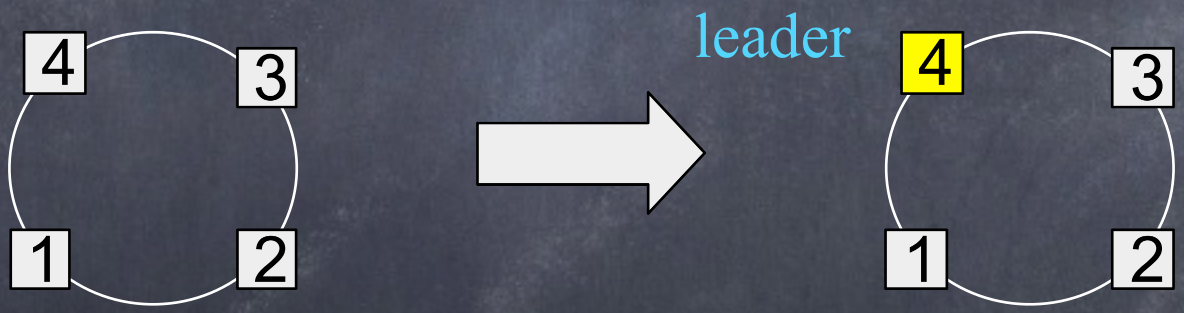
# Computing on Quantum Anonymous Networks

Seiichiro Tani

ioint work with Hirotada & Keiii

# Leader Election Problem

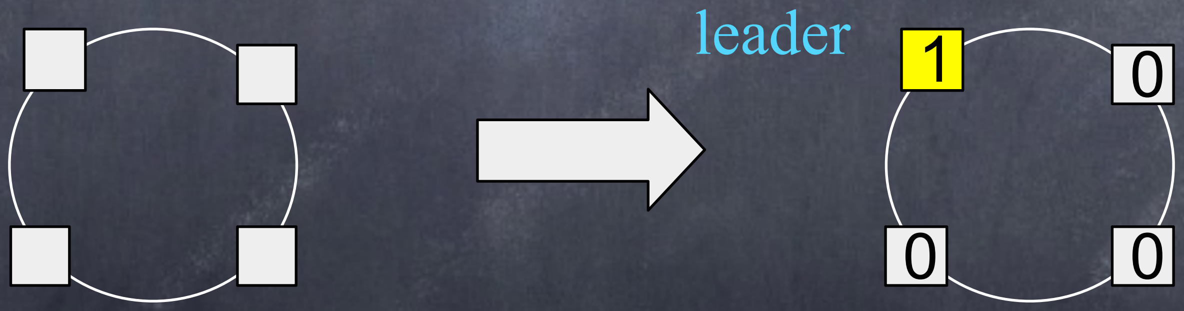
- n parties are connected by communication channels.
- The goal of all parties is to elect a unique leader.



Just to find the maximum ID,
if every party has a unique ID.

# Leader Election Problem
# on an anonymous network

- n parties are connected by communication channels, and no party has a unique identifier.
- The goal of all parties is to elect a unique leader.

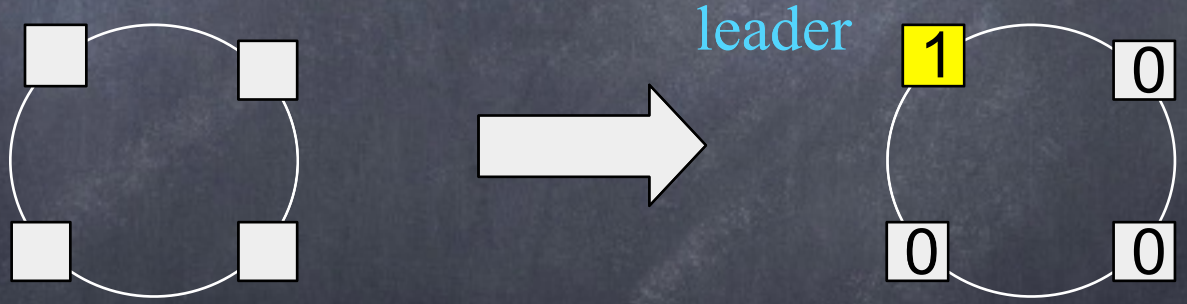


Yes, it is easy to solve the problem with some probability < 1

# Leader Election Problem on an anonymous network

- n parties are connected by communication channels, and no party has a unique identifier.

- The goal of all parties is to exactly elect a unique leader.



[Fact (A80,YM88)] For some large family of network topologies, no classical algorithm can exactly solve the problem even if n is known.

# Computing on Anonymous Networks

- LE is the hardest in the sense that, once it is solved, the leader can gather all distributed inputs and locally solve any distributed problem.
- But, LE can exactly be solved only for limited families of network topologies.
- Easier problem: Edge election problem can exactly be solved for wider families.
- Much easier problem: Symmetric functions can exactly be computed for all networks.

# Computing on Anonymous Quantum Networks

MODEL: n parties are connected by quantum communication channels, and every party can perform quantum computation.

[Fact (TKM05)] LE can exactly be solved on an anonymous quantum network of any unknown topology, if n is known.

Replacing classical network with quantum network
makes LE easy
from the viewpoint of computability.

How easy is LE made?

# Our Result (Informal)

- LE is quantumly reducible to computing symmetric Boolean functions.
  - LE can be solved by calling constant-times distributed algorithms for computing symmetric functions.
- As a corollary, we give a more efficient quantum LE algorithm than existing ones.

# Our Results (formal)

- $H_k$:{0,1}$^n$ →{true, false} s.t. $H_k(x)$=true iff Ham(x)=k
- $Q^{rnd}(H_k)$ and $Q^{bit}(H_k)$ are the round and bit complexities for exactly and reversibly computing $H_k$ for a superposed input.

[Th.1] If the number n of parties is given, LE can exactly be solved in $O(Q^{rnd}(H_1)+Q^{rnd}(H_0))$ rounds with bit complexity $O(Q^{bit}(H_1)+Q^{bit}(H_0))$.

[Th.2] If n is given, $H_1$ can be computed exactly and reversibly in $O(Q^{rnd}(H_0))$ rounds with bit complexity $O(n\cdot Q^{bit}(H_0))$.

# Our Results (formal)

[Corollary] LE can exactly be solved in $O(Q^{rnd}(H_0))$ rounds with bit complexity $O(n \cdot Q^{bit}(H_0))$.

- NOTE1: Computing $H_0$ can be interpreted as just checking if all parties have the same value.
- NOTE2: This does not have a classical counterpart: LE cannot exactly be solved for all networks while $H_0$ can be.

# Applications

[Corollary] If the number n of parties is given, LE can exactly solved in O(n) rounds with bit complexity $O(n^2|E|)$.

There is an $H_0$-algorithm with $Q^{rnd}(H_0)=O(n)$ and $Q^{bit}(H_0)=O(n^2\cdot|E|)$, where |E| is the # of edges.

| | Ours | Alg.I<br>[TKM05] | Alg.II<br>[TKM05] |
|---|---|---|---|
| Round | $O(n)$ | $O(n^2)$ | $O(n \log n)$ |
| Bit | $O(n^2\|E\|)$ | $O(n^2\|E\|)$ | $O(n^4\|E\| \log n)$ |

# Applications

Once a unique leader is elected,

- it is possible to compute any Boolean function that is computable on a non-anonymous network,
- it is possible to share symmetric quantum state (e.g., n-partite W-state and GHZ state),
- in $O(n)$ rounds with bit complexity $O(n^{2} \cdot |E|)$.

NOTE: The bit complexity is smaller than those of existing classical algorithms while keeping classically optimal $O(n)$ rounds.

# Proof of Th.1

# Exact Quantum Amplitude Amplification [CK98,BHMT02]

- Let unitary operator A be any quantum algorithm without intermediate measurement and suppose that

$$|\Psi> = A|0>^{\otimes n} = \Sigma_z \alpha_z|z>.$$

- Let $\chi(z)$: $\{0,1\}^{\otimes n}=\{\text{true},\text{false}\}$.

- If the initial success probability of A, $a=\Sigma_{z:\chi(z)=\text{true}} |\alpha_z|^2$, is known and it is ≥1/4, then

$$(-AF_0(\phi_a)A^\dagger F_\chi(\theta_a))\ |\Psi>=\Sigma_{z:\chi(z)=\text{true}}\alpha_z|z>$$

where $F_\chi(\theta_a)$: $|z> \rightarrow \exp(i\theta_a)\ |z>$ if $\chi(z)=\text{true}$

$F_0(\phi_a)$: $|z> \rightarrow \exp(i\phi_a)\ |z>$ if z is $0^n$

# Probabilistic Algorithm for LE

- Consider the following probabilistic algorithm.

  1. Every party flips a coin that gives the head w.p. 1/n and the tail w.p. 1-1/n.
  2. If exactly one party sees the head, the party becomes a unique leader.

- ✓ The probability of this successful case is

$$s(n) = \binom{n}{1} \cdot \frac{1}{n} \cdot \left(\frac{n-1}{n}\right)^{n-1} = \left(1 - \frac{1}{n}\right)^{n-1} \geq \frac{1}{e} \geq \frac{1}{4}.$$

# Quantization

1. Every party creates $(1-1/n)^{1/2}\,|0\rangle + (1/n)^{1/2}|1\rangle$ .

2. Every party measures the state.

3. If exactly one party measures $|1\rangle$ , the party becomes a unique leader.

☑ The probability of this successful case is

$$s(n) = \binom{n}{1} \cdot \frac{1}{n} \cdot \left(\frac{n-1}{n}\right)^{n-1} = \left(1 - \frac{1}{n}\right)^{n-1} \geq \frac{1}{e} \geq \frac{1}{4}.$$

# Quantization

1. Every party creates (1-1/n)$^{1/2}$ |0⟩ + (1/n)$^{1/2}$| 1⟩ .

$$A|0\rangle = \frac{1}{\sqrt{n}}\begin{pmatrix} \sqrt{n-1} & 1 \\ 1 & -\sqrt{n-1} \end{pmatrix}|0\rangle$$

2. Every party measures the state.

Let $\mathcal{A} = A^{\otimes n}$

3. If exactly one party measures |1⟩ , the party becomes a unique leader.

- ☑ The probability of this successful case is

$$s(n) = \binom{n}{1}\cdot\frac{1}{n}\cdot\left(\frac{n-1}{n}\right)^{n-1} = \left(1-\frac{1}{n}\right)^{n-1} \geq \frac{1}{e} \geq \frac{1}{4}.$$

# Applying EQAA

- $|\Psi\rangle = A|0\rangle^{\otimes n} = \{ (1-1/n)^{1/2} |0\rangle + (1/n)^{1/2}|1\rangle\}^{\otimes n} = \Sigma_z \alpha_z|z\rangle$
- $\chi(z)$=true iff Ham(z)=1.
- $(-AF_0(\phi_{s(n)})A^{\dagger}F_{\chi}(\theta_{s(n)}))\ |\Psi\rangle = \Sigma_{z:\chi(z)=true}\, \alpha_z|z\rangle$
- $F_{\chi}(\theta_{s(n)})$: every party multiplies the amplitudes of state $|z\rangle$ with $\chi(z)$=true, using $H_1$-algorithm, by a factor of $\exp(i\phi^{s(n)}/n)$, which is $\exp(i\phi^{s(n)})$ as a whole.
- $F_0(\phi_{s(n)})$: implemented in a similar way using $H_0$-algorithm.

# Apply EQAA

- All communications are performed for computing $H_0$ and $H_1$.

↓

[Th.1] If the number n of parties is given, LE can exactly solved in $O(Q^{rnd}(H_1)+Q^{rnd}(H_0))$ rounds with bit complexity $O(Q^{bit}(H_1)+Q^{bit}(H_0))$.

# Proof of Th.2

# Outline of Reduction

1. Reduce computing $H_1$ to computing $H_0$ and CONSISTENCY (defined later).
2. Reduce computing CONSISTENCY to computing $H_{0.}$

- Since $H_0$ is easy to compute, it is also easy to distinguish Ham(x)=0 from Ham(x)>0.
- To compute $H_1$, it is sufficient to consider the case of Ham(x)>0

# $H_0$ is easy to compute

1. Every party sends his input to all his neighbors (including himself).
2. Every party computes OR of all received bits and sends the results to all his neighbors (including himself).
3. Repeat Step 2 (n-2) times.

Round complexity O(n) and bit complexity O(n|E|)

# Probabilistic Algorithm for $H_1(x)$ with $Ham(x)>0$

Let's say: party with input 1 is ``marked''; party with input 0 is ``unmarked''.

Algorithm

- marked party i generates a random bit $b_i$.
- unmarked party i sets $b_i$ to 0.

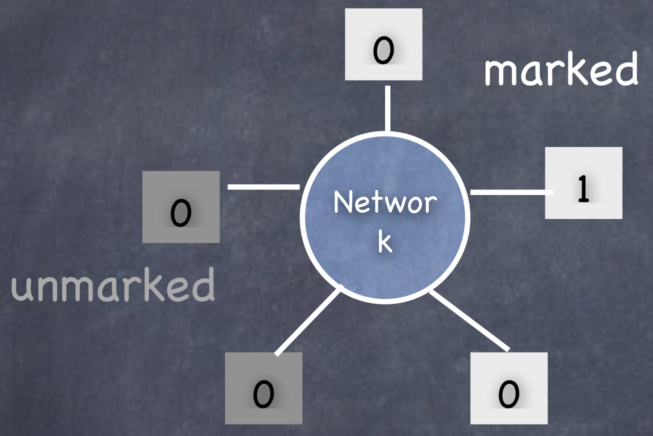
Observation

- If $Ham(x)=1$, $b_i$ of the marked party is either 0 or 1.
- If $Ham(x)>1$, there exists marked parties i,j s.t. $b_i \neq b_j$ with high prob.

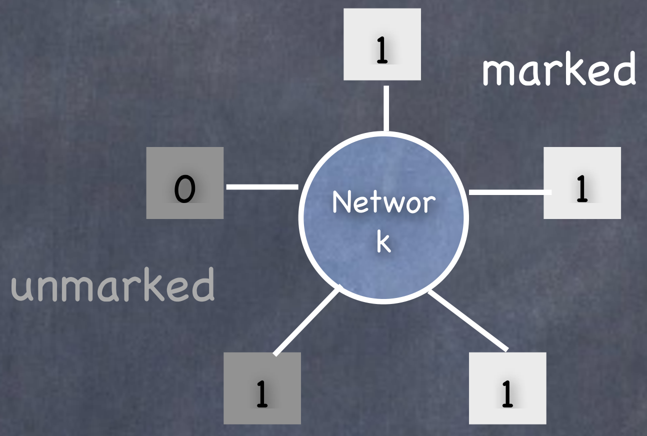
(By simply exchanging $b_i$'s, every party can detect the current situation among the above two.)

# (In)Consistent String/States

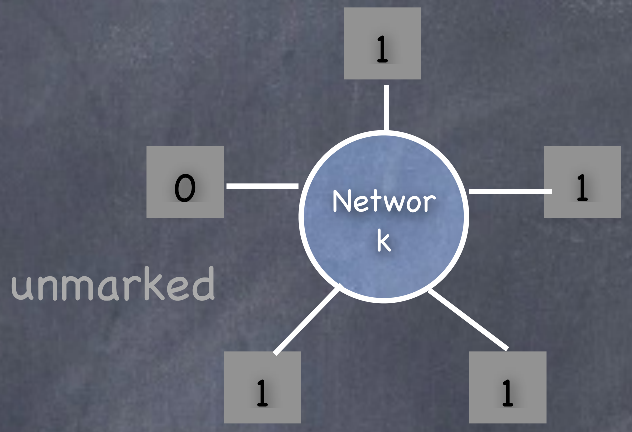
- S: the set of marked parties (|S|=Ham(x)).



string 01001 is inconsistent over S



string 11101 is consistent over S



string 11101 is consistent over S

- In the quantum case, $\Sigma_z \alpha_z |z\rangle$ is said consistent (inconsistent) over S, if $\alpha_z \neq 0$ only for

# Quantum version

Algorithm

- Every marked party i creates $|b_i\rangle=(|0\rangle+|1\rangle)$
- Every unmarked party i creates $|b_i\rangle=|0\rangle$

The entire state is

$$((|0\rangle+|1\rangle)/2^{1/2})^{\otimes|S|} \otimes |0\rangle^{\otimes(n-|S|)} = (1/2^{|S|/2}) \sum_z |z\rangle$$

- Want to amplify the amplitude of the inconsistent states $|Z\rangle$ to 1.

  Then, we could distinguish $\mathrm{Ham}(x)\leq 1$ from $\mathrm{Ham}(x)\geq 2$

# Applying EQAA (1)

- $|\Psi\rangle = A|0\rangle^{\otimes n}$

  $= \Sigma_z \alpha_z |z\rangle$

- Marked parties perform $H|0\rangle = (|0\rangle + |1\rangle)/2^{1/2}$
- Unmarked parties perform $I|0\rangle = |0\rangle$

- $\chi$(z) is true iff z is inconsistent over S

The following state would be what we want.

- $(-AF_0(\phi_a)A^{\dagger}F_{\chi}(\theta_a))\ |\Psi\rangle = \Sigma_{z:\ \chi(z)=\mathrm{true}}\ \alpha_z |z\rangle$

Suppose we are given:

- algorithm for computing $H_0$
- algorithm for computing $\mathrm{CONSISTENCY}_S$ (that is true iff input z is consistent over S)

In a way similar to Th. 1, we can implement $F_{\chi}(\theta_a)$,

$F_0(\phi_a)$ if we know a.

# Applying EQAA (2)

- Unfortunately, the initial success probability a is unknown: $a= 1-2/2^{|S|}$ and $|S|=\mathrm{Ham}(x)$ is unknown.

- Instead, we pick a guess t of |S|, set $a(t)=1-2/2^{t}>1/4$ (for $t\geq 2$), and perform

$$B(t)^{\otimes n}|0^n\rangle=(-AF_0(\phi_{a(t)})A^{\dagger}F_{\chi}(\theta_{a(t)}))\ |\Psi\rangle$$

Complexity: All communications are performed for computing $H_0$ and CONSISTENCY$_S$.

round complexity: $O(Q^{rnd}(H_0)+Q^{rnd}(\mathrm{CONSISTENCY}_S))$
bit complexity: $O(Q^{bit}(H_0)+Q^{bit}(\mathrm{CONSISTENCY}_S))$

# How to use B(t)

$$B(t)^{\otimes n}|0^n\rangle=(-AF_0(\phi_{a(t)})A^{\dagger}F_{\chi}(\theta_{a(t)}))\,|\Psi\rangle$$

Observation

- If $t=|S|>1$, $B(t)^{\otimes n}|0^n\rangle$ is inconsistent due to EQAA theorem.
- If $|S|=Ham(x)=1$, $B(t)^{\otimes n}|0^n\rangle$ is consistent for any t (Because the amplitude of any consistent state in $|\Psi\rangle=A|0^n\rangle$ is 0.)

We run $B(t)^{\otimes n}|0^n\rangle$ for every t=2,...,n.

- If $|S|>1$, $B(t)^{\otimes n}|0^n\rangle$ is inconsistent for some t w.p. 1.
- If $|S|=1$, $B(t)^{\otimes n}|0^n\rangle$ is consistent for any t.

# The Algorithm

```
1. Run $H_0$-algorithm and store the result into $S_0$
2. Do the following steps for every t=2,…,n in parallel.
    (i) Perform B(t) to register R initialized to |0>.
    (ii) Run CONSISTENCY$_S$-algorithm over R.
3. (Decision step)
    (i) If $S_0$ is true, output ``false”;
    (ii) else if $S_t$ is inconsistent for some t, output ``false”;
    (iii) else output ``true”.
```

round complexity: $O(Q^{rnd}(H_0)+Q^{rnd}(\mathrm{CONSISTENCY}_S))$
bit complexity: $O(\ n\ (Q^{bit}(H_0)+Q^{bit}(\mathrm{CONSISTENCY}_S)))$

# Complexity of Computing CONSISTENCY$_S$

- Can be computed with O(1) calls of $H_0$-algorithm.

1. Unmarked parties set their input to 0.
2. Run $H_0$-algorithm.
3. Marked parties flip their input, and run $H_0$-algorithm.
4. If one of the results is true, output true; otherwise false.

round complexity: $Q^{rnd}(\text{CONSISTENCY}_S)=O(Q^{rnd}(H_0))$
bit complexity: $Q^{bit}(\text{CONSISTENCY}_S)=O(Q^{bit}(H_0))$

# Putting Together

[Th.2] If n is given, $H_1$ can be computed exactly and reversibly

in $O(Q^{rnd}(H_0))$ rounds

with bit complexity $O(n \cdot Q^{bit}(H_0))$.

# Discussions

- Our quantum leader election algorithm assumes undirected networks, because the $H_0$-algorithm we used needs ``uncomputing" procedure to erase garbage. Is there a linear-round quantum leader election algorithm on directed anonymous networks?

- All existing quantum leader election algorithms use the gates depending on the number n of parties. Is there algorithm that works for every n with constant-sized gate set?

- Even if only a constant-sized universal gate set is available, our algorithm can elect a unique leader with arbitrarily small error probability without increasing communication cost. In the classical setting, communication cost seems to grow as error probability is made small (true?).