# Quantum Algorithms

Michele Mosca
23 February 2016

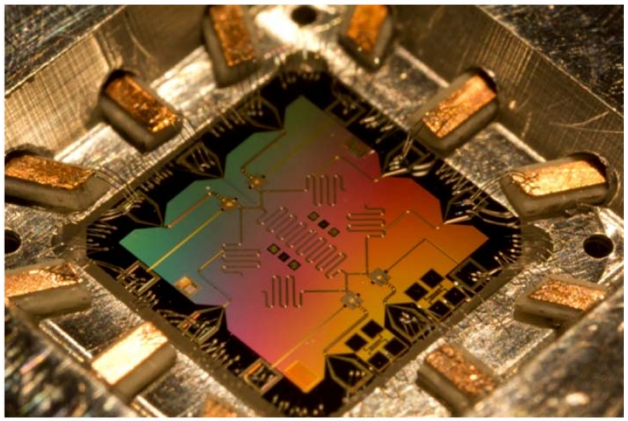
PQCrypto 2016
Fukuoka, Japan

Post-Quantum Cryptography Winter School

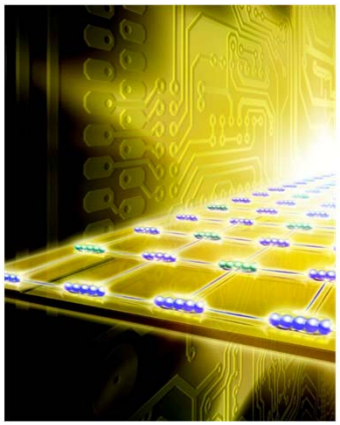



CryptoWorks21

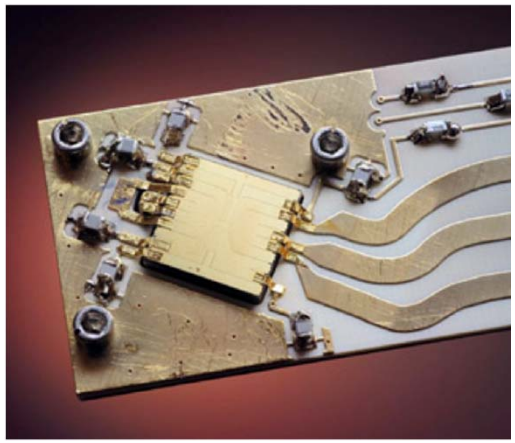
# Computationally secure cryptography in the context of quantum computers



E. Lucero, D. Mariantoni, and M. Mariantoni



© Harald Ritsch



Y. Colombe/NIST

UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing

SCIENCE VOL 339 8 MARCH 2013

REVIEW

# Superconducting Circuits for Quantum Information: An Outlook

M. H. Devoret[1,2] and R. J. Schoelkopf[1]*

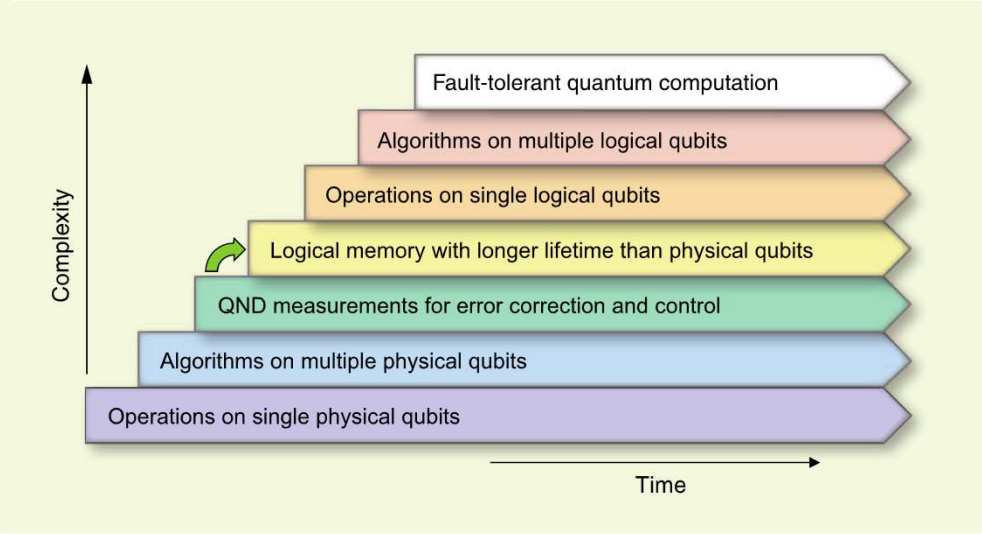
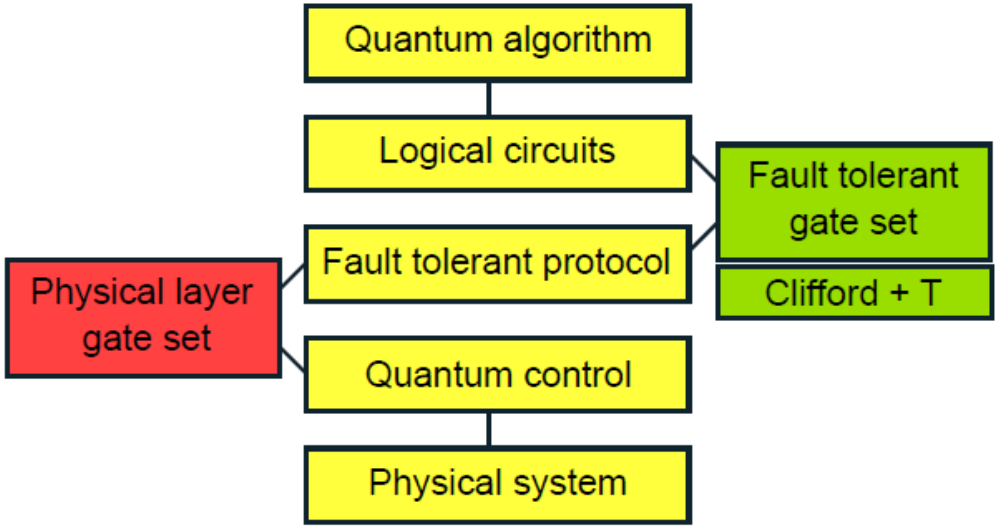



**Fig. 1.** Seven stages in the development of quantum information processing. Each advancement requires mastery of the preceding stages, but each also represents a continuing task that must be perfected in parallel with the others. Superconducting qubits are the only solid-state implementation at the third stage, and they now aim at reaching the fourth stage (green arrow). In the domain of atomic physics and quantum optics, the third stage had been previously attained by trapped ions and by Rydberg atoms. No implementation has yet reached the fourth stage, where a logical qubit can be stored, via error correction, for a time substantially longer than the decoherence time of its physical qubit components.

UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing

# Quantum compilers

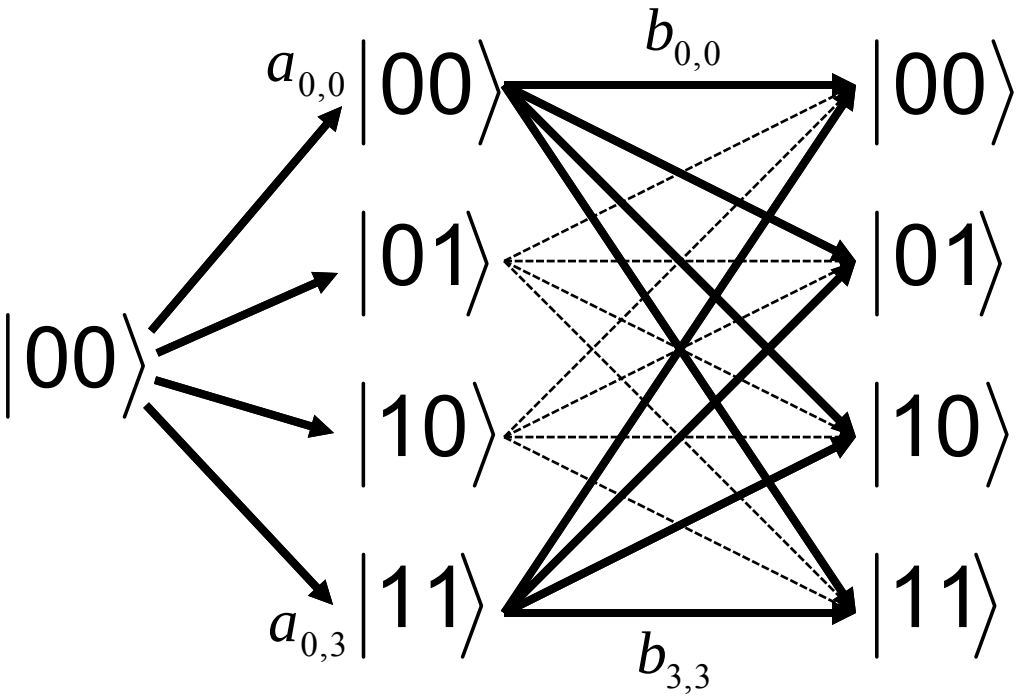
The efficiency of each step in the translation from high level algorithm to physical device impacts the efficiency of quantum attacks.



# How do quantum algorithms work?

UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing

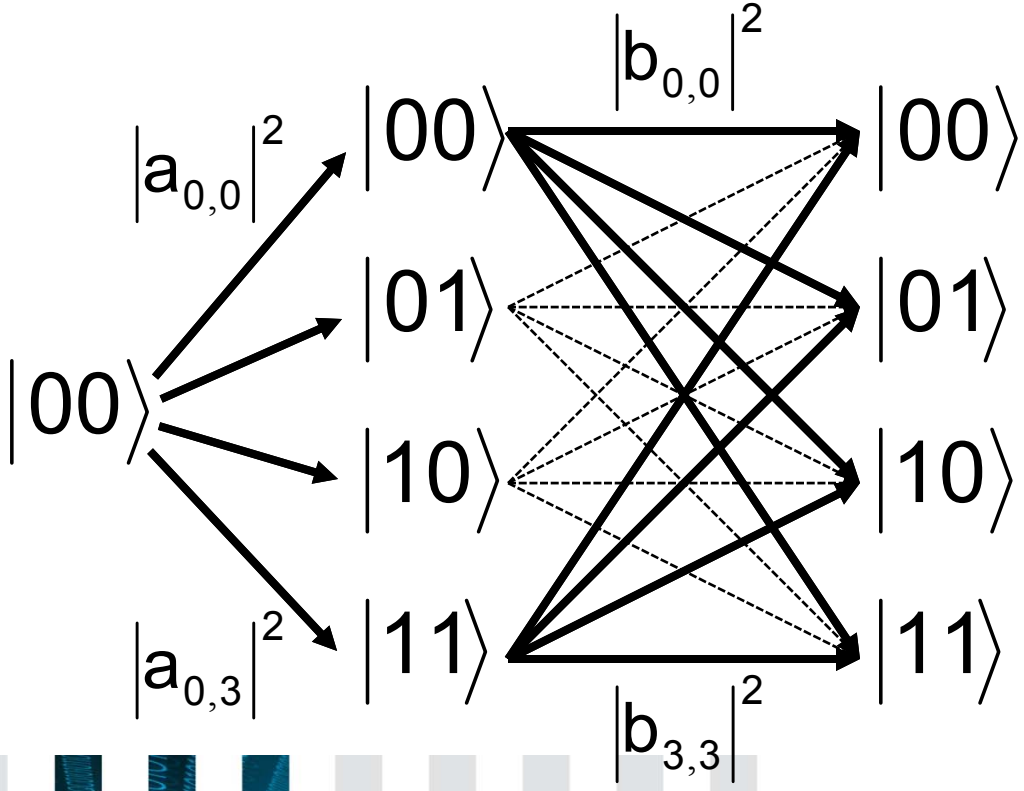
# Quantum algorithm




$$\Pr(00) = \left| \sum_j a_{0,j} b_{j,0} \right|^2$$

UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing

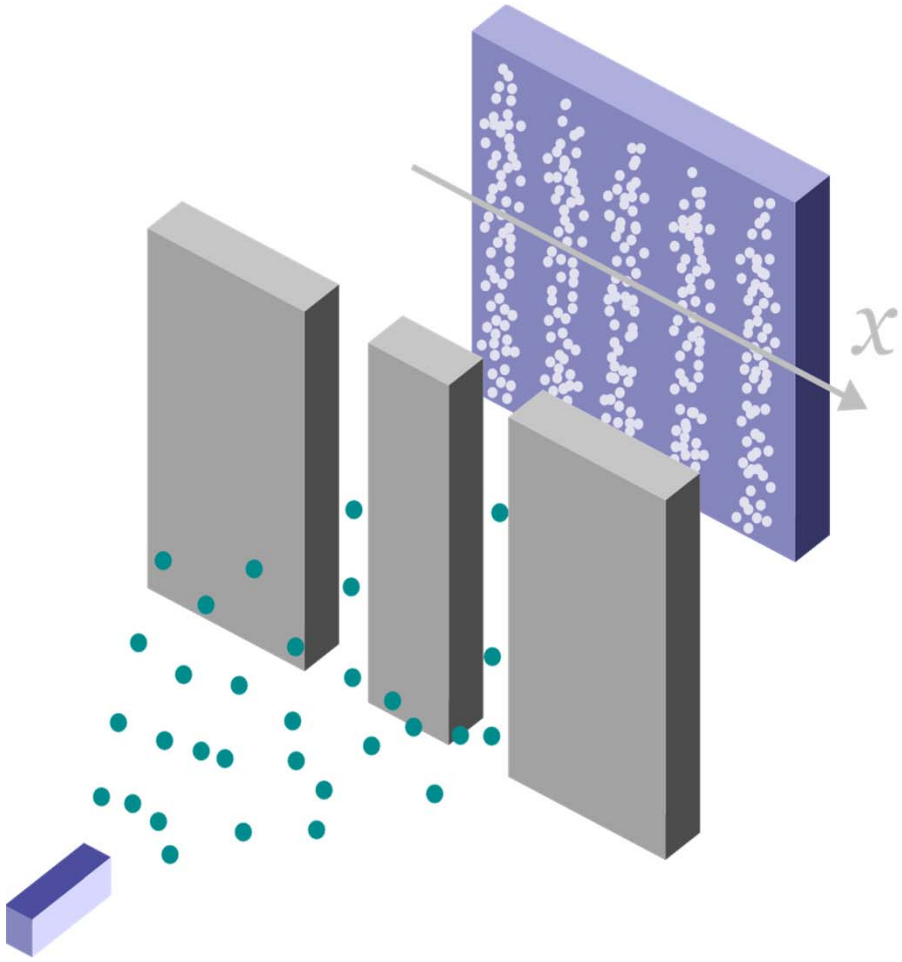
If we look at the state of the system at each step, it behaves like a classical randomized algorithm.




$$\Pr(00) = \sum_j |a_{0,j}|^2 |b_{j,0}|^2$$

$$= \sum_j |a_{0,j} b_{j,0}|^2$$

UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing



The art of quantum algorithmics is to choreograph *constructive interference* on *desirable outcomes* and *destructive interference* on *undesirable outcomes*.

UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing

# Bernstein-Vazirani problem

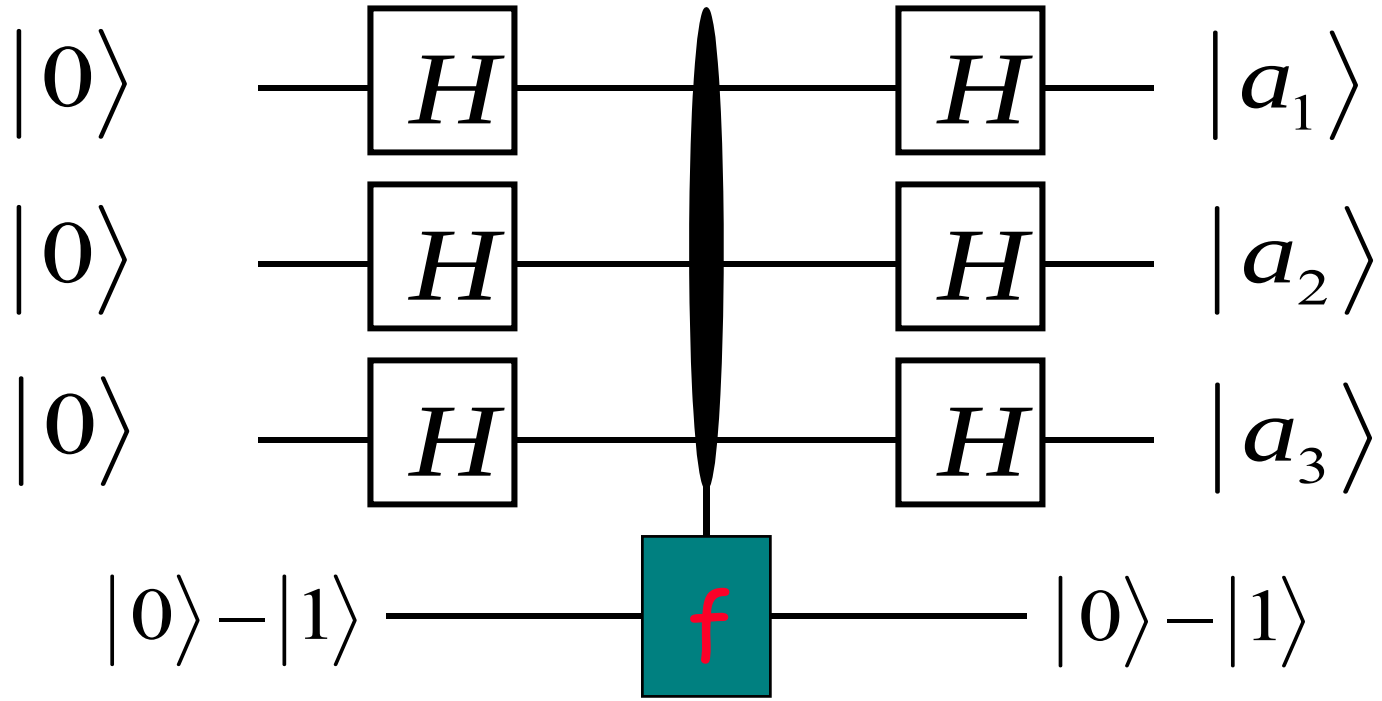
Suppose $f : \{0,1\}^n \rightarrow \{0,1\}$ is of the form $f(x) = a \cdot x$

for some $a \in \{0,1\}^n$

Given $|x\rangle|c\rangle \overset{U_f}{\mapsto} |x\rangle|c \oplus f(x)\rangle$ determine

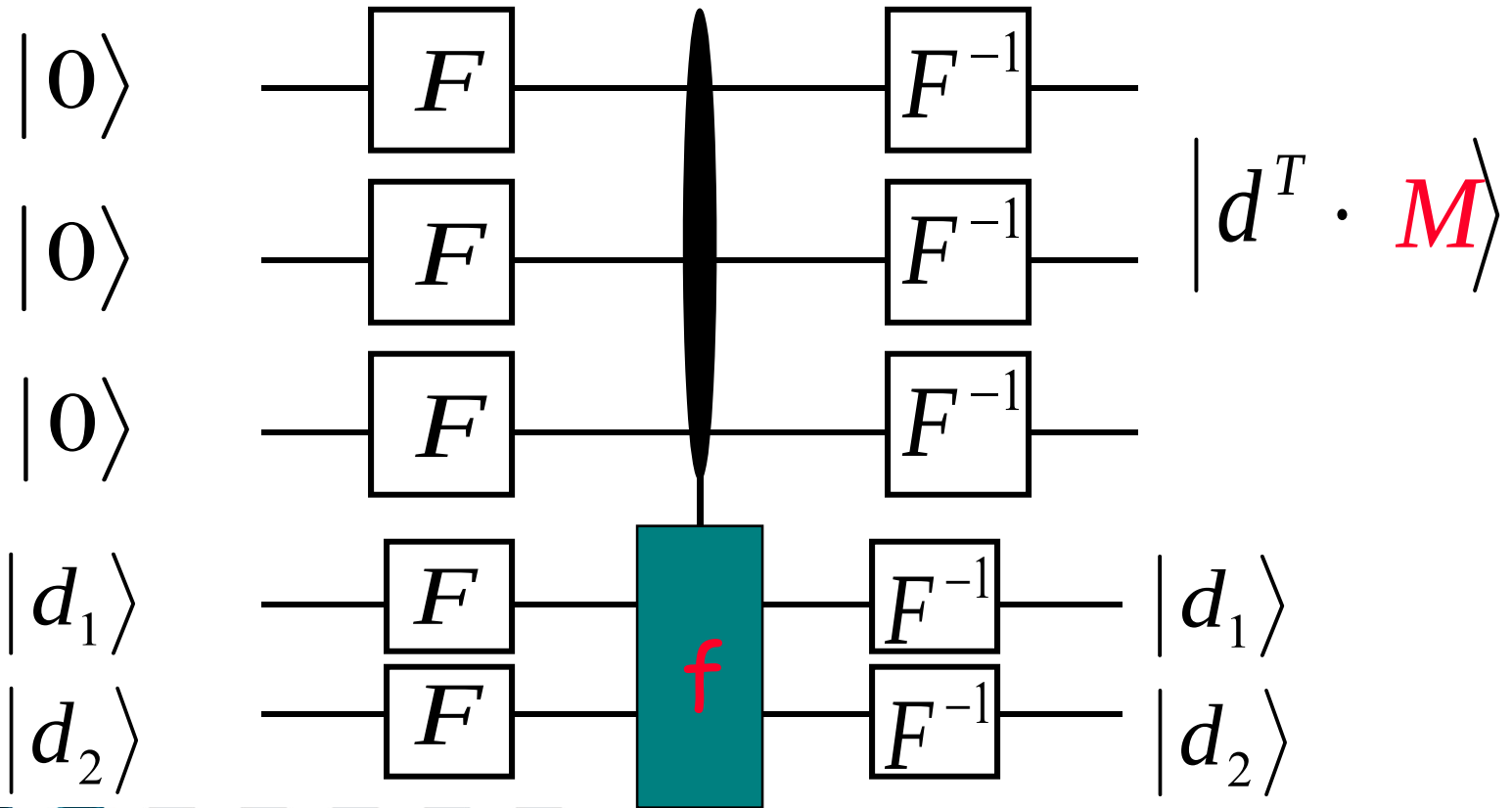
$$a = a_1 a_2 \ldots a_n$$

# Bernstein-Vazirani problem



# Generally

$$f : Z_p^n \rightarrow Z_p^m \qquad x \mapsto Mx$$




UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing

# A property of Hadamard transformation

Consider $S \leq Z_2^{\,n}$

$$S^{\perp} = \left\{ t : t \in Z_2^{\,n}, s \cdot t = 0 \ \forall s \in S \right\}$$

Let $$\left| y + S \right\rangle = \sum_{s \in S} \frac{1}{\sqrt{|S|}} \left| y + s \right\rangle$$

Then $$H^{\otimes n} \left| y + S \right\rangle = \sum_{t \in S^{\perp}} \frac{(-1)^{y \cdot t}}{\sqrt{|S^{\perp}|}} \left| t \right\rangle$$

UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing

# Simon's problem

Suppose $f : \{0,1\}^n \to X$ has the property that

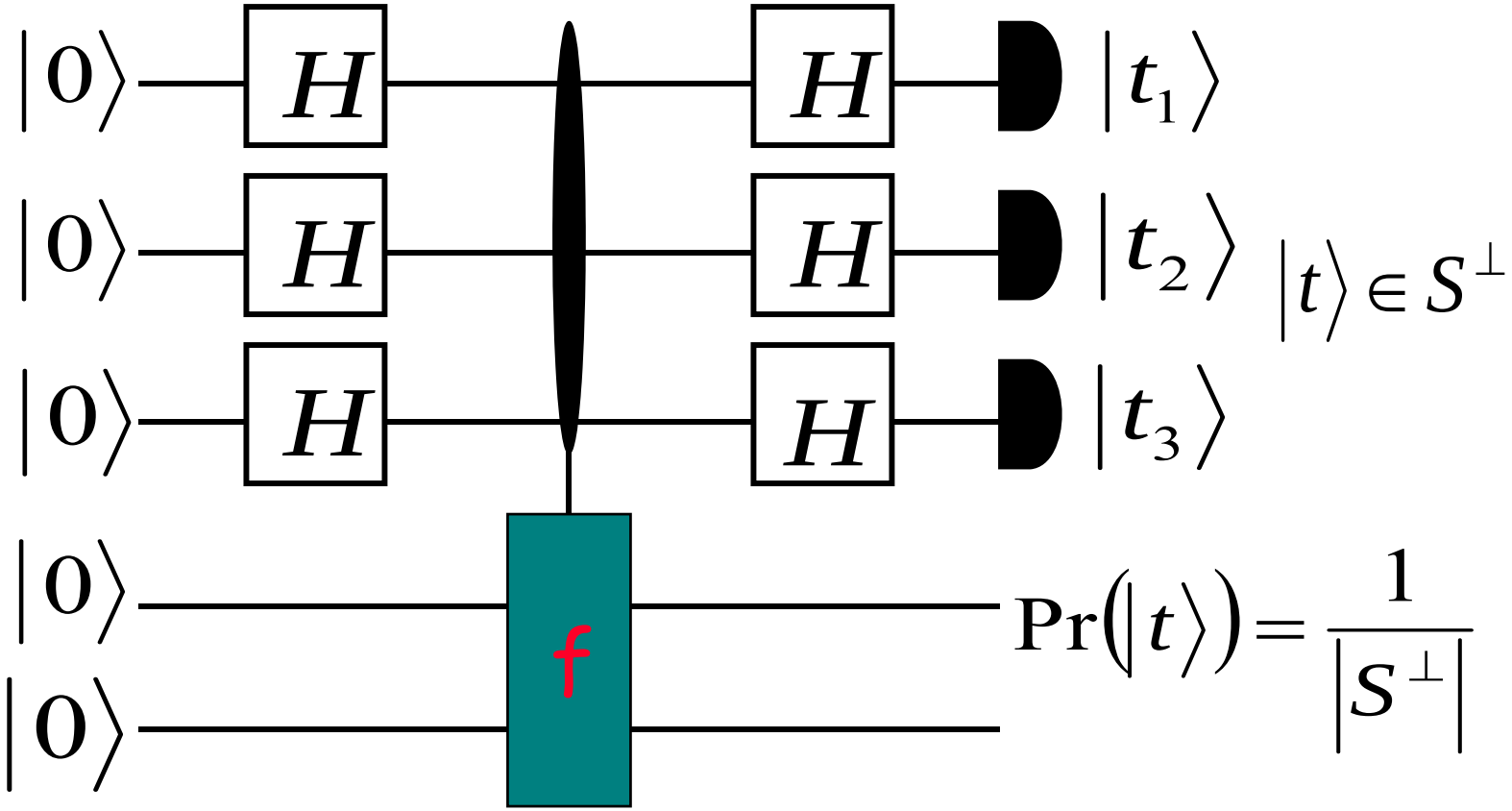
$$f(x) = f(y) \quad \text{iff} \quad x + S = y + S$$

For some "hidden subgroup" $S \leq Z_2{}^n$

Given $|x\rangle|0\rangle \overset{U_f}{\mapsto} |x\rangle|f(x)\rangle$ find $S$

UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing

# Simon's algorithm



# Simon's algorithm

Sample $t_1, t_2, \cdots, t_{n+O(1)} \in S^{\perp}$

Solve $$\begin{bmatrix} t_1 \\ t_2 \\ \vdots \\ t_{n+O(1)} \end{bmatrix} \begin{pmatrix} \\ s \\ \\ \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing

# Applications of Simon's algorithm??

Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer*

Peter W. Shor†

n 1996

computer. Indeed, while Bernstein and Vaziarni's problem appears contrived, Simon's problem looks quite natural. Simon's algorithm inspired the work presented in this paper.

UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing

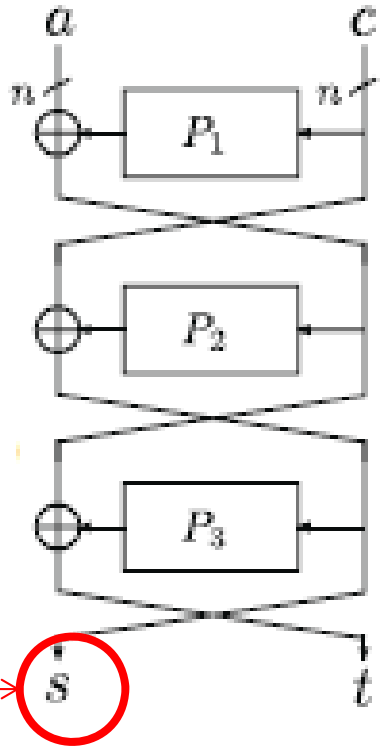
ISIT 2010, Austin, Texas, U.S.A., June 13 - 18, 2010

# Quantum Distinguisher Between the 3-Round Feistel Cipher and the Random Permutation

Hidenori Kuwakado
Graduate School of Engineering
Kobe University
1-1 Rokkodai-cho Nada-ku Kobe 657-8501, Japan

Masakatu Morii
Graduate School of Engineering
Kobe University
1-1 Rokkodai-cho Nada-ku Kobe 657-8501, Japan

$$
\begin{aligned}
\mathrm{FP}(x) &= \mathrm{FP}(a \parallel c) \\
&= c \oplus P_2(a \oplus P_1(c)) \\
&\quad \parallel (a \oplus P_1(c)) \oplus P_3(c \oplus P_2(a \oplus P_1(c))) \\
&= s \parallel t.
\end{aligned}
$$




Denote **W(x)=W(a||c)=s**

Fig. 1. The 3-round Feistel cipher with internal permutations, $\mathrm{FP}(a \parallel c) = s \parallel t$.

$$W(a\|c) = c \oplus P_2\left(a \oplus P_1(c)\right)$$

Let $\alpha, \beta \in \{0,1\}^n, \alpha \neq \beta, \quad b \in \{0,1\}$

Let $$f(b\|a) = \begin{cases} W(a\|\alpha) \oplus \beta & if \ \ b = 0 \\ W(a\|\beta) \oplus \alpha & if \ \ b = 1 \end{cases}$$

Then $$f(b\|a) = f(b'\|a') \ \ iff \ \ (b'\|a') \oplus (b\|a) = (1\|z) \ or \ (0\|0)$$

where $$z = P_1(\alpha) \oplus P_1(\beta)$$

UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing

So $f(b\|a) = f(b'\|a') \text{ iff } (b'\|a') \oplus (b\|a) = (1\|z) \text{ or } (0\|0)$

where $z = P_1(\alpha) \oplus P_1(\beta)$

(N.B. the “only if” part is important, at least approximately)

In other words, if W is based on the 3-round Feistel cipher, the derived function f will have the above property.

Simon’s algorithm will randomly sample vectors orthogonal to (1||z).

In other words, if W is based on the 3-round Feistel cipher, the derived function f will have the above property, and Simon's algorithm will randomly sample vectors orthogonal to (1||z).

However, if W is based on a random permutation, no such pattern is likely to emerge.

Thus, ***a quantum algorithm can efficiently distinguish a 3-round Feistel cipher with internal permutations from a random permutation.***

Recent work and additional references in Kaplan et al.:
http://arxiv.org/abs/1602.05973

UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing

# Generalization of Simon's problem, order-finding and DLP: "Hidden subgroup problem"

- A unifying framework was developed for these problems

$$f : G \rightarrow X$$

$$f(x) = f(y) \quad \text{iff} \quad x + S = y + S$$

$$\text{for some } S \leq G$$

- If $G$ is Abelian, finitely generated, and represented in a reasonable way, we can efficiently find $S$.

**Order finding (basis of quantum factoring):**

$G = Z \qquad X$ any group

$f(x) = a^x$

$K = rZ$

(applies more generally to finding the period of any periodic function f)

**Discrete Log** of $b = a^k$ to base $a$ :

$$G = Z_r \times Z_r \qquad X \quad \text{any group}$$

$$f(x, y) = a^x b^y$$

$$K = \langle (k, -1) \rangle$$

**Self-shift equivalences** (Grigoriev):

$$G = GF(q)^n \qquad X = GF(q)[X_1, X_2, \ldots, X_n]$$

$$f(a_1, a_2, \ldots, a_n) = P(X_1 - a_1, \ldots, X_n - a_n)$$

$$K = \{(a_1, \ldots, a_n) : P(X_1 - a_1, \ldots, X_n - a_n) = P(X_1, \ldots, X_n)\}$$

**Abelian Stabilizer Problem** (Kitaev)

## Decomposing Abelian groups

- Any finite Abelian group $G$ is the direct sum of finite cyclic groups

$$\langle g_1 \rangle \oplus \langle g_2 \rangle \oplus \cdots \oplus \langle g_n \rangle$$

$$g_1, g_2, \cdots, g_n \qquad G = \langle g_1 \rangle \oplus \langle g_2 \rangle \oplus \cdots \oplus \langle g_n \rangle$$

$$e.g. \quad G = Z_N^*$$

- Given any polynomial sized set of generators, we can use the Abelian HSP algorithm to find new generators that decompose $G$ into a direct sum of finite cyclic groups. http://arxiv.org/abs/cs/0101004

- Leads directly to an algorithm for computing the class group and class number of a quadratic number field [Watrous ‘00] (computing the class group of a more general number field is a much more difficult task).
- Decomposition of Abelian groups was also applied by
    - Friedl, Ivanyos and Santha [FIS05] to test if a finite set with a binary operation is an Abelian group,
    - Kedlaya [Ked06] to compute the zeta function of a genus *g* curve over a finite field F*q* in time polynomial in *g* and *q*, and
    - Childs, Jao and Soukharev [CJS10] in order to construct elliptic curve isogenies in subexponential time.

# What about non-Abelian HSP?

- Consider the symmetric group $G = S_n$
- $S_n$ is the set of permutations of $n$ elements
- Let $G$ be an $n$-vertex graph
- Let $X_G = \{\pi(G) \mid \pi \in S_n\}$

$$f_G : S_n \to X_G \qquad f_G(\pi) = \pi(G)$$

$$f_G(\pi_1) = f_G(\pi_2) \Leftrightarrow \pi_1 K = \pi_2 K$$

$$K = AUT(G) = \{\pi \mid \pi(G) = G\}$$

- So the hidden subgroup of $f_G$ is the automorphism group of $G$

# Dihedral Hidden Subgroup Problem

$$f : D_n \to X$$

$$f(b,x) = f(b',x') \Leftrightarrow (b-b', x-x') \in \{(0,0),(1,s)\}$$

- A quantum computer can easily compute states of the form ("coset states") for random x:

$$|0,x\rangle + |1, x+s \bmod n\rangle$$

- This can be easily converted to a state of the form (for random known k):

$$|0\rangle + e^{2\pi iks/n}|1\rangle$$

# Dihedral Hidden Subgroup Problem

- It is easy to find **s** given

$$|0\rangle + e^{2\pi i s/n}|1\rangle$$

$$|0\rangle + e^{2\pi i 2s/n}|1\rangle$$

$$|0\rangle + e^{2\pi i 4s/n}|1\rangle$$

$$|0\rangle + {}^{2\pi i 8 s/n}|1\rangle$$

$$\vdots$$

- Kuperberg's **sieving** method constructs these states from

$$e^{O\left(\sqrt{n}\right)}$$ samples of

$$|0\rangle + e^{2\pi i k s/n}|1\rangle$$

with random k.

# Dihedral Hidden Subgroup Problem

- It is easy to find **s** given

$$|0\rangle + e^{2\pi i s/n}|1\rangle$$

$$|0\rangle + e^{2\pi i 2s/n}|1\rangle$$

$$|0\rangle + e^{2\pi i 4s/n}|1\rangle$$

$$|0\rangle + {}^{2\pi i 8 s/n}|1\rangle$$

$$\vdots$$

- Solving **average-case subset sum** or **LWE** also suffices (Regev)

# Applications of Dihedral Hidden Subgroup Algorithm

- Regev:

THEOREM 1.1. *If there exists a solution to the dihedral coset problem with failure parameter* f *then there exists a quantum algorithm that solves the* $\Theta(n^{\frac{1}{2}+2\mathrm{f}})$*-unique-SVP.*

# Applications of Dihedral Hidden Subgroup Algorithm

- Consider this approach to Diffie-Hellman-like key exchange:
- Group G acting on a set X $g \in G,\ g^n = 1,\ x \in X,\ a, b \in Z_{>0}$
- Alice sends Bob $g^a(x)$
- Bob send Alice $g^b(x)$
- They both compute the key $g^{ab}(x) = g^{ba}(x)$
- (Childs-Ivanyos) Can use sieving to find a,b in time $e^{O(\sqrt{n})}$
- Childs-Ivanyos also find efficient algorithms for discrete logs in semi-groups

# Non-Abelian HSP

- Tools include non-Abelian QFT, “pretty good” measurements, “sieving”, and non-trivial reductions to Abelian HSP in some cases.

# Generalizations of Abelian HSP

- Finding Hidden Shifts and Translations
- Can generalize to finding hidden “non-linear” structures. E.g. hidden radius problem, shifted subset problem, hidden polynomial problem
- Estimating “Gauss sums”
- Etc.

# Generalizations of Abelian HSP

- Can view HSP has a hidden sub-lattice problem for
  $Z \otimes Z \otimes \cdots \otimes Z = Z^n$
  One way to generalize the problem, is to find a hidden sub-lattice of
  $R \otimes R \otimes \cdots \otimes R = R^n$ .

  Need to define appropriate ways for specifying/approximating inputs and outputs.

  Applications include solving Pell's equation, Principal Ideal Problem, and finding the unit group of a number field.

# Continuous HSP on $\mathbb{R}^m$

(unit vectors $|\cdot\rangle$ in a complex vector space)

**Given** $f: \mathbb{R}^m \to \{\text{quantum states}\}$, s.t.: $\exists$ discrete $H \leq \mathbb{R}^m$,

1. (Periodic) $x - y \in H \Rightarrow |f(x)\rangle = |f(y)\rangle$.
2. (Pseudo-injective) $x - y$ **far** from $H \Rightarrow |f(x)\rangle \perp |f(y)\rangle$
3. (Lipschitz continuitiy) $x - y$ **close** to $H \Rightarrow |f(x)\rangle \approx |f(y)\rangle$

$\min_{v \in H} ||x - y - v|| \geq r \Rightarrow \langle f(x)|f(y)\rangle \leq \epsilon.$

$|||f(x)\rangle - |f(y)\rangle|| \leq a \cdot ||x - y||.$

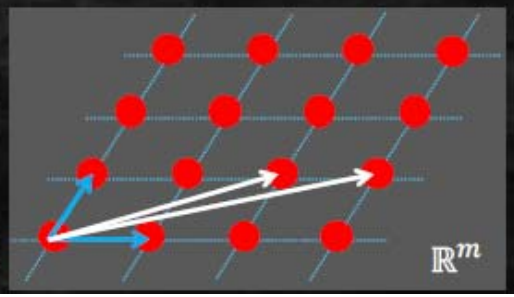
**Goal**: Find (hidden subgroup) $H$.

**Theorem** [EHKS14] $\exists$ efficient quantum algorithm solving continuous HSP on $\mathbb{R}^m$.

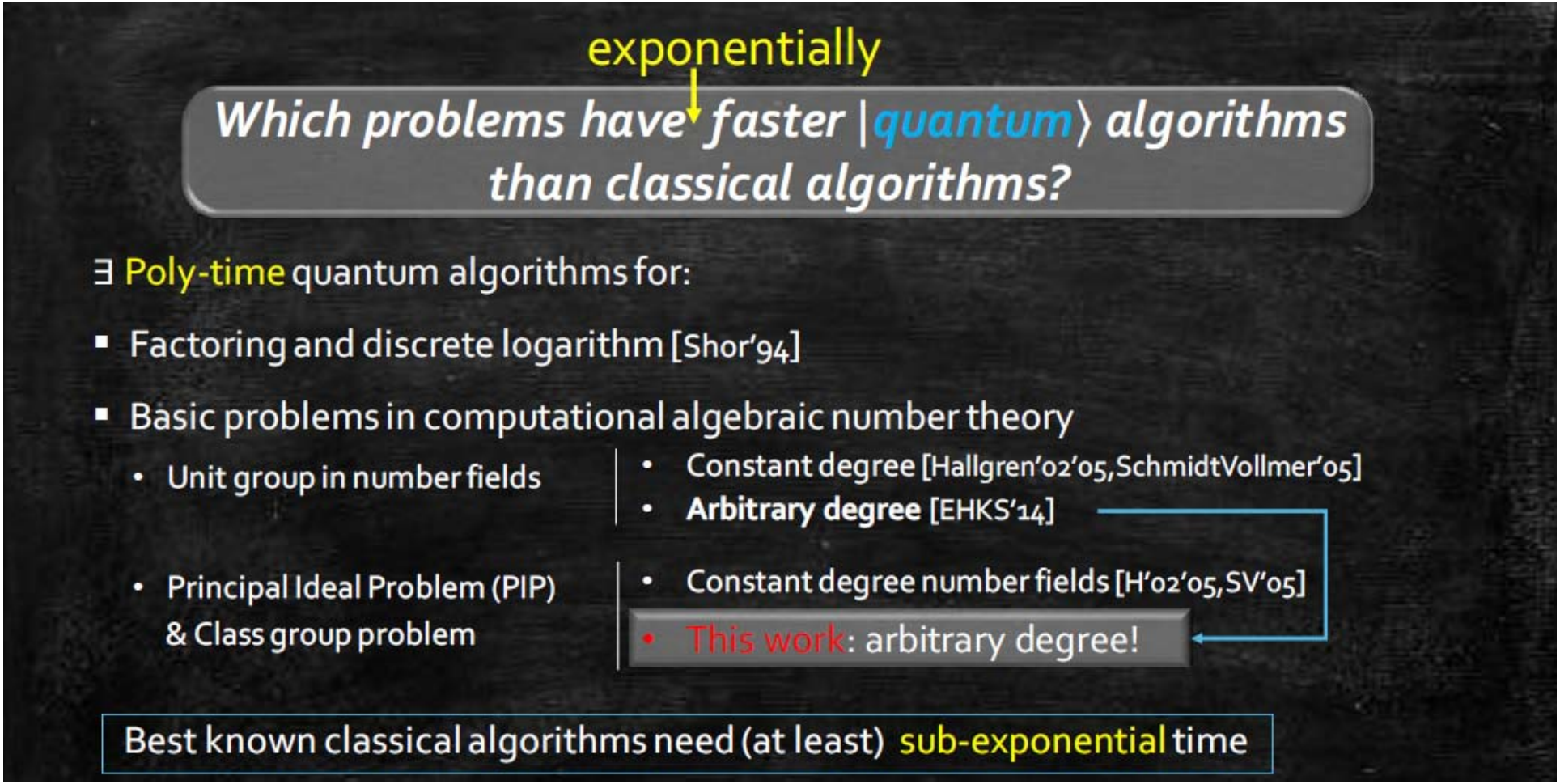
N.B.: $H$ is a Lattice

$L(B) = \{a_1 v_1 + \cdots + a_m v_m : a_i \in \mathbb{Z}\} \subseteq \mathbb{R}^m$

- Basis $B: \{v_i \in \mathbb{R}^n : i = 1, \ldots, m\}$
- $L$ has (infinitely) many bases



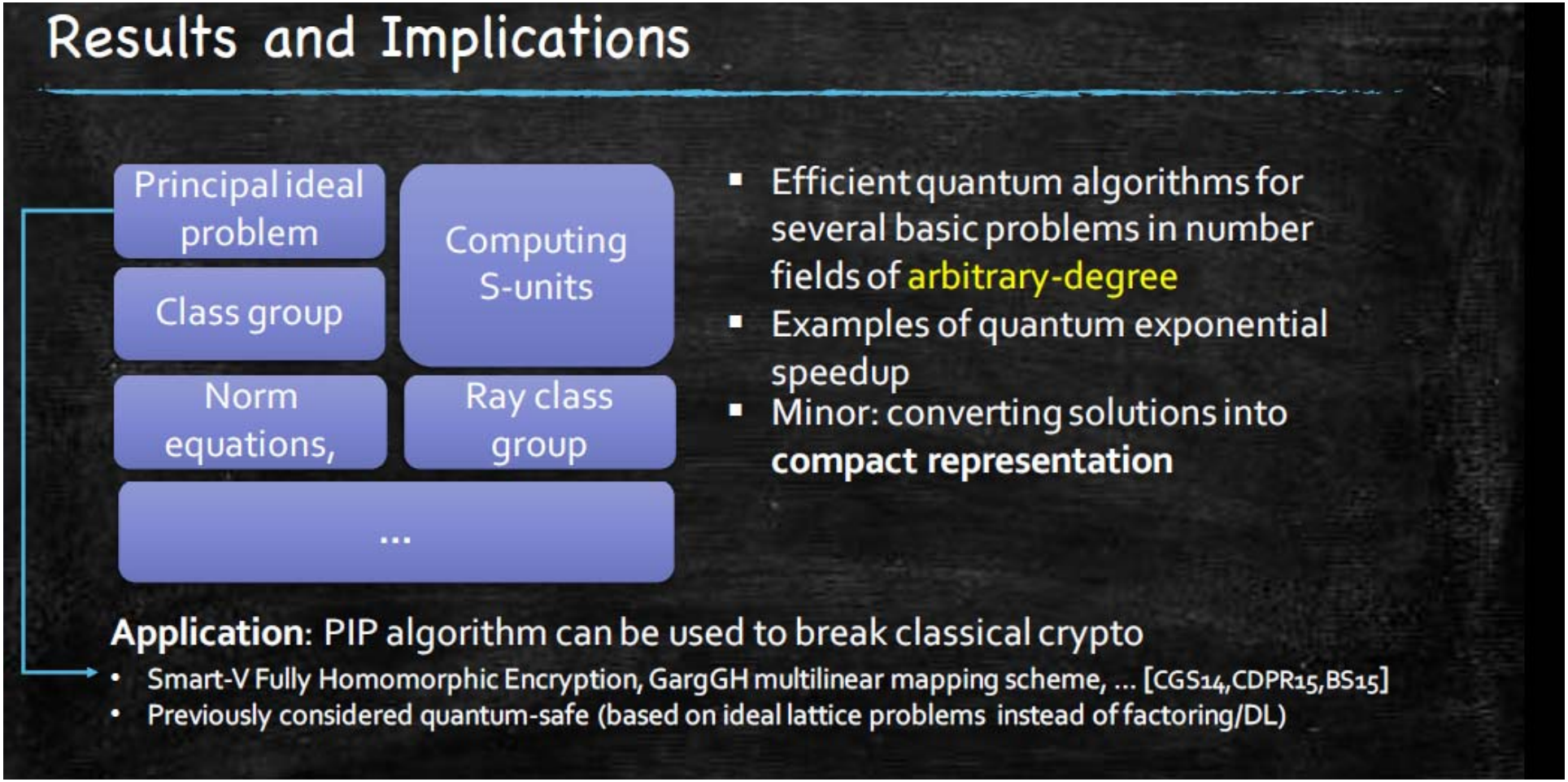

(borrowed from Fang Song, SODA 2016 talk)




(borrowed from Fang Song, SODA 2016 talk, "Efficient quantum algorithms for the principal ideal problem and class group problem in arbitrary-degree number fields", J.F. Biasse and F. Song)

UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing

# Results and Implications




(borrowed from Fang Song, SODA 2016 talk)

UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing

# QUANTUM SEARCHING

# Searching problem

Consider

$$f : \{0,1\}^n \rightarrow \{0,1\}$$

Given

$$U_f : |x\rangle|0\rangle \mapsto |x\rangle|f(x)\rangle$$

Find an $x$ satisfying $f(x) = 1$

# Application

Consider a 3-SAT formula

$$\Phi = C_1 \wedge C_2 \wedge \cdots \wedge C_M$$

$$C_j = (y_{j,1} \vee y_{j,2} \vee y_{j,2})$$

$$y_{j,k} \in \{x_1, x_2, \cdots, x_n, \overline{x}_1, \overline{x}_2, \cdots, \overline{x}_n\}$$

For a given assignment $\mathrm{x} = \mathrm{x}_1 \mathrm{x}_2 \cdots \mathrm{x}_n$

$$f_\Phi(\mathrm{x}) = \begin{cases} 1 & \text{if } \mathbf{x} \text{ satisfies } \Phi \\ 0 & \text{otherwise} \end{cases}$$

# Running times

Suppose there are t solutions to $f(x)=1$

Can find a solution to $f(x)=1$ using $O\left(\sqrt{\frac{2^n}{t}}\right)$ applications of $U_f$ and $\tilde{O}\left(\sqrt{\frac{2^n}{t}}\right)$ other operations (without knowing t).

# Parallelizing Brute-Force Search

Given M parallel quantum processors, finding an n-bit key requires time (measured in terms of function evaluations):

http://arxiv.org/abs/quant-ph/9711070

$$\Theta\left(\sqrt{\frac{2^n}{M}}\right)$$

e.g. Parallel quantum attacks on AES-128 (in terms of function evaluations):

| | Classical running time (1 processor) | Classical running time ($2^{40}$ processors) | Quantum running time (1 processor) | Quantum running time ($2^{40}$ processors) |
|---|---|---|---|---|
| AES-128 | $2^{128}$ | $2^{88}$ | $2^{64}$ | $2^{44}$ |

Des. Codes Cryptogr. (2015) 77:375–400
DOI 10.1007/s10623-015-0067-5

# Finding shortest lattice vectors faster using quantum search

**Thijs Laarhoven**[1] · **Michele Mosca**[2,3,4] · **Joop van de Pol**[5]

Can be applied to speed up parts of complex classical algorithms, e.g. finding short vectors in a lattice.

Received: 15 October 2014 / Revised: 12 March 2015 / Accepted: 16 March 2015 /
Published online: 14 April 2015


**Abstract** By applying a quantum search algorithm to various heuristic and provable sieve algorithms from the literature, we obtain improved asymptotic quantum results for solving the shortest vector problem on lattices. With quantum computers we can provably find a shortest vector in time $2^{1.799n+o(n)}$, improving upon the classical time complexities of $2^{2.465n+o(n)}$ of Pujol and Stehlé and the $2^{2n+o(n)}$ of Micciancio and Voulgaris, while heuristically we expect to find a shortest vector in time $2^{0.268n+o(n)}$, improving upon the classical time complexity of $2^{0.298n+o(n)}$ of Laarhoven and De Weger. These quantum complexities will be an important guide for the selection of parameters for post-quantum cryptosystems based on the hardness of the shortest vector problem.

UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing

# On Quantum RAM

Some quantum algorithms require poly(n) computational qubits and $\exp(n^c)$ “quantumly accessible” classical bits.

What is the cost of $\exp(n^c)$ “quantumly accessible” classical bits compared to $\exp(n^c)$ computational qubits?

For superpolynomially many queries, it’s not clear if there is much advantage. http://arxiv.org/abs/1502.03450

UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing

# Generalization: Amplitude Amplification

Consider any algorithm $A$ that successfully guesses a solution to $f(x)=1$ with probability $p$

Quantum Amplitude Amplification finds a solution to $f(x)=1$ using $O\left(\frac{1}{\sqrt{p}}\right)$ (quantum) applications of $A$ and of $U_f$

# Analysis

Let S = cost of implementing $A$ - “sampling” cost

Let C = cost of implementing $U_f$ - “checking” cost

Let p = probability that a sample is a solution.

A classical search would have expected cost $\frac{1}{p}(S+C)$

A quantum search would have expected cost $\frac{1}{\sqrt{p}}(S+C)$

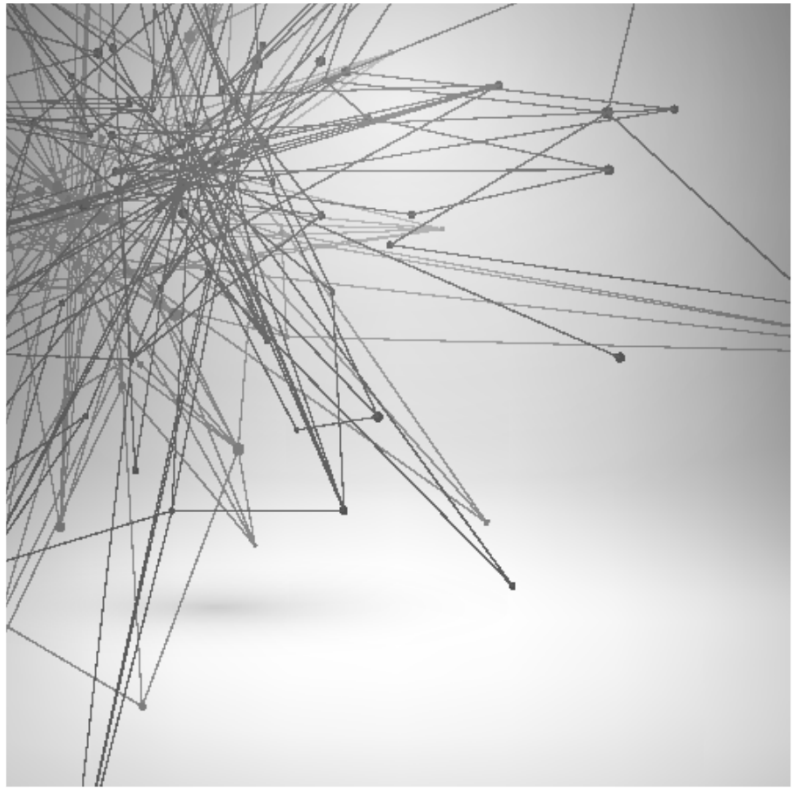
# Element Distinctness

- Consider $f : \{0,1\}^n \to X$
- Find $x \neq y$ such that $f(x) = f(y)$
- Classically (in the worst case) this takes $O(N)$ evaluations of $f$

# Element Distinctness

- Let $A$ sample $\sqrt{N}$ random elements $f(x_j)$
- Thus $p \approx \frac{1}{\sqrt{N}}$
- Checking if any of the samples are not distinct over the range of f can be done in time $\tilde{O}(\sqrt{N})$
- Thus $\frac{1}{\sqrt{p}}(S+C) \in \tilde{O}\left(N^{3/4}\right)$

# WALK-BASED QUANTUM SEARCHING



# Quantum walk algorithms

- Can generalize notion of classical random walks
- Can get up to quadratic speed-up for “mixing time”
- Can get up to an exponential speed-up for “hitting time” (“glued-trees” problem)
- Applications include:

Element distinctness, triangle-finding, element k-distinctness, AND-OR trees, MIN-MAX trees, etc.

# Analysis of search by walk

Let S = “set-up” cost
Let C = “checking” cost
Let U = “update” cost

Let ε = probability that a sample is a solution.
Let δ = spectral gap of random walk matrix

A classical search would have expected cost $S + \frac{1}{\varepsilon}\left(\frac{1}{\delta}U + C\right)$

A quantum search would have expected cost $S + \frac{1}{\sqrt{\varepsilon}}\left(\frac{1}{\sqrt{\delta}}U + C\right)$

# Element Distinctness by Quantum Walk

(Ambainis)

Set-up = sample $N^{2/3}$ elements $f(x_j)$

Check = check for non-distinct elements in the current sample

Update = remove one element and replace with an new random element

$$\delta \approx \frac{1}{N^{2/3}} \qquad \varepsilon \approx \frac{1}{N^{2/3}}$$

Quantum walk running time is $\tilde{O}\left(N^{2/3}\right)$

UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing

# Collision-Finding by Quantum Walk (Ambainis)

Set-up = sample $N^{1/3}$ elements $f(x_j)$

Check = check for a collision in the current sample

Update = remove one element and replace with an new random element

$$\delta \approx \frac{1}{N^{1/3}} \qquad \varepsilon \approx \frac{1}{N^{1/3}}$$

Quantum walk running time is $\tilde{O}\left(N^{1/3}\right)$

UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing

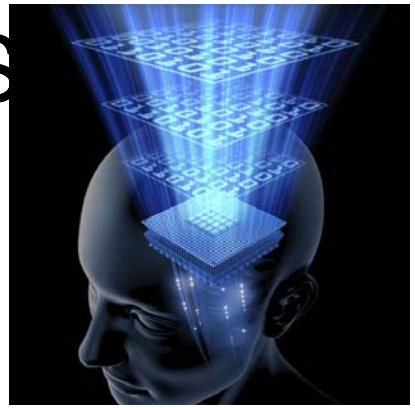
# Comparison to Classical Parallel Collision Finding

Classical parallel collision-finding heuristics using $O\left(N^{1/3}\right)$ processors find collisions in time $\tilde{O}\left(N^{1/6}\right)$

http://link.springer.com/article/10.1007/PL00003816 (van Oorschot-Wiener)
http://cr.yp.to/hash/collisioncost-20090823.pdf (Bernstein)
https://uwspace.uwaterloo.ca/handle/10012/6200 (Jeffery)

Can parallel quantum collision do better?

# OTHER ALGORITHMS AND ALGORITHIC PARADIGMS



# Hamiltonian simulation

Under appropriate conditions we can efficiently approximate some properties of $e^{iHt}|\phi\rangle$

One application, in combination with eigenvalue estimation and other tools, is to determine some properties of the solution to ("well-conditioned") **sparse linear equations** (by Harrow, Hassidim and Lloyd, 2008).

# And more…

- Adiabatic algorithms
- Topological algorithms
- Span programs
- Etc.

http://math.nist.gov/quantum/zoo/ *(maintained by S. Jordan)*



## Quantum algorithms for algebraic problems

Andrew M. Childs*

*Department of Combinatorics & Optimization and Institute for Quantum Computing*
*University of Waterloo, Waterloo, Ontario, Canada N2L 3G1*

Wim van Dam†

*Departments of Computer Science and Physics*
*University of California, Santa Barbara, California 93106, USA*

Quantum computers can execute algorithms that dramatically outperform classical computation. As the best-known example, Shor discovered an efficient quantum algorithm for factoring integers, whereas factoring appears to be difficult for classical computers. Understanding what other computational problems can be solved significantly faster using quantum algorithms is one of the major challenges in the theory of quantum computation, and such algorithms motivate the formidable task of building a large-scale quantum computer. This article reviews the current state of quantum algorithms, focusing on algorithms with superpolynomial speedup over classical computation, and in particular, on problems with an algebraic flavor.

PACS numbers: 03.67.Lx

Contents



*amchilds@uwaterloo.ca
†vandam@cs.ucsb.edu



Quantum Algorithms

Michele Mosca
Institute for Quantum Computing and Dept. of Combinatorics & Optimizati
University of Waterloo and St. Jerome's University,
and Perimeter Institute for Theoretical Physics
www.iqc.ca/ ~ mmosca/web

## Article Outline

Glossary





## Algorithms for Quantum Computers

Jamie Smith and Michele Mosca

### 1 Introduction

Quantum computing is a new computational paradigm created by reformulating information and computation in a quantum mechanical framework [30, 27]. Since the laws of physics appear to be quantum mechanical, this is the most relevant framework to consider when considering the fundamental limitations of information processing. Furthermore, in recent decades we have seen a major shift from just observing quantum phenomena to actually controlling quantum mechanical systems. We have seen the communication of quantum information over long distances, the "teleportation" of quantum information, and the encoding and manipulation of quantum information in many different physical media. We still appear to be a long way from the implementation of a large-scale quantum computer, however it is a serious goal of many of the world's leading physicists, and progress continues at a fast pace.

In parallel with the broad and aggressive program to control quantum mechanical systems with increased precision, and to control and interact a larger number of subsystems, researchers have also been aggressively pushing the boundaries of what useful tasks one could perform with quantum mechanical devices. These in-

Jamie Smith
Institute for Quantum Computing and Dept. of Combinatorics & Optimization
University of Waterloo,
with support from the Natural Sciences and Engineering Research Council of Canada
e-mail: ja5smith@iqc.ca

Michele Mosca
Institute for Quantum Computing and Dept. of Combinatorics & Optimization
University of Waterloo and St. Jerome's University,
and Perimeter Institute for Theoretical Physics,
with support from the Government of Canada, Ontario-MRI, NSERC, QuantumWorks, MITACS, CIFAR, CRC, ORF, and DTO-ARO
e-mail: mmosca@iqc.ca

# cryptoworks21.com


UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing

# Thank you!

Feedback welcome: mmosca@uwaterloo.ca


Canada Foundation for Innovation

Fondation canadienne pour l'innovation

evolution

CIFAR
CANADIAN INSTITUTE FOR ADVANCED RESEARCH

NSERC CRSNG

PERIMETER INSTITUTE FOR THEORETICAL PHYSICS

Ontario

CryptoWorks21

Canada

UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing