Lecture Notes in Computer Science 5106

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Yasuhito Kawano Michele Mosca (Eds.)

Theory of Quantum Computation, Communication, and Cryptography

Third Workshop, TQC 2008
Tokyo, Japan, January 30–February 1, 2008
Revised Selected Papers

Springer

Volume Editors

Yasuhito Kawano
NTT Communication Science Laboratories
3-1 Morinosato Wakamiya, Atsugi, Kanagawa 243-0198, Japan
E-mail: kawano@theory.brl.ntt.co.jp

Michele Mosca
Institute for Quantum Computing
University of Waterloo
Waterloo, Ontario N2L 3G1, Canada
E-mail: mmosca@iqc.ca
and
Perimeter Institute for Theoretical Physics
31 Caroline Street North, Waterloo, Ontario N2L 2Y5, Canada

Library of Congress Control Number: 2008938495

CR Subject Classification (1998): F, D, C.2, G.1-2, E.3, J.2

LNCS Sublibrary: SL 1 – Theoretical Computer Science and General Issues

ISSN 0302-9743
ISBN-10 3-540-89303-2 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-89303-5 Springer Berlin Heidelberg New York

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12443833 06/3180 5 4 3 2 1 0

Preface

The Workshop on Theory of Quantum Computation, Communication, and Cryptography (TQC) focuses on theoretical aspects of quantum computation, quantum communication, and quantum cryptography, which are part of a larger interdisciplinary field that casts information science in a quantum mechanical framework.

The third TQC was held from January 30 to February 1, 2008, at the University of Tokyo, Tokyo, Japan. It consisted of invited talks, contributed talks and a poster session. A selection of these contributors were invited to submit a paper to this *Lecture Notes in Computer Science* (LNCS) proceedings.

The field of quantum information processing is rapidly growing in depth and in breadth. TQC is a workshop dedicated to the presentation and discussion of original research. While most research in quantum information is published in a wide range of journals and conference proceedings in computer science, physics, mathematics and other traditional areas of science, there is a growing niche for high-quality journals and proceedings dedicated to research in quantum information. TQC is one of the first such conferences or workshops that has decided to publish a selection of the submissions in an official proceedings of the workshop to be published in the LNCS series.

We are extremely fortunate to have had the support and advice of our Program Committee (listed here) and are very grateful for all their hard work. We also appreciate the help of the following additional reviewers: Jean-Christian Boileau, Jop Briet, David Feder, Francois Le Gall, Hector Garcia, Tohya Hiroshima, and Casey Myers.

We also extend our sincere thanks to the local Organizing Committee for pulling together all the local and logistical aspects of the workshop so successfully.

Lastly, many thanks to NTT for sponsoring TQC 2008, to the University of Tokyo for their generous support, and to Springer for agreeing to publish these proceedings in the LNCS series.

May 2008

Yasuhito Kawano
Michele Mosca

Organization

Program Committee

Michele Mosca	Institute for Quantum Computing (UW), and Perimeter Institute, Waterloo, Chair
Richard Cleve	Institute for Quantum Computing (UW), and Perimeter Institute, Waterloo
Masahito Hayashi	JST ERATO-SORST/Tohoku University
Peter Høyer	University of Calgary
Hiroshi Imai	University of Tokyo/JST ERATO-SORST
Nobuyuki Imoto	Osaka University
Kazuo Iwama	Kyoto University
Richard Jozsa	University of Bristol
Yasuhito Kawano	NTT
Takeshi Koshiba	Saitama University
Hoi-Kwong Lo	University of Toronto
Igor L. Markov	University of Michigan
Mio Murao	University of Tokyo
Tatsuaki Okamoto	NTT
Masanao Ozawa	Tohoku University
Adam Smith	Pennsylvania State University
Barbara Terhal	IBM T.J. Watson Research Center
Guifre Vidal	University of Queensland
Shigeru Yamashita	Nara Institute of Science and Technology

Organizing Committee

Yasuhito Kawano	NTT, Chair
Go Kato	NTT
Yumi Nakajima	NTT
Yasuhiro Takahashi	NTT
Seiichiro Tani	NTT/JST ERATO-SORST

Table of Contents

Classical and Quantum Algorithms for Exponential Congruences

Wim van Dam[1] and Igor E. Shparlinski[2]

[1] Department of Computer Science, Department of Physics, University of California, Santa Barbara, CA 93106-5110, USA
vandam@cs.ucsb.edu
[2] Department of Computing, Macquarie University, NSW 2109, Australia
igor@ics.mq.edu.au

Abstract. We discuss classical and quantum algorithms for solvability testing and finding integer solutions x, y of equations of the form $af^x + bg^y = c$ over finite fields $\mathbb{F}_q$. A quantum algorithm with time complexity $q^{3/8}(\log q)^{O(1)}$ is presented. While still superpolynomial in $\log q$, this quantum algorithm is significantly faster than the best known classical algorithm, which has time complexity $q^{9/8}(\log q)^{O(1)}$. Thus it gives an example of a natural problem where quantum algorithms provide about a cubic speed-up over classical ones.

1 Introduction

Let $\mathbb{F}_q$ be a finite field of q elements and let $\mathbb{F}_q^*$ denote the multiplicative group of nonzero elements of $\mathbb{F}_q$. For $a, b, c, f, g \in \mathbb{F}_q^*$ we consider the equations

$$af^x + bg^y = c \tag{1}$$

in nonnegative integers x and y.

Equation (1) has a long history of study in number theory. In particular, it is dual closely related to the classical problem of finding $f, g \in \mathbb{F}_q$ for fixed a, b and x, y from the theory *cyclotomic classes*, see [2,11], which looks like a dual problem to studying Equation (1) but in fact, after a change of variables, become equivalent.

Furthermore, Equation (1) and variants of it also appeared in recent work of A. Lenstra and B. de Weger [8] and have been shown to be of cryptographic significance. In particular, the question about difficulty of finding solutions to Equation (1) has been discussed in [8] but now concrete results have been know before the present work.

In the theory of quantum computing the task of finding the solutions to Equation (1) is of importance when trying to solve the *hidden subgroup problem* for semi-direct product groups $\mathbb{Z}/N \rtimes \mathbb{Z}/p$ with $p = \Theta(\sqrt{N})$, see [1], where, as usual, $A = \Theta(B)$ means that $A = O(B)$ and $B = O(A)$ (hereafter all implied constants are absolute). Furthermore it is also natural to consider this problem as a generalization of the discrete logarithm problem in $\mathbb{F}_q$, which can be solved efficiently using Shor's algorithm [10].

Y. Kawano and M. Mosca (Eds.): TQC 2008, LNCS 5106, pp. 1–10, 2008.

In this article we use some number theoretic tools to design classical and quantum algorithms that are more efficient than the brute force search (but unfortunately still have a running time exponential in the input size $\log q$). We use our classical algorithm to measure the level of improvement that can be achieved by allowing quantum algorithms. Ignoring $\log q$ terms, the classical algorithm that we present here has complexity $O^*(q^{9/8})$ (which seems to be the best known) whereas we also present a quantum algorithm with complexity $O^*(q^{3/8})$, where, as usual, $A = O^*(B)$ means that $A = B(\log B)^{O(1)}$. In particular, it gives an example of a natural problem where quantum algorithms provide an asymptotically cubic speed-up over classical ones.

Certainly if f or g is a primitive root, which generates all of $\mathbb{F}_q^*$, then the problem is not harder than the discrete logarithm problem. Moreover, in general our results suggest that finding solutions to Equation (1) becomes easier in case f or g is of large order, but still it appears to be much harder than the discrete logarithm problem.

2 The Number of Solutions to the Equation

2.1 The Worst Case

Here we use bounds of multiplicative character sums over finite fields to show that if the orders of f and g are large enough, then Equation (1) has a solution with at least one reasonably small component x or y.

Lemma 1. *Let $a, b, c \in \mathbb{F}_q^*$ and let f and $g \in \mathbb{F}_q$ be of multiplicative orders s and t, respectively. Then for any positive integer $r \le t$, the equation $af^x + bg^y = c$ has $rs/(q-1) + O(q^{1/2}\log q)$ solutions in nonnegative integers x and y with $x \in \{0, \ldots, s-1\}$ and $y \in \{0, \ldots, r-1\}$.*

Proof. Let $k = (q-1)/s$ and let $\mathcal{X}_k$ be the group of all k multiplicative characters $\chi : \mathbb{F}_q \to \mathbb{C}$ of order k, that is, $\chi^k = \chi_0$, the principal character, for any $\chi \in \mathcal{X}_k$ (see [9]). Note that for all non-empty $\mathcal{X}_k$ this group contains k elements. For any $u \in \mathbb{F}_q$ we have

$$\frac{1}{k}\sum_{\chi\in\mathcal{X}_k}\chi(u) = \begin{cases}1, & \text{if } u^s = 1,\\ 0, & \text{otherwise.}\end{cases}$$

Noting that $u \in \mathbb{F}_q$ belongs to the group generated by f if and only if $u^s = 1$, we derive that the number $N_{a,b,c}(r,s)$ of solutions to Equation (1) with $x \in \{0, \ldots, s-1\}$ and $y \in \{0, \ldots, r-1\}$ equals

$$N_{a,b,c}(r,s) = \sum_{y=0}^{r-1}\frac{1}{k}\sum_{\chi\in\mathcal{X}_k}\chi(a^{-1}(c-bg^y)).$$

Changing the order of summation and separating the term r/k corresponding to the principal character χ_0 we obtain

$$\left|N_{a,b,c}(r,s) - \frac{r}{k}\right| \le \frac{1}{k}\sum_{\chi\in\mathcal{X}_k\setminus\{\chi_0\}}\chi(a^{-1})\sum_{y=0}^{r-1}\chi(c-bg^y).$$

By [12, Theorem 3] (see also [5]) each summation over y is bounded by $O(q^{1/2}\log q)$ (provided $1 \le r \le t$), hence we have

$$N_{a,b,c}(r,s) = \frac{r}{k} + O(q^{1/2}\log q),$$

which concludes the proof. □

From Lemma 1 we can immediately conclude the following.

Corollary 1. *Let $a, b, c \in \mathbb{F}_q^*$ and let f and $g \in \mathbb{F}_q$ be of multiplicative orders s and t, respectively. There exists an absolute constant $C > 0$ such that if for some integer r we have*

$$Cq^{3/2}s^{-1}\log q \le r \le t,$$

then the equation $af^x + bg^y = c$ has a solution in integers x and y with $x \in \{0,\dots,s-1\}$ and $y \in \{0,\dots,r-1\}$.

We remark that the constant C in Corollary 1 is independent of all variables involved (a, b, c, f, g and q) and that it is effectively computable. This result reduces the number of (x, y) pairs that has to be searched for a solution to Equation (1). In Sections 3.1 and 4.1 we show how this reduction can be used to construct non-trivial worst case algorithms, both classical and quantum.

2.2 The Typical Case

To solve the equation $af^x + bg^y = c$ for typical $c \in \mathbb{F}_q$ we now show that for almost all $c \in \mathbb{F}_q^*$ the results of Corollary 1 can be improved, which in turn will yield better average case algorithms for the central problem.

Lemma 2. *Let $a, b, c \in \mathbb{F}_q^*$ and let f and $g \in \mathbb{F}_q$ be of multiplicative orders s and t, respectively. For any positive integer $r \le t$ and $\delta > 0$, for all but q/δ^2 elements $c \in \mathbb{F}_q^*$, the equation $af^x + bg^y = c$ has $rs/q + \vartheta\delta\sqrt{r}$ solutions in nonnegative integers x and y with $x \in \{0,\dots,s-1\}$, $y \in \{0,\dots,r-1\}$ and $-1 < \vartheta < 1$.*

Proof. Let $\psi : \mathbb{F}_q \to \mathbb{C}$ be a nontrivial additive character. We recall that for for any $u \in \mathbb{F}_q$ we have

$$\frac{1}{q}\sum_{\lambda\in\mathbb{F}_q}\psi(\lambda u) = \begin{cases} 1, & \text{if } v = 0, \\ 0, & \text{if } v \in \mathbb{F}_q^*. \end{cases}$$

As in the proof of Lemma 1 we use $N_{a,b,c}(r,s)$ to denote the number of solutions to Equation (1) with $x \in \{0,\dots,s-1\}$ and $y \in \{0,\dots,r-1\}$. We have

$$\begin{aligned} N_{a,b,c}(r,s) &= \sum_{x=0}^{s-1}\sum_{y=0}^{r-1}\frac{1}{q}\sum_{\lambda\in\mathbb{F}_q}\psi(\lambda(af^x + bg^y - c)) \\ &= \frac{sr}{q} + \frac{1}{q}\sum_{\lambda\in\mathbb{F}_q^*}\sum_{x=0}^{s-1}\sum_{y=0}^{r-1}\psi(\lambda(af^x + bg^y - c)), \end{aligned}$$

which averaged over $c \in \mathbb{F}_q$ equals sr/q. To calculate the variance from its average, we look at the value defined by

$$W_{a,b}(r,s) = \sum_{c \in \mathbb{F}_q} \left(N_{a,b,c}(r,s) - \frac{rs}{q} \right)^2,$$

which equals

$$\frac{1}{q^2} \sum_{c \in \mathbb{F}_q} \sum_{\lambda_1, \lambda_2 \in \mathbb{F}_q^*} \sum_{x_1, x_2 = 0}^{s-1} \sum_{y_1, y_2 = 0}^{r-1} \psi(\lambda_1(af^{x_1} + bg^{y_1} - c) + \lambda_2(af^{x_2} + bg^{y_2} - c))$$

$$= \frac{1}{q^2} \sum_{\lambda_1, \lambda_2 \in \mathbb{F}_q^*} \sum_{x_1, x_2 = 0}^{s-1} \psi(a(\lambda_1 f^{x_1} + \lambda_2 f^{x_2})) \sum_{y_1, y_2 = 0}^{r-1} \psi(b(\lambda_1 g^{y_1} + \lambda_2 g^{y_2})) \times \sum_{c \in \mathbb{F}_q} \psi(c(\lambda_2 + \lambda_1)).$$

The inner sum over c vanishes unless $\lambda_1 = -\lambda_2$ (in which case it is q) and therefore

$$W_{a,b}(r,s) = \frac{1}{q} \sum_{\lambda \in \mathbb{F}_q^*} \sum_{x_1, x_2 = 0}^{s-1} \psi(a\lambda(f^{x_1} - f^{x_2})) \sum_{y_1, y_2 = 0}^{r-1} \psi(b\lambda(g^{y_1} - g^{y_2}))$$

$$= \frac{1}{q} \sum_{\lambda \in \mathbb{F}_q^*} \left| \sum_{x=0}^{s-1} \psi(a\lambda f^x) \right|^2 \left| \sum_{y=0}^{r-1} \psi(b\lambda g^y) \right|^2.$$

It is well known that

$$\left| \sum_{x=0}^{s-1} \psi(a\lambda f^x) \right|^2 \leq q^{1/2},$$

for example, this follows from [9, Theorem 8.78] taken with $k = 1$ and $g^0, g^1, \ldots, g^{s-1}$ the impulse response sequence (it can also be derived from the bound of Gauss sums, see [9, Theorem 5.32]). Therefore

$$W_{a,b}(r,s) \leq \sum_{\lambda \in \mathbb{F}_q} \left| \sum_{x=0}^{s-1} \psi(a\lambda f^x) \right|^2$$

(note that we have added $\lambda = 0$ into the last sum). We also have the straightforward equality

$$\sum_{\lambda \in \mathbb{F}_q} \left| \sum_{y=0}^{r-1} \psi(b\lambda g^y) \right|^2 = \sum_{\lambda \in \mathbb{F}_q} \left| \sum_{y=0}^{r-1} \psi(\lambda g^y) \right|^2 = qr$$

(essentially, this is Parseval's identity, i.e. we used the unitarity of the Fourier transformation over $\mathbb{F}_q$ on the characteristic vector of the set $\{g^0, \ldots, g^{r-1}\}$) and thus

$$W_{a,b}(r,s) = \sum_{c \in \mathbb{F}_q} \left| N_{a,b,c}(r,s) - \frac{rs}{q} \right|^2 \le qr.$$

Hence, for any $\delta > 0$, the violation

$$\left| N_{a,b,c}(r,s) - \frac{rs}{q} \right| \ge \delta \sqrt{r}$$

holds for no more than q/δ^2 values of $c \in \mathbb{F}_q^*$. □

Using $\delta = \sqrt{\log q}$ in Lemma 2, we see that for all but $q/\log q = o(q)$ elements $c \in \mathbb{F}_q^*$ the equation $af^x + bg^y = c$ has $rs/q + \vartheta\sqrt{r \log q}$ solutions in $x \in \{0, \ldots, s-1\}$, $y \in \{0, \ldots, r-1\}$ with $-1 < \vartheta < 1$. Therefore we can immediately conclude the following.

Corollary 2. *Let $a, b, c \in \mathbb{F}_q^*$ and let f and $g \in \mathbb{F}_q$ be of multiplicative orders s and t, respectively. If for some integer r we have*

$$q^2 s^{-2} \log q \le r \le t,$$

then for all but $o(q)$ elements $c \in \mathbb{F}_q^*$, the equation $af^x + bg^y = c$ has a solution in integers x and y with $x \in \{0, \ldots, s-1\}$ and $y \in \{0, \ldots, r-1\}$.

3 Classical Algorithms

3.1 Worst Case Classical Algorithm

We start with a classical deterministic algorithm that is more efficient than brute search.

Theorem 1. *Let $a, b, c, f, g \in \mathbb{F}_q^*$. One can either find a solution $x, y \in \mathbb{Z}_{\ge 0}$ of the equation $af^x + bg^y = c$ or decide that it does not have a solution in deterministic time $q^{9/8}(\log q)^{O(1)}$ on a classical computer.*

Proof. Using a standard deterministic factorization algorithm, we factor $q-1$ and find the orders s and t of f and g in time $q^{1/2}(\log q)^{O(1)}$. Assume without loss of generality that $s \ge t$ (otherwise of the roles of s and t are reversed in the proof below). Let C be the constant of Corollary 1 and define

$$r = \left\lceil C q^{3/2} s^{-1} \log q \right\rceil. \tag{2}$$

By Corollary 1, if $r \le t$ then the central equation $af^x + bg^y = c$ is solvable for $(x, y) \in \{0, \ldots, s-1\} \times \{0, \ldots, r-1\}$. Otherwise, if $r > t$, there may or may not be a solution with $(x, y) \in \{0, \ldots, s-1\} \times \{0, \ldots, t-1\}$. As a result, the following algorithm proves the theorem.

If $r \leq t$ then for every $y \in \{0, \ldots, r-1\}$ we evaluate $a^{-1}(c - bg^x)$ and then try to compute its discrete logarithm to base f, that is, an integer x with $f^x = a^{-1}(c - bg^y)$, in deterministic time $s^{1/2}(\log q)^{O(1)}$, see [4, Section 5.3]. When found, the algorithm outputs (x, y) and terminates. The condition $t \geq r$ and assumption $s \geq t$ implies for s:

$$s^2 \geq st \geq sr \geq Cq^{3/2} \log q,$$

which gives for the time complexity of this case

$$r \cdot s^{1/2}(\log q)^{O(1)} = q^{3/2}s^{-1/2}(\log q)^{O(1)} \leq q^{9/8}(\log q)^{O(1)}.$$

If $r > t$ we perform the same procedure for every $y \in \{0, \ldots, t-1\}$. If none of the y yield a solution, the algorithm reports that the central equation has no solution. In this case, the condition $t < r$ implies that

$$st < sr \leq Cq^{3/2} \log q$$

and since $t \leq s$, the time complexity of this case is also bounded by

$$t \cdot s^{1/2}(\log q)^{O(1)} \leq (st)^{3/4}(\log q)^{O(1)} \leq q^{9/8}(\log q)^{O(1)},$$

which completes the proof. □

It is natural to ask whether one can design a faster probabilistic algorithm. For some fields this is indeed possible due to the existence of subexponential algorithms for computing discrete logarithms, see [4, Section 6.4]. However in general probabilistic algorithms do not seem to give any significant advantage for our problem.

3.2 Typical Case Classical Algorithm

Similarly, using Corollary 2 instead of Corollary 1 and repeating the arguments of the proof of Theorem 1 with

$$r = \left\lceil q^2 s^{-2} \log q \right\rceil \tag{3}$$

we obtain that for almost all c a stronger result than Theorem 1 holds.

Theorem 2. *Let $a, b, c, f, g \in \mathbb{F}_q^*$. For all but $o(q)$ elements $c \in \mathbb{F}_q^*$, one can either find a solution $x, y \in \mathbb{Z}_{\geq 0}$ of the equation $af^x + bg^y = c$ or decide that it does not have a solution in deterministic time $q(\log q)^{O(1)}$ on a classical computer.*

4 Quantum Algorithms

4.1 Worst Case Quantum Algorithms

On a quantum computer one has the advantage that calculating discrete logarithms can be done efficiently in time $(\log q)^{O(1)}$. In combination with the quadratic speed-up of quantum searching this gives the following quantum algorithm for the central problem. We start with an algorithm that works for *any* f and g.

Theorem 3. *Let $a, b, c, f, g \in \mathbb{F}_q^*$. One can either find a solution $x, y \in \mathbb{Z}_{\geq 0}$ of the equation $af^x + bg^y = c$ or decide that it does not have a solution in time $q^{3/8}(\log q)^{O(1)}$ on a quantum computer.*

Proof. We use Shor's algorithm [10] to compute s and t in polynomial time. Without loss of generality we assume that $s \geq t$ and we define r by Equation (2) as in the proof of Theorem 1. A polynomial time quantum subroutine $\mathcal{S}(y)$ is constructed that, using Shor's discrete logarithm algorithm [10], for a given y either finds and returns the integer x with $f^x = a^{-1}(c - bg^x)$ or reports that no such x exists.

If $r \leq t$, then, using Grover's search algorithm [6], we search the subroutines $\mathcal{S}(y)$ for all $y \in \{0, \ldots, r-1\}$ in time

$$r^{1/2}(\log q)^{O(1)} = q^{3/4}s^{-1/2}(\log q)^{O(1)} \leq q^{3/8}(\log q)^{O(1)}.$$

If $r > t$, we search the $\mathcal{S}(y)$ for all $y \in \{0, \ldots, t-1\}$ in time

$$t^{1/2}(\log q)^{O(1)} \leq (st)^{1/4}(\log q)^{O(1)} \leq q^{3/8}(\log q)^{O(1)}.$$

As in the proof of Theorem 1, we conclude that due to our choice of r we either find a solution to Equation (1) or conclude that there is no solution. □

We now show that if f and g are of large order then there is a more efficient quantum algorithm.

Theorem 4. *Let $a, b, c, f, g \in \mathbb{F}_q^*$ and let f and g be of multiplicative orders s and t, respectively. There is an absolute constant C such that if*

$$st > Cq^{3/2}(\log q)^{1/2}$$

then one can either find a solution $x, y \in \mathbb{Z}_{\geq 0}$ of the equation $af^x + bg^y = c$ or decide that it does not have a solution in time $q^{1/2}(st)^{-1/4}(\log q)^{O(1)}$ on a quantum computer.

Proof. Assume without loss of generality that $s \geq t$. It follows from the condition of the theorem and Lemma 1 that for some appropriate constant C and

$$r = \left\lfloor Cq^{3/2}s^{-1}(\log q)^{1/2} \right\rfloor \leq t$$

there are

$$\frac{rs}{q-1} + O(q^{1/2}\log q) \geq \frac{rs}{2q}$$

solutions to Equation (1) with $x \in \{0, \ldots, s-1\}$ and $y \in \{0, \ldots, r-1\}$.

We now use the version of Grover's search algorithm as described in [3] that finds one out of m matching items in a set of size r using only $O(\sqrt{r/m})$ queries. Here we search the subroutines $\mathcal{S}(y)$ for all $y \in \{0, \ldots, r-1\}$ with the promise (which follows from Lemma 1) that there are $m = rs/(q-1) + O(q^{1/2}\log q)$ solutions (x, y). Because for each value y there can be at most one value

$x \in \{0, \ldots, s-1\}$ such that $af^x + bg^y = c$ there are m different values y for which $\mathcal{S}$ will report a solution x, hence a solution will be found in time

$$(r/m)^{1/2}(\log q)^{O(1)} = q^{1/2}s^{-1/2}(\log q)^{O(1)}.$$

Since $s \geq (st)^{1/2}$, this concludes the proof. □

In particular, the running time of the algorithm of Theorem 4 is upper bounded by $O(q^{1/8}(\log q)^{O(1)})$.

4.2 Typical Case Quantum Algorithms

Similarly to the classical case, for almost all $c \in \mathbb{F}_q$ stronger results than those of Theorems 3 and 4 are possible. For example, defining again r by Equation (3) and arguing as in the proof of Theorem 3, we obtain the following result.

Theorem 5. *Let $a, b, c, f, g \in \mathbb{F}_q^*$. For all but $o(q)$ elements $c \in \mathbb{F}_q^*$, one can either find a solution $x, y \in \mathbb{Z}_{\geq 0}$ of the equation $af^x + bg^y = c$ or decide that it does not have a solution in time $q^{1/3}(\log q)^{O(1)}$ on a quantum computer.*

Finally, taking

$$r = \left\lfloor q^2 s^{-2} \log q \right\rfloor$$

and using Lemma 1 in the argument of the proof of Theorem 4, we see that for almost all $c \in \mathbb{F}_q$ the complexity estimate of Theorem 4 holds for a wider range of s and t.

Theorem 6. *Let $a, b, c, f, g \in \mathbb{F}_q^*$ and let f and g be of multiplicative orders s and t, respectively. For all but $o(q)$ elements $c \in \mathbb{F}_q^*$, if*

$$st > q^{4/3}(\log q)^{2/3}$$

then one can either find a solution $x, y \in \mathbb{Z}_{\geq 0}$ of the equation $af^x + bg^y = c$ or decide that it does not have a solution in time $q^{1/2}(st)^{-1/4}(\log q)^{O(1)}$ on a quantum computer.

5 Connection with the Hidden Subgroup Problem

The *pretty good measurement* approach [1] to the Hidden Subgroup Problem (HSP) over the non-abelian group $\mathbb{Z}/q \rtimes \mathbb{Z}/p$ with q a prime and $q/p^2 = (\log q)^{O(1)}$ shows that the HSP can be solved efficiently on a quantum computer if one can efficiently solve the equation $af^x + bf^y = c$, where f has multiplicative order p in $\mathbb{Z}/q$. All algorithms presented in this article have superpolynomial complexity in $\log q$ and hence fall short of this goal.

For this restricted problem with $f = g$ and f of order $p \approx \sqrt{q}$, there are p^2 possible solutions (x, y), hence even a classical algorithm has $O^*(q)$ time complexity instead of the $O^*(q^{9/8})$ of Theorem 1. Quantum mechanically, one can 'Grover search' the set of solutions $x \in \{0, \ldots, p-1\}$ in time $O^*(q^{1/4})$, which, although better than the $O^*(q^{3/8})$ of Theorem 3, is still far from polynomial in $\log q$.

6 Remarks and Open Problems

We remark that in some finite fields classical subexponential probabilistic algorithms are possible for the discrete logarithm problem, see [4, Section 6.4]. In such fields, a version of Theorem 1 can be obtained with an algorithm that runs in probabilistic time $q^{3/4+o(1)}$, which is still much slower that the quantum algorithm of Theorems 3 and 4. We note that although over the last several years fast heuristic algorithms for the discrete logarithm problem have been designed to work over any finite field, rigorous subexponential algorithms are know only for fields of special types (such as prime fields $\mathbb{F}_p$ or binary fields $\mathbb{F}_{2^m}$), see [4, Section 6.4] for more details. Clearly using probabilistic algorithms one can also get additional speed up in the classical case if the multiplicative orders s and t are large (similar to Theorems 4 and 6).

To try to strengthen the presented results one can consider the analogue to Equation (1) for elliptic curves $\mathbb{E}$ over $\mathbb{F}_q$. For example, given two $\mathbb{F}_q$-rational points $F, G \in \mathbb{E}(\mathbb{F}_q)$ and the values $a, b, c \in \mathbb{F}_q$ one can ask for solutions to the equation

$$a \cdot x([u]F) + b \cdot x([v]G) = c$$

in integers u and v (where $x(Q)$ denotes the x-coordinate of a point $Q \in \mathbb{E}(\mathbb{F}_q)$ in a fixed affine model of $\mathbb{E}$ and $[n]Q$ denotes the n-fold sum $Q \oplus Q \oplus \cdots \oplus Q$ in the group of $\mathbb{E}$). Using bounds of character sums over subgroups of elliptic curves, see [7], one can obtain full analogues of our results (in fact at the cost of only typographical changes). This case is interesting since in the classical scenario even heuristic subexponential algorithms are not known.

But above of this all, it still remains an open problem whether or not there exist efficient quantum algorithms that run in time $(\log q)^{O(1)}$ for the determining the integer solutions x, y to the equation $af^x + bg^y = c$ and even the more restricted version $af^x + bf^y = c$ over $\mathbb{F}_q$.

Acknowledgments. The authors are grateful to Michele Mosca for useful and encouraging discussions.

This work was initiated during a very pleasant visit by I.S. to the University of California at Santa Barbara whose hospitality is gratefully acknowledged. W.v.D. is supported by the Disruptive Technology Office (DTO) under Army Research Office (ARO) contract number W911NF-04-R-0009 and the NSF CAREER award no. 0803963; I.S. is supported by ARC grant DP0556431.

References

1. Bacon, D., Childs, A.M., van Dam, W.: From optimal measurement to efficient quantum algorithms for the hidden subgroup problem over semidirect product groups. In: Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2005), pp. 469–478 (2005)
2. Berndt, B., Evans, R., Williams, K.S.: Gauss and Jacobi Sums. Canadian Mathematical Society Series of Monographs and Advanced Texts, vol. 21. John Wiley & Sons, Chichester (1998)

3. Boyer, M., Brassard, G., Høyer, P., Tapp, A.: Tight bounds on quantum searching. Fortschritte der Physik 46, 493–505 (1998)
4. Crandall, R., Pomerance, C.: Prime numbers: A computational perspective. Springer, Berlin (2005)
5. Dobrowolski, E., Williams, K.S.: An upper bound for the sum $\sum_{n=a+1}^{a+H} f(n)$ for a certain class of functions f. Proceedings of the American Mathematical Society 114, 29–35 (1992)
6. Grover, L.: A fast quantum-mechanical algorithm for database search. In: Proceedings of the 28th Annual ACM Symposium on Theory of Computing (STOC 1996), pp. 212–219 (1996)
7. Kohel, D.R., Shparlinski, I.E.: Exponential sums and group generators for elliptic curves over finite fields. In: Bosma, W. (ed.) ANTS 2000. LNCS, vol. 1838, pp. 395–404. Springer, Heidelberg (2000)
8. Lenstra, A., de Weger, B.: On the possibility of constructing meaningful hash collisions for public keys. In: Boyd, C., González Nieto, J.M. (eds.) ACISP 2005. LNCS, vol. 3574, pp. 267–279. Springer, Heidelberg (2005)
9. Lidl, R., Niederreiter, H.: Finite Fields. Encyclopedia of Mathematics and Its Applications, vol. 20. Cambridge University Press, Cambridge (1997)
10. Shor, P.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM Journal on Computing 26, 1484–1509 (1997)
11. Storer, T.: Cyclotomy and Difference Sets. Lectures in Advanced Mathematics. Markham Publishing Company (1967)
12. Yu, H.B.: Estimates of character sums with exponential function. Acta Arithmetica 97, 211–218 (2001)

Quantum Algorithms for Evaluating Min-Max Trees

Richard Cleve[1,2], Dmitry Gavinsky[1], and D. L. Yonge-Mallo[1]

[1] David R. Cheriton School of Computer Science and Institute for Quantum Computing, University of Waterloo
[2] Perimeter Institute for Theoretical Physics

Abstract. We present a bounded-error quantum algorithm for evaluating Min-Max trees with $N^{\frac{1}{2}+o(1)}$ queries, where N is the size of the tree and where the allowable queries are comparisons of the form $[x_j < x_k]$. This is close to tight, since there is a known quantum lower bound of $\Omega(N^{\frac{1}{2}})$.

A Min-Max tree is a tree whose internal nodes are *minimum* and *maximum* gates, at alternating levels, and whose leaves are values from some underlying ordered set. The size N of such a tree is the number of its leaves, whose values are referred to as $x_1, \ldots, x_N$. The value of a Min-Max tree is the value of its root, a function of $x_1, \ldots, x_N$. In the *input value* query model, queries explicitly access the values of the leaves. In the *comparison* query model, the values of $x_1, \ldots, x_N$ are not directly accessible; rather, queries are comparisons of the form $[x_j < x_k]$. In this latter model, the appropriate output is any $j \in \{1, \ldots, N\}$ such that x_j is the value of the tree. The comparison query model is commonly used in the analysis of classical parallel algorithms for searching and sorting.

Note that, when the ordered set is $\{0, 1\}$, a Min-Max tree reduces to an And-Or tree. This implies that Barnum and Saks's lower bound of $\Omega(N^{\frac{1}{2}})$ [3] for the quantum query complexity of And-Or trees applies to Min-Max trees.

Recent results initiated by Farhi *et al.* have shown that quantum algorithms can evaluate all And-Or trees with order $N^{\frac{1}{2}+o(1)}$ queries [2,6,7,9]. We show that these results carry over to Min-Max trees in both the input value model and the comparison model.

Let $W(N)$ be the query complexity for And-Or trees of size N. We show that Min-Max trees can be evaluated with $O(W(N)\log(N))$ queries in both the input value model and the comparison model. Our algorithm combines the results on And-Or trees in Refs. [2,7] with the lemma below and Grover's search algorithm [10].

Lemma 1. *Let $\mathcal{T}$ be a* Min-Max *tree with inputs $x_1, x_2, \ldots, x_N$. Let $\mathcal{T}^v$ be an* And-Or *tree with identical structure to $\mathcal{T}$, but with* And *and* Or *gates in place of* Min *and* Max *gates (respectively), and with the k^{th} input assigned to* 1 *if and only if $x_k \geq v$. Then* $\text{value}(\mathcal{T}^v) = 1$ *if and only if* $\text{value}(\mathcal{T}) \geq v$.

Y. Kawano and M. Mosca (Eds.): TQC 2008, LNCS 5106, pp. 11–15, 2008.

Lemma 1 is easy to prove by induction. It implies that, if the underlying ordered set is a numerical range of size $N^{O(1)}$, then the tree can be evaluated in $\log(N)$ stages by a simple binary search. Each stage can be implemented with $O(W(N)\log\log(N))$ queries, which reflects the cost of evaluating an AND-OR tree amplified so that its error probability is $O(1/\log(N))$. The result is an $O(W(N)\log(N)\log\log(N))$ query algorithm.

A complication arises in performing such a binary search in the comparison model, where it is not possible to directly compute the midpoint of an interval like $[x_j, x_k]$. Problems can also arise in the input value model when the numerical range is too large: the binary search may not converge in a logarithmic number of steps. For this reason, we avoid the standard binary search approach where a midpoint is chosen as a pivot. Instead, we take a *random* input value among those that lie within a current interval as our pivot. What is noteworthy about this simple approach is that *it does not work efficiently in the classical case*: given an interval $[x_j, x_k]$, finding an interior point is as hard as searching, which can cost $\Omega(N)$ queries to do even once [4]. In the setting of *quantum* algorithms, we can utilize Grover's search algorithm [5,10] which costs $O(\sqrt{N})$.

As an aside, we note that there is a classical reduction from MIN-MAX trees to AND-OR trees that yields an $O(N^{0.753})$ query algorithm for balanced MIN-MAX trees [11]. We can use that reduction with an $N^{\frac{1}{2}}$ query *quantum* algorithm for balanced AND-OR trees; however, the resulting algorithm for MIN-MAX costs $\Omega(N^{0.58})$, due to the recursive structure of the algorithm[1]. Our alternate approach yields exponent $\frac{1}{2} + o(1)$ and is not restricted to balanced trees.

What follows is a description of our algorithm with the analysis of its error. For convenience, let $\perp$ and $\top$ be such that $x_\perp < x_j$ and $x_\top > x_j$ for any $j \in \{1, \ldots, N\}$ and let c be a constant.

QUANTUM MIN-MAX TREE EVALUATION

1. Let $\gamma \leftarrow \perp$ and $\delta \leftarrow \top$, and initialize the stack.
2. Repeat the following steps for $c\log(N)$ iterations, then go to Step 3:
   (a) Find a random pivot:
   Call the quantum search subroutine to find a random pivot index j with $x_\gamma < x_j < x_\delta$. If no value is found, go to Step 2(c).
   (b) Refine the search:
   Call the AND-OR tree subroutine to check if $\text{value}(\mathcal{T}) < x_j$. If so, let $\delta \leftarrow j$; otherwise, let $\gamma \leftarrow j$.
   (c) Backtrack if out of range:
   Call the AND-OR subroutine to check if $x_\gamma \leq \text{value}(\mathcal{T}) < x_\delta$. If so, push (γ, δ) onto the stack. Otherwise, pop (γ, δ) off the stack. (If the stack is empty, let $\gamma \leftarrow \perp$ and $\delta \leftarrow \top$.)
3. Return γ as an index corresponding to the value of the MIN-MAX tree.

[1] The expected number of calls to the binary subtrees is 3/2, essentially yielding a recurrence of the form $C(N) = (3/2)C(N/2) + N^{\frac{1}{2}}$ for the cost.

Clearly, the algorithm makes $O(W(N)\log(N))$ queries. We claim the following.

Theorem 1. *The algorithm returns the value of the* Min-Max *tree with probability at least* $\frac{2}{3}$.

To prove Theorem 1, we must consider the progress made by the random choices of pivots as well as the error probabilities of the subroutines for And-Or and the searches (each errs with constant probability).

To begin with, assume that the subroutines for And-Or and search never err (thus, $x_\gamma \leq \text{value}(\mathcal{T}) < x_\delta$ at all times). Under this assumption, the progress of the algorithm is determined by how quickly the subinterval converges. Once no value in Step 2(a) is found the algorithm has *converged* (with $x_\gamma = \text{value}(\mathcal{T})$) and can go to Step 3 and terminate (however it is harmless to perform more iterations before doing this).

Let $C(m)$ denote the expected number of iterations of the algorithm until it converges, assuming that m of its inputs are within its current range.

Then, for $m > 1$, $C(m)$ satisfies the recurrence

$$C(m) \leq \frac{2}{m}\left(\sum_{k=\lfloor m/2 \rfloor}^{m-1} C(k)\right) + 1. \tag{1}$$

This can be seen by assuming that the pivot is uniformly selected among all m possible positions within the subinterval and that value$(\mathcal{T})$ always lies in the larger side of the pivot. It is straightforward to verify that the recurrence implies $C(m) \in O(\log(m))$. Therefore, the expected number of iterations of Step 2 made by the algorithm before $x_\gamma = \text{value}(\mathcal{T})$, under the assumption that the subroutines never err, is $O(\log(N))$. By the Markov bound, $O(\log(N))$ iterations suffice to obtain error probability less than any particular constant.

We now consider the fact that the subroutines for And-Or and searching can fail. First, note that, by incurring a multiplicative factor of only $O(\log\log(N))$, each call to the And-Or and search algorithm can be amplified so that its error probability is $O(1/\log(N))$. This results in an $O(W(N)\log(N)\log\log(N))$ algorithm for Min-Max.

These amplification costs are not necessary in our algorithm, since it can cope with a constant fraction of errors in subroutine calls. To see why this is so, let ε be the probability that one or more subroutines err during one iteration of Step 2 of the algorithm. The algorithm begins some $O(\log(N))$ steps away from reaching a *good* state — of the form (α, δ) such that $x_\alpha = \text{value}(\mathcal{T})$. Before reaching a good state, an "incorrect" step for the algorithm places value$(\mathcal{T})$ outside the search interval, and a "correct" step either narrows the search interval or backtracks from a previous error. After reaching a good state, a "correct" step pushes a pair of the form (α, δ) onto the stack and an "incorrect" step pops it off. In each iteration, the algorithm takes a correct step with probability at least $1-\varepsilon$ and an incorrect step with probability at most ε. Therefore, with all but exponentially small probability, the number of correct steps minus the number of incorrect

ones after $c\log(N)$ iterations is at least $\frac{c}{2}\log(N)$. For suitably large c this means that, with constant probability, when the algorithm terminates, $\gamma = \alpha$ (typically with many copies of pairs of the form (α, δ) on the top of its stack).

Finally, we note that, in game-playing contexts, it is useful to determine optimal moves. This corresponds to finding the subtree of a MIN-MAX tree that attains its value. If the leaf values $x_1, \ldots, x_N$ are distinct, this is easily deduced from value($\mathcal{T}$). Otherwise, we can combine the MIN-MAX tree evaluation algorithm with the quantum minimum (maximum) finding algorithm of of Dürr and Høyer [8] to obtain the correct minimax decision. Suppose that we have a balanced MIN-MAX tree and that its root is a MAX gate with c children (each of which is a MIN-MAX tree with a MIN node at its root and N/c leaves). Then the Dürr-Høyer algorithm will make $O(\sqrt{c})$ evaluations of the subtrees, and each subtree will cost $O((N/c)^{\frac{1}{2}+\varepsilon})$ to evaluate. The case of unbalanced trees is more elaborate, but can be solved with the same asymptotic cost by using a generalization of Grover's algorithm that accommodates variable query times [1]. Thus, the minimax decision can be obtained in the same asymptotic cost that it takes to evaluate the tree.

Acknowledgements

We would like to thank Peter van Beek, Peter Høyer and Pascal Poupart for helpful discussions, and the anonymous reviewers for their comments. This research was supported in part by Canada's NSERC, CIAR, MITACS, QuantumWorks, and the U.S. ARO/DTO.

References

1. Ambainis, A.: Quantum search with variable times (arXiv:quant-ph/0609168)
2. Ambainis, A.: A nearly optimal discrete query quantum algorithm for evaluating NAND formulas (arXiv:quant-ph/0704.3628)
3. Barnum, H., Saks, M.: A lower bound on the query complexity of read-once functions. Journal of Computer and System Science 69(2), 244–258 (2004)
4. Bennett, C.H., Bernstein, E., Brassard, G., Vazirani, U.: Strengths and weaknesses of quantum computing. SIAM Journal on Computing 26(5), 1510–1523 (1997)
5. Boyer, M., Brassard, G., Høyer, P., Tapp, A.: Tight bounds on quantum searching. Fortschritte Der Physik 46(4–5), 493–505 (1998)
6. Childs, A.M., Cleve, R., Jordan, S.P., Yeung, D.L.: Discrete-query quantum algorithm for NAND trees (arXiv:quant-ph/070 (2160))
7. Childs, A.M., Reichardt, B.W., Špalek, R., Zhang, S.: Every NAND formula on N variables can be evaluated in time $O(N^{\frac{1}{2}+\varepsilon})$ (arXiv:quant-ph/0703015)
8. Dürr, C., Høyer, P.: A quantum algorithm for finding the minimum (arXiv:quant-ph/9607014)
9. Farhi, E., Goldstone, J., Gutmann, S.: A Quantum Algorithm for the Hamiltonian NAND Tree (arXiv:quant-ph/0702144)

10. Grover, L.K.: A fast quantum mechanical algorithm for database search. In: Proceedings of the 28th Annual ACM Symposium on Theory of Computing (STOC 1996), pp. 212–219 (1996)
11. Saks, M., Wigderson, A.: Probabilistic Boolean Decision Trees and the Complexity of Evaluating Game Trees. In: Proceedings of the 27th Annual IEEE Symposium on Foundations of Computer Science (FOCS 1986), pp. 29–38 (1986)

Irreversibility of Entanglement Loss

Francesco Buscemi

ERATO-SORST Quantum Computation and Information Project,
Japan Science and Technology Agency
buscemi@qci.jst.go.jp

Abstract. The action of a channel on a quantum system, when non trivial, always causes deterioration of initial quantum resources, understood as the entanglement initially shared by the input system with some reference purifying it. One effective way to measure such a deterioration is by measuring the loss of coherent information, namely the difference between the initial coherent information and the final one: such a difference is "small", if and only if the action of the channel can be "almost perfectly" corrected with probability one.

In this work, we generalise this result to different entanglement loss functions, notably including the entanglement of formation loss, and prove that many inequivalent entanglement measures lead to equivalent conditions for approximate quantum error correction. In doing this, we show how different measures of bipartite entanglement give rise to corresponding distance-like functions between quantum channels, and we investigate how these induced distances are related to the cb-norm.

1 Introduction

What is irreversibility of a process? This question, in this form, does not make much sense. We first have to specify "irreversibility with respect to what". It means we first need to decide a set of rules—i. e. a set of allowed transformations together with some free resource—to which one has to conform when trying to revert the process. We can then say that irreversibility basically measures the deterioration of some resource that does not come for free, within the rules we specified. When studying quantum error correction, one usually considers an extremely strict scenario, where legitimate corrections only amount to a fixed quantum channel applied after the action of the noise[1]. This scenario corresponds to the task of trying to restore the entanglement initially shared by the input system (undergoing the noise) with an inaccessible reference, only by using local

[1] This is different, for example, from the correction of quantum measurements [1]: in such a case, we can access classical information produced by the measurement apparatus. Therefore, in general, it is easier (in the sense that the set of allowed transformations is larger) to correct quantum measurements than quantum channels. Another case is that of environment assisted quantum error correction, where we are allowed not only to access classical information from the environment, but we can also choose the measurement to perform onto it [2].

Y. Kawano and M. Mosca (Eds.): TQC 2008, LNCS 5106, pp. 16–28, 2008.

actions on the output system, being any kind of communication between the two systems impossible.

Being quantum error correction a basic task in quantum information theory, the literature on the subject grew rapidly in the last 15 years [3]. It is however possible to devise two main sectors of research: the first one is devoted to the design of good quantum error correcting codes, and directly stems from an algebraic approach to *perfect* quantum error correction; the second one tries to understand conditions under which *approximate* quantum error correction is possible. Usually, while the former is more practically oriented, the latter is able to give information theoretical bounds on the performance of the optimum correction strategy, even when perfect correction is not possible, while leaving unspecified the optimum correction scheme itself.

Our contribution follows the second approach: we will derive some bounds relating the loss of entanglement due to the local action of a noisy channel on a bipartite state with the possibility of undoing such a noise. The original point in our analysis is that we will consider many inequivalent ways to measure entanglement in bipartite mixed states, hence obtaining many inequivalent measures of irreversibility. After reviewing the main results of Ref. [4], we will show how we can relate such entropic quantities with different norm-induced measures of irreversibility, like those exploiting the cb-norm distance [5] or the channel fidelity [6], therefore providing measures of the overall—i. e. state independent—irreversibility of a quantum channel.

2 Evaluating the Coherence of an Evolution

In the following, quantum systems will be often identified with the (finite dimensional) Hilbert spaces supporting them, that is, the roman letter A [resp. B], rigorously denoting the system only, will also serve as a shorthand notation instead of the more explicit $\mathscr{H}^A$ [resp. $\mathscr{H}^B$]. The (complex) dimension of $\mathscr{H}^A$ [resp. $\mathscr{H}^B$] will be denoted as d_A [resp. d_B]. The set of possible states of the system A [resp. B], that is, the set of positive semi-definite operators with unit trace acting on $\mathscr{H}^A$ [resp. $\mathscr{H}^B$], will be equivalently denoted with $\mathcal{S}(\mathscr{H}^A)$ [resp. $\mathcal{S}(\mathscr{H}^B)$] or $\mathcal{S}(A)$ [resp. $\mathcal{S}(B)$].

A general quantum noise $\mathcal{N} : \mathcal{S}(A) \to \mathcal{S}(B)$ is described as a completely positive trace-preserving map—i. e. a *channel*. If the input system A is initially described by the state ρ^A, we will write σ^B to denote $\mathcal{N}(\rho^A)$. The aim of this section is to understand how one can measure the coherence of the evolution

$$\rho^A \mapsto \sigma^B := \mathcal{N}(\rho^A) \tag{1}$$

induced by $\mathcal{N}$ on ρ^A. (We will see in the following how to get rid of the explicit dependence on the input state and obtain a quantity measuring the overall invertibility of a given channel, as a function the channel only.)

Before continuing the discussion, we should clarify what we mean with the term "coherence". Imagine that the input system A is actually the subsystem of

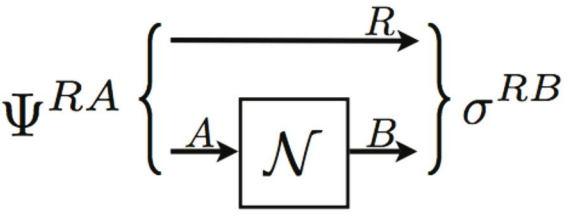


Fig. 1. The input state ρ^A is purified with respect to a reference system R into the state $|\Psi^{RA}\rangle$. The noise $\mathcal{N} : A \to B$ acts on the system A only, in such a way that $|\Psi^{RA}\rangle$ is mapped into $\sigma^{RB} := (\mathrm{id}^R \otimes \mathcal{N}^A)(\Psi^{RA})$.

a larger bipartite system RA, where the letter R stands for *reference*, initially described by a pure state $|\Psi^{RA}\rangle$, such that

$$\mathrm{Tr}_R[\Psi^{RA}] = \rho^A. \tag{2}$$

The situation is depicted in Fig. 1. Notice that the input state ρ^A is mixed if and only if the pure state $|\Psi^{RA}\rangle$ is entangled. Then, the coherence of the evolution (1) can be understood as the amount of residual entanglement survived in the bipartite output (generally mixed) state $\sigma^{RB} := (\mathrm{id}^R \otimes \mathcal{N}^A)(\Psi^{RA})$ after the noise locally acted on A only. However, any naive attempt to formalise such an intuitive idea is soon frustrated by the fact that there exist many different and generally inequivalent ways to measure the entanglement of a mixed bipartite system [7,8,9]. This well-known phenomenon turns out in the existence of many different and generally inequivalent, but all in principle valid, ways to measure the coherence of an evolution.

One possibility to overcome such a problem was considered already in Ref. [10]. There, Schumacher introduced the quantity called *entanglement fidelity* of a channel $\mathcal{N} : A \to A$ with respect to an input state ρ^A, defined as

$$F_e(\rho^A, \mathcal{N}) := \langle \Psi^{RA} | (\mathrm{id}^R \otimes \mathcal{N}^A)(\Psi^{RA}) | \Psi^{RA} \rangle. \tag{3}$$

Such a quantity (which does not depend on the particular purification $|\Psi^{RA}\rangle$ considered) accurately describes how close the channel $\mathcal{N}$ is to the noiseless channel id on the support of ρ^A [10]. However, it was noticed that, as defined in Eq. (3), $F_e(\rho^A, \mathcal{N})$ is *not* related to the coherence of the evolution $\rho^A \mapsto \mathcal{N}(\rho^A)$, in that it is easy to see that a unitary channel—i. e. completely coherent—can result in a null entanglement fidelity. We then have to consider a more general situation, like the one depicted in Fig. 2. After the local noise produced the bipartite state σ^{RB}, we apply a local restoring channel $\mathcal{R} : B \to A$ to obtain

$$\tilde{\rho}^{RA} := (\mathrm{id}^R \otimes \mathcal{R}^B \circ \mathcal{N}^A)(\Psi^{RA}). \tag{4}$$

Notice that in general the restoring channel can explicitly depend on the input state ρ^A and on the noise $\mathcal{N}$. However, for sake of clarity of notation, we will leave

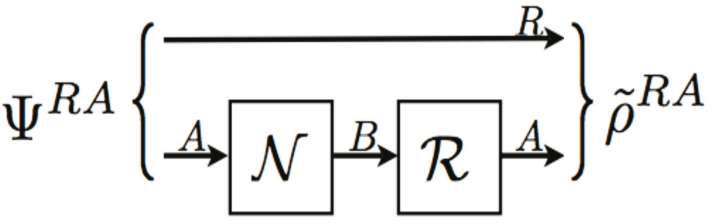


Fig. 2. With respect to Fig. 1, here, after the noise $\mathcal{N}$, we apply a subsequent correction via a local restoring channel $\mathcal{R} : B \to A$. The corrected bipartite output state $(\mathrm{id}^R \otimes \mathcal{R}^B \circ \mathcal{N}^A)(\Psi^{RA})$ is denoted by $\tilde{\rho}^{RA}$.

such dependence understood, and make it explicit again, by writing $\mathcal{R}^B_{\rho,\mathcal{N}}$, only when needed. We now compute the *corrected* entanglement fidelity $F_e(\rho^A, \mathcal{R}\circ\mathcal{N})$ and take the supremum over all possible corrections

$$\overline{F}_e(\rho^A, \mathcal{N}) := \sup_{\mathcal{R}_{\rho,\mathcal{N}}} F_e(\rho^A, \mathcal{R}_{\rho,\mathcal{N}} \circ \mathcal{N}). \tag{5}$$

This is now a good measure of the coherence of the noisy evolution $\rho^A \mapsto \mathcal{N}(\rho^A)$: by construction it is directly related to the degree of invertibility of the noise $\mathcal{N}$ on the support of ρ^A.

3 Coherent Information Loss

The maximisation over all possible correcting channels in Eq. (5) can be extremely hard to compute. Moreover, we are still interested in understanding how the coherence of a transformation is related to the theory of bipartite entanglement. The idea is that of finding some quantity (typically an entropic-like function) which is able to capture at one time both the amount of coherence preserved by the channel as well as the invertibility of the channel itself, possibly bypassing the explicit evaluation of $\overline{F}_e(\rho^A, \mathcal{N})$, for which accurate upper and lower bounds would suffice.

A key-concept in the theory of approximate quantum error correction is that of *coherent information* [10,11], which, for a bipartite state τ^{AB}, is defined as

$$I_c^{A\to B} := S(\tau^B) - S(\tau^{AB}), \tag{6}$$

where $S(\tau) := -\operatorname{Tr}[\tau \log_2 \tau]$ is the von Neumann entropy of the state τ. Notice that, in the definition of coherent information, system A and system B play apparently different roles: such asymmetry acknowledges that the flow of quantum information is considered as being from A to B. Accordingly, *channel coherent information* is defined as

$$I_c(\rho^A, \mathcal{N}) := I_c^{R\to B}(\sigma^{RB}) = S(\sigma^B) - S(\sigma^{RB}), \tag{7}$$

where R, A, B stand for reference, input, and output system, respectively. In our picture, the input state $|\Psi^{RA}\rangle$ is pure, so that $I_c^{R\to A}(\Psi^{RA}) = S(\rho^A) = S(\rho^R)$.

We then compute the coherent information loss due to the action of the noise $\mathcal{N}$ on subsystem A as

$$\begin{aligned}\delta_c(\rho^A,\mathcal{N}) &:= I_c^{R\rightarrow A}(\Psi^{RA}) - I_c^{R\rightarrow B}(\sigma^{RB}) \\ &= S(\rho^A) - I_c(\rho^A,\mathcal{N}) \\ &\geq 0,\end{aligned} \tag{8}$$

where the non-negativity follows from the data-processing inequality [12].

The following theorem (whose first part is in Ref. [13] and second part in Ref. [14]) is exactly what we were searching for

Theorem 1. *Let ρ^A be the input state for a channel $\mathcal{N}: A \rightarrow B$. Let $\delta_c(\rho^A,\mathcal{N})$ be the corresponding loss of coherent information. Then, there exists a recovering channel $\mathcal{R}_{\rho,\mathcal{N}}: B \rightarrow A$ such that*

$$F_e(\rho^A, \mathcal{R}_{\rho,\mathcal{N}} \circ \mathcal{N}) \geq 1 - \sqrt{2\delta_c(\rho^A,\mathcal{N})}. \tag{9}$$

Conversely, for every channel $\mathcal{R}: B \rightarrow A$, it holds

$$\delta_c(\rho^A,\mathcal{N}) \leq \mathrm{g}(1 - F_e(\rho^A, \mathcal{R} \circ \mathcal{N})), \tag{10}$$

where $\mathrm{g}(x)$ is an appropriate positive, continuous, concave, monotonically increasing function such that $\lim_{x\rightarrow 0}\mathrm{g}(x) = 0$. In particular, for $x \leq 1/2$, we can take $\mathrm{g}(x) = 4x\log_2(d_A/x)$. ■

Notice that, in particular, we have

$$\overline{F}_e(\rho^A,\mathcal{N}) \geq 1 - \sqrt{2\delta_c(\rho^A,\mathcal{N})}, \tag{11}$$

and

$$\delta_c(\rho^A,\mathcal{N}) \leq \mathrm{g}(1 - \overline{F}_e(\rho^A,\mathcal{N})), \tag{12}$$

where $\overline{F}_e(\rho^A,\mathcal{N})$ was given in Eq. (5).

The above theorem can be summarised by stating that the loss of coherent information of an input pure state $|\Psi^{RA}\rangle$ due to a channel $\mathcal{N}$ acting on A is small (that means $\delta_c(\rho^A,\mathcal{N})$ close to zero) if and only if the channel $\mathcal{N}$ can be approximately corrected on the support of ρ^A (that means $\overline{F}_e(\rho^A,\mathcal{N})$ close to one). This has been a very important generalisation of the previous theorem appeared Ref. [12] concerning *exact* channel correction, namely $\mathcal{R}\circ\mathcal{N} = \mathrm{id}$ on the support of the input state ρ^A, which turns out to be possible, as a corollary, if and only if $\delta_c(\rho^A,\mathcal{N}) = 0$. Such a generalisation lies at the core of some recent coding theorems for quantum channel capacity—see for example Ref. [15].

Coherent information loss is now an extremely handy quantity to deal with, easy to compute and providing sufficiently tight bounds on the invertibility of the noise. However, coherent information still lacks of some requirements we asked for in our original program. Indeed, we would like to relate the degree of invertibility of a general quantum noise to some function quantifying the loss of entanglement. In fact, it is known that coherent information is not a satisfactory measure of entanglement, and it is not straightforward to generalise Theorem 1 to other entanglement measures loss. To find a relation between noise invertibility and various entanglement measures will be the aim of the next section.

4 Entanglement Loss(es)

In the following we will focus on a widely studied family of possible entanglement measures[2], namely those which stem from von Neumann entropy and analogous quantities. Among these measures, that often gain an operational interpretation as the optimum asymptotic rate at which a particular entanglement transformation can be done, we find, e. g. the *distillable entanglement* E_d, the *distillable key* K_d, the *squashed entanglement* E_{sq}, the *relative entropy of entanglement* E_r, the *entanglement cost* E_c, and the *entanglement of formation* E_f, just to mention some of them (for an accurate review of definitions and properties of a large class of entanglement measures see Ref. [7] and references therein). In particular, in the following we will explicitly call for the entanglement of formation, which is defined as [16]

$$E_f(\tau^{AB}) := \min \sum_i p_i E(\phi_i^{AB}), \tag{13}$$

where the minimum is taken over all possible ensemble decompositions $\tau^{AB} = \sum_i p_i \phi_i^{AB}$, for pure ϕ_i's, and where $E(\phi^{AB}) := S(\mathrm{Tr}_B[\phi^{AB}])$, is the so-called *entropy of pure-state entanglement*. Here we refrain from provide even a short review of the other entropic-like entanglement measures we mentioned, which would be far beyond the scope of the present contribution. The interested reader is directed to Refs. [7] and [8]. For our purposes, we are content with recalling that, given a bipartite state τ^{AB}, the following inequalities hold

$$\begin{aligned} E_d(\tau^{AB}) &\leq K_d(\tau^{AB}) \leq E_{sq}(\tau^{AB}) \leq I^{A:B}(\tau^{AB})/2, \\ K_d(\tau^{AB}) &\leq E_r(\tau^{AB}) \leq E_f(\tau^{AB}), \\ E_{sq}(\tau^{AB}) &\leq E_c(\tau^{AB}) \leq E_f(\tau^{AB}), \end{aligned} \tag{14}$$

where $I^{A:B}(\tau^{AB}) := S(\tau^A)+S(\tau^B)-S(\tau^{AB})$ is the *quantum mutual information*. Moreover

$$\begin{aligned} \max\{I_c^{A\to B}(\tau^{AB}), 0\} &\leq E_d(\tau^{AB}), \\ E_f(\tau^{AB}) &\leq \min\{S(\tau^A), S(\tau^B)\}. \end{aligned} \tag{15}$$

Notice that it is commonly found that

$$E_d(\tau^{AB}) \ll E_{sq}(\tau^{AB}) \ll E_f(\tau^{AB}), \tag{16}$$

and, as dimensions of subsystems A and B increase, a mixed state picked up at random in the convex set of mixed bipartite states almost certainly (that is, with probability approaching one exponentially fast in the dimension) displays an even more dramatic separation [9]

$$E_d(\tau^{AB}) \approx 0, \qquad E_f(\tau^{AB}) \approx \min\{S(\tau^A), S(\tau^B)\}. \tag{17}$$

[2] This is the reason for the plural in the title.

Our motivation is to work out a result analogous to Theorem 1, where, instead of the coherent information loss $\delta_c(\rho^A,\mathcal{N})$ introduced in Eq. (8), we would like to use some other entanglement measure loss

$$\delta_x(\rho^A,\mathcal{N}) := S(\rho^A) - E_x(\sigma^{RB}), \tag{18}$$

where the letter "x" could stand, for example, for "sq" (squashed entanglement loss) or "f" (entanglement of formation loss).

Already at a first glance, we can already say that, thanks to Eqs. (14-15), the second part of Theorem 1 can be extended to other entanglement loss measures, that is

$$\delta_x(\rho^A,\mathcal{N}) \leq \delta_c(\rho^A,\mathcal{N}) \leq \mathrm{g}(1 - F_e(\rho^A,\mathcal{R}\circ\mathcal{N})), \tag{19}$$

for every channel $\mathcal{R} : B \to A$. Instead, the generalisation of the first part of Theorem 1 is not straightforward: because of the typical entanglement behaviour summarised in Eq. (17), we could easily have, for example, a channel causing a *vanishingly small* entanglement of formation loss with, at the same time, a relatively *severe* coherent information loss.

Still, the following argument suggests that *there must be* an analogous of Eq. (11) for alternative entanglement losses: In fact, when evaluated on pure states, all mentioned entanglement measures coincide with the entropy of pure-state entanglement. Moreover, many of these entanglement measures are known to be continuous in the neighbourhood of pure states. This is equivalent to the fact that, in the neighbourhood of pure states, they have to be reciprocally boundable. Therefore, if the action of the noise $\mathcal{N}$ is "sufficiently gentle" and the output state σ^{RB} exhibits an entanglement structure which is "sufficiently close" to pure-state entanglement[3], then it should be possible to write the analogous of Eq. (11) in terms of $\delta_{sq}(\rho^A,\mathcal{N})$ or $\delta_f(\rho^A,\mathcal{N})$, for example, as well. The problem is to explicitly write down such analogous formula.

In Ref. [4], the interested reader can find the proof of the following theorem.

Theorem 2. *Let ρ^A be the input state for a channel $\mathcal{N} : A \to B$. Let $\delta_{sq}(\rho^A,\mathcal{N})$ and $\delta_f(\rho^A,\mathcal{N})$ be the corresponding losses of squashed entanglement and entanglement of formation, respectively. Then*

$$\overline{F}_e(\rho^A,\mathcal{N}) \geq 1 - 2\sqrt{\delta_{sq}(\rho^A,\mathcal{N})}, \tag{20}$$

and

$$\overline{F}_e(\rho^A,\mathcal{N}) \geq 1 - \sqrt{2(2d_Ad_B - 1)^2\delta_f(\rho^A,\mathcal{N})}. \tag{21}$$

■

Notice the large numerical factor, depending on the dimensions of the underlying subsystems, in front of the entanglement of formation loss: this feature

[3] Notice that this is not equivalent to the state ρ^{RB} itself being pure. A trivial example of a mixed state with pure-state entanglement structure is given by $\rho^{RB} = \Psi^{RB_1} \otimes \rho^{B_2}$, where B_1 and B_2 are two subsystems of B.

is reminiscent of the previously mentioned irreversibility gap between distillable entanglement and entanglement of formation, and makes it possible the situation where the noise causes a vanishingly (in the dimensions) small entanglement of formation loss, even though its action is extremely dissipative with respect to the loss of coherent information. On the contrary, the loss of squashed entanglement seems to be an efficient indicator of irreversibility, almost as good as the coherent information loss—in fact, only an extra constant factor of $\sqrt{2}$ appears in Eq. (20) with respect to Eq. (11)—; on the other hand, it is symmetric under the exchange of the input system with the output system, a property that does not hold for the coherent information loss. Summarising this section, the important thing is that there always exist a threshold (which is strictly positive for finite dimensional systems) below which all entanglement losses become equivalent, in the sense that they can be reciprocally bounded (it is noteworthy that, in the case of squashed entanglement loss and coherent information loss, we can have dimension-independent bounds, which is a desirable property when dealing with quantum channels alone, see Section 5 below).

4.1 Distillable Entanglement vs Entanglement of Formation

It is interesting now to forget for a moment about the channel $\mathcal{N}$ itself, and see what Eqs. (11), (20), and (21) mean in terms of a given bipartite mixed state only. First of all, notice that, for every mixed state τ^{AB}, there exist two pure states, $|\phi^{AA'}\rangle$ and $|\psi^{B'B}\rangle$, and two channels, $\mathcal{N} : A' \to B$ and $\mathcal{M} : B' \to A$, such that $(\mathrm{id}^A \otimes \mathcal{N}^{A'})(\phi^{AA'}) = \tau^{AB}$ and $(\mathcal{M}^{B'} \otimes \mathrm{id}^B)(\psi^{B'B}) = \tau^{AB}$.

Now, for a given state τ^{AB}, let us define

$$\delta_c^{A\to B}(\tau^{AB}) := S(\tau^A) - I_c^{A\to B}(\tau^{AB}), \tag{22}$$

and

$$\delta_x^{A\to B}(\tau^{AB}) := S(\tau^A) - E_x(\tau^{AB}), \tag{23}$$

where the letter x is used as before[4]. Then, Theorems 1 and 2 tell us that there exist channels $\mathcal{R} : B \to A'$ and $\mathcal{T} : A \to B'$, and two pure states, $|\tilde{\phi}^{AA'}\rangle$ and $|\tilde{\psi}^{B'B}\rangle$, with $\mathrm{Tr}_{A'}[\tilde{\phi}^{AA'}] = \tau^A$ and $\mathrm{Tr}_{B'}[\tilde{\psi}^{B'B}] = \tau^B$, such that

$$\begin{aligned} \langle\tilde{\phi}^{AA'}|(\mathrm{id}^A \otimes \mathcal{R}^B)(\tau^{AB})|\tilde{\phi}^{AA'}\rangle &\geq 1 - \sqrt{2K_x\delta_x^{A\to B}(\tau^{AB})}, \\ \langle\tilde{\psi}^{B'B}|(\mathcal{T}^A \otimes \mathrm{id}^B)(\tau^{AB})|\tilde{\psi}^{B'B}\rangle &\geq 1 - \sqrt{2K_x\delta_x^{B\to A}(\tau^{AB})}, \end{aligned} \tag{24}$$

where $K_c = 1$, $K_{sq} = 2$, and $K_f = (2d_Ad_B - 1)^2$. In a sense, either $\delta_x^{A\to B}(\tau^{AB})$ or $\delta_x^{B\to A}(\tau^{AB})$ being small[5], it means that the entanglement present in the state τ^{AB} is basically pure-state entanglement, even if τ^{AB} is itself a mixed state. This is the reason for which we can establish a quantitative relation between typically

[4] The analogous quantities $\delta_x^{B\to A}(\tau^{AB})$ are defined in the same way, by simply exchanging subsystems labels, as $\delta_x^{B\to A}(\tau^{AB}) := S(\tau^B) - E_x^{B\to A}(\tau^{AB})$.

[5] That means $\delta_x(\tau^{AB}) \ll (2K_x)^{-1}$.

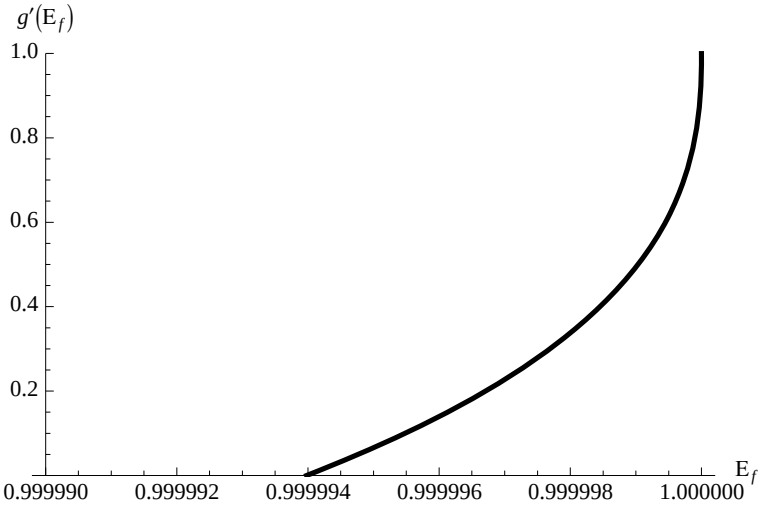


Fig. 3. The plot (axes are normalised so that $\log_2(3) \mapsto 1$) shows the behaviour, for a bipartite system of two qutrits, of the lower bound in Eq. (26) for coherent information as a function of entanglement of formation. Coherent information, and hence distillable entanglement, are bounded from below by the thick curve. Notice that E_f has to be extraordinarily close to its maximum value in order to have a non trivial bound from Eq. (26). This fact suggests that the bound itself could be improved.

inequivalent entanglement measures, as the following corollary of Theorems 1 and 2 clearly states [4].

Corollary 1. *For an arbitrary bipartite mixed state τ^{AB}, with $S(\tau^A) \leq S(\tau^B)$, the following inequality holds*

$$\delta_c^{A\to B}(\tau^{AB}) \leq \mathrm{g}\left(\sqrt{2(2d_A d_B - 1)^2 \delta_f^{A\to B}(\tau^{AB})}\right), \tag{25}$$

where $\mathrm{g}(x)$ is a function as in Eq. (10) in Theorem 1. ■

This corollary is in a sense the quantitative version of the intuitive argument given before Theorem 2, and it represents a first attempt in complementing the findings of Ref. [9], summarised in Eq. (17). It is also possible to invert Eq. (25) and obtain a function $\mathrm{g}'(E_f)$ such that

$$\mathrm{g}'(E_f(\tau^{AB})) \leq I_c^{A\to B}(\tau^{AB}) \leq E_d(\tau^{AB}), \tag{26}$$

for all bipartite state $\tau^{AB} \in \mathcal{S}(A \otimes B)$. The plot of $\mathrm{g}'(E_f)$ is given in Fig. 3 for $d_A = d_B = 3$ (for qubits every entangled state is also distillable), for a state for which $\tau^A = \tau^B = \mathbb{1}/3$. The plotted curve displays the typical behaviour of the bound (26). Notice from Fig. 3 that entanglement of formation has to

be extremely close to its maximum attainable value in order to obtain a non trivial bound from Eq. (26). This is a strong evidence that the bound itself could probably be improved. Nonetheless, we believe that such an improvement, if possible, would only make smaller some (unimportant) constants which are independent of the dimension, while leaving the leading order of dependence on $d = d_A d_B$ in the right hand side of Eq. (25) untouched.

5 Overall Channel Invertibility: Relations between Entanglement Losses and Other Measures of Invertibility

The previous analysis, following Ref. [4], was done in order to quantify the invertibility of a noisy evolution with respect to *a given* input state ρ^A. In this section, we want to derive quantities characterising the "overall" invertibility of a given channel. In other words, we would like to get rid of the explicit dependence on the input state and obtain the analogous of Eqs. (11), (12), (20), and (21) as functions of the channel $\mathcal{N}$ only.

Intuitively, to do this, we should quantify how close the corrected channel $\mathcal{R} \circ \mathcal{N}$ can be to the noiseless channel id, for all possible corrections $\mathcal{R}$. However, in doing this, we have to be very careful about which channel distance function we adopt in order to measure "closeness". A safe choice consists in using the distance induced by the so-called *norm of complete boundedness*, for short *cb-norm*, defined as

$$\|\mathcal{N}\|_{cb} := \sup_n \|\mathrm{id}_n \otimes \mathcal{N}\|_\infty, \tag{27}$$

where id_n is the identity channel on $n \times n$ density matrices, and

$$\|\mathcal{N}\|_\infty := \sup_{\rho \geq 0 : \mathrm{Tr}[\rho] \leq 1} \mathrm{Tr}\left[\, |\mathcal{N}(\rho)| \,\right]. \tag{28}$$

(We put the absolute value inside the trace because in literature one often deals also with non completely positive maps, so that the extension $\mathrm{id}_n \otimes \mathcal{N}$ can be non positive.) Notice, that, in general, $\|\mathcal{N}\|_{cb} \geq \|\mathcal{N}\|_\infty$, and the two norms can be inequivalent [17]. A part of the rather technical definition of cb-norm (the extension in Eq. (27) is necessary, basically for the same reasons for which we usually consider complete positivity instead of the simple positivity), we will be content with knowing that, for channels, $\|\mathcal{N}\|_{cb} = 1$ and $\|\mathcal{N}_1 \otimes \mathcal{N}_2\|_{cb} = \|\mathcal{N}_1\|_{cb} \|\mathcal{N}_2\|_{cb}$, and that the following theorem holds [5].

Theorem 3. *Let $\mathcal{N} : A \to A$ be a channel, with $d_A < \infty$. Then*

$$\begin{aligned} 1 - \inf_{\rho^A} F_e(\rho^A, \mathcal{N}) &\leq 4\sqrt{\|\mathcal{N} - \mathrm{id}\|_{cb}} \\ \|\mathcal{N} - \mathrm{id}\|_{cb} &\leq 4\sqrt{1 - \inf_{\rho^A} F_e(\rho^A, \mathcal{N})}, \end{aligned} \tag{29}$$

where the infimum of the entanglement fidelity is done over all normalised states $\rho^A \in \mathcal{S}(A)$. ■

It is then natural to define a cb-norm–based measure of the overall invertibility of a given channel $\mathcal{N} : A \to B$ as

$$Q_{cb}(\mathcal{N}) := \inf_{\mathcal{R}} \|\mathcal{R} \circ \mathcal{N} - \mathrm{id}\|_{cb}, \tag{30}$$

with the infimum taken over all possible correcting channels $\mathcal{R} : B \to A$.

For a moment, let us now go back to the other functions we introduced before. We will be able to relate them, in some cases with dimension independent bounds, to the cb-norm–based invertibility $Q_{cb}(\mathcal{N})$. Given the loss function $\delta_x(\rho^A, \mathcal{N})$, where $x \in \{c, sq, f\}$ is used to denote the coherent information loss, the squashed entanglement loss, and the entanglement of formation loss, respectively, we define the following quantity

$$\Delta_x(\mathcal{N}) := \sup_{\rho^A} \delta_x(\rho^A, \mathcal{N}), \tag{31}$$

where the supremum is taken over all possible input states ρ^A. Analogously, from Eq. (5), let us define

$$\begin{aligned} \Phi(\mathcal{N}) &:= \inf_{\rho^A} \overline{F}_e(\rho^A, \mathcal{N}) \\ &= \inf_{\rho^A} \sup_{\mathcal{R}_{\rho,\mathcal{N}}} F_e(\rho^A, \mathcal{R}_{\rho,\mathcal{N}} \circ \mathcal{N}). \end{aligned} \tag{32}$$

Such quantities are now functions of the channel only, and we want to understand how well $\Delta_x(\mathcal{N})$ and $\Phi(\mathcal{N})$ capture the "overall" invertibility of a channel.

First of all, let us understand how they are related. Let $\overline{\rho}$ be the state for which $\Delta_x(\mathcal{N})$ is achieved. Then,

$$\begin{aligned} \Delta_x(\mathcal{N}) &= \delta_x(\overline{\rho}, \mathcal{N}) \\ &\leq \mathrm{g}(1 - \overline{F}_e(\overline{\rho}, \mathcal{N})) \\ &\leq \mathrm{g}(1 - \Phi(\mathcal{N})). \end{aligned} \tag{33}$$

On the other hand, let $\Phi(\mathcal{N})$ be achieved with $\underline{\rho}$. Then,

$$\begin{aligned} \Phi(\mathcal{N}) &= \overline{F}_e(\underline{\rho}, \mathcal{N}) \\ &\geq 1 - \sqrt{2K_x \delta_x(\underline{\rho}, \mathcal{N})} \\ &\geq 1 - \sqrt{2K_x \Delta_x(\mathcal{N})}, \end{aligned} \tag{34}$$

where, as usual, $K_c = 1$, $K_{sq} = 2$, and $K_f = (2d_A d_B - 1)^2$.

We are now in position, thanks to Theorem 3, to show how $Q_{cb}(\mathcal{N})$, $\Delta_x(\mathcal{N})$, and $\Phi(\mathcal{N})$ are related to each other. Let the value $\Phi(\mathcal{N})$ be achieved by the couple $(\underline{\rho}, \overline{\mathcal{R}})$. Then,

$$\begin{aligned} Q_{cb}(\mathcal{N}) &\leq \|\overline{\mathcal{R}} \circ \mathcal{N} - \mathrm{id}\|_{cb} \\ &\leq 4\sqrt{1 - \inf_{\rho^A} F_e(\rho^A, \overline{\mathcal{R}} \circ \mathcal{N})} \\ &= 4\sqrt{1 - F_e(\underline{\rho}, \overline{\mathcal{R}} \circ \mathcal{N})} \\ &= 4\sqrt{1 - \Phi(\mathcal{N})} \\ &\leq 4\sqrt[4]{2K_x \Delta_x(\mathcal{N})}, \end{aligned} \tag{35}$$

where in the second line we used Theorem 3, since the channel $\overline{\mathcal{R}} \circ \mathcal{N}$ has equal input and output spaces.

Conversely, let $\Delta_x(\mathcal{N})$ be achieved by $\overline{\rho}$. Then, thanks to Eq. (19)

$$\begin{aligned} \Delta_x(\mathcal{N}) &\leq \mathrm{g}(1 - F_e(\overline{\rho}, \mathcal{R} \circ \mathcal{N})) \\ &\leq \mathrm{g}(1 - \inf_{\rho^A} F_e(\rho^A, \mathcal{R} \circ \mathcal{N})), \end{aligned} \tag{36}$$

for all channels $\mathcal{R} : B \to A$. Let $\underline{\mathcal{R}}$ be the channel achieving the infimum in Eq. (30). Then,

$$\begin{aligned} \Delta_x(\mathcal{N}) &\leq \mathrm{g}\left(1 - \inf_{\rho^A} F_e(\rho^A, \underline{\mathcal{R}} \circ \mathcal{N})\right) \\ &\leq \mathrm{g}\left(4\sqrt{\|\underline{\mathcal{R}} \circ \mathcal{N} - \mathrm{id}\|_{cb}}\right) \\ &= \mathrm{g}\left(4\sqrt{Q_{cb}(\mathcal{N})}\right). \end{aligned} \tag{37}$$

Summarising, we found that

$$\begin{aligned} \Delta_x(\mathcal{N}) &\leq \mathrm{g}\left(4\sqrt{Q_{cb}(\mathcal{N})}\right) \\ Q_{cb}(\mathcal{N}) &\leq 4\sqrt[4]{2K_x \Delta_x(\mathcal{N})}. \end{aligned} \tag{38}$$

In the function $\mathrm{g}(x)$, the dependence on the dimension d is present (see Theorem 1), however only inside a logarithm: this is not bad, in view of coding theorems. The dependence on d can instead be dramatic in K_f; on the contrary, both K_c and K_{sq} are independent on the dimension.

Acknowledgements

This work is funded by Japan Science and Technology Agency, through the ERATO-SORST Project on Quantum Computation and Information. The Author would like to thank in particular M Hayashi, and K Matsumoto for interesting discussions and illuminating suggestions.

References

1. Buscemi, F., Hayashi, M., Horodecki, M.: Phys. Rev. Lett. 100, 210504 (2008)
2. Gregoratti, M., Werner, R.F.: J. Mod. Opt. 50, 915 (2003); Buscemi, F., Chiribella, G., D'Ariano, G.M.: Phys. Rev. Lett. 95, 090501 (2005); Smolin, J.A., Verstraete, F., Winter, A.: Phys. Rev. A 72, 052317 (2005); Buscemi, F.: Phys. Rev. Lett. 99, 180501 (2007)
3. Nielsen, M.A., Chuang, I.L.: Quantum Computation and Quantum Information. Cambridge University Press, Cambridge (2000); Kempe, J.: Quantum Decoherence. In: Poincaré Seminar 2005, Progress in Mathematical Physics Series. Birkhauser Verlag, Berlin (2006)
4. Buscemi, F.: Phys. Rev. A 77, 012309 (2008)
5. Kretschmann, D., Werner, R.F.: N. J. Phys. 6, 26 (2004)
6. Belavkin, V.P., D'Ariano, G.M., Raginsky, M.: J. Math. Phys. 46, 062106 (2005)
7. Christandl, M.: arXiv:quant-ph/0604183v1
8. Hayashi, M.: Quantum Information: an Introduction. Springer, Heidelberg (2006)
9. Hayden, P., Leung, D.W., Winter, A.: Comm. Math. Phys. 265, 95 (2006)
10. Schumacher, B.: Phys. Rev. A 54, 2614 (1996)
11. Lloyd, S.: Phys. Rev. A 55, 1613 (1997)
12. Schumacher, B., Nielsen, M.A.: Phys. Rev. A 54, 2629 (1996)
13. Schumacher, B., Westmoreland, M.D.: Quant. Inf. Processing 1, 5 (2002)
14. Barnum, H., Nielsen, M.A., Schumacher, B.: Phys. Rev. A 57, 4153 (1998)
15. Hayden, P., Horodecki, M., Yard, J., Winter, A.: arXiv:quant-ph/0702005v1
16. Bennett, C.H., Di Vincenzo, D.P., Smolin, J.A., Wootters, W.K.: Phys. Rev. A 54, 3824 (1996)
17. Kretschmann, D., Schlingemann, D., Werner, R.F.: arXiv:quant-ph/0605009v1

Quadratic Form Expansions for Unitaries

Niel de Beaudrap[1], Vincent Danos[2], Elham Kashefi[3], and Martin Roetteler[4]

[1] IQC, University of Waterloo
[2] School of Informatics, University of Edinburgh
[3] Laboratoire d'Informatique de Grenoble
[4] NEC Laboratories America, Inc.

Abstract. We introduce techniques to analyze unitary operations in terms of *quadratic form expansions*, a form similar to a sum over paths in the computational basis where the phase contributed by each path is described by a quadratic form over $\mathbb{R}$. We show how to relate such a form to an entangled resource akin to that of the one-way measurement model of quantum computing. Using this, we describe various conditions under which it is possible to efficiently implement a unitary operation U, either when provided a quadratic form expansion for U as input, or by finding a quadratic form expansion for U from other input data.

1 Introduction

In the one-way measurement model [1,2], quantum states are transformed using single-qubit measurements on an entangled state, which is prepared from an input state by performing controlled-Z operations on pairs of qubits, including the input system and ancillas prepared in the $|+\rangle$ state. This model lends itself to ways of analyzing quantum computation which do not naturally arise in the circuit model, *e.g.* with respect to depth complexity [3] and discrete structures underlying unitary operations [6,8]. In this article, we present another result of this variety, by introducing *quadratic form expansions*.

Definition 1. *Let V be a set of n indices, and $I, O \subseteq V$ be (possibly intersecting) subsets. For a binary string $\mathbf{x} \in \{0,1\}^V$, let $\mathbf{x}_I$ and $\mathbf{x}_O$ be the restriction of $\mathbf{x}$ to those bit-positions indexed by elements of I and O, respectively. Then a* quadratic form expansion *is a matrix-valued expression of the form*

$$U \;=\; \frac{1}{C} \sum_{\mathbf{x} \in \{0,1\}^V} \mathrm{e}^{iQ(\mathbf{x})} \;\; |\mathbf{x}_O\rangle\langle \mathbf{x}_I| \,, \tag{1}$$

$U : \mathcal{H}_2^{\otimes I} \to \mathcal{H}_2^{\otimes O}$, where Q is a real-valued quadratic form on $\mathbf{x}$, and $C \in \mathbb{C}$.

Quadratic form expansions bear a formal similarity to a representation of a propagator of a quantum system in terms of a sum over paths. For a unitary U given as in (1), the outer product $|\mathbf{x}_O\rangle\langle \mathbf{x}_I|$ essentially specifies a particular coefficient, in the row indexed by the substring $\mathbf{x}_I$ and the column indexed by $\mathbf{x}_O$: the amplitude of the transition between these standard basis states is proportional to a sum of complex units specified by $\mathbf{x}_I$, $\mathbf{x}_O$, and the auxiliary variables $v \in V \setminus (I \cup O)$.

Y. Kawano and M. Mosca (Eds.): TQC 2008, LNCS 5106, pp. 29–46, 2008.

Representations of unitary transformations as sums over paths is a well-developed subject in theoretical physics (see e.g. [4,5]); and a representation of unitaries as a sum over paths was used in [9] to provide a simple proof of $\mathsf{BQP} \subseteq \mathsf{PP}$.[1] However, there are also examples of quadratic form expansions which arise without explicitly seeking to represent unitaries in terms of path integrals: the quantum Fourier Transform over $\mathbb{Z}_{2^n}$ can readily be expressed in such a form, and quadratic form expansions for Clifford group operations are implicit in the work of Schlingemann [10] and of Dehaene and de Moor [11].

Given such an expression for a unitary U, we show how to obtain a decomposition of U in terms of operations similar to those used in the one-way measurement model. Using this connection, we demonstrate techniques involving quadratic form expansions to efficiently implement a unitary operator, when the coefficients of the quadratic form satisfies certain constraints related to "generalized flows" (or *gflows*) [8] or Clifford group operations. In particular, we exhibit an $O(n^3/\log n)$ algorithm to obtain a reduced *measurement pattern* (an algorithm in the one-way model) for Clifford group operations from a description of how they transform the Pauli group, based on the results of [11].

2 Connection to the One-Way Model

2.1 Review of the One-Way Model

We can formulate the one-way measurement model as a way of transforming quantum states in the following way. Given a state $|\psi\rangle$ on a set of qubits I (the *input system*), we embed I in a larger system V, where the qubits of $V \setminus I$ are prepared in the $|+\rangle \propto |0\rangle + |1\rangle$ state. We then perform entangling operations on the qubits of V, by performing controlled-Z (denoted $\wedge Z$) operations on some sets of pairs of qubits. (These operations are symmetric and commute with each other, and so we may characterize the entangling stage by a simple graph G whose vertices are the qubits of V: we call this the *entanglement graph* of the procedure.) We then measure each of the qubits of V in some sequence, except for some set of qubits $O \subseteq V$ (the *output subsystem*) which will support a final quantum state. We may represent the measurement result for each qubit v by a bit $\mathsf{s}_v \in \{0, 1\}$ which indexes the orthonormal basis states of the measurement. The measurement basis for each qubit may depend on the results of previous measurements, but without loss of generality may be expressed in terms of a "default" basis which is used when all preceding measurements yield the result 0. Depending on the measurement results, a final Pauli operator may be applied to the qubits in the output subsystem O.[2]

In the original formulation of the one-way measurement model, the measurement bases were described by some axis of the Bloch sphere lying on the XY plane, which

[1] Unitaries were expressed in [9] in terms of paths whose phase contributions are described by cubic polynomials over $\mathbb{Z}_2$; comments made in Section VI of that paper essentially anticipate quadratic form expansions with discretized coefficients. We describe how their techniques provide a means of constructing quadratic form expansions from circuits, in Appendix A.

[2] The reason for using the same variables V, I, and O for these sets of (labels for) qubits as for the sets in Definition 1 will become apparent in the next section.

is sufficient for universal quantum computation. It is also easy to prove that restricting this to states which are an angle $\theta \in \frac{\pi}{4}\mathbb{Z}$ from the X axis is sufficient for approximately universal quantum computation [14]. While it is reasonable to extend beyond this for choices of measurement bases [7], we will only need to consider measurement bases from the XY plane.

2.2 Phase Map Decompositions from Quadratic Form Expansions

Consider a unitary U given by a quadratic form expansion as in (1), where the quadratic form Q is given by

$$Q(\mathbf{x}) \;=\; \sum_{\{u,v\}\subseteq V} \theta_{uv} x_u x_v \;, \tag{2}$$

for some angles $\{\theta_{uv}\}_{u,v\in V}$, and where the sum includes terms for $u = v$. Note that $Q(\mathbf{x})$ can be expressed as an expectation value $\langle \mathbf{x}| \, H \, |\mathbf{x}\rangle$, where H is a 2-local diagonal operator:

$$H \;=\; \sum_{\substack{\{u,v\}\subseteq V \\ u\neq v}} \theta_{uv} \Big[\, |1\rangle\langle 1|_u \otimes |1\rangle\langle 1|_v \, \Big] \;+\; \sum_{v\in V} \theta_{vv} \, |1\rangle\langle 1|_v \;. \tag{3}$$

Then we may decompose U as follows:

$$\begin{aligned} U \;\propto \sum_{\mathbf{x}\in\{0,1\}^V} |\mathbf{x}_O\rangle\langle \mathbf{x}| \, \mathrm{e}^{iH} \, |\mathbf{x}\rangle\langle \mathbf{x}_I| \;&=\; \left[\sum_{\mathbf{y}\in\{0,1\}^V} |\mathbf{y}_O\rangle\langle \mathbf{y}| \right] \mathrm{e}^{iH} \left[\sum_{\mathbf{x}\in\{0,1\}^V} |\mathbf{x}\rangle\langle \mathbf{x}_I| \right] \\ &\propto \;\; R_O \, \mathrm{e}^{iH} \, P_I \;, \end{aligned} \tag{4}$$

where P_I is a unitary embedding which introduces fresh ancillas (indexed by $v \in I^{\mathsf{c}} = V \setminus I$) initialized to the $|+\rangle$ state, and R_O is a map projecting onto the $|+\rangle$ state for all qubits in $O^{\mathsf{c}} = V \setminus O$ (tracing those qubits out afterwards).

Equation (4) is a *phase map decomposition* [12] for U: that is, it expresses U in terms of a process of postselecting observables, as follows. Decompose H into terms H_O, H_1, and H_2, where H_O consists of the 1-local terms on the qubits of O, H_1 consists of the 1-local term on the remaining qubits, and H_2 contains the remaining terms from (3). We then have $U \;\propto\; R_O \, \mathrm{e}^{iH_O} \mathrm{e}^{iH_1} \mathrm{e}^{iH_2} P_I$. Note that e^{iH_O} and e^{iH_1} are simply single-qubit Z rotations applied to the elements of O and O^{c} respectively, where in each case the qubits v in those sets are rotated by an angle θ_{vv}. Then, the composite map $\tilde{R}_O = R_O \mathrm{e}^{iH_1}$ projects each the state of each qubit $v \in O^{\mathsf{c}}$ onto the vector $|0\rangle + \mathrm{e}^{-i\theta_{vv}} |1\rangle$ for each $v \in O^{\mathsf{c}}$. We then have $U = \mathrm{e}^{iH_O} \tilde{R}_O \, \mathrm{e}^{iH_2} P_I$, which is a decomposition of U into the preparation of some number of $|+\rangle$ states, followed by a diagonal unitary operator consisting of two-qubit (fractional) controlled-Z operations, followed by post-selection of states on the Bloch equator for $v \in O^{\mathsf{c}}$, and (unconditionally applied) single-qubit Z rotations on the remaining qubits. If $\theta_{uv} \in \{0, \pi\}$ for all distinct

$u, v \in V$ and for $u = v \in O$, the above describes precisely the action of a measurement-based computation in which the qubits $v \in O^{\mathsf{c}}$ are measured in the eigenbases of observables of the form $M(-\theta_{vv}) = \cos(\theta_{vv})X - \sin(\theta_{vv})Y$, in the special case where all measurements result in the $+1$ eigenstate (which we may label with the bit $\mathsf{s}_v = 0$).

If we are able to extend the above into a complete measurement algorithm, with defined behavior when not all measurements yield a specific outcome, we obtain a measurement-based algorithm for U: we discuss this problem in the next section. Conversely, from every measurement based algorithm, we may obtain a quadratic form expansion:

Theorem 1. *Every unitary operator on n qubits may be expressed by a quadratic form expansion with $|I| = |O| = n$, and where the quadratic form has coefficients $\theta_{uv} \in \{0, \pi\}$ for all cross-terms $x_u x_v$ and $-\pi < \theta_{vv} \leqslant \pi$ for all terms x_v^2. Furthermore, any unitary can be approximated to arbitrary precision by such an expansion where we further require $\theta_{vv} \in \frac{\pi}{4}\mathbb{Z}$.*

Proof. From [13] (and using the notation of that article), the measurement pattern $X_v^{\mathsf{s}_u} M_u^{-\alpha} E_{uv} N_v$ performs the unitary transformation $J(\alpha) = \frac{1}{\sqrt{2}} \left[\begin{smallmatrix} 1 & \mathrm{e}^{i\alpha} \\ 1 & -\mathrm{e}^{i\alpha} \end{smallmatrix}\right]$ for $\alpha \in \mathbb{R}$, from the state space of a qubit u to that of a "fresh" qubit v. These operations generate $\mathsf{SU}(2)$, and generate a group dense in $\mathsf{SU}(2)$ if we restrict to $\alpha \in \frac{\pi}{4}\mathbb{Z}$, by [14].

For any n qubit unitary U, there exists a measurement pattern composed of such patterns together with two-qubit controlled-Z operations (which we denote $\wedge Z$) which implements U. Let G be the entanglement graph of this pattern, and I and O be the qubits defining the input space and output space (respectively) of the measurement pattern. By [6], in this measurement pattern, the probability of every measurement resulting in the $+1$ eigenvalue (i.e. $\mathsf{s}_v = 0$ for all $v \in O^{\mathsf{c}}$) is non-zero. Then, $U \propto R_O \, \mathrm{e}^{iH} P_I$, where

$$H = \sum_{uv \in E(G)} \pi \Big[|1\rangle\langle 1|_u \otimes |1\rangle\langle 1|_v \Big] - \sum_{v \in O^{\mathsf{c}}} \alpha_v \, |1\rangle\langle 1|_v \ . \tag{5}$$

By (4), this yields a quadratic form expansion for U, with

$$Q(\mathbf{x}) = \sum_{uv \in E(G)} \pi x_u x_v - \sum_{v \in O^{\mathsf{c}}} \alpha_v x_v^2 \ . \tag{6}$$

For a quadratic form expansion approximating U, it is sufficient to consider measurement patterns approximating U using angles $\alpha_v \in \frac{\pi}{4}\mathbb{Z}$. □

2.3 Measurement Pattern Interpolation

As we remarked above, the connection from quadratic form expansions to phase map decompositions may allow us to obtain an implementation for U, provided we can determine how to adapt measurements in case the measurements for qubits $v \in O^{\mathsf{c}}$ do not all yield the result $\mathsf{s}_v = 0$.

In a measurement pattern performing N measurements, the computation may follow any of 2^N branches, corresponding to the different combinations of measurement results. Let us call the branch in which every measurement produces the result $\mathsf{s}_v = 0$

the *positive branch* of the measurement pattern.[3] Without loss of generality, we may restrict our attention to patterns where no classical feed-forward is required in the positive branch: then, the positive branch of a measurement pattern is characterized by the *geometry* (G, I, O) of the pattern (where G is the entanglement graph of the measurement algorithm, and $I, O \subseteq V(G)$ are the sets of qubits defining the input/output space of the pattern), and the angles $\mathbf{a} = \{\alpha_v\}_{v \in O^c}$ defining the measurements to be performed.

To extend the description of the positive branch of a measurement algorithm into a *complete* measurement algorithm performing a unitary is the subject of the following problem:

Measurement Pattern Interpolation (MPI). *For input data* $(G, I, O, \mathbf{a})$*, describing a unitary embedding* U *as* the positive branch *of a measurement pattern with geometry* (G, I, O) *and performing measurements* $\mathbf{a}$*, determine if there exists a measurement pattern* $\mathfrak{P}$ *with geometry* (G, I, O) *which performs the transformation* U*.*

This problem is open, and seems to be difficult in general. We may attempt to make the problem easier by considering a more restricted problem:

Generic Measurement Pattern Interpolation (GMPI). *For an input geometry* (G, I, O)*, determine if there exist measurement patterns* $\mathfrak{P}(\mathbf{a})$ *parameterized by a choice* $\mathbf{a}$ *of measurement angles, each with geometry* (G, I, O)*, such that the pattern* $\mathfrak{P}(\mathbf{a})$ *performs a unitary embedding for all* $\mathbf{a}$*.*

GMPI addresses, in an *angle-independent* manner, the subject of the structure of measurement patterns which perform unitary transformations. A special case of the GMPI which has been solved are those geometries (G, I, O) which have a "generalized flow" (or *gflow*), which are the "*yes*" instances of GMPI such that the patterns $\mathfrak{P}(\mathbf{a})$ yield maximally random outcomes on all of their measurements [8]. The following is the definition of gflows in [15], for measurements restricted to the XY plane:[4]

Definition 2. *Given a geometry* (G, I, O) *for a measurement pattern, a* gflow *is a pair* $(g, \preccurlyeq)$*, where* g *is a function from* O^c *to subsets of* I^c *and* $\preccurlyeq$ *is a partial order, such that the following conditions hold for all* u *and* v *in the graph* G*:*

$$v \in g(u) \implies u \prec v\,, \tag{7a}$$

$$v \in \mathrm{odd}(g(u)) \implies u \preccurlyeq v\,, \tag{7b}$$

$$u \in \mathrm{odd}(g(u))\,, \tag{7c}$$

where $\mathrm{odd}(S)$ *is the set of vertices adjacent to an odd number of elements of* S*.*

Here, $u \preccurlyeq v$ essentially represents, for two qubits u and v, that v is measured no earlier than u; a gflow then specifies an ordering in which the qubits are to be measured (with the function g providing a description of how to adapt later measurements). Mhalla and Perdrix [15] present an algorithm which determines if a geometry has a gflow in this

[3] This choice of terminology refers to all measurements yielding the $+1$ eigenvalues of their respective observables $M(-\theta_{vv})$.

[4] The original definition of gflows in [8] also allows for YZ plane and XZ plane measurements, which do not play a role either in our analysis or in [15].

sense in polynomial time, which in turn yields a polynomial time solution to the GMPI for that case. As a result, any instance of the MPI where the geometry (G, I, O) has a gflow can be efficiently solved.

A different special case of the Measurement Pattern Interpolation problem which has been solved is that where the measurement angles are restricted to multiples of $\frac{\pi}{2}$ (or slightly more generally, where the measurement observables are Pauli operations). In this case, as noted in [7], no measurement adaptations are necessary, and the corrections can be determined via the stabilizer formalism [18].

In the following section, we apply these solutions to special cases of the MPI to describe how to synthesize implementations for a unitary U given by a quadratic form expansion.

3 Synthesis Via Measurement Pattern Interpolation

In order to apply the partial solutions to the MPI described above, it will be useful to define the following:

Definition 3. *For a quadratic form expansion*

$$\frac{1}{C} \sum_{\mathbf{x} \in \{0,1\}^V} \mathrm{e}^{iQ(\mathbf{x})} \; |\mathbf{x}_O\rangle\langle \mathbf{x}_I| \qquad \textit{where} \quad Q(\mathbf{x}) \;=\; \sum_{\{u,v\} \subseteq V} \theta_{uv} x_u x_v \;, \tag{8}$$

the geometry induced by the quadratic form *is a triple* (G, I, O)*, where* G *is a* weighted *graph with vertex-set* V*, edge-set* $\{uv \mid u \neq v \text{ and } \theta_{uv} \neq 0\}$*, and edge-weights* $W_G(uv) = \theta_{uv}/\pi$.

Because we can require $-\pi < \theta_{uv} \leqslant \pi$ for all $u, v \in V$, we may without loss of generality restrict G to have edge-weights $-1 < W_G(uv) \leqslant 1$. We will assume that this holds for the remainder of the article, and speak of edges being either of *unit weight* or *fractional weight*.

In this section, we consider the problem of synthesizing an efficient implementation of unitaries U in terms of the geometry induced by a quadratic form expansion for U by reduction to the solved cases of the Measurement Pattern Interpolation problem discussed in the previous section.

3.1 Measurement Pattern Synthesis Via Gflows

Consider a geometry (G, I, O) induced by a quadratic form expansion for a unitary embedding U, where G has only edges of unit weight: then (G, I, O) is also a geometry for a measurement pattern. To obtain a measurement pattern for U, it suffices to find a gflow for (G, I, O): in that case, by Theorem 2 of [8], for any choice of measurement angles $\mathbf{a} = \{\alpha_v\}_{v \in O^{\mathrm{c}}}$, we may consider the pattern

$$\left[\prod_{u \in O^{\mathrm{c}}}^{\succcurlyeq} \left(\bigotimes_{\substack{v \in \mathrm{odd}(g(u)) \\ v \neq u}} Z_v \right) \left(\bigotimes_{v \in g(u)} X_v \right) M_u^{\alpha_u} \right] \left[\prod_{u \sim v} E_{uv} \right] \left[\prod_{u \in I^{\mathrm{c}}} N_u \right] \tag{9}$$

where the left-hand product may be ordered right-to-left in any linear extension of the order $\preccurlyeq$, and $\sim$ denotes the adjacency relation of G. This pattern thus steers the reduced state after every measurement to the state which would occur if the result had been the $+1$ eigenvalue. Every branch of the pattern then performs the same operation as the positive branch, and so the pattern implements a unitary operation U. To obtain a pattern in standard form (with corrections only on output qubits), it is sufficient to propagate the corrections to the left, absorbing them into the measurement bases.

In [15], an $O(n^4)$ algorithm is provided to determine whether or not a geometry (G, I, O) has a gflow where every qubit is to be measured in the XY plane (and obtain one in the case that one exists), where $n = |V(G)|$. The measurement pattern of (9) can be constructed in time $O(n^2)$ by first producing a pattern where corrections undo byproduct operations after each measurement, commuting these corrections to the end, and simplifying; the resulting pattern will have $O(n)$ operations each with complexity $O(n)$. Thus:

Theorem 2. *For a unitary embedding U given as a quadratic form expansion with geometry (G, I, O) with unit edge-weights, there is an $O(n^4)$ algorithm which either determines that (G, I, O) has no gflow, or constructs a measurement pattern consisting of $O(n^2)$ operations[5] implementing U, where $n = |V(G)|$.*

3.2 Circuit Synthesis Via Flows

A geometry (G, I, O) which has fractional edges lies, at first glance, outside of the domain of the Measurement Pattern Interpolation problems described above. However, given a quadratic form expansion with such a geometry, we may still be able to synthesize a circuit for a unitary U represented by that expansion by considering *flows*, which correspond to gflows where the function g maps each vertex $v \in O^{\mathsf{c}}$ to a singleton set: we may say $(f, \preccurlyeq)$ is a flow if and only if $(g_f, \preccurlyeq)$ is a gflow, where $g_f(v) = \{f(v)\}$.

Geometries which have flows are a solvable special case of the GMPI, where the resulting measurement patterns are very "circuit-like". Specifically, the positive branch of a measurement pattern whose geometry has a flow can be represented by a circuit with the following characteristics [6]:

- edges of the form $v\, f(v)$ for $v \in O^{\mathsf{c}}$ correspond to $J(-\alpha_v)$ gates on some wire, separating two wire segments which we label v and $f(v)$;
- edges $uv \in E(G)$ for $u \neq f(v)$ and $v \neq f(u)$ correspond to $\wedge Z$ operations acting on the wire segments labelled by u and v;
- wires whose initial segments are labelled by vertices of I accept arbitrary input states, while those labelled by vertices $I^{\mathsf{c}} \setminus \mathrm{img}(f)$ take input $|+\rangle$.

In the above formulation, the edges of the form $v\, f(v)$ can be interpreted as implementing single-qubit teleportation, in which case a fully entangling unitary is important in order to transfer the information of the "source" qubit to the "target" qubit

[5] These operations may involve measurement angles of arbitrary precision. A corresponding approximate measurement pattern may use $O(n^2 + n\,\mathrm{polylog}(n/\varepsilon))$ operations by the Solovay-Kitaev Theorem [16], where ε is the precision of the coefficients of Q.

upon measurement. However, considering the analysis of [6], it is not important that the edges of the second kind above be fully entangling operations: using such edges to represent fractional powers of $\wedge Z$ will also yield unitary circuits. This motivates the following definition:

Definition 4. *Suppose (G, I, O) is a geometry of a quadratic form expansion for a unitary transformation U. We may say that $(f, \preccurlyeq)$ is a* fractional-edge flow *for (G, I, O) if it is a flow for that geometry, and for all $ab \in E(G)$ with $W_G(ab) < 1$, we have $f(a) \neq b$ and $f(b) \neq a$.*

If (G, I, O) has a fractional-edge flow, we may synthesize a circuit from a quadratic form expansion for U using the description above, where edges ab of fractional weight correspond to $\wedge Z^{W_G(ab)}$ gates on the wire segments labelled by a and b rather than simple $\wedge Z$ gates. We will make use the following easily verified Lemma to consider how to compose/decompose quadratic form expansions:

Lemma 1. *Let U_1, U_2 be matrices given by quadratic form expansions*

$$U_j \;=\; \frac{1}{C_j} \sum_{\mathbf{x} \in \{0,1\}^{V_j}} \mathrm{e}^{iQ_j(\mathbf{x})} \; \big|\mathbf{x}_{O_j}\big\rangle\big\langle\mathbf{x}_{I_j}\big| \;. \tag{10}$$

In the following, $C = C_1 C_2$, and sums are over $\{0,1\}^{V_1 \cup V_2}$.

(i) If $V_1 \cap V_2 = I_2 = O_1$, then $U_2 U_1 \;=\; \frac{1}{C} \sum_{\mathbf{x}} \mathrm{e}^{\,iQ_1(\mathbf{x}) + iQ_2(\mathbf{x})} \; |\mathbf{x}_{O_2}\rangle\langle\mathbf{x}_{I_1}|$.

(ii) If V_1 and V_2 are disjoint, then $U_1 \otimes U_2 \;=\; \frac{1}{C} \sum_{\mathbf{x}} \mathrm{e}^{\,iQ_1(\mathbf{x}) + iQ_2(\mathbf{x})} \; |\mathbf{x}_O\rangle\langle\mathbf{x}_I|$, where $I = I_1 \cup I_2$ and $O = O_1 \cup O_2$.

We prove the circuit construction given by inducting on the number of edges of fractional weight. For the base case, if (G, I, O) has no fractional-weight edges at all, we may synthesize a circuit for U as above, as it corresponds to a normal measurement pattern with a flow, and so falls under the analysis of [6]. We may then induct for geometries with fractional edge-weights if we can show we can decompose the geometry into ones with fewer fractional edge-weights.

For any arbitrary fractional edge $ab \in E(G)$ and each each $z \in O$, we may define $m(ab, z)$ to be the maximal vertex $v \in V(G)$ in the ordering $\preccurlyeq$ subject to z being in the orbit of v under f (that is, $z = f^\ell(v)$ for some $\ell \geqslant 0$), such that at least one of $v \preccurlyeq a$ or $v \preccurlyeq b$ holds. For a set $S \subseteq V(G)$, let $G[S]$ represent the subgraph of G induced by S (i.e. by deleting all vertices in G not in S). Then, define the following subgraphs of G, and corresponding geometries:

- Let V_2 be the set of vertices $m(ab, z)$ for each $z \in O^{\mathsf{c}}$: it is easy to show that $a, b \in V_2$. Let $G_2 = G[V_2]$, and let $\mathcal{G}_2 = (G_2, V_2, V_2)$.
- Let V_1 be the set of vertices $u \in V(G)$ such that $u \preccurlyeq v$ for some $v \in V_2$; let $G_1 = G[V_1] \smallsetminus \big\{uv \,\big|\, u, v \in V_2\big\}$; and let $\mathcal{G}_1 = (G_1, I, V_2)$.
- Let V_3 be the set of vertices $u \in V(G)$ such that $u \succcurlyeq v$ for some $v \in V_2$; let $G_3 = G[V_3] \smallsetminus \big\{uv \,\big|\, u, v \in V_2\big\}$; and let $\mathcal{G}_3 = (G_3, V_2, O)$.

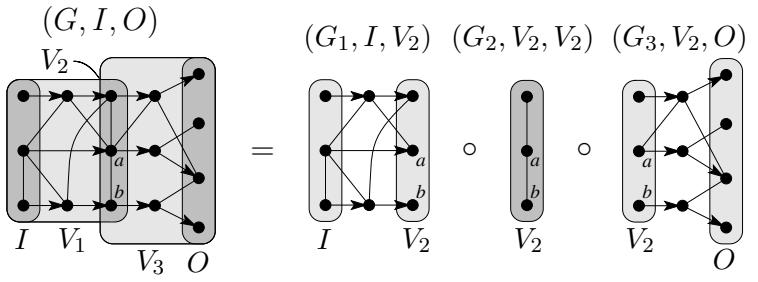


Fig. 1. Illustration of the decomposion of a quadratic form expansion about an edge ab, expressed in terms of geometries. V_2 is a set of maximal vertices under the constraint of being bounded from above, by the vertices a and b, in a partial order $\preccurlyeq$ associated with a fractional-edge flow. Arrows represent the action of the corresponding fractional-edge flow function, f.

This decomposes the geometry (G, I, O) into three geometries with fractional-edge flows, as illustrated in Figure 1.

Let Q_1 be a quadratic form on $\{0,1\}^{V_1}$ consisting of the terms $x_u x_v$ of Q for $u \in V_1$ or $v \in V_1$, but not both; Q_2 be a quadratic form on $\{0,1\}^{V_2}$ consisting of the terms $x_u x_v$ of Q for *distinct* $u, v \in V_2$; and similarly let Q_3 be defined on $\{0,1\}^{V_3}$, and consist of the remaining terms of Q. Then Q_1, Q_2, and Q_3 define quadratic form expansions for some operations U_1, U_2, and U_3 (respectively) with geometries $\mathcal{G}_1$, $\mathcal{G}_2$, and $\mathcal{G}_3$ (respectively).

- U_2 in particular will be a product of operations $\wedge Z^{W_G(uv)}$ for distinct $u, v \in V_2$, as it is a quadratic form expansion whose input and output indices coincide. Then U_2 can be represented as a circuit with a wire for each $u \in V_2$, with fractional controlled-Z gates $\wedge Z^{W_G(uv)}$ for each edge $uv \in E(G)$.
- Both $\mathcal{G}_1$ and $\mathcal{G}_3$ have fractional-edge flows, but fewer fractional edges than (G, I, O). By induction, U_1 and U_3 are also unitary embeddings, and have circuits with wire-segments connected by $J(\theta_v)$ gates (where θ_v are the coefficients of the terms x_v^2 in each quadratic form) and possibly fractional $\wedge Z$ gates (as in the case for U_2).
- In the circuits described above, the terminal wire-segments for U_1 and (a subset of) the initial wire-segments for U_3 have the same labels as the wires for U_2. The composite circuit for $U_3 U_2 U_1$ can then use these labels to arrive at a unified labelling of its' wire-segments.

Because $Q_1(\mathbf{x}_{V_1}) + Q_2(\mathbf{x}_{V_2}) + Q_3(\mathbf{x}_{V_3}) = Q(\mathbf{x})$ for all $\mathbf{x} \in \{0,1\}^V$ by construction, the composite operation $U_3 U_2 U_1$ can differ from U by at most a scalar factor by Lemma 1; so the circuit obtained implements the operation U.

In [15], an $O(kn)$ algorithm is provided to determine whether or not a geometry (G, I, O) has a flow, and obtain one if it exists, where $n = |V(G)|$ and $k = |O|$. For each edge uv, we may check whether one of $W_G(uv) = 1$ or $\big[u \neq f(v)$ and $v \neq f(u)\big]$ holds: if all edges satisfy this constraint, the circuit described above is well-defined. By iterating through the vertices of $V(G)$ in an arbitrary linear extension of $\preccurlyeq$, we may

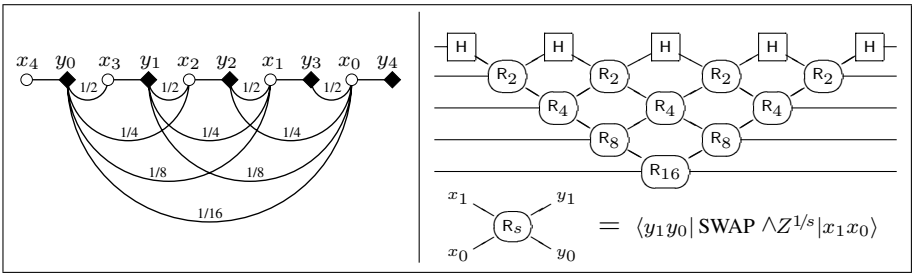


Fig. 2. The geometry for the quadratic form expansion of the QFT for $\mathbb{Z}_{32}$, and the corresponding circuit due to [22]. In the geometry (on the left), input vertices are labelled by circles, output vertices by lozenges, and fractional edges are labelled with their edge-weights.

construct the circuit described above can be constructed in time $O(m)$, and the size of the resulting circuit will also be $O(m)$, where $m = |E(G)|$. By an extremal result [17], any geometry with a flow has $m \leqslant kn$: thus, the total running time of this algorithm is $O(kn)$.

In the case $|I| = |O|$, a flow function f is unique if it exists, by [21]; so in this case, if (G, I, O) has a flow but there is an edge $v\, f(v)$ of fractional weight, there is no fractional-weight flow for (G, I, O). We then have:

Theorem 3. *For a unitary transformation U given as a quadratic form expansion with geometry (G, I, O), there is an $O(kn)$ algorithm which either determines that (G, I, O) has no fractional-edge flow, or constructs a circuit consisting of $O(kn)$ operations*[6] *implementing U, where $n = |V(G)|$ and $k = |O|$.*

Example. The Fourier Transform over $\mathbb{Z}_{2^n}$ is given by the matrix formula

$$\mathcal{F}_n \;=\; \frac{1}{\sqrt{2^n}} \sum_{\mathbf{x},\mathbf{y} \in \{0,1\}^n} \mathrm{e}^{2\pi i \left[\sum_{h=0}^{n-1} 2^h x_h\right] \left[\sum_{j=0}^{n-1} 2^j y_j\right] / 2^n} \, |\mathbf{y}\rangle\langle\mathbf{x}| \,, \tag{11}$$

which is a quadratic form expansion; its quadratic form can be given by

$$Q(\mathbf{x}, \mathbf{y}) \;=\; \sum_{h=0}^{n-1} \sum_{j=0}^{n-1-h} \frac{2^{(h+j)}}{2^{n-1}} \, \pi x_h y_j \;. \tag{12}$$

This has a fractional-edge flow for all n. Figure 2 illustrates this geometry for $n = 5$, and the circuit (due to [22]) which may be synthesized from it.

3.3 Synthesizing Measurement Patterns for the Clifford Group

If a quadratic form expansion has a geometry whose edges all have unit weight, and its' other coefficients are multiples of $\frac{\pi}{2}$, then it corresponds to the positive branch of a

[6] These operations may consist of $J(\alpha)$ gates and fractional $\wedge Z$ gates of arbitrary precision. A corresponding circuit using a finite elementary gate set may be of size $O(kn \,\mathrm{polylog}(kn/\varepsilon))$ by the Solovay-Kitaev Theorem [16], where ε is the precision of the coefficients of Q.

measurement pattern which measures only X or Y observables. A measurement pattern of this sort, if it performs a unitary operation, performs a Clifford group operation in particular.

An algorithm of Aaronson and Gottesman [20] can produce a circuit of size $O(n^2/\log n)$ in classical deterministic time $O(n^3/\log n)$ for a Clifford group operation U acting on n qubits, from a description of how U transforms Pauli operators by conjugation. By converting the circuit into a measurement-based algorithm, and performing the graph transformations of [19] to remove auxiliary qubits, we may obtain a pattern of at most $3n$ qubits[7] in time $O(n^4/\log n)$. Building on the results of [11], we show how to classically compute such a minimal pattern in time $O(n^3/\log n)$ by solving the MPI for a quadratic form expansion for U.

Obtaining a Quadratic Form Expansion. For the sake of completeness, we outline the relevant results of [11]. Define the following notation for bit-flip and phase-flip operators on a qubit t out of a collection $\{1,\ldots,n\}$:

$$P_t \;=\; X_t\,, \qquad\qquad P_{n+t} \;=\; Z_t\,. \tag{13}$$

Let $\mathrm{diag}(M) \in \mathbb{Z}_2^m$ represent the vector of the diagonal elements of any square boolean matrix M; and let $\mathbf{d}(M) = \mathrm{diag}\left(M^\top \left[\begin{smallmatrix} 0 & \mathbb{1}_n \\ 0 & 0 \end{smallmatrix}\right] M\right) \in \mathbb{Z}_2^{2n}$ for a $2n \times 2n$ matrix M over $\mathbb{Z}_2$. Then, we may represent an n qubit Clifford operation U by a $2n \times 2n$ boolean matrix C and a vector $\mathbf{h} \in \{0,1\}^{2n}$, whose coefficients are jointly given by

$$UP_tU^\dagger \;=\; i^{d_t(C)}\big(-1\big)^{h_t} \bigotimes_{j=1}^{n} \left[Z_j^{C_{(n+j)t}} X_j^{C_{jt}}\right] \tag{14}$$

for each $1 \leqslant t \leqslant 2n$. (Note that the factor of $i^{d_t(C)}$ is only necessary to ensure that the image of P_t is Hermitian, and does not serve as a constraint on the value of C as a matrix.) We will call an ordered pair $(C, \mathbf{h})$ a *Leuven tableau* for a Clifford group element U if it satisfies (14).[8]

Provided a Leuven tableau $(C, \mathbf{h})$ for a Clifford group operation U, [11] provides a matrix formula for U which we may obtain for U, as follows. Decompose C as a block matrix $C = \left[\begin{smallmatrix} E & F \\ G & H \end{smallmatrix}\right]$ with $n \times n$ blocks, and then find invertible matrices $\tilde{R}_1, \tilde{R}_2$ over $\mathbb{Z}_2$ such that $\tilde{R}_1^{-1} G \tilde{R}_2 = \left[\begin{smallmatrix} 0 & 0 \\ 0 & \mathbb{1}_r \end{smallmatrix}\right]$ for some $r < n$ (using *e.g.* the decomposition algorithm of [23] to obtain $\tilde{R}_1$ and $\tilde{R}_2$ in terms of elementary row operations). Then, define the matrices

$$\begin{bmatrix} \tilde{E}_{11} & \tilde{E}_{12} \\ \tilde{E}_{21} & \tilde{E}_{22} \end{bmatrix} \;=\; \tilde{R}_1^\top E \tilde{R}_2\,, \qquad R_1 \;=\; \tilde{R}_1\,, \qquad R_2 \;=\; \begin{bmatrix} \tilde{E}_{11}^{-1} & 0 \\ 0 & \mathbb{1}_r \end{bmatrix}^\top \tilde{R}_2^\top\,, \tag{15}$$

[7] In [7], Clifford operations on n qubits are described as having minimal patterns for are described as requiring at most $2n$ qubits; however, this only holds up to local Clifford operations on the output qubits.

[8] Note that the block matrix $\left[C^\top \;\mathbf{h}\right]$ is similar to a *destabilizer tableau* as defined in [20].

where $\tilde{E}_{11}$ is taken to be a block of size $(n-r)\times(n-r)$. We may then obtain the block matrices

$$\begin{bmatrix} \mathbb{1}_{n-r} & E_{12} & F_{11} & F_{12} \\ E_{21} & E_{22} & F_{21} & F_{22} \\ 0 & 0 & H_{11} & H_{12} \\ 0 & \mathbb{1}_r & H_{21} & H_{22} \end{bmatrix} \;=\; \begin{bmatrix} R_1^\top & 0 \\ 0 & R_1^{-1} \end{bmatrix} C \begin{bmatrix} R_2^\top & 0 \\ 0 & R_2^{-1} \end{bmatrix}, \tag{16}$$

and use these to construct the $n\times n$ boolean matrices

$$M_{br} \;=\; \begin{bmatrix} F_{11}+E_{12}H_{21} & E_{12} \\ E_{12}^\top & E_{22} \end{bmatrix}, \qquad M_{bc} \;=\; \begin{bmatrix} 0 & H_{21}^\top \\ H_{21} & H_{22} \end{bmatrix}. \tag{17}$$

Next, define

$$\begin{aligned} \mathbf{d}_{br} &= \operatorname{diag}(M_{br}), & \mathbf{d}_{bc} &= \operatorname{diag}(M_{bc}), \\ L_{br} &= \operatorname{lower}\big(M_{br}+\mathbf{d}_{br}\mathbf{d}_{br}^\top\big), & L_{bc} &= \operatorname{lower}\big(M_{bc}+\mathbf{d}_{bc}\mathbf{d}_{bc}^\top\big), \end{aligned} \tag{18}$$

where $\operatorname{lower}(M)$ is the strictly lower-triangular part of a square matrix M (with all other coefficients set to 0). Finally, define $\Pi_r = \left[\begin{smallmatrix} 0 & 0 \\ 0 & \mathbb{1}_r \end{smallmatrix}\right]$ and $\Pi_r^\perp = \mathbb{1}_n - \Pi_r$ for the sake of brevity, and let[9]

$$\mathbf{t} \;=\; \big[\, \mathbb{1}_n \;\; 0 \,\big]\, \mathbf{h} \;+\; \operatorname{diag}\Big(\big[R_2^{-1}\Pi_r\big] L_{br} \big[R_2^{-1}\Pi_r\big]^\top \Big), \tag{19}$$

$$\begin{aligned} \mathbf{h}_{bc} \;=\; \big[\, 0 \;\; R_2^{-\top} \,\big]\, \mathbf{h} \;+\; R_2^{-\top} \operatorname{diag}\Big(R_2^\top \Big[& L_{bc} \;+\; \Pi_r M_{bc} \\ & +\; \big(\Pi_r^\perp + \Pi_r M_{bc}\big)\, L_{br}\, \big(\Pi_r^\perp + M_{bc}\Pi_r\big) \Big] R_2 \Big). \end{aligned} \tag{20}$$

Then Theorem 6 of [11] states that the unitary operation U for the Clifford operation characterized by $(C,\mathbf{h})$ is given by the matrix formula

$$U \;=\; \frac{1}{\sqrt{2^r}} \sum_{\substack{\mathbf{x}_b\in\{0,1\}^{n-r} \\ \mathbf{x}_c,\mathbf{x}_r\in\{0,1\}^r}} \Big[\, (-1)^{\left(\mathbf{x}_{br}^\top L_{br}\mathbf{x}_{br} + \mathbf{x}_r^\top\mathbf{x}_c + \mathbf{x}_{bc}^\top L_{bc}\mathbf{x}_{bc} + \mathbf{h}_{bc}^\top\mathbf{x}_{bc}\right)} \times (-i)^{\left(\mathbf{d}_{br}^\top\mathbf{x}_{br} + \mathbf{d}_{bc}^\top\mathbf{x}_{bc}\right)} \; \big|R_1\mathbf{x}_{br}\big\rangle\big\langle R_2^{-1}\mathbf{x}_{bc}+\mathbf{t}\big| \, \Big], \tag{21}$$

where $\mathbf{x}_{br} = \left[\begin{smallmatrix} \mathbf{x}_b \\ \mathbf{x}_r \end{smallmatrix}\right]$ and $\mathbf{x}_{bc} = \left[\begin{smallmatrix} \mathbf{x}_b \\ \mathbf{x}_c \end{smallmatrix}\right]$ are n bit boolean vectors.

The formula in (21) shows strong similarities to a quadratic form expansion. In particular, consider disjoint sets of indices V_b, V_r, and V_c, with $|V_b| = n-r$ and $|V_r| = |V_c| = r$. Let $V = V_b \cup V_c \cup V_r$, $I = V_b \cup V_c$, and $O = V_b \cup V_r$, and define the following notation for $\mathbf{x}\in\{0,1\}^V$:

$$\mathbf{x}_I \;=\; \begin{bmatrix} \mathbf{x}_b \\ \mathbf{x}_c \end{bmatrix} \;=\; \begin{bmatrix} \mathbf{x}_{V_b} \\ \mathbf{x}_{V_c} \end{bmatrix} \in \{0,1\}^I, \quad \mathbf{x}_O \;=\; \begin{bmatrix} \mathbf{x}_b \\ \mathbf{x}_r \end{bmatrix} \;=\; \begin{bmatrix} \mathbf{x}_{V_b} \\ \mathbf{x}_{V_r} \end{bmatrix} \in \{0,1\}^O, \tag{22}$$

[9] These formulae for $\mathbf{t}$ and $\mathbf{h}_{bc}$ may be obtained by repeated application of Theorem 2 of [11].

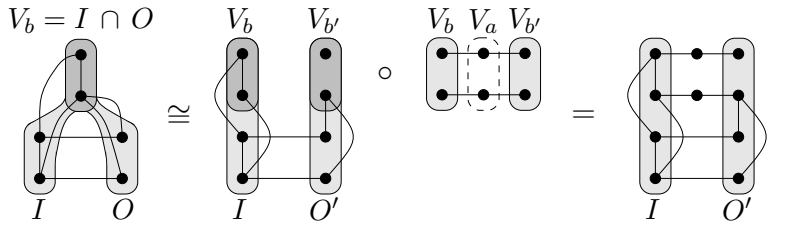


Fig. 3. Illustration of geometries arising from quadratic form expansions yielding the same matrix. On the left is a geometry whose inputs and output intersect; on the right is a geometry from an equivalent quadratic form expansion, constructed so that the input and output indices are disjoint.

$$\begin{aligned} Q(\mathbf{x}) \;=\; \pi\Big(\mathbf{x}_O^\top L_{br}\mathbf{x}_O \;+\; \mathbf{x}_O^\top \Pi_r \mathbf{x}_I \;+\; \mathbf{x}_I^\top L_{bc}\mathbf{x}_I \;+\; \mathbf{x}_I^\top \mathbf{h}_{bc}\mathbf{h}_{bc}^\top \mathbf{x}_I\Big) \\ -\;\frac{\pi}{2}\Big(\mathbf{x}_O^\top \mathbf{d}_{br}\mathbf{d}_{br}^\top \mathbf{x}_O \;+\; \mathbf{x}_I^\top \mathbf{d}_{bc}\mathbf{d}_{bc}^\top \mathbf{x}_I\Big)\;. \end{aligned} \tag{23}$$

Then, (21) is equivalent to

$$U \;=\; \frac{1}{\sqrt{2^r}} \sum_{\mathbf{x}\in\{0,1\}^V} \mathrm{e}^{iQ(\mathbf{x})} \left|R_1\mathbf{x}_O\right\rangle\!\left\langle R_2^{-1}\mathbf{x}_I + \mathbf{t}\right|\;, \tag{24}$$

which is essentially a quadratic form expansion sandwiched between two networks of controlled-not and X gates. To obtain a simple quadratic form expansion, we would like to perform a change of variables on $\mathbf{x}_I$ and $\mathbf{x}_O$; but this cannot be done as I and O intersect at V_b, and the changes of variables do not necessarily respect the partitioning of I and O with respect to this intersection. However, we may add auxiliary variables in order to produce an expansion with disjoint input and output indices. Note that

$$\mathbb{1}_2 \;=\; \sum_{\mathbf{x}\in\{0,1\}^2} \delta_{x_1,x_2} \left|x_2\right\rangle\!\left\langle x_1\right| \;=\; \frac{1}{2}\sum_{\mathbf{x}\in\{0,1\}^3} (-1)^{x_1x_3+x_2x_3} \left|x_2\right\rangle\!\left\langle x_1\right| \tag{25}$$

where $\delta_{x,y}$ is the Kronecker delta. Let V_a and $V_{b'}$ be disjoint copies of V_b, and set $V' = V \cup V_a \cup V_{b'}$ and $O' = V_{b'} \cup V_r$. Writing $\mathbf{x}_a$ and $\mathbf{x}_{b'}$ for the restriction of $\mathbf{x} \in \{0,1\}^{V'}$ to V_a and $V_{b'}$, we then define

$$\mathbf{x}_I \;=\; \begin{bmatrix}\mathbf{x}_b \\ \mathbf{x}_c\end{bmatrix} \in \{0,1\}^I\;, \qquad\qquad \mathbf{x}_{O'} \;=\; \begin{bmatrix}\mathbf{x}_{b'} \\ \mathbf{x}_r\end{bmatrix} \in \{0,1\}^{O'}\;, \tag{26}$$

$$\begin{aligned} Q'(\mathbf{x}_I\,,\,\mathbf{x}_a\,,\,\mathbf{x}_{O'}) \;=\; \pi\Big(\mathbf{x}_{O'}^\top L_{br}\mathbf{x}_{O'} \;+\; \mathbf{x}_{O'}^\top \Pi_r \mathbf{x}_I \;+\; \mathbf{x}_I^\top L_{bc}\mathbf{x}_I \;+\; \mathbf{h}_{bc}^\top \mathbf{x}_I\Big) \\ +\;\pi\mathbf{x}_I^\top \begin{bmatrix}\mathbb{1}_{n-r}\\ 0\end{bmatrix}\mathbf{x}_a \;+\; \pi\mathbf{x}_{O'}^\top \begin{bmatrix}\mathbb{1}_{n-r}\\ 0\end{bmatrix}\mathbf{x}_a \\ -\;\frac{\pi}{2}\Big(\mathbf{d}_{br}^\top \mathbf{x}_{O'} \;+\; \mathbf{d}_{bc}^\top \mathbf{x}_I\Big)\;. \end{aligned} \tag{27}$$

Note that the difference between the expressions for Q' and Q is essentially that all instances of $\mathbf{x}_O$ have been replaced with $\mathbf{x}_{O'}$ (which is independent from $\mathbf{x}_I$), and the

presence of the terms involving $\mathbf{x}_a$. (This manipulation is illustrated in Figure 3 as a transformation of geometries.) We therefore have

$$\begin{aligned}\sum_{\mathbf{x}\in\{0,1\}^V} & \mathrm{e}^{iQ(\mathbf{x})} \left|R_1\mathbf{x}_O\right\rangle\!\left\langle R_2^{-1}\mathbf{x}_I + \mathbf{t}\right| \\ &= \sum_{\mathbf{x}_I,\,\mathbf{x}_{O'}} \delta_{\mathbf{x}_b,\mathbf{x}_{b'}}\, \mathrm{e}^{iQ'(\mathbf{x}_I,\,\mathbf{0},\,\mathbf{x}_{O'})} \left|R_1\mathbf{x}_{O'}\right\rangle\!\left\langle R_2^{-1}\mathbf{x}_I + \mathbf{t}\right| \\ &= \frac{1}{2^{n-r}} \sum_{\mathbf{x}\in\{0,1\}^{V'}} \mathrm{e}^{iQ'(\mathbf{x}_I,\,\mathbf{x}_a,\,\mathbf{x}_{O'})} \left|R_1\mathbf{x}_{O'}\right\rangle\!\left\langle R_2^{-1}\mathbf{x}_I + \mathbf{t}\right| . \qquad (28)\end{aligned}$$

Substituting the final expression of (28) into (24) and performing the appropriate change of variables, we have

$$U \;=\; \frac{\sqrt{2^r}}{2^n} \sum_{\mathbf{x}\in\{0,1\}^{V'}} \mathrm{e}^{iQ'(R_2(\mathbf{x}_I+\mathbf{t}),\,\mathbf{x}_a,\,R_1^{-1}\mathbf{x}_{O'})} \left|\mathbf{x}_{O'}\right\rangle\!\left\langle\mathbf{x}_I\right| . \qquad (29)$$

Note that the quadratic form of the expansion in (29) has only angles θ_{uv} which are multiples of $\frac{\pi}{2}$, with $\theta_{uv} \in \{0, \pi\}$ for $u \neq v$. This then represents the positive branch of a one-way measurement pattern on the geometry (G', I, O') of the quadratic form expansion of 29, using only X or Y basis measurements, and having only $n - r$ auxiliary vertices.

Interpolating the Measurement Pattern. We can augment this to a measurement pattern by applying the techniques of the stabilizer formalism [18] to the stabilizer code generated by the operators $K(v) = X_v \prod_{v\sim w} Z_w$ for $v \in I^{\mathsf{c}}$ (where again $\sim$ is the adjacency relation of G), as follows. To obtain the final correction, we do classical pre-processing simulating the evolution of the *state space* when we perform one measurement at a time. For each measured qubit u, there is an associated correction σ_v which we may perform immediately after the measurement if we obtain the result $\mathsf{s}_u = 1$. We store for each qubit v two boolean formulas β_v and γ_v, representing the X and Z components of the accumulated corrections to be performed on v. When v is measured, the pending X corrections will affect the result of any Y measurement, and the pending Z corrections will affect the result of any X or Y measurement by interchanging the significance of the two measurement outcomes.[10] Just prior to the (simulated) measurement of v, assign $\delta_v = \gamma_v$ if v is to be measured with an X observable, and $\delta_v = \beta_v + \gamma_v$ if v is to be measured with a Y observable. Thus, upon measuring v, the following operations are accumulated into the corrections which must be performed:

- For every qubit w where σ_v acts with an X or Y operation, we must add $\mathsf{s}_v + \delta_v$ to β_w;
- For every qubit w where σ_v acts with a Y or Z operation, we must add $\mathsf{s}_v + \delta_v$ to γ_w.

[10] This corresponds to the process of *signal shifting* as described in [13].

This accounts for the accumulated corrections due to the measurement of v and every preceding measurement which affects it. By simulating measurement for all of the qubits in O^{c} in this way, we obtain boolean formulae for the corrections on O in terms of the results of the measurements: the correction to be performed for some $w \in O$ is $X^{\beta_w} Z^{\gamma_w}$, for β_w and γ_w constructed after all of the (simulated) measurements. To obtain β_w and γ_w for all $w \in O$ in this way takes time $O(n^2)$.

It is easy to show that the resulting measurement pattern cannot be reduced by the techniques of [19], as follows. Let A denote the set of auxiliary vertices corresponding to the bit positions of $\mathbf{x}_a$: note that in the measurement pattern, these are all to be measured with the observable X, and are adjacent only to the input/output vertices (those which index $\mathbf{x}_I$ and $\mathbf{x}_O$).[11] To eliminate a vertex $v \in A$ using the methods of [19] on the geometry induced by the quadratic form expansion, we would have to identify an output variable $b_0 \in O$ adjacent to x, and apply the graph transformation in [19, Proposition 1]. This would result in a geometry where b_0 has the former neighbors of v in G (and in particular is not adjacent to any more removable vertices), and where a local Clifford (which is not a Pauli operator) must be applied to b_0 after the entangling procedure. Because b_0 is not adjacent to any other auxiliary qubit after this transformation, the local Clifford cannot be undone or made into a Pauli operator by e.g. another vertex removal; then, except by extending the computational model to allow for corrections which are local Clifford operations, performing the local Clifford can only be done by introducing another auxiliary qubit (or rather, a new output qubit following b_0, making the latter an auxiliary qubit). Thus:

Theorem 4. *For an n-qubit Clifford group operation U given in the form of a Leuven tableau, there is an $O(n^3/\log n)$ algorithm which produces a minimal one-way measurement pattern for U.*

This represents a more efficient algorithm to find totally reduced Clifford patterns than the existing techniques to obtain one via the circuit model. The quadratic form of (29) can be found from a Leuven tableau $(C, \mathbf{h})$ in time $O(n^3/\log n)$, which is dominated by the time required to compute R_1 and R_2, and by the time required to do the boolean row-reductions implicit in the classical pre-processing via the stabilizer formalism, using the techniques of [23]. To contrast, an approximately optimal quantum circuit for a Clifford group operation (*i.e.* consisting of $O(n^2/\log n)$ gates) can be found from a Leuven tableau in time $O(n^3/\log n)$ by transforming it into a destabilizer tableau, and then applying the algorithm of [20]. To obtain a measurement pattern from such a circuit by composing the patterns for each gate, removing vertices opportunistically (with each removal taking time $O(n^2)$), requires time $O(n^4/\log n)$. Thus, making use of quadratic form expansions provides us with a faster algorithm to obtain reduced measurement patterns for Clifford group operations.

4 Conclusions and Open Problems

We have introduced quadratic form expansions, and developed techniques which suggest that they may be useful for synthesizing efficient implementations for unitary

[11] There are no square terms x_v^2 for $v \in A$ or cross-term $x_u x_v$ for $u, v \in A$ before the change of variables in (28), and the change of variables itself does not introduce any.

operations. We described conditions under which implementations may be efficiently found for unitaries specified by quadratic form expansions; and we showed how quadratic form expansions leads to more efficient algorithms for obtaining reduced patterns for Clifford operations in the one way measurement model.

In the introduction, we mentioned that quadratic form expansions are similar in form to a sum-over-paths representation of unitary operations, which is a well-developed subject in theoretical physics. This raises the question of whether the techniques developed here are useful e.g. for developing algorithms to simulate physical systems. It is not known whether the solved cases of the Measurement Pattern Interpolation problem correspond to *natural* (in the more literal sense) unitaries expressed as sums over paths: this question, and how to extend the solved cases of the MPI to include propagators for interesting physical systems, remain open.

References

1. Raussendorf, R., Briegel, H.: A one-way quantum computer. Physical Review Letters 86, 5188–5191 (2001)
2. Raussendorf, R., Briegel, H.: Computational model underlying the one-way quantum computer. Quantum Information & Computation 2(6), 443–486 (2002)
3. Broadbent, A., Kashefi, E.: Parallelizing quantum circuits. arXiv:0704.1736 (2007)
4. Feynmann, R.P., Hibbs, A.R.: Quantum Mechanics and Path Integrals. McGraw-Hill, New York (1965)
5. Schulman, L.S.: Techniques and Application of Path Integration. Wiley-Interscience, New York (1981)
6. Danos, V., Kashefi, E.: Determinism in the one-way model. Physical Review A 74(052310) (2006) arXiv:quant-ph/0506062
7. Browne, D.E., Briegel, H.J.: One-way Quantum Computation — a tutorial introduction. arXiv:quant-ph/0603226 (2006)
8. Browne, D.E., Kashefi, E., Mhalla, M., Perdrix, S.: Generalized flow and determinism in measurement-based quantum computation. New J. Physics 9, 250 (2007) arXiv:quant-ph/0702212
9. Dawson, C.M., Haselgrove, H.L., Hines, A.P., Mortimer, D., Nielsen, M.A., Osborne, T.J.: Quantum computing and polynomial equations over $\mathbb{Z}_2$. Quantum Information & Computation 5 (2), 102–112 (2004) arXiv:quant-ph/0408129
10. Schlingemann, D.: Cluster states, algorithms and graphs. Quantum Information & Computation 4(4), 287–324 (2004)
11. Dehaene, J., De Moor, B.: Clifford group, stabilizer states, and linear and quadratic operations over GF(2). Physical Review A 68(042318) (2003) arXiv:quant-ph/0304125
12. de Beaudrap, N., Danos, V., Kashefi, E.: Phase map decompositions for unitaries (2006) arXiv:quant-ph/0603266
13. Danos, V., Kashefi, E., Panangaden, P.: The measurement calculus. J. ACM 54(8) (2007) arXiv:quant-ph/0412135
14. Danos, V., Kashefi, E., Panangaden, P.: Parsimonious and robust realizations of unitary maps in the one-way model. Physical Review A 72(064301) (2005) arXiv:quant-ph/0411071
15. Mhalla, M., Perdrix, S.: Finding optimal flows efficiently (2007) arXiv:0709.2670
16. Kitaev, A., Shen, A., Vylalyi, M.: Classical and quantum computation. Graduate Texts in Mathematics 47 (2002)

17. de Beaudrap, N., Pei, M.: An extremal result for geometries in the one-way measurement model. Quantum Information and Computation 8(5), 430–437 (2008) arXiv:quant-ph/0702229
18. Gottesman, D.: Stabilizer codes and quantum error correction. PhD thesis, Caltech (1997) arXiv:quant-ph/9705052
19. Hein, M., Eisert, J., Briegel, H.J.: Multi-party entanglement in graph states. Physical Review A 69(62311) (2004) arXiv:quant-ph/0307130
20. Aaronson, S., Gottesman, D.: Improved simulation of stabilizer circuits. Physical Review A 70(052328) (2004) arXiv:quant-ph/0406196
21. de Beaudrap, N.: Finding flows in the one-way measurement model. Physical Review A 77(022328) (2008) arXiv:quant-ph/0611284
22. Fowler, A., Devitt, S., Hollenberg, L.: Implementation of Shor's algorithm on a linear nearest neighbor qubit array. Quantum Information & Computation 4(4) 237–251 (2004)
23. Patel, K.N., Markov, I.L., Hayes, J.P.: Efficient synthesis of linear reversible circuits. Quantum Information & Computation 8 (to appear, 2002) arXiv:quant-ph/0302002

A Quadratic Form Expansions as Sums over Paths

Let (G, I, O) be the geometry of a quadratic form expansion, as defined on page 34. In the special case when (G, I, O) has a fractional-edge flow as defined in Section 3.2, the quadratic form expansion corresponds exactly to a sum over paths as described in [9], for the elementary gate set of H, Z^t, and $\wedge Z^t$, where $t \in R$ (*i.e.* admitting arbitrary Z rotations and fractional controlled-Z gates). In order to demonstrate the sense in which quadratic form expansions are sums over paths in this case, and because it represents a reasonably simple algorithm for converting quantum circuits into quadratic form expansions, we now present an alternate proof of Theorem 1 based on the techniques of [9]. That any quadratic form expansion with geometry with a fractional-edge flow can be constructed in this way follows by reversing the construction below.

Proof of Theorem 1. Consider a quantum circuit implementing U exactly, using the operations H, $\wedge Z^t$, and Z^t. Enumerate the wires of the circuit from 1 to k, and for each wire $1 \leqslant j \leqslant k$, introduce a *path label* x_j for the input end of the wire, corresponding to an input bit $x_j \in \{0, 1\}$. We set $I = \{1, \ldots, k\}$. Divide each wire into *segments*, bounded on each end by either a Hadamard gate, the input terminal of the wire, or the output terminal. We label the wire segments with path variables: for the segments at the inputs, we apply the labels x_j for $j \in I$, and we introduce new path variables to label the remaining wire segments. Computational paths in the circuit are then described by setting all of the the path variables $x_1 \cdots x_n$ collectively to some particular binary string in $\{0, 1\}^n$. The phase contribution of each paths, governing how they interfere to produce an output state for any given input state, is described by a function $\varphi(\mathbf{x})$ depending the gates of the circuit as follows:

(*i*) For every Hadamard gate on a single wire, with a path variable x_h labelling the segment preceding the Hadamard and a path variable x_j labelling the segment following the Hadamard, we add a term $x_h x_j$.

(*ii*) For every $\wedge Z^t$ operation between two wires, with a path variable x_h labelling the segment of one wire and x_j labelling the segment of the other wire in which the $\wedge Z^t$ operation is performed, we add a term $t x_h x_j$.

(*iii*) For every Z^t operation on a wire segment labelled with a path variable x_j , we add a term tx_j^2. (Because the path variable x_j ranges over $\{0,1\}$, the extra power of 2 has no effect.)

In particular, the function $\varphi(\mathbf{x})$ is a quadratic form, where without loss of generality the coefficients may be constrained to $-1 < t \leqslant 1$. The phase of a given path, described by a bit-string $\mathbf{x} \in \{0,1\}^n$, is then given by $(-1)^{\varphi(\mathbf{x})} = \mathrm{e}^{i\pi\varphi(\mathbf{x})}$. Each path also has an associated amplitude of $2^{-r/2}$, where $r = n - k$ is the number of Hadamard gates in the circuit.[12]

Let O be the set of indices j such that some wire is labelled by the path-variable x_j at its' output end. Then, the initial points of computational paths are described by bit-vectors $\mathbf{a} \in \{0,1\}^I$, and the terminal points of paths are described by $\mathbf{b} \in \{0,1\}^O$. The coefficients $U_{\mathbf{b},\mathbf{a}}$ can then be given as the sum of the contributions of all paths beginning at $\mathbf{x}_I = \mathbf{a}$ and ending at $\mathbf{x}_O = \mathbf{b}$:

$$U_{\mathbf{b},\mathbf{a}} \;=\; \frac{1}{\sqrt{2^r}} \sum_{\substack{\mathbf{x}\in\{0,1\}^n \\ \mathbf{x}_I=\mathbf{a} \\ \mathbf{x}_O=\mathbf{b}}} \mathrm{e}^{i\pi\varphi(\mathbf{x})} \,, \tag{30}$$

which is an expression of the coefficients of U as a quadratic form expansion.

To obtain a proof of Theorem 1, it is sufficient to note that without loss of generality we may restrict ourselves to using $\wedge Z^t$ gates only for $t = 1$ to implement U exactly; and that to implement U to arbitrary precision, it suffices to use Z^t gates where t is restricted to multiples of $\frac{1}{4}$. □

[12] Although it is quite reasonable to consider φ to be simply a polynomial over $\mathbb{R}$, in terms of the descriptions used in Section VI of [9], one may consider φ to be a polynomial over the ring $\mathbb{R}/2\mathbb{Z}$. If we restrict to $t \in \frac{\pi}{4}\mathbb{Z}$, we may simplify this to the finite ring $\mathbb{Z}_8$ by multiplying all of the coefficients by 4, and using it to describe powers of $\sqrt{i}$ rather than of -1.

Improved Constructions of Quantum Automata

Andris Ambainis and Nikolajs Nahimovs*

Department of Computer Science, University of Latvia, Raina bulv. 19, Riga, LV-1586, Latvia
andris.ambainis@lu.lv, kolja.nahimov@gmail.com

Abstract. We present a simple construction of quantum automata which achieve an exponential advantage over classical finite automata. Our automata use $\frac{4}{\epsilon}\log 2p + O(1)$ states to recognize a language that requires p states classically. The construction is both substantially simpler and achieves a better constant in the front of $\log p$ than the previously known construction of [2].

Similarly to [2], our construction is by a probabilistic argument. We consider the possibility to derandomize it and present some preliminary results in this direction.

1 Introduction

Quantum finite automata are a mathematical model for quantum computers with limited memory. A quantum finite automaton has a finite state space and applies a sequence of transformations, corresponding to the letter of the input word to this state space. At the end, the state of the quantum automaton is measured and the input word is accepted or rejected, depending on the outcome of the measurement.

Most commonly, finite automata (including quantum finite automata) are studied in 1-way model where the transformations corresponding to the letters of the input word are applied in the order of the letters in the word, from the left to the right. (More general 2-way models [8] allow the order of the transformations to depend on the results of the previous transformations.)

For 1-way model (which we consider the most natural model in the quantum setting), the set of languages (computational problems) that can be recognized (computed) by a quantum automaton is the same for classical automata[1]. However, quantum automata can be exponentially more space-efficient than classical automata [2]. This is one of only two results that show an exponential advantage for quantum algorithms in space complexity. (The other is the recent exponential separation for online algorithms by Le Gall [9].)

* Supported by University of Latvia research project Y2-ZP01-100.

[1] More precisely, this is true for sufficiently general models of quantum automata, such as one proposed in [5] or [7]. There are several results claiming that quantum automata are weaker than classical (e.g. [8,3,4]) but this is an artifact of restrictive models of quantum automata being used.

Y. Kawano and M. Mosca (Eds.): TQC 2008, LNCS 5106, pp. 47–56, 2008.

Our first result is an improved exponential separation between quantum and classical finite automata, for the same computational problem as in [2]. The construction in [2] is quite inefficient. While it produces an example where classical automata require p states and quantum automata require $C \log p$ states, the constant C is fairly large. In this paper, we provide a new construction with a better constant and, also, a much simpler analysis. (A detailed comparison between our results and [2] is given in section 3.1.)

Second, both construction of QFAs in [2] and this paper are probabilistic. That is, they employ a sequence of parameters that are chosen at random and hardwired into the QFA. In the last section, we give two non-probabilistic constructions of QFAs for the same language. The first of them gives QFAs with $O(\log p)$ states but its correctness is only shown by numerical experiments. The second construction gives QFAs with $O(\log^{2+\epsilon} p)$ states but is provably correct.

2 Definitions

2.1 Quantum Finite Automata

We consider 1-way quantum finite automata (QFA) as defined in [10]. Namely, a 1-way QFA is a tuple $M = (Q, \Sigma, \delta, q_0, Q_{acc}, Q_{rej})$ where Q is a finite set of states, Σ is an input alphabet, δ is a transition function, $q_0 \in Q$ is a starting state, Q_{acc} and Q_{rej} are sets of accepting and rejecting states and $Q = Q_{acc} \cup Q_{rej}$. $\not\in$ and \$ are symbols that do not belong to Σ. We use $\not\in$ and \$ as the left and the right endmarker, respectively. The *working alphabet* of M is $\Gamma = \Sigma \cup \{\not\in, \$\}$.

A superposition of M is any element of $l_2(Q)$ (the space of mappings from Q to $\mathbb{C}$ with l_2 norm). For $q \in Q$, $|q\rangle$ denotes the unit vector with value 1 at q and 0 elsewhere. All elements of $l_2(Q)$ can be expressed as linear combinations of vectors $|q\rangle$. We will use ψ to denote elements of $l_2(Q)$.

The transition function δ maps $Q \times \Gamma \times Q$ to $\mathbb{C}$. The value $\delta(q_1, a, q_2)$ is the amplitude of $|q_2\rangle$ in the superposition of states to which M goes from $|q_1\rangle$ after reading a. For $a \in \Gamma$, V_a is a linear transformation on $l_2(Q)$ defined by

$$V_a(|q_1\rangle) = \sum_{q_2 \in Q} \delta(q_1, a, q_2)|q_2\rangle. \tag{1}$$

We require all V_a to be unitary.

The computation of a QFA starts in the superposition $|q_0\rangle$. Then transformations corresponding to the left endmarker $\not\in$, the letters of the input word x and the right endmarker \$ are applied. The transformation corresponding to $a \in \Gamma$ is just V_a. If the superposition before reading a is ψ, then the superposition after reading a is $V_a(\psi)$.

After reading the right endmarker, the current state ψ is observed with respect to the observable $E_{acc} \oplus E_{rej}$ where $E_{acc} = span\{|q\rangle : q \in Q_{acc}\}$, $E_{rej} = span\{|q\rangle : q \in Q_{rej}\}$. This observation gives $x \in E_i$ with the probability equal to the square of the projection of ψ to E_i. After that, the superposition collapses to this projection.

If we get $\psi \in E_{acc}$, the input is accepted. If $\psi \in E_{rej}$, the input is rejected.

Another definition of QFAs. Independently of [10], quantum automata were introduced in [8]. There is one difference between these two definitions. In [8], a QFA is observed after reading each letter (after doing each V_a). In [10], a QFA is observed only after all letters have been read. The definition of [8] is more general. But, in this paper, we follow the definition of [10] because it is simpler and sufficient to describe our automaton.

2.2 Unitary Transformations

We use the following theorem from linear algebra.

Theorem 1. *Let $\alpha_1, \ldots, \alpha_m$ be such that $|\alpha_1|^2 + \ldots + |\alpha_m|^2 = 1$. Then,*

1. *there is a unitary transformation U_1 such that $U_1|q_1\rangle = \alpha_1|q_1\rangle + \ldots + \alpha_m|q_m\rangle$.*
2. *there is a unitary transformation U_2 such that, for all $i \in \{1, \ldots, m\}$, $U_2|q_i\rangle$ is equal to $\alpha_i|q_1\rangle$ plus some combination of $|q_2\rangle, \ldots, |q_m\rangle$.*

In the second case, we also have

$$U_2(\alpha_1|q_1\rangle + \ldots + \alpha_m|q_m\rangle) = |q_1\rangle.$$

3 Space-Efficient Quantum Automaton

3.1 Summary of Results

Let p be a prime. We consider the language $L_p = \{\ a^i \mid i \text{ is divisible by } p\ \}$. It is easy to see that any deterministic 1-way finite automaton recognizing L_p has at least p states. However, there is a much more efficient QFA! Namely, Ambainis and Freivalds [2] have shown that L_p can be recognized by a QFA with $O(\log p)$ states.

The big-O constant in this result depends on the required probability of correct answer. For $x \in L_p$, the answer is always correct with probability 1. For $x \notin L_p$, [2] give

- A QFA with $16 \log p$ states that is correct with probability at least 1/8 on inputs $x \notin L_p$.
- A QFA with $poly(\frac{1}{\epsilon}) \log p$ states that is correct with probability at least $1-\epsilon$ on inputs $x \notin L_p$ (where $poly(x)$ is some polynomial in x).

In this paper, we present a simpler construction of QFAs that achieves a better big-O constant.

Theorem 2. *For any $\epsilon > 0$, there is a QFA with $4\frac{\log 2p}{\epsilon}$ states recognizing L_p with probability at least $1 - \epsilon$.*

3.2 Proof of Theorem 2

Let U_k, for $k \in \{1, \ldots, p-1\}$, be a quantum automaton with a set of states $Q = \{q_0, q_1\}$, a starting state $|q_0\rangle$, $Q_{acc} = \{q_0\}$, $Q_{rej} = \{q_1\}$. The transition function is defined as follows. Reading a maps $|q_0\rangle$ to $\cos\phi|q_0\rangle + \sin\phi|q_1\rangle$ and

$|q_1\rangle$ to $-\sin\phi|q_0\rangle + \cos\phi|q_1\rangle$ where $\phi = \frac{2\pi k}{p}$. (It is easy to check that this transformation is unitary.) Reading ¢ and \$ leaves $|q_0\rangle$ and $|q_1\rangle$ unchanged.

Lemma 1. *After reading a^j, the state of U_k is*

$$\cos\left(\frac{2\pi jk}{p}\right)|q_0\rangle + \sin\left(\frac{2\pi jk}{p}\right)|q_1\rangle.$$

Proof. By induction. □

If j is divisible by p, then $\frac{2\pi jk}{p}$ is a multiple of 2π, $\cos(\frac{2\pi jk}{p}) = 1$, $\sin(\frac{2\pi jk}{p}) = 0$, reading a^j maps the starting state $|q_0\rangle$ to $|q_0\rangle$. Therefore, we get an accepting state with probability 1. This means that all automata U_k accept words in L with probability 1.

Let $k_1, \ldots, k_d$ be a sequence of $d = c\log p$ numbers. We construct an automaton U by combining U_{k_1}, …, U_{k_d}. The set of states consists of $2d$ states $q_{1,0}$, $q_{1,1}$, $q_{2,0}$, $q_{2,1}$, …, $q_{d,0}$, $q_{d,1}$. The starting state is $q_{1,0}$.

The transformation for left endmarker ¢ is such that $V_{¢}(|q_{1,0}\rangle) = |\psi_0\rangle$ where

$$|\psi_0\rangle = \frac{1}{\sqrt{d}}(|q_{1,0}\rangle + |q_{2,0}\rangle + \ldots |q_{d,0}\rangle).$$

This transformation exists by first part of Theorem 1. The transformation for a is defined by

$$V_a(|q_{i,0}\rangle) = \cos\frac{2k_i\pi}{p}|q_{i,0}\rangle + \sin\frac{2k_i\pi}{p}|q_{i,1}\rangle,$$

$$V_a(|q_{i,1}\rangle) = -\sin\frac{2k_i\pi}{p}|q_{i,0}\rangle + \cos\frac{2k_i\pi}{p}|q_{i,1}\rangle.$$

The transformation $V_{\$}$ is as follows. The states $|q_{i,1}\rangle$ are left unchanged. On the states $|q_{i,0}\rangle$, $V_{\$}|q_{i,0}\rangle$ is $\frac{1}{\sqrt{d}}|q_{1,0}\rangle$ plus some other state (part 2 of Theorem 1, applied to $|q_{1,0}\rangle$, …, $|q_{d,0}\rangle$). In particular,

$$V_{\$}|\psi_0\rangle = |q_{1,0}\rangle.$$

The set of accepting states Q_{acc} consists of one state $q_{1,0}$. All other states $q_{i,j}$ belong to Q_{rej}.

Claim. If the input word is a^j and j is divisible by p, then U accepts with probability 1.

Proof. The left endmarker maps the starting state to $|\psi_0\rangle$. Reading j letters a maps each $|q_{i,0}\rangle$ to itself (see analysis of U_k). Therefore, the state $|\psi_0\rangle$ which consists of various $|q_{i,0}\rangle$ is also mapped to itself. The right endmarker maps $|\psi_0\rangle$ to $|q_{1,0}\rangle$ which is an accepting state. □

Claim. If the input word is a^j, j not divisible by p, U accepts with probability

$$\frac{1}{d^2}\left(\cos\frac{2\pi k_1 j}{p} + \cos\frac{2\pi k_2 j}{p} + \ldots + \cos\frac{2\pi k_d j}{p}\right)^2. \tag{2}$$

Proof. By Lemma 1, a^j maps $|q_{i,0}\rangle$ to $\cos\frac{2\pi k_i j}{p}|q_{i,0}\rangle + \sin\frac{2\pi k_i j}{p}|q_{i,1}\rangle$. Therefore, the state before reading the right endmarker \$ is

$$\frac{1}{\sqrt{d}}\sum_{i=1}^{d}(\cos\frac{2\pi k_i j}{p}|q_{i,0}\rangle + \sin\frac{2\pi k_i j}{p}|q_{i,1}\rangle).$$

The right endmarker maps each $|q_{i,0}\rangle$ to $\frac{1}{\sqrt{d}}|q_{1,0}\rangle$ plus superposition of other basis states. Therefore, the state after reading the right endmarker \$ is

$$\frac{1}{d}\sum_{i=1}^{d}\cos\frac{2\pi k_i j}{p}|q_{1,0}\rangle$$

plus other states $|q_{i,j}\rangle$. Since $|q_{1,0}\rangle$ is the only accepting state, the probability of accepting is the square of the coefficient of $|q_{1,0}\rangle$. This proves the lemma. □

We use the following theorem from probability theory (variant of Azuma's theorem[11]).

Theorem 3. *Let $X_1, \ldots, X_d$ be independent random variables such that $E[X_i]$= 0 and the value of X_i is always between -1 and 1. Then,*

$$Pr[|\sum_{i=1}^{d} X_i| \geq \lambda] \leq 2e^{-\frac{\lambda^2}{2d}}.$$

We apply this theorem as follows. Fix $j \in \{1, \ldots, p-1\}$. Pick each of $k_1, \ldots, k_d$ randomly from $\{0, \ldots, p-1\}$. Define $X_i = \cos\frac{2\pi k_i j}{p}$. We claim that X_i satisfy the conditions of theorem. Obviously, the value of cos function is between -1 and 1. The expectation of X_i is

$$E[X_i] = \frac{1}{p}\sum_{k=0}^{p-1}\cos\frac{2\pi kj}{p}$$

since $k_i = k$ for each $k \in \{0, \ldots, p-1\}$ with probability $1/p$. We have $\cos\frac{2\pi kj}{p} = \cos\frac{2\pi(kj \bmod p)}{p}$ because $\cos(2\pi+x) = \cos x$. Consider the numbers 0, j, $2j \bmod p$, …, $(p-1)j \bmod p$. They are all distinct. (Since p is prime, $kj = k'j (\mathrm{mod} p)$ implies $k = k'$.) Therefore, the numbers 0, j, $2j \bmod p$, …, $(p-1)j \bmod p$ are just $0, 1, \ldots, p-1$ in a different order. This means that the expectation of X_i is

$$E[X_i] = \frac{1}{p}\sum_{k=0}^{p-1}\cos\frac{2\pi k}{p}.$$

This is equal to 0.

By equation (2), the probability of accepting a^j is $\frac{1}{d^2}(X_1 + \ldots + X_d)^2$. To achieve

$$\frac{1}{d^2}(X_1 + \ldots + X_d)^2 \leq \epsilon,$$

we need $|X_1 + \ldots + X_d| \leq \sqrt{\epsilon}d$. By Theorem 3, the probability that this does not happen is at most $2e^{-\frac{\epsilon d}{2}}$.

There are $p-1$ possible inputs not in L: a^1, ..., a^{p-1}. The probability that one of them gets accepted with probability more than ϵ is at most $2(p-1)e^{-\frac{\epsilon d}{2}}$. If

$$2(p-1)e^{-\frac{\epsilon d}{2}} < 1, \tag{3}$$

then there is at least one choice of $k_1, \ldots, k_d$ for which U does not accept any of a^1, ..., a^{p-1} with probability more than ϵ. The equation (3) is true if we take $d = 2\frac{\log 2p}{\epsilon}$. The number of states for U is $4\frac{\log 2p}{\epsilon}$. □

4 Explicit Constructions of QFAs

In the previous section, we proved what for every $\epsilon > 0$ and $p \in P$, there is a QFA with $4\frac{\log 2p}{\epsilon}$ states recognizing L_p with probability at least $1-\epsilon$. The proposed QFA construction depends on $d = 2\frac{\log 2p}{\epsilon}$ parameters $k_1, \ldots, k_d$ and accepts input word $a^j \notin L_p$ with probability

$$\frac{1}{d^2}\left(\sum_{i=1}^{d} \cos\frac{2\pi k_i j}{p}\right)^2.$$

It is possible to choose $k_1, \ldots, k_d$ values to ensure

$$\frac{1}{d^2}\left(\sum_{i=1}^{d} \cos\frac{2\pi k_i j}{p}\right)^2 < \epsilon$$

or, equivalently,

$$\left|\sum_{i=1}^{d} \cos\frac{2\pi k_i j}{p}\right| < \sqrt{\epsilon} d \tag{4}$$

for every $a^j \notin L_p$.

However, our proof is by a probabilistic argument and does not give an explicit sequence $k_1, \ldots, k_d$. We now present two constructions of explicit sequences. The first construction works well in numerical experiments and gives a QFA with $O(\log p)$ states in all the cases that we tested. The second construction uses a slightly larger number of states but has a rigorous proof of correctness.

4.1 The First Construction: Cyclic Sequences

We conjecture

Hypothesis 1. *If g is a primitive root modulo $p \in P$, then sequence $S_g = \{k_i \equiv g^i \bmod p\}_{i=1}^{d}$ for all d and all $j : a^j \notin L_p$ satisfies (4).*

We will call g a *sequence generator*. The corresponding sequence will be referred as cyclic sequence. We have checked all $p \in \{2, \ldots, 9973\}$, all generators g and all sequence lengths $d < p$ (choosing a corresponding ϵ value) and haven't found any counterexample to our hypothesis.

Table 1. ϵ_{rand} and ϵ_g for different p and g

p	ϵ	d	g	ϵ_{rand}	ϵ_g
1523	0,1	161	948	0,03635	0,01517
2689	0,1	172	656	0,03767	0,01950
3671	0,1	179	2134	0,03803	0,02122
4093	0,1	181	772	0,03822	0,01803
5861	0,1	188	2190	0,03898	0,01825
6247	0,1	189	406	0,03922	0,02006
7481	0,1	193	6978	0,03932	0,01691
8581	0,1	196	5567	0,03942	0,02057
9883	0,1	198	1260	0,04011	0,01905

We now describe numerical experiments comparing two strategies: using a random sequence $k_1, \ldots, k_d$ and using a cyclic sequence.

We will use S_{rand} to denote random sequence and S_g to denote a cyclic sequence with generator g. We will also use ϵ_{rand} and ϵ_g to denote the maximum probability with which a corresponding automata accepts input word $a^j \notin L_p$.

Table 1 shows ϵ_{rand} and ϵ_g for different p and g values. ϵ_{rand} is calculated as an average over 5000 randomly selected sequences. ϵ_g is for one specific generator. ϵ in the second column shows the theoretical upper bound given by Theorem 2.

In 99.98% - 99.99% of our experiments, random sequences achieved the bound of Theorem 2. Surprisingly, cyclic sequences substantially outperform random ones in almost all the cases.

More precisely, for randomly selected $p \in P$, $\epsilon > 0$ and generator g, a cyclic sequence S_g gives a better result than a random sequence S_{rand} in 98.29% of cases. A few random instances are shown in Figure 1. For each instance, we show the bound $d\sqrt{\epsilon}$ on (4) obtained by a probabilistic argument, the maximum of $f_{rand}(j)$ (which is defined as the value of (4) for the sequence S_{rand}) over all j, $a^j \notin L_p$ and the maximum of $f_g(j)$ (defined in a similar way using S_g instead of S_{rand}).

In 1.81% of cases, we got that $\sup |f_g(j)| > \sup |f_{rand}(j)|$, where $\sup |f_{rand}(j)|$ is calculated as an average over 5000 randomly selected sequences. Figure 2 shows one of these cases: $p = 9059$, $\epsilon = 0.09$ and $g = 2689$, comparing the cyclic sequence with 9 different randomly chosen sequences. The cyclic sequence gives a slightly worse result than most of the random ones, but still beats the probabilistic bound on (4) by a substantial amount.

Comparing different generators. Every $p \in P$ might have multiple generators. Table 2 shows ϵ_g values for $p = 9059$ and $\epsilon = 0.1$ (sequence length $d = 197$, $\sqrt{\epsilon}d = 62.0101221453601$).

Different generators have different ϵ_g values. We will use g_{min} to refer a minimal generator, i.e. one having a minimal ϵ_g. Table 3 shows minimal generators for p values from table 1.

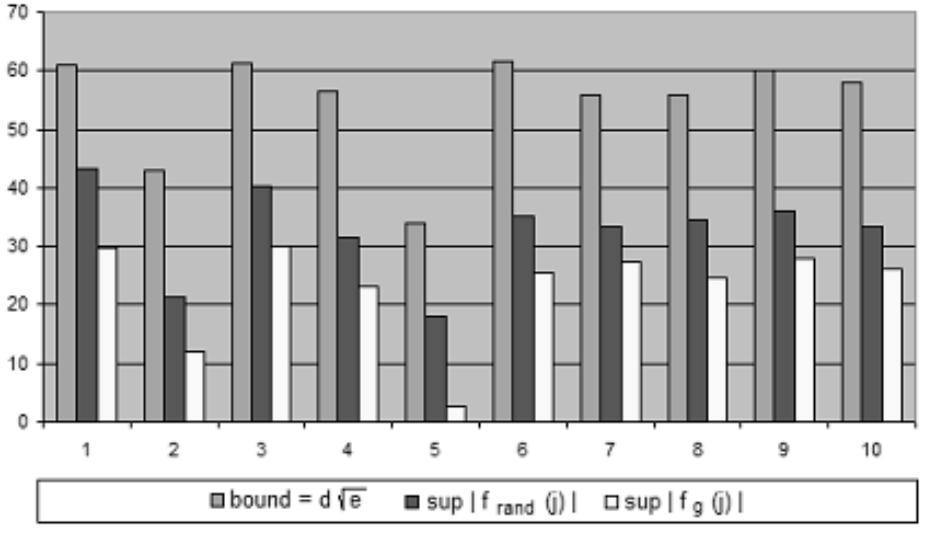


Fig. 1. $\sup |f_g(j)|$ and $\sup |f_{rand}(j)|$ for random p, ϵ and g

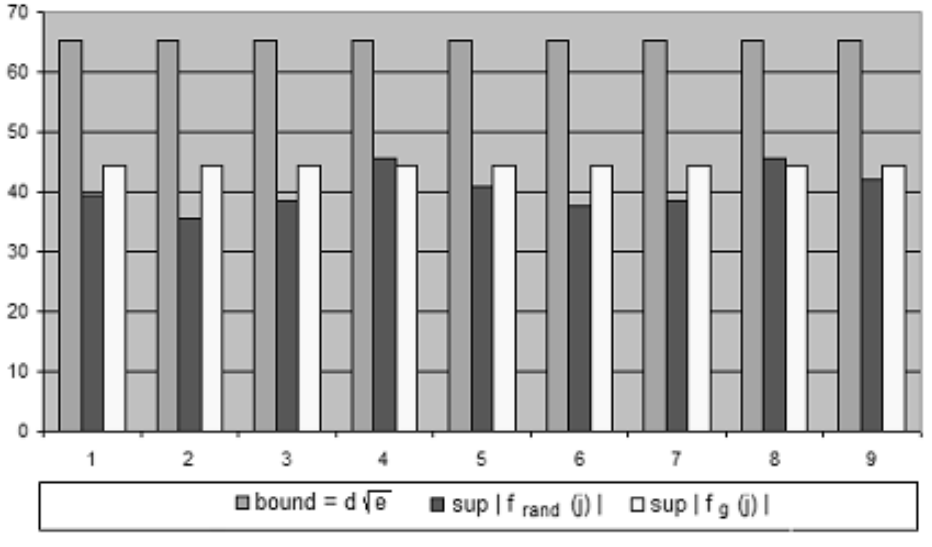


Fig. 2. $\sup |f_g(j)|$ and $\sup |f_{rand}(j)|$ for $p = 9059$, $\epsilon = 0.09$ and $g = 2689$

We see that, typically, the minimal generators give a QFA with substantially smaller probability of error. It remains open whether one could find a minimal generator without an exhaustive search of all generators.

Table 2. ϵ_g values for different generators. $p = 9059$

g	ϵ_g	g	ϵ_g	g	ϵ_g
102	0,02533	1545	0,01858	9023	0,01807
103	0,03758	1546	0,02235	9033	0,01413
105	0,01999	1549	0,02896	9034	0,01485
106	0,02852	1552	0,02873	9036	0,02509
110	0,01685	1553	0,02624	9039	0,02311

Table 3. Minimal generators for different p

p	ϵ	d	g	ϵ_g	g_{min}	$\epsilon_{g_{min}}$
1523	0,1	161	948	0,01517	624	0,00919
2689	0,1	172	656	0,01950	1088	0,01060
3671	0,1	179	2134	0,02122	1243	0,01121
4093	0,1	181	772	0,01803	1063	0,01154
5861	0,1	188	2190	0,01825	5732	0,01133
6247	0,1	189	406	0,02006	97	0,01182
7481	0,1	193	6978	0,01691	2865	0,01205
8581	0,1	196	5567	0,02057	4362	0,01335
9883	0,1	198	1260	0,01905	5675	0,01319

4.2 The Second Construction: AIKPS Sequences

Fix $\epsilon > 0$. Let

$$P = \{r | r \text{ is prime, } (\log p)^{1+\epsilon}/2 < r \leq (\log p)^{1+\epsilon}\},$$

$$S = \{1, 2, \ldots, (\log p)^{1+2\epsilon}\},$$

$$T = \{s \cdot r^{-1} | r \in R, s \in S\},$$

with r^{-1} being the inverse modulo p. Ajtai et al. [1] have shown

Theorem 4. *[1] For all* $k\{1, \ldots, p-1\}$,

$$|\sum_{t \in T} e^{2tk\pi i/p}| \leq (\log p)^{-\epsilon}|T|.$$

Razborov et al. [12] have shown that powers $e^{2tk\pi i/p}$ satisfy even stronger uniformity conditions. We, however, only need Theorem 4.

By taking the real part of the left hand side, we get

$$\left|\sum_{t \in T} \cos\left(\frac{2tk\pi i}{p}\right)\right| \leq (\log p)^{-\epsilon}|T|.$$

Thus, taking our construction of QFAs and using elements of T as $k_1, \ldots, k_d$ gives an explicit construction of a QFA for our language with $O(\log^{2+3\epsilon})$ states.

For our first, cyclic construction, the best provable result is by applying a bound on exponential sums by Bourgain [6]. That gives a QFA with $O(p^{c/\log\log p})$ states which is weaker than both the numerical results and the rigorous construction in this section.

Acknowledgment. We thank Igor Shparlinski for pointing out [1] and [6] to us.

References

1. Ajtai, M., Iwaniec, H., Komlos, J., Pintz, J., Szemeredi, E.: Construction of a thin set with small Fourier coefficients. Bulletin of the London Mathematical Society 22, 583–590 (1990)
2. Ambainis, A., Freivalds, R.: 1-way quantum finite automata: strengths, weaknesses and generalizations. In: Proceedings of the 39th IEEE Conference on Foundations of Computer Science, pp. 332–341 (1998) (quant-ph/9802062)
3. Ambainis, A., Kikusts, A., Valdats, M.: On the Class of Languages Recognizable by 1-Way Quantum Finite Automata. In: Ferreira, A., Reichel, H. (eds.) STACS 2001. LNCS, vol. 2010, pp. 75–86. Springer, Heidelberg (2001)
4. Ambainis, A., Nayak, A., Ta-Shma, A., Vazirani, U.: Dense quantum coding and quantum finite automata. Journal of the ACM 49(4), 496–511 (2002)
5. Bertoni, A., Mereghetti, C., Palano, B.: Quantum Computing: 1-Way Quantum Automata. In: Ésik, Z., Fülöp, Z. (eds.) DLT 2003. LNCS, vol. 2710, pp. 1–20. Springer, Heidelberg (2003)
6. Bourgain, J.: Estimates on exponential sums related to Diffie-Hellman distributions. Geometric and Functional Analysis 15, 1–34 (2005)
7. Ciamarra, M.: Quantum Reversibility and a New Model of Quantum Automaton. In: Freivalds, R. (ed.) FCT 2001. LNCS, vol. 2138, pp. 376–379. Springer, Heidelberg (2001)
8. Kondacs, A., Watrous, J.: On the power of quantum finite state automata. In: Proceedings of the 38th IEEE Conference on Foundations of Computer Science, pp. 66–75 (1997)
9. Le Gall, F.: Exponential separation of quantum and classical online space complexity. In: Proceedings of SPAA 2006, pp. 67–73 (2006)
10. Moore, C., Crutchfield, J.: Quantum automata and quantum grammars. Theoretical Computer Science 237(1-2), 275–306 (2000) (quant-ph/9707031)
11. Motwani, R., Raghavan, P.: Randomized Algorithms. Cambridge University Press, Cambridge (1994)
12. Razborov, A., Szemeredi, E., Wigderson, A.: Constructing small sets that are uniform in arithmetic progressions. Combinatorics, Probability and Computing 2, 513–518 (1993)

An Application of the Deutsch-Jozsa Algorithm to Formal Languages and the Word Problem in Groups

Michael Batty[1], Andrea Casaccino[2,⋆], Andrew J. Duncan[1], Sarah Rees[1], and Simone Severini[3]

[1] Department of Mathematics, University of Newcastle upon Tyne, United Kingdom
[2] Information Engineering Department, University of Siena, Italy
ndr981@tin.it
[3] Institute for Quantum Computing and Department of Combinatorics and Optimization, University of Waterloo, Canada

Abstract. We adapt the Deutsch-Jozsa algorithm to the context of formal language theory. Specifically, we use the algorithm to distinguish between trivial and nontrivial words in groups given by finite presentations, under the promise that a word is of a certain type. This is done by extending the original algorithm to functions of arbitrary length binary output, with the introduction of a more general concept of parity. We provide examples in which properties of the algorithm allow to reduce the number of oracle queries with respect to the deterministic classical case. This has some consequences for the word problem in groups with a particular kind of presentation.

1 The Deutsch-Jozsa Algorithm and Formal Languages

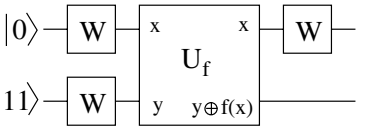
We apply a direct generalization of the Deutsch-Jozsa algorithm to the context of formal language theory. More particularly, we extend the algorithm to distinguish between trivial and nontrivial words in groups given by finite presentations, under the promise that a word is of a certain type. For background information, we refer the reader to [1] and [2].

The Deutsch-Jozsa algorithm concerns maps $f : \{0,1\}^n \longrightarrow \{0,1\}$, which we may think of as words of length 2^n in a two-letter alphabet. Instead, let us consider words of length 2^n in a four-letter alphabet $\mathcal{A} = \{a, b, c, d\}$. We identify the letters with binary strings of length 2: $a \leftrightarrow 00$, $b \leftrightarrow 01$, $c \leftrightarrow 10$ and $d \leftrightarrow 11$. In this way we can look at words of length 2^n as in one-to-one correspondence with maps $f : \{0,1\}^n \longrightarrow \{00, 01, 10, 11\}$.

First, consider the case $n = 1$, that is when the words have length 2. As with the standard formulation of the Deutsch-Jozsa algorithm, the promise will be vacuous in this case. We use the quantum circuit represented below. The circuit essentially implements the Deutsch's algorithm, but with input $|11\rangle$ rather than $|1\rangle$:

⋆ Corresponding author.

Y. Kawano and M. Mosca (Eds.): TQC 2008, LNCS 5106, pp. 57–69, 2008.



After applying the Hadamard gates, the state of the system is

$$W \otimes W_2(|0\rangle \otimes |11\rangle) = \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right) \otimes \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)^{\otimes 2}.$$

If $x \in \{0,1\}$ then we have

$$\begin{aligned} & U_f \left(|x\rangle \otimes \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)^{\otimes 2} \right) \\ &= |x\rangle \otimes \frac{1}{2} \left(|00 \oplus f(x)\rangle - |01 \oplus f(x)\rangle - |10\rangle \oplus f(x)\rangle + |11 \oplus f(x)\rangle \right) \\ &= (-1)^{p(f(x))} |x\rangle \otimes \left(\frac{|1\rangle - |0\rangle}{\sqrt{2}}\right)^{\otimes 2}. \end{aligned}$$

For a binary string y, we denote by $p(y)$ the *parity* of y, that is if m is the number of 1s in y then $p(y) = m(mod 2)$. After querying the oracle U_f, the state is

$$\frac{(-1)^{p(f(0))}|0\rangle + (-1)^{p(f(1))}|1\rangle}{\sqrt{2}} \otimes \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)^{\otimes 2}.$$

Finally, after the last Hadamard gate, the first qubit is in the state $|0\rangle$ if $p(f(0)) = p(f(1))$ or $|1\rangle$ if $p(f(0)) \neq p(f(1))$. We shall say that f is *parity constant* if $p(f(0)) = p(f(1))$; *parity balanced*, otherwise. By measuring the final state, we obtain $|0\rangle$ with probability 1 if f is parity balanced and $|1\rangle$ with probability 1 if f is parity constant.

Let us now introduce some terminology related to formal languages. Given a word $w : \{0,1\}^n \longrightarrow \{a,b,c,d\}$, an *anagram* of w is a word of the form $w \circ \phi$, where $\phi : \{0,1\}^n \longrightarrow \{0,1\}^n$ is a permutation. We write $[w]$ for the set of all anagrams of w. More formally, let F denote the free monoid on $\{a,b,c,d\}$ and let M denote the free commutative monoid on $\{a,b,c,d\}$. Let R denote the natural map from F to M and suppose that $w \in M$. Then $R(w) = [w]$, the set of all anagrams of w. It is clear that the definition of parity balanced and parity constant words extends to M in this way. The set of parity constant words is then a union of sets of anagrams

$$\mathcal{C}_1^{11}(a,b,c,d) = [aa] \cup [bb] \cup [cc] \cup [dd] \cup [bc] \cup [ad].$$

Similarly, the set of parity balanced words is

$$\mathcal{B}_1^{11}(a,b,c,d) = [ab] \cup [ac] \cup [bd] \cup [cd].$$

Note that both the terms of the alphabet in the bracket have the same parity with the notation $a \leftrightarrow 00$, $b \leftrightarrow 01$, $c \leftrightarrow 10$ and $d \leftrightarrow 11$.

Suppose not to input $|11\rangle$ into the auxiliary workspace, but rather some other number $0 \leq n \leq 3$. How does this affect the sets of words we can distinguish between? It is interesting to observe that we may define as follows a more general type of "parity". Let x be a nontrivial element of $\{00, 01, 10, 11\}$, where the latter set is considered in the natural way as the vector space $(\mathbb{Z}_2)^2 = \mathbb{Z}_2 \oplus \mathbb{Z}_2$. Define $p^x(y)$ to be equal to 0, if y is in the subspace $\langle x \rangle = \{00, x\}$ and equal to 1, otherwise. With this notation, $p^{11}(y) = p(y)$, the usual parity function. A similar circuit, taking the auxiliary input $\neg(x)$, that is the binary complement of x, will distinguish between whether the word is x-constant or x-balanced (where these terms have the obvious meaning). Again, measurement of the state will yield this information with certainty. It is clear that if $x = 00$ then the output of the circuit is independent of f, and so this is of no use. Let us now suppose that $x = 01$. Then x-constant means that the output of f are in the same coset of $\{0, 1\}$ in $(\mathbb{Z}_2)^2$ and x-balanced means that $f(0)$ and $f(1)$ are in different cosets, or, in other words, both in or out the subspace $\langle x \rangle = \{00, x\}$. The set of 01-constant words is

$$\mathcal{C}_1^{01}(a, b, c, d) = [aa] \cup [bb] \cup [cc] \cup [dd] \cup [ab] \cup [cd]$$

and the set of 01-balanced words is

$$\mathcal{B}_1^{01}(a, b, c, d) = [ac] \cup [ad] \cup [bc] \cup [bd].$$

With the same notation,

$$\mathcal{C}_1^{10}(a, b, c, d) = [aa] \cup [bb] \cup [cc] \cup [dd] \cup [ac] \cup [bd]$$

and

$$\mathcal{B}_1^{10}(a, b, c, d) = [ab] \cup [ad] \cup [bc] \cup [cd].$$

As before, the first term and the second term in the bracket represent the first output and the second output of the function, respectively. Also the parity is the same as described before. Note that when the set is parity constant both terms are in or out the subspace $\langle x \rangle$, while in the parity balanced case one term is in the subspace and the other one is out.

It is now useful to introduce some general notation. Let $x \in \{0, 1\}^2$. We denote the set of *x-constant* words of length 2^n over $\mathcal{A}$ by $\mathcal{C}_n^x(\mathcal{A})$ and the set of *x-balanced* words of length 2^n over $\mathcal{A}$ by $\mathcal{B}_n^x(\mathcal{A})$. We write $\mathcal{F}_n^x(\mathcal{A}) = \mathcal{C}_n^x(\mathcal{A}) \cup \mathcal{B}_n^x(\mathcal{A})$ and call this the set of *x-feasible* words of length 2^n. The following fact is important in the context of our discussion.

Theorem 1. *Given a word $w \in \mathcal{F}_n^x$, we can decide with a single quantum query whether it is in $\mathcal{C}_n^x$ or $\mathcal{B}_n^x$.*

It is useful to observe that already in the seminal work [3], it was pointed out that a classical randomized algorithm solves the Deutsch-Jozsa task with 3 classical queries on average, whereas the quantum approach solve it with probability 1 using one single quantum query (see also [4]). Here the output of the function f is no more a single bit but a bit string. Particulary, the possible output of

the function is an n bit string where n is $\log_2$ (the alphabet symbol length). Therefore, first of all, a word is given by k repeated random output of the function, where k is the length of the word. In other terms, a word is like a sequence obtained by tossing a dice with j faces, where j is the number of letters in the alphabet. The parity constant output and the parity balanced output are not mutually exclusive over a fixed word of length k and do not cover all the possible k output combinations of the function. This means that it is possible to see a word as obtained by k repeated queries to the function (*e.g.*, given a word in a certain set it is possible to deduce the parity of the function).

From this point of view, it is easy to see that the probability of being constant over all possible anagrams, interpreting the output binary string of k queries as anagrams of k letters, is higher than the balanced case, when k is small and the difference decreases while the number of queries to the function increases. As long as any possible parity function partitions into two classes the function co-domain, it is clear that an higher number of possible outputs is not relevant to the number of classical queries required to distinguish between the parity constant and parity balanced cases. What is remarkable about this method is that it allows to extend the Deutsch-Jozsa algorithm to functions with output of any dimension, $f : \{0,1\}^n \longrightarrow \{0,1\}^n$. Defining appropriate parities, like for groups or code membership problems, could give arise to potentially interesting applications.

It is clarifying to fully work out an example for $n = 2$. If X is a subset of $\{01, 10, 11\}$ then we write $\mathcal{F}_n^X(\mathcal{A})$ for $\cap_{x\in X}\mathcal{F}_n^x(\mathcal{A})$. We have

$$\begin{aligned}
\mathcal{C}_2^{11}(a,b,c,d) &= [aaaa] \cup [bbbb] \cup [cccc] \cup [dddd] \cup [aaad] \\
&\quad \cup [aadd] \cup [addd] \cup [bbbc] \cup [bbcc] \cup [bccc], \\
\mathcal{B}_2^{11}(a,b,c,d) &= [aabb] \cup [aacc] \cup [bbdd] \cup [ccdd] \\
&\quad \cup [aabc] \cup [bcdd] \cup [abbd] \cup [accd] \cup [abcd], \\
\mathcal{C}_2^{01}(a,b,c,d) &= [aaaa] \cup [bbbb] \cup [cccc] \cup [dddd] \cup [aaab] \\
&\quad \cup [aabb] \cup [abbb] \cup [cccd] \cup [ccdd] \cup [cddd], \\
\mathcal{B}_2^{01}(a,b,c,d) &= [aacc] \cup [aadd] \cup [bbcc] \cup [bbdd] \\
&\quad \cup [aacd] \cup [bbcd] \cup [abcc] \cup [abdd] \cup [abcd], \\
\mathcal{F}_2^{\{01,11\}} &= [aaaa] \cup [bbbb] \cup [cccc] \cup [dddd] \cup [aabb] \\
&\quad \cup [aacc] \cup [aadd] \cup [bbcc] \cup [bbdd] \cup [ccdd] \cup [abcd].
\end{aligned}$$

When $n = 2$, in the parity balanced case half of the output is of the same parity. We also have

$$\begin{aligned}
\mathcal{B}_2^{11}(a,b,c,d) \cap \mathcal{B}_2^{01}(a,b,c,d) &= [abcd] \cup [aacc] \cup [bbdd], \\
\mathcal{C}_2^{11}(a,b,c,d) \cap \mathcal{B}_2^{01}(a,b,c,d) &= [aadd] \cup [bbcc], \\
\mathcal{B}_2^{11}(a,b,c,d) \cap \mathcal{C}_2^{01}(a,b,c,d) &= [aabb] \cup [ccdd], \\
\mathcal{C}_2^{11}(a,b,c,d) \cap \mathcal{C}_2^{01}(a,b,c,d) &= [aaaa] \cup [bbbb] \cup [cccc] \cup [dddd].
\end{aligned}$$

Therefore, given a word in $\mathcal{F}_2^{\{01,11\}}$, we can decide with two quantum queries in which of these four languages the words is. The remaining possibilities for x are

$$\begin{aligned}
\mathcal{C}_2^{10}(a,b,c,d) &= [aaaa] \cup [bbbb] \cup [cccc] \cup [dddd] \cup [aaac] \\
&\quad \cup [aacc] \cup [accc] \cup [bbbd] \cup [bbdd] \cup [bddd], \\
\mathcal{B}_2^{10}(a,b,c,d) &= [aabb] \cup [aadd] \cup [bbcc] \cup [ccdd] \cup [aabd] \\
&\quad \cup [bccd] \cup [abbc] \cup [acdd] \cup [abcd], \\
\mathcal{F}_2^{\{10,11\}} &= [aaaa] \cup [bbbb] \cup [cccc] \cup [dddd] \cup [aabb] \cup [aacc] \\
&\quad \cup [aadd] \cup [bbcc] \cup [bbdd] \cup [ccdd] \cup [abcd].
\end{aligned}$$

We then have

$$\mathcal{F}_2^{\{01,11\}} = \mathcal{F}_2^{\{10,11\}}.$$

It can be checked that this is also equal to $\mathcal{F}_2^{\{01,10\}}$. However, the three possibilities $X = \{01, 11\}, \{10, 11\}$ and $\{01, 10\}$ all distinguish between different languages, since we have

$$\begin{aligned}
\mathcal{B}_2^{11}(a,b,c,d) \cap \mathcal{B}_2^{10}(a,b,c,d) &= [abcd] \cup [aabb] \cup [ccdd], \\
\mathcal{C}_2^{11}(a,b,c,d) \cap \mathcal{B}_2^{10}(a,b,c,d) &= [aadd] \cup [bbcc], \\
\mathcal{B}_2^{11}(a,b,c,d) \cap \mathcal{C}_2^{10}(a,b,c,d) &= [aacc] \cup [bbdd], \\
\mathcal{C}_2^{11}(a,b,c,d) \cap \mathcal{C}_2^{10}(a,b,c,d) &= [aaaa] \cup [bbbb] \cup [cccc] \cup [dddd], \\
\mathcal{B}_2^{01}(a,b,c,d) \cap \mathcal{B}_2^{10}(a,b,c,d) &= [abcd] \cup [aadd] \cup [bbcc], \\
\mathcal{C}_2^{01}(a,b,c,d) \cap \mathcal{B}_2^{10}(a,b,c,d) &= [aabb] \cup [ccdd], \\
\mathcal{B}_2^{01}(a,b,c,d) \cap \mathcal{C}_2^{10}(a,b,c,d) &= [aacc] \cup [bbdd], \\
\mathcal{C}_2^{01}(a,b,c,d) \cap \mathcal{C}_2^{10}(a,b,c,d) &= [aaaa] \cup [bbbb] \cup [cccc] \cup [dddd].
\end{aligned}$$

This provides an improvement over the classical deterministic setting, where we need three queries to distinguish any of these sets of four languages.

We have then seen that

$$\mathcal{F}_2^{\{01,11\}} = \mathcal{F}_2^{\{10,11\}} = \mathcal{F}_2^{\{01,10\}} = \mathcal{F}_2^{\{01,10,11\}}.$$

It may be interesting to consider larger alphabets:

$$\{a, b, c, d, e, f, g, h\} \to \{000, 001, 010, 011, 100, 101, 110, 111\}.$$

Here, it still possible to define a parity, based on the even number of 1s, like p^{11}. This is equivalent to determine if a word w is in the subspace {000,011,101,110}, also denoted p^{adfg}. In this case, the set of parity constant and parity balanced words can be obtained using the auxiliary input $|111\rangle$ in the circuit described before:

$$
\begin{aligned}
& U_f\left(|x\rangle \otimes \left(\frac{|0\rangle - |1\rangle}{\sqrt{3}}\right)^{\otimes 3}\right) \\
&= |x\rangle \otimes \frac{1}{2}(|000 \oplus f(x)\rangle - |001 \oplus f(x)\rangle - |010 \oplus f(x)\rangle + |011 \oplus f(x)\rangle \\
&- |100 \oplus f(x)\rangle + |101 \oplus f(x)\rangle + |110 \oplus f(x)\rangle - |111 \oplus f(x)\rangle) \\
&= (-1)^{p(f(x))}|x\rangle \otimes \left(\frac{|1\rangle - |0\rangle}{\sqrt{3}}\right)^{\otimes 3}.
\end{aligned}
$$

As result of this input, U_f gives (*i.e*, it is possible to recognize the membership of a letter from the plus sign in front of term) the following set

$$
\begin{aligned}
\mathcal{C}_1^{adfg}(a,b,c,d,e,f,g) &= [aa] \cup [bb] \cup [cc] \cup [dd] \cup [ee] \cup [ff] \cup [gg] \cup [hh] \cup [ad] \cup [af] \\
&\cup [ag] \cup [df] \cup [dg] \cup [fg] \cup [bc] \cup [be] \cup [bh] \cup [ce] \cup [ch] \cup [eh].
\end{aligned}
$$

Similarly, the set of parity balanced words is

$$
\begin{aligned}
\mathcal{B}_1^{adfg}(a,b,c,d,e,f,g) = [ab] &\cup [ac] \cup [bd] \cup [cd] \cup [ae] \cup [ah] \cup [de] \cup [dh] \\
&\cup [bf] \cup [bg] \cup [cf] \cup [cg] \cup [fe] \cup [fh] \cup [ge] \cup [gh].
\end{aligned}
$$

Other parities can be defined considering different set of vectors. For our purposes it is sufficient to define a set composed by the elements $p^{abcd} = \{000, 001, 010, 011\}$. This plays the same role as p^{01}. In this case, the set of parity constant word can be obtained by using $|100\rangle$ as auxiliary input. The circuit has the following output:

$$
\begin{aligned}
& U_f\left(|x\rangle \otimes \left(\frac{|0\rangle - |1\rangle}{\sqrt{3}}\right)\left(\frac{|0\rangle + |1\rangle}{\sqrt{3}}\right)^{\otimes 2}\right) \\
&= |x\rangle \otimes \frac{1}{2}(|000 \oplus f(x)\rangle + |001 \oplus f(x)\rangle + |010\rangle \oplus f(x)\rangle + |011 \oplus f(x)\rangle \\
&- |100 \oplus f(x)\rangle - |101 \oplus f(x)\rangle - |110 \oplus f(x)\rangle - |111 \oplus f(x)\rangle) \\
&= (-1)^{p(f(x))}|x\rangle \otimes \left(\frac{|1\rangle - |0\rangle}{\sqrt{3}}\right)\left(\frac{|1\rangle + |0\rangle}{\sqrt{3}}\right)^{\otimes 2}.
\end{aligned}
$$

As discussed before this procedure gives the following sets:

$$
\begin{aligned}
\mathcal{C}_1^{abcd}(a,b,c,d,e,f,g) &= [aa] \cup [bb] \cup [cc] \cup [dd] \cup [ee] \cup [ff] \cup [gg] \cup [hh] \cup [ab] \cup [ac] \\
&\cup [ad] \cup [bc] \cup [bd] \cup [cd] \cup [ef] \cup [eg] \cup [eh] \cup [fg] \cup [fh] \cup [gh].
\end{aligned}
$$

The set of parity balanced words is

$$
\begin{aligned}
\mathcal{B}_1^{abcd}(a,b,c,d,e,f,g) = [ae] &\cup [af] \cup [ag] \cup [ah] \cup [be] \cup [bf] \cup [bg] \cup [bh] \\
&\cup [ce] \cup [cf] \cup [cg] \cup [ch] \cup [de] \cup [df] \cup [dg] \cup [dh].
\end{aligned}
$$

For reasons that will be clear later, it is important to define also the parity, based on the subspace $p^{adeh} = \{000, 011, 101, 111\}$, for which the set of parity constant words is obtained by setting as auxiliary input the state $|011\rangle$:

$$
\begin{aligned}
& U_f \left(|x\rangle \otimes \left(\frac{|0\rangle + |1\rangle}{\sqrt{3}} \right) \left(\frac{|0\rangle - |1\rangle}{\sqrt{3}} \right)^{\otimes 2} \right) \\
&= |x\rangle \otimes \frac{1}{2}(|000 \oplus f(x)\rangle - |001 \oplus f(x)\rangle - |010\rangle \oplus f(x)\rangle + |011 \oplus f(x)\rangle + \\
&- |100 \oplus f(x)\rangle + |101 \oplus f(x)\rangle - |110 \oplus f(x)\rangle + |111 \oplus f(x)\rangle) \\
&= (-1)^{p(f(x))}|x\rangle \otimes \left(\frac{|1\rangle - |0\rangle}{\sqrt{3}} \right)^{\otimes 2} \left(\frac{|1\rangle + |0\rangle}{\sqrt{3}} \right).
\end{aligned}
$$

The sets produced are represented by

$$
\begin{aligned}
\mathcal{C}_1^{adeh}(a,b,c,d,e,f,g) \;=\; & [aa] \cup [bb] \cup [cc] \cup [dd] \cup [ee] \cup [ff] \cup [gg] \cup [hh] \cup [ad] \cup [ae] \\
& \cup [ah] \cup [de] \cup [dh] \cup [eh] \cup [bc] \cup [bf] \cup [bg] \cup [cf] \cup [cg] \cup [fg];
\end{aligned}
$$

for the balanced case, we have

$$
\begin{aligned}
\mathcal{B}_1^{adeh}(a,b,c,d,e,f,g) = & [ab] \cup [ac] \cup [af] \cup [ag] \cup [db] \cup [dc] \cup [df] \cup [dg] \\
& \cup [eb] \cup [ec] \cup [ef] \cup [eg] \cup [hb] \cup [hc] \cup [hf] \cup [hg].
\end{aligned}
$$

It is indeed possible to generalize the circuit for an arbitrary length binary function co-domain. In particular, the length of the output binary string will be determined by log_2 of the cardinality of the alphabet considered (for example, two bits for a 4-elements alphabet). Moreover, to each parity function subspace corresponds a unique input to be fed into the circuit shown before. The Hadamard gate transforms each bit of the input binary string into the state $|+\rangle$ or $|-\rangle$ depending on the value of the bit. For the generic input $|0 \ldots 1\rangle$, we have

$$
U_f \left(|x\rangle \otimes \left(\frac{|0\rangle + |1\rangle}{\sqrt{n}} \right) \ldots \left(\frac{|0\rangle - |1\rangle}{\sqrt{n}} \right) \right) = (-1)^{p(f(x))}|x\rangle \otimes \left(\frac{|1\rangle + |0\rangle}{\sqrt{n}} \right)^{\otimes n} \left(\frac{|1\rangle + |0\rangle}{\sqrt{n}} \right).
$$

Now we are going to analyze longer words. For example, if $n = 2$, for p^{adfg}, the set of parity balanced words is

$$
\begin{aligned}
\mathcal{C}_2^{adfg}(a,b,c,d,e,f,g) = & [aaaa] \cup [bbbb] \cup [cccc] \cup [dddd] \cup [eeee] \\
& \cup [ffff] \cup [gggg] \cup [hhhh] \cup [aaad] \cup [aadd] \\
& \cup [addd] \cup [aaaf] \cup [aaff] \cup [afff] \cup [aaag] \\
& \cup [aggg] \cup [dddf] \cup [ddff] \cup [dfff] \cup [ffgg] \\
& \cup [ffgg] \cup [fggg] \cup [bbbc] \cup [bbcc] \cup [bccc] \\
& \cup [bbbe] \cup [bbee] \cup [beee] \cup [ccce] \cup [ccee] \\
& \cup [ceee] \cup [bbbh] \cup [bbhh] \cup [bhhh] \cup [ccch] \\
& \cup [cchh] \cup [chhh] \cup [eeeh] \cup [eehh] \cup [ehhh];
\end{aligned}
$$

while the set of parity balanced words is

$$\begin{aligned}\mathcal{B}_2^{adfg}(a,b,c,d,e,f,g) = {} & [aabb] \cup [aacc] \cup [aaee] \cup [aahh] \cup [ddbb] \\ & \cup [ddcc] \cup [ddee] \cup [ddhh] \cup [ffbb] \cup [ffcc] \\ & \cup [ffee] \cup [ffhh] \cup [ggbb] \cup [ggcc] \cup [ggee] \\ & \cup [gghh] \cup [adbc] \cup [afce] \cup [agbc] \cup [agbe] \\ & \cup [agce] \cup [adce] \cup [adbe] \cup [adhe] \cup [agch] \\ & \cup [afce] \cup [afch] \cup [adbe] \cup [adbh] \cup [afbc] \\ & \cup [afbh] \cup [agbh] \cup [ageh] \cup [afeh] \cup [afbe].\end{aligned}$$

For p^{abcd}, the set of parity balanced words is

$$\begin{aligned}\mathcal{C}_2^{abcd}(a,b,c,d,e,f,g) = {} & [aaaa] \cup [bbbb] \cup [cccc] \cup [dddd] \cup [eeee] \\ & \cup [ffff] \cup [gggg] \cup [hhhh] \cup [aaab] \cup [aabb] \\ & \cup [abbb] \cup [aaac] \cup [aacc] \cup [accc] \cup [aaad] \\ & \cup [aadd] \cup [addd] \cup [bbbc] \cup [bbcc] \cup [bccc] \\ & \cup [bbbd] \cup [bbdd] \cup [bddd] \cup [cccd] \cup [ccdd] \\ & \cup [cddd] \cup [eeef] \cup [eeff] \cup [efff] \cup [eeeg] \\ & \cup [eegg] \cup [eggg] \cup [eeeh] \cup [eehh] \cup [ehhh] \\ & \cup [fffg] \cup [ffgg] \cup [ffgh] \cup [ffhh] \cup [fhhh] \\ & \cup [gggh] \cup [gghh] \cup [ghhh];\end{aligned}$$

the set of parity balanced words is

$$\begin{aligned}\mathcal{B}_2^{abcd}(a,b,c,d,e,f,g) = {} & [aaee] \cup [aaff] \cup [aagg] \cup [aahh] \cup [bbee] \\ & \cup [bbff] \cup [bbgg] \cup [bbhh] \cup [ccee] \cup [ffcc] \\ & \cup [ccgg] \cup [cchh] \cup [ddee] \cup [ddff] \cup [ddgg] \\ & \cup [ddhh] \cup [abef] \cup [abeg] \cup [abeh] \cup [acef] \\ & \cup [aceg] \cup [aceh] \cup [adef] \cup [adeg] \cup [adeh] \\ & \cup [abfg] \cup [abfh] \cup [acfg] \cup [acfh] \cup [adfg] \\ & \cup [adfh] \cup [agbh] \cup [agch] \cup [adgh].\end{aligned}$$

The same reasoning carried on for a four-letters alphabet can be applied to form the set of words

$$\mathcal{F}_n^x(\mathcal{A}) = \mathcal{C}_n^x(\mathcal{A}) \cup \mathcal{B}_n^x(\mathcal{A})$$

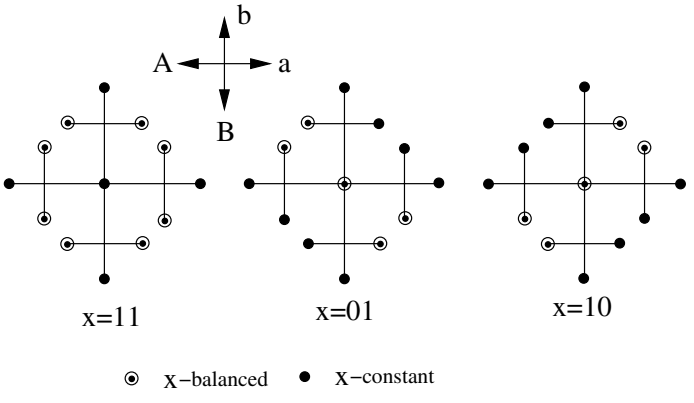
and the relative intersections. A potential generalization comes from error correcting codes. This could be based on introducing an encoding in which the letters of the alphabet are associated to the codewords of a subspace quantum error correcting code. A form of parity could be defined by considering the remaining subspaces.

2 Application to the Word Problem in Groups

Let $\{a, b, c = B, d = A\}$ be a paired alphabet, where A represents a^{-1} and B represents b^{-1}. We first consider words of length 2. Parity constant words are "character constant", *i.e.* consist of only one letter, whether it be lower or upper case. Parity balanced words are "character balanced". The words corresponding to the parity constant case are aa,aA,bb,bB,Bb,BB,Aa,AA. Those corresponding to the parity balanced case are ab,aB,ba,bA,Ba,BA,Ab,AB. The words w in the first list all satisfy $w \in \langle a \rangle \cup \langle b \rangle$ (in fact we have $w \in \langle a^2 \rangle \cup \langle b^2 \rangle$), whereas those w in the second list all satisfy $w \notin \langle a \rangle \cup \langle b \rangle$. Thus, for words of length 2, we can determine with a single measurement whether or not $w \in \langle a \rangle \cup \langle b \rangle$.

If $x = 01$ then the x-constant words are aa,ab,ba,bb,BA,BB,AA,AB and the x-balanced words are aB,aA,bB,bA,Aa,Ab,Ba,Bb. So, 01-constant and 01-balanced may be thought of as "case constant" and "case balanced" where the case can be upper or lower. For example, a commutator word (reduced or not) is always case balanced. "Case constant" and "case balanced" are properties of $\overline{w}$, rather than w. This is not the case for "parity constant" and "parity balanced".

If $x = 10$ then the x-constant words are aa,aB,bb,bA,Ba,BB,Ab,AA and the x-balanced words are ab,aA,ba,bB,Bb,BA,AB,Aa. This does not seem to have any nice interpretation, while the 10-balanced corresponds to the cyclic subgroup generated by ab and the 11-constant set solve a problem of union of subgroup membership for $\langle a \rangle \cup \langle b \rangle$. The elements represented by these words are depicted on the following Cayley graph portions:



Note that w is 11-constant but not 01-constant and w is 11-constant and not 10-constant then $w = F_2^{11}$. This gives a method of solving the word problem for words of length 2 using two quantum queries.

Definition 1. *We say that words in $\mathcal{F}_1^{11} \cap \mathcal{F}_1^{01}$ are $\mathcal{QWP}$-*feasible *(quantum word problem feasible). We write the set of such words as $\mathcal{QWP}_1$.*

For $n = 1$, if we are promised that w is $\mathcal{QWP}$-feasible then the quantum query complexity of the property "is w trivial?" seems to be 2. But this is not a

reduction in complexity from the classical case. However, there is hope that an analogous method might be an improvement in quantum query complexity for longer words. We have the following

Proposition 1. *For all n, if we are promised that the word w of length 2^n is 11-feasible then the quantum query complexity of the property "Does w represent an element of $\langle a\rangle \cup \langle b\rangle$?" is 1.*

This is directly analogous to the Deutsch-Jozsa algorithm, and the proof is the same. It is unclear how to extend the definition of $\mathcal{QWP}$ beyond two letters. Here are examples of two groups where we require different promises:

Proposition 2. *Consider the free abelian group $G = \langle a, b \mid ab = ba\rangle$. Let w be a four-letter word in $\mathcal{A}$ which is in $\mathcal{F}_1^{11} \cap \mathcal{F}_2^{01} \cap \mathcal{F}_2^{10}$. Then the quantum query complexity of the question "Does w represent the trivial element of G?" is at most 3.*

Proof. The first query asks whether $w \in \mathcal{C}_2^{01}$ or $w \in \mathcal{B}_2^{01}$. If the former is true then w is not trivial so stop. If $w \in \mathcal{B}_2^{01}$ then proceed to the second query, which is whether $w \in \mathcal{C}_2^{11}$ or $w \in \mathcal{B}_2^{11}$. If the former is true then w is trivial so stop. Otherwise we know that $w \in \mathcal{B}_2^{11} \cap B_2^{01}$ and we may proceed to the third query. There are two possibilities. The first possibility is that we have a word with two As and two bs or a word with two as and two Bs. That is, w is a cyclic rotation of $(AAbb)^{\pm 1}$. The second possibility is that we have one each of A, b, a and B. In the first case, w is nontrivial and in $\mathcal{C}_2^{10}$; in the second case, w is trivial and in $\mathcal{B}_2^{01}$. So our third query is whether $w \in \mathcal{C}_2^{10}$ or $w \in \mathcal{B}_2^{10}$; this solves the word problem provided w is as promised. ∎

It is indeed possible to generalize this theorem to the 8-letters alphabet introduced earlier, by considering the four-paired alphabet $\{a, b, c, d, e = D, f = C, g = B, h = A\}$, where the upper-case A, B, C, D letters represent respectively $a^{-1}, b^{-1}, c^{-1}, d^{-1}$. In particular, we have the following statement:

Proposition 3. *Consider the free group $G = \langle a, b, c, d \mid abcd = dcba\rangle$. Let w be a 8-letter word in $\mathcal{A}$ which is in $\mathcal{F}_3^{adfg} \cap \mathcal{F}_3^{abcd} \cap \mathcal{F}_3^{adeh}$. Then the quantum query complexity of the question "Does w represent the trivial element of G?" is at most 3.*

Proof. The first query asks whether $w \in \mathcal{C}_3^{abcd}$ or $w \in \mathcal{B}_3^{abcd}$. If the former is true then w is not trivial so stop. If $w \in \mathcal{B}_3^{abcd}$ then proceed to the second query, which is whether $w \in \mathcal{C}_3^{adfg}$ or $w \in \mathcal{B}_3^{adfg}$. If the former is true then w is trivial so stop. Otherwise we know that $w \in \mathcal{B}_3^{adfg} \cap B_3^{abcd}$ and we may proceed to the third query. There are two possibilities. The first possibility is that we have a word with two As two Ds and two as and two ds or a word with two Cs two Bs, two cs and two bs. That is, w is a cyclic rotation of $(AADDaadd)^{\pm 1}$ or $(BBCCbbcc)^{\pm 1}$. The second possibility is that we have one each of A, b, a B, C, d,c, and D . In the first case, w is nontrivial and in $\mathcal{C}_3^{adeh}$; in the second, w is trivial and in $\mathcal{B}_3^{abcd}$. Our third query is whether $w \in \mathcal{C}_3^{adeh}$ or $w \in \mathcal{B}_3^{abcd}$; this solves the word problem provided w is as promised. ∎

Looking at the first two queries it seems possible to generalize this result for every paired alphabet of dimension 2^{n-1} and words of length 2^n, by defining parities based on the even number of ones, like p^{adfg}. This is always possible because of the equipartition of the binary strings with respect to the number of ones and on the subspaces formed by the first 2^{n-1} vectors labeled from zero to 2^n. The last parities required is the one used to identify words that are cyclic permutations of elements of the alphabet, for example, p^{adeh}. Moreover it's also easy to see that the setting is independent on the dimension of the alphabet as long it's possible to define a parity function that splits in two part the number of symbols and it's an efficient way of recognizing the membership of an element to a given subset. It does not seem easy to distinguish between trivial and nontrivial four-letter words in the free group of rank 2 using less than 4 quantum queries. However, the first indication that classical query complexity can be improved upon in a nonabelian finitely presented group is the following:

Proposition 4. *Consider the group presented by* $G = \langle a, b \mid a^2 = b^2 \rangle$. *Suppose we are given a word* w *of length* 4 *in* $\mathcal{A}$ *such that* $w \in \mathcal{F}_2^{11} \cap \mathcal{F}_2^{01}$. *Then the quantum query complexity of the question "Does* w *represent the trivial element of* G*?" is at most* 3.

Proof. The first two queries are as in the proof of the last proposition. So we can assume that if we do not already know whether or not w is trivial, $w \in \mathcal{B}_2^{11} \cap B_2^{01}$ and we may proceed to the third query. For this, we construct a "syllable function"

$$f : \{0,1\} \to \{aa, ab, aB, aA, ba, bb, bB, bA, Ba, Bb, BB, Ba, Aa, Ab, AB, AA\}.$$

It maps AA,BB,Aa,aA,Ab,AB,ab,aB to 0 and Bb,bB,BA,bA,Ba,ba,aa,bb to 1. Note that, since $w \in \mathcal{B}_2^{11} \cap B_2^{01}$, w is either a cyclic rotation of $(AAbb)^{\pm 1}$ or w is an anagram of $AaBb$. Words in the first case are all trivial, because $a^2 = b^2$ is a relation in G, and these words are all balanced under the syllable function. Words in the second case are nontrivial if and only if they are nontrivial commutators. Commutators are constant under the syllable function. Words in the second case which are trivial (*i.e.*, not commutators) are all balanced under the syllable function. Thus a third query of "is w syllable-balanced or syllable-constant" will complete the solution of the word problem. The following table lists all 0-syllabs and 1-syllabs:

0-syllabs	1-syllabs
AA	aa
BB	bb
Aa	Bb
aA	bB
Ab	bA
AB	BA
ab	ba
aB	Ba

■

While the group G in the last proposition is nonabelian, it can be shown to have a free abelian subgroup of rank 2 and index 4; it is an extension of $\mathbb{Z} \oplus \mathbb{Z}$ by the Klein 4-group.

Proposition 5. *Consider the group presented by* $G = \langle a, b, c, d \mid a^2b^2 = b^2a^2 \rangle$. *Suppose we are given a word* w *of length* 8 *in* $\mathcal{A}$ *such that* $w \in \mathcal{F}_3^{adfg} \cap \mathcal{F}_3^{abcd}$. *Then the quantum query complexity of the question "Does* w *represent the trivial element of* G*?" is at most* 3.

Proof. The first two queries are as in the proof of the last proposition. So we can assume that if we do not already know whether or not w is trivial, $w \in \mathcal{B}_3^{adfg} \cap B_3^{abcd}$ and we may proceed to the third query. For this, we construct an extended syllable function whose output has a cardinality of 2^{n-1}. Some of the elements are listed below:

$$\begin{aligned} f : \{0,1\} \longrightarrow \{&aaaa, bbbb, BBBB, AAAA, aaab, aabb, abbb,\\ &aaaB, aaBB, aBBB, aaaA, aaAA, aAAA, aaAA,\\ &aAAA, bbbB, bbBB, bBBB, bbbA, bbAA, bAAA,\\ &bbba, bbaa, baaa, BBBa, BBaa, Baaa, BBBb,\\ &BBbb, Bbbb, BBBA, BBAA, BAAA, AAAa, AAaa,\\ &Aaaa, AAAb, AAbb, Abbb, AAAB, AABB, ABBB, ...\} \end{aligned}$$

Examples of this map are

$$AAAA, BBBB, Abbb, AAaa, aBBB, aaBB, aaaB, abAB, ABab, ABab, AABB \ldots \text{to } 0$$

and

$$aaaa, bbbb, Bbbb, BBbb, bBBB, bAAA, BBAA, BAAA, baaa, bAAA, aabb, bbaa \ldots \text{to } 1.$$

Note that since $w \in \mathcal{B}_3^{adfg} \cap B_3^{abcd}$, w is either a cyclic rotation of $(AABBaabb)^{\pm 1}$ or w is an anagram of $AAaaBBbb$. Words in the first case are all trivial, because $a^2b^2 = b^2a^2$ is a relation in G, and these words are all balanced under the syllable function. Words in the second case are nontrivial if and only if are nontrivial sequence of letters, that is not commutator-like sequence with respect to the presentation. Words in the second case which are trivial (*i.e.*, not trivial sequence) are all balanced under the extended syllable function. Thus a third query of "is w syllable-balanced or syllable-constant" will complete the solution of the word problem. ∎

The same considerations can be made by looking at different sets of generators or relations like $G = \langle a, b, c, d \mid c^2d^2 = d^2c^2 \rangle$ and $G = \langle a, b, c, d \mid b^2c^2 = c^2b^2 \rangle$. It is important to notice that all the alternate sets of relations five groups isomorphic to the group considered in Proposition 5. To see this, it is sufficient to relabel the generators. The relation in G is in fact very general and it is possible to obtain the same result with a whole family of similar relations. This can be done by varying the parity function used for the queries, choosing the presentation accordingly. Moreover such a group is a free group of rank 2 with $G = \langle a, b, c, d \mid a^2b^2 = b^2a^2 \rangle$.

It is simple to see that since the other two generators, c and d, are not involved in the proof, it is possible to take the free product of G with any free group and get to the same conclusion. In particular it is possible to extend the free product with *any* group and see the invariance of those three quantum queries under free products.

Notice that the choice of some particular kind of relations and an higher number of generators in the setting of the problem may increase the number of queries required. The reason of this is the exponential growth in the number of permutations, in particular, in those cases where splitting the words in parity balanced and parity constant does not help. Generalize to other different sets of generators and possibly for free products, and limiting to commutator words might give interesting promises. It is also useful to remark the importance of having the correct and certain answer to each of the queries used to prove the above propositions. As we have mentioned before, although it is possible to obtain *on average* with three classical queries the same results given by the Deutsch-Jozsa algorithm (see [4]), we assume for our proofs to have three certain answers to the queries. This means that on average the solution to the problems proposed requires at least nine classical queries (*i.e.*, three classical queries for each quantum one). This makes explicit the gain in number of queries of the quantum setting with respect to the classical deterministic one.

3 Conclusions

We have extended the original Deutsch-Jozsa algorithm to functions of arbitrary length binary output with the introduction of a more general concept of parity. This setting allows to consider a mapping between a binary string and the elements of an alphabet. The mapping helps to solve some instances of the word problem, using small alphabets and free groups, in a reduced number of queries with respect to the deterministic classical case. Extensions to more general groups and presentations may give interesting promises.

Acknowledgments. The authors would like to thank Andrew Childs for useful remarks. Part of this work as been carried out while Andrea Casaccino was attending "The Seventh Canadian Summer School on Quantum Information", hosted by Perimeter Institute and the Institute for Quantum Computing, at the University of Waterloo.

References

1. Lyndon, R.C., Schupp, P.E.: Combinatorial group theory. Classics in Mathematics. Springer, Heidelberg (2001) (reprint of the 1977 edition)
2. Nielsen, M.A., Chuang, I.L.: Quantum computation and quantum information. Cambridge University Press, Cambridge (2000)
3. Deutsch, D., Jozsa, R.: Rapid Solution of Problems by Quantum Computation. Proc. R. Soc. of London A 439, 553–558 (1992)
4. Farhi, E., Goldstone, J., Gutmann, S., Sipser, M.: Limit on the speed of quantum computation on determining parity. Phys. Rev. Lett. 81, 5552–5554 (1998)

An Elementary Optical Gate for Expanding Symmetrically Shared Entanglement

Toshiyuki Tashima[1,2], Şahin Kaya Özdemir[1,2,3],
Takashi Yamamoto[1,2], Masato Koashi[1,2], and Nobuyuki Imoto[1,2]

[1] Division of Materials Physics, Department of Materials Engineering Science,
Graduate School of Engineering Science, Osaka University,
Toyonaka, Osaka 560-8531, Japan
[2] CREST Photonic Quantum Information Project,
4-1-8 Honmachi, Kawaguchi, Saitama 331-0012, Japan
[3] ERATO Nuclear Spin Electronics Project,
Sendai 980-8578, Japan

Abstract. We introduce an elementary optical gate based on post-selection strategy that enables not only to prepare polarization entangled W state, but also to grow this into a large-scale multi-photon W state. The gate is composed of a pair of 50:50 beamsplitters and a phase shifter, and it requires a two-photon ancillary state. When the input is a photon from an n-photon W state, the gate produces an $(n+2)$-photon W state for post-selected events. Moreover, we show that this gate can be used to prepare and expand GHZ states by a simple modification of the ancillary state.

1 Introduction

Entanglement is a purely quantum mechanical resource which plays an important role in various quantum information processing tasks, e.g., quantum teleportation [1], quantum key distribution (QKD) [2], quantum computation [3]. With its promising features in decoherence-free quantum information processing and distributed multi-party quantum communication and computation, multi-particle entanglement has become a topic of intense research [4, 5, 6, 7]. In the multipartite case ($N \geq 3$), it is known that there are inequivalent classes of states that cannot be transformed into each other by local operations and classical communication (LOCC) [8]. For example, for three-photon entangled states, the W state $|\,\mathrm{W}_3\rangle = (|\mathrm{HHV}\rangle + |\mathrm{HVH}\rangle + |\mathrm{VHH}\rangle)/\sqrt{3}$ and the Greenberger-Horne-Zeilinger (GHZ) state $|\,\mathrm{GHZ}_3\rangle = (|\mathrm{HHH}\rangle + |\mathrm{VVV}\rangle)/\sqrt{2}$ are inequivalent to each other [8]. Here H and V represent horizontal and vertical polarizations, respectively. Those two states can be generalized to the N-photon system as $|\,\mathrm{W}_N\rangle = |\,N-1,1\rangle/\sqrt{N}$ and $|\,\mathrm{GHZ}_N\rangle = (|\,N,0\rangle + |\,0,N\rangle)/\sqrt{2}$, where $|\,N-k,k\rangle$ is the sum of all the states with $N-k$ H-polarized photons and k V-polarized photons. For $N \geq 4$, there are many other inequivalent states in addition to the W and GHZ states, such as cluster states [9].

Y. Kawano and M. Mosca (Eds.): TQC 2008, LNCS 5106, pp. 70–82, 2008.

The total amount of entanglement contained in $|\mathrm{W}_N\rangle$ is smaller than the GHZ states and cluster states [10, 11]. Instead, it has a higher persistency of entanglement, which is $N-1$ if we quantify the persistency as the minimum number of local measurements to completely destroy entanglement. The GHZ and cluster states have the persistency of one and $N/2$, respectively [9]. It has also been shown that $|\mathrm{W}_N\rangle$ is optimal in the amount of pairwise entanglement when $N-2$ particles are discarded [11].

Cluster states have been proposed as a universal substrate for measurement based quantum computation [12]. GHZ class has not only been shown to be useful for quantum teleportation [13], quantum secret sharing [14,15] and QKD [16] but also to be the only one for reaching consensus in distributed networks when no classical post-processing is allowed [17, 18]. On the other hand, W class is proposed as a resource for QKD [19] and for the optimal universal quantum cloning machine [20] as well as shown to be the only pure state to exactly solve the problem of leader election in anonymous quantum networks [17,18]. It is thus important to experimentally prepare states of different classes not only for practical applications but also for the fundamental study of quantum information.

So far, a number of schemes have been proposed for the preparation of these inequivalent classes [21, 22, 23, 24, 25, 26, 27, 28, 29], and experimental demonstrations in NMR, ion trap and optical systems have been performed [30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41]. In the framework of ion trap systems, 6-qubit GHZ states [40] and up to 8-qubit W states [39] have been achieved. The ion trap systems have been shown to be scalable, but generation of multipartite entangled states requires the coupling of all the ions to a common vibrational mode of the trap as well as collective gate operations [39]. Moreover, with increasing number of qubits, the initialization process becomes more complicated [39]. On the other hand, in the optical domain $|\mathrm{GHZ}_3\rangle$, $|\mathrm{GHZ}_4\rangle$, $|\mathrm{W}_3\rangle$ and cluster states up to six qubits have been experimentally constructed [30, 31, 32, 33, 34, 35, 36, 37, 38]. It has also been shown that using EPR pairs as the initial seeds, quantum parity checking gates can be efficiently used to grow large scale GHZ and cluster states [13]. However, study of W states using linear optics have been lagging, and it begs for efficient elementary-gate-based approach which will enable large scale W-state networks. Current proposals for W states either suffer from low success probability or the requirement of fragile interferometers besides their non-cascadable structures.

What we want here is a scheme that converts N photons in state $|\mathrm{W}_N\rangle$ into $|\mathrm{W}_{N+1}\rangle$ by attaching an ancilla photon. It is desirable to accomplish this conversion by merely interacting the ancilla photon with only one photon among the original N photons. This task is not trivial because the marginal state of the $N-1$ untouched qubits for the initial state $|\mathrm{W}_N\rangle$ is different from the one for the final state $|\mathrm{W}_{N+1}\rangle$. It is thus impossible to achieve the conversion by a unitary operation. More importantly, the newly added qubit must form pairwise entanglement with each of the uninteracted $N-1$ qubits [see Fig. 1(a)]. This is in contrast with the GHZ states, for which the marginal states of the untouched $N-1$ qubits are the same for $|\mathrm{GHZ}_N\rangle$ and $|\mathrm{GHZ}_{N+1}\rangle$. Hence the conversion

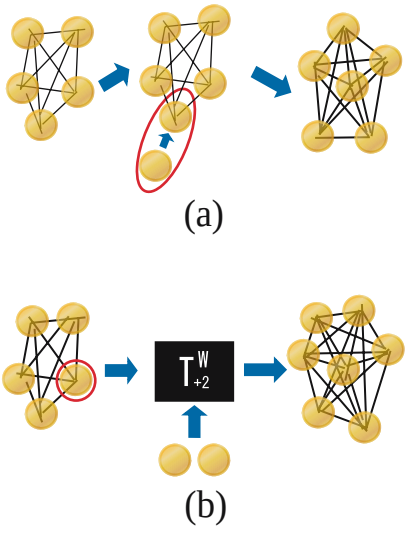


Fig. 1. (a) Local extension of W states. (b) The proposed optical gate denoted as $\mathbf{T}^{\mathbf{W}}_{\mathbf{+2}}$ converts $|\,\mathrm{W}_N\rangle$ to $|\,\mathrm{W}_{N+2}\rangle$.

is acheived unitarily, and even in linear optics, it can be done with a polarizing beamsplitter and post-selection [13].

In this study, we propose an elementary optical gate ($\mathrm{T}^{\mathrm{W}}_{+2}$ gate) for expanding polarization entangled W state. It is composed of a two-photon Fock state as the ancillary, two 50:50 beam splitters (BS), and a phase shifter (PS). The successful events are post-selected. Interestingly, the same gate can be used for the expansion $|\,\mathrm{W}_N\rangle \rightarrow |\,\mathrm{W}_{N+2}\rangle$ of any size N [see Fig. 1(b)]. If we extrapolate the definition of W states, it is seen that similar operations can be done for $N = 1, 2$: If the seed (the state to be expanded) is $|\,\mathrm{W}_1\rangle = |\,1\rangle$, the gate prepares $|\,\mathrm{W}_3\rangle$, and if the seed is an Einstein-Podolsky-Rosen (EPR) pair $|\,\mathrm{W}_2\rangle = (|\,01\rangle + |\,10\rangle)/\sqrt{2}$, the gate prepares the state $|\,\mathrm{W}_4\rangle$. This simple gate, interestingly, prepares $|\,\mathrm{W}_4\rangle$ more efficiently than any other linear optical scheme proposed for $|\,\mathrm{W}_4\rangle$ so far. If one starts with $|\,\mathrm{W}_1\rangle$ and applies the gate successively $N-1$ times, W states with odd number of photons, $|\,\mathrm{W}_{2N+1}\rangle$, can be prepared. In the same way, states with even number of photons $|\,\mathrm{W}_{2N}\rangle$ can be prepared starting with $|\,\mathrm{W}_2\rangle$. Thus, in principle it is possible to prepare any $|\,\mathrm{W}_N\rangle$ using this gate.

This paper is organized as follows: In Sec. 2, we give the working principle of the proposed gate ($\mathrm{T}^{\mathrm{W}}_{+2}$ gate). Sections 3 and 4 include, respectively, the use of this gate in the preparation and expansion of polarization entangled W states, and the feasibility analysis for $|\,\mathrm{W}_4\rangle$ preparation. In Sec. 5, we show that by changing the state of the ancilla photons, this gate can be used for the expansion of GHZ states, and finally Sec. 6 will contain a summary of this work.

2 Working Principle of the Proposed Gate

In Fig. 2, we show the schematic of the proposed gate. The gate receives one photon from mode 1 as the input, and mixes it by a 50:50 beamsplitter (BS1) with two ancilla photons in the state $|\,2_{\rm H}\rangle_2$ where H stands for the horizontal polarization and the subscript number signifies the spatial mode. One of the output modes of BS1 is further divided into two modes by another 50:50 beamsplitter (BS2). The gate is successful when one photon is found in each of the output modes 4, 5, and 6. The phase shifter (PS) placed in mode 4 is a half-wave plate which introduces a π-phase shift between H and V polarizations to keep the final W state in the standard symmetric form.

First we analyze the operation of this gate for H-polarized ($|\,1_{\rm H}\rangle_1$) or V-polarized ($|\,1_{\rm V}\rangle_1$) seed states in mode 1. The action of the polarization-independent BS1 on H (V) polarization is represented by the transformation $\hat{a}^\dagger_{1{\rm H(V)}}=(\hat{a}^\dagger_{3{\rm H(V)}}-\hat{a}^\dagger_{4{\rm H(V)}})/\sqrt{2}$ and $\hat{a}^\dagger_{2{\rm H(V)}}=(\hat{a}^\dagger_{3{\rm H(V)}}+\hat{a}^\dagger_{4{\rm H(V)}})/\sqrt{2}$, where $\hat{a}^\dagger_{j{\rm H(V)}}$ is the photon creation operator for mode j in H(V) polarization. Using these relations we find that with the action of BS1, the initial states $|\,1_{\rm H(V)}\rangle_1\otimes|\,2_{\rm H}\rangle_2=2^{-1/2}\hat{a}^\dagger_{1{\rm H(V)}}(\hat{a}^\dagger_{2{\rm H}})^2|\,0\rangle$ evolve as

$$
\begin{aligned}
|\,1_{\rm H}\rangle_1|\,2_{\rm H}\rangle_2 \rightarrow{}& \frac{\sqrt{3}}{2\sqrt{2}}|\,3_{\rm H}\rangle_3|\,0\rangle_4+\underline{\frac{1}{2\sqrt{2}}|\,2_{\rm H}\rangle_3|\,1_{\rm H}\rangle_4}-\frac{1}{2\sqrt{2}}|\,1_{\rm H}\rangle_3|\,2_{\rm H}\rangle_4\\
&-\frac{\sqrt{3}}{2\sqrt{2}}|\,0\rangle_3|\,3_{\rm H}\rangle_4,\\
|\,1_{\rm V}\rangle_1|\,2_{\rm H}\rangle_2 \rightarrow{}& \frac{1}{2\sqrt{2}}|\,1_{\rm V}2_{\rm H}\rangle_3|\,0\rangle_4+\underline{\frac{1}{2}|\,1_{\rm H}1_{\rm V}\rangle_3|\,1_{\rm H}\rangle_4}+\frac{1}{2\sqrt{2}}|\,1_{\rm V}\rangle_3|\,2_{\rm H}\rangle_4\\
&-\underline{\frac{1}{2\sqrt{2}}|\,2_{\rm H}\rangle_3|\,1_{\rm V}\rangle_4}-\frac{1}{2}|\,1_{\rm H}\rangle_3|\,1_{\rm H}1_{\rm V}\rangle_4-\frac{1}{2\sqrt{2}}|\,0\rangle_3|\,1_{\rm V}2_{\rm H}\rangle_4.
\end{aligned}
\tag{1}
$$

The underlined terms, in which there are two photons in mode 3 and one photon in mode 4, are the only ones leading to the successful gate operation. Hence we are interested only in the underlined terms. The states $|\,2_{\rm H}\rangle_3$ and $|\,1_{\rm H}1_{\rm V}\rangle_3$ appearing in the underlined terms are transformed at BS2 as

$$
\begin{aligned}
|\,2_{\rm H}\rangle_3 \rightarrow{}& \frac{1}{2}|\,2_{\rm H}\rangle_5|\,0\rangle_6+\underline{\frac{1}{\sqrt{2}}|\,1_{\rm H}\rangle_5|\,1_{\rm H}\rangle_6}+\frac{1}{2}|\,0\rangle_5|\,2_{\rm H}\rangle_6,\\
|\,1_{\rm H}1_{\rm V}\rangle_3 \rightarrow{}& \frac{1}{2}|\,1_{\rm H}1_{\rm V}\rangle_5|\,0\rangle_6+\underline{\frac{1}{2}|\,1_{\rm H}\rangle_5|\,1_{\rm V}\rangle_6}+\underline{\frac{1}{2}|\,1_{\rm V}\rangle_5|\,1_{\rm H}\rangle_6}+\frac{1}{2}|\,0\rangle_5|\,1_{\rm H}1_{\rm V}\rangle_6.
\end{aligned}
\tag{2}
$$

Since successful gate operation requires that there is one photon in each of the modes 4, 5 and 6, it is apparent that only the underlined terms in Eq. (2) has

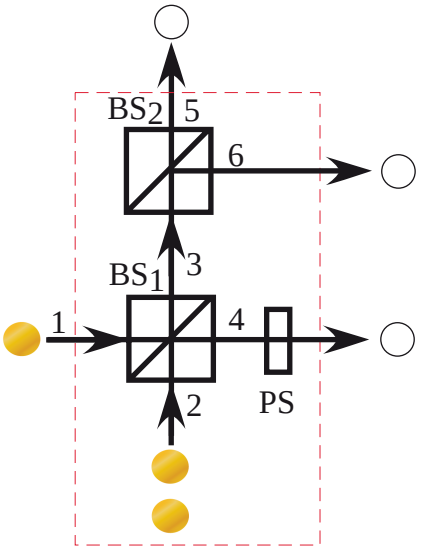


Fig. 2. The schematic diagram of the proposed elementary optical gate

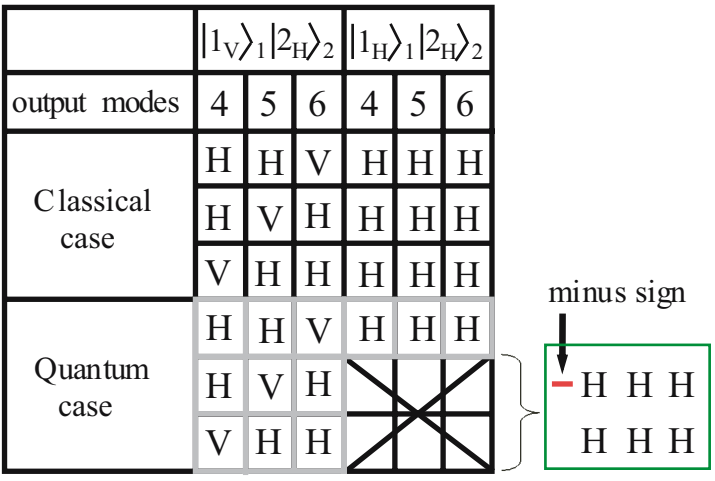
contribution. If we postselect these terms with a coincidence detection scheme, the operation of the gate will be given by the following state transformations:

$$|1_{\mathrm{H}}\rangle_1|2_{\mathrm{H}}\rangle_2 \rightarrow \frac{1}{4}|1_{\mathrm{H}}\rangle_4|1_{\mathrm{H}}\rangle_5|1_{\mathrm{H}}\rangle_6, \tag{3}$$

$$|1_{\mathrm{V}}\rangle_1|2_{\mathrm{H}}\rangle_2 \rightarrow \frac{1}{4}|1_{\mathrm{H}}\rangle_4|1_{\mathrm{H}}\rangle_5|1_{\mathrm{V}}\rangle_6 + \frac{1}{4}|1_{\mathrm{H}}\rangle_4|1_{\mathrm{V}}\rangle_5|1_{\mathrm{H}}\rangle_6 + \frac{1}{4}|1_{\mathrm{V}}\rangle_4|1_{\mathrm{H}}\rangle_5|1_{\mathrm{H}}\rangle_6, \tag{4}$$

where we have included the effect of the PS in mode 4. All the four terms appearing in Eq. (3) and Eq. (4) have the same amplitude implying that the success probability is 1/16 for the $|1_{\mathrm{H}}\rangle_1$ input and 3/16 for the $|1_{\mathrm{V}}\rangle_1$ input. If the seed photon is a part of a polarization entangled system, we will have a coherent superposition of the above two cases. It is interesting to see here that the gate performs a symmetrization among the input and the ancillary states. The insight into the above equations can be gained as follows [see Table 1]: When the seed is a V-polarized photon, we have a classical situation where two H-polarized photons are distributed among three output modes in three different ways and the remaining one port is occupied by the V-polarized photon from the seed. These three cases are $|1_{\mathrm{H}}\rangle_4|1_{\mathrm{H}}\rangle_5|1_{\mathrm{V}}\rangle_6$, $|1_{\mathrm{H}}\rangle_4|1_{\mathrm{V}}\rangle_5|1_{\mathrm{H}}\rangle_6$, $-|1_{\mathrm{V}}\rangle_4|1_{\mathrm{H}}\rangle_5|1_{\mathrm{H}}\rangle_6$. Note that the last term with the minus sign corresponds to the situation where there are two H-polarized photons at the input of BS2. On the other hand, when the seed is an H-polarized photon, we have three indistinguishable particles and quantum effects come into play. Contrary to the expectation with distinguishable particles, we end up with only one term because the three possible cases to

Table 1. Input-output relation for the intuitive understanding of gate operations



distribute the particles among the output modes correspond exactly to the same state $|1_H\rangle_4|1_H\rangle_5|1_H\rangle_6$ with one having a minus sign. This minus sign leads to destructive interference resulting in only one term, $|1_H\rangle_4|1_H\rangle_5|1_H\rangle_6$. This is the important feature of the gate which will be exploited in expanding the symmetrically shared entanglement in W state.

3 Seeding and Expanding Polarization Entangled W States

We observe that the expression on the right hand side of Eq. (4) corresponds to $|W_3\rangle$ which implies that the gate prepares polarization entangled $|W_3\rangle$ state with a success probability of 3/16 if the seed $|W_1\rangle$ is a V-polarized single photon. Equations (3) and (4) also tell that if the input photon in mode 1 has formed an EPR pair $|W_2\rangle = (|1_H\rangle_0|1_V\rangle_1 + |1_V\rangle_0|1_H\rangle_1)/\sqrt{2}$ with another photon in mode 0, the output state for the post-selected events are given by

$$|W_2\rangle \rightarrow \frac{1}{4\sqrt{2}}[\,|1_H\rangle_0|1_H\rangle_4|1_H\rangle_5|1_V\rangle_6 + |1_H\rangle_0|1_H\rangle_4|1_V\rangle_5|1_H\rangle_6 + |1_H\rangle_0|1_V\rangle_4|1_H\rangle_5|1_H\rangle_6 + |1_V\rangle_0|1_H\rangle_4|1_H\rangle_5|1_H\rangle_6\,], \quad (5)$$

which means that the $|W_4\rangle$ state is produced with probability 1/8.

The success probabilities 3/16 and 1/8 for the preparation of $|W_3\rangle$ and $|W_4\rangle$, respectively, are significant improvements over the other linear optics-based schemes proposed for these states. For instance, the most efficient schemes so far are those in [24] and [26], respectively for $|W_4\rangle$ and $|W_3\rangle$ with the corresponding success probabilities of 2/27 and 1/9 which is lower than those of our proposal.

If the input photon is from a general W-state $|\,\mathrm{W}_N\rangle$, the application of the gate will result in the transformation $|\,\mathrm{W}_N\rangle \rightarrow [|\,N-2,1\rangle \otimes |\,3,0\rangle + |\,N-1,0\rangle \otimes |\,2,1\rangle]/4\sqrt{N} = |\,N+1,1\rangle/4\sqrt{N}$, implying that $|\,\mathrm{W}_{N+2}\rangle$ is prepared with a success probability of $(N+2)/(16N)$. When N becomes large, the success probability will approach to the constant $1/16$.

An interesting feature of this gate is that it can be cascaded to prepare any desired size of W state. Starting with an input state of $|\,1_{\mathrm{V}}\rangle_1$ to the first gate in the cascaded series of k proposed gates, a $2k+1$-photon polarization entangled W state, $|\,\mathrm{W}\rangle_{2\mathrm{k}+1}$, can be prepared provided that coincidence detection is observed at $2k+1$ output spatial modes. The success probability of such an event scales as $p_{\mathrm{success}} = (2k+1)2^{-4k}$. Similarly, starting with a photon from a EPR pair and cascading k gates, one can prepare $2(k+1)$-photon polarization entangled W state, $|\,\mathrm{W}\rangle_{2(\mathrm{k}+1)}$ with a success probability of $p_{\mathrm{success}} = (k+1)2^{-4k}$.

Besides our current proposal, the scheme based on $N \times N$ multiport interferometers [26, 27] is so far the only proposal encompassing generation of $|\,\mathrm{W}_N\rangle$ with arbitrary N. This scheme requires a different multiport device for each N. In addition, numerical calculation up to $N = 7$ shows that our proposal has better efficiency, e.g., for $N = 5$ our proposal succeeds with a probability 12 times higher than that of the multiport interferometer. Note also that $N \times N$ interferometer cannot generate the $|\,\mathrm{W}_6\rangle$ state because of the zero probability of coincidence detection due to destructive interference.

4 Feasibility Analysis for Preparing W_4 State

So far, several linear optical schemes for preparing the state $|\,\mathrm{W}_4\rangle$ have been proposed [24, 25, 26], but no experiments have been done yet. It is thus important to consider the feasibility of our scheme with practical photon sources, namely, parametric down-conversion (PDC) and/or weak coherent pulses (WCP) obtained by attenuating laser pulses. We give a schematic configuration of a possible experimental scheme in Fig. 3.

In this scheme, the light from a mode-locked Ti:sapphire laser (wavelength 790nm; pulse width 80fs; repetition rate 82MHz) is frequency-doubled to a wavelength of 395nm with a second harmonic generator (SHG) to prepare ultraviolet (UV) light. Then the UV light is divided into two parts one of which is used to prepare the EPR photon pair and the other to prepare the H-polarized two-photon ancillary state using spontaneous parametric down conversion (SPDC). For EPR photon pair generation the polarization of the UV pulse is set to $\pi/4$, and it is used to pump BBO_1, which is formed by stacking together two Type I phase matched 2mm thick β-barium borate (BBO) crystals with their optical axes orthogonal to each other. For the ancilla state preparation, the UV pulse is set to vertical polarization and then it is used to pump a Type I BBO_2 to prepare two photons in H-polarization collinearly. Then one photon of the EPR photon pair is sent to the proposed gate where it mixes with the ancilla photons. The correct events are post-selected by a four-fold coincidence detection at silicon avalanche photodiodes, $\mathrm{D}_{j=1,2,3,4}$, (EG&G single photon counting module-SPCM with quantum efficiency $\eta_d \sim 0.55$ and dark count rate 50 count/s).

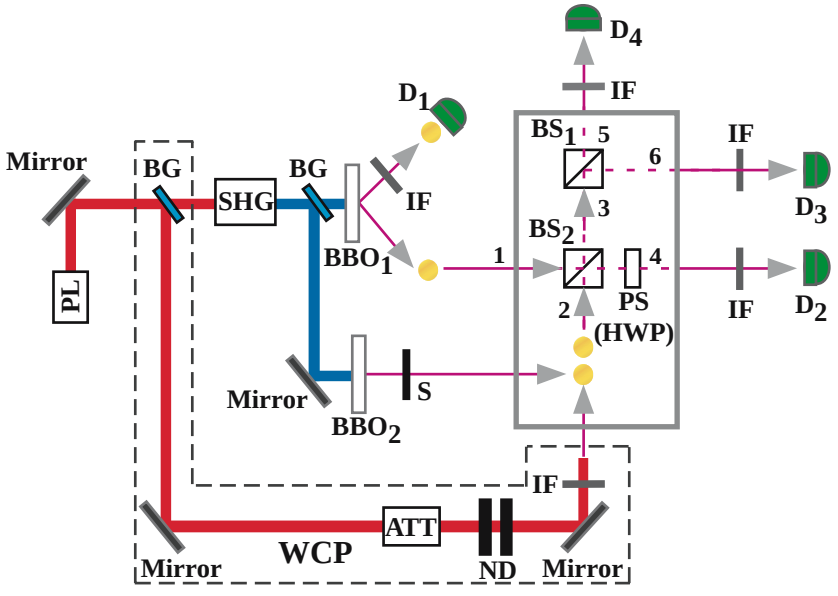


Fig. 3. Schematic configuration of experimental setup for realizing the proposed gate. PL, pulsed laser; SHG, second harmonic generator; BG, brewster window; $BBO_{1,2}$, Type I phase matched 2mm thick β-barium borate crystal for spontaneous parametric down conversion (SPDC); BS1 and BS2, 50:50 symmetric beamsplitters; PS, phase shifter implemented using half wave plate (HWP); IF, interference filter with central wavelength $\lambda = 790$nm and full-width at half-maximum bandwidth $\Delta\lambda = 3$nm; ND, Neutral density filter; ATT, optical attenuator; D_j, photodetectors. The H-polarized two-photon ancilla state can be prepared by either using SPDC or weak coherent pulse (WCP). When it is prepared using SPDC, the part of the figure in dashed box is not used, and the shutter (S) is open. On the other hand, when it is prepared from WCP then the dashed part will be used and the shutter will be closed. For the details of the scheme refer to the text.

In the following we discuss the effect of imperfections of detectors and the photon source on our scheme. The errors due to dark counts of the detectors cannot be eliminated even by post-selection, but the dark count rates of current detectors are pretty low for multi-photon coincidence measurements (e.g., in a detection window of 2.5ns, there will be 1.25×10^{-6} count/window), thus errors due to them can be neglected [42, 43]. Hence, the errors in the post-selected state are mainly caused by multi-photon pairs from the SPDC. The probability of correct events in this scheme is $\mathcal{O}(\eta^4\gamma^4)$ where $\gamma^2 \sim 10^{-4}$ is the photon pair generation rate per pulse in a typical SPDC process, and $\eta \sim 10^{-1}$ is the overall system efficiency which takes into account the detector efficiency and losses due to coupling and optical components [43]. On the other hand, the probability of false events due to generation of excess pairs scales as $\mathcal{O}(\eta^4\gamma^n)$ with $n \geq 6$. Thus the contribution of false events in the post-selected events is $\mathcal{O}(10^{-4})$, and hence it can be neglected.

Alternatively, we may also use WCP instead of SPDC for the ancillary photons in mode 2. In this case, the experimental setup will be modified as follows: The light pulses from Ti:sapphire laser is divided into two unequal parts by a beamsplitter. The stronger portion will go to SHG to prepare the UV used for EPR photon pair generation at the Type I BBOs. The weak part will be further attenuated through a combination of HWPs, polarizers and neutral density (ND) filters to obtain a WCP with mean photon number ν. Then the desired events occur with a rate $O(\eta^4\gamma^2\nu^2)$. If we assume that $\nu \ll 1$, then the main source of error will be the two-pair production at SPDC which leads to two photons in the input mode 1. Then even one photon in the WCP will lead to triple coincidence at modes 4, 5, and 6 with a rate $O(\eta^4\gamma^4\nu)$. The contribution of false events in the post-selected events is $\mathcal{O}(\gamma^2/\nu)$ so the error is small if $\gamma^2 \ll \nu \ll 1$. Another possible case which may lead to error is the presence of three photons in WCP while an EPR photon pair is generated at the SPDC. In that case the fourfold coincidence will occur with the rate $\mathcal{O}(\eta^4\gamma^2\nu^3)$, thus the contribution of the false events to the post-selected events will be $\mathcal{O}(\nu)$ which can be safely neglected if we choose $\nu \sim \mathcal{O}(10^{-2})$.

Mode mismatch, which decreases the fidelity of the prepared states, can be minimized by proper spectral and spatial filtering as discussed in Ref. [44]. However, this will reflect itself as reduced rate of post-selected events. Thus, there is a trade-off between the efficiency and the fidelity.

Next we consider the effect of imperfections in the beamsplitters. We assume that the reflection coefficients of the beamplitters BS_1 and BS_2 are deviated from their ideal values of $1/\sqrt{2}$ by a value of x and y, respectively. Then the action of imperfect BS1 on H (V) polarization is represented by the transformation $\hat{a}^\dagger_{1\mathrm{H(V)}} = \sqrt{(1-x)/2}\hat{a}^\dagger_{3\mathrm{H(V)}} - \sqrt{(1+x)/2}\hat{a}^\dagger_{4\mathrm{H(V)}}$ and $\hat{a}^\dagger_{2\mathrm{H(V)}} = \sqrt{(1+x)/2}\hat{a}^\dagger_{3\mathrm{H(V)}} + \sqrt{(1-x)/2}\hat{a}^\dagger_{4\mathrm{H(V)}}$, where $-1 < x < 1$. Similar expressions can be written for BS_2 by considering the corresponding modes and by replacing x with y where $-1 < y < 1$. After some straightforward but lengthy calculations, we find that the fidelity of the prepared state upon a four-fold coincidence detection becomes

$$F = \frac{(x+1)^2}{3x^2+2x+1} \tag{6}$$

where we see that the fidelity is dependent on the imperfection of only the BS_1. On the other hand, the probability of four-fold coincidence detection which is found as

$$p = \frac{(1-x)(1-y^2)}{8(3x^2+2x+1)} \tag{7}$$

depends on the imperfections of both beamsplitters. This can be simply understood if we notice that BS_1 decides the weight of the components forming the W-state: If there is a deviation from $1/\sqrt{2}$ in BS_1, the superposition will not be equally weighted which will result in lower fidelity since we accept the events whenever we have a four-fold coincidence. On the other hand, if BS_1 is ideal, then the probability of four-fold detection will be determined by the imperfections in BS_2. The plots of the dependence of fidelity and the probability of

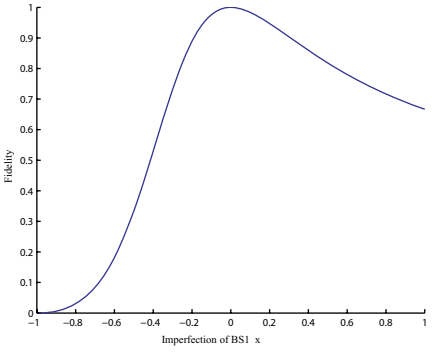


Fig. 4. Effect of deviation in the parameters of beamsplitters (BS) on fidelity of the prepared W_4. Fidelity is dependent only on the error x in BS_1 reflection coefficient.

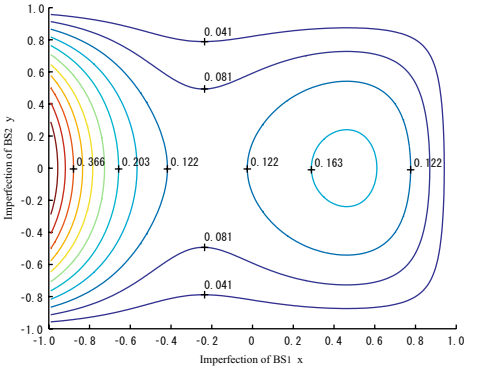


Fig. 5. Effect of imperfect 50:50 beamsplitters (BS) on the probability of four-fold coincidence detection. Deviations of reflection coefficients of BS_1 and BS_2 are represented by x and y, respectively.

four-fold coincidence are depicted in Figs. 4 and 5. It is worth noting here that if we know the amount of imperfection in BS_1, we can compensate its effect by introducing losses on only V-polarized photons. In that case, we can obtain a unit fidelity state generation but with a lower success probability. For $x < 0$, the success probability becomes $(1 + 3x)^2(1 - x)(1 - y^2)/32$; for $x > 0$ it becomes $(1 - x)^2(1 - x)(1 - y^2)/32$.

For the verification of the prepared W_4 state using WCP (e.g., $\nu \sim 2.5 \times 10^{-2}$) for the ancillary photons , we plan to perform quantum state tomography which requires measurements on 256 different bases. In order to obtain statistically significant number of coincidence events with the photon generation rates given above, we need to run the experiment at least 0.4 hours per basis which will lead to around 60 correct events. Thus experiments should be continuously run for

around 4.2 days which is lengthy but possible as such experiments have already been performed [6, 45].

Putting all together, we conclude that the proposed elementary gate is easy to implement and feasible with the current experimental technologies.

5 Expansion of Polarization Entangled GHZ States

In our proposed gate, a polarization entangled state is prepared or expanded using only passive polarization-independent components. (The phase shifter PS is used only for the purpose of making the output state in the standard form.) The polarization dependence of the gate comes from the polarization of the ancilla photons. This may suggest the possibility of expanding states other than W states by modifying the polarization of the ancilla photons. Let us consider the case in which, instead of the H-polarized two-photon ancilla state $|\,2_{\rm H}\rangle_2$, we choose the state $|\,1_{\rm H}1_{\rm V}\rangle_2$ as the state of the two ancillary photons. We still require that one photon is present in each of the output spatial modes. As we will see, it turns out that this modified gate can be used for the expansion of GHZ states.

Suppose that a V-polarized photon is present at the input mode 1. This photon undergoes two-photon interference at BS1 with the V-polarized photon from the ancilla mode 2, resulting in either both photons emerging at mode 3 or both photons emerging at mode 4. Since mode 4 must have exactly one photon for the post-selection, only the former case leads to the post-selection. In addition, the remaining H-polarized ancilla photon must go to mode 4. Hence the transformation leading to post-selection is written by $|\,1_{\rm V}\rangle_1|\rangle 1_{\rm H}1_{\rm V}{}_2 \rightarrow 2^{-3/2}|\,1_{\rm H}\rangle_4|\,1_{\rm V}\rangle_5|\,1_{\rm V}\rangle_6$. Similarly, an H-polarized photon at the input is transformed as $|\,1_{\rm H}\rangle_1|\,1_{\rm H}1_{\rm V}\rangle_2 \rightarrow 2^{-3/2}|\,1_{\rm V}\rangle_4|\,1_{\rm H}\rangle_5|\,1_{\rm H}\rangle_6$. If we rotate the polarization of the photon in mode 4 by $\pi/2$, all the photons in the output modes should have the same polarization as the input, namely, $|\,1_{\rm V}\rangle_1|\,1_{\rm H}1_{\rm V}\rangle_2 \rightarrow 2^{-3/2}|\,1_{\rm V}\rangle_4|\,1_{\rm V}\rangle_5|\,1_{\rm V}\rangle_6$ and $|\,1_{\rm H}\rangle_1|\,1_{\rm H}1_{\rm V}\rangle_2 \rightarrow 2^{-3/2}|\,1_{\rm H}\rangle_4|\,1_{\rm H}\rangle_5|\,1_{\rm H}\rangle_6$.

From the above discussion, we see that if the input photon is an equal superposition of H- and V-polarized photons, $(|\,1_{\rm H}\rangle_1 + |\,1_{\rm V}\rangle_1)/\sqrt{2}$, it will evolve into a superposition state $|\,{\rm GHZ}_3\rangle = 2^{-1/2}(|\,1_{\rm H}\rangle_4|\,1_{\rm H}\rangle_5|\,1_{\rm H}\rangle_6 + |\,1_{\rm V}\rangle_4|\,1_{\rm V}\rangle_5|\,1_{\rm V}\rangle_6)$ with a success probability of 1/8. Similarly, it transforms any size of GHZ state $|\,{\rm GHZ}_N\rangle$ to $|\,{\rm GHZ}_{N+2}\rangle$ with the same probability. Hence the use of a different

Table 2. Comparison of our scheme and other schemes in the literature for the preparation of W_4. It is seen that our scheme has higher probability of success.

Scheme	Probability	Fidelity
Our scheme	1/8	1
X. Zou et al [24]	2/27	1
B. -S. Shi et al [26]	1/16	1
Y. Li et al [25]	3/200	0.985

polarization state for the two photons in the ancillary mode enables us to expand a different class of multipartite entangled states. Here we need to mention that it is already known a parity check gate with an single-photon ancilla state will extend GHZ states by one with a probability of 1/2 [13]. In order to extend a GHZ state by 2, then the gate should be applied twice leading to an overall probability of 1/4 which is twice as high as that of our gate.

6 Conclusion

We have proposed a simple elementary optical gate which is based on post-selection for both preparing and expanding the symmetrically shared entanglement in polarization entangled W states. It has a larger success probability than other preparation methods proposed so far [see Table 2]. We believe that the proposed gate provides an easy-to-implement scheme which is feasible with the current experimental technologies. In our gate, polarization-dependent components play no essential role, and the desired transformation is achieved by multi-photon interference between the input photon and the ancilla photons. In fact, we were able to show that just by changing the state of the ancilla photons, the gate can be used for the preparation and extension of GHZ states. Observing that this gate can work equally well for both W and GHZ states put forward the question of whether it can be applied for the expansion of other classes of entangled states, such as Dicke states. The work in this direction is in progress.

Acknowledgments

This work was supported by 21st Century COE Program by the JSPS and by a MEXT Grant-in-Aid for Young Scientists (B) 17740265.

References

1. Bennett, C.H., Brassard, G., Crepeau, C., Jozsa, R., Peres, A., Wootters, W.K.: Phys. Rev. Lett. 70, 1895 (1993)
2. Ekert, A.K.: Phys. Rev. Lett. 67, 661 (1991)
3. Nielsen, M.A., Chuang, I.L.: Quantum Computation and Quantum Information. Cambridge university Press, Cambridge (2000)
4. Bourennane, M., Eibl, M., Gaertner, S., Kurtsiefer, C., Cabello, A., Weinfurter, H.: Phys. Rev. Lett. 92, 107901 (2004)
5. Gaertner, S., Kurtsiefer, C., Bourennane, M., Weinfurter, H.: Phys. Rev. Lett. 98, 020503 (2007)
6. Kiesel, N., Schmid, C., Tóth, G., Solano, E., Weinfurter, H.: Phys. Rev. Lett. 98, 063604 (2007)
7. Lu, C.-Y., Zhou, X.-Q., Gühne, O., Gao, W.-B., Zhang, J., Yuan, Z.-S., Goebel, A., Yang, T., Pan, J.-W.: Nature Physics (London) 3, 91 (2007)
8. Dür, W., Vidal, G., Cirac, J.I.: Phys. Rev. A. 62, 062314 (2000)
9. Briegel, H.J., Raussendorf, R.: Phys. Rev. Lett. 86, 910 (2001)
10. Koashi, M., Bužek, V., Imoto, N.: Phys. Rev. A. 62, 050302(R) (2000)

11. Dür, W.: Phys. Rev. A. 63, 020303(R) (2001)
12. Raussendorf, R., Browne, D.E., Briegel, H.J.: Phys. Rev. A. 68, 022312 (2003)
13. Zhao, Z., Chen, Y.-A., Zhang, A.-N., Yang, T., Bregel, H.J., Pan, J.-W.: Nature (London). 430, 54 (2004)
14. Hillery, M., Bužek, V., Berthiaume, A.: Phys. Rev. A. 59, 1829 (1999)
15. Xiao, L., Long, G.L., Deng, F.-G., Pan, J.-W.: Phys. Rev. A. 69, 052307 (2004)
16. Kempe, J.: Phys. Rev. A. 60, 910 (1999)
17. D'Hondt, E., Panangaden, P.: Quant. Inf. and Comp. 6(2), 173 (2005)
18. Okubo, Y., Wang, X.-.B., Jiang, Y.-.K., Tani, S., Tomita, A.: quant-ph/0709.4314v2 (2007)
19. Joo, J., Lee, J., Jang, J., Park, Y.-J.: quant-ph/0204003 (2002)
20. Murao, M., Jonathan, D., Plenio, M.B., Vedral, V.: Phys. Rev. A. 59, 156 (1999)
21. Yamamoto, T., Tamaki, K., Koashi, M., Imoto, N.: Phys. Rev. A. 66, 064301 (2002)
22. Zeilinger, A., Horne, M.A., Weinfurter, H., zukowski, M.: Phys. Rev. Lett. 78, 3031 (1997)
23. Rarity, J.G., Tapster, P.R.: Phys. Rev. A. 59, R35 (1999)
24. Zou, X., Pahlke, K., Mathis, W.: Phys. Rev. A. 66, 044302 (2002)
25. Li, Y., Kobayashi, T.: Phys. Rev. A. 70, 014301 (2004)
26. Shi, B.-S., Tomita, A.: J. Mod. Opt. 52, 755 (2005)
27. Lim, Y.L., Beige, A.: Phys. Rev. A. 71, 062311 (2005)
28. Tokunaga, Y., Yamamoto, T., Koashi, M., Imoto, N.: Phys. Rev. A. 71, 030301(R) (2005)
29. Walther, P., Aspelmeyer, M., Zeilinger, A.: Phys. Rev. A. 75, 012313 (2007)
30. Bouwmeester, D., Pan, J.-W., Daniell, M., Weinfurter, H.W., Zeilinger, A.: Phys. Rev. Lett. 82, 1314 (1999)
31. Resch, K.J., Walther, P., Zeilinger, A.: Phys. Rev. Lett. 94, 070402 (2005)
32. Walther, P., Resch, K.J., Rudolph, T., Schenck, E., Weinfurter, H., Vedral, V., Aspelmeyer, M., Zeilinger, A.: Nature (London) 434, 169 (2005)
33. Kiesel, N., schmid, C., Weber, U., Tóth, G., Gühne, O., Ursin, R., Weinfurter, H.: Phys. Rev. Lett. 95, 210502 (2005)
34. Kiesel, N., Bourennane, M., Kurtsiefer, C., Laskowski, W., Zukowski, M.: J. Mod. Opt. 50, 1131 (2003)
35. Eibl, M., Kiesel, N., Bourennane, M., Kurtsiefer, C., Weinfurter, H.: Phys. Rev. Lett. 92, 077901 (2004)
36. Mikami, H., Li, Y., Fukuoka, K., Kobayashi, T.: Phys. Rev. Lett. 95, 150404 (2005)
37. Resch, K.J., Walther, P., Zeilinger, A.: Phys. Rev. Lett. 94, 240501 (2005)
38. Lu, C.-Y., Zhou, X.-Q., Gühne, O., Gao, W.-B., Zhang, J., Yuan, Z.-S., Goebe, A., Yang, T., Pan, J.-W.: Nature Physics. 3, 91 (2007)
39. Häffner, H., et al.: Nature (London) 438, 643 (2005)
40. Leibfried, D., et al.: Nature (London) 438, 639 (2005)
41. Teklemariam, G., et al.: Phys. Rev. A. 66, 012309 (2002)
42. Özdemir, Ş.K., Miranowicz, A., Koashi, M., Imoto, N.: Phys. Rev. A. 64, 063818 (2001)
43. Yamamoto, T., Koashi, M., Imoto, N.: Phys. Rev. A. 64, 012304 (2001)
44. Özdemir, Ş.K., Miranowicz, A., Koashi, M., Imoto, N.: Phys. Rev. A. 66, 053809 (2002)
45. Yamamoto, T., Koashi, M., Özdemir, Ş.K., Imoto, N.: Nature (London) 421, 343 (2003)

Security Bounds for Quantum Cryptography with Finite Resources

Valerio Scarani[1] and Renato Renner[2]

[1] Centre for Quantum Technologies and Department of Physics, National University of Singapore, Singapore
[2] Institute for Theoretical Physics, ETH Zurich, Switzerland

Abstract. A practical quantum key distribution (QKD) protocol necessarily runs in finite time and, hence, only a finite amount of communication is exchanged. This is in contrast to most of the standard results on the security of QKD, which only hold in the limit where the number of transmitted signals approaches infinity. Here, we analyze the security of QKD under the realistic assumption that the amount of communication is finite. At the level of the general formalism, we present new results that help simplifying the actual implementation of QKD protocols: in particular, we show that symmetrization steps, which are required by certain security proofs (e.g., proofs based on de Finetti's representation theorem), can be omitted in practical implementations. Also, we demonstrate how two-way reconciliation protocols can be taken into account in the security analysis. At the level of numerical estimates, we present the bounds with finite resources for "device-independent security" against collective attacks.

1 Introduction

Quantum key distribution (QKD) is one of the most mature fields of quantum information science, both from the theoretical and the experimental point of view [1,2,3]. This does not mean, however, that the open questions are merely technical ones: in this paper, we are concerned with an issue that is in fact rather crucial for the assessment of security of real devices.

Most unconditional security proofs of QKD have provided an asymptotic bound for the secret key rate r, valid only in the limit of *infinitely long keys* [4-8] This reads in general [9]

$$r = S(X|E) - H(X|Y) \ , \tag{1}$$

where $S(X|E) := S(XE) - S(E)$ and $H(X|Y) := H(XY) - H(Y)$ are the conditional von Neumann and Shannon entropies, respectively, evaluated for the joint state of Alice and Bob's raw key and the system controlled by Eve (after the sifting step).

Y. Kawano and M. Mosca (Eds.): TQC 2008, LNCS 5106, pp. 83–95, 2008.

In real experiments, obviously, *finite resources* are used. As a matter of fact, the need for finite key analysis was recognized several years ago [10]. In early security proofs though, the *security parameter*

$$\text{“Deviation from the ideal case”} \leq \varepsilon\,. \tag{2}$$

was defined in terms of “accessible information”. This measure of deviation had two shortcomings, namely (i) it does not provide composable security, as proved in [11], and (ii) it has no operational interpretation. It turns out that both shortcomings are not problematic for asymptotic bounds[1], but for finite-key analysis a different definition must be used. A correct definition was used for the first time in [13], but the authors considered only a restricted class of attacks. While partial, these and other studies [14,15,16] triggered the awareness that a large N would be required for a QKD experiment to produce a secure key.

More recently, Hayashi used a valid definition (although the concern for composable security is not addressed explicitly) in his analysis of the BB84 protocol with decoy states [17]. Hayashi's bound has been applied to experimental data [18]. Apart from being possibly the first creation of a truly unconditional secure key, this experiment provides an instructive example of how critical finite key analysis is. Indeed, for the observed error rate $Q \approx 5\%$ and the choice $\varepsilon = 2^{-9}$, 4100 secret bits could be extracted from each raw key block of $n \approx \frac{N}{2} = 10^5$ bit: in other words, the final secret key rate was $r \approx 2\%$, instead of the $r \approx 43\%$ predicted by the asymptotic bound. Security bounds for finite resources are definitely one of the most urgent tasks for practical QKD [3].

Recently we have shown that the theoretical tools developed by one of us [19] can be used to provide a compact approach to security proofs in the non-asymptotic limit [20]. Our formalism leads to a generalized version of the secret key rate that reads

$$r = (n/N)\,\left[S_\xi(X|E) - \Delta - \mathrm{leak}_{\mathrm{EC}}/n\right]\,. \tag{3}$$

Comparing with (1), four modifications should be noticed: (i) only a fraction n of the signals contributes to the key, the rest must be used for parameter estimation; (ii) the parameter estimation has finite precision ξ; (iii) the task of privacy amplification itself has a security parameter Δ; and (iv) the error correction protocol may not reach the Shannon limit, so $\mathrm{leak}_{\mathrm{EC}} \geq nH(X|Y)$.

In this paper, we revisit our previous work and improve it by two important observations (Lemmas 1 and 2 below), then we present a new example of explicit calculation (Section 4.2).

[1] The absence of an operational interpretation of ε is not a problem since any deviation is supposed to vanish for asymptotically long keys. Furthermore, the fact that asymptotic bounds can be “redeemed” for composability is a consequence of the result of [12] saying that keys obtained by two-universal hashing provide composable security.

2 Basic Definitions

2.1 Definition of Security

In the existing literature on QKD, not only the analysis, but also the very *definition* of security is mostly limited to the asymptotic case; and we therefore need to revisit it here. Most generally, the security of a key K can be parametrized by its *deviation* ε from a *perfect key*, which is defined as a uniformly distributed bit string whose value is completely independent of the adversary's knowledge. In an *asymptotic* scenario, a key K of length ℓ is commonly said to be *secure* if this deviation ε tends to zero as ℓ increases. In the *non-asymptotic* scenario studied here, however, the deviation ε is always finite. This makes it necessary to attribute an *operational interpretation* to the parameter ε. Only then is it possible to choose a meaningful security threshold (i.e., an upper bound for ε) reflecting the level of security we are aiming at. Another practically relevant requirement that we need to take into account is *composability* of the security definition. Composability guarantees that a key generated by a QKD protocol can safely be used for applications, e.g., as a one-time-pad for message encryption. Although this requirement is obviously crucial for practice, it is not met by most security definitions considered in the literature [11].

Our results are formulated in terms of a security definition that meets both requirements, i.e., it is composable and, in addition, the parameter ε has an operational interpretation. The definition we use was proposed in [21,12]: for any $\varepsilon \geq 0$, a key K is said to be *ε-secure with respect to an adversary E* if the joint state ρ_{KE} satisfies

$$\frac{1}{2}\left\|\rho_{KE} - \tau_K \otimes \rho_E\right\|_1 \leq \varepsilon \ , \tag{4}$$

where τ_K is the completely mixed state on K. The parameter ε can be seen as the maximum probability that K differs from a perfect key (i.e., a fully random bit string) [12]. Equivalently, ε can be interpreted as the *maximum failure probability*, where failure means that "something went wrong", e.g., that an adversary might have gained some information on K. From this perspective, it is also easy to understand why the definition is composable. In fact, the failure probability of any cryptosystem that uses a perfect secret key only increases by (at most) ε if we replace the perfect key by an ε-secure key. In particular, because one-time pad encryption with a perfect key has failure probability 0 (the ciphertext gives zero information about the message), it follows that one-time-pad encryption based on an ε-secure key remains perfectly confidential, except with probability at most ε.

2.2 Description of the Generic Protocol

Although most practical quantum key distribution protocols are *prepare-and-measure* schemes, for analyzing their security it is often more convenient to consider an *entanglement-based* formulation. In fact, such a formulation can be

obtained by simply replacing all classical randomness by quantum entanglement and postponing all measurements. In the following, we describe the general type of protocol our analysis applies to.

1. *Distribution of quantum information:* Alice and Bob communicate over an (insecure) quantum channel to generate N identical and independent pairs of entangled particles.[2] The joint state of the N particle pairs together with the information that an adversary might have on them (e.g., acquired by eavesdropping) is denoted by $\rho_{A^N B^N E^N}$.
2. *Parameter estimation:* Alice and Bob apply a LOCC-measurement[3] to m particle pairs selected at random (using the authentic communication channel). We denote the resulting statistics by λ_m and the joint state of the remaining (not measured) particles and Eve's system by $\rho_{A^{N-m}B^{N-m}E^N}$. If the statistics λ_m fails to satisfy certain criteria, Alice and Bob abort the protocol.
3. *Measurement and advantage distillation:* Alice and Bob apply block-wise measurements $\mathcal{E}_{A^bB^b}$ on their remaining particles to get raw keys X^n and Y^n, respectively. More precisely, $\mathcal{E}_{A^bB^b}$ is an arbitrary LOCC-measurement applied sequentially to blocks A^b of b particles on Alice's side and the corresponding particles B^b on Bob's side. In a protocol without advantage distillation, $\mathcal{E}_{A^bB^b} = \mathcal{E}_A \otimes \mathcal{E}_B$ simply consists of local measurements on single particles, i.e., $b = 1$. However, $\mathcal{E}_{A^bB^b}$ might describe any operation that can be performed by Alice and Bob on a finite block of particle pairs. The resulting state is then given by $\rho_{X^nY^nE^N} = (\mathcal{E}_{A^bB^b}^{\otimes n} \otimes \mathrm{id}_{E^N})(\rho_{A^{bn}B^{bn}E^N})$, where n is the number of blocks, i.e., $nb \leq N - m$.
4. *Error correction:* Alice and Bob exchange classical messages, summarized by C, which allow Bob to compute a guess $\hat{X}^{bn}$ for Alice's string X^{bn}.
5. *Privacy amplification:* Alice and Bob generate the final key by applying an appropriately chosen hash function to X^{bn} and $\hat{X}^{bn}$, respectively. The requirement on the hash function is that it maps strings with sufficiently high min-entropy to uniform strings of a certain length ℓ (such functions are sometimes called *strong (quantum) extractors*). A typical (and currently the only known) class of functions satisfying this requirement are *two-universal hash functions* (see Section 3.4 for examples of two-universal function families).

3 Security Analysis

3.1 Security Against Collective Attacks

An attack is said to be *collective* if the interaction of Eve with the quantum channel during the distribution step is i.i.d. This implies that the state after

[2] We use the term *particle* here only for concreteness. More generally, they might be arbitrary subsystems.

[3] A *LOCC-measurement* is a measurement on a bipartite system that can be performed by local measurements on the subsystems combined with classical communication.

the distribution step is i.i.d., too, that is, $\rho_{A^N B^N E^N} = \sigma_{ABE}^{\otimes N}$, where σ_{ABE} is the density operator describing a single particle pair together with the corresponding ancilla E held by Eve.

The following analysis is subdivided into four parts. Each part gives rise to separate errors, denoted by $\varepsilon_{\mathrm{PE}}$, $\bar{\varepsilon}$, $\varepsilon_{\mathrm{EC}}$, and $\varepsilon_{\mathrm{PA}}$, respectively. These sum up to

$$\varepsilon = \varepsilon_{\mathrm{PE}} + \bar{\varepsilon} + \varepsilon_{\mathrm{EC}} + \varepsilon_{\mathrm{PA}} \ , \tag{5}$$

where ε is the security of the final key (cf. (4) for the definition of security). Making the individual contributions smaller comes at the cost of reducing other parameters that, eventually, result in a reduction of the size of the final key (see equations (6), (8), (10), and (11)).

– *Parameter estimation (minimize set of compatible states Γ and number of sample points m vs. minimize failure probability $\varepsilon_{\mathrm{PE}}$).*
Parameter estimation allows Alice and Bob to determine properties of σ_{AB}. We express this by defining a set $\Gamma_{\varepsilon_{\mathrm{PE}}}$ containing all states σ_{AB} that are *compatible* with the outcomes of the parameter estimation. For concreteness, we assume here that Alice and Bob—depending on the statistics of their measurements—either continue with the execution of the protocol or abort. The set $\Gamma_{\varepsilon_{\mathrm{PE}}}$ is then defined as the set of states σ_{AB} for which the protocol continues with probability at least $\varepsilon_{\mathrm{PE}}$ (i.e., the states from which a key will be extracted with non-negligible probability). The quantity $\varepsilon_{\mathrm{PE}}$ corresponds therefore to the probability that the parameter estimation passes although the raw key does not contain sufficient secret correlation. In particular, if Alice and Bob continue the protocol whenever they observe a statistics λ_m using a POVM with d possible outcomes then (Lemma 3 of [20])

$$\Gamma_{\varepsilon_{\mathrm{PE}}} \subseteq \left\{ \sigma_{AB} : \|\lambda_m - \lambda_\infty(\sigma_{AB})\| \leq \sqrt{\tfrac{2\ln(1/\varepsilon_{\mathrm{PE}}) + d\,\ln(m+1)}{m}} \right\} \tag{6}$$

where $\lambda_\infty(\sigma_{AB})$ denotes the (perfect) statistics in the limit of infinitely many measurements.

– *Calculation of the min-entropy (minimize decrease of min-entropy δ vs. minimize error probability $\bar{\varepsilon}$).*
Under the assumption of collective attacks, the joint state of Alice and Bob's as well as the relevant part of Eve's system after the measurement and advantage distillation step is of the form $\rho_{X^n Y^n E^{b n}} = \sigma_{XYE^b}^{\otimes n}$ where

$$\sigma_{XYE^b} := (\mathcal{E}_{A^b B^b} \otimes \mathrm{id}_{E^b})(\sigma_{ABE}^{\otimes b}) \tag{7}$$

This property allows to compute a lower bound on the smooth min-entropy of X^n given Eve's overall information E^N (before error correction), which will play a crucial role in the analysis of the remaining part of the protocol. More precisely, the min-entropy can be expressed in terms of the von Neumann entropy S evaluated for the state σ_{XE^b},

$$H_\infty^{\bar{\varepsilon}}(X^n|E^N) \geq n(S(X|E^b)_{\sigma_{XE^b}} - \delta) \tag{8}$$

where $\delta := 7\sqrt{\frac{\log_2(2/\bar{\varepsilon})}{n}}$.

– *Error correction (information leakage* leak *vs. failure probability* $\varepsilon_{\mathrm{EC}}$*).*
Error correction necessarily involves communication C between Alice and Bob. The maximum leakage of information to an adversary is expressed in terms of min- and max-entropies,

$$\mathrm{leak} := H_0(C) - H_\infty(C|X^nY^n) \ .$$

While $H_0(C)$ corresponds to the total number of relevant bits exchanged during error correction, we subtract $H_\infty(C|X^nY^n)$ which is the number of bits that are *independent* of the raw key pair (X^n, Y^n). Note the formal resemblance of this expression to the mutual information $I(C : X^nY^n)$. Indeed, the quantity leak counts the number of bits of C that are *correlated* to the raw key. In particular, any information that is independent of the raw key, such as the description of an error correcting code, does not contribute. Also, in a protocol where redundant messages are exchanged (this is for instance the case for two-way error correction schemes such as the Cascade protocol [22]), the quantity leak is generally much smaller than the total number of communicated bits.

Typically, there is a trade-off between the leakage leak and the failure probability, i.e., the maximum probability that $\hat{X} \neq X$ (where the maximum is taken over all possible states in $\Gamma_{\varepsilon_{\mathrm{PE}}}$), which we denote by $\varepsilon_{\mathrm{EC}}$. This trade-off depends strongly on the actual error correction scheme that is employed, but typically has the form

$$\mathrm{leak}_{\varepsilon_{\mathrm{EC}}} = fH_0(X|Y) + \log_2 \frac{2}{\varepsilon_{\mathrm{EC}}} \tag{9}$$

where f is a constant larger than 1. In theory, there are error correction schemes with f arbitrarily close to 1, but the decoding is usually not feasible due to computational limitations. In practice, $f \approx 1.05 - 1.2$.

– *Privacy amplification (maximize final key length* ℓ *vs. minimize failure probability* $\varepsilon_{\mathrm{PA}}$*).*
To evaluate the final key size, we need to bound the decrease of min-entropy after the leakage of information that occurred in error correction. It follows from Lemma 2 below that the smooth min-entropy of X^n given Eve's information after error correction is bounded by

$$H_\infty^{\bar{\varepsilon}}(X^n|E^NC) \geq H_\infty^{\bar{\varepsilon}}(X^n|E^N) - \mathrm{leak}_{\varepsilon_{\mathrm{EC}}} \ . \tag{10}$$

The security of the final key only depends on this quantity and the efficiency of the hash function used for privacy amplification. More precisely, if two-universal hashing[4] is used then, for any fixed $\varepsilon_{\mathrm{PA}} > 0$, the maximum length ℓ of the final key is bounded by

$$\ell \leq H^{\bar{\varepsilon}}(X^n|E^NC) - 2\log_2 \frac{1}{\varepsilon_{\mathrm{PA}}} \ . \tag{11}$$

[4] Two-universal hashing is the procedure normally used for privacy amplification.

Combining (8), (10) and (11), we conclude that the final key is ε-secure, for $\varepsilon = \varepsilon_{\mathrm{PE}} + \bar{\varepsilon} + \varepsilon_{\mathrm{EC}} + \varepsilon_{\mathrm{PA}}$ as in (5), if

$$\ell \leq n \left[\min_{\sigma_{ABE} \in \Gamma_{\varepsilon_{\mathrm{PE}}}} S(X|E^b)_{\sigma_{XE^b}} - \delta(\bar{\varepsilon}) \right] - \mathrm{leak}_{\varepsilon_{\mathrm{EC}}} - 2 \log_2 \frac{1}{\varepsilon_{\mathrm{PA}}} \qquad (12)$$

where σ_{XE^b} is related to σ_{AB} via (7) applied to a purification of σ_{AB} and where $\delta(\bar{\varepsilon}) = 7\sqrt{\frac{\log_2(2/\bar{\varepsilon})}{n}}$.

3.2 Security Analysis Against General Attacks

A general method to turn a proof against collective attacks into a proof against the most general coherent attacks is to introduce additional symmetries. Here we highlight two aspects that have been dealt with only partially in previous works.

A Lemma on symmetrization. The following lemma states that the smooth min-entropy of the state before the symmetry operations have been applied is lower bounded by the smooth min-entropy of the symmetrized state.

Lemma 1. *Let ρ_{XE} be a cq-state and let $\{f_R\}$ be a family of functions on X. Then, for any $\varepsilon \geq 0$ and R chosen at random*

$$H_\infty^\varepsilon(X|E) \geq H_\infty^\varepsilon(f_R(X)|ER) \ .$$

Proof. The statement is proved by sequentially applying rules of the smooth entropy calculus.

$$\begin{aligned} H_\infty^\varepsilon(X|E) &= H_\infty^\varepsilon(X|E) + H_\infty(R|R) \\ &= H_\infty^\varepsilon(XR|ER) \\ &= H_\infty^\varepsilon(f_R(X)XR|ER) \\ &\geq H_\infty^\varepsilon(f_R(X)|ER) \ . \end{aligned}$$

The first equality holds because $H_\infty(R|R) = 0$ (there is no uncertainty about R if R is known), and the second is a consequence of the additivity of the min-entropy (Lemma 3.1.6 of [19]). The third equality is a simply consequence of the fact that the computation of the value $f_R(X)$ while keeping the input is a unitary operation, under which the min-entropy is invariant. Finally, the inequality holds because tracing out the classical systems X and R can only decrease the smooth min-entropy (see Lemma 3.1.9 of [19]).

An important practical consequence of this Lemma is that *the symmetrization needs not be actually implemented*. Indeed, the smooth min-entropy is basically the only quantity that is relevant for the security of the final key: then, the statement of the Lemma implies that, if the symmetrized version of the protocol is secure, the original version is also secure.

Permutation symmetry. Lemma 1 above is valid for any symmetrization. Typically, one considers permutation symmetry. This can be achieved, for instance, by randomly permuting the positions of the bits [19] (more precisely, Alice and Bob both apply the same, randomly chosen, reordering to their bitstring). The symmetric states can then be shown to have properties similar to those of i.i.d. states, e.g. via the quantum de Finetti theorem [23]. This in turn leads to a bound of the form (8), with a different definition of the parameter δ (cf. Theorem 6.5.1 in [19], referring to Table 6.2 for the parameters; the corrections due to the de Finetti theorem are the terms that involve the quantities k and r). Thus, a lower bound for security using finite resources can be computed for any discrete-variable protocol.

Such a bound turns out to be very pessimistic: this is the price to pay for its generality[5]. When considering some specific protocols, there can be other, more efficient ways to obtain i.i.d. Specifically, for the BB84 [24] and the six-state protocol [25,26,27], suitable symmetries can be implemented in the protocol itself by random but coordinated bit- and phase flips [28,29]. Security bounds against general attacks can be computed by considering i.i.d. states just because of these symmetries, thus by-passing the need for the de Finetti theorem.

3.3 Decrease of the Smooth Min-Entropy by Information Leakage

An essential part of the technical security proof presented above is the following lemma, which provides a bound on the decrease of the min-entropy by information leakage in the error correction step. The statement shown here is a generalization of a corresponding statement in [19], which has been restricted to one-way error correction.

Lemma 2. *The decrease of the smooth min-entropy by the leakage of information in the error correction step is given by*

$$H_\infty^\varepsilon(X|EC) \geq H_\infty^\varepsilon(X|E) - \text{leak} \ .$$

Proof.

$$\begin{aligned} H_\infty^\varepsilon(X|EC) &\geq H_\infty^\varepsilon(XC|E) - H_0(C) \\ &\geq H_\infty^\varepsilon(X|E) + H_\infty(C|XE) - H_0(C) \\ &\geq H_\infty^\varepsilon(X|E) + H_\infty(C|XYE) - H_0(C) \\ &= H_\infty^\varepsilon(X|E) + H_\infty(C|XY) - H_0(C) \end{aligned}$$

The first two inequalities are chain rules and the third is the strong subadditivity for the smooth min-entropy. The last equality follows from the fact that $E \leftrightarrow (X,Y) \leftrightarrow C$ is a Markov chain, because the communication C is computed by Alice and Bob.

[5] Also, it is an open question whether the existing de Finetti theorem provides tight estimates, or if the bounds can be improved.

3.4 Two-Universal Hashing

As explained above, privacy amplification is usually done by two-universal hashing.

Definition 1. *A set $\mathcal{F}$ of functions f from $\mathcal{X}$ to $\mathcal{Z}$ is called* two-universal *if*

$$\Pr_{f\in\mathcal{F}}\big[f(x)=f(x')\big]\leq\frac{1}{|\mathcal{Z}|}\ ,$$

for any distinct $x,x'\in\mathcal{X}$ and f chosen at random from $\mathcal{F}$ according to the uniform distribution.

To perform the privacy amplification step, the two parties simply have to choose at random a function f from a two-universal set $\mathcal{F}$ of functions that output strings of length ℓ, where ℓ is chosen such that it satisfies (12). As shown below, there exist constructions of two-universal sets $\mathcal{F}$ of functions that are both easy to describe (the description length is equal to the input length) and that can be efficiently evaluated.

Examples of two-universal function families have first been proposed by Carter and Wegman [30,31]. One of the constructions mapping n-bit strings to ℓ-bit strings, for any $\ell\leq n$, only involves addition and multiplication in the field $\mathrm{GF}(2^n)$. It is defined as the family $\mathcal{F}=\{f_r\}_{r\in\mathrm{GF}(2^n)}$ of functions f_r that, on input x, output the ℓ least significant bits of $r\cdot x$ (where $\cdot$ denotes the multiplication in $\mathrm{GF}(2^n)$), i.e.,

$$\begin{array}{rccc} f_r: & \mathrm{GF}(2^n) & \longrightarrow & \mathrm{GF}(2^\ell) \\ & x & \longmapsto & [r\cdot x]_\ell\ . \end{array}$$

4 Computing Security Bounds

4.1 Summary of the Previous Section

Let us re-phrase the results obtained above in a more operational way. An experiment is characterized by the following parameters:

- The protocol, in particular d the number of outcomes of the measurements;
- The number of exchanged quantum signals N;
- The estimates of the channel parameters;
- The performances of the error correction protocol, in particular $\varepsilon_{\mathrm{EC}}$ and f (recall that these are functions of the parameters);
- The desired level of security ε.

We have found above the bound (12) for the extractable secret key length ℓ, which is valid for collective attacks, and also for general attacks in the case of the BB84 and the six-state protocols. By setting $r=\frac{\ell}{N}$, one gets the announced expression (3) for the secret key rate.

The expression for r is thus a function of the parameters listed above and several others, namely:

– n, b and m, subject to the constraint $nb + m \leq N$;
– $\varepsilon_{\mathrm{PE}}$, $\bar{\varepsilon}$ and $\varepsilon_{\mathrm{PA}}$, subject to the constraint $\varepsilon = \varepsilon_{\mathrm{PE}} + \bar{\varepsilon} + \varepsilon_{\mathrm{EC}} + \varepsilon_{\mathrm{PA}}$.

The best value for r is therefore obtained by optimizing (12) over the free parameters[6], for a given experiment.

In Ref. [20], we have presented such an optimization for the BB84 and the six-state protocols implemented with single photons, under the restriction that f is a constant and $b = 1$ (one-way error correction). Here, we present the computation of the security bound with finite resources for another protocol.

4.2 An Application: "Device-Independent Security" Against Collective Attacks

In 1991, Ekert noticed that the security of QKD could be related to the violation of Bell's inequalities [32]. This remark provided him with the basic intuition, but it remained purely qualitative. Only recently, on a modified version of the Ekert protocol [33], it has been possible to provide a quantitative bound on Eve's information that depends only on the violation of a particular Bell-type inequality [34]. The remarkable property of this study is that this bound is "device-independent": the knowledge of (i) the dimension of the Hilbert space in which Alice's and Bob's signals are encoded and of (ii) the details of the measurements that are performed, is *not* required. The price to pay for such generality is that there is, as of today, no argument to conclude to unconditional security[7]: the bound has been proved only for collective attacks. It is also worth stressing that, as long as the detection loophole remains open, device-independent security cannot be assessed on real setups [34,35].

Using our approach, we are going to obtain the non-asymptotic bound for device-independent security against collective attacks. We can use (12) directly. Two elements depend on the protocol and must be discussed:

– The relation between n and m depends on the measurements specified by the protocol (here we set $b = 1$). The protocol specifies that Alice performs three measurements A_0, A_1 and A_2, while Bob performs two measurements B_1 and B_2. The key is extracted out of the events (A_0, B_1). Coherence in the channel is checked by the Clauser-Horne-Shimony-Holt (CHSH) inequality [36] using $(A_1, A_2; B_1, B_2)$, i.e. from the quantity

$$\mathcal{C} = E(A_1B_1) + E(A_1B_2) + E(A_2B_1) - E(A_2B_2) \tag{13}$$

[6] Note that a parameter may be free *a priori* but be fixed in a given experiment. For instance, if in BB84 the choice of the basis is made passively through a 50-50 beam splitter, one has the additional constraint $m = nb$.

[7] This is in particular true because one does not bound the dimension of the Hilbert space; so the available de Finetti theorem cannot be used. It is important to stress that the usual unconditional security bounds *do* rely on the assumption that the dimension of the Hilbert space is known — and this is actually more serious than just a technical assumption for the proofs: most protocols, like BB84 and six-state, become provably *insecure* if one cannot rely on the fact that a meaningful fraction of the measurements are done on two-qubit signals.

where $E(A_iB_j) = \text{Prob}(a_i = b_j) - \text{Prob}(a_i \neq b_j)$ is the correlation coefficient for bits. We suppose that Alice chooses A_0 with probability p_{a0} and the other settings with equal probability $p_{a1} = p_{a2} = (1-p_{a0})/2$; and that Bob chooses B_1 with probability p_{b1} and B_2 with probability $1 - p_{b1}$. Therefore

$$n = p_{a0}p_{b1}N \ , \ m_{ij} = \frac{1}{2}(1 - p_{a0})p_{bj}N \tag{14}$$

and the other events are discarded.

– In (12), only $S_\xi(X|E) \equiv \max_{\sigma_{ABE} \in \Gamma_{\varepsilon_{\text{PE}}}} S(X|E^b)_{\sigma_{XE^b}}$ depends on the protocol, and this quantity contains only the imprecision of the parameter estimation as a finite-key effect — indeed, the other three modifications due to the finite resources, listed in Section 3.1, give rise to the other terms in (12) that are independent of the protocol. Therefore, we only have to allow a deviation of the measured parameters by the quantity $\xi(m, d) = \sqrt{\frac{2\ln(1/\varepsilon_{\text{PE}}) + d\ \ln(m+1)}{m}}$ as defined in (6). The asymptotic version [34]

$$S_{\xi=0}(X|E) = 1 - h\left(\frac{1 + \sqrt{(\mathcal{C}/2)^2 - 1}}{2}\right) \tag{15}$$

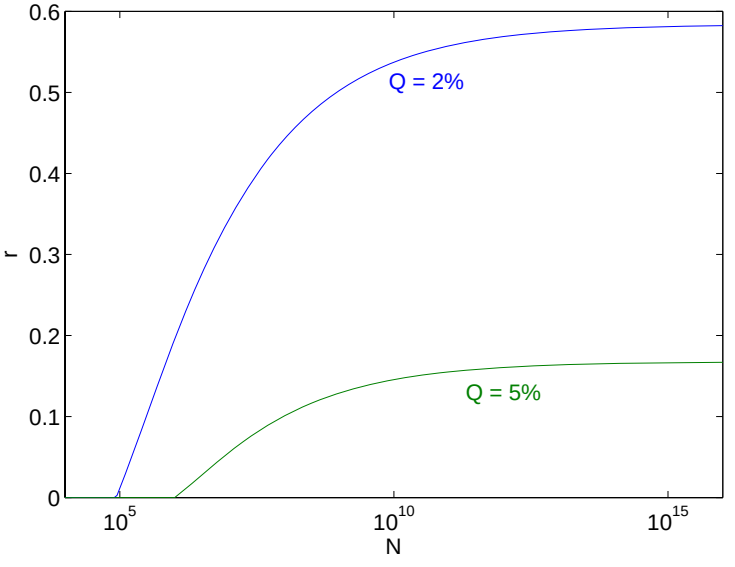


Fig. 1. Finite-key bound for device-independent security against collective attacks: secret key rate r as a function of the number of exchanged quantum signals N, for two values of the observed error rate Q; we have assumed the relation $\mathcal{C} = 2\sqrt{2}(1 - 2Q)$, which implies $\mathcal{C} \approx 2.715$ for $Q = 2\%$ and $\mathcal{C} \approx 2.546$ for $Q = 5\%$. We have fixed $\varepsilon = 10^{-5}$, $\varepsilon_{\text{EC}} = 10^{-10}$ and $f = 1.2$; we have supposed symmetric errors $\text{Prob}(a_0 \neq b_1) = Q$, so that $H_0(X|Y)$ in (9) is replaced by $h(Q)$.

depends only on $\mathcal{C}$ given in (13). Now, the deviation on the estimate of $E(A_iB_j)$ is $\xi(m_{ij}, 2)$ because a correlation coefficient can be measured by a POVM with $d = 2$ outcomes ("equal bits" and "different bits"). The most unfavorable case being obviously the one when the true value of $\mathcal{C}$ is lower than the estimated one, we obtain

$$S_\xi(X|E) = 1 - h\left(\frac{1 + \sqrt{[(\mathcal{C} - \xi)/2]^2 - 1}}{2}\right) \tag{16}$$

with $\xi = \sum_{i,j=1}^{2} \xi(m_{ij}, 2)$.

Having described the quantities that depend on the protocol, we can run the optimization of r for any N and for some chosen values of ε, $\varepsilon_{\mathrm{EC}}$, f and the observed parameters ($\mathcal{C}$ and the error rate Q). The result is plotted in Fig. 1. Similarly to what observed for BB84 and six-states [20], no key can be extracted for $N \lesssim 10^5$, and the asymptotic value is reached only for $N \gtrsim 10^{15}$. By monitoring the parameters of the optimization, one finds also that p_{a0} and p_{b_1} tend to 1 in the limit $N \to \infty$, as expected.

5 Conclusion

In this paper, we have built on our previous work on finite-key analysis [20] and completed it with some important remarks. Lemma 1 shows that the symmetrization of the data, although required to achieve security proofs, does not need to be done actively, because the min-entropy of the symmetrized data provides a bound for the min-entropy of the non-symmetrized ones. Lemma 2 extends our formalism to include two-way information reconciliation. After completing the general formalism with these Lemmas, we have applied it to derive a finite-key bound for device-independent security against collective attacks (Section 4.2).

Acknowledgments. This work is supported by the National Research Foundation and Ministry of Education, Singapore.

References

1. Gisin, N., Ribordy, G., Tittel, W., Zbinden, H.: Rev. Mod. Phys. 74, 145 (2002)
2. Dušek, M., Lütkenhaus, N., Hendrych, M.: Progress in Optics, Edt. E. Wolf, vol. 49, p. 381. Elsevier, Amsterdam (2007)
3. Scarani, V., Bechmann-Pasquinucci, H., Cerf, N.J., Dušek, M., Lütkenhaus, N., Peev, M.: arXiv:0802.4155v1
4. Shor, P.W., Preskill, J.: Phys. Rev. Lett. 85, 441 (2000)
5. Mayers, D.: Journal of the ACM 48, 351 (2001); and quant-ph/9802025
6. Lo, H.-K., Chau, H.F.: Science. 283, 2050 (1999)
7. Koashi, M.: quant-ph/0505108

8. Ben-Or, M.: Security of BB84 QKD Protocol, http://www.msri.org/publications/ln/msri/2002/quantumintro/ben-or/2/
9. Devetak, I., Winter, A.: Proc. R. Soc. Lond. A 461, 207 (2005)
10. Inamori, H., Lütkenhaus, N., Mayers, D.: Eur. J. Phys. D 41, 599 (2007) and quant-ph/0107017
11. König, R., Renner, R., Bariska, A., Maurer, U.: Phys. Rev. Lett. 98, 140502 (2007)
12. Renner, R., König, R.: Second Theory of Cryptography Conference TCC. In: Kilian, J. (ed.) TCC 2005. LNCS, vol. 3378. Springer, Heidelberg (2005)
13. Meyer, T., Kampermann, H., Kleinmann, M., Bruß, D.: Phys. Rev. A 74, 042340 (2006)
14. Lo, H.-K., Chau, H.F., Ardehali, M.: J. Cryptology. 18, 133 (2005) and quant-ph/9803007
15. Ma, X., Qi, B., Zhao, Y., Lo, H.-K.: Phys. Rev. A. 72, 012326 (2005)
16. Wang, X.-B.: Phys. Rev. Lett. 94, 230503 (2005)
17. Hayashi, M.: Phys. Rev. A 76, 012329 (2007)
18. Hasegawa, J., Hayashi, M., Hiroshima, T., Tanaka, A., Tomita, A.: arXiv:0705.3081
19. Renner, R.: Security of Quantum Key Distribution, PhD thesis, Diss. ETH No 16242, quant-ph/0512258
20. Scarani, V., Renner, R.: arXiv:0708.0709v1
21. Ben-Or, M., Horodecki, M., Leung, D.W., Mayers, D., Oppenheim, J.: Theory of Cryptography. In: Kilian, J. (ed.) TCC 2005. LNCS, vol. 3378, pp. 386–406. Springer, Heidelberg (2005) (quant-ph/0409078)
22. Brassard, G., Salvail, L.: Secret-Key Reconciliation by Public Discussion. In: Helleseth, T. (ed.) EUROCRYPT 1993. LNCS. vol. 765, pp. 410–423. Springer, Heidelberg (1994)
23. Renner, R.: Nature Physics. 3, 645 (2007)
24. Bennett, C.H., Brassard, G.: Proceedings IEEE Int. Conf. on Computers, Systems and Signal Processing, Bangalore, India, pp. 175–179. IEEE, New York (1984)
25. Bennett, C.H., Brassard, G., Breidbart, S., Wiesner, S.: IBM Technical Disclosure Bulletin. 26, 4363 (1984)
26. Bruß, D.: Phys. Rev. Lett. 81, 3018 (1998)
27. Bechmann-Pasquinucci, H., Gisin, N.: Phys. Rev. A 59, 4238 (1999)
28. Gottesman, D., Lo, H.-K.: IEEE Trans. Inf. Theory 49, 457 (2003)
29. Kraus, B., Gisin, N., Renner, R.: Phys. Rev. Lett. 95, 080501 (2005); Renner, R., Gisin, N., Kraus, B.: Phys. Rev. A. 72, 012332 (2005)
30. Carter, J.L., Wegman, M.N.: Journal of Computer and System Sciences. 18, 143 (1979)
31. Wegman, M.N., Carter, J.L.: Journal of Computer and System Sciences. 22, 265 (1981)
32. Ekert, A.K.: Phys. Rev. Lett. 67, 661 (1991)
33. Acín, A., Massar, S., Pironio, S.: New J. Phys. 8, 126 (2006)
34. Acín, A., Brunner, N., Gisin, N., Massar, S., Pironio, S., Scarani, V.: Phys. Rev. Lett. 98, 230501 (2007)
35. Zhao, Y., Fung, C.-H.F., Qi, B., Chen, C., Lo, H.-K.: arXiv:0704.3253
36. Clauser, J.F., Horne, M.A., Shimony, A., Holt, R.A.: Phys. Rev. Lett. 23, 880 (1969)

On the Design and Optimization of a Quantum Polynomial-Time Attack on Elliptic Curve Cryptography

Donny Cheung[1], Dmitri Maslov[2], Jimson Mathew[3], and Dhiraj K. Pradhan[3]

[1] Department of Computer Science, and Institute for Quantum Information Science, University of Calgary, Calgary, AB, T2N 1N4, Canada
[2] Department of Combinatorics and Optimization, and Institute for Quantum Computing, University of Waterloo, Waterloo, ON, N2L 3G1, Canada
[3] Department of Computer Science, University of Bristol, Bristol, BS8 1UB, UK

Abstract. We consider a quantum polynomial-time algorithm which solves the discrete logarithm problem for points on elliptic curves over $GF(2^m)$. We improve over earlier algorithms by constructing an efficient circuit for multiplying elements of binary finite fields and by representing elliptic curve points using a technique based on projective coordinates. The depth of our proposed implementation is $O(m^2)$, which is an improvement over the previous bound of $O(m^3)$.

1 Introduction

Quantum computing [1] has the ability to solve problems whose best classical solutions are considered inefficient. Perhaps the best-known example is Shor's polynomial-time integer factorization algorithm [2], where the best known classical technique, the General Number Field Sieve, has superpolynomial complexity $\exp(O(\sqrt[3]{n \log^2 n}))$ in the number of bits n [3]. Since a hardware implementation of this algorithm on a suitable quantum mechanical system could be used to crack the RSA cryptosystem [3], these results force researchers to rethink the assumptions of classical cryptography and to consider optimized circuits for the two main parts of Shor's factorization algorithm: the quantum Fourier transform [1,4] and modular exponentiation [5]. Quantum noise and issues of scalability in quantum information processing proposals require circuit designers to consider optimization carefully.

Since the complexity of breaking RSA is subexponential, cryptosystems such as Elliptic Curve Cryptography (ECC) have become increasingly popular. The best known classical attack on ECC requires an exponential search with complexity $O(2^{n/2})$. The difference is substantial: a 256-bit ECC key requires the same effort to break as a 3072-bit RSA key. The largest publicly broken ECC system has a key length of 109 bits [6], while the key lengths of 1024 bits and higher are strongly recommended for RSA. ECC has been recently acknowledged by

Y. Kawano and M. Mosca (Eds.): TQC 2008, LNCS 5106, pp. 96–104, 2008.

National Security Agency as a secure protocol and included in their Suite B [7]. Most ECC implementations are built over $GF(2^m)$. Software implementations, such as ECC over $GF(2^{155})$, are also publicly available [8].

However, there does exist a quantum polynomial-time algorithm that cracks elliptic curve cryptography [9]. As with Shor's factorization algorithm, this algorithm should be studied in detail by anyone interested in studying the threat posed by quantum computing to ECC. The quantum algorithm for solving discrete logarithm problems in cyclic groups such as the one used in ECC requires computing sums and products of finite field elements, such as $GF(2^m)$ [10]. Addition in $GF(2^m)$ requires only a depth-1 circuit consisting of parallel CNOT gates [11]. We present a depth $O(m)$ multiplication circuit for $GF(2^m)$ based on the construction by Mastrovito [12]. Previously, a depth $O(m^2)$ circuit was given in [11].

In Section 2 we give an overview of quantum computation, $GF(2^m)$ field arithmetic, and elliptic curve arithmetic. Section 3 outlines the quantum algorithm, and presents our improvements: the $GF(2^m)$ multiplication circuit, and projective coordinate representation. The paper concludes with some observations and suggestions for further research.

2 Preliminaries

We will be working in the quantum circuit model, where data is stored in qubits and unitary operations are applied to various qubits at discrete time steps as quantum gates. We assume that any set of non-intersecting gates may be applied within one time step. The total number of time steps required to execute an algorithm as a circuit is the *depth*. Further details on quantum computation in the circuit model can be found in [1].

We will make use of the CNOT and Toffoli gates. The CNOT gate is defined as the unitary operator which performs the transformation $|a\rangle\,|b\rangle \mapsto |a\rangle\,|a \oplus b\rangle$. The Toffoli gate [13] can be described as a controlled CNOT gate, and performs the transformation over the computational basis given by the formula $|a\rangle\,|b\rangle\,|c\rangle \mapsto |a\rangle\,|b\rangle\,|c \oplus ab\rangle$.

2.1 Binary Field Arithmetic

The finite field $GF(2^m)$ consists of a set of 2^m elements, with an addition and multiplication operation, and additive and multiplicative identities 0 and 1, respectively. $GF(2^m)$ forms a commutative ring over these two operations where each non-zero element has a multiplicative inverse. The finite field $GF(2^m)$ is unique up to isomorphism.

We can represent the elements of $GF(2^m)$ where $m \geq 2$ with the help of an irreducible *primitive polynomial* of the form $P(x) = \sum_{i=0}^{m-1} c_i x^i + x^m$, where $c_i \in GF(2)$ [14]. The finite field $GF(2^m)$ is isomorphic to the set of polynomials

over $GF(2)$ modulo $P(x)$. In other words, elements of $GF(2^m)$ can be represented as polynomials over $GF(2)$ of degree at most $m-1$, where the product of two elements is the product of their polynomial representations, reduced modulo $P(x)$ [14,15]. As the sum of two polynomials is simply the bitwise XOR of the coefficients, it is convenient to express these polynomials as bit vectors of length m. Additional properties of finite fields can be found in [14].

Mastrovito has proposed an algorithm along with a classical circuit implementation for polynomial basis (PB) multiplication [12,16], popularly known as the Mastrovito multiplier. Based on Mastrovito algorithm, [15] presents a formulation of PB multiplication and a generalized parallel-bit hardware architecture for special types of primitive polynomials, namely trinomials, equally spaced polynomials (ESPs), and two classes of pentanomials.

Consider the inputs $\boldsymbol{a}$ and $\boldsymbol{b}$, with $\boldsymbol{a} = [a_0, a_1, a_2, \ldots, a_{m-1}]^T$ and $\boldsymbol{b} = [b_0, b_1, b_2, \ldots, b_{m-1}]^T$, where the coordinates a_i and b_i, $0 \leq i < m$, are the coefficients of two polynomials $A(x)$ and $B(x)$ representing two elements of $GF(2^m)$ with respect to a primitive polynomial $P(x)$. We use three matrices in this construction:

1. an $m \times (m-1)$ reduction matrix Q,
2. an $m \times m$ lower triangular matrix L, and
3. an $(m-1) \times m$ upper triangular matrix U.

We define vectors $\boldsymbol{d}$ and $\boldsymbol{e}$ as:

$$\boldsymbol{d} = L\boldsymbol{b} \tag{1}$$

$$\boldsymbol{e} = U\boldsymbol{b}, \tag{2}$$

where L and U are defined as

$$L = \begin{bmatrix} a_0 & 0 & \ldots & 0 & 0 \\ a_1 & a_0 & \ldots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{m-2} & a_{m-3} & \ldots & a_0 & 0 \\ a_{m-1} & a_{m-2} & \ldots & a_1 & a_0 \end{bmatrix}, \quad U = \begin{bmatrix} 0 & a_{m-1} & a_{m-2} & \ldots & 0 & a_1 \\ 0 & 0 & a_{m-1} & \ldots & 0 & a_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \ldots & a_{m-1} & a_{m-2} \\ 0 & 0 & 0 & \ldots & 0 & a_{m-1} \end{bmatrix}.$$

Note that $\boldsymbol{d}$ and $\boldsymbol{e}$ correspond to polynomials $D(x)$ and $E(x)$ such that $A(x)B(x) = D(x) + x^m E(x)$. Using $P(x)$, we may construct a matrix Q which converts the coefficients of any polynomial $x^m E(x)$ to the coefficients of an equivalent polynomial modulo $P(x)$ with degree less than m. Thus, the vector

$$\boldsymbol{c} = \boldsymbol{d} + Q\boldsymbol{e} \tag{3}$$

gives the coefficients of the polynomial representing the product of $\boldsymbol{a}$ and $\boldsymbol{b}$. The construction of the matrix Q, which is dependent on the primitive polynomial $P(x)$, is given in [15].

2.2 Elliptic Curve Groups

In the most general case, we define an *elliptic curve* over a field F as the set of points $(x, y) \in F \times F$ which satisfy the equation

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_5.$$

By extending this curve to the projective plane, we may include the point at infinity $\mathcal{O}$ as an additional solution. By defining a suitable addition operation, we may interpret the points of an elliptic curve as an Abelian group, with $\mathcal{O}$ as the identity element.

In the specific case of the finite field $GF(2^m)$, it is possible to reduce the degrees of freedom in the coefficients defining the elliptic curve by the use of linear transformations on the variables x and y. In addition, it was shown in [17] that for a class of elliptic curves called *supersingular* curves, it is possible to reduce the discrete logarithm problem for the elliptic curve group to a discrete logarithm problem over a finite field in such a way that makes such curves unsuitable for cryptography. For $GF(2^m)$, these correspond to elliptic curves with parameter $a_1 = 0$. We will restrict our attention to non-supersingular curves over $GF(2^m)$, which are of the form $y^2 + xy = x^3 + ax^2 + b$, where $b \neq 0$.

The set of points over an elliptic curve also forms an Abelian group with $\mathcal{O}$ as the identity element. For a non-supersingular curve over $GF(2^m)$, the group operation is defined in the following manner. Given a point $P = (x_1, y_1)$ on the curve, we define $(-P)$ as $(x_1, x_1 + y_1)$. Given a second point $Q = (x_2, y_2)$, where $P \neq \pm Q$, we define the sum $P + Q$ as the point (x_3, y_3) where $x_3 = \lambda^2 + \lambda + x_1 + x_2 + a$ and $y_3 = (x_1 + x_3)\lambda + x_3 + y_1$, with $\lambda = \frac{y_1 + y_2}{x_1 + x_2}$. When $P = Q$, we define $2P$ as the point (x_3, y_3) where $x_3 = \lambda^2 + \lambda + a$ and $y_3 = x_1^2 + \lambda x_3 + x_3$, with $\lambda = x_1 + \frac{y_1}{x_1}$. Also, any group operation involving $\mathcal{O}$ simply conforms to the properties of a group identity element. Finally, scalar multiplication by an integer can be easily defined in terms of repeated addition or subtraction.

The elliptic curve discrete logarithm problem (ECDLP) is defined as the problem of retrieving a constant scalar d given that $Q = dP$ for known points P and Q. With this definition, we may define cryptographic protocols using the ECDLP by modifying analogous protocols using the discrete logarithm problem over finite fields.

3 Quantum Polynomial-Time Attack

With a reversible implementation for the basic elliptic curve group operations, it is possible to solve the ECDLP with a polynomial-depth quantum circuit. Given a base point P and some scalar multiple $Q = dP$ on an elliptic curve over $GF(2^m)$, Shor's algorithm for discrete logarithms [2] constructs the state

$$\frac{1}{2^m} \sum_{x=0}^{2^m-1} \sum_{y=0}^{2^m-1} |x\rangle\, |y\rangle\, |xP + yQ\rangle\,,$$

then performs a two-dimensional quantum Fourier transform over the first two registers. It was shown in [9] that this task can be reduced to adding a classically known point to a superposition of points.

3.1 Linear-Depth Circuit for $GF(2^m)$ Multiplication

We now discuss how to implement multiplication over $GF(2^m)$ as a quantum circuit. We perform the following steps:

1. Using equations (1-3), derive expressions for $\boldsymbol{d}$, $\boldsymbol{e}$ and $\boldsymbol{c}$.
2. Compute $\boldsymbol{e}$ in an ancillary register of m qubits.
3. Transform $\boldsymbol{e}$ into $Q\boldsymbol{e}$, using a linear reversible implementation.
4. Compute and add $\boldsymbol{d}$ to the register occupied by $Q\boldsymbol{e}$.

We illustrate the above steps with an example using $P(x) = x^4 + x + 1$.

1. Expressions for $\boldsymbol{d}$ and $\boldsymbol{e}$ derived from equations (1-2) are shown below.

$$\boldsymbol{d} = \begin{bmatrix} a_0b_0 \\ a_1b_0 + a_0b_1 \\ a_2b_0 + a_1b_1 + a_0b_2 \\ a_3b_0 + a_2b_1 + a_1b_2 + a_0b_3 \end{bmatrix}, \quad \boldsymbol{e} = \begin{bmatrix} a_3b_1 + a_2b_2 + a_1b_3 \\ a_3b_2 + a_2b_3 \\ a_3b_3 \end{bmatrix}.$$

   We also construct the matrix $Q = \begin{bmatrix} 1\,0\,0 \\ 1\,1\,0 \\ 0\,1\,1 \\ 0\,0\,1 \end{bmatrix}$.

   From (3), we compute the multiplier output

$$\boldsymbol{c} = \boldsymbol{d} + Q\boldsymbol{e} = \begin{bmatrix} d_0 + e_0 \\ d_1 + e_1 + e_0 \\ d_2 + e_1 + e_2 \\ d_3 + e_2 \end{bmatrix}.$$

2. We first compute e_0, e_1, and e_2 in the ancilla, as shown in Figure 1 (gates 1-6).
3. We next implement the matrix transformation $Q\boldsymbol{e}$ (gates 7-9).
4. Finally, we compute the coefficients d_i, $0 \leq i < m$, and add them to the ancilla to compute $\boldsymbol{c}$ (gates 10-19).

At this point, we have a classical reversible circuit which implements the transformation $|a\rangle|b\rangle|0\rangle \mapsto |a\rangle|b\rangle|a \cdot b\rangle$. However, if we input a superposition of field elements, then the output register will be entangled with the input. If one of the inputs, such as $|b\rangle$ is classically known, then we may also obtain $|b^{-1}\rangle$ classically. Since we may construct a circuit which maps $|a \cdot b\rangle|b^{-1}\rangle|0\rangle \mapsto |a \cdot b\rangle|b^{-1}\rangle|a\rangle$, we may apply the inverse of this circuit to the output of the first circuit to obtain $|a\rangle|b\rangle|a \cdot b\rangle \mapsto |0\rangle|b\rangle|a \cdot b\rangle$ using an ancilla set to $|b^{-1}\rangle$. This gives us a quantum circuit which takes a quantum input $|a\rangle$ and classical

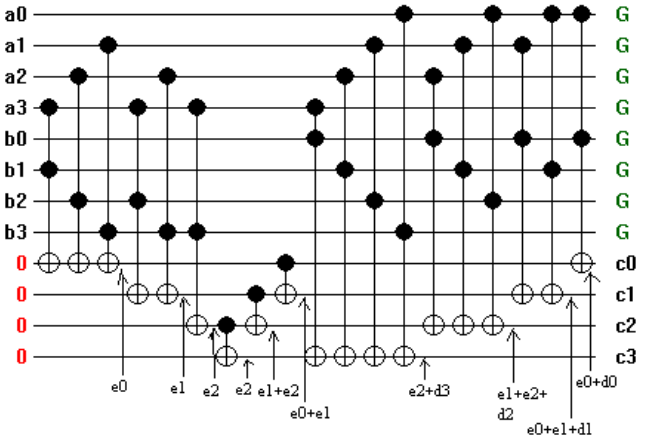


Fig. 1. Circuit for $GF(2^4)$ multiplier with $P(x) = x^4 + x + 1$

input $|b\rangle$, and outputs $|a \cdot b\rangle\,|b\rangle$. When $|b\rangle$ is not a classical input, the output of the circuit may remain entangled with the input, and other techniques may be required to remove this entanglement. However, we emphasize that this is not required for a polynomial-time quantum algorithm for the ECDLP [9].

This technique can be applied for any primitive polynomial $P(x)$. In some circumstances, we may derive exact expressions for the number of gates required.

Lemma 1. *A binary field multiplier for primitive polynomial $P(x)$ can be designed using at most $2m^2 - 1$ gates. If $P(x)$ is a trinomial or an all-one polynomial, where each coefficient is 1, we require only $m^2 + m - 1$ gates.*

Proof. There are three phases to the computation: computing $\boldsymbol{e}$, computing $Q\boldsymbol{e}$, and adding $\boldsymbol{d}$ to the result. For $\boldsymbol{e}$ and $\boldsymbol{d}$, each pair of coefficients which are multiplied and then added to another qubit requires one Toffoli gate. This requires

$$\sum_{i=0}^{m-1} i = \frac{m(m-1)}{2}, \text{ and } \sum_{i=0}^{m} i = \frac{m(m+1)}{2}$$

gates respectively, for a total of m^2 gates. Now, we consider the implementation of the transformation Q.

In general, $m^2 - 1$ CNOT gates suffice for any linear reversible computation defined by the matrix Q in equation (3) [18]. This gives a general upper bound of $2m^2 - 1$ gates. In the specific case of the All-One-Polynomial, the operation Q consists of adding e_1 to each of the other qubits, requiring $m - 1$ CNOT operations. This gives a total of $m^2 + m - 1$ operations.

For a trinomial, we have a primitive polynomial $P(x) = x^m + x^k + 1$ for some constant k such that $1 \leq k < m$. To upper bound the number of gates required to implement Q, we may consider the inverse operation, in which we begin with a

polynomial of degree at most $m-1$, and we wish to find an equivalent polynomial where each term has degree between $m-1$ and $2m-2$. Increasing the minimum degree of a polynomial requires one CNOT operation, and this must be done $m-1$ times. Again, this gives a total of m^2+m-1 operations.

3.2 Parallelization

We construct a parallelized version of this network by considering the three parts of the computation: $\boldsymbol{e}$, $Q\boldsymbol{e}$ and adding $\boldsymbol{d}$. For $\boldsymbol{e}$ and $\boldsymbol{d}$, note that given coefficients a_i and b_j where the value of $i-j$ is fixed, the target qubit of each separate term a_ib_j is different. This means that they may be performed in parallel. In the case of $\boldsymbol{e}$, we evaluate a_ib_j whenever $i+j \geq m$. This means that the values of $i-j$ may range from $-(m-2)$ to $m-2$, giving a depth $2m-3$ circuit for finding e. Similarly, for $\boldsymbol{d}$, we evaluate a_ib_j whenever $i+j < m$. The values of $i-j$ range from $-(m-1)$ to $m-1$, giving a depth $2m-1$ circuit.

To compute $Q\boldsymbol{e}$, at most m^2-1 CNOT gates are used. In [18], it is shown that such a computation can be done in a linear number of stages, with a depth of $6m+O(1)$. This gives us a total depth of $10m+O(1)$ for the multiplication circuit. An implementation which replaces the Toffoli gate with 1- and 2-qubit gates requires a circuit depth of $26m+O(1)$.

3.3 Projective Representation

When points on an elliptic curve are represented as *affine coordinates* (x, y), performing group operations on such points requires finding the multiplicative inverse of elements of $GF(2^m)$. This operation takes much longer to perform than the other field operations required, and it is desirable to minimize the number of division operations. For example, [19] gives a quantum circuit of depth $O(m^2)$ which uses the extended Euclidean algorithm.

By using *projective* coordinate representation, we can perform group operations without division. Instead of using two elements of $GF(2^m)$ to represent a point, we use three elements, (X, Y, Z) to represent the point $(\frac{X}{Z}, \frac{Y}{Z})$ in affine coordinates. Dividing X and Y by a certain quantity is now equivalent to multiplying the third coordinate (Z) by this quantity. Extensions to this concept have also been explored, where different information about an elliptic curve point is stored in several coordinates. Another advantage to projective coordinates is that the point at infinity $\mathcal{O}$ can simply be represented by setting Z to zero. Unfortunately, one issue with projective coordinates for reversible computing is that there are more than one representation for any particular point.

To represent the point (x, y), we use the equal superposition of all of these representations

$$|P(x,y)\rangle = \frac{1}{\sqrt{2^m}} \sum_{z \in GF(2^m)} |xz\rangle\,|yz\rangle\,|z\rangle\,.$$

We construct this state by starting with the state $1/\sqrt{2^m}\sum_z |z\rangle\,|z\rangle\,|z\rangle$, and multiplying the first and second registers by x and y, respectively.

Exact formulas for point addition in projective coordinates can be easily derived by taking the formulas for the affine coordinates under a common denominator and multiplying the Z coordinate by this denominator. These are detailed in [20]. Since the ECDLP can be solved by implementing elliptic curve point addition where one point is "classically known" [9], we may implement these formulas using the multiplication algorithm presented in Section 3.1 and by being careful to uncompute any temporary registers used. Since the number of multiplication operations used in these formulas is fixed, we may implement elliptic curve point addition with a known classical point with a linear depth circuit. This represents an improvement on the algorithm of [19], which makes use of an $O(m^2)$-depth circuit for inversion of $GF(2^m)$ field elements.

Finally, to construct the state required for solving the ECDLP, we use the standard "double and add" technique, which requires implementing the point addition circuit for each value 2^iP and 2^iQ, where $0 \leq i < m$. Performing $2m$ instances of a linear depth circuit, followed by a quantum Fourier transform gives a final depth complexity of $O(m^2)$ for the circuit which solves the ECDLP over $GF(2^m)$. This improves the previously known upper bound of $O(m^3)$ [9].

4 Conclusion

We considered the optimization of the quantum attack on the elliptic curve discrete logarithm problem, on which elliptic curve cryptography is based. Our constructions include a linear depth circuit for binary field multiplication and efficient data representation using projective coordinates. Our main result is the depth $O(m^2)$ circuit for computing the discrete logarithm over elliptic curves over $GF(2^m)$. Further research may be devoted toward a better optimization, study of architectural implications, and the fault tolerance issues.

References

1. Nielsen, M., Chuang, I.: Quantum Computation and Quantum Information. Cambridge University Press, Cambridge (2000)
2. Shor, P.W.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM Journal of Computing 26, 1484–1509 (1997)
3. Von Zur Gathen, J., Gerhard, J.: Modern Computer Algebra. Cambridge University Press, Cambridge (1999)
4. Cleve, R., Watrous, J.: Fast parallel circuits for the quantum Fourier transform. IEEE Symposium on Foundations of Computer Science 41, 526–536 (2000)
5. Meter, R.V., Itoh, K.M.: Fast quantum modular exponentiation. Physical Review A 71, 052320 (2005)
6. Certicom. Certicom announces elliptic curve cryptography challenge winner. Certicom press release (2004)
7. NSA Suite B Factsheet, `http://www.nsa.gov/ia/industry/crypto_suite_b.cfm`
8. Agnew, G.B., Mullin, R.C., Vanstone, S.A.: An implementation of elliptic curve cryptosystems over $GF(2^{155})$. IEEE Journal on Selected Areas in Communications 11(5), 804–813 (1993)

9. Proos, J., Zalka, C.: Shor's discrete logarithm quantum algorithm for elliptic curves. Quantum Information and Computation 3, 317–344 (2003)
10. Jozsa, R.: Quantum algorithms and the Fourier transform. Proc. R. Soc. Lond. A 454, 323–337 (1998)
11. Beauregard, S., Brassard, G., Fernandez, J.M.: Quantum arithmetic on Galois fields. arXiv:quant-ph/0301163 (2003)
12. Mastrovito, E.D.: VLSI designs for multiplication over finite fields $GF(2^m)$. In: Proceedings of the Sixth Symposium on Applied Algebra, Algebraic Algorithms, and Error Correcting Codes, vol. 6, pp. 297–309 (1988)
13. Toffoli, T.: Reversible computing. Tech memo MIT/LCS/TM-151, MIT Lab for Computer Science (1980)
14. Pradhan, D.K.: A theory of Galois switching functions. IEEE Transactions on Computers 27, 239–248 (1978)
15. Reyhani-Masoleh, A., Hasan, M.A.: Low complexity bit parallel architectures for polynomial basis multiplication over $GF(2^m)$. IEEE Transactions on Computers 53, 945–959 (2004)
16. Mastrovito, E.D.: VLSI Architectures for Computation in Galois Fields. PhD Thesis, Linkoping University, Linkoping, Sweden (1991)
17. Menezes, A.J., Okamoto, T., Vanstone, S.A.: Reducing elliptic curve logarithms to logarithms in a finite field. IEEE Transactions on Information Theory 39, 1639–1646 (1993)
18. Maslov, D.: Linear depth stabilizer and quantum Fourier transformation circuits with no auxiliary qubits in finite neighbor quantum architectures. Physical Review A 76, 052310 (2007)
19. Kaye, P.: Optimized quantum implementation of elliptic curve arithmetic over binary fields. Quantum Information and Computation 5, 474–491 (2005)
20. Hankerson, D., López Hernandez, J., Menezes, A.: Software implementation of elliptic curve cryptography over binary fields. In: Proceedings of the Second International Workshop on Cryptographic Hardware and Embedded Systems, vol. 2, pp. 1–24 (2000)

Architecture of a Quantum Multicomputer Implementing Shor's Algorithm

Rodney Van Meter[1], W.J. Munro[2,3], and Kae Nemoto[3]

[1] Faculty of Environment and Information Studies, Keio University
5322 Endo, Fujisawa, Kanagawa, 252-8520, Japan
[2] Hewlett-Packard Laboratories
Filton Road, Stoke Gifford, Bristol BS34 8QZ, United Kingdom
[3] National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan

Abstract. We have created the architecture of a quantum multicomputer and analyzed its performance for running Shor's algorithm for factoring large numbers. In this paper, we combine fault tolerance techniques with performance goals for our architecture, which uses a linear interconnect and six logical qubits per node. Our performance target of factoring a 1,024-bit number in one month requires teleporting 6.2 logical qubits per second on each link in the system, which translates to 3,300 physical teleportations per second on each link. Starting from a Bell state with fidelity $F = 0.638$, as a qubus-based cavity QED interconnect might generate with a qubit-to-qubit loss of 3.4dB, about 1.5 million physical entanglement attempts per second are enough to reach this level of performance. Our analysis suggests that systems capable of solving classically intractable problems are well within reach; once basic technological hurdles are overcome, the multicomputer architecture supports rapid scaling to very large systems.

1 Introduction

Researchers have begun designing systems for distributed quantum computation (DQC). The basic principle of distributed quantum computation has been known for a decade [1, 2, 3, 4]. A few uses for geographically distributed entanglement have been developed [5, 6, 7], but more recently, interest has grown in the use of distributed quantum computation within a single laboratory. We refer to such distributed-memory systems as quantum multicomputers [8, 9].

The interest in laboratory-level DQC stems from the difficulty of scaling up any individual quantum computer to hold the millions of physical qubits that may be necessary for some applications. Any quantum computing technology will have a limit to the number of physical qubits that can be supported in a single device. These limits are not yet well understood, but may range into the thousands for some solid-state systems. The quantum dot cavity QED systems we are currently investigating and Josephson junction phase qubits both require qubit-to-qubit spacing of around 50-100μm; a 1cm^2 chip can therefore hold a maximum of only 10-40,000 physical qubits, even before considering various sources of overhead. Quantum error correction reduces the number available to applications to only a handful of logical qubits. Thus, researchers are designing devices to be connected into multicomputers [10, 11].

Y. Kawano and M. Mosca (Eds.): TQC 2008, LNCS 5106, pp. 105–114, 2008.

In previous work, we have determined that a linear network of nodes will perform well on the addition subroutine, and that the most efficient form of the algorithm on a quantum multicomputer teleports data qubits, rather than gates [8]. The mechanics of the original Vedral-Barenco-Ekert (VBE) ripple-carry adder [12] work best for nodes that hold at least four logical qubits. The links between nodes may be serial, and two layers of the [23,1,7] Steane quantum error correction code will allow teleportation error rates of around one percent [13]. We have also established a performance goal of factoring a 1,024-bit number in one month of wall-clock time on our system, which should exceed the performance of the best classical systems available [14].

In this paper, we establish new performance requirements for the interconnect links, increase the level of detail on the node requirements and on QEC, and provide a more complete analysis of the application algorithm. Our previous algorithm work was at the *logical* level; here we combine those results with the *physical*-level work on the interconnect, and match the system up with the operational goals. We show that physical qubits must be teleported between nodes at about 3,300 teleportations per second on each link in the system, and individual nodes must contain about 10,000 physical qubits. Using the qubus scheme for the interconnect [15], initial fidelities are low, and purification must be used. For a cavity QED system as a candidate technology, the resulting requirement is about 1.5 million entanglement attempts per second on each link.

2 Shor's Algorithm

Before we can design a computer, we have to understand how it will be used. Characterizing the workload of a proposed system is the first important task in the design process. For our quantum multicomputer design, we have chosen Shor's algorithm for factoring large numbers as our primary target application [16, 17]. Shor's algorithm requires arithmetic and the quantum Fourier transform (QFT), both of which are considered fundamental building blocks of other algorithms. Thus, we expect that a system designed for Shor's algorithm will work well on a variety of problems.

2.1 Overview

The algorithm consists of both classical and quantum portions, with the quantum portion being a period-finding method based on the QFT and arithmetic to calculate the modular exponentiation of two integers. The period-finding method operates on two quantum registers, the control register and the function result register; in the end, we will actually measure the *control* register to find the period of the function (this is perhaps the most counter-intuitive feature of the algorithm).

To factor a number N whose length is n bits, we begin by checking that the number is not even and determining that it not an integer power, a^b, for $a \geq 1$ and $b > 2$. Efficient classical methods are known for this calculation and for finding the greatest common divisor ($\gcd$), which we will not present. Next, choose an integer $2 < x < N$, and check that $\gcd(x, N) = 1$; if not, return $\gcd(x, N)$. The value of x need not be strictly random, but is not important except that repeating the algorithm after a failure sometimes requires that x be changed.

Next, use the quantum period-finding method to determine the order r of x modulo N. If r is even and $x^{r/2} \neq -1 \bmod N$, calculate $\gcd(x^{r/2}-1, N)$ and $\gcd(x^{r/2}-1, N)$. One of these should be a factor of N. If not, or if r is odd, repeat the algorithm, choosing a different x.

The order of x modulo N is found by noting that we can calculate the modular exponentiation $x^a \bmod N$ for all a. We use two quantum registers, which will hold, respectively, a and $x^a \bmod N$. The register for a must be $2n$ qubits long. Starting from the state

$$\frac{1}{2^n} \sum_{a=0}^{2^{2n}-1} |a\rangle|1\rangle \tag{1}$$

in which all of the qubits are disentangled, the modular exponentiation then produces the state

$$\frac{1}{2^n} \sum_{a=0}^{2^{2n}-1} |a\rangle|x^a \bmod N\rangle. \tag{2}$$

Once we have that entangled state [18], we apply the QFT *to the first register*, measure both registers, and use the value in the first register (discarding the second) to find the order of x modulo N, and from there the factors of N.

2.2 Modular Exponentiation

The modular exponentiation portion of the algorithm constitutes the bulk of the run time. In this paper, we will primarily concentrate on the cost of doing arithmetic, rather than the QFT.

In general, quantum modular exponentiation algorithms are created from building blocks that do modular multiplication,

$$|\alpha\rangle|0\rangle \rightarrow |\alpha\rangle|\alpha\beta \bmod N\rangle \tag{3}$$

where β and N may or may not appear explicitly in quantum registers. This modular multiplication is built from blocks that perform modular addition,

$$|\alpha\rangle|0\rangle \rightarrow |\alpha\rangle|\alpha+\beta \bmod N\rangle \tag{4}$$

which, in turn, are usually built from blocks that perform addition and comparison. The addition algorithm that we use is Vedral, Barenco and Ekert's ripple-carry adder [12]. As shown in Figure 1, this adder consists of two blocks called CARRY and SUM.

In most modular exponentiation algorithms, the multiplication step is performed $2n$ times, once for each bit in the register $|a\rangle$ [12, 19]. The running product is multiplied by a value held in a quantum register. That value is either 1, if the corresponding bit of $|a\rangle$ is zero, or x^{2^i}, if the corresponding bit is one. Let $d_i = x^{2^i}$, and $a_{n-1}a_{n-2}..a_0$ be the binary expansion of a. The d_i can be calculated classically, but $|a\rangle$ is a quantum register. The value $x^a \bmod N$ can be rewritten [12, 20] as

$$\prod_{j=0}^{2n} d_j^{a_j} \bmod N. \tag{5}$$

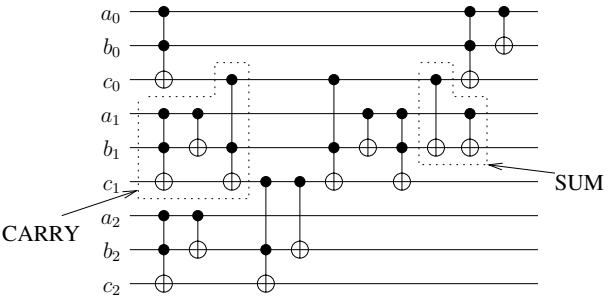


Fig. 1. Execution of a three-qubit VBE adder on an abstract architecture. This circuit will add the register $|a\rangle$ to the register $|b\rangle$, so that $|a\rangle|b\rangle \rightarrow |a\rangle|a+b\rangle$, using $|c\rangle$ as ancillae for calculating the carry.

Fundamentally, quantum modular exponentiation is $O(n^3)$; that is, the number of quantum gates or operations scales with the cube of the length in bits of the number to be factored [12,16,19]. It consists of $2n$ modular multiplications, each of which consists of $O(n)$ additions, each of which requires $O(n)$ operations. The form we use requires about $4n^2$ calls to the addition routine [21].

3 Multicomputer Architecture

Our quantum multicomputer (QMC) architecture consists of a group of semi-autonomous nodes, connected by a quantum network and a real-time classical network, all controlled from a classical front-end computer that determines the program to be run and the role to be played by each node. Each node contains four logical qubits for the algorithmic data and two logical buffer qubits for communications, which are used to send and receive data through a qubus entangling channel, giving a total of six logical qubits per node. All quantum error correction is performed locally, within a single node. Our complete system will consist of 1,024 nodes for the arithmetic unit, plus a few more for the control variables for the algorithm. An overview of the quantum components of the architecture is shown in Figure 2.

3.1 Workload

A computer system cannot be designed without an understanding of its target workload and expected performance. Above, we suggested a goal of factoring a 1,024-bit number in one month of wall-clock execution time. Next we show that achieving this performance level will require 6.2 logical qubit teleportations per second per node.

The computational core of Shor's algorithm is the quantum modular exponentiation. The modular exponentiation can be performed in roughly $4n^2$ calls to the addition subroutine, or about four million calls for a 1,024-bit number. The execution time of the adder can be optimized down to the time to teleport two logical qubits between a pair of nodes, by using all of the links concurrently, when the time to generate a high-fidelity Bell pair is long compared to the local gate time [8].

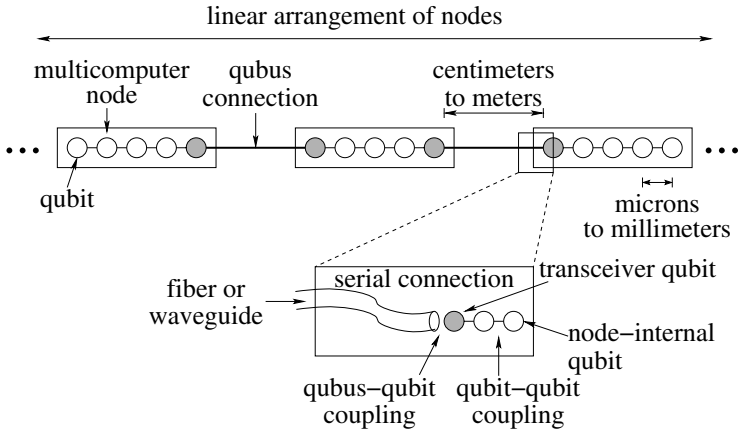


Fig. 2. Quantum portions of our quantum multicomputer

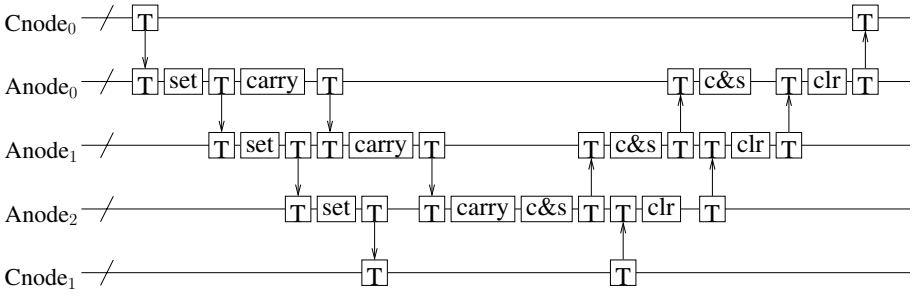


Fig. 3. Execution of a three-qubit controlled VBE adder in a five-node quantum multicomputer. Each horizontal line represents a node with six logical qubits. "A" nodes are used to execute the adder circuit and hold active data, while "C" nodes hold control variables. The "T" boxes and vertical arrows are the local gates and classical communication of teleportation, using Bell pairs created and purified using the qubus interconnect. The "set" boxes are local gates to set addends. "carry" boxes execute the VBE CARRY circuit, and "c&s" boxes the inverse of CARRY and SUM. "clr" boxes clear the addends. Note the four teleportation operations between Anode_0 and Anode_1, and the four between Anode_1 and Anode_2.

To compose a series of additions into a multiplication, a control variable must be used to set one of the operands. We must teleport the control variable into the node, use it to set the operand via a CNOT gate, then pass it on to another node to use. In our linear network, the control variable can be assumed to arrive from the node on the left, and will be sent on to the right. This sequence is reversed to clear the operand after the addition. Thus, we use a total of four teleportation operations on each link to complete one addition cycle, half for algorithm control and half for propagating the carry in addition operations, as shown in Figure 3. Sixteen million logical qubit teleportations can be accomplished in one month if the rate of logical teleportations is 6.2 per second.

For two levels of the [23,1,7] Steane error correcting code, teleporting a logical qubit requires $23^2 = 529$ physical qubit teleportations. Because we are using serial links between the multicomputer nodes, each link must support $6.2 \times 529 = 3300$ physical teleportations per second.

3.2 Interconnect Requirements

Teleportation consumes Bell pairs. The interconnect subsystem is tasked with creating the high-fidelity Bell pairs that we need. When designing an interconnect, we must worry about the *physical* and *link* levels, as well as the *interconnect topology*. For the physical level, we have focused on the qubus system as our preferred candidate for the interconnect [15], though the principles demonstrated here apply to single-photon interconnects, as well. At the link level, as mentioned in Section 3.1, we have shown that serial links and two levels of the [23,1,7] code will do. For the topology, we have chosen a linear network.

Simulations of a cavity QED form of qubus using homodyne measurement for quantum repeaters, as proposed by van Loock et al., reveal an upper bound of about 5dB loss qubit-to-qubit, beyond which entanglement fidelity is too low for purification to operate [22]. For a more practical limit, a loss of 3.4dB will give an initial fidelity $F = 0.638$. The entanglement attempts succeed about 40% of the time, and purification itself is a probabilistic process, so using purification to reach a final fidelity $F = 0.98$ or better requires an average of about 450 qubus entanglement attempts. To execute 6.2 logical teleportations per second requires $450 \times 3300 = 1.5 \times 10^6$ physical entanglement attempts per second.

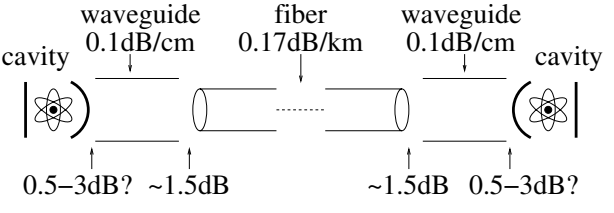


Fig. 4. Example qubit-to-qubit losses in a cavity QED system. The "atom" symbol represents a solid-state quantum dot held in a micromachined cavity.

Figure 2 shows prospective signal strength losses from qubit to qubit for a cavity QED system. The lower row of loss numbers in the figure is for the couplings in the system, and the upper row is the transmission losses in the fiber and waveguide. There are additional losses before the laser beam reaches the first cavity, and between the second cavity and the measurement device, that are not shown in the figure. However, those losses do not affect the fidelity of the entanglement created by the qubus system; only losses *between* the two cavities do.

For a multicomputer configuration, with a fiber length of only a few meters and a chip size of around a square centimeter, the loss is clearly dominated by the couplings.

Using a holographic lens, for example, state of the art insertion loss at telecom wavelengths is around 1.5dB, but simulations suggest that reducing that to a mere fraction of a dB may be possible [23]. The coupling of the waveguide to the cavity may range from a fraction of a dB for an overcoupled system to e.g. 3dB for a microdisk cavity due to backscattering, but is highly dependent on the design and potentially subject to improvement with careful engineering [24]. To reach our preferred loss level of three to four dB, modest improvement in coupling efficiency is needed. This figure makes it clear that avoiding any unnecessary losses, such as switches, in the quantum path is an important goal when designing qubus-based interconnects.

In prior work, we investigated several possible topologies for interconnecting the nodes: a shared bus, and a fully-connected architecture in which external switching is assumed to route the laser pulses from any node to any other, both of which use only one transceiver qubit per node; and three topologies using two transceiver qubits per node, the linear network, two shared buses, and two fully-connected networks that operate independently [8]. We found that, for our workload, the computation time is dominated by the time to create high-fidelity Bell pairs. Parallelism in the interconnect is critical, ruling out the two bus-based topologies, despite their ability to move data between arbitrary pairs of nodes. We rule out the fully-connected networks both because of the switching they require and the fact that, even under ideal circumstances, they do not improve system performance; most of their advantages in routing data do not help us.

For Shor's algorithm and other arithmetic-based algorithms, the flow of information turns out to be both regular and local. In the multicomputer, a simple ripple-carry adder is faster than the more complex carry-lookahead adder up to very large problem sizes. In the abstract, the linear network is well-suited to ripple-carry. It provides natural parallelism, in that each link can be operating independently and concurrently. The linear network requires no switching, which would add implementation complexity as well as loss that reduces the fidelity of the base-level Bell pairs generated by the qubus interconnect.

Given that the bottleneck is the interconnect, it is desirable to increase the parallelism on each link. Switching or demultiplexing of incoming signals to multiple transceiver qubits is impractical, because switch losses are often several dB. However, in the cQED system, the on-chip waveguide is expected to span multiple physical qubits. Only a single qubit is "enabled" at a time, by bringing the cavity into resonance with the waveguide. Because this process is fast relative to signal propagation times within the larger multicomputer, we can treat the behavior of the link as approximately a parallel link, providing some of the physical simplicity of a serial link with some of the performance benefits of a parallel link [13].

We have therefore concluded that the linear network is the best option, given the limitations of quantum technology for the foreseeable future.

3.3 Node Requirements

Above, we have established that each node holds six logical qubits and transceiver qubits to connect it to its neighbors on the left and the right using the qubus interconnect.

Using two layers of the [23,1,7] code, each logical qubit requires $23^2 = 529$ physical qubits, requiring a minimum of $6 \times 529 = 3174$ physical qubits per node.

Single-qubit rotations on logical qubits are difficult, and fault tolerance demands that direct data qubit-to-data qubit interactions be minimized. Fowler therefore recommends using three registers for each logical qubit [25], boosting the number of qubits per node to $6 \times 3 \times 23^2 = 9522$ physical qubits. Including transceiver qubits for the interconnect and purification buffering, we require about 10,000 physical qubits per node.

We, in conjunction with several collaborators, are working on using semiconductor quantum dots in microcavities, with two-qubit interactions mediated by qubus-style use of weak nonlinearities, similar to a technology being studied for use in quantum repeaters [15, 22]. In this system, each node will be a single chip, but the details of internal and external communication differ somewhat from the abstract model we have examined to date. Other researchers are also designing systems that are variants of quantum multicomputers [11, 26, 27].

As mentioned in Section 1, in earlier work, we determined that it is preferable to have nodes large enough to hold one or more logical qubits. However, most of the technologies currently being studied have limits far below the ten thousand qubits we suggested above. Methods to allow very small nodes to successfully run distributed quantum error correction are being investigated, but the detailed architecture and performance analysis of systems to be built on these techniques remain incomplete [11, 26].

4 Conclusion

In this paper, we have described a quantum multicomputer built from about ten million physical qubits packaged in a thousand separate nodes, connected by a qubus-based interconnect. Simulations of the creation of Bell pairs and their purification suggest that the performance will be more than adequate. The next major step in architectural design will be the internal architecture of individual CQED nodes, including scheduling of resources, control, quality, and performance.

We believe a multicomputer structure can be developed more easily than a large-scale monolithic system, allowing a variety of physical qubit technologies to scale quickly beyond currently-perceived limits. The multicomputer offers a simple path to incremental progress: once the basic intra- and inter-node technological hurdles are cleared, systems of a few nodes will immediately become realizable. Moreover, our research on large-scale architectures shows that scaling from a few nodes to over a thousand imposes essentially only the requirement for stronger quantum error correction inherent in longer computations; new technologies such as a complex, switched optical interconnect or many more I/O ports are unnecessary. Thus, the multicomputer has the potential to dramatically accelerate the arrival of quantum computers that generate results beyond the reach of classical systems.

Acknowledgments. The authors thank MEXT and QAP for partial support for this research, and Thaddeus D. Ladd, Kohei M. Itoh and Austin Fowler for technical guidance.

References

1. Grover, L.K.: Quantum telecomputation (April 1997), http://arXiv.org/quant-ph/9704012
2. Cirac, J.I., Ekert, A., Huelga, S.F., Macchiavello, C.: Distributed quantum computation over noisy channels. Physical Review A 59, 4249 (1999)
3. Cleve, R., Buhrman, H.: Substituting quantum entanglement for communication. Physical Review A 56(2), 1201–1204 (1997)
4. Buhrman, H., Röhrig, H.: Distributed Quantum Computing. In: Mathematical Foundations of Computer Science 2003, pp. 1–20. Springer, Heidelberg (2003)
5. Ekert, A.K.: Quantum cryptography based on Bell's theorem. Physical Review Letters 67(6), 661–663 (1991)
6. Ribordy, G., Brendel, J., Gautier, J.D., Gisin, N., Zbinden, H.: Long-distance entanglement-based quantum key distribution. Physical Review A 63(1), 12309 (2000)
7. Chuang, I.L.: Quantum algorithm for distributed clock synchronization. Physical Review Letters 85(9), 2006–2009 (2000)
8. Van Meter, R., Munro, W.J., Nemoto, K., Itoh, K.M.: Arithmetic on a distributed-memory quantum multicomputer. ACM Journal of Emerging Technologies in Computing Systems 3(4), 17 (2008)
9. Van Meter III, R.D.: Architecture of a Quantum Multicomputer Optimized for Shor's Factoring Algorithm. PhD thesis, Keio University (2006) arXiv:quant-ph/0607065
10. Clark, S.M., Fu, K.M.C., Ladd, T.D., Yamamoto, Y.: Quantum computers based on electron spins controlled by ultra-fast, off-resonant, single optical pulses. Physical Review Letters 99, 040501 (2007)
11. Oi, D.K.L., Devitt, S.J., Hollenberg, L.C.L.: Scalable error correction in distributed ion trap computers. Physical Review A 74, 052313 (2006)
12. Vedral, V., Barenco, A., Ekert, A.: Quantum networks for elementary arithmetic operations. Phys. Rev. A 54, 147–153 (1996), http://arXiv.org/quant-ph/9511018
13. Van Meter, R., Nemoto, K., Munro, W.J.: Communication links for distributed quantum computation. IEEE Transactions on Computers 56(12), 1643–1653 (2007)
14. Lenstra, A., Tromer, E., Shamir, A., Kortsmit, W., Dodson, B., Hughes, J., Leyland, P.: Factoring estimates for a 1024-bit RSA modulus. In: Laih, C.-S. (ed.) ASIACRYPT 2003. LNCS, vol. 2894, pp. 55–74. Springer, Heidelberg (2003)
15. Spiller, T.P., Nemoto, K., Braunstein, S.L., Munro, W.J., van Loock, P., Milburn, G.J.: Quantum computation by communication. New Journal of Physics 8, 30 (February 2006)
16. Shor, P.W.: Algorithms for quantum computation: Discrete logarithms and factoring. In: Proc. 35th Symposium on Foundations of Computer Science, pp. 124–134. IEEE Computer Society Press, Los Alamitos (1994)
17. Ekert, A., Jozsa, R.: Quantum computation and Shor's factoring algorithm. Review of Modern Physics 68(3), 733–753 (1996)
18. Kendon, V.M., Munro, W.J.: Entanglement and its role in Shor's algorithm. Quantum Information and Computation 6(7), 630–640 (2006)
19. Beckman, D., Chari, A.N., Devabhaktuni, S., Preskill, J.: Efficient networks for quantum factoring. Phys. Rev. A 54, 1034–1063 (1996), http://arXiv.org/quant-ph/9602016
20. Kunihiro, N.: Practical running time of factoring by quantum circuits. In: Proc. ERATO Conference on Quantum Information Science (EQIS 2003) (September 2003)
21. Van Meter, R., Itoh, K.M.: Fast quantum modular exponentiation. Physical Review A 71(5), 052320 (2005)

22. van Loock, P., Ladd, T.D., Sanaka, K., Yamaguchi, F., Nemoto, K., Munro, W.J., Yamamoto, Y.: Hybrid quantum repeater using bright coherent light. Physical Review Letters 96, 240501 (2006)
23. Gunn, C.: CMOS photonics for high-speed interconnects. IEEE Micro 26(2), 58–66 (2006)
24. Ladd, T.D.: private communication (February 2008)
25. Fowler, A.G.: Constructing arbitrary single-qubit fault-tolerant gates. quant-ph/0411206 (December 2005)
26. Jiang, L., Taylor, J.M., Sorensen, A.S., Lukin, M.D.: Scalable quantum networks based on few-qubit registers. quant-ph/0703029 (2007)
27. Kim, J., Kim, C.: Integrated Optical Approach to Trapped Ion Quantum Computation. eprint arXiv: 0711.3866 (2007)

Author Index