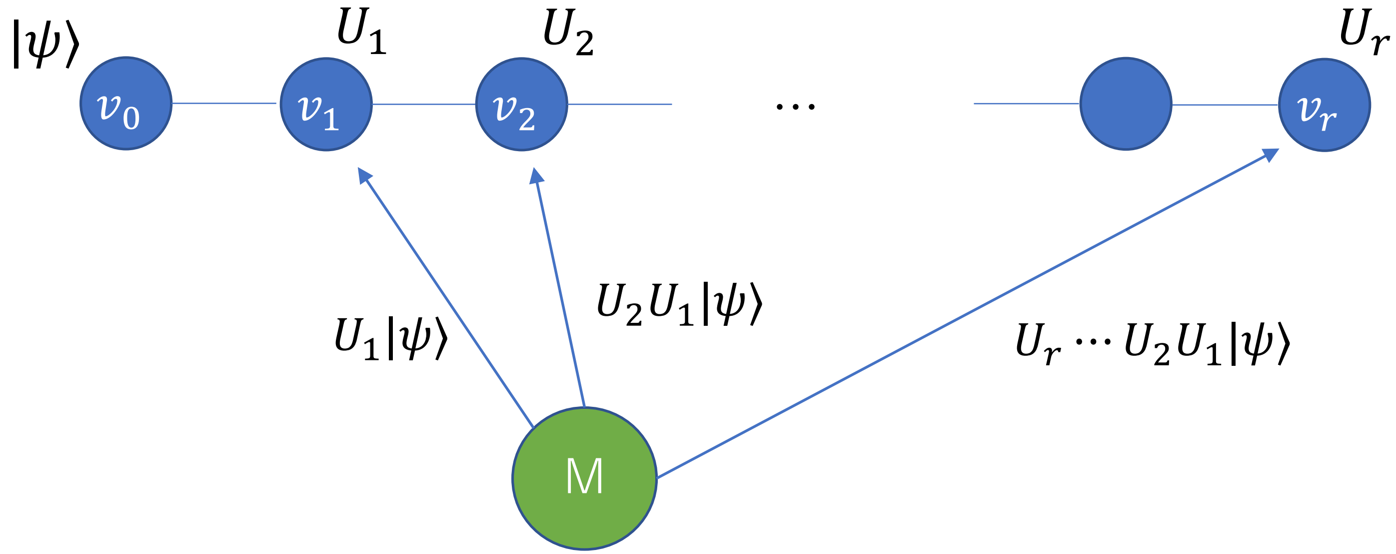
$$F^2(\rho, |\varphi_r\rangle) \geq 1 - \varepsilon$$

[LMN22-1:Thm1]
For any constant $\varepsilon > 0$, there is a dQMA protocol for SGDI with
- certificate size $O(n^2 r^5)$
- Message size $O(nr^2)$

Proof idea of Thm1

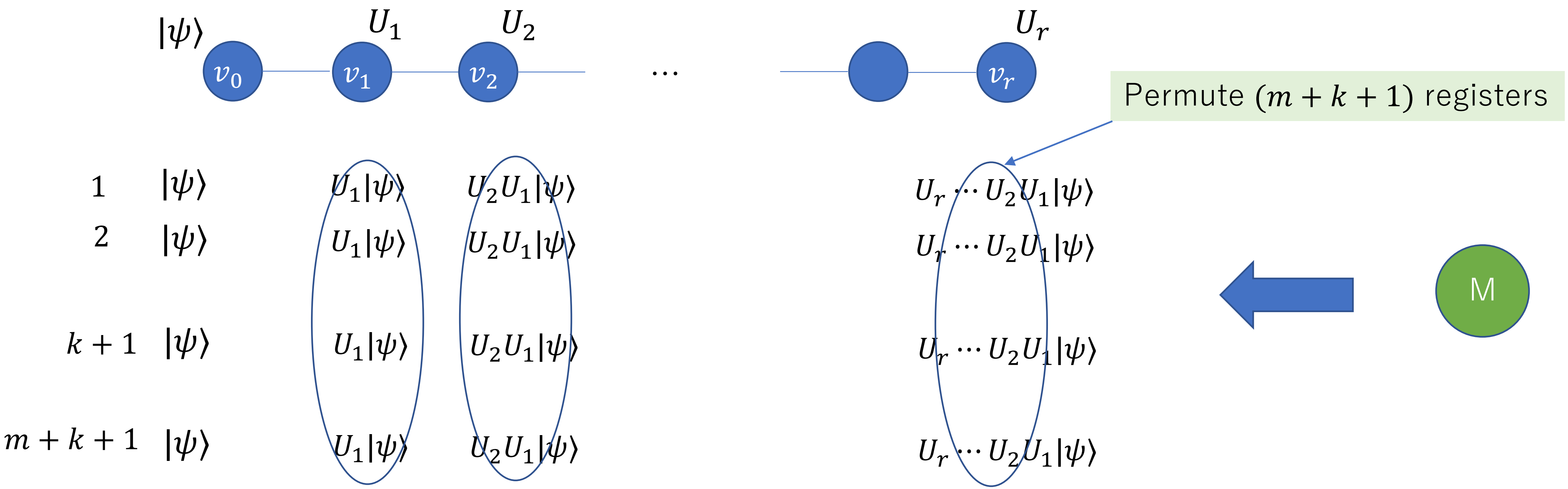
- Incorporate FLNP20 protocol into the idea by Morimae-Takeuchi-Hayashi [MTH17] for the verification of graph states (one-way LOCC de Finnetti by Li-Smith [LS15])

FLNP-like test



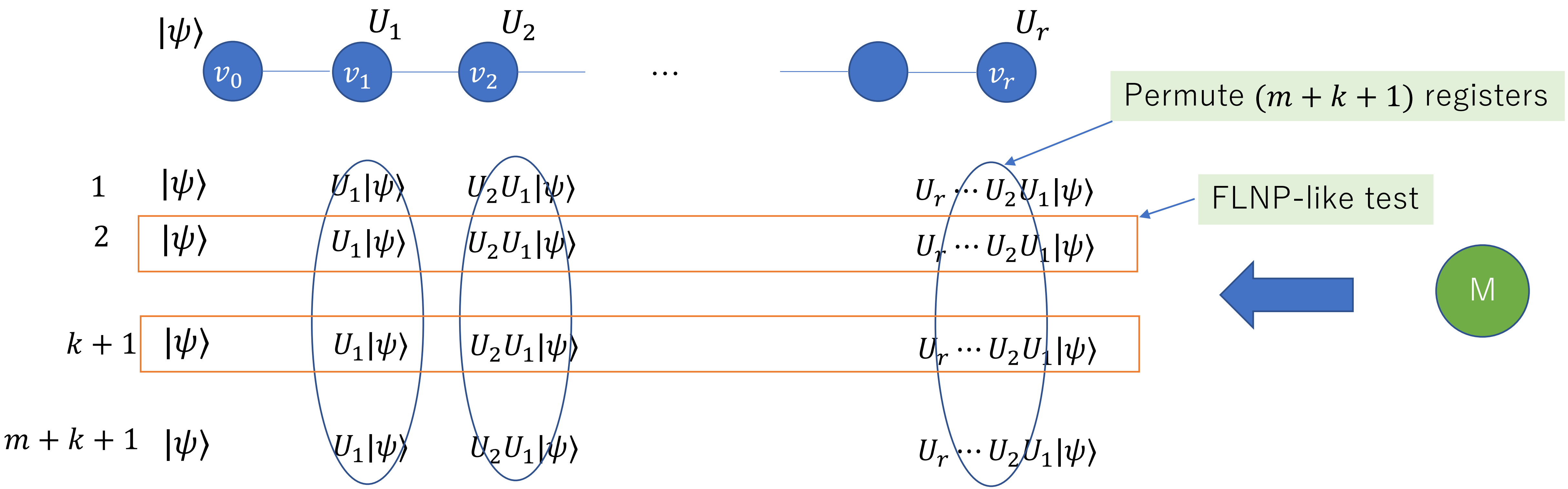
Proof idea of Thm1

- Incorporate FLNP20 protocol into the idea by Morimae-Takeuchi-Hayashi [MTH17] for the verification of graph states (one-way LOCC de Finnetti by Li-Smith [LS15])



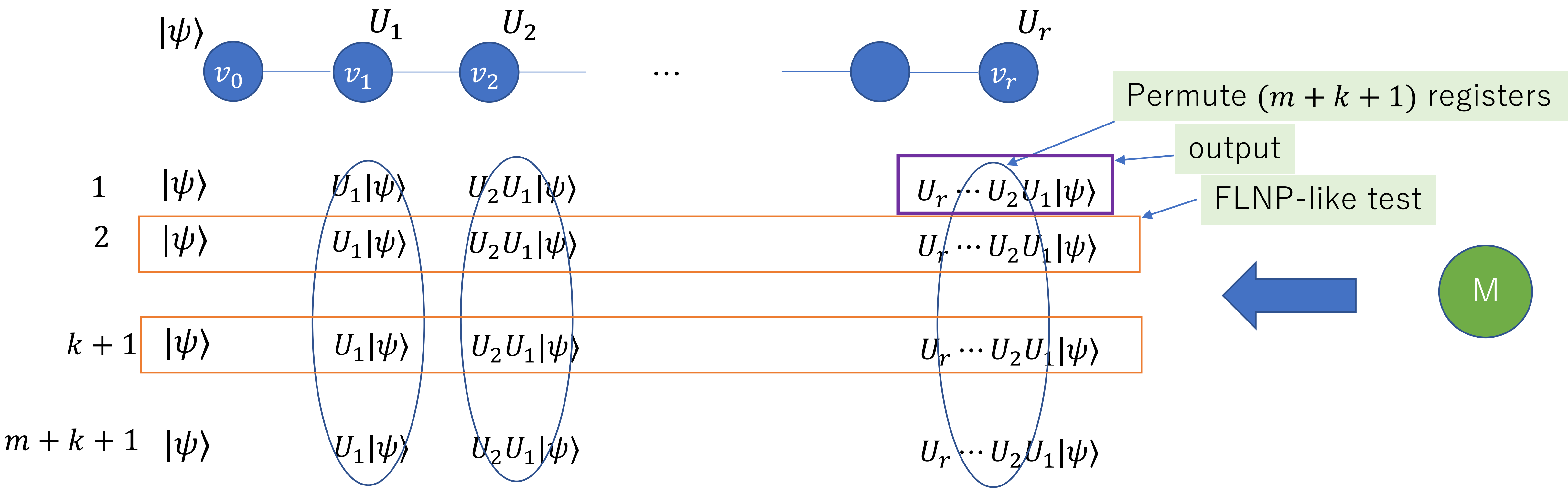
Proof idea of Thm1

- Incorporate FLNP20 protocol into the idea by Morimae-Takeuchi-Hayashi [MTH17] for the verification of graph states (one-way LOCC de Finnetti by Li-Smith [LS15])



Proof idea of Thm1

- Incorpolate FLNP20 protocol into the idea by Morimae-Takeuchi-Hayashi [MTH17] for the verification of graph states (one-way LOCC de Finnetti by Li-Smith [LS15])

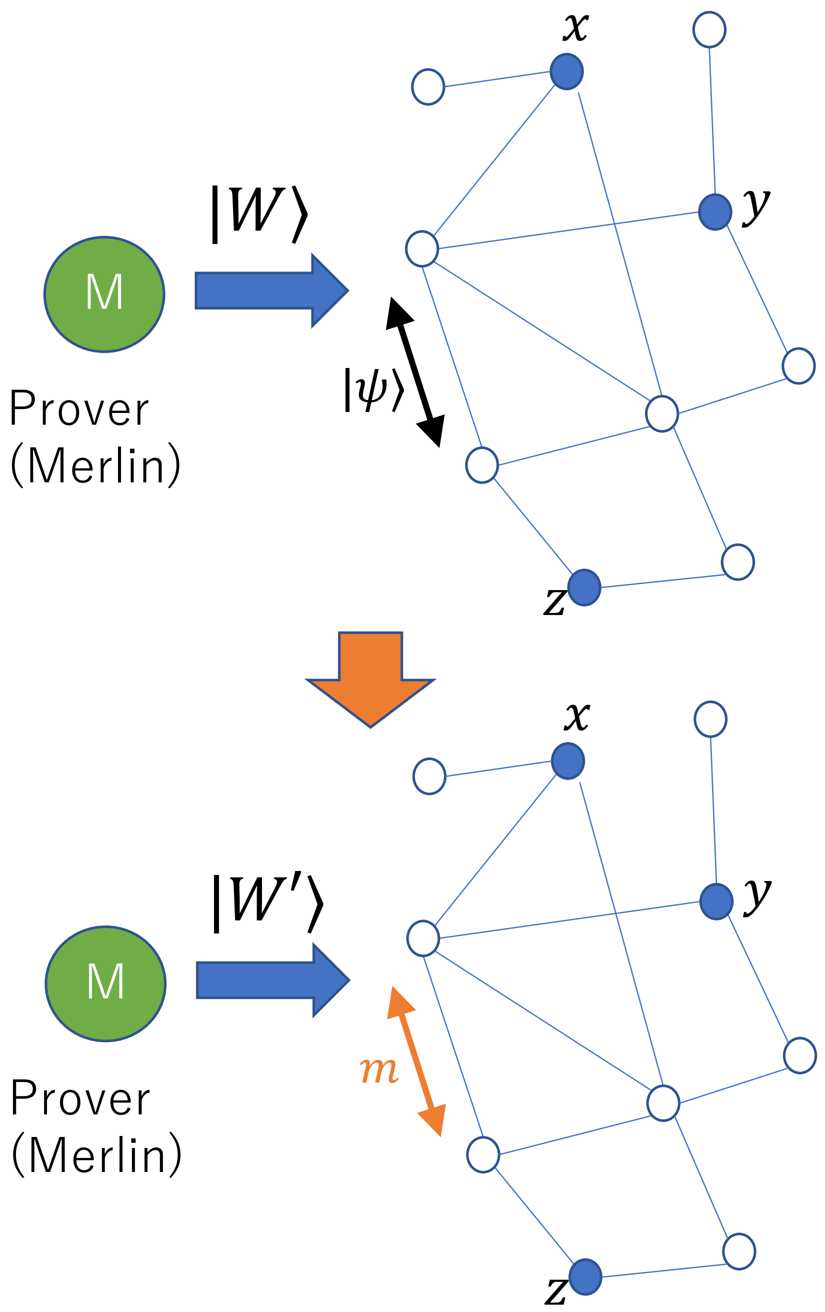


Another Improvement

- dQMA protocols have two phases:
  - Prover phase
  - Verification phase

Q. Can we replace the quantum communication of the verification phase into classical communication?

  - Verification by local operation and classical communication (**LOCC dQMA protocol**)



Another Improvement

[LMN22-1, Thm5] For any constant p_c and p_s such that $0 \leq p_s < p_c \leq 1$, let P be a dQMA protocol for some problem on a network G with completeness p_c and soundness p_s, certificate size s_c^P and message size s_m^P. For any small enough constant $\gamma > 0$, there is an LOCC dQMA protocol P' for the same problem on G with completeness p_c, soundness $p_s + \gamma$, certificate size $s_c^P + O(d_{max} s_m^P s_{tm}^P)$, where d_{max} is the maximum degree of G, and s_{tm}^P is the total number of qubits sent in the verification stage of P.

Another Improvement

[LMN22-1, Thm5] For any constant p_c and p_s such that $0 \leq p_s < p_c \leq 1$, let P be a dQMA protocol for some problem on a network G with completeness p_c and soundness p_s, certificate size s_c^P and message size s_m^P. For any small enough constant $\gamma > 0$, there is an LOCC dQMA protocol P' for the same problem on G with completeness p_c, soundness $p_s + \gamma$, certificate size $s_c^P + O(d_{max} s_m^P s_{tm}^P)$, where d_{max} is the maximum degree of G, and s_{tm}^P is the total number of qubits sent in the verification stage of P.

[LMN22-1, Cor1] For any small enough constant $\varepsilon > 0$, there is an LOCC dQMA protocol for EQ_n^t with completeness 1, soundness ε, certificate size $O(d_{max}|V|t^2r^4 \log^2(n+r))$ and message size $O(|V|t^2r^4 \log^2(n+r))$, where r is the radius of the set of the t terminals and $|V|$ is the number of nodes of the network $G = (V, E)$.

Cf. ∃ dQMA protocol for EQ_n^t with $O(tr^2 \log(n+r))$-qubit certificates & messages

- Still exponentially better in the length of data n

Another Improvement

[LMN22-1, Thm5] For any constant p_c and p_s such that $0 \le p_s < p_c \le 1$, let P be a dQMA protocol for some problem on a network G with completeness p_c and soundness p_s, certificate size s_c^P and message size s_m^P. For any small enough constant $\gamma > 0$, there is an LOCC dQMA protocol P' for the same problem on G with completeness p_c, soundness $p_s + \gamma$, certificate size $s_c^P + O(d_{max} s_m^P s_{tm}^P)$, where d_{max} is the maximum degree of G, and s_{tm}^P is the total number of qubits sent in the verification stage of P.

Proof idea:

- Replace quantum communication in the verification phase into classical communication by sharing EPR pairs sent from the prover with the original witness
- Use Zhu-Hayashi result [ZH19] for verification of a EPR pair in adversarial scenario.

Summary

- Quantum protocols for distributed certification
  - EQ
  - SetEQ
  - SGDI（State generation for distributed inputs）
- Conversion of dQMA protocols to LOCC dQMA ones
- Future work
  - Extend SetEQ and SGDI to general graphs
  - Quantum advantage on graph size
  - Non-trivial lower bound of quantum proof lengths for any problem

[FLNP20] Fraigniaud, Le Gall, N, Paz, arXiv: 2002.10018
[LMN22-1] Le Gall, Miyamoto, N, arXiv: 2210.01389
[LMN22-2] Le Gall, Miyamoto, N, arXiv: 2210.01390