量子アルゴリズム

名古屋大学大学院多元数理科学研究科
フランソワ　ルガル

量子計算

✓ 量子力学の法則に基づく計算パラダイ

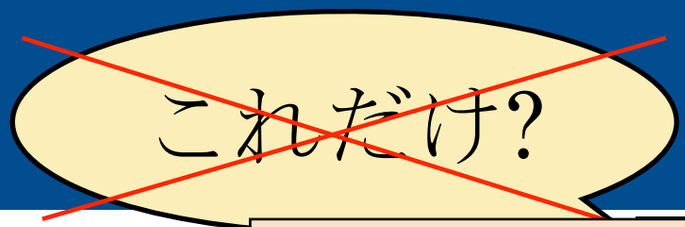
✓ 様々な応用

➢ 高速アルゴリズムの実現

Shor’s algorithm (1994)
素因数分解

Grover’s algorithm (1996)
量子探索

量子アルゴリズム



- quantum algorithms with amplitude amplification [Brassard+ 1999]
- quantum algorithms for element disjointness [Ambainis 2002]
- quantum algorithms for Gauss sums [van Dam + 2002]
- quantum algorithms for solving Pell's equation [Hallgren 200
- quantum algorithms for quantum simulations [Childs 2004]
- quantum algorithms for hidden subgroups [Kuperberg 2004]
- quantum algorithms for finding an unit group [Hallgren 2005]
- quantum algorithms for triangle finding [Magniez+ 2005]
- quantum algorithms for computing knot invariants [Aharonov
- quantum algorithms for data streams [Le Gall 2006]
- quantum algorithms for hidden nonlinear structures [Childs+
- quantum algorithms for evaluating NAND formulas [Fahri+ 2
- quantum algorithms for group isomorphism [Le Gall 2010]
- quantum algorithms for matrix multiplication [Le Gall 2011]
- quantum algorithms using span programs [Belovs 2011]
- quantum algorithms for matrix inversion [Ta-Shma 2013]
- quantum algorithms for pattern matching [Montanaro 2014]
- ….

Shor's algorithm (1994)
素因数分解

Quantum Algorithm Zoo

This is a comprehensive catalog of quantum algorithms. If you notice any errors or omissions, please email me at stephen.jordan@microsoft.com. (Alternatively, you may submit a pull request to the repository on github.) Your help is appreciated and will be acknowledged.

Algebraic and Number Theoretic Algorithms

Algorithm: Factoring
Speedup: Superpolynomial
Description: Given an *n*-bit integer, find the prime factorization. The quantum algorithm of Peter Shor solves this in $\widetilde{O}(n^3)$ time [82,125]. The fastest known classical algorithm for integer factorization is the general number field sieve, which is believed to run in time $2^{\widetilde{O}(n^{1/3})}$. The best rigorously proven upper bound on the classical complexity of factoring is $O(2^{n/4+o(1)})$ via the Pollard-Strassen algorithm [252, 362]. Shor's factoring algorithm breaks RSA public-key encryption and the closely related quantum algorithms for discrete logarithms break the DSA and ECDSA digital signature schemes and the Diffie-Hellman key-exchange protocol. A quantum algorithm even faster than Shor's for the special case of factoring "semiprimes", which are widely used in cryptography, is given in [271]. If small factors exist, Shor's algorithm can be beaten by a quantum algorithm using Grover search to speed up the elliptic curve factorization method [366]. Additional optimized versions of Shor's algorithm are given in [384, 386]. There are proposed classical public-key cryptosystems not believed to be broken by quantum algorithms, *cf.* [248]. At the core of Shor's factoring algorithm is order finding, which can be reduced to the Abelian hidden subgroup problem, which is solved using the quantum Fourier transform. A number of other problems are known to reduce to integer factorization including the membership problem for matrix groups over fields of odd order [253], and certain diophantine problems relevant to the synthesis of quantum circuits [254].

Algorithm: Discrete-log
Speedup: Superpolynomial
Description: We are given three *n*-bit numbers *a*, *b*, and *N*, with the promise that $b = a^s \mod N$ for some *s*. The task is to find *s*. As shown by Shor [82], this can be achieved on a quantum computer in poly(*n*) time. The fastest known classical algorithm requires time superpolynomial in *n*. By similar techniques to those in [82], quantum computers can solve the discrete logarithm problem on elliptic curves, thereby breaking elliptic curve cryptography [109, 14]. A further optimization to Shor's algorithm is given in [385]. The superpolynomial quantum speedup has also been extended to the discrete logarithm problem on semigroups [203, 204]. See also Abelian hidden subgroup.

Algorithm: Pell's Equation
Speedup: Superpolynomial
Description: Given a positive nonsquare integer *d*, Pell's equation is $x^2 - dy^2 = 1$. For any such *d* there are infinitely many pairs of integers (*x*,*y*) solving this equation. Let (x_1, y_1) be the pair that

420 件 (2020年6月29日時点)

本日の予定

I. 量子探索アルゴリズム ］ 1時間

II. その他の量子アルゴリズムの簡単な紹介 ］ 30分

- Shorの素因数アルゴリズム
- HHLアルゴリズム
- 量子ウォーク

III. 量子探索アルゴリズムの応用例 ］ 1時間

量子分散計算

本日の予定

I. 量子探索アルゴリズム ］ 1時間

II. その他の量子アルゴリズムの簡単な紹介

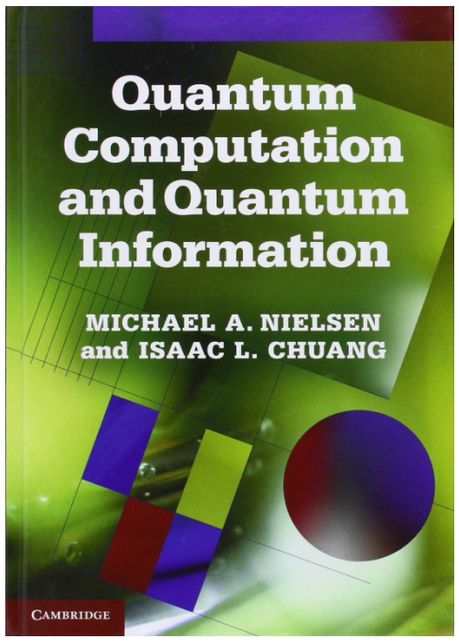
- Shorの素因数アルゴリズム
- HHLアルゴリズム
- 量子ウォーク

］ 30分

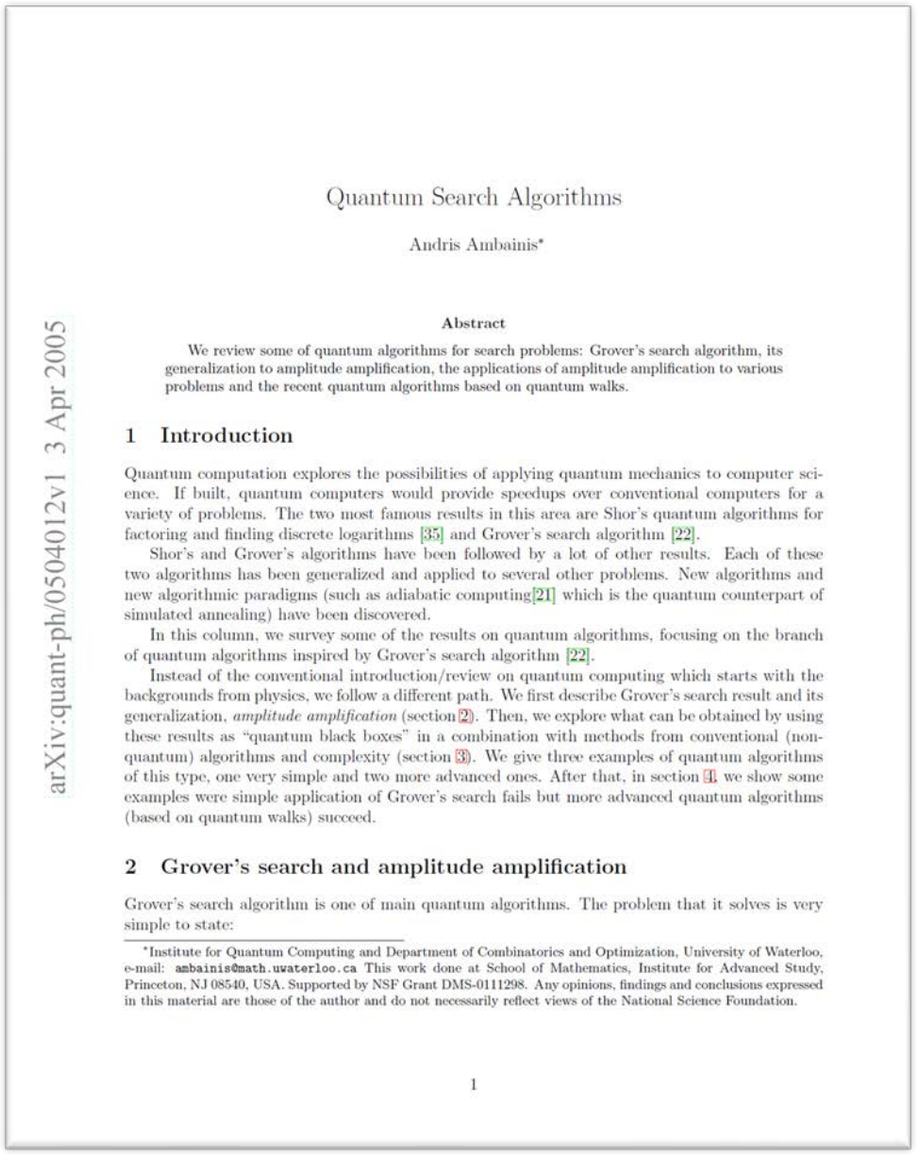
III. 量子探索アルゴリズムの応用例 ］ 1時間

量子分散計算

おすすめの参考文献



6章：量子探索



Quantum Search Algorithms

Andris Ambainis*

Abstract

We review some of quantum algorithms for search problems: Grover's search algorithm, its generalization to amplitude amplification, the applications of amplitude amplification to various problems and the recent quantum algorithms based on quantum walks.

1 Introduction

Quantum computation explores the possibilities of applying quantum mechanics to computer science. If built, quantum computers would provide speedups over conventional computers for a variety of problems. The two most famous results in this area are Shor's quantum algorithms for factoring and finding discrete logarithms [35] and Grover's search algorithm [22].

Shor's and Grover's algorithms have been followed by a lot of other results. Each of these two algorithms has been generalized and applied to several other problems. New algorithms and new algorithmic paradigms (such as adiabatic computing[21] which is the quantum counterpart of simulated annealing) have been discovered.

In this column, we survey some of the results on quantum algorithms, focusing on the branch of quantum algorithms inspired by Grover's search algorithm [22].

Instead of the conventional introduction/review on quantum computing which starts with the backgrounds from physics, we follow a different path. We first describe Grover's search result and its generalization, *amplitude amplification* (section 2). Then, we explore what can be obtained by using these results as "quantum black boxes" in a combination with methods from conventional (non-quantum) algorithms and complexity (section 3). We give three examples of quantum algorithms of this type, one very simple and two more advanced ones. After that, in section 4, we show some examples were simple application of Grover's search fails but more advanced quantum algorithms (based on quantum walks) succeed.

2 Grover's search and amplitude amplification

Grover's search algorithm is one of main quantum algorithms. The problem that it solves is very simple to state:

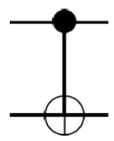
*Institute for Quantum Computing and Department of Combinatorics and Optimization, University of Waterloo, e-mail: ambainis@math.uwaterloo.ca This work done at School of Mathematics, Institute for Advanced Study, Princeton, NJ 08540, USA. Supported by NSF Grant DMS-0111298. Any opinions, findings and conclusions expressed in this material are those of the author and do not necessarily reflect views of the National Science Foundation.

Andris Ambainis. Quantum search algorithms. SIGACT News, 35 (2):22-35, 2004.
ArXiv: quant-ph/0504012

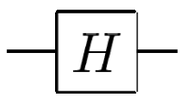
(量子探索の応用について)

1.量子回路計算量

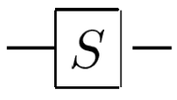
- ✓ Elementary quantum gates: 1-qubit gates and CNOT gates
  (Discrete set: H,S,T, CNOT)

CNOT 

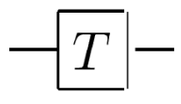
$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Hadamard 

$$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

Phase 

$$\begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$$

$\pi/8$ 

$$\begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix}$$

- ✓ The complexity of a quantum algorithm is the minimum number of elementary gates necessary to implement it

Theorem:

> Any unitary transform over n qubits can be implemented using $O(n^2 4^n)$ elementary quantum gates

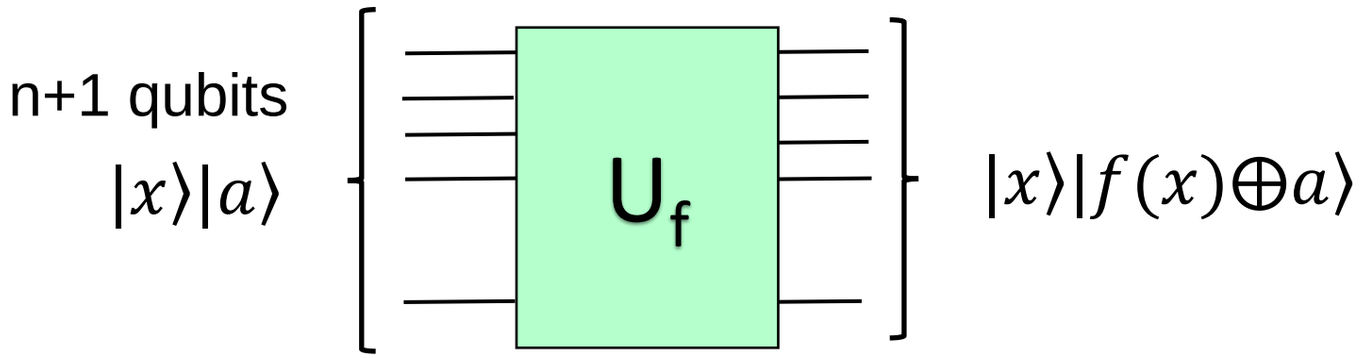
Efficient implementation: implementation using a number of elementary gates polynomial in n

1.量子回路計算量

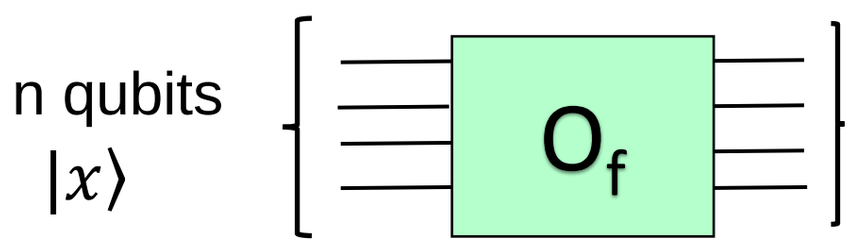
- ✓ For any function f: $\{0,1\}^n \rightarrow \{0,1\}$ there exists a unitary matrix U_f of size 2^{n+1} x 2^{n+1} such that

$$|x_1, x_2, \dots, x_n\rangle|0\rangle \xrightarrow{U_f} |x_1, x_2, \dots, x_n\rangle|f(x_1, x_2, \dots, x_n)\rangle$$

$$|x_1, x_2, \dots, x_n\rangle|1\rangle \xrightarrow{U_f} |x_1, x_2, \dots, x_n\rangle|f(x_1, x_2, \dots, x_n)\oplus 1\rangle$$



- ✓ If the function f can be computed efficiently classically, then the unitary U_f can be implemented efficiently
- ✓ If the function f can be computed classically in time polynomial in n, then the unitary U_f can be implemented using a number of elementary gates polynomial in n



"equivalent"

2. Groverアルゴリズの概要

Let f: $\{0,1\}^n \rightarrow \{0,1\}$ be a Boolean function given as a black box

Goal: find an element $x \in \{0,1\}^n$ such that f(x) = 1

Classically this can be done using $O(2^n)$ calls to the black box
("brute force search: try all the elements x")

There is a quantum algorithm solving this problem with $O(\sqrt{2^n})$ calls to O_f

Quantum search
[Grover 96]

Example of application: quantum algorithm for Boolean satisfiability (SAT)

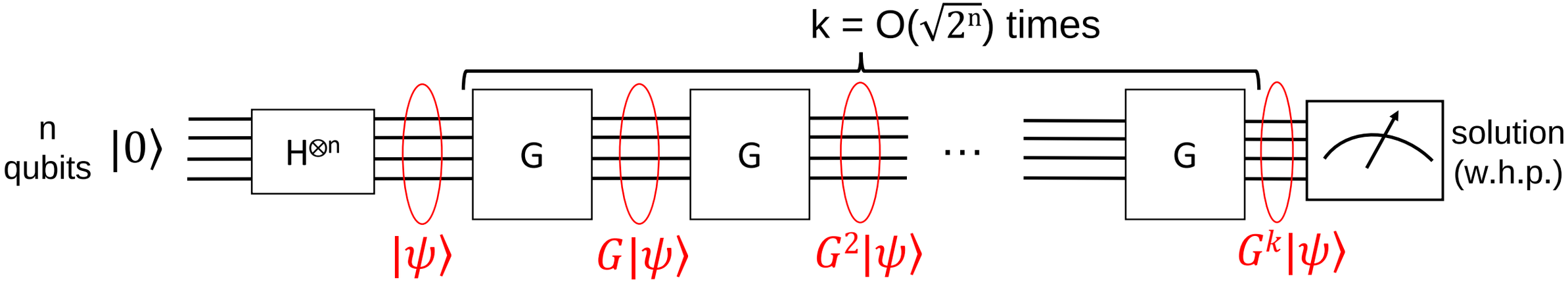
SAT: given a Boolean formula f of poly size on n variables, find a satisfying assignment (if such an assignment exists)

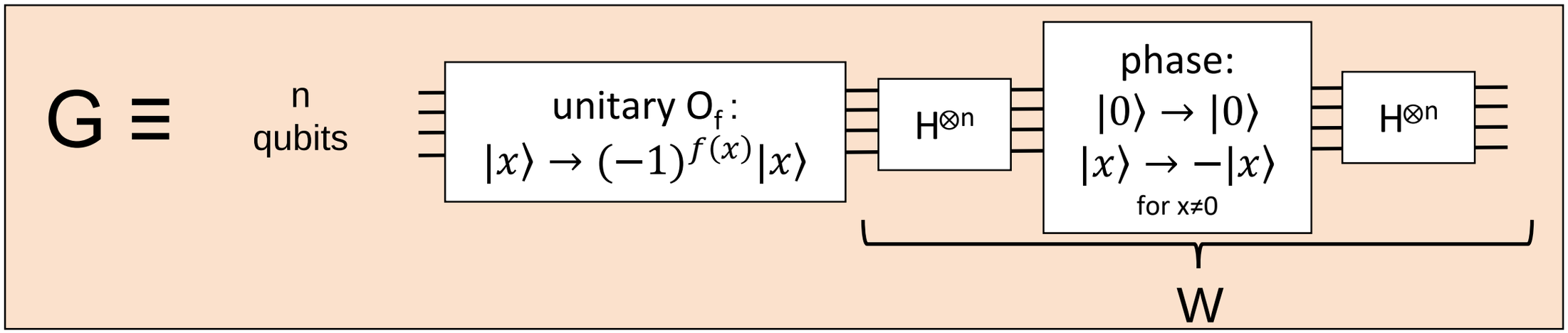
x = one possible assignment ⟵ 2^n possibilities

Black box: computes f(x) from x ⟵ poly(n) time

⇨ Quantum search solves SAT in $O(2^{n/2} \times \text{poly}(n))$ time

3. Groverアルゴリズムの詳細

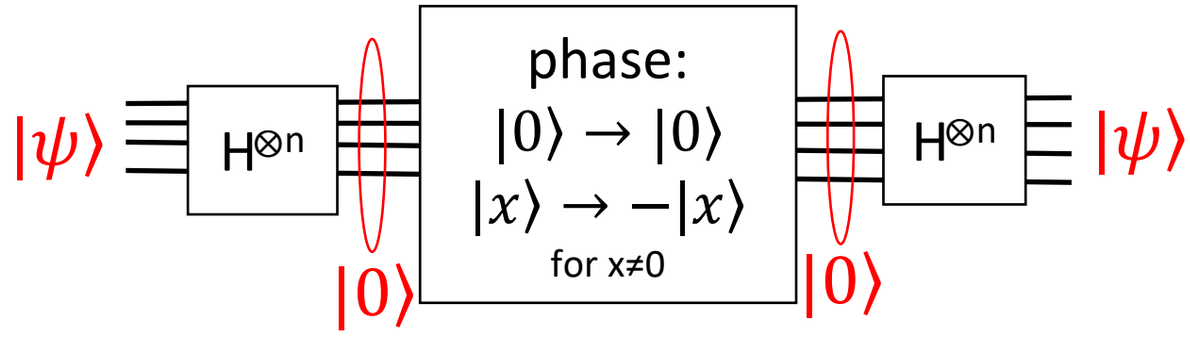




$$|\psi\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle$$

Property:

- ✓ $W|\psi\rangle = |\psi\rangle$
- ✓ $W|\varphi\rangle = -|\varphi\rangle$ if $\langle\psi|\varphi\rangle = 0$



3. Groverのアルゴリズム

M: number of solutions
Assume $1 \leq M \ll 2^n$

$$|\psi_A\rangle = \frac{1}{\sqrt{M}} \sum_{x \in A} |x\rangle$$

$$|\psi_B\rangle = \frac{1}{\sqrt{2^n - M}} \sum_{x \in B} |x\rangle$$

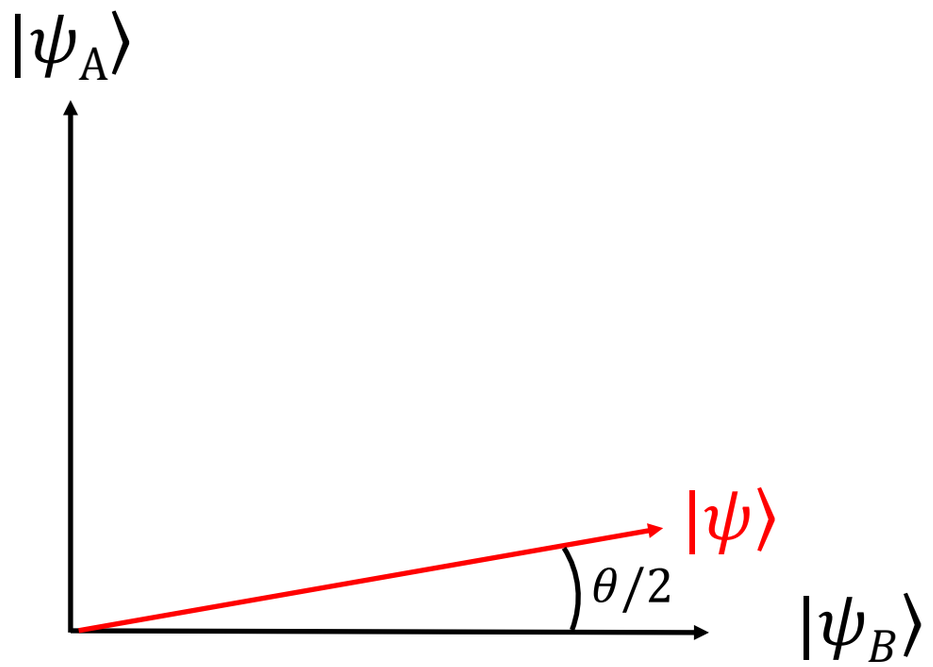
A = {x ∈{0,1}ⁿ | f(x) = 1}

B = {x ∈{0,1}ⁿ | f(x) = 0}

write M=|A|

$$|\psi\rangle = \sqrt{\frac{M}{2^n}}\ |\psi_A\rangle + \sqrt{\frac{2^n - M}{2^n}}\ |\psi_B\rangle \quad = \sin(\theta/2)\,|\psi_A\rangle + \cos(\theta/2)\ |\psi_B\rangle$$

$$|\psi\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle$$



3. Groverのアルゴリズム

M: number of solutions
Assume $1 \le M \ll 2^n$

Consider the plane spanned by $\{|\psi_A\rangle, |\psi_B\rangle\}$

for all $a, b \in \mathbb{R}$ we have $\quad a|\psi_A\rangle + b|\psi_B\rangle \xrightarrow{O_f} -a|\psi_A\rangle + b|\psi_B\rangle$

⇨ O_f is a reflection about axis $|\psi_B\rangle$ (鏡映)

$$a|\psi\rangle + b|\psi^{\perp}\rangle \xrightarrow{W} a|\psi\rangle - b|\psi^{\perp}\rangle$$

⇨ W is a reflection about axis $|\psi\rangle$ (鏡映)

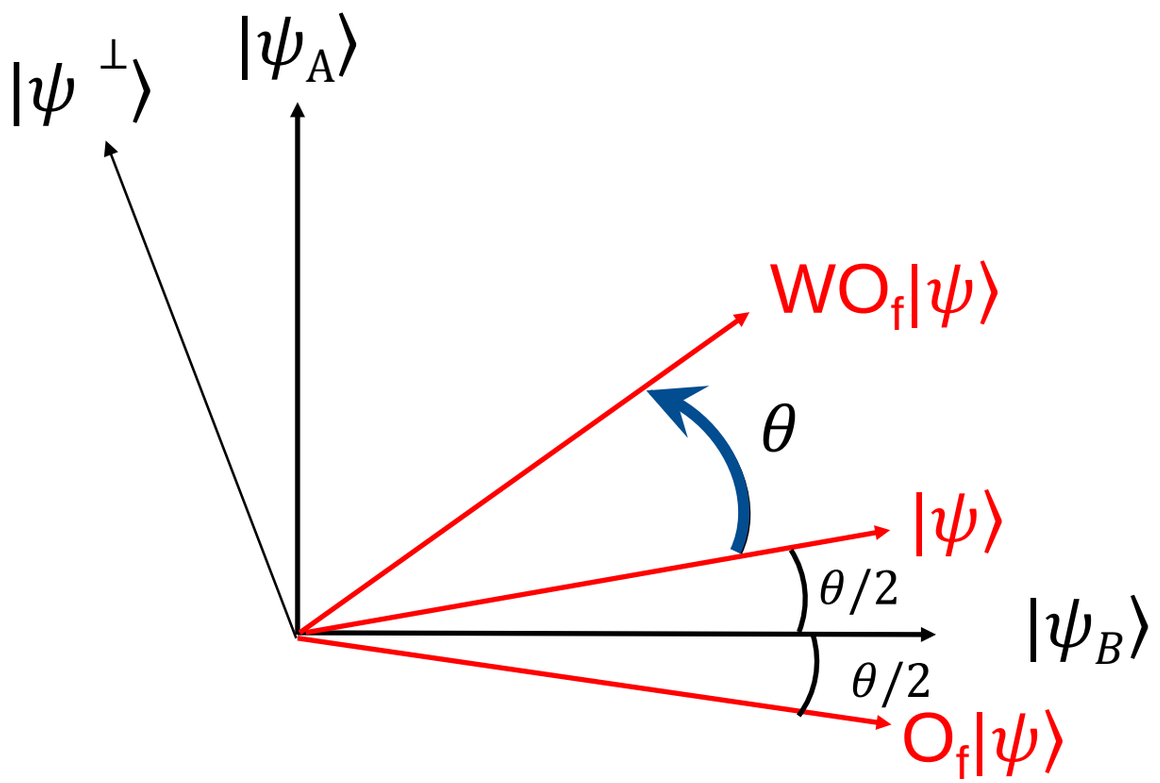
unitary O_f :
$|x\rangle \to (-1)^{f(x)}|x\rangle$

⇨ $G=WO_f$ is a rotation of angle θ (回転)

$$|\psi\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle$$

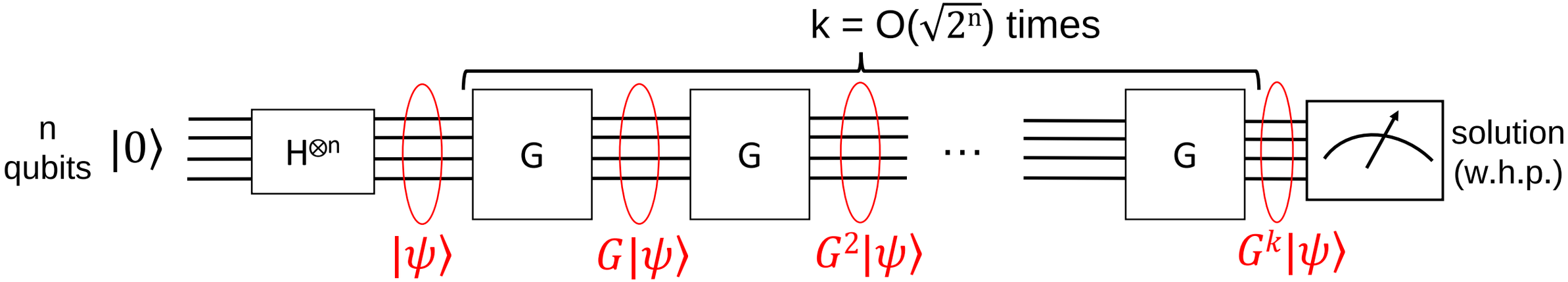
Property:

- ✓ $W|\psi\rangle = |\psi\rangle$
- ✓ $W|\varphi\rangle = -|\varphi\rangle$ if $\langle\psi|\varphi\rangle = 0$



3. Groverのアルゴリズム

M: number of solutions
Assume $1 \leq M \ll 2^n$



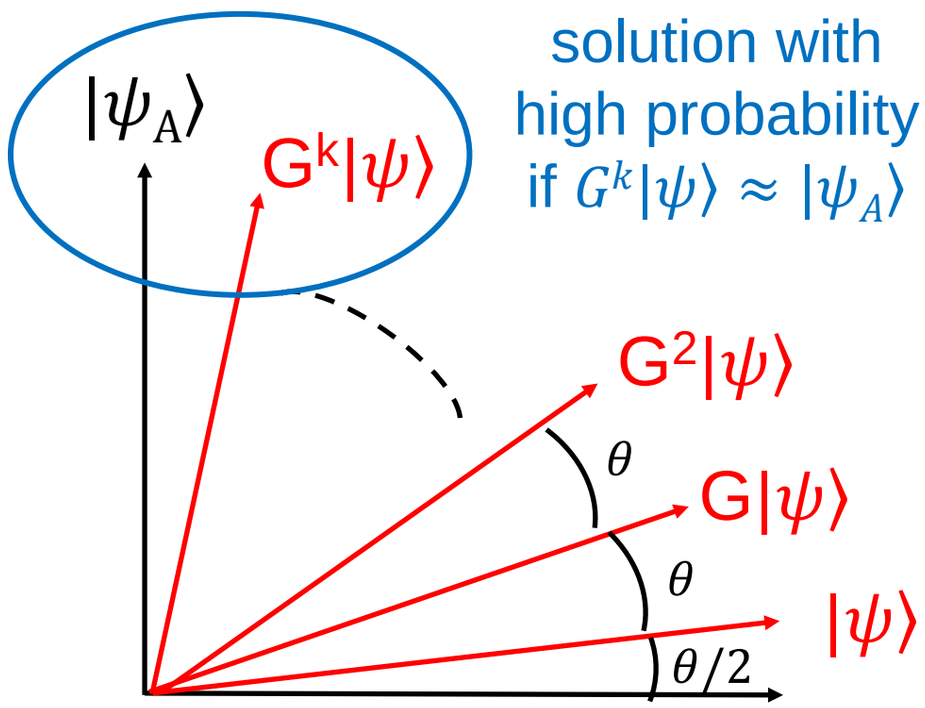
$$|\psi\rangle = \sqrt{\frac{M}{2^n}}\,|\psi_A\rangle + \sqrt{\frac{2^n - M}{2^n}}\,|\psi_B\rangle \quad = \sin(\theta/2)\,|\psi_A\rangle + \cos(\theta/2)\;|\psi_B\rangle$$

$$G^k|\psi\rangle = \sin((2k+1)\theta/2)\,|\psi_A\rangle + \cos((2k+1)\theta/2)\;|\psi_B\rangle$$

$$G^k|\psi\rangle \approx |\psi_A\rangle \text{ if } (2k+1)\theta/2 \approx \frac{\pi}{2}$$

$$k \approx \frac{\pi}{2\theta} \approx \frac{\pi}{4}\sqrt{\frac{2^n}{M}}$$

$$\sqrt{\frac{M}{2^n}} = \sin(\theta/2) \approx \theta/2$$



4. Groverアルゴリズのまとめ

M: number of solutions
Assume $1 \leq M \ll 2^n$

Let f: $\{0,1\}^n \rightarrow \{0,1\}$ be a Boolean function given as a black box

Goal: find an element $x \in \{0,1\}^n$ such that f(x) = 1

Classically this can be done using $O(2^n/M)$ calls to the black box
("try $O(2^n/M)$ element taken uniformly at random")

There is a quantum algorithm solving this problem with $O(\sqrt{2^n/M})$ calls to O_f

Quantum search
[Grover 96]

- ✓ Works even for larger values of M
- ✓ Can be adapted to work even if M is unknown

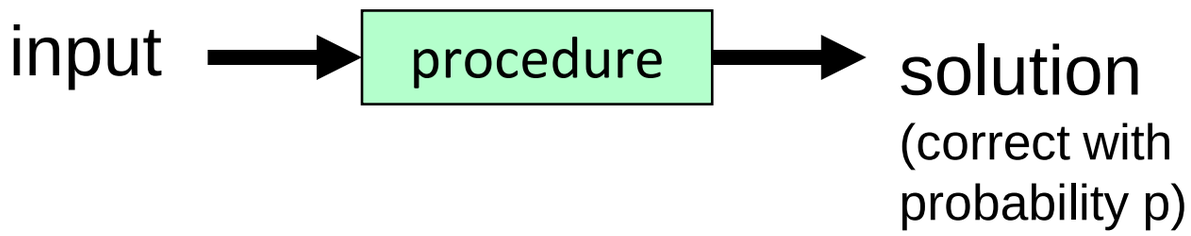
M. Boyer, G. Brassard, P. Høyer, and A. Tapp. Tight bounds on quantum searching. Fortsch.Phys.46:493-506,1998. ArXiv: 9605034.

- ✓ M can even be estimated efficiently

G. Brassard, P. Høyer, and A. Tapp. Quantum counting. Proceedings of ICALP'98, pp. 820–831. Arxiv: 9805082.

5.量子振幅増加

Consider a randomized procedure (or a quantum algorithm) that solves some problem with probability p



Classically we need to repeat the algorithm $O(1/p)$ times in order to get a solution with high probability

There is a quantum algorithm that get a solution with high probability with $O(\sqrt{1/p})$ repetitions

Quantum amplitude amplification
[Brassard- Høyer 97]

Example of application: quantum algorithm for Boolean satisfiability (SAT)

find a solution among 2^n candidates (we can check if a candidate is a solution efficiently)

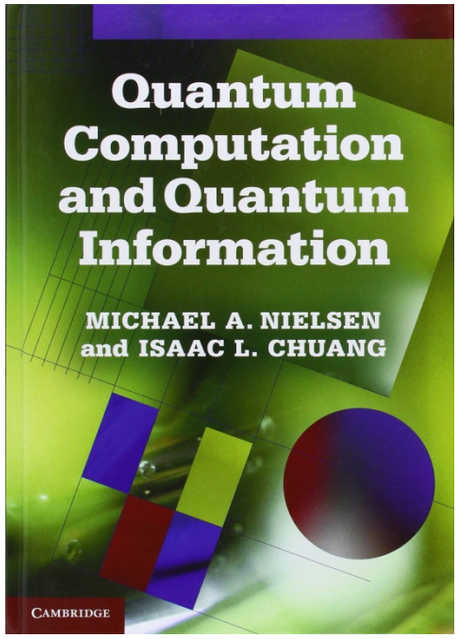
Procedure: take a candidate uniformly at random and check

Success probability: $p=1/2^n$ (if there is a unique solution)

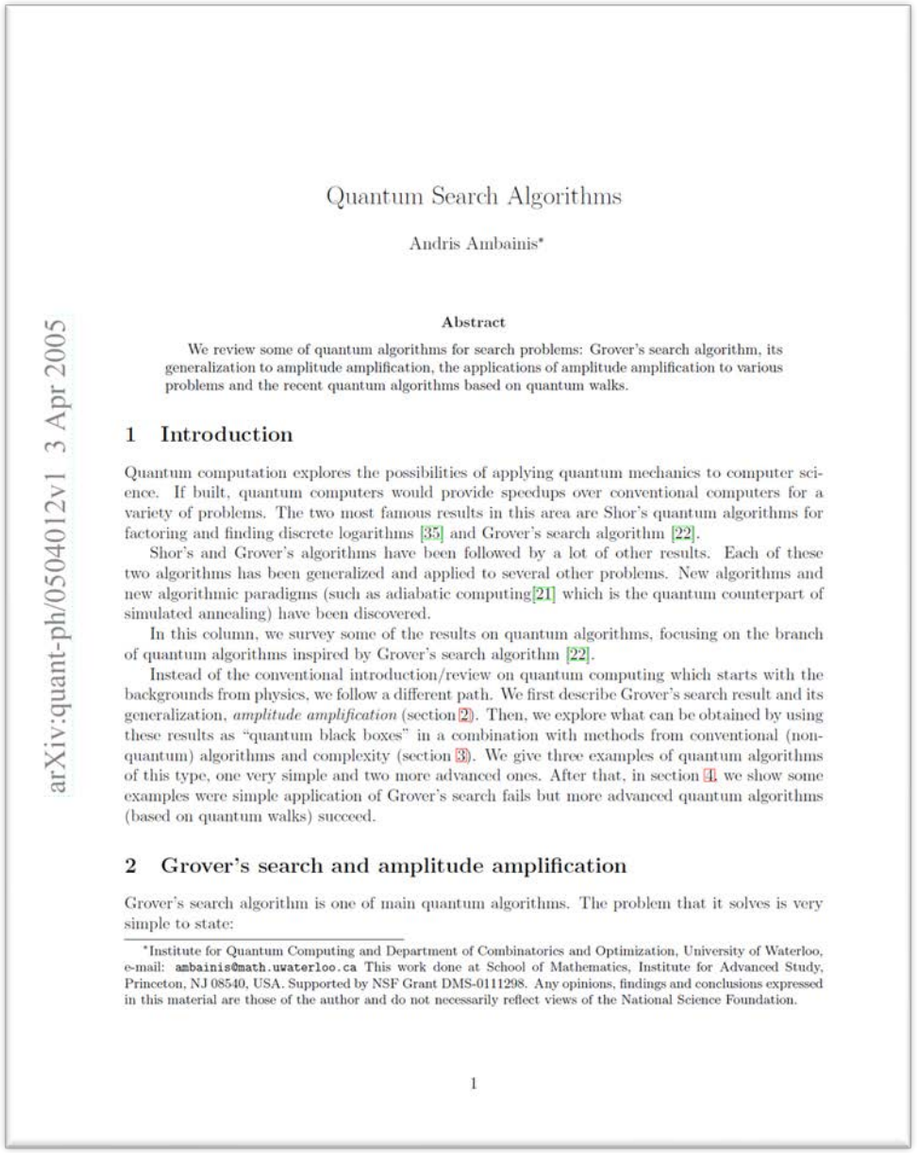
Classically we need to repeat the procedure $O(2^n)$ times

Quantumly only $O(\sqrt{2^n})$ repetitions are enough

おすすめの参考文献



第6章：量子探索



Quantum Search Algorithms

Andris Ambainis*

Abstract

We review some of quantum algorithms for search problems: Grover's search algorithm, its generalization to amplitude amplification, the applications of amplitude amplification to various problems and the recent quantum algorithms based on quantum walks.

1 Introduction

Quantum computation explores the possibilities of applying quantum mechanics to computer science. If built, quantum computers would provide speedups over conventional computers for a variety of problems. The two most famous results in this area are Shor's quantum algorithms for factoring and finding discrete logarithms [35] and Grover's search algorithm [22].

Shor's and Grover's algorithms have been followed by a lot of other results. Each of these two algorithms has been generalized and applied to several other problems. New algorithms and new algorithmic paradigms (such as adiabatic computing[21] which is the quantum counterpart of simulated annealing) have been discovered.

In this column, we survey some of the results on quantum algorithms, focusing on the branch of quantum algorithms inspired by Grover's search algorithm [22].

Instead of the conventional introduction/review on quantum computing which starts with the backgrounds from physics, we follow a different path. We first describe Grover's search result and its generalization, *amplitude amplification* (section 2). Then, we explore what can be obtained by using these results as "quantum black boxes" in a combination with methods from conventional (non-quantum) algorithms and complexity (section 3). We give three examples of quantum algorithms of this type, one very simple and two more advanced ones. After that, in section 4, we show some examples were simple application of Grover's search fails but more advanced quantum algorithms (based on quantum walks) succeed.

2 Grover's search and amplitude amplification

Grover's search algorithm is one of main quantum algorithms. The problem that it solves is very simple to state:

*Institute for Quantum Computing and Department of Combinatorics and Optimization, University of Waterloo, e-mail: ambainis@math.uwaterloo.ca This work done at School of Mathematics, Institute for Advanced Study, Princeton, NJ 08540, USA. Supported by NSF Grant DMS-0111298. Any opinions, findings and conclusions expressed in this material are those of the author and do not necessarily reflect views of the National Science Foundation.

Andris Ambainis. Quantum search algorithms. SIGACT News, 35 (2):22-35, 2004.
ArXiv: 0504012

(量子探索の応用について)

本日の予定

I. 量子探索アルゴリズム ］ 1時間

II. その他の量子アルゴリズムの簡単な紹介

- Shorの素因数アルゴリズム
- HHLアルゴリズム
- 量子ウォーク

］ 30分

III. 量子探索アルゴリズムの応用例 ］ 1時間

量子分散計算

Integer Factoring

15 = 3 x 5

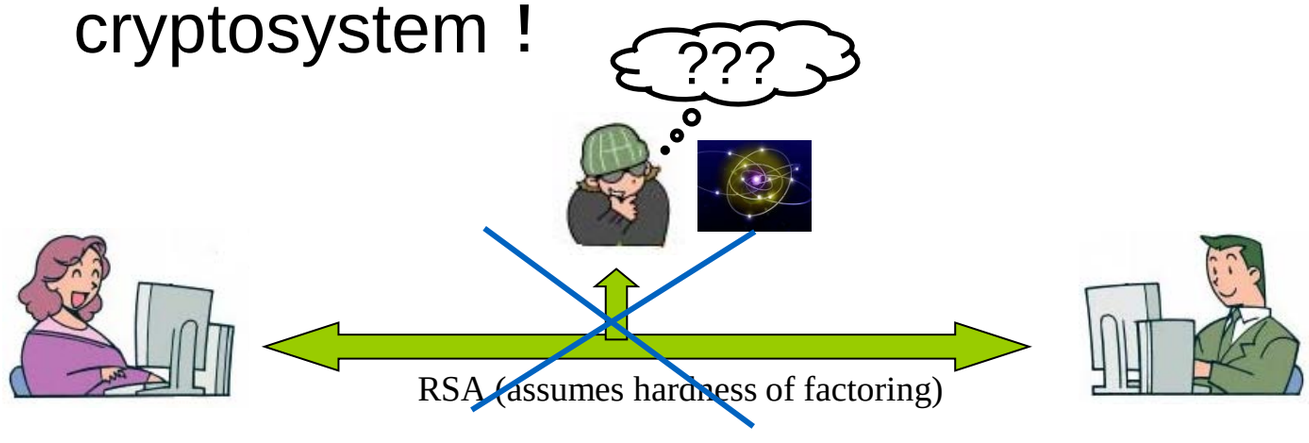
1475739525896764129 27 = 193707721 x 761838257287

✓ requires exponential time with the best known algorithms (this is the basis of the widely used RSA cryptosystem)

✓ there exists a polynomial-time quantum algorithm

Designed in 1994 by Peter Shor

→ If we can construct a quantum computer, we can break RSA cryptosystem !



Integer Factoring

15 = 3 x 5

1475739525896764129 27 = 193707721 x 761838257287

✓ requires exponential time with the best known algorithms
(this is the basis of the widely used RSA cryptosystem)

✓ there exists a polynomial-time quantum algorithm

Designed in 1994 by Peter Shor

✓ 量子Fourier変換を初めて導入
✓ 量子Fourier変換は多項式サイズの量子回路で実現できることを証明
✓ 量子Fourier変換を用いて、
　任意の群の元の位数は簡単に求められることを証明

本日の予定

I. 量子探索アルゴリズム ］ 1時間

II. その他の量子アルゴリズムの簡単な紹介

- Shorの素因数アルゴリズム
- HHLアルゴリズム
- 量子ウォーク

］ 30分

III. 量子探索アルゴリズムの応用例 ］ 1時間

量子分散計算

HHL Algorithm [Harrow, Hassidim, Lloyd 09]

The HHL Algorithm:
Quantum Algorithm for Problems from Linear Algebra

Problems Related to Matrix Multiplication

Compute the product of two $n \times n$ matrices A and B over a field $\mathbb{F}$

Determinant (DET)

Compute the determinant of an n x n matrix over a field

Inversion (INV)

Compute the inverse of an n x n invertible matrix over a field

Solution of a linear system (SYS)

Solve the system Ax=b, where A is an n x n invertible matrix over a field

In the classical time complexity setting, all these problems are equivalent to matrix multiplication (they can be solved in time $O(n^3)$ or even $O(n^{2.373})$)

HHL Algorithm for “Systems of Linear Equations”

[Harrow, Hassidim, Lloyd 09]

κ is called the condition number (条件数) of A

Input:

we assume that the non-zero entries can be accessed efficiently

- ✓ A Hermitian matrix A ∈ $\mathbb{C}^{n\times n}$ that has eigenvalues in the range [-1,-1/κ] ∪ [1/κ,1], and has ≤ s nonzero entries per row
- ✓ A vector b ∈ $\mathbb{C}^{n}$ of norm 1 given as a quantum state $|b\rangle$

Output:

write $x = A^{-1}b$ and $\bar{x} = \frac{A^{-1}b}{\|A^{-1}b\|}$

An approximation of the quantum state $|\bar{x}\rangle$

Compare with the task: solve the system of linear equations Ax = b (i.e., compute $A^{-1}b$)

HHL Algorithm for "Systems of Linear Equations"

[Harrow, Hassidim, Lloyd 09]

κ is called the condition number (条件数) of A

Input:

we assume that the non-zero entries can be accessed efficiently

- ✓ A Hermitian matrix A ∈ $\mathbb{C}^{n\times n}$ that has eigenvalues in the range [-1,-1/κ] ∪ [1/κ,1], and has ≤ s nonzero entries per row
- ✓ A vector b ∈ $\mathbb{C}^{n}$ of norm 1 given as a quantum state $|b\rangle$

Output:

write $x = A^{-1}b$ and $\bar{x} = \frac{A^{-1}b}{\|A^{-1}b\|}$

An approximation of the quantum state $|\bar{x}\rangle$

Theorem ([Harrow, Hassidim, Lloyd 09])

There is a quantum algorithm that computes an approximation of $|\bar{x}\rangle$ in time $O(\log(n)s^2\kappa^2/\varepsilon)$, where ε is the precision of the approximation.

Exponentially better than the known classical algorithms for inverting the system when $s,\kappa,\varepsilon^{-1} \ll n$

Classical conjugate gradient method: $O(ns\kappa^{1/2}\log(1/\varepsilon))$ time

HHL Algorithm for “Systems of Linear Equations”

- ✓ Exponentially better than the best classical algorithm for matrix inversion for sparse and well-conditioned matrices
- ✓ Main issue: the solution is output as a quantum state
- ✓ Possible applications of the HHL Algorithm: estimate $\langle \bar{x}|M|\bar{x}\rangle$ for some operator M

“extract statistics about the solution x”

applications for quantum machine learning ? [Wiebe, Braun, Lloyd 12],…

But actually we can often do the same classically [Tang 19]…

Output:

An approximation of the quantum state $|\bar{x}\rangle$

Theorem ([Harrow, Hassidim, Lloyd 09])

There is a quantum algorithm that computes an approximation of $|\bar{x}\rangle$ in time $O(\log(n)s^2\kappa^2/\varepsilon)$, where ε is the precision of the approximation.

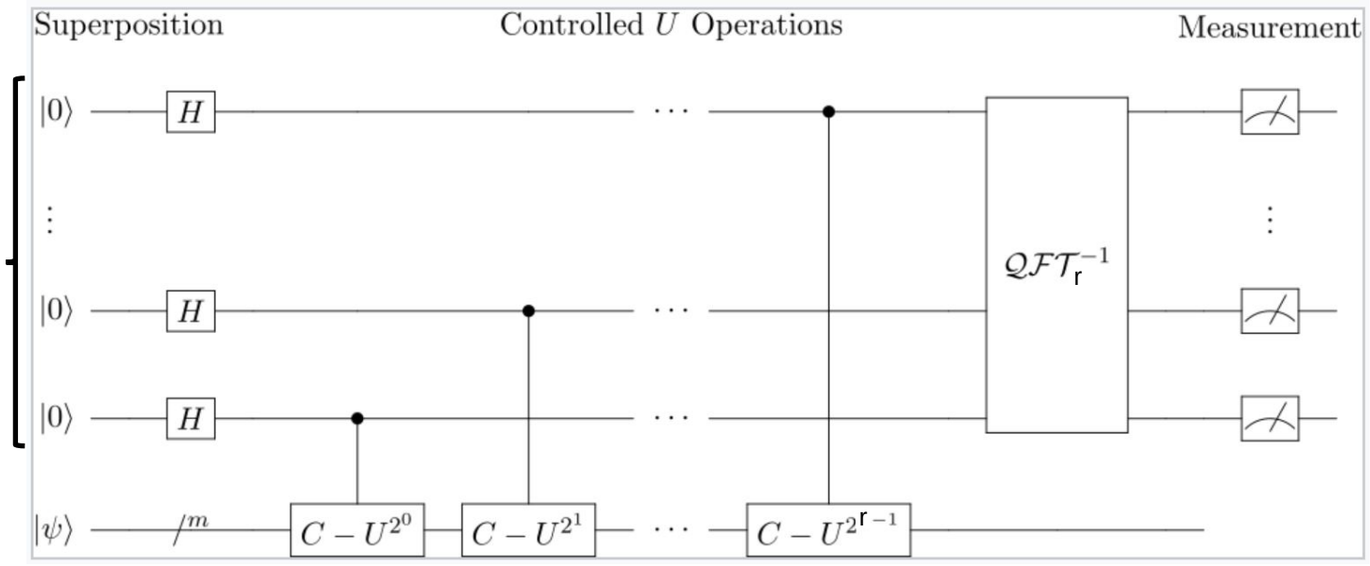
Exponentially better than the known classical algorithms for inverting the system when $s,\kappa,\varepsilon^{-1} \ll n$

Classical conjugate gradient method: $O(ns\kappa^{1/2}\log(1/\varepsilon))$ time

Phase Estimation

Given a unitary matrix U on m qubits and an eigenvector $|\psi\rangle$ such that $U|\psi\rangle = e^{2\pi i\theta}|\psi\rangle$, output a value $\tilde{\theta}$ such that $\left|\theta - \tilde{\theta}\right| < 1/2^{r}$

r qubits



This outputs a good approximation $\tilde{\theta}$ with constant probability

Complexity: roughly the complexity of applying $U^{2^{r-1}}$

Using some additional work, we can obtain a quantum unitary circuit that, for any $|\psi\rangle$ such that $U|\psi\rangle = e^{2\pi i\theta}|\psi\rangle$, maps the quantum state $|0 \ldots 0\rangle|\psi\rangle|0 \ldots 0\rangle$ to a state close to $|0 \ldots 0\rangle|\psi\rangle|\tilde{\theta}\rangle$ for some value $\tilde{\theta}$ such that $\left|\theta - \tilde{\theta}\right| < 1/2^{r}$

HHL Algorithm for "Systems of Linear Equations"

Input: [Harrow, Hassidim, Lloyd 09]

- ✓ A Hermitian matrix $A \in \mathbb{C}^{n\times n}$ that has eigenvalues in the range $[-1,-1/\kappa] \cup [1/\kappa,1]$, and has ≤ s nonzero entries per row
- ✓ A vector $b \in \mathbb{C}^n$ given as a quantum state $|b\rangle$

Output: write $x = A^{-1}b$ and $\bar{x} = \frac{A^{-1}b}{\|A^{-1}b\|}$

An approximation of the quantum state $|\bar{x}\rangle$

Write $|b\rangle = \sum_j \beta_j |u_j\rangle$

Then $|x\rangle = \frac{\beta_j}{\lambda_j}|u_j\rangle$

Let $\lambda_1, \ldots, \lambda_n$ denote the eigenvalues of A, and $u_1, \ldots, u_n$ the eigenvectors

The unitary matrix $e^{2\pi iA}$ has same eigenvectors, and eigenvalues $e^{2\pi i\lambda_1}, \ldots, e^{2\pi i\lambda_n}$

Applying phase estimation for $U = e^{2\pi iA}$ on $|0 \ldots 0\rangle|u_j\rangle|0 \ldots 0\rangle$ gives a state close to $|0 \ldots 0\rangle|u_j\rangle|\widetilde{\lambda_j}\rangle$

Applying phase estimation for $U = e^{2\pi iA}$ on $|0 \ldots 0\rangle|b\rangle|0 \ldots 0\rangle$ gives a state close to $\sum_j \beta_j |0 \ldots 0\rangle|u_j\rangle|\widetilde{\lambda_j}\rangle$

Theorem ([Harrow, Hassidim, Lloyd 09])

There is a quantum algorithm that computes an approximation of $|\bar{x}\rangle$ in time $O(\log(N)s^2\kappa^2/\varepsilon)$, where ε is the precision of the approximation.

HHL Algorithm for "Systems of Linear Equations"

Input: [Harrow, Hassidim, Lloyd 09]

✓ A Hermitian matrix $A \in \mathbb{C}^{n \times n}$ that has eigenvalues in the range
✓ s entries per row
✓ $|b\rangle$

Complexity depends crucially on the efficiency of applying (powers of) U

Using methods for Hamiltonian simulation this can be done in time $O(\log(n)s^2\kappa/\varepsilon)$

Output:

An approximation of the quantum state $|\bar{x}\rangle$

Write $|b\rangle = \sum_j \beta_j |u_j\rangle$
Then $|x\rangle = \frac{\beta_j}{\lambda_j}|u_j\rangle$

Let $\lambda_1, \ldots, \lambda_n$ denote the eigenvalues of A, and $u_1, \ldots, u_n$ the eigenvectors

The unitary matrix $e^{2\pi iA}$ has same eigenvectors, and eigenvalues $e^{2\pi i\lambda_1}, \ldots, e^{2\pi i\lambda_n}$

Applying phase estimation for $U = e^{2\pi iA}$ on $|0 \ldots 0\rangle|u_j\rangle|0 \ldots 0\rangle$ gives a state close to $|0 \ldots 0\rangle|u_j\rangle|\widetilde{\lambda_j}\rangle$

Applying phase estimation for U

apply amplitude amplification to boost the success probability
(number of iterations depends on the $\widetilde{\lambda_j}$s) ← $O(\kappa)$ iterations

$\sum_j \beta_j |0 \ldots 0\rangle|u_j\rangle|\widetilde{\lambda_j}\rangle$

can be converted to normalized $\sum_j \frac{\beta_j}{\widetilde{\lambda_j}} |0 \ldots 0\rangle|u_j\rangle|\widetilde{\lambda_j}\rangle$ with some probability depending on the $\widetilde{\lambda_j}$s

can be converted to normalized $\sum_j \frac{\beta_j}{\widetilde{\lambda_j}} |0 \ldots 0\rangle|u_j\rangle|0 \ldots 0\rangle \approx$ normalized $|0 \ldots 0\rangle|x\rangle|0 \ldots 0\rangle$ using uncomputation

$= |0 \ldots 0\rangle|\bar{x}\rangle|0 \ldots 0\rangle$

The HHL Algorithm

The HHL Algorithm:
Quantum Algorithm for Problems from Linear Algebra:

✓ The HHL algorithm

➡ Exponential “speedup” for sparse and well-conditioned matrices

What applications? Applications to machine learning?

本日の予定

I. 量子探索アルゴリズム ］ 1時間

II. その他の量子アルゴリズムの簡単な紹介

- Shorの素因数アルゴリズム
- HHLアルゴリズム
- 量子ウォーク

］ 30分

III. 量子探索アルゴリズムの応用例 ］ 1時間

量子分散計算

量子ウォーク

first paper: 1993

PHYSICAL REVIEW A VOLUME 48, NUMBER 2 AUGUST 1993

Quantum random walks

Y. Aharonov,* L. Davidovich,† and N. Zagury†
Center for Advanced Studies and Department of Physics and Astronomy, University of New Mexico, Albuquerque, New Mexico 87131
(Received 1 October 1992)

We introduce the concept of *quantum random walk*, and show that due to quantum interference effects the average path length can be much larger than the maximum allowed path in the corresponding classical random walk. A quantum-optics application is described.

PACS number(s): 03.65.Bz, 42.50.Dv, 42.52.+x

We introduce in this paper the notion of *quantum random walk,* which is the counterpart of classical random walks for particles which cannot be precisely localized due to quantum uncertainties. A classical one-dimensional random walk is defined in terms of the probabilities for a particle to make a step of a given length to the left or to the right. Quantum random walks are described instead in terms of probability amplitudes. The actual detection process is incorporated into the theory by correlating each possible step to another degree of freedom (say spin), which plays the role of a *quantum coin:* measurement of this observable will select the transition actually undergone. Interesting effects arise when there is a considerable overlap between the probability amplitudes for going left or right. In this case the average displacement of the particle can be well beyond the maximum classically allowed displacement. All these notions are easily generalized to the multidimensional case.

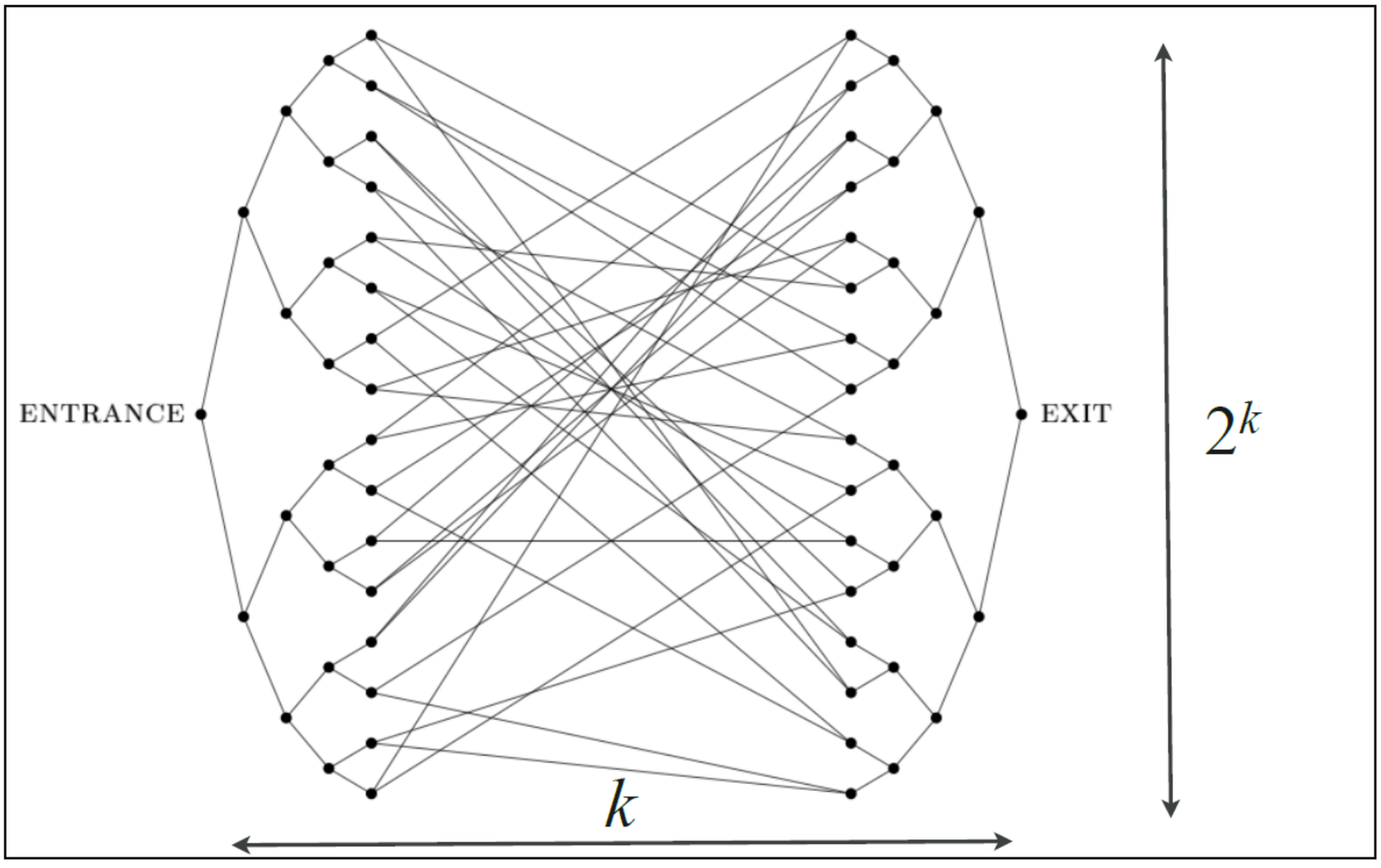
come, whether the particle would be described, after the first step, by the state $|\psi(x_0 + l)\rangle$ (if the spin is up) or by the state $|\psi(x_0 - l)\rangle$ (spin down). After measuring the spin, thus determining the new state of the particle, we reestablish the initial condition of the measurement apparatus, and let the state evolve again as described by Eq. (1). It is clear that repetition of this procedure will lead, after N steps, to an average displacement given by $\langle x\rangle = Nl(|c_+|^2 - |c_-|^2)$. These results coincide precisely with those expected from a classical random walk.

A more interesting outcome is obtained by making use of the "multisided" character of quantum coins, and considering a new pair of sides. One measures instead the spin components along a direction (θ, ϕ), where ϕ is the argument of c_-/c_+. The corresponding eigenstates are $|\theta, \phi, +\rangle = \cos(\theta/2)|+\rangle + \exp(i\phi)\sin(\theta/2)|-\rangle$ and $|\theta, \phi, -\rangle = \sin(\theta/2)|+\rangle - \exp(i\phi)\cos(\theta/2)|-\rangle$. Immediately after the measurement, if the spin is found to be

理論計算機科学者による再発見

[Childs et al. 03]

"glued graph"



A quantum walk starting in ENTRANCE reach EXIT after poly(k) steps

A number of steps exponential in k is needed for any random walk

Artificial problem

量子ウォーク型探索

SIAM J. COMPUT.
Vol. 37, No. 1, pp. 210–239

QUANTUM WALK ALGORITHM FOR ELEMENT DISTINCTNESS*

ANDRIS AMBAINIS†

Abstract. We use quantum walks to construct a new quantum algorithm for element distinctness and its generalization. For element distinctness (the problem of finding two equal items among N given items), we get an $O(N^{2/3})$ query quantum algorithm. This improves the previous $O(N^{3/4})$ quantum algorithm of Buhrman et al. [*SIAM J. Comput.*, 34 (2005), pp. 1324–1330] and matches the lower bound of Aaronson and Shi [*J. ACM*, 51 (2004), pp. 595–605]. We also give an $O(N^{k/(k+1)})$ query quantum algorithm for the generalization of element distinctness in which we have to find k equal items among N items.

Key words. quantum computing, quantum query algorithms, element distinctness

AMS subject classifications. 81P68, 68Q25, 68Q10

DOI. 10.1137/S0097539705447311

1. Introduction. Element distinctness is the following problem: Given numbers $x_1, \ldots, x_N \in [M]$, are they all distinct?

This problem has been extensively studied in both classical and quantum computing. Classically, the best way to solve element distinctness is by sorting, which requires $\Omega(N)$ queries. In the quantum setting, Buhrman et al. [14] have constructed a quantum algorithm that uses $O(N^{3/4})$ queries. Aaronson and Shi [1] have shown that any quantum algorithm requires at least $\Omega(N^{2/3})$ quantum queries.

In this paper, we give a new quantum algorithm that solves element distinctness with $O(N^{2/3})$ queries to $x_1, \ldots, x_N$. This matches the lower bound of [1, 5].

Our algorithm uses a combination of the following ideas: quantum search on graphs [2] and quantum walks [30]. While each of those ideas has been used before, the present combination is new.

We first reduce element distinctness to searching a certain graph with vertices $S \subseteq \{1, \ldots, N\}$ as vertices. The goal of the search is to find a marked vertex. Both examining the current vertex and moving to a neighboring vertex cost one time step. (This contrasts with the usual quantum search [26], where only examining the current vertex costs one time step.)

We then search this graph by quantum random walk. We start in a uniform superposition over all vertices of a graph and perform a quantum random walk with one transition rule for unmarked vertices of the graph and another transition rule for marked vertices of the graph. The result is that the amplitude gathers in the marked vertices and, after $O(N^{2/3})$ steps, the probability of measuring the marked state is a constant.

Johnsonグラフ(探索グラフ)上の
量子ウォーク

様々な応用があるが、
古典に対する高速化は高々quadratic

本日の予定

I. 量子探索アルゴリズム ］ 1時間

II. その他の量子アルゴリズムの簡単な紹介

- Shorの素因数アルゴリズム
- HHLアルゴリズム
- 量子ウォーク

］ 30分

III. 量子探索アルゴリズムの応用例 ］ 1時間

量子分散計算

量子分散計算の理論研究

- ✓ For classical computational problems (i.e., problems with classical inputs/outputs), quantum distributed algorithms have mostly been studied in the framework of 2-party communication complexity

- ✓ Relatively few results focusing on n ≫ 2 parties:

  - ➢ exact quantum algorithms for leader election on anonymous networks
    [Tani, Kobayashi, Matsumoto 2005]
  - ➢ study of quantum distributed algorithms on non-anonymous networks
    [Gavoille, Kosowski, Markiewicz 2009]
    [Elkin, Klauck, Nanongkai, Pandurangan 2014]

negative results: show impossibility of quantum distributed computing faster than classical distributed computing for many important problems (shortest paths, minimum spanning tree,…)

Question: can quantum distributed algorithms be useful?
(over non-anonymous networks)

最近の結果

Question: can quantum distributed algorithms be useful?
(over non-anonymous networks)

Two main models in distributed computing

CONGEST model
(limited bandwidth)

Quantum can be useful!
[LG, Magniez 2018]
PODC'18, QIP'19
Arxiv: 1804.02917

量子探索に基づく

LOCAL model
(unlimited bandwidth)

Quantum can be useful!
[LG, Nishimura, Rosmanis 2019]
STACS'19, TQC'19
Arxiv: 1810.10838

Classical Distributed Computing

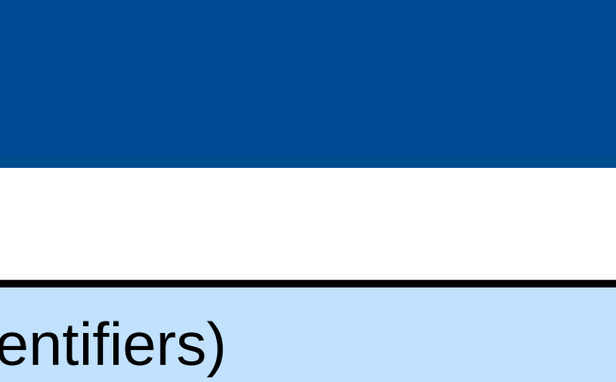
Basic setting: non-faulty, non-anonymous, synchronous

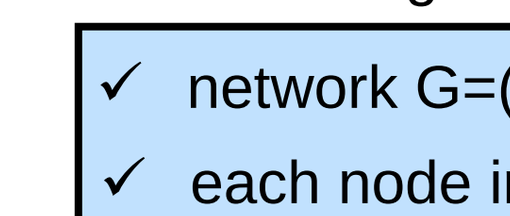
- ✓ network G=(V,E) of n nodes (all nodes have distinct identifiers)
- ✓ each node initially knows only the identifiers of all its neighbors (and knows n)
- ✓ synchronous communication between adjacent nodes:
  one message through each edge per round (in each direction)

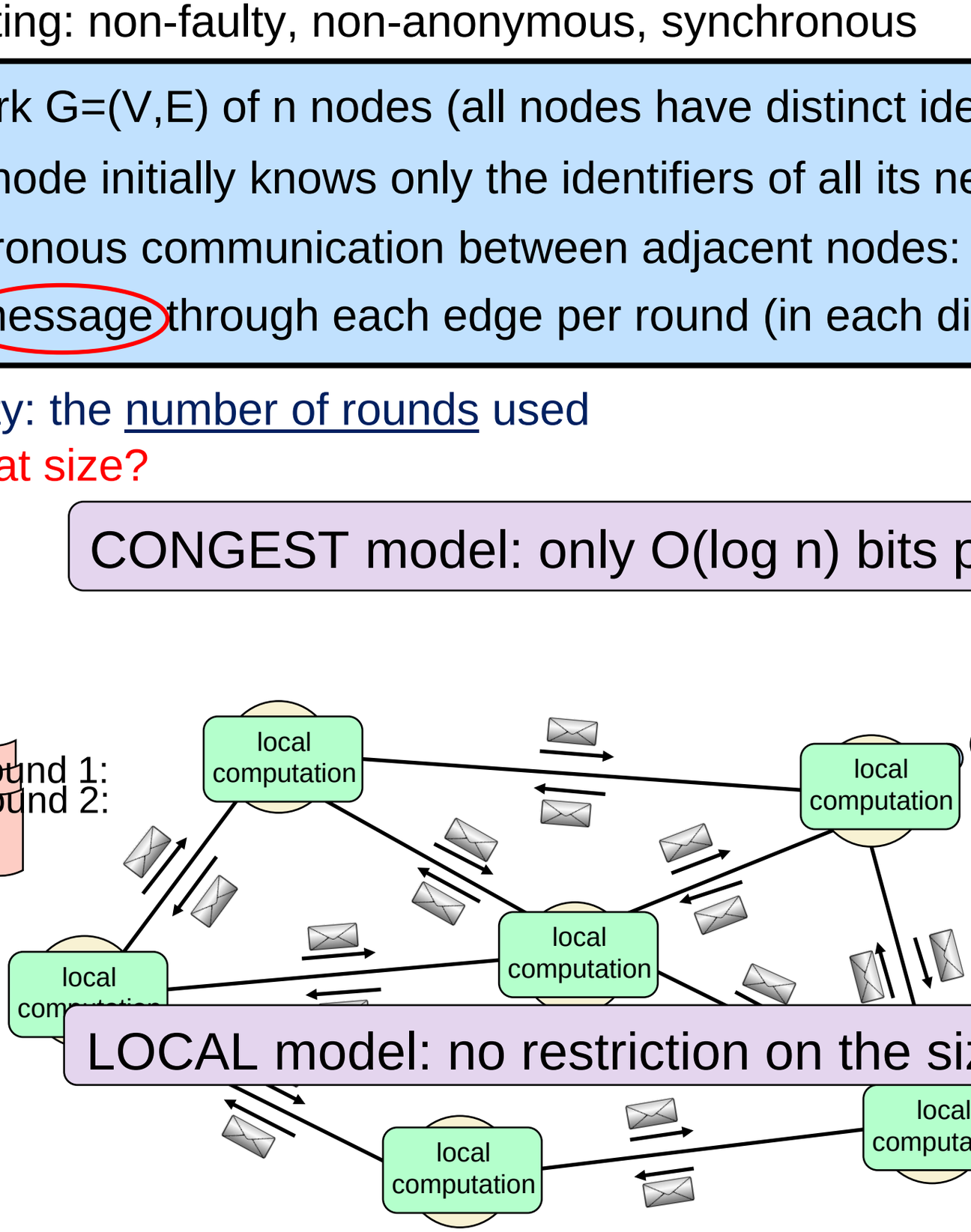
Complexity: the number of rounds used

what size?

CONGEST model: only O(log n) bits per message







LOCAL model: no restriction on the size of each message

Quantum Distributed Computing

Quantum distributed computing

Now qubits can be sent instead of bits

(no prior entanglement between nodes)

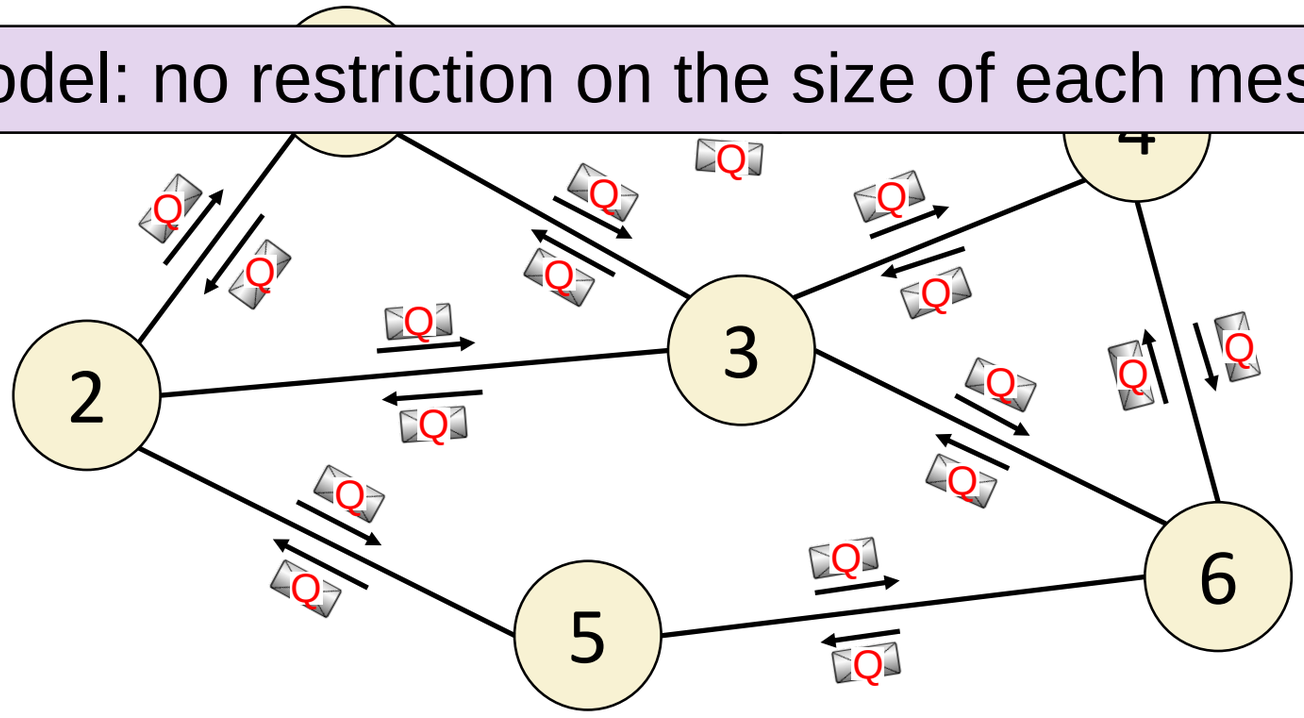
more formally:

- ✓ network G=(V,E) of n nodes (all nodes have distinct identifiers)
- ✓ each node only knows the identifiers of all its neighbors (and knows n)
- ✓ synchronous communication between adjacent nodes:
  one message of qubits through each edge per round (in each direction)
- ✓ each node is a quantum processor

CONGEST model: only O(log n) qubits per message

Complexity: the number of rounds needed for the computation

LOCAL model: no restriction on the size of each message



First Result: CONGEST model

Quantum distributed computing

Now qubits can be sent instead of bits

(no prior entanglement between nodes)

n: number of nodes of the network

CONGEST model: only O(log n) qubits per message

[LG, Magniez 18]

The diameter of the network can be computed in $\Theta(\sqrt{n})$ rounds in the quantum CONGEST model but requires $\Theta(n)$ rounds in the classical CONGEST model (when the diameter is constant)

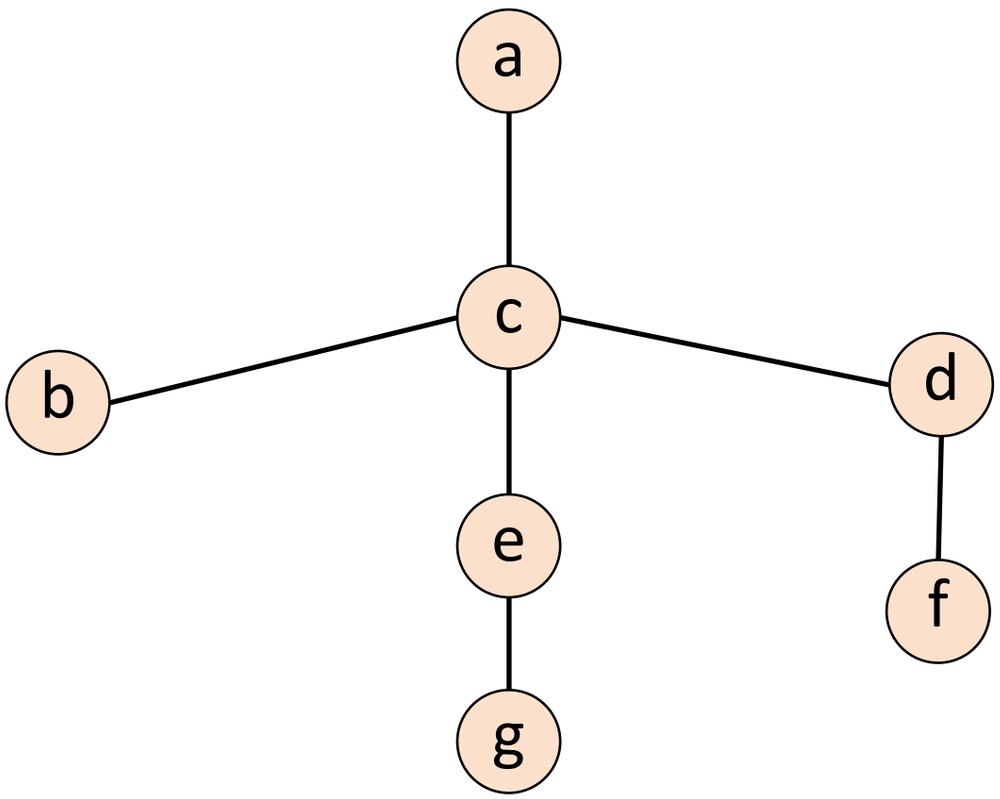
Diameter and Eccentricity

Consider an undirected and unweighted graph G = (V,E)

The diameter of the graph is the maximum distance between two nodes
(直径)

$$D = \max_{u,v \in V} \{d(u,v)\}$$

d(u,v) = distance between u and v



Diameter and Eccentricity

Consider an undirected and unweighted graph G = (V,E)

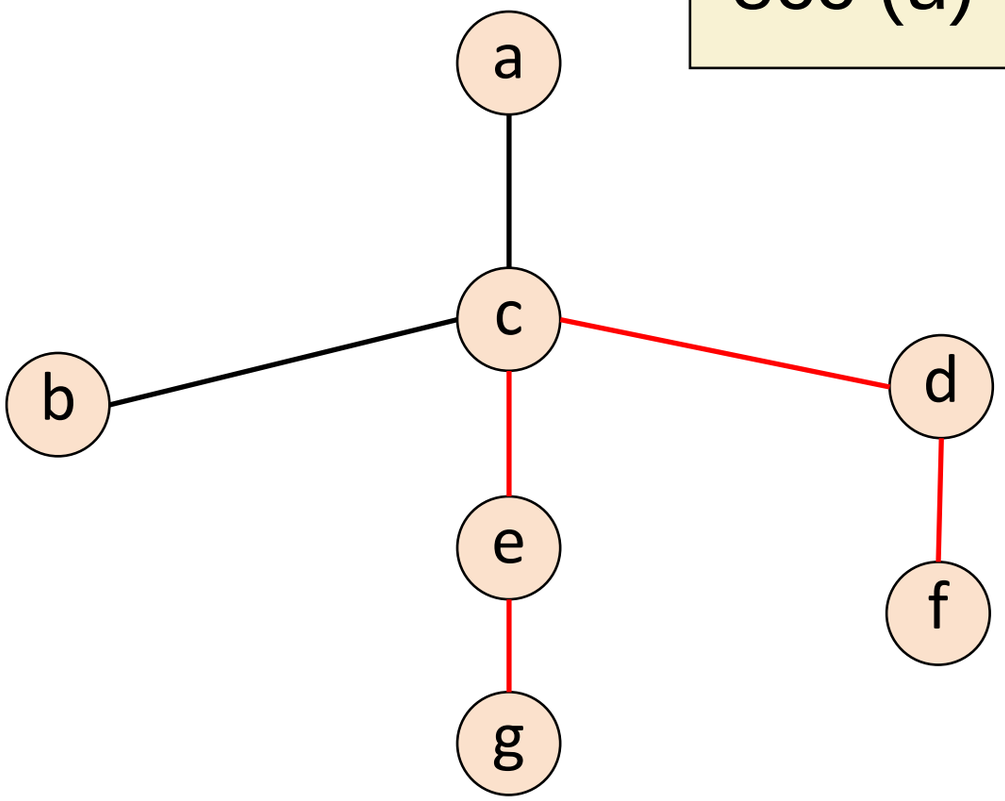
The diameter of the graph is the maximum distance between two nodes
(直径)

$$D = \max_{u,v \in V} \{d(u,v)\} = \max_{u \in V} \{ecc(u)\}$$

d(u,v) = distance between u and v

The eccentricity of a node u is defined as
(離心数)

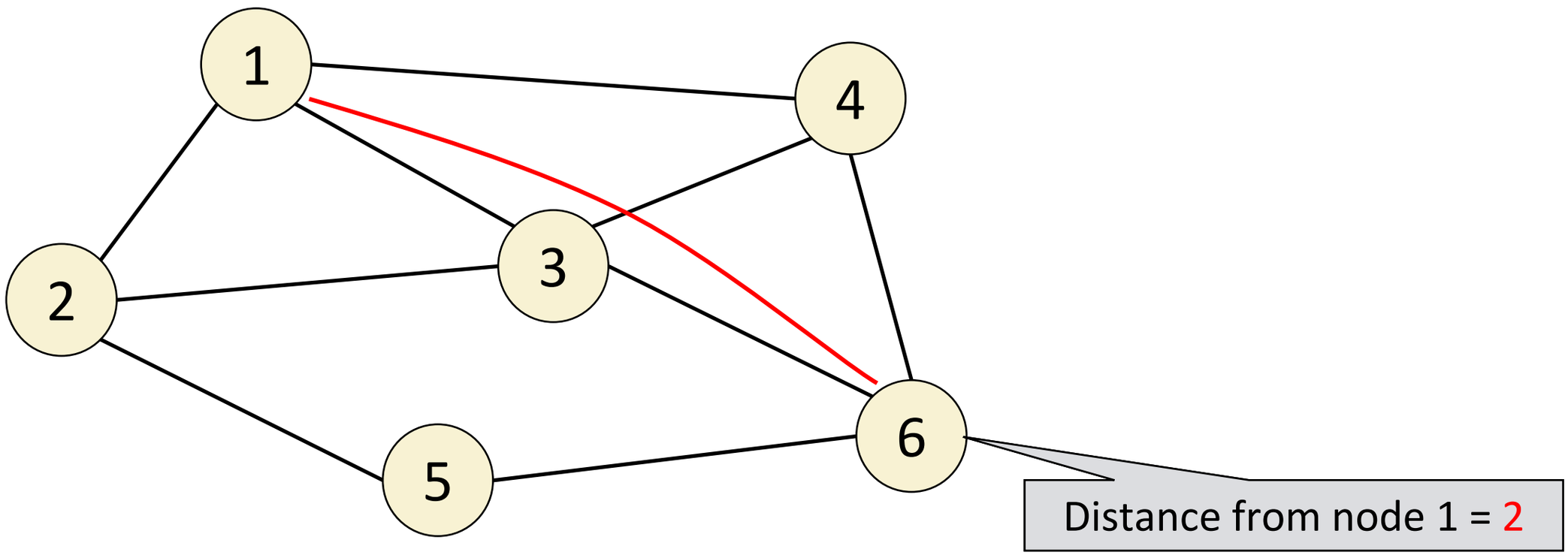
$$ecc(u) = \max_{v \in V} \{d(u,v)\}$$



ecc (a) = 3
ecc (b) = 3
ecc (c) = 2
ecc (d) = 3
ecc (e) = 3
ecc (f) = 4
ecc (g) = 4

D = 4

Classical Distributed Computing: Computing Distances



Classical Distributed Computing: Computing Distances

The distances from node 1 can be computed using the Breadth-First Search algorithm

Complexity: ecc(1) rounds (≤ D rounds)

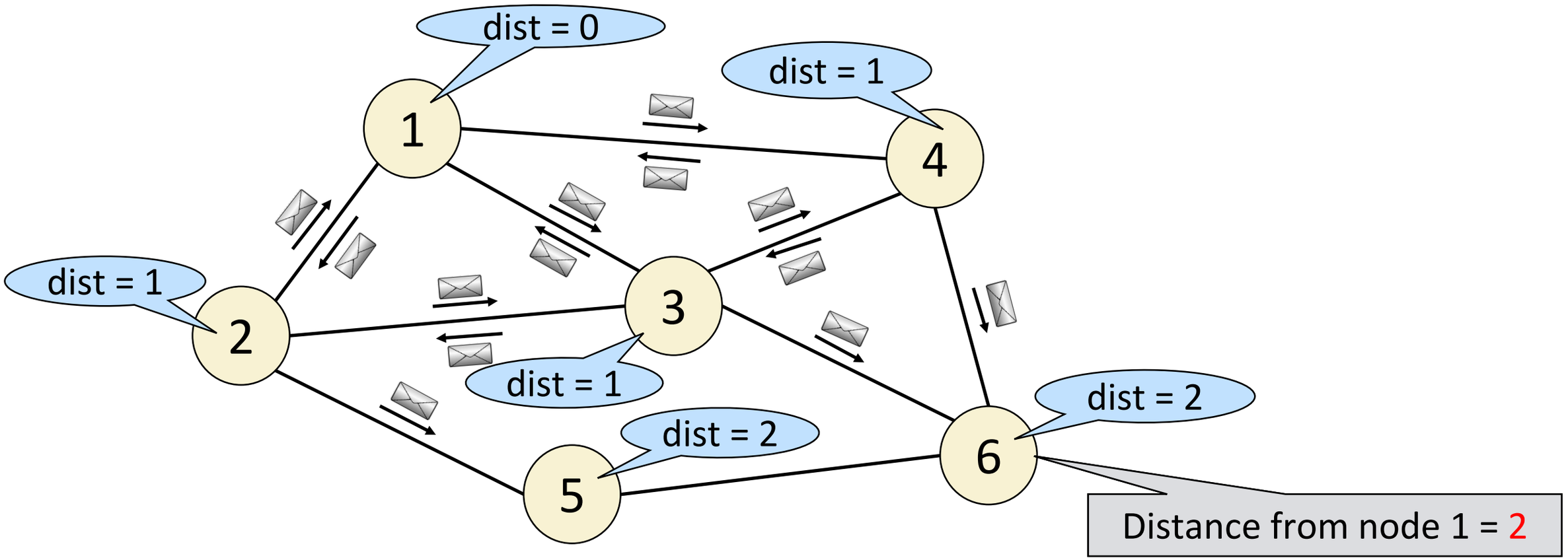
(幅優先探索アルゴリズム)

Round 1 the source node sends a message to its neighbors

at the end of Round 1: each node updates its distance
(nodes that received a message at Round 1 set “dist = 1”)

Round 2 nodes tell new knowledge to neighbors

at the end of Round 2: each node updates its distance



Classical Distributed Computing: Computing Distances

The distances from node 1 can be computed using the Breadth-First Search algorithm

Complexity: ecc(1) rounds (≤ D rounds)

eccentricity: $\text{ecc}(u) = \max_{v \in V} \{d(u,v)\}$

diameter: $D = \max_{u,v \in V} \{d(u,v)\} = \max_{u \in V} \{\text{ecc}(u)\}$

In classical distributed computing (CONGEST model):

- ✓ for any fixed node u, the eccentricity ecc(u) can be computed in O(D) rounds by the Breadth-First Search algorithm (starting at u)
- ✓ but computing the diameter (i.e., the maximum eccentricity) requires Θ(n) rounds even for constant D

  [Frischknecht+12, Holzer+12, Peleg+12, Abboud+16]

We show that we can do better in the quantum setting

Computation of the Diameter in the CONGEST model

main result: sublinear-round quantum computation of the diameter whenever D=o(n)
(our algorithm uses $O((\log n)^2)$ qubits of quantum memory per node)

first gap between classical and quantum in the CONGEST model for a major problem of interest to the distributed computing community

	Classical	Quantum (our results)
Exact computation (upper bounds)	$O(n)$ [Holzer+12, Peleg+12]	$O(\sqrt{nD})$
Exact computation (lower bounds)	$\widetilde{\Omega}(n)$ [Frischknecht+12]	$\widetilde{\Omega}(\sqrt{nD})$ [conditional]
3/2-approximation (upper bounds)	$O(\sqrt{n}+D)$ [Lenzen+13, Holzer+14]	$O(\sqrt[3]{nD}+D)$
(3/2-ε)-approximation (lower bounds)	$\widetilde{\Omega}(n)$ [Holzer+12, Abboud+16]	$\widetilde{\Omega}(\sqrt{n}+D)$ [unconditional]

number of rounds needed to compute the diameter (n: number of nodes, D: diameter)

the tilde notation removes polylog(n) factors

condition: holds for quantum distributed algorithms using only polylog(n) qubits of memory per node

Our Upper Bound

main result: sublinear-round quantum computation of the diameter whenever D=o(n)
(our algorithm uses $O((\log n)^2)$ qubits of quantum memory per node)

first gap between classical and quantum in the CONGEST model for a major problem of interest to the distributed computing community

	Classical	Quantum (our results)
Exact computation (upper bounds)	$O(n)$ [Holzer+12, Peleg+12]	$O(\sqrt{nD})$

number of rounds needed to compute the diameter (n: number of nodes, D: diameter)

Quantum Distributed Computation of the Diameter

Computation of the diameter (decision version)

Given an integer d, decide if diameter ≥ d

there is a vertex u such that ecc (u) ≥ d

This is a search problem
Idea: try to use Grover search

Define the function f: V → {0,1} such that f(u) = $\begin{cases} 1 \text{ if ecc (u)} \geq d \\ 0 \text{ otherwise} \end{cases}$

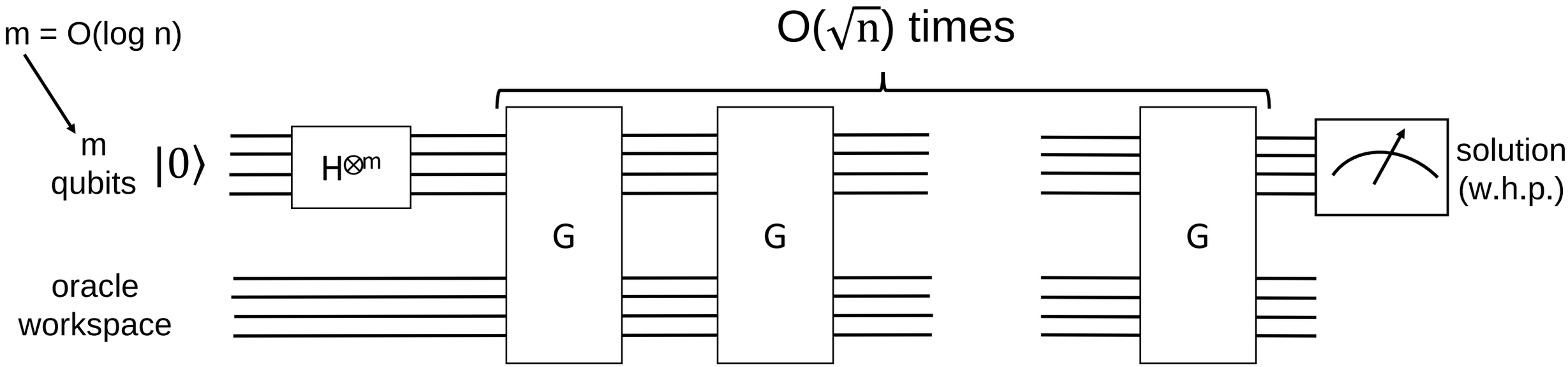
Goal: find u such that f(u) = 1 (or report that no such vertex exists)

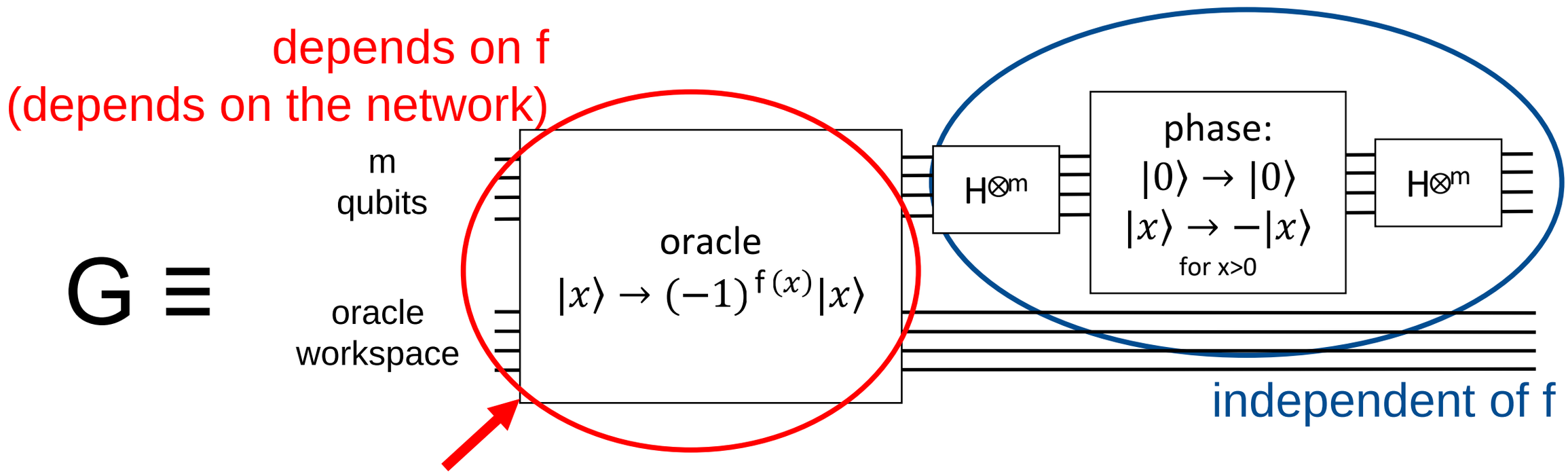
There is a quantum algorithm for this search problem using $O(\sqrt{n})$ calls to a black box evaluating f

Quantum search [Grover 96]

n = |V| (number of nodes)

Recap: Grover Algorithm





To implement the oracle, the leader node needs to communicate with the other nodes

Total number of rounds of communication = $O(\sqrt{n}$ x number of rounds to implement the oracle)

Recap: Grover Algorithm

can be done locally
(i.e., without communication)

$m = O(\log n)$

$O(\sqrt{n})$ times

can be done locally
(i.e., without communication)

m qubits

$|0\rangle$

$H^{\otimes m}$

G

G

G

solution (w.h.p.)

oracle workspace

We compute the diameter by implementing this circuit in the distributed setting:

One arbitrary node (the "leader") will implement this circuit

depends
(depends on the network)

$G \equiv$

m qubits

oracle workspace

oracle
$|x\rangle \rightarrow (-1)^{f(x)}|x\rangle$

$H^{\otimes m}$

phase:
$|0\rangle \rightarrow |0\rangle$
$|x\rangle \rightarrow -|x\rangle$
for x>0

$H^{\otimes m}$

independent of f

To implement the oracle, the leader node needs to communicate with the other nodes

can be done locally

Total number of rounds of communication = $O(\sqrt{n}$ x number of rounds to implement the oracle)

Implementatic

$$\sum_{u\in V}\alpha_u|u\rangle|0\rangle$$

Node a introduces 1 register $\sum_{u\in V}\alpha_u|u\rangle_{\mathrm{a}}|0\rangle$

Node a applies CNOTs $\sum_{u\in V}\alpha_u|u\rangle_{\mathrm{a}}|u\rangle$

Node a sends the second register to c $\sum_{u\in V}\alpha_u|u\rangle_{\mathrm{a}}|u\rangle_{\mathrm{c}}$

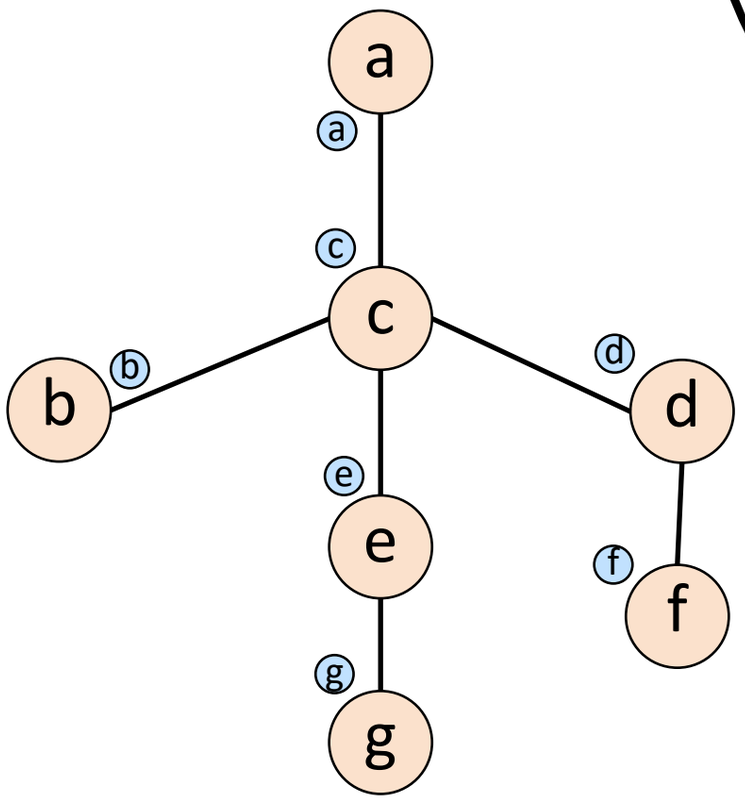
Node c introduces 3 registers $\sum_{u\in V}\alpha_u|u\rangle_{\mathrm{a}}|u\rangle_{\mathrm{c}}|0\rangle\ |0\rangle\ |0\rangle$

Node c applies CNOTs $\sum_{u\in V}\alpha_u|u\rangle_{\mathrm{a}}|u\rangle_{\mathrm{c}}|u\rangle\ |u\rangle\ |u\rangle$

Node c sends the registers to b,e,d $\sum_{u\in V}\alpha_u|u\rangle_{\mathrm{a}}|u\rangle_{\mathrm{c}}|u\rangle_{\mathrm{b}}|u\rangle_{\mathrm{e}}|u\rangle_{\mathrm{d}}$

……

Example:
V={a,b,c,d,e,f,g}
here leader = node a



Initially node a owns $\sum_{u\in V}\alpha_u|u\rangle_{\mathrm{a}}$

1. "Broadcast" this state, which gives [ecc(a) ≤ D rounds]

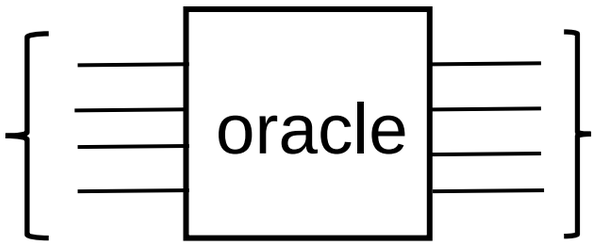
$$\sum_{u\in V}\alpha_u|u\rangle_{\mathrm{a}}|u\rangle_{\mathrm{b}}|u\rangle_{\mathrm{c}}|u\rangle_{\mathrm{d}}|u\rangle_{\mathrm{e}}|u\rangle_{\mathrm{f}}|u\rangle_{\mathrm{g}}$$

2. The nodes implement the classical protocol [O(D) rounds] for computing the eccentricity of u, which gives

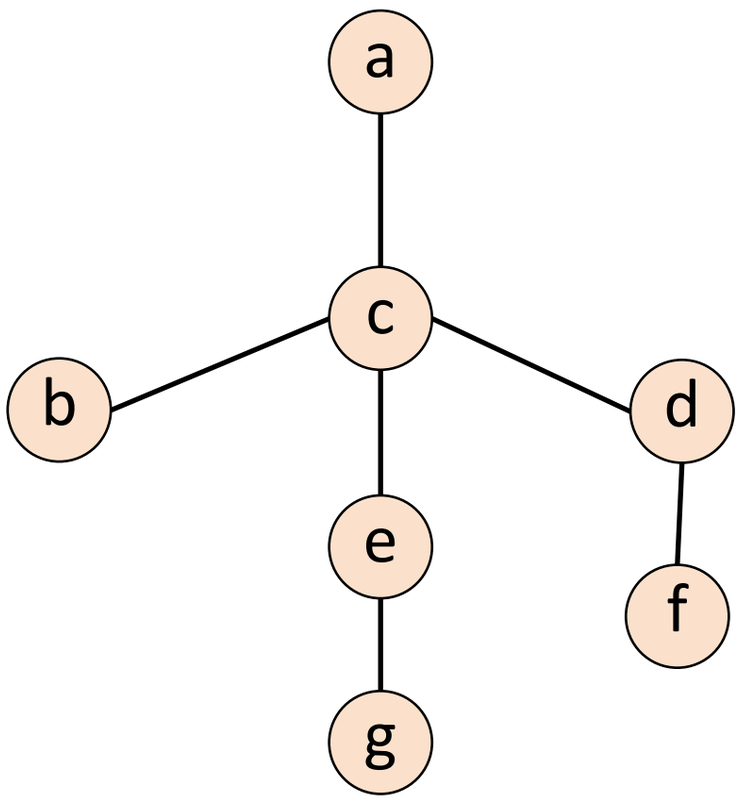
$$\sum_{u\in V}\alpha_u|u\rangle_{\mathrm{a}}|u\rangle_{\mathrm{b}}|u\rangle_{\mathrm{c}}|u\rangle_{\mathrm{d}}|u\rangle_{\mathrm{e}}|u\rangle_{\mathrm{f}}|u\rangle_{\mathrm{g}}|ecc(u)\rangle_{\mathrm{a}}$$

3. The nodes revert Step 1 [ecc(a) ≤ D rounds]

Implementation of the Oracle in O(D) rounds

$$\sum_{u\in V}\alpha_u|u\rangle|0\rangle$$  $$\sum_{u\in V}\alpha_u|u\rangle|ecc(u)\rangle$$

Example:
V={a,b,c,d,e,f,g}
here leader = node a



Initially node a owns $\sum_{u\in V}\alpha_u|u\rangle_{\mathrm{a}}$

1. "Broadcast" this state, which gives [ecc(a) ≤ D rounds]

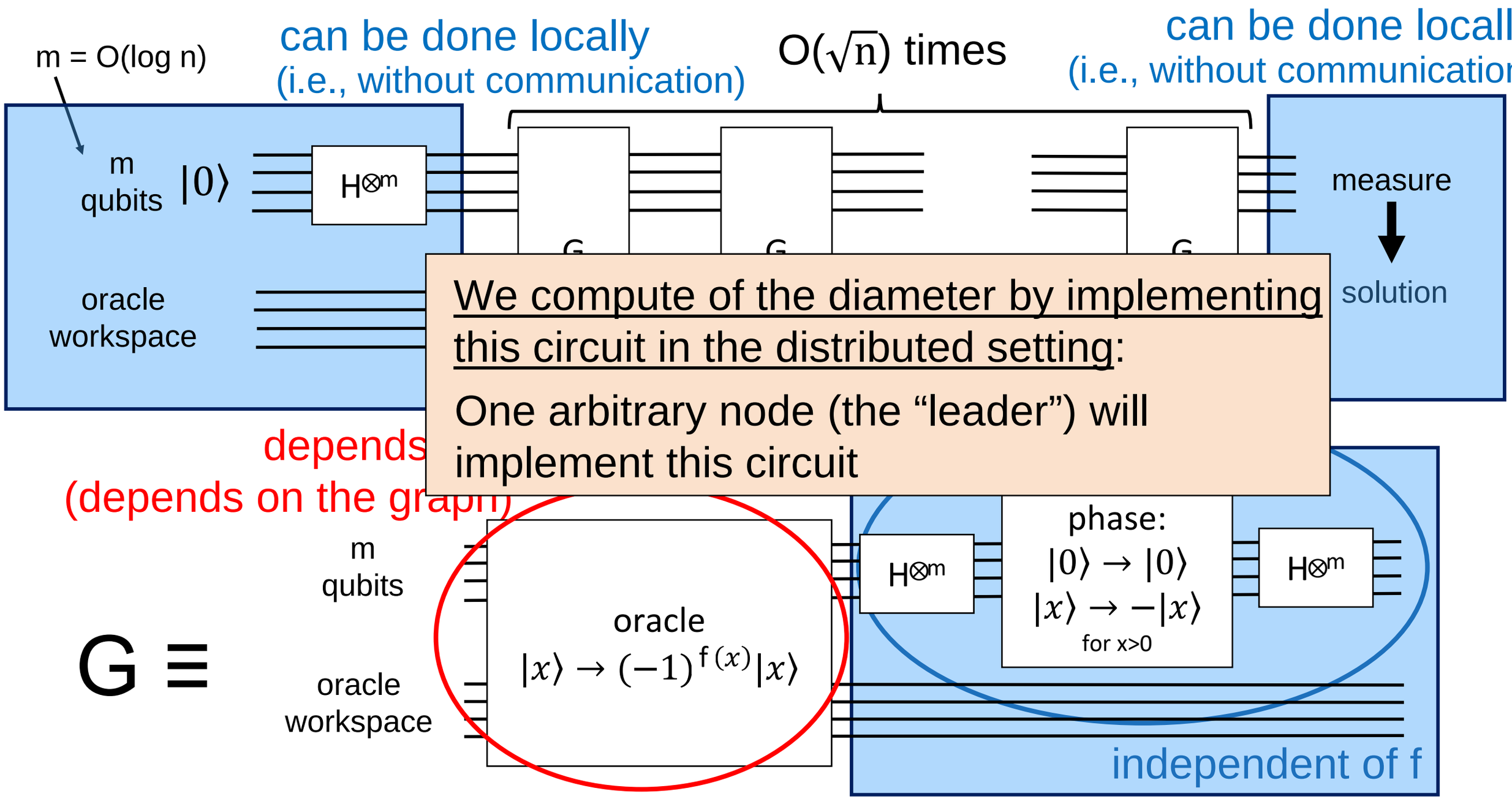
$$\sum_{u\in V}\alpha_u|u\rangle_{\mathrm{a}}|u\rangle_{\mathrm{b}}|u\rangle_{\mathrm{c}}|u\rangle_{\mathrm{d}}|u\rangle_{\mathrm{e}}|u\rangle_{\mathrm{f}}|u\rangle_{\mathrm{g}}$$

2. The nodes implement the classical protocol [O(D) rounds] for computing the eccentricity of u, which gives

$$\sum_{u\in V}\alpha_u|u\rangle_{\mathrm{a}}|u\rangle_{\mathrm{b}}|u\rangle_{\mathrm{c}}|u\rangle_{\mathrm{d}}|u\rangle_{\mathrm{e}}|u\rangle_{\mathrm{f}}|u\rangle_{\mathrm{g}}|ecc(u)\rangle_{\mathrm{a}}$$

3. The nodes revert Step 1 [ecc(a) ≤ D rounds]

Usual Grover Algorithm



To implement the oracle, the leader node needs to communicate with the other nodes

Total number of rounds of communication = O(√n x number of rounds to implement the oracle)
= O(√n x D)

Quantum Distributed Computation of the Diameter: Summary

Classically in O(D) rounds it is possible to simultaneously compute the eccentricities of D vertices [Peleg+12]

Thus we can instead do a Grover search over groups of D vertices (there are n/D groups) in

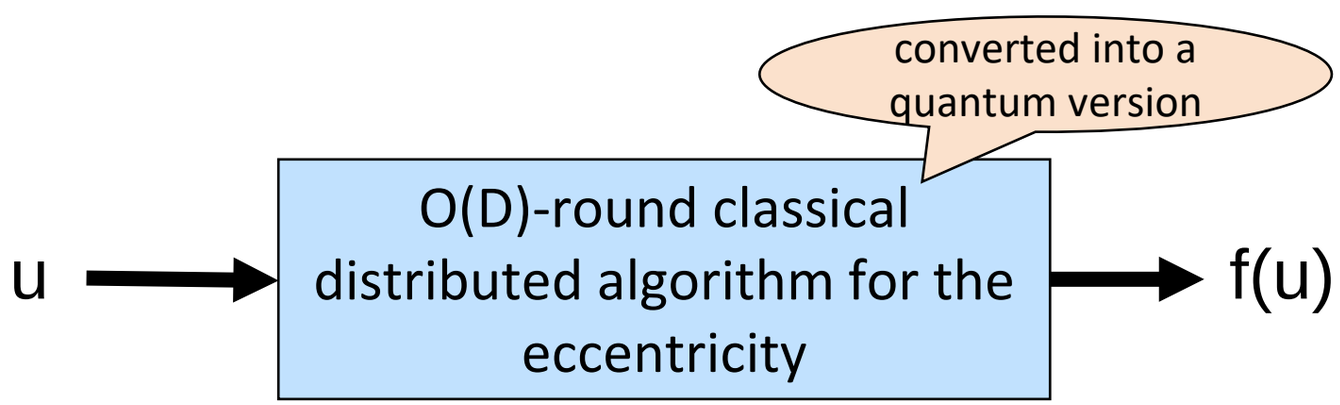
$$O(\sqrt{n/D} \times \mathrm{D}) = O(\sqrt{nD}) \text{ rounds}$$

Quantum distributed algorithm computing the diameter

- ✓ The network elects a leader
- ✓ The leader locally implements Grover algorithm. Each call to the black box is implemented by using the standard O(D)-round classical algorithm computing the eccentricity.

Complexity: $O(\sqrt{n} \times \mathrm{D})$ rounds

With further work, the complexity can be reduced to $O(\sqrt{nD})$ rounds



Our Upper Bound

main result: sublinear-round quantum computation of the diameter whenever D=o(n)
(our algorithm uses $O((\log n)^2)$ qubits of quantum memory per node)

first gap between classical and quantum in the CONGEST model for a major problem of interest to the distributed computing community

	Classical	Quantum (our results)
Exact computation (upper bounds)	$O(n)$ [Holzer+12, Peleg+12]	$O(\sqrt{nD})$

number of rounds needed to compute the diameter (n: number of nodes, D: diameter)

The Lower Bounds

	Classical	Quantum (our results)
Exact computation (lower bounds)	$\widetilde{\Omega}(n)$ [Frischknecht+12]	$\widetilde{\Omega}(\sqrt{n}+D)$ [unconditional] $\widetilde{\Omega}(\sqrt{nD})$ [conditional]

via two-party communication complexity of the disjointness function (DISJ)

classical lower bound

- ✓ reduce DISJ to the distributed computation of diameter [Frischknecht+12]
- ✓ the (two-party) communication complexity of $DISJ_n$ is Ω(n) bits [Kalyanasundaram+92]

unconditional quantum lower bound

- ✓ same reduction from DISJ to the distributed computation of diameter
- ✓ the (two-party) communication complexity of $DISJ_n$ is $\Omega(\sqrt{n})$ qubits [Razborov03]

remark: D is an obvious lower bound

conditional quantum lower bound

- ✓ Claim: if the quantum distributed algorithm for diameter uses few quantum memory per node, then the reduction can be adjusted to give a two-party protocol for DISJ using few messages (idea: send communication in batches)
- ✓ the (two-party) r-message quantum communication complexity of $DISJ_n$ is $\Omega(\ n/r + r\ )$ qubits [Braverman+15]

Summary

main result: sublinear-round quantum computation of the diameter in the CONGEST model (when D is small enough)

	Classical	Quantum (our results)
Exact computation (upper bounds)	$O(n)$ [Holzer+12, Peleg+12]	$O(\sqrt{nD})$
Exact computation (lower bounds)	$\tilde{\Omega}(n)$ [Frischknecht+12]	$\tilde{\Omega}(\sqrt{n}+D)$ [unconditional] $\tilde{\Omega}(\sqrt{nD})$ [conditional]

number of rounds needed to compute the diameter (n: number of nodes, D: diameter)

Interesting Research Direction

main result: sublinear-round quantum computation of the diameter in the CONGEST model (when D is small enough)

	Classical	Quantum (our results)
Exact computation (upper bounds)	$O(n)$ [Holzer+12, Peleg+12]	$O(\sqrt{nD})$
Exact computation (lower bounds)	$\widetilde{\Omega}(n)$ [Frischknecht+12]	$\widetilde{\Omega}(\sqrt{n}+D)$ [unconditional] $\widetilde{\Omega}(\sqrt{nD})$ [conditional]

Our upper bound is obtained by showing how to implement quantum search in a distributed setting

more generally, we give a generic framework
for distributed quantum optimization (see paper)

✓ Research Direction: find other applications of our technique

Quantum Distributed Computing

Quantum distributed computing

Now qubits can be sent instead of bits

(no prior entanglement between nodes)

n: number of nodes of the network

CONGEST model: only O(log n) qubits per message

rounds in
s in the
ant)

[LG, N

LO

Whereas the time distinction between $\mathcal{LOCAL}^+\mathcal{S}$ and $\mathcal{LOCAL}^+\mathcal{E}$ given by Theorem 1 is remarkable (since it considers the feasibility of solving problems, or when discussing connected graphs, a speed-up from $\Omega(n)$ to 0 communication rounds), the situation is less clear between $\mathcal{LOCAL}^+\mathcal{Q}$ and $\mathcal{LOCAL}$. Although a speed-up factor of 2 as expressed by Proposition 2 looks like a natural limit, the authors know of no conclusive arguments to show that it cannot be increased further.

[Gavoille et al. 09]

There is a compu
quantum LOCAL

e

unbounded amount of quantum communication
vs.
unbounded amount of classical communication

[LG, Rosmanis, Nishimura 18]

There is a computational problem that can be solved in 2 rounds in the quantum LOCAL model but requires Ω(n) rounds classically.

Superiority of the Quantum LOCAL model

Also used in some of the recent results on quantum shallow circuits [Bravyi, Gosset, König 18]

We use a construction from [Barrett, Caves, Eastin, Elliot, Pironio 07]

[LG, Rosmanis, Nishimura 18]

There is a computational problem that can be solved in 2 rounds in the quantum LOCAL model but requires Ω(n) rounds classically.

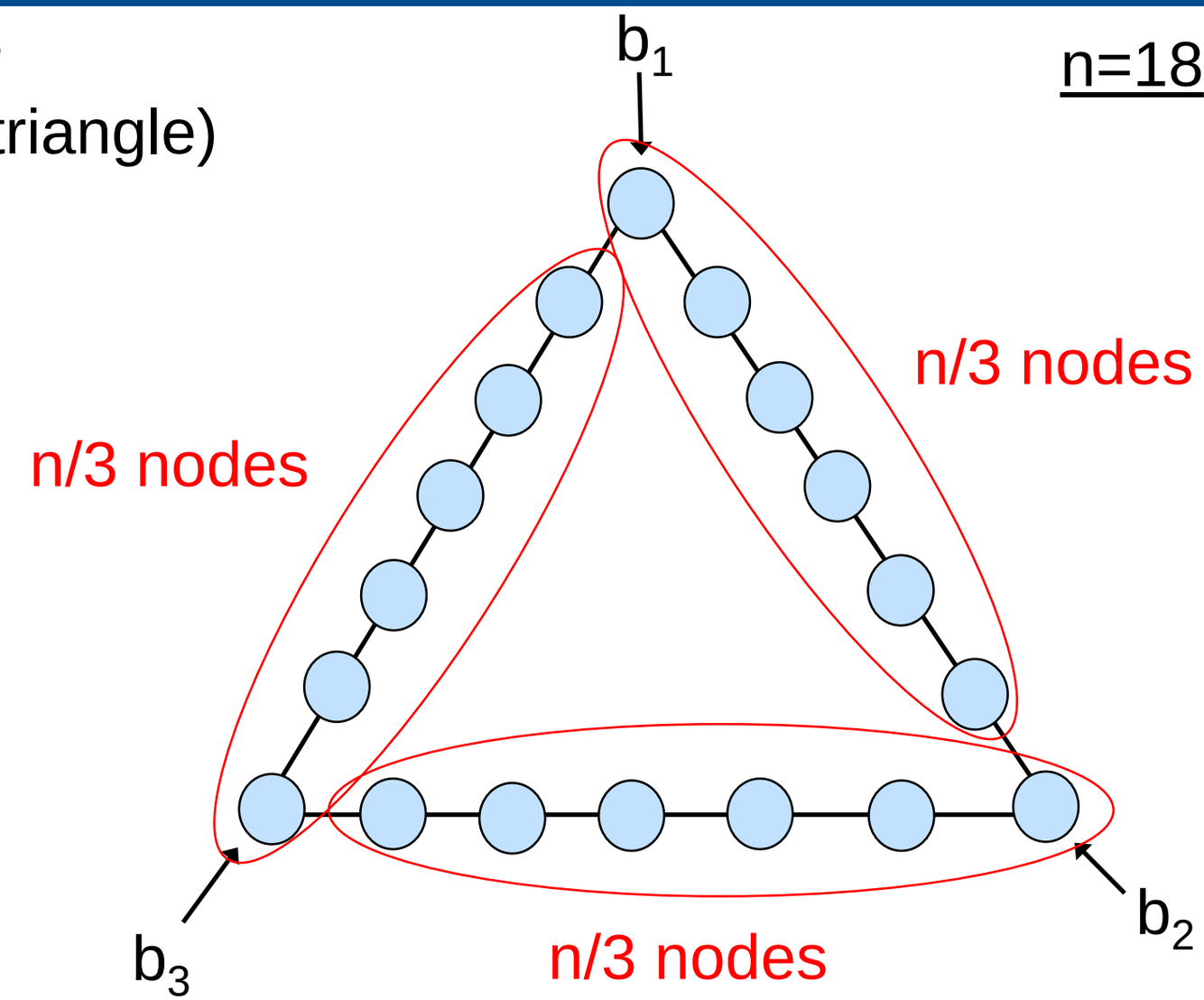
Superiority of the Quantum LOCAL model

multiple of 3

Consider a ring of size n (seen as a triangle)

Each “corner” gets a bit as input

Each node will output one bit



We use a construction from [Barrett, Caves, Eastin, Elliot, Pironio 07]

[LG, Rosmanis, Nishimura 18] ⇨ There is a computational problem that can be solved in 2 rounds in the quantum LOCAL model but requires Ω(n) rounds classically.

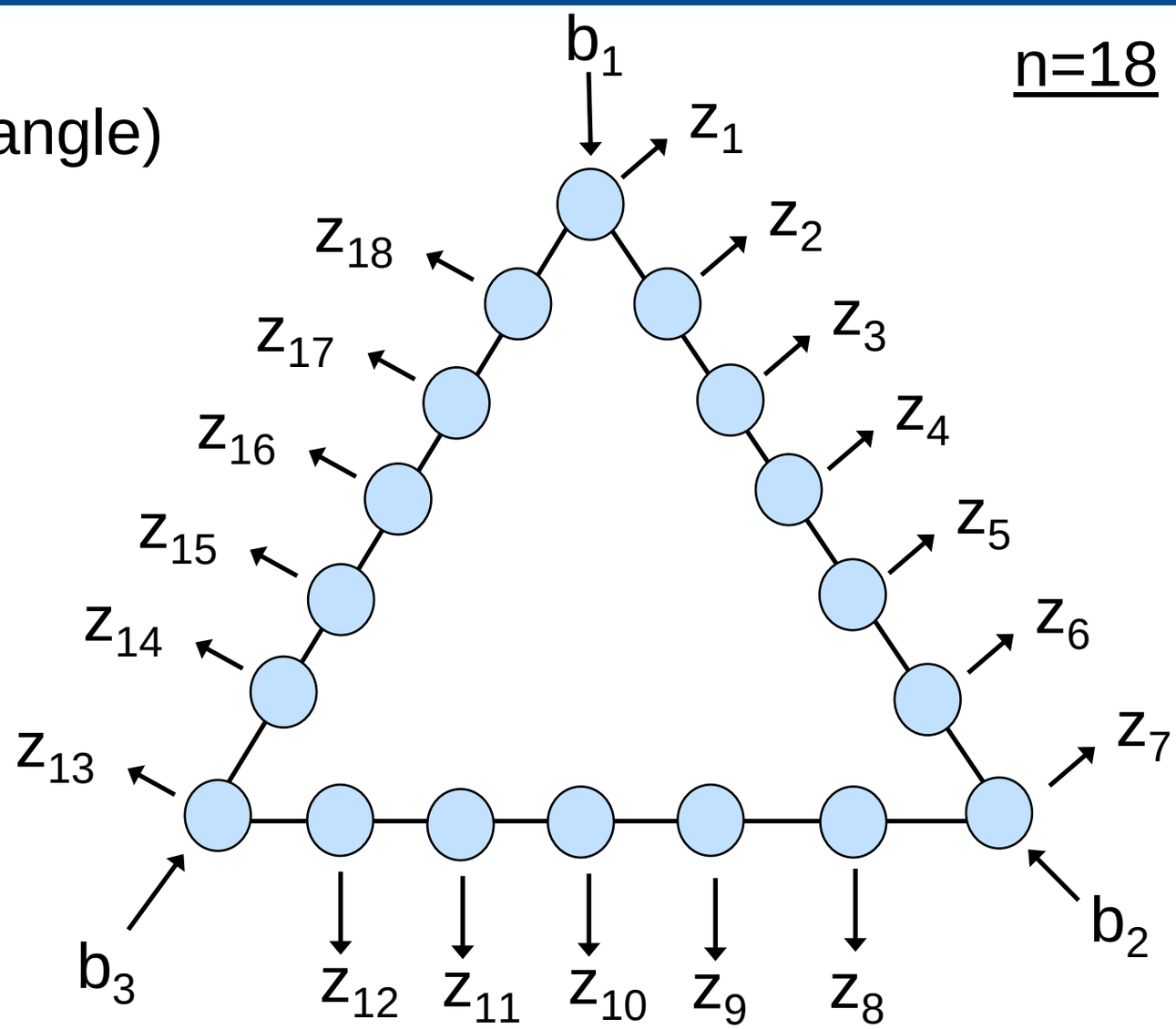
Superiority of the Quantum LOCAL model

multiple of 3

Consider a ring of size n (seen as a triangle)

Each “corner” gets a bit as input

Each node will output one bit



Superiority of the Quantum LOCAL model

multiple of 3

Consider a ring of size n (seen as a triangle)

Each “corner” gets a bit as input

Each node will output one bit

Define the following four bits:

$m_R = z_2 \oplus z_4 \oplus z_6$

(parity of the outputs of the nodes of even index on the right)

$m_B = z_8 \oplus z_{10} \oplus z_{12}$

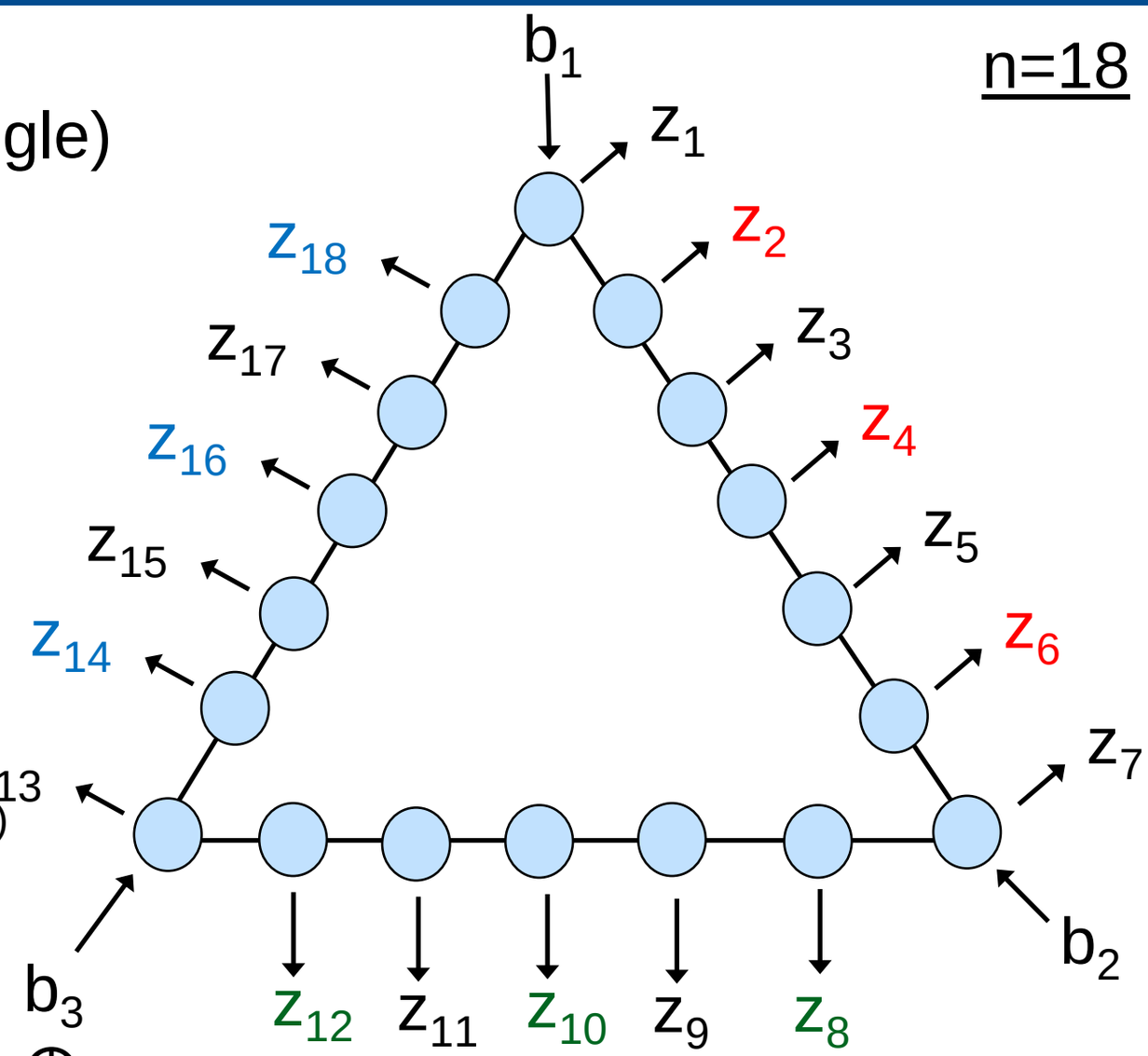
(parity of the outputs of the nodes of even index on the bottom)

$m_L = z_{14} \oplus z_{16} \oplus z_{18}$

(parity of the outputs of the nodes of even index on the left)

$m_{odd} = z_1 \oplus z_3 \oplus z_5 \oplus z_7 \oplus z_9 \oplus z_{11} \oplus z_{13} \oplus z_{15} \oplus z_{17}$

(parity of the outputs of all the nodes of odd index)



Superiority of the Quantum LOCAL model

Consider a ring of size n (seen as a triangle) ← multiple of 3

Each “corner” gets a bit as input

Each node will output one bit

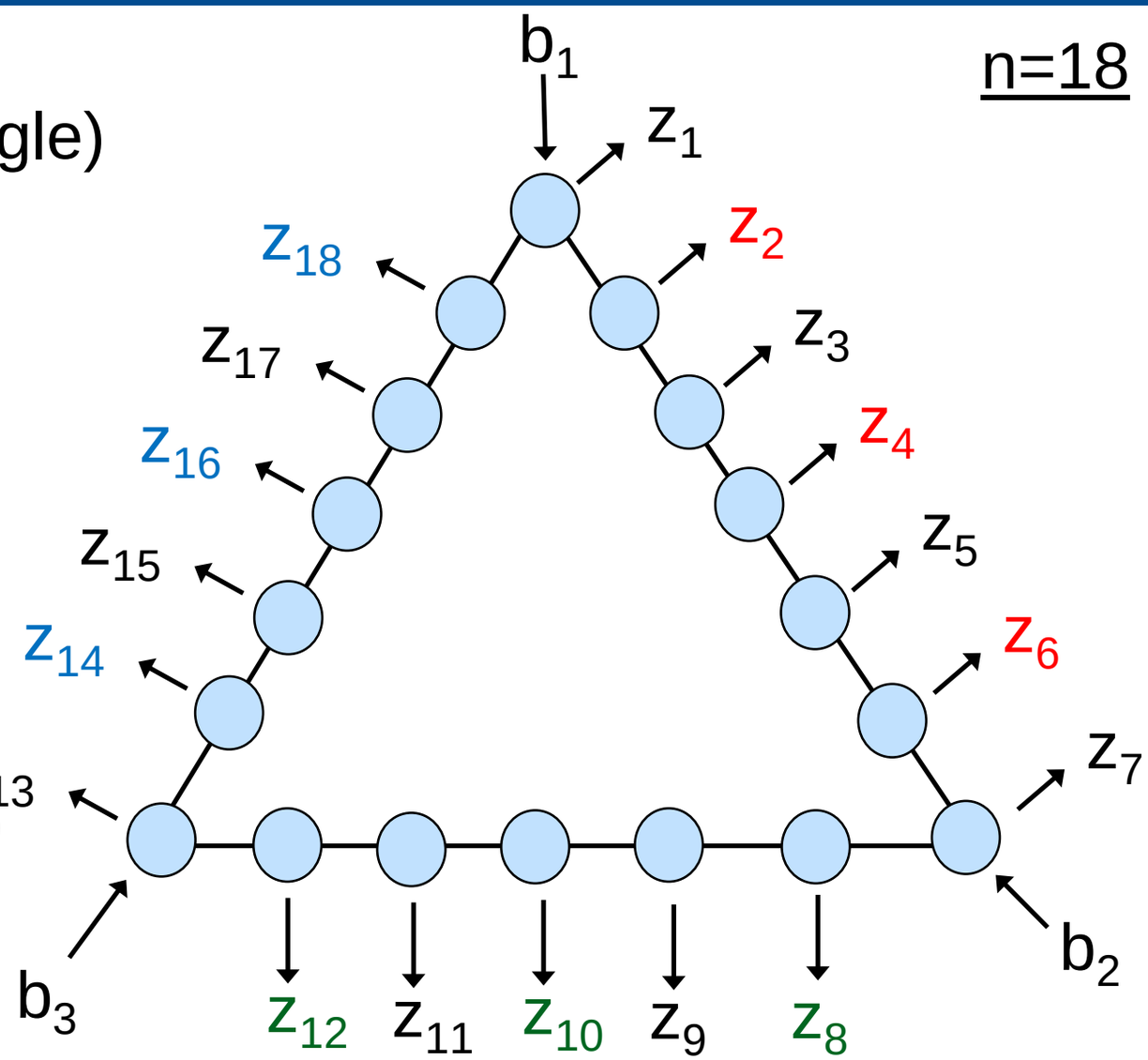
Define the following four bits:

$m_R = z_2 \oplus z_4 \oplus z_6$
(parity of the outputs of the nodes of even index on the right)

$m_B = z_8 \oplus z_{10} \oplus z_{12}$
(parity of the outputs of the nodes of even index on the bottom)

$m_L = z_{14} \oplus z_{16} \oplus z_{18}$

$m_{odd} = z_1 \oplus z_3 \oplus z_5 \oplus z_7 \oplus z_9 \oplus z_{11} \oplus z_{13} \oplus z_{15} \oplus z_{17}$



Claim 1: There is a 2-round quantum algorithm that samples from the uniform distribution over all binary strings $(z_1, z_2, \dots, z_n) \in \{0,1\}^n$ satisfying the following condition:

$$\begin{cases} m_{odd} = 0 & \text{if } (b_1, b_2, b_3) = (0,0,0) \\ m_{odd} \oplus m_R = 1 & \text{if } (b_1, b_2, b_3) = (1,1,0) \\ m_{odd} \oplus m_B = 1 & \text{if } (b_1, b_2, b_3) = (0,1,1) \\ m_{odd} \oplus m_L = 1 & \text{if } (b_1, b_2, b_3) = (1,0,1) \end{cases}$$

Claim 2: In the LOCAL model, any classical algorithm that samples (even approximately) from the same distribution must use at least n/6 rounds.

multiple of 3

Consider a ring of size n (seen as a triangle)

Each “corner” gets a bit as input

Each node will output one bit

Define the following four bits:

$m_R = z_2 \oplus z_4 \oplus z_6$

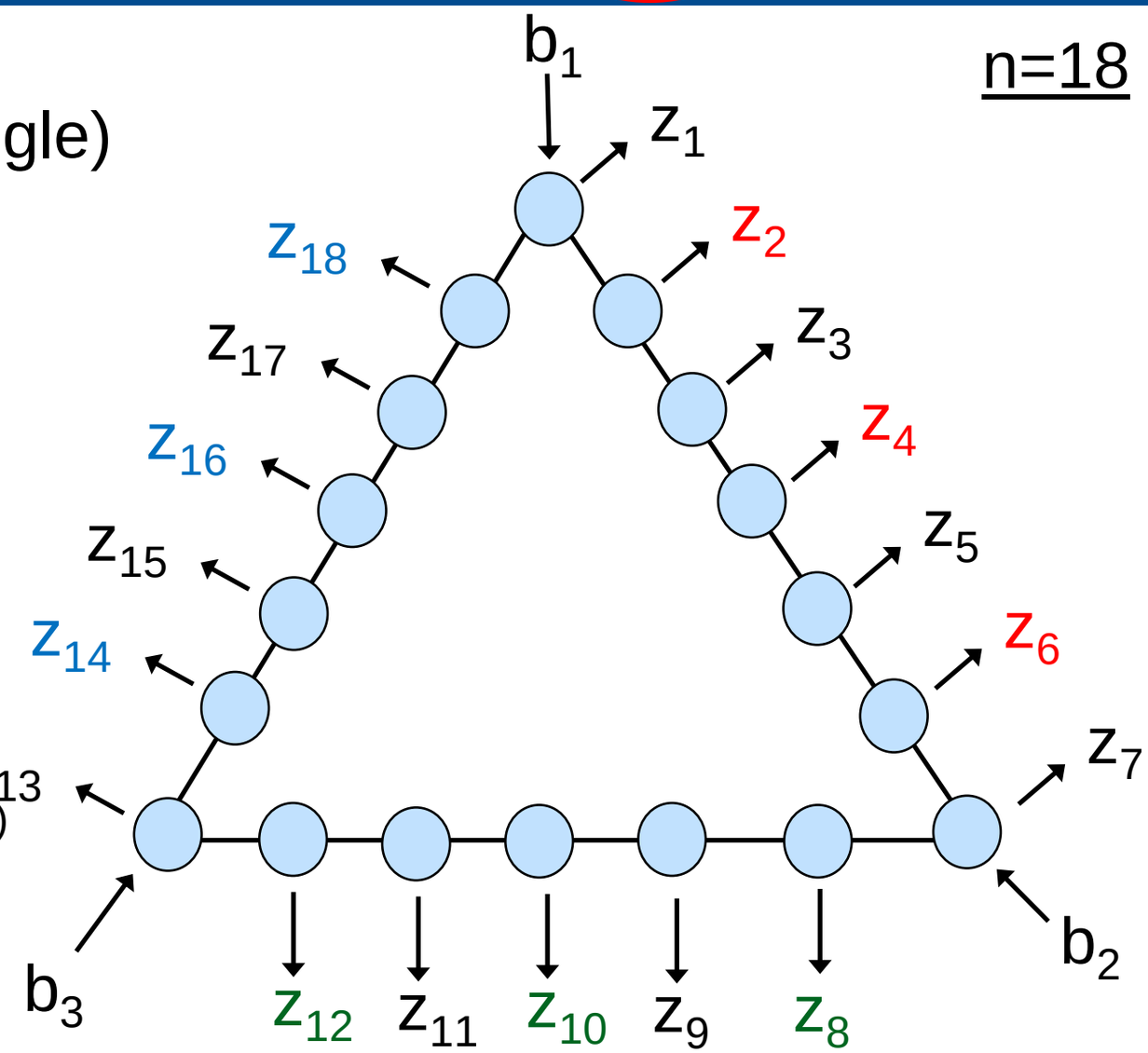
(parity of the outputs of the nodes of even index on the right)

$m_B = z_8 \oplus z_{10} \oplus z_{12}$

(parity of the outputs of the nodes of even index on the bottom)

$m_L = z_{14} \oplus z_{16} \oplus z_{18}$

$m_{odd} = z_1 \oplus z_3 \oplus z_5 \oplus z_7 \oplus z_9 \oplus z_{11} \oplus z_{13} \oplus z_{15} \oplus z_{17}$



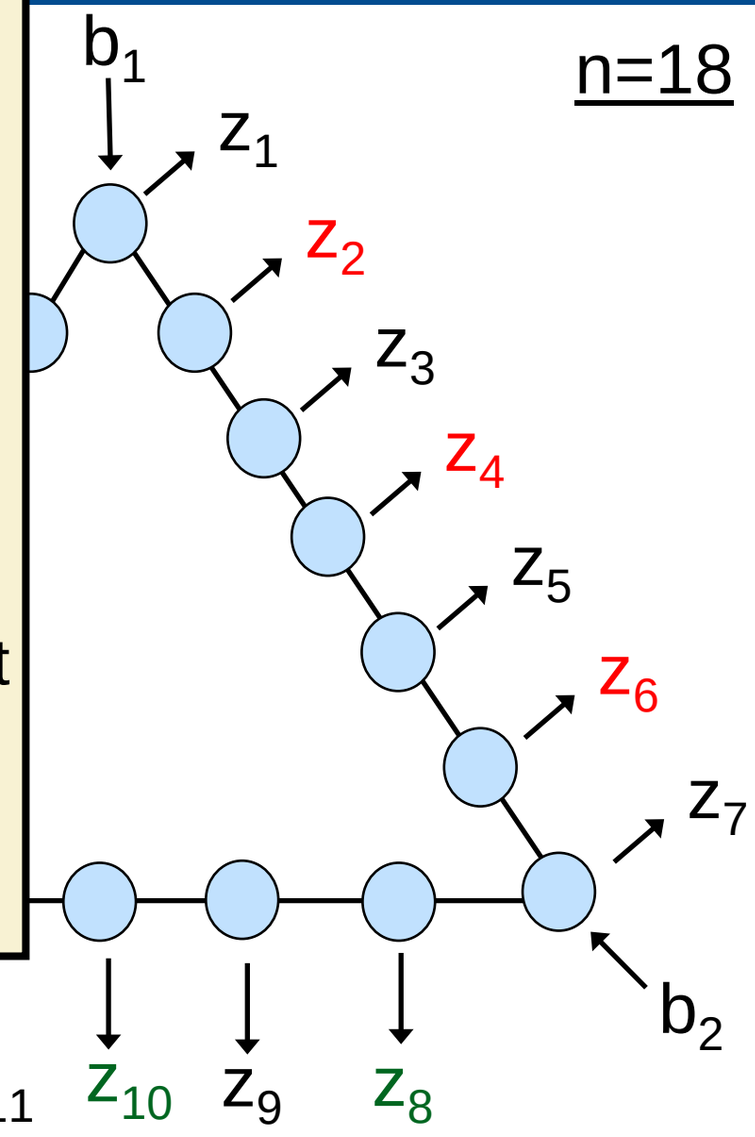
Claim 1: There is a 2-round quantum algorithm that samples from the uniform distribution over all binary strings $(z_1, z_2, \dots, z_n) \in \{0,1\}^n$ satisfying the following condition:

$$\begin{cases} m_{odd} = 0 & \text{if } (b_1, b_2, b_3) = (0,0,0) \\ m_{odd} \oplus m_R = 1 & \text{if } (b_1, b_2, b_3) = (1,1,0) \\ m_{odd} \oplus m_B = 1 & \text{if } (b_1, b_2, b_3) = (0,1,1) \\ m_{odd} \oplus m_L = 1 & \text{if } (b_1, b_2, b_3) = (1,0,1) \end{cases}$$

1. The nodes prepare the graph state corresponding to the whole triangle
   (this can be done in 2 rounds --- see next slides)

2. Each non-corner node measures its qubit in the X basis and then outputs the bit corresponding to the measurement outcome
   (no communication)

3. Each corner node measures its qubit in the X basis if its input bit is 0, or measures it in the Y basis if its input bit is 1, and then outputs the bit corresponding to the measurement outcome
   (no communication)



$m_L = z_{14} \oplus z_{16} \oplus z_{18}$

$m_{odd} = z_1 \oplus z_3 \oplus z_5 \oplus z_7 \oplus z_9 \oplus z_{11} \oplus z_{13} \oplus z_{15} \oplus z_{17}$

Claim 1: There is a 2-round quantum algorithm that samples from the uniform distribution over all binary strings $(z_1, z_2, \ldots, z_n) \in \{0,1\}^n$ satisfying the following condition:

$$\begin{cases} m_{odd} = 0 & \text{if } (b_1, b_2, b_3) = (0,0,0) \\ m_{odd} \oplus m_R = 1 & \text{if } (b_1, b_2, b_3) = (1,1,0) \\ m_{odd} \oplus m_B = 1 & \text{if } (b_1, b_2, b_3) = (0,1,1) \\ m_{odd} \oplus m_L = 1 & \text{if } (b_1, b_2, b_3) = (1,0,1) \end{cases}$$

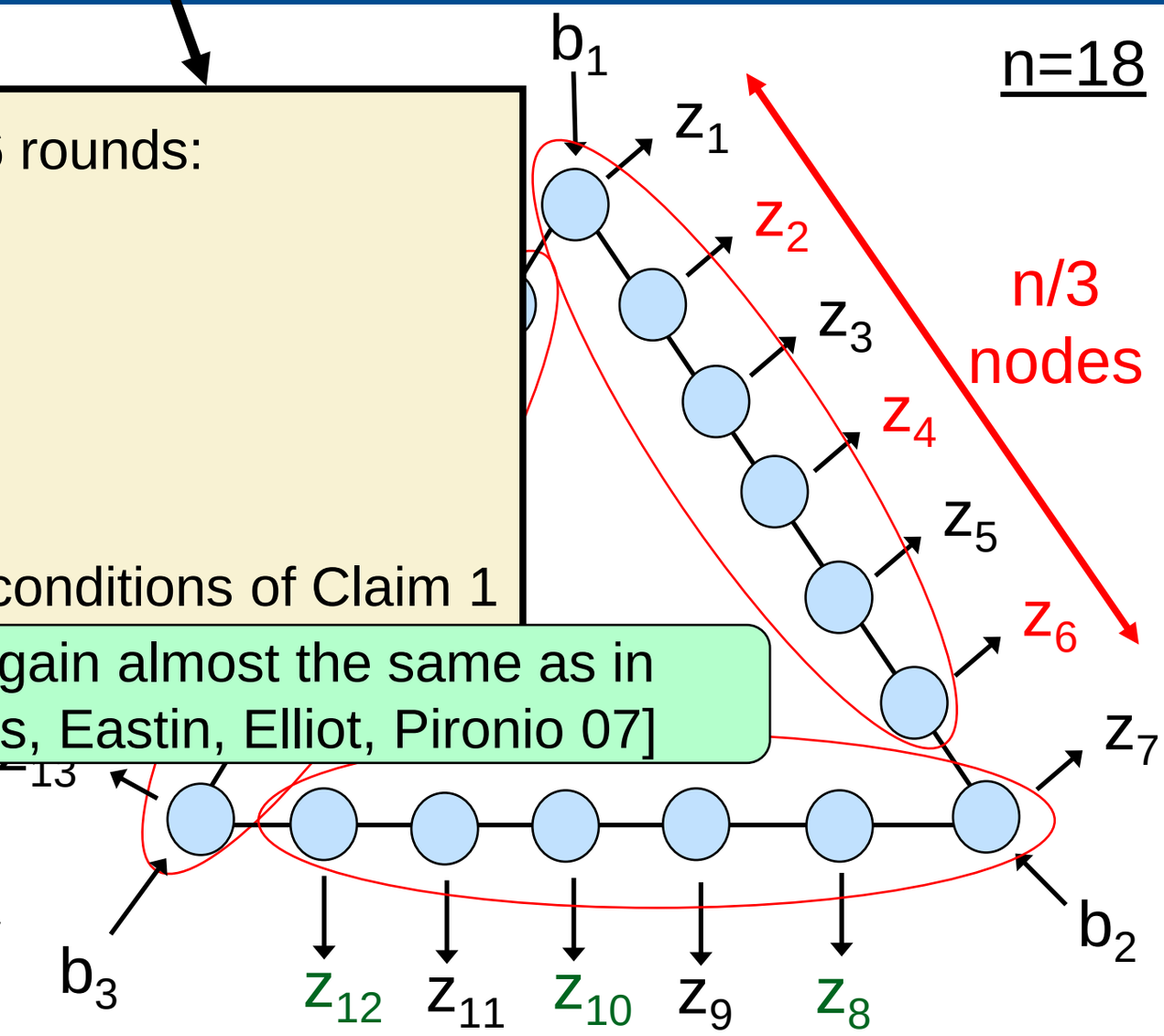
Claim 2: In the LOCAL model, any classical algorithm that samples (even approximately) from the same distribution must use at least n/6 rounds.

- ✓ In any classical protocol using less than n/6 rounds:
  - m_R is an affine function of b_1 and b_2
  - m_B is an affine function of b_2 and b_3
  - m_L is an affine function of b_1 and b_3
  - m_{odd} is an affine function of b_1, b_2 and b_3
- ✓ Such functions cannot satisfy all the linear conditions of Claim 1

The proof is again almost the same as in [Barrett, Caves, Eastin, Elliot, Pironio 07]

$m_R = z_2 \oplus z_4 \oplus z_6$
(parity of the outputs of the nodes of even index on the right)

$m_{odd} = z_1 \oplus z_3 \oplus z_5 \oplus z_7 \oplus z_9 \oplus z_{11} \oplus z_{13} \oplus z_{15} \oplus z_{17}$
(parity of the outputs of all the nodes of odd index)



Claim 1: There is a 2-round quantum algorithm that samples from the uniform distribution over all binary strings $(z_1, z_2, \dots, z_n) \in \{0,1\}^n$ satisfying the following condition:

$$\begin{cases} m_{odd} = 0 & \text{if } (b_1, b_2, b_3) = (0,0,0) \\ m_{odd} \oplus m_R = 1 & \text{if } (b_1, b_2, b_3) = (1,1,0) \\ m_{odd} \oplus m_B = 1 & \text{if } (b_1, b_2, b_3) = (0,1,1) \\ m_{odd} \oplus m_L = 1 & \text{if } (b_1, b_2, b_3) = (1,0,1) \end{cases}$$

Claim 2: In the LOCAL model, any classical algorithm that samples (even approximately) from the same distribution must use at least n/6 rounds.

multiple of 3

Consider a ring of size n (seen as a triangle)

Each “corner” gets a bit as input

Each node will output one bit

Define the following four bits:

$m_R = z_2 \oplus z_4 \oplus z_6$

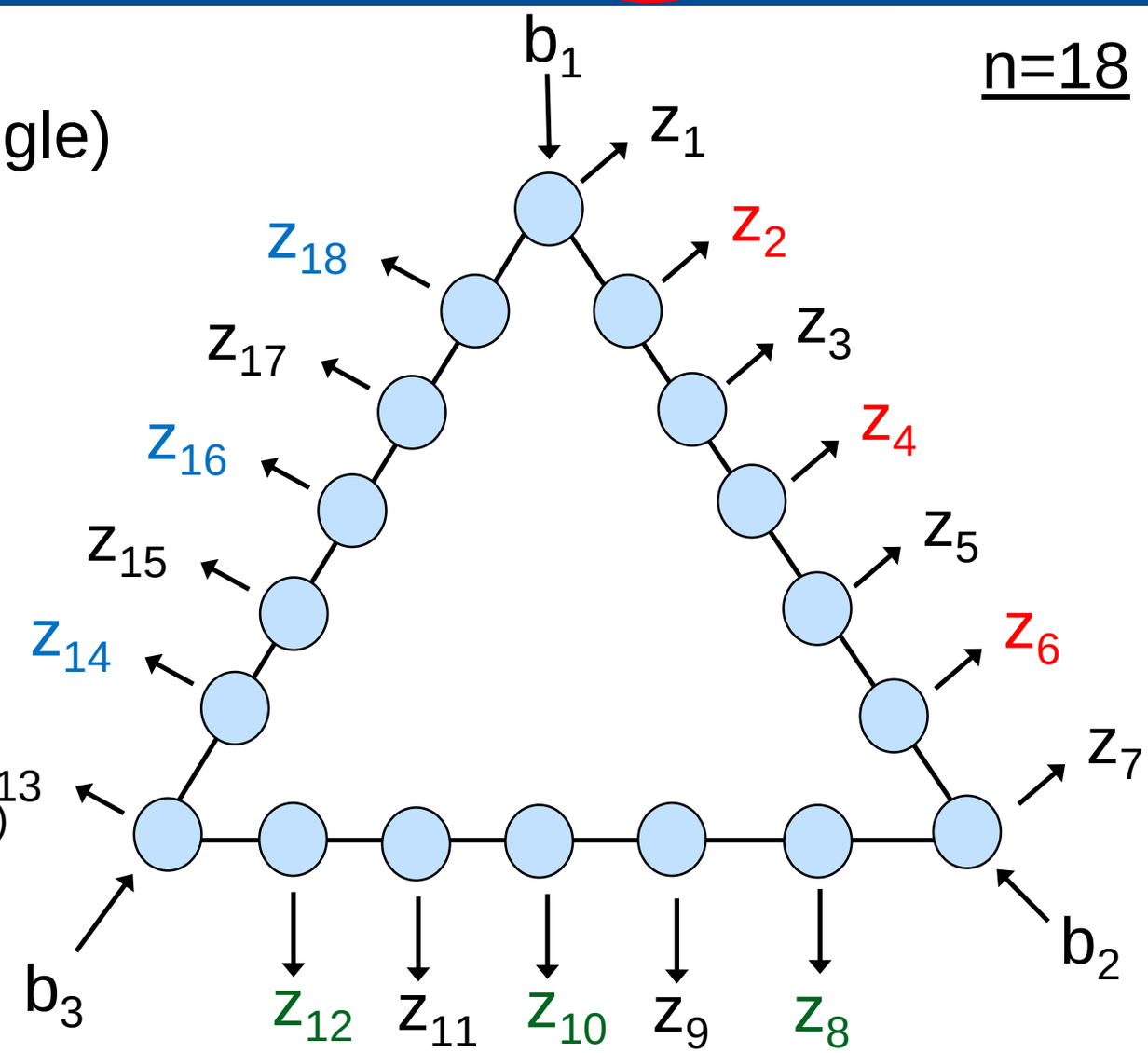
(parity of the outputs of the nodes of even index on the right)

$m_B = z_8 \oplus z_{10} \oplus z_{12}$

(parity of the outputs of the nodes of even index on the bottom)

$m_L = z_{14} \oplus z_{16} \oplus z_{18}$

$m_{odd} = z_1 \oplus z_3 \oplus z_5 \oplus z_7 \oplus z_9 \oplus z_{11} \oplus z_{13} \oplus z_{15} \oplus z_{17}$



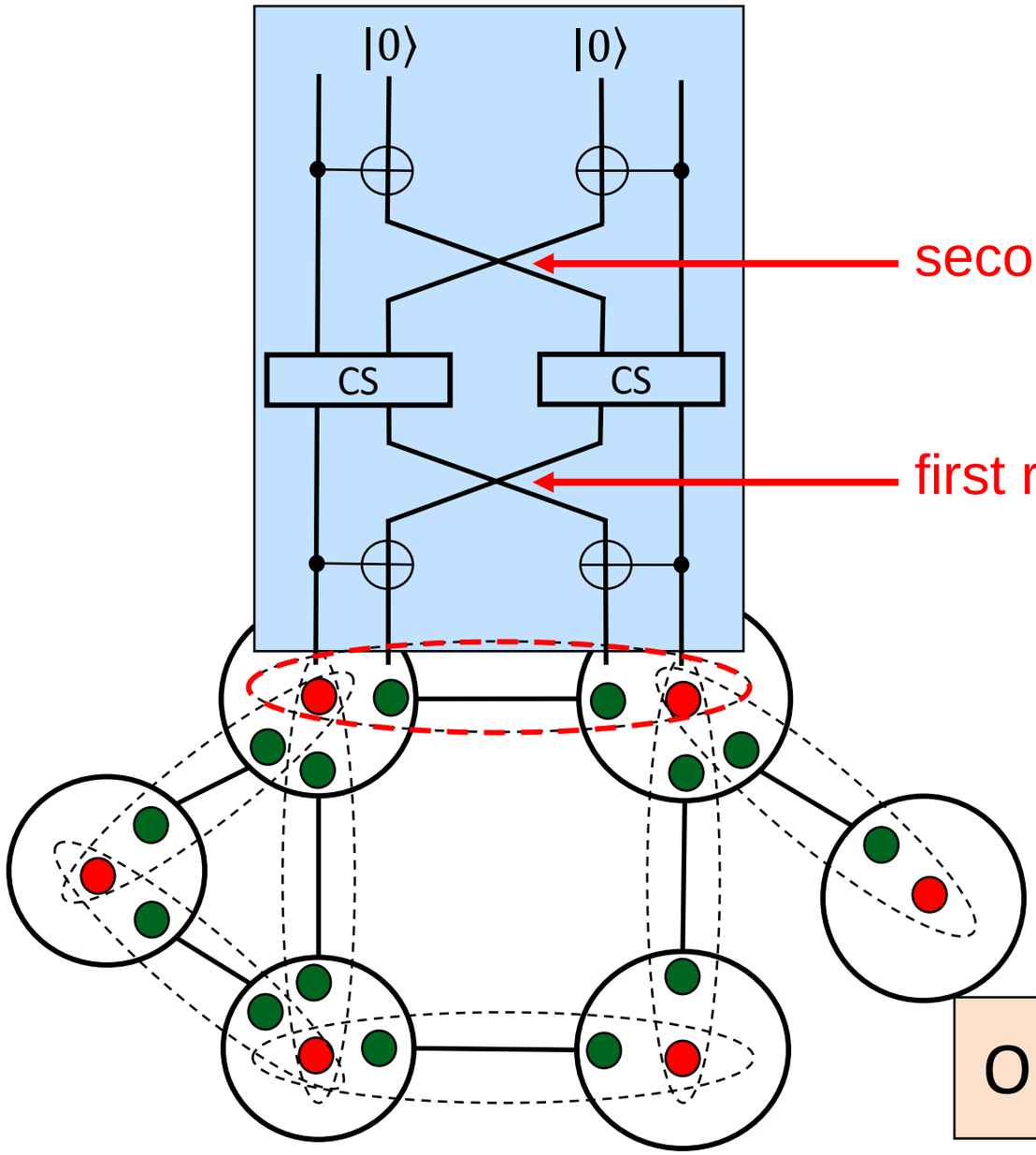
Claim 1: There is a 2-round quantum algorithm that samples from the uniform distribution over all binary strings $(z_1, z_2, \dots, z_n) \in \{0,1\}^n$ satisfying the following condition:

$$\begin{cases} m_{odd} = 0 & \text{if } (b_1, b_2, b_3) = (0,0,0) \\ m_{odd} \oplus m_R = 1 & \text{if } (b_1, b_2, b_3) = (1,1,0) \\ m_{odd} \oplus m_B = 1 & \text{if } (b_1, b_2, b_3) = (0,1,1) \\ m_{odd} \oplus m_L = 1 & \text{if } (b_1, b_2, b_3) = (1,0,1) \end{cases}$$

Preparing the Graph State of a Network in 2 Rounds

works for any network

1. Each node prepares one qubit in state $\frac{|0\rangle+|1\rangle}{\sqrt{2}}$
2. Each node prepares one ancilla qubit initialized to $|0\rangle$ for each edge
3. For each edge a controlled-Z operation is implemented by using the ancilla qubits in two rounds of communication



$$CZ = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$$

$$CS = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & i \end{bmatrix}$$

$$CS^2 = CZ$$

● : one qubit initialized to state $\frac{|0\rangle+|1\rangle}{\sqrt{2}}$

● : one ancilla qubit initialized to state $|0\rangle$

Open problem: can we do it in 1 round?

Conclusions

- ✓ We have shown that in the CONGEST model the diameter of the network can be computed faster using quantum distributed algorithms
  (for constant diameter: $\Theta(\sqrt{n})$ rounds quantumly vs. $\Theta(n)$ rounds classically)
- ✓ We have shown that in the LOCAL model quantum distributed algorithms can also be faster, at least for some computational task
  (for our problem: 2 rounds quantumly vs. $\Theta(n)$ rounds classically)

Interesting research directions:

- ✓ Find other applications of quantum distributed algorithms in the CONGEST and LOCAL model
- ✓ Prove the superiority of quantum distributed algorithms in other models

Recent result [Izumi, LG 2019]: $O(n^{1/4})$-round quantum algorithm for the All-Pairs Shortest Path problem in the CONGEST-CLIQUE model (classically the best known is $O(n^{1/3})$ rounds)

本日の予定

I. 量子探索アルゴリズム ］ 1時間

II. その他の量子アルゴリズムの簡単な紹介

- Shorの素因数アルゴリズム
- HHLアルゴリズム
- 量子ウォーク

］ 30分

III. 量子探索アルゴリズムの応用例

量子分散計算

］ 1時間