Multiparty Quantum Simultaneous Message Passing Communication Complexity

Harumichi Nishimura (Nagoya U.)

Based on arXiv:2412.08091, joint work with Francois Le Gall (Nagoya U.), Oran Nadler (Tel Aviv U.), Rotem Oshman (Tel Aviv U.)

Shenzhen-Nagoya Workshop on Quantum Science 2025

September 26, 2025

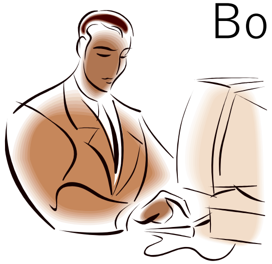
Communication Complexity

- Introduced by Yao in 1979 [Yao79]
- Multiple parties with separate inputs want to compute some function with small amount of communication

Alice

communication

Bob



Input: $x \in X$

Output $f(x, y) \in Z$

Input $y \in Y$

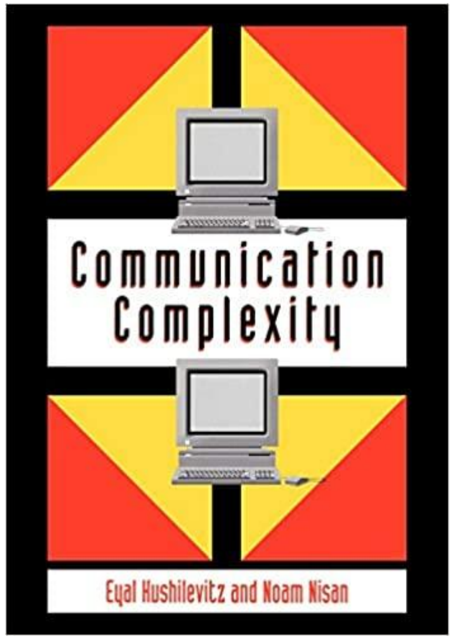
Output $f(x, y)$

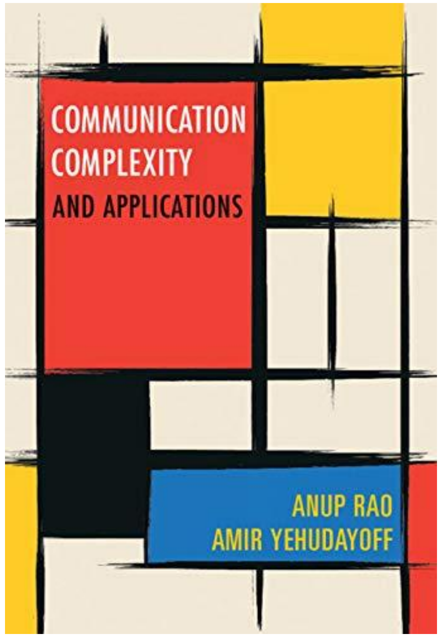
[Yao79] A. Yao, STOC1979, pp.209-213

Communication Complexity

- Many applications to computational complexity lower bounds
  - VLSI
  - Decision trees & Data structures
  - Boolean circuits
  - Time-space tradeoff
  - Streaming algorithms
  - Proof complexity
  - Distributed computing

  etc

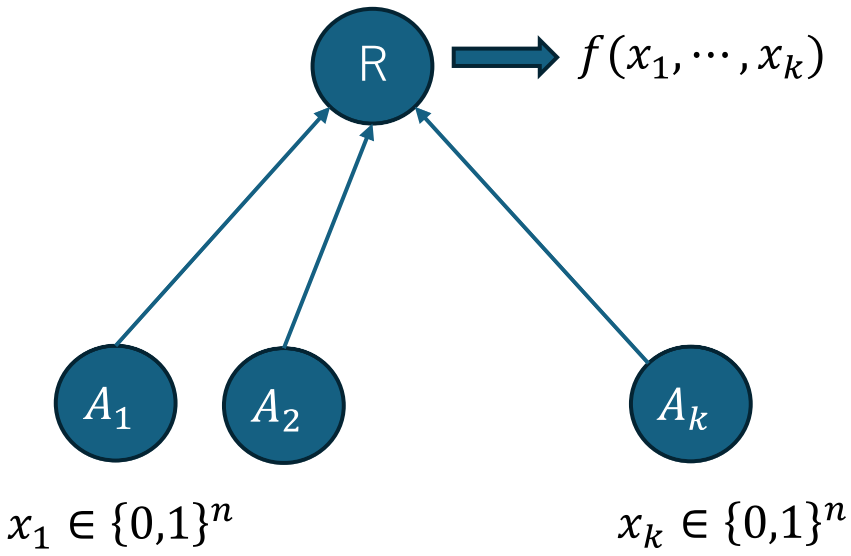




[KN97] E. Kushilevitz, N. Nisan, Communication Complexity, Cambridge University Press, 1997
[RY20] A. Rao, A. Yehudayoff, Communication Complexity and Applications, Cambridge University Press, 2020

Simultaneous Message Passing (SMP)

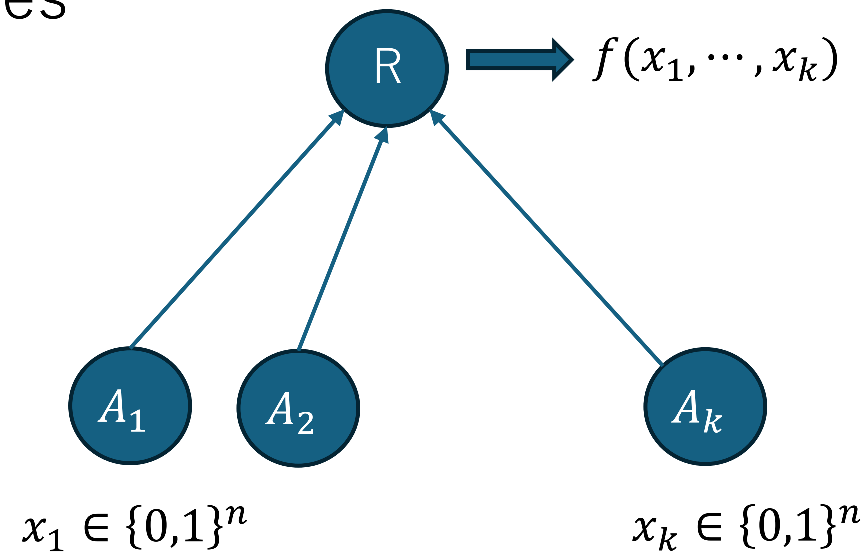
- Weakest model in communication complexity [Yao79]
  - Each of k parties A_ℓ has input $x_\ell \in \{0,1\}^n$
  - A_ℓ sends a message to the referee
  - The referee computes a function value $f(x_1, \cdots, x_k)$
  - Complexity (cost):=the total length of the messages



[Yao79] A. Yao, STOC1979, pp.209-213

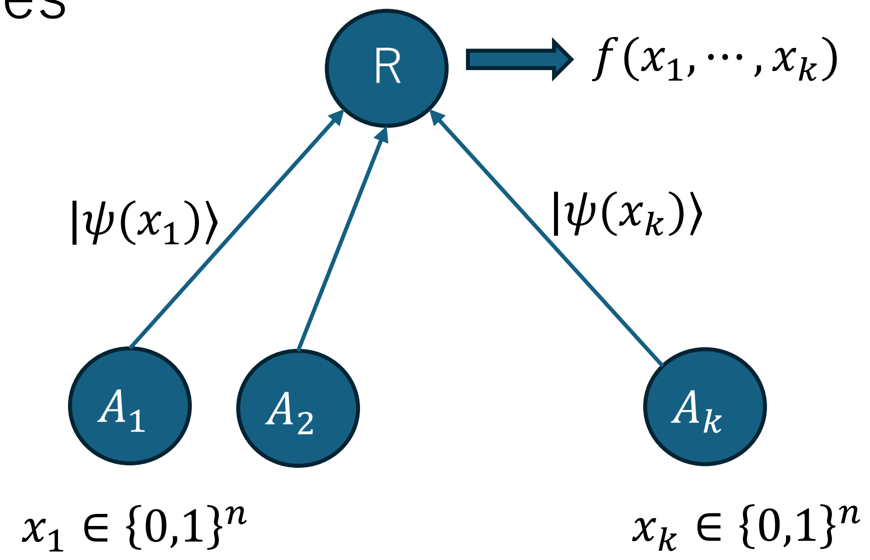
Simultaneous Message Passing (SMP)

- Weakest model in communication complexity [Yao79]
  - Each of k parties A_ℓ has input $x_\ell \in \{0,1\}^n$
  - A_ℓ sends a message to the referee
  - The referee computes a function value $f(x_1, \cdots, x_k)$
  - Complexity:=the total length of the messages
- Computation modes
  - Deterministic
  - Randomized
    - **Public coin**: all parties A_ℓ share randomness
    - **Private coin**: no shared randomness



Simultaneous Message Passing (SMP)

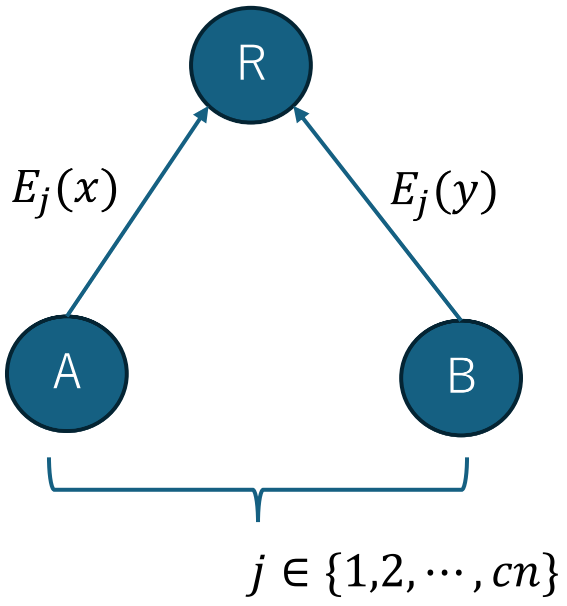
- Weakest model in communication complexity [Yao79]
  - Each of k parties A_ℓ has input $x_\ell \in \{0,1\}^n$
  - A_ℓ sends a message to the referee
  - The referee computes a function value $f(x_1, \cdots, x_k)$
  - Complexity:=the total length of the messages
- Computation modes
  - Deterministic
  - Randomized
  - Quantum [Yao93]
    - Shared randomness or entanglement
    - **No shared resources (private)**



[Yao93] A. Yao, STOC1993, pp.352-361

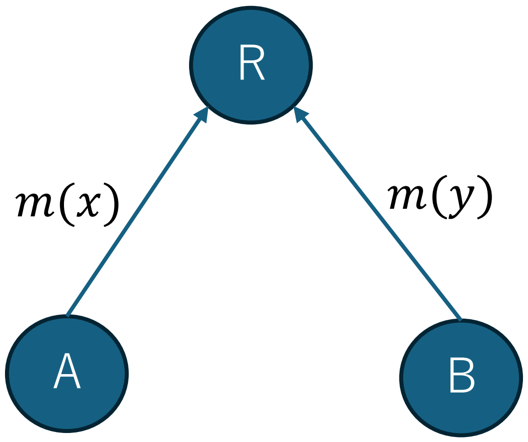
2-party SMP

- 2-party case is well-studied
  - Equality
    - Whether Alice's input $x \in \{0,1\}^n$ is the same as Bob's input $y \in \{0,1\}^n$
  - Classical
    - **Public coin** $O(1)$
    - Use a good code $E(x): \{0,1\}^n \to \{0,1\}^{cn}$
      - $\frac{\#\{j \in \{1,2,\cdots,cn\}: E(x)_j \neq E(y)_j\}}{cn} \geq \frac{9}{10}$ if $x \neq y$
    - Shared randomness $j \in \{1,2,\cdots,cn\}$
    - Alice sends $E_j(x)$ & Bob sends $E_j(y)$ to the referee
    - The referee outputs 1 if and only if $E_j(x) = E_j(y)$
    - $E_j(x) :=$ the jth bit of $E(x)$



2-party SMP

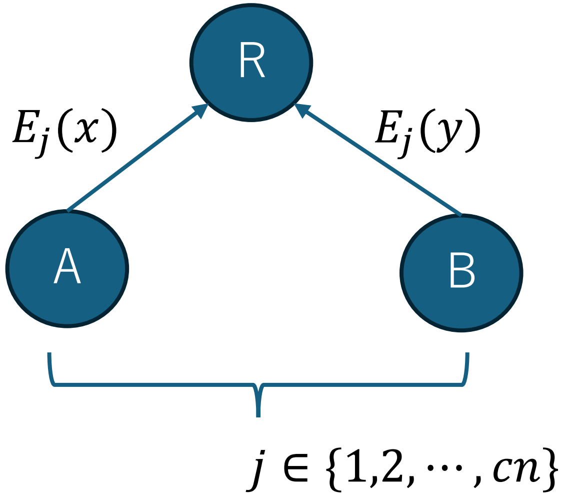
- 2-party case is well-studied
  - Equality
    - Whether Alice's input $x \in \{0,1\}^n$ is the same as Bob's input $y \in \{0,1\}^n$
  - Classical
    - Public coin $O(1)$
    - **Private coin** $\Omega(\sqrt{n})$ [NS96,BK97]

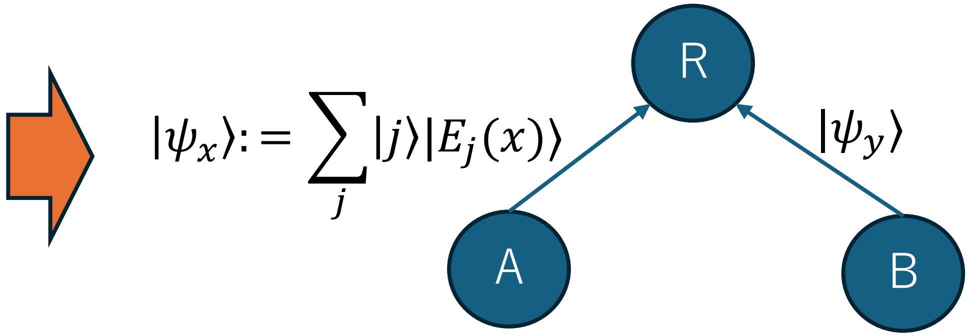


[NS96] I. Newman, M. Szegedy, STOC1996, pp.561-570
[BK97] L. Babai, P. Kimmel, CCC1997, pp.239-246

Quantum SMP

- **Exponential quantum advantage** for Equality [BCWW01]
  - Quantum $O(\log n)$
  - Classical $\Omega(\sqrt{n})$ [NS96,BK97]
  - Use of "quantum" fingerprints $\{|\psi_x\rangle\}_{x\in\{0,1\}^n}$
    - $|\psi_x\rangle$ is short (consists of $O(\log n)$ qubits) but available for checking whether $x = y$
    - Convert shared randomness into quantum fingerprints





[BCWW01] H. Buhrman, R. Cleve, J. Watrous, R. de Wolf, Physical Review Letters, 87, 167902 (2001)

Quantum SMP

- 2-party case is well-studied
  - Exponential quantum advantage for Equality [BCWW01]
    - Quantum $O(\log n)$
    - Classical $\Omega(\sqrt{n})$ [NS96,BK97]
    - Use of "quantum" fingerprints $\{|\psi_x\rangle\}_{x\in\{0,1\}^n}$
      - $|\psi_x\rangle$ is short (consists of $O(\log n)$ qubits) but available for checking whether $x = y$
      - Convert shared randomness into quantum fingerprints
  - More results
    - **Hamming distance** [Yao03]
      - $\mathrm{Ham}_d(x, y) = \begin{cases} 1 & \text{(Hamming distance } \Delta(x, y) \text{ is at most } d) \\ 0 & \text{(otherwise)} \end{cases}$
      - d is a constant
  - **Tomorrow's talk by Hasegawa-san**

[Yao03] A. Yao, STOC2003, pp.77-81

Quantum Multiparty SMP

- Multiparty case is not explored
  - Negative result [GIW13]
  - Positive results [This talk]

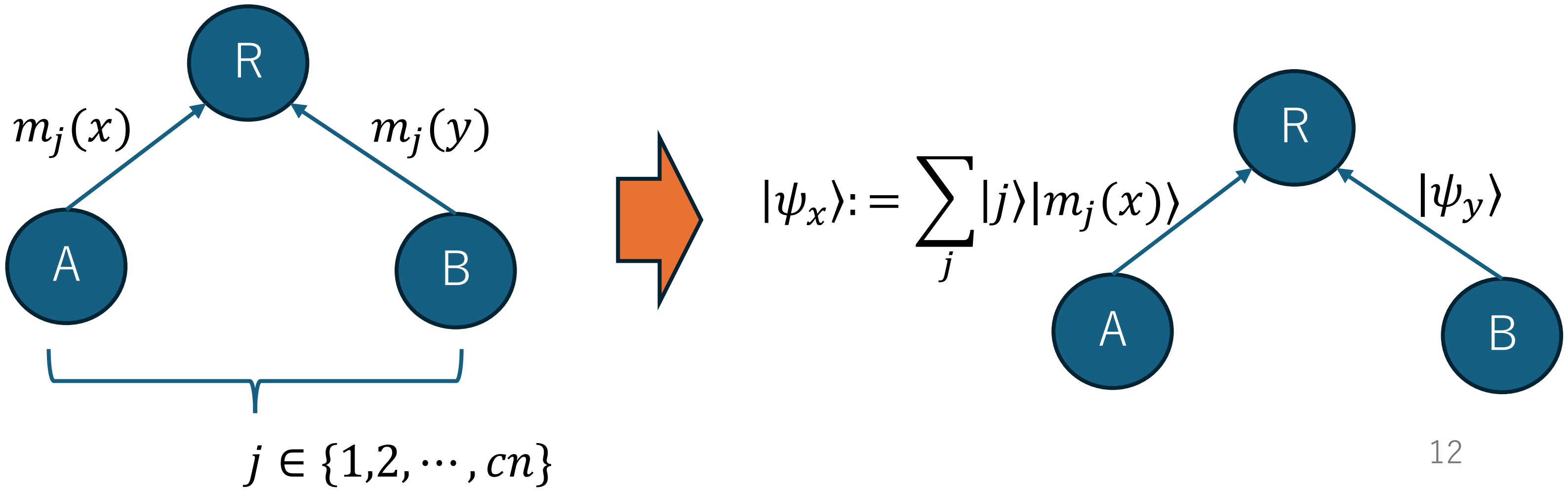
[GIW13] D. Gavinsky, T. Ito, G. Wang, CCC2013, pp.34-43

Public-coin SMP vs QSMP

Q: Is QSMP efficient (logarithmic order of input length) when public-coin (classical) SMP is efficient?

- EQ has an efficient public-coin SMP ⇒ QSMP is also efficient [BCWW01]

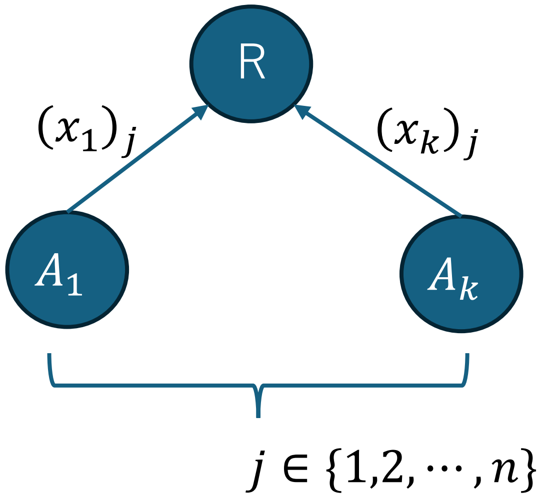
- 2-party case: **YES**
  - Public-coin SMP complexity is $O(1)$ ⇒ QSMP is efficient [Yao03]



Public-coin SMP vs QSMP

Q: Is QSMP efficient when public-coin (classical) SMP is efficient?

- Multiparty case: **NO**
  - Public-coin SMP complexity is $O(1)$ but QSMP is not efficient [GIW13]
  - Gap-Parity
    - $GP_k(x_1, \cdots, x_k) \coloneqq \begin{cases} 1 & (\text{Hamming weight of } x_1 \oplus \cdots \oplus x_k \geq 2n/3) \\ 0 & (\text{Hamming weight of } x_1 \oplus \cdots \oplus x_k \leq n/3) \end{cases}$
    - Public coin (Classical): $O(1)$
    - Quantum: $\Omega(kn^{1-\frac{2}{k}})$



[GIW13] D. Gavinsky, T. Ito, G. Wang, CCC2013, pp.34-43

Public-coin SMP vs QSMP

Q: Is QSMP efficient when public-coin (classical) SMP is efficient?

- Multiparty case: **NO**
  - Public-coin SMP complexity is $O(1)$ but QSMP is not efficient [GIW13]

Q: For which problems efficient multiparty QSMPs can be constructed from public-coin SMPs?

Our Results

- Efficient multiparty QSMP protocols for:
  - Equality functions
  - Frequency moments based on equality
  - Neighborhood diversity
  - Reconstruction of
    - **P3/P4-induced subgraph free graphs [KMRS15]**
    - **Distance-hereditary graphs [MPRT20]**
  - Enumeration of isolated cliques

[KMRS15] J. Kari, M. Matamala, I. Rapaport, V. Salo, SIROCCO2015, pp.370-384
[MPRT20] P. Montealegre, S. Perez-Salazar, I. Rapaport, I. Todinca, Journal of Computer and System Sciences 113, pp.1-17 (2020)

Our Results [LNNO24]

Problem	Total complexity	Local complexity	Comments
Group-by-EQ	$k \log k \log n$	$\log k \log n$	total complexity $\Omega(k\sqrt{n})$ in classical case
Neighborhood diversity	$k(\log k)^2$	$(\log k)^2$	NIH Network model
Reconstruction of P3/P4-induced subgraph free graphs	$k(\log k)^2$	$(\log k)^2$	NIH Network model
Reconstruction of distance hereditary graphs	$k(\log k)^2$	$(\log k)^2$	NIH Network model
Enumeration of max-d-isolated cliques	$kd(\log k)^2$	$d(\log k)^2$	NIH Network model

[LNNO24] F. Le Gall, O. Nadler, H. Nishimura, R. Oshman, arXiv:2412.08091, OPODIS2024

Our Results

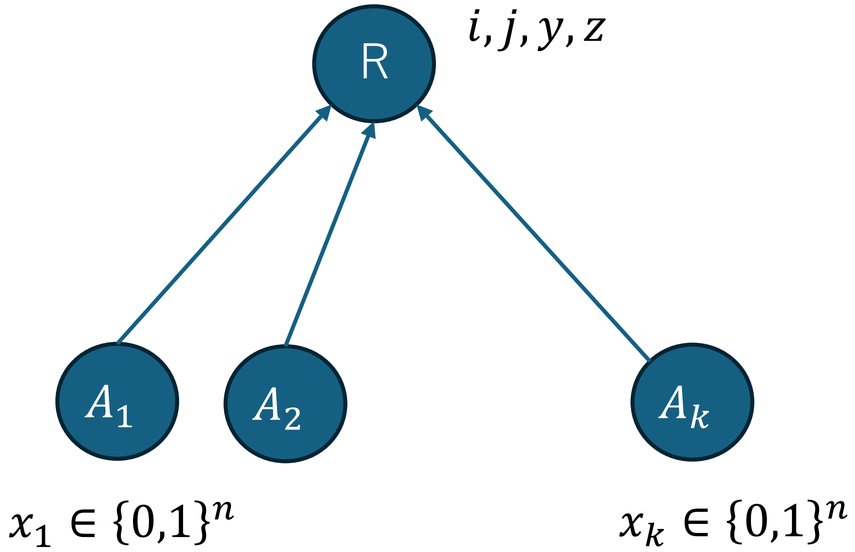
- Efficient multiparty QSMP protocols for:
    - Equality functions
    - Frequency moments based on equality
    - Neighborhood diversity
    - Reconstruction of
        - P3/P4-induced subgraph free graphs [KMRS15]
        - Distance-hereditary graphs [MPRT20]
    - Enumeration of isolated cliques
- Our only quantum technique
    - **Conversion from efficient decision trees based on “modified equality queries” to efficient multiparty QSMP protocols**

Modified Equality Queries

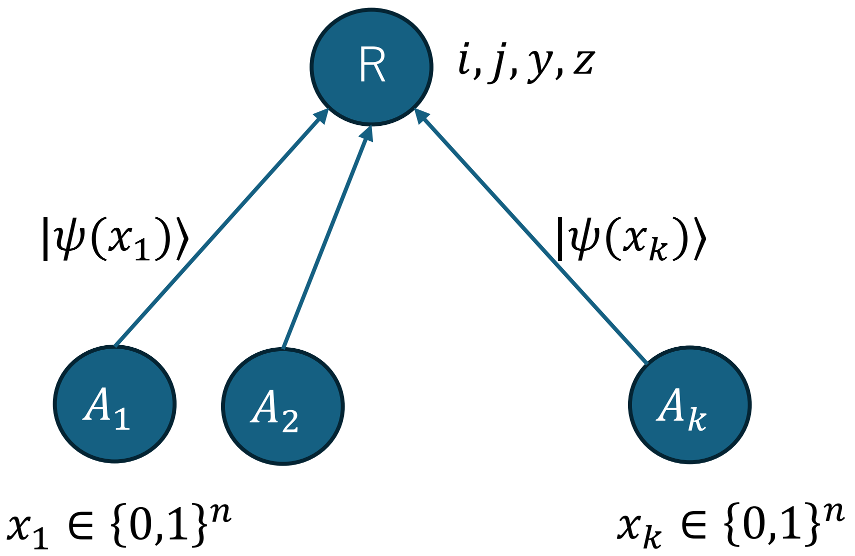
- $\mathrm{MEQ}_{k,n}(i, j, y, z)$
  - Input (Query): indices $i, j \in [k]$ & strings $y, z \in \{0,1\}^n$
  - Output (Answer): $x_i \oplus y = x_j \oplus z$?

  # $x_j \in \{0,1\}^n$ is the input of the jth player

  # each player must send the state without knowing the query (i, j, y, z)



Modified Equality Queries



- $\mathrm{MEQ}_{k,n}(i, j, y, z)$
  - Input (Query): indices $i, j \in [k]$ & strings $y, z \in \{0,1\}^n$
  - Output (Answer): $x_i \oplus y = x_j \oplus z$?

  # $x_j \in \{0,1\}^n$ is the input of the jth player

  # each player must send the state without knowing the query (i, j, y, z)

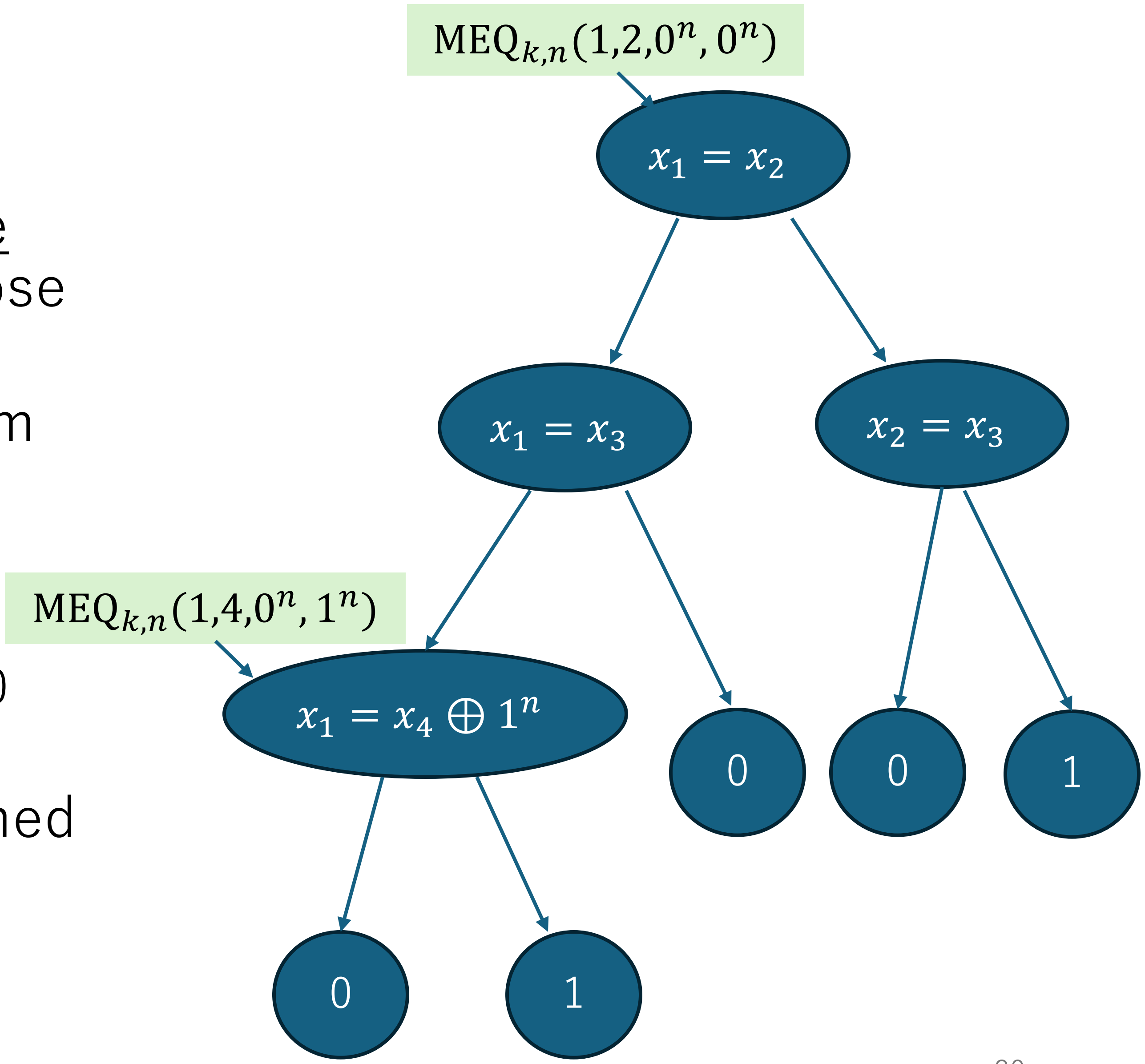
Quantum protocol for MEQ

Lemma 1: There are quantum fingerprints of $O\left(\log n \cdot \log \frac{1}{\varepsilon}\right)$ qubits $\{|\psi(x)\rangle\}_{x \in \{0,1\}^n}$ such that the ℓth player sends a state $|\psi(x_\ell)\rangle$ to the referee, who can compute $\mathrm{MEQ}_{k,n}(i, j, y, z)$, for any given i, j, y, z, with error probability ε

Proof: Quantum fingerprint based on good linear error-correcting codes can be modified from $|\psi(x_\ell)\rangle$ to $|\psi(x_\ell \oplus y)\rangle$ without knowing the original fingerprint but with knowing y

MEQ decision tree

- Rooted binary tree whose inner node are labeled by MEQ queries and whose leaves are labeled by output values
  - The tree is evaluated starting from the root
  - At each step, the query at the current node is evaluated
    - Go to the left child if the answer is 0
    - Go to the right child if it is 1
  - Output the value of the leaf reached finally



Our Conversion Result

Theorem 2: Any $\mathrm{MEQ}_{k,n}$ decision tree of depth D (by the referee) can be implemented by a k-party QSMP with error probability δ that uses

$O(k\left(\log D + \log\left(\frac{1}{\delta}\right)\right)\mathbf{\log n})$ qubits

<u>Proof idea</u>:

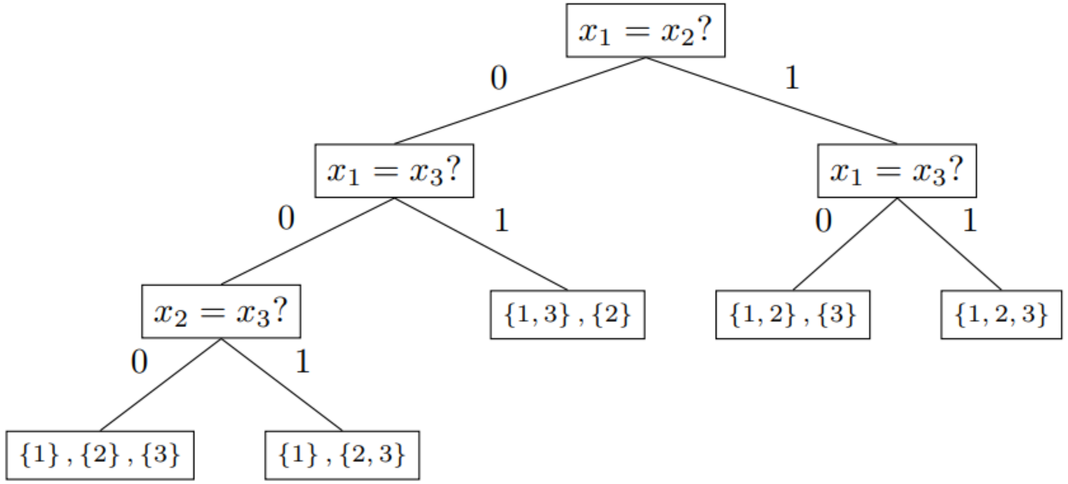
- Lemma 1 (quantum fingerprint that can modify according to the modified equality queries)
- Gentle measurement lemma (Gao's quantum union bound [Gao15])
    - If a measurement result is obtained with probability close to 1, the measured quantum state does not change so much
    - We can reuse the quantum fingerprint of Lemma 1

[Gao15] J. Gao, Physical Review A, 92, 052331 (2015)

Application 1: Grouping by Equality

- $\mathbf{GroupByEQ}_{k,n}$
  - Input: $x_\ell \in \{0,1\}^n$ for the ℓth party in k parties
  - Output: partition $S_1, \cdots, S_t$ of $[k]$ satisfying that for every $i, j \in [k]$, there is an index u such that $i, j \in S_u$ if and only if $x_i = x_j$
- Solved by $\mathbf{MEQ}_{k,n}$ decision tree of depth $\binom{k}{2}$
  - On each path, compare players' inputs against one another until the correct partition
  - By Thm 2, we have a <u>QSMP protocol of cost $O(k \log k \log n)$.</u>

Ex: $x_1 = 0000, x_2 = 1001, x_3 = 1001$
$\rightarrow \{1\}, \{2,3\}$



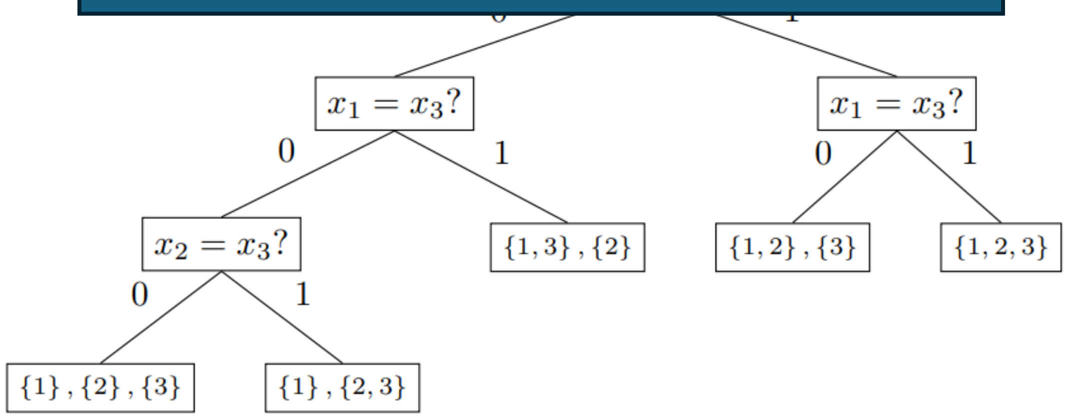
Theorem 2: Any $\mathbf{MEQ}_{k,n}$ decision tree of depth D can be implemented by a k-party QSMP with error probability δ that uses $O(k\left(\log D + \log\left(\frac{1}{\delta}\right)\right) \mathbf{log}\, \boldsymbol{n})$ qubits

Application 1: Grouping by Equality

- $\mathrm{GroupByEQ}_{k,n}$
  - Input: $x_\ell \in \{0,1\}^n$ for the ℓth party in k parties
  - Output: partition $S_1, \cdots, S_t$ of $[k]$ satisfying that for every $i, j \in [k]$, there is an index u such that $i, j \in S_u$ if and only if $x_i = x_j$
- Solved by $\mathrm{MEQ}_{k,n}$ decision tree of depth $\binom{k}{2}$
  - On each path, compare players' inputs against one another until the correct partition
  - By Thm 2, we have a <u>QSMP protocol of cost $O(k \log k \log n)$.</u>

Corollary:
QSMPs of cost $O(k \log k \log n)$ for:
- Whether all x_ℓ are equal
- Whether there is a pair (i, j) such that $x_i = x_j$

<u>Note</u>: Exponential quantum advantage in n

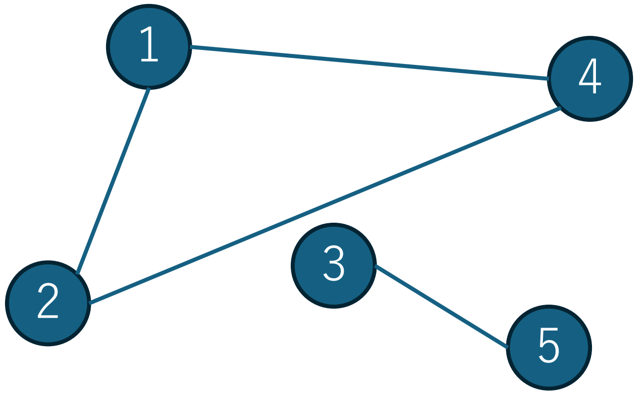


Theorem 2: Any $\mathrm{MEQ}_{k,n}$ decision tree of depth D can be implemented by a k-party QSMP with error probability δ that uses $O(k\left(\log D + \log\left(\frac{1}{\delta}\right)\right)\mathbf{\log n})$ qubits

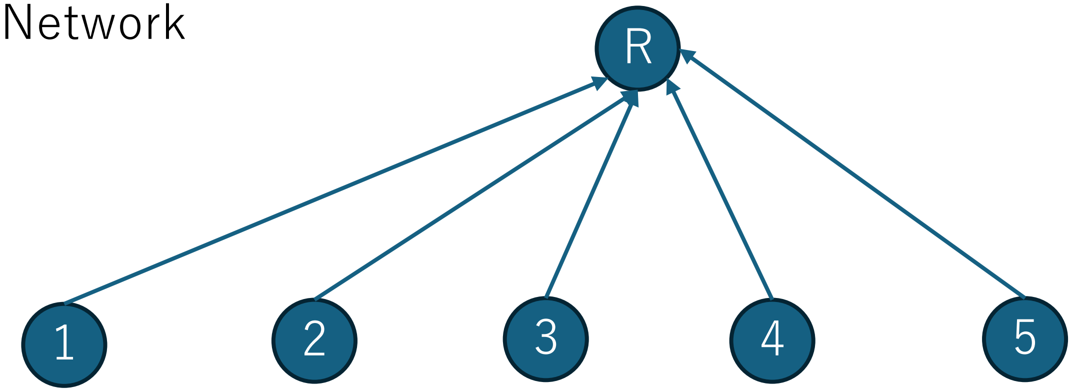
Another corollary: P3-induced subgraph freeness

- NIH (Number-In-Hand) Network
  - A special case of multiparty SMP
    - Input length n = # of parties k
  - Each party u is a node of a k-node graph G, and has a neighborhood vector v_u of G (i.e., $v_u[v] = 1$ iff $v \in N(u)$) as input
  - Goal is that the referee solves a designated problem on G

5-node graph G



NIH Network

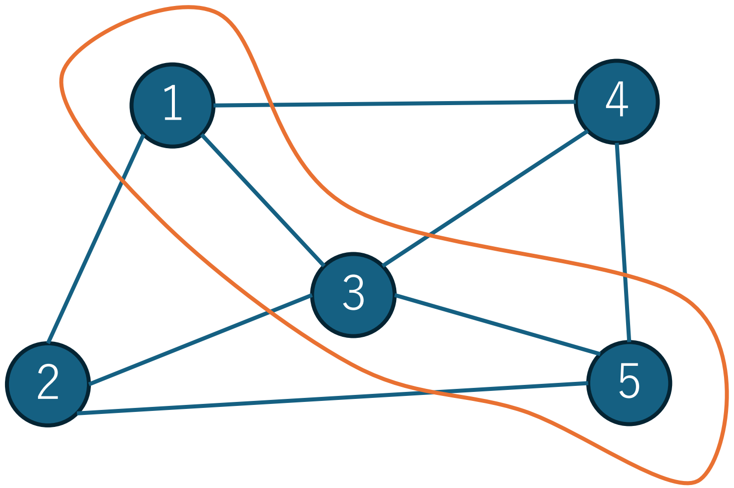


$v_1 = 01010$ $v_2 = 10010$ $v_3 = 00001$ $v_4 = 11000$ $v_5 = 00100$

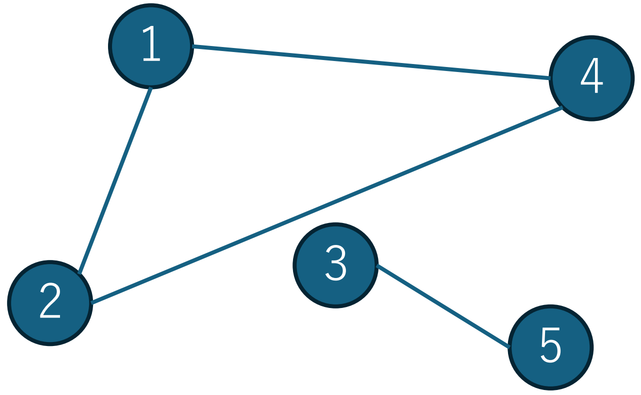
Another corollary: P3-induced subgraph freeness

- NIH (Number-In-Hand) Network
  - A special case of multiparty SMP
  - Each party u is a node of a k-node graph G, and has a neighborhood vector v_u of G (i.e., $v_u[v] = 1$ iff $v \in N(u)$) as input
  - Goal is that the referee solves a designated problem on G
- P3-induced subgraph free graph
  - A graph that does not contain a 3-node path P_3 as an induced subgraph

Not P3-induced subgraph free

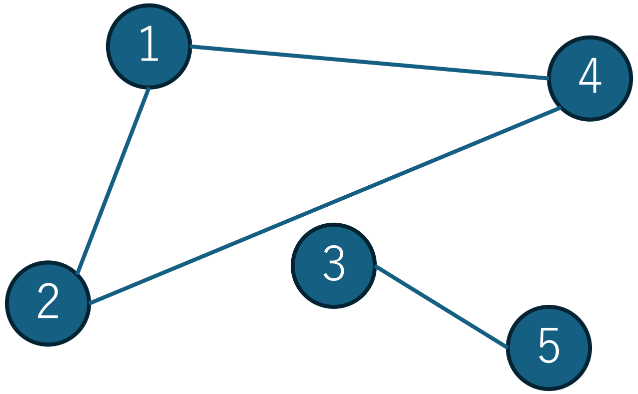


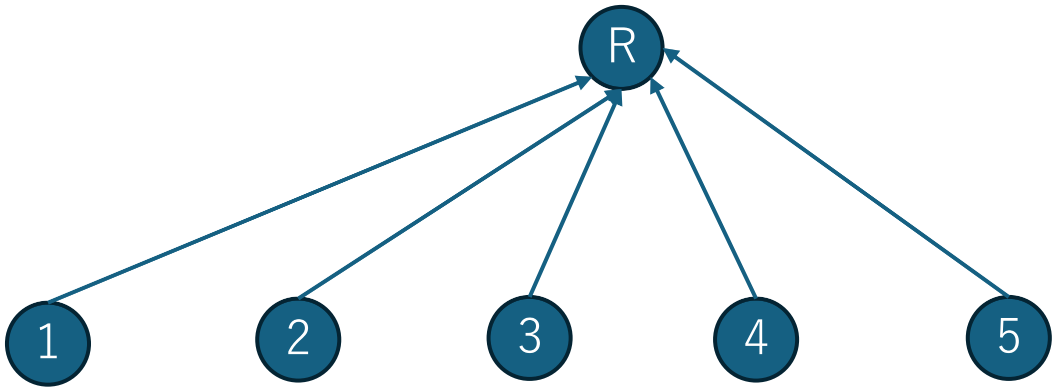
P3-induced subgraph free



Another corollary: P3-induced subgraph freeness

- NIH (Number-In-Hand) Network
  - A special case of multiparty SMP
  - Each party u is a node of a k-node graph G, and has a neighborhood vector v_u of G (i.e., $v_u[v] = 1$ iff $v \in N(u)$) as input
  - Goal is that the referee solves a designated problem on G
- P3-induced subgraph free graph
  - A graph that does not contain a 3-node path P_3 as an induced subgraph
  - A graph is <u>P3-induced subgraph free if and only if it is a collection of node-disjoint cliques</u> [KMRS15]
  - Solved by $\mathbf{GroupByEQ}_{k,k}$ to input
    $$\{\mu_u \coloneqq v_u \oplus e_u\}_u$$
    - We can reconstruct the input graph if it is P3-induced subgraph free
  - QSMP of Cost $O(k(\log k)^2)$





$v_1 = 01010$ $\mu_1 = 11010$

$v_2 = 10010$ $\mu_2 = 11010$

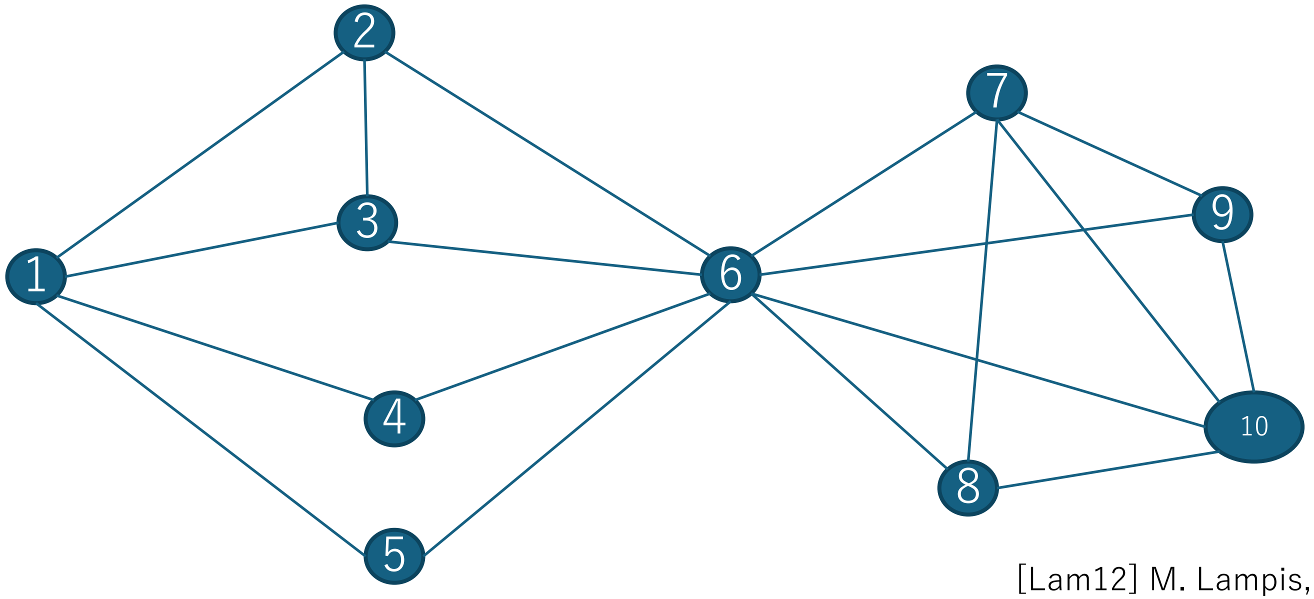
$v_3 = 00001$ $\mu_3 = 00101$

$v_4 = 11000$ $\mu_4 = 11010$

$v_5 = 00100$ $\mu_5 = 00101$

Application 2: Neighborhood Diversity

- Two nodes u, v are called **twin** if
  - $N(u) = N(v)$ (false twin)
  - $N(u) \setminus \{v\} = N(v) \setminus \{u\}$ (true twin)
- A graph has **neighborhood diversity** d if its node can be partitioned into d set but no fewer such that all nodes in each set are twins of one another [Lam12]



Neighborhood diversity=5
Partition of the same type
$\{1\},\{2,3\},\{4,5\},\{6\},\{7,8,9,10\}$

[Lam12] M. Lampis, Algorithmica, 64, pp.19-37 (2012)

Application 2: Neighborhood Diversity

- Two nodes u, v are called **twin** if
  - $N(u) = N(v)$ (false twin)
  - $N(u) \setminus \{v\} = N(v) \setminus \{u\}$ (true twin)
- A graph has **neighborhood diversity** d if its node can be partitioned into d set but no fewer such that all nodes in each set are twins of one another [Lam12]
- Solved by $\mathrm{MEQ}_{k,k}$ decision tree of depth $2\binom{k}{2}$ by queries
  - $\mathrm{MEQ}_{k,k}(v_u, v_v, 0^k, 0^k)$ (whether $N(u) = N(v)$)
  - $\mathrm{MEQ}_{k,k}(v_u, v_v, e_u, e_w)$ (whether $N(u) \setminus \{v\} = N(v) \setminus \{u\}$)
- By Thm 2, we have a QSMP protocol of cost $O(k(\log k)^2)$ for neighborhood diversity

Theorem 2: Any $\mathrm{MEQ}_{k,n}$ decision tree of depth D can be implemented by a k-party QSMP with error probability δ that uses $O(k\left(\log D + \log\left(\frac{1}{\delta}\right)\right)\log n)$ qubits

Application 3: P4-induced subgraph freeness

- ✓ P3-induced subgraph free graph
- P4-induced subgraph free graph
  - A graph that does not contain a 4-node path P_4 as an induced subgraph
- We can solve (reconstruct the input graph if it is P4-induced subgraph free) by a $\mathrm{MEQ}_{k,k}$ decision tree of depth $2(k-1)\binom{k}{2}$, and thus we have a **QSMP of cost $O(k(\log k)^2)$.**

Theorem 2: Any $\mathrm{MEQ}_{k,n}$ decision tree of depth D can be implemented by a k-party QSMP with error probability δ that uses $O(k\left(\log D + \log\left(\frac{1}{\delta}\right)\right)\log n)$ qubits

Application 3: P4-induced subgraph freeness

✓ P4-induced subgraph free graph

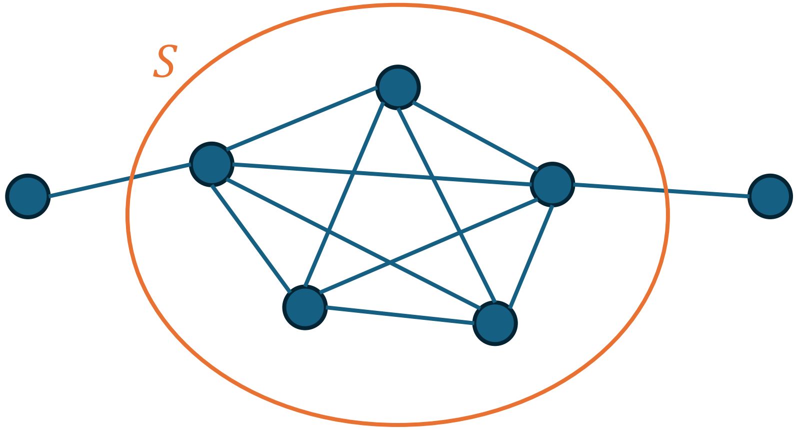
- A graph that does not contain a 4-node path P_4 as an induced subgraph

- We can solve (reconstruct the input graph if it is P4-induced subgraph free) by a $\mathrm{MEQ}_{k,k}$ decision tree of depth $2(k-1)\binom{k}{2}$, and thus we have a **QSMP of cost $O(k(\log k)^2)$.**
- **Characterization of P4-induced subgraph free graphs**
  - Characterized by the existence of a decomposition [KMRS15]: a sequence of nodes $(v_1, v_2, \cdots, v_k)$ such that for each $j \in [k-1]$, one of the following holds:
    - v_j has a true twin in $G[\{v_j, \cdots, v_k\}]$
    - v_j has a false twin in $G[\{v_j, \cdots, v_k\}]$
  - Key point: this decomposition can be described by two families of binary vectors $\{a_v\}_v$ and $\{b_v\}_v$ updated sequentially and checking the following type of queries:
    - $\exists w, u[b_w = b_u]$
    - $\exists w, u[b_w \oplus a_w = b_u \oplus a_u]$

Theorem 2: Any $\mathrm{MEQ}_{k,n}$ decision tree of depth D can be implemented by a k-party QSMP with error probability δ that uses $O\left(k\left(\log D + \log\left(\frac{1}{\delta}\right)\right)\boldsymbol{\log n}\right)$ qubits

Application 4: Enumeration of Isolated Cliques

- Clique:=complete graph
- Clique enumeration
  - Enumerate all the cliques
  - Well-studied in complex network analysis
- Isolated pseudo clique enumeration [IIO05,KHMN09]
  - (**max isolated clique [KHMN09]**) A subgraph S of $G=(V,E)$ is called a max-d-isolated clique if the subgraph induced by S is a clique, and each node in S has at most d edges to $V \setminus S$



[IIO05] H. Ito, K. Iwama, T. Osumi, ESA2005, 119-130
[KHMN09] C. Komusiewicz, F. Hüffner, H. Moser, R. Niedermeier, Theoretical Computer Science, 410, 3640-3654 (2009)

Application 4: Enumeration of Isolated Cliques

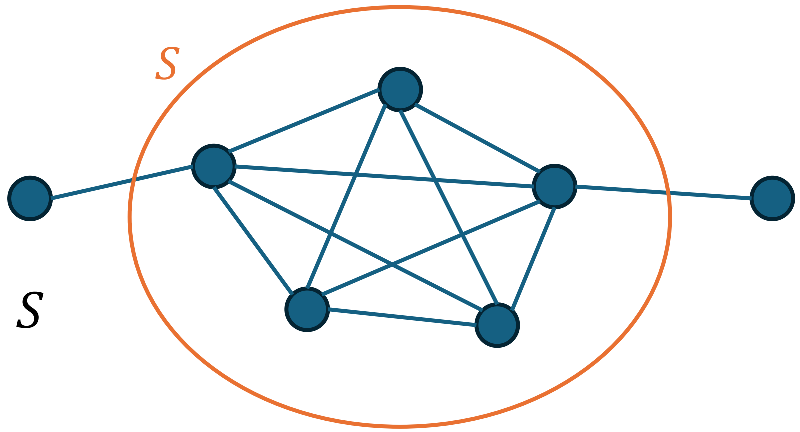
(**max isolated clique [KHMN09]**) A subgraph S of $G = (V, E)$ is called a max-d-isolated clique if the subgraph induced by S is a clique, and each node in S has at most d edges to $V \setminus S$

Theorem: There is QSMP protocol of cost $O(kd\,(\log k)^2)$ for enumerating all the max-$\underline{d}$-isolated cliques

Proof: Use the queries on Hamming distance, MHAM:

- $\mathrm{MHAM}_n^d(i, j, y, z) = \begin{cases} \Delta(x_i \oplus y, x_j \oplus z) & (\Delta(x_i \oplus y, x_j \oplus z) \leq d) \\ \perp & (\Delta(x_i \oplus y, x_j \oplus z) > d) \end{cases}$ can be computed by a $\mathrm{MEQ}_{k,n}$ decision tree of depth $\sum_{c=0}^{d} \binom{n}{c} = O(n^d)$
- Check the following conditions

1. $\mathrm{MHAM}_k^{2d}(u, v, e_u, e_v) \neq \perp$ for all $u, v \in S$
2. $\mathrm{MHAM}_k^{2d+2}(u, v, 0^k, 0^k) = \mathrm{MHAM}_k^{2d}(u, v, e_u, e_v) + 2$ for all $u, v \in S$
3. $\deg(u) \leq |S| + d - 1$ for all $u \in S$



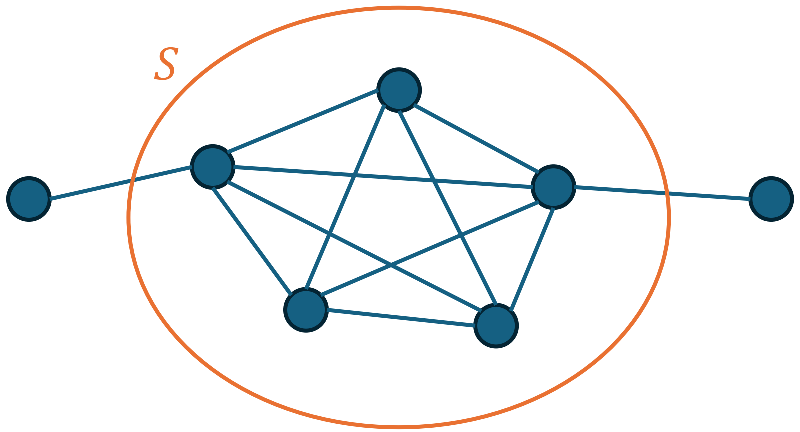
Summary

Problem	Total complexity	Local complexity	Comments
Group-by-EQ	$k \log k \log n$	$\log k \log n$	total complexity $\Omega(k\sqrt{n})$ in classical case
Neighborhood diversity	$k(\log k)^2$	$(\log k)^2$	NIH Network model
Reconstruction of P3/P4-induced subgraph free graphs	$k(\log k)^2$	$(\log k)^2$	NIH Network model
Reconstruction of distance hereditary graphs	$k(\log k)^2$	$(\log k)^2$	NIH Network model
Enumeration of max-d-isolated cliques	$kd(\log k)^2$	$d(\log k)^2$	NIH Network model

Our only quantum technique: Conversion from efficient decision trees based on "modified EQ (equality) queries" to efficient multiparty QSMP protocols

Our (rough) message: If your problem reduces to "modified EQ queries", you can find an efficient QSMP

Future Work



- More efficient multiparty QSMP protocols
  - Reconstruction of P5-induced subgraph free graphs
  - Enumerations of isolated pseudo cliques by other closeness factor
    - Max-d-isolated clique → average-d-isolated clique [IIO05]
  - Graph connectivity
    - Efficient public-coin classical SMP (graph sketch [AGM12])
- Lower bounds
  - Extension of Gap-Parity in [GIW13]
    - $GP_k(x_1, \cdots, x_k) \coloneqq \begin{cases} 1 & (\text{Hamming weight of } x_1 \oplus \cdots \oplus x_k \geq 2n/3) \\ 0 & (\text{Hamming weight of } x_1 \oplus \cdots \oplus x_k \leq n/3) \end{cases}$
    - Quantum: $\Omega(kn^{1-\frac{2}{k}})$

[AGM12] K. Ahn, S. Guha, A. McGregor, SODA2012, pp.459-467