Available online at www.sciencedirect.com

SCIENCE @ DIRECT®

Information Processing Letters 97 (2006) 208–211

Information Processing Letters

www.elsevier.com/locate/ipl

Quantum lower bounds for the Goldreich–Levin problem

Mark Adcock [a], Richard Cleve [a,b], Kazuo Iwama [c,*], Raymond Putra [c,e], Shigeru Yamashita [d]

[a] Department of Computer Science, University of Calgary, Canada
[b] School of Computer Science and Institute for Quantum Computing, University of Waterloo, Canada
[c] Graduate School of Informatics, Kyoto University/QCI, ERATO, JST, Japan
[d] Nara Institute of Science and Technology, Japan
[e] School of Computer Science, McGill University, Canada

Received 3 June 2004; received in revised form 25 October 2004; accepted 10 January 2005

Available online 19 December 2005

Communicated by P.M.B. Vitányi

Abstract

At the heart of the Goldreich–Levin theorem is the problem of determining an n-bit string a by making queries to two oracles, referred to as IP (inner product) and EQ (equivalence). The IP oracle, on input x, returns a bit that is biased towards $a \cdot x$ (the modulo two inner product of a with x) in the following sense. For a random x, the probability that $\mathrm{IP}(x) = a \cdot x$ is at least $\frac{1}{2}(1+\varepsilon)$. The EQ oracle, on input x, returns a bit specifying whether or not $x = a$. It has been shown that a quantum algorithm can solve this problem with $\mathrm{O}(1/\varepsilon)$ IP and EQ queries, whereas any classical algorithm requires $\Omega(n/\varepsilon^2)$ such queries. Also, the quantum algorithm requires only $\mathrm{O}(n/\varepsilon)$ auxiliary one- and two-qubit gates in addition to its queries. We show that the above quantum algorithm is optimal in terms of both EQ and IP queries. Specifically, $\Omega(1/\varepsilon)$ EQ queries are necessary, and $\Omega(1/\varepsilon)$ IP queries are necessary if the number of EQ queries is $\mathrm{o}(\sqrt{2^n})$.

Keywords: Computational complexity; Quantum computing; Cryptography

1. Introduction and summary of results

The Goldreich–Levin theorem is a cryptographic reduction which enables a cryptographically hard predicate to be based on the computational difficulty of a one-way function [5]. It can be abstracted as the following problem, which we henceforth refer to as the *GL problem*. Let $a \in \{0,1\}^n$ and ε satisfy $0 < \varepsilon \leqslant 1$. Let information about a be available only from IP (inner product) and EQ (equivalence) oracle queries. The IP oracle has the property that, for a uniformly-distributed random $x \in \{0,1\}^n$, $\Pr[\mathrm{IP}(x) = a \cdot x] \geqslant \frac{1}{2}(1+\varepsilon)$. The EQ oracle, on input $x \in \{0,1\}^n$, returns a bit specifying whether or not $x = a$. The task is to determine a.

For an algorithm solving the GL problem, its efficiency corresponds to the overhead in the underlying cryptographic reduction. The more efficient an algorithm for the GL problem is, the tighter the correspon-

[*] Corresponding author.
E-mail addresses: mark.adcock@gdcanada.com (M. Adcock), cleve@cs.uwaterloo.ca (R. Cleve), iwama@kuis.kyoto-u.ac.jp (K. Iwama), raymond@kuis.kyoto-u.ac.jp (R. Putra), ger@is.aist-nara.ac.jp (S. Yamashita).

0020-0190/$ – see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.ipl.2005.01.016

dence is between the cryptographic primitives that it is applied to. Determining the most efficient algorithm for the GL problem is therefore a matter of interest in complexity-theory based cryptography in both classical and quantum frameworks (see, e.g., [1] for further discussion).

When there are no errors (i.e., $\varepsilon = 1$), it is straightforward to show that n queries are necessary and sufficient for any classical algorithm; however, with a quantum algorithm, one query suffices [4,10].

For smaller ε, Levin [8] shows how to solve the problem classically with $\mathrm{O}(n/\varepsilon^2)$ IP and EQ queries; however the approach requires superpolynomial (in n/ε) auxiliary operations. Goldreich and Levin [5] show how to solve this problem classically with a number of queries *and auxiliary operations* that is polynomial in n/ε, and this can be refined into an efficient algorithm that makes $\mathrm{O}(n/\varepsilon^2)$ IP queries followed by $\mathrm{O}(1/\varepsilon^2)$ EQ queries [9,6].

Adcock and Cleve [1] show that the classical IP query complexity for solving the GL problem with bounded-error probability is $\Omega(n/\varepsilon^2)$ whenever the number of EQ queries is at most $\sqrt{2^n}$ (for a reasonable range of values of ε). It can also be shown that $\Omega(1/\varepsilon^2)$ EQ queries are necessary classically.

For quantum algorithms, Adcock and Cleve [1] show that $\mathrm{O}(1/\varepsilon)$ IP queries, $\mathrm{O}(1/\varepsilon)$ EQ queries, and $\mathrm{O}(n/\varepsilon)$ auxiliary one- and two-qubit gates are sufficient to solve the GL problem; however, they do not address the question whether these costs are necessary. We address this question by showing the following.

Theorem 1. *Any quantum algorithm solving the GL problem with constant success probability requires $\Omega(1/\varepsilon)$ EQ queries, whenever $\varepsilon \geqslant (\frac{1}{2})^{n/2}$.*

It is not possible to lower bound the number of IP queries independently of the number of EQ queries, because $\mathrm{O}(\sqrt{2^n})$ EQ queries would eliminate the need for any IP queries [7]. The next theorem implies that, whenever the number of EQ queries is $\mathrm{o}(\sqrt{2^n})$, the number of IP queries must be $\Omega(1/\varepsilon)$.

Theorem 2. *Any quantum algorithm solving the GL problem with constant success probability requires either $\Omega(\sqrt{2^n})$ EQ queries or $\Omega(1/\varepsilon)$ IP queries, whenever $\varepsilon \geqslant (\frac{1}{2})^{n/2}$.*

For the quantum case, a query that, on input $x \in \{0,1\}^n$, returns one bit can be regarded as a unitary operation U, where the output bit is understood to be the last qubit of $U|x\rangle|0\rangle$. The stochastic property of IP queries is in terms of the measured result of the output qubit (see [1] for further discussion about formalizing quantum IP queries).

Our proof technique for the former theorem is by combining a lower bound arising in the list decoding of Hadamard codes (which we show explicitly), in conjunction with known lower bounds for quantum searching [2]. The latter theorem is proved by considering a special class of amplitude amplification problems that easily reduce to the GL problem and can be lower bounded by a standard hybrid argument.

2. Proof of Theorem 1

For any even k such that $0 < k \leqslant n$, define $f_k : \{0,1\}^n \to \{0,1\}$ as

$$f_k(x_1, x_2, \ldots, x_n) = x_1x_2 \oplus x_3x_4 \oplus \cdots \oplus x_{k-1}x_k.$$

Let $\varepsilon \geqslant (\frac{1}{2})^{n/2}$ be given, and set k to the unique even number such that $(\frac{1}{2})^{k/2+1} < \varepsilon \leqslant (\frac{1}{2})^{k/2}$. Now *fix* the IP oracle to $\mathrm{IP}(x) = f_k(x)$. Note that fixing the IP oracle makes all IP queries in the algorithm redundant. We will show that this particular setting of the IP oracle has the interesting property that there are $\Omega(1/\varepsilon^2)$ different $a \in \{0,1\}^n$ that are consistent with it in the sense that $\Pr_x[f_k(x) = a \cdot x] \geqslant \frac{1}{2}(1+\varepsilon)$. Since there are $\Omega(1/\varepsilon^2)$ candidates for the actual solution—which must be found using EQ queries—the well-known lower bound for searching [2] implies that the number of EQ queries necessary (for constant success probability) is $\Omega(\sqrt{1/\varepsilon^2}) = \Omega(1/\varepsilon)$.

We now provide the technical details of the proof, starting with the following simple lemma.

Lemma 3. *Let k be even and $x_1, \ldots, x_k$ be independent uniformly distributed random bits. Then*

$$\Pr[x_1x_2 \oplus \cdots \oplus x_{k-1}x_k = 0] = \tfrac{1}{2}\bigl(1 + \bigl(\tfrac{1}{2}\bigr)^{k/2}\bigr).$$

Proof. Define $Y = (-1)^{x_1x_2 \oplus \cdots \oplus x_{k-1}x_k}$. Then $\mathrm{E}[Y] = \mathrm{E}[(-1)^{x_1x_2}] \cdots \mathrm{E}[(-1)^{x_{k-1}x_k}] = (\frac{1}{2})^{k/2}$, from which it follows that $\Pr[x_1x_2 \oplus \cdots \oplus x_{k-1}x_k = 0] = \frac{1}{2}(1 + E[Y]) = \frac{1}{2}(1 + (\frac{1}{2})^{k/2})$. □

The following proposition provides a characterization of several $a \in \{0,1\}^n$ that are consistent with the IP oracle.

Proposition 4. *For all $a \in \{0,1\}^n$ such that $f_k(a) = 0$ and $a_{k+1} = a_{k+2} = \cdots = a_n = 0$, if $x \in \{0,1\}^n$ is randomly chosen then* $\Pr[f_k(x) = a \cdot x] \geqslant \frac{1}{2}(1+\varepsilon)$.

Proof.

$$\begin{aligned}
&\Pr\bigl[f_k(x) = a\cdot x\bigr]\\
&\quad = \Pr\bigl[(x_1x_2 \oplus \cdots \oplus x_{k-1}x_k)\\
&\qquad \oplus (a_1x_1 \oplus \cdots \oplus a_kx_k) = 0\bigr]\\
&\quad = \Pr\bigl[(x_1x_2 \oplus a_1x_1 \oplus a_2x_2) \oplus \cdots\\
&\qquad \oplus (x_{k-1}x_k \oplus a_{k-1}x_{k-1} \oplus a_kx_k) = 0\bigr]\\
&\quad = \Pr\bigl[(x_1 \oplus a_2)(x_2 \oplus a_1) \oplus \cdots\\
&\qquad \oplus (x_{k-1} \oplus a_k)(x_k \oplus a_{k-1}) \oplus f_k(a) = 0\bigr]\\
&\quad = \Pr\bigl[x_1x_2 \oplus \cdots \oplus x_{k-1}x_k = 0\bigr]\\
&\quad = \tfrac{1}{2}\bigl(1 + \bigl(\tfrac{1}{2}\bigr)^{k/2}\bigr) \quad \text{(by Lemma 3)}\\
&\quad \geqslant \tfrac{1}{2}(1+\varepsilon). \quad \square
\end{aligned}$$

The following proposition, in conjunction with Proposition 4, lower bounds the number of $a \in \{0,1\}^n$ that are consistent with the IP oracle.

Proposition 5. *The number of $a \in \{0,1\}^n$ such that $f_k(a) = 0$ and $a_{k+1} = a_{k+2} = \cdots = a_n = 0$ is at least $\frac{1}{8}(1/\varepsilon^2)$.*

Proof. Lemma 3 implies that the number of $a \in \{0,1\}^k$ such that $f_k(a) = 0$ is $\frac{1}{2}(1 + (\frac{1}{2})^{k/2})2^k = 2^{k-1} + 2^{k/2-1} > \frac{1}{8}2^{k+2} > \frac{1}{8}(1/\varepsilon^2)$. $\square$

3. Proof of Theorem 2

Let $\varepsilon > (\frac{1}{2})^{n/2}$ be given. For each $a \in \{0,1\}^n$ such that $a \neq 0$, define two oracles. The first is the aforementioned EQ oracle (that, on input $x \in \{0,1\}^n$, returns a bit specifying whether or not $x = a$). To define the second type of oracle, first define the unitary operation A acting on n qubits such that, for all $y \in \{0,1\}^n$,

$$A|y\rangle = \sqrt{1-\varepsilon^2}|y\rangle + \mathrm{i}\varepsilon|a \oplus y\rangle. \tag{1}$$

Note that $|\langle a|A|0\rangle| = \varepsilon$. The second type of query is a *controlled-A* operation, denoted as cont-A, where cont-$A|y\rangle|b\rangle = (A^b|y\rangle)|b\rangle$, for all $y \in \{0,1\}^n$ and $b \in \{0,1\}$.

Consider the following *amplitude amplification* problem. There is an unknown $a \in \{0,1\}^n$ such that $a \neq 0$. Information about a is available by EQ, cont-A, and cont-$A^\dagger$ queries. The goal is to determine a. The well-known amplitude amplification algorithm [3] solves this problem using $\mathrm{O}(1/\varepsilon)$ EQ, cont-A, and cont-$A^\dagger$ queries. We first show that this is optimal in the following sense.

Lemma 6. *The amplitude amplification problem requires either $\Omega(\sqrt{2^n})$ EQ queries or $\Omega(1/\varepsilon)$ cont-A or cont-$A^\dagger$ queries, whenever $\varepsilon \geqslant (\frac{1}{2})^{n/2}$.*

Proof. This is straightforward to prove by modifying the quantum lower bound for searching that uses the hybrid method [2]. That lower bound proof shows that there is a state $|\phi\rangle$ such that, if only t EQ queries are available, then, averaging over all values of a, the final state of the algorithm has distance only $t(2/\sqrt{2^n-1})$ from $|\phi\rangle$ (note that, since $a \neq 0$, the size of the search space is $2^n - 1$).

The present scenario is different in that cont-A and cont-$A^\dagger$ queries can be interleaved into the computation. This is addressed by showing that each cont-A and cont-$A^\dagger$ query can have a limited effect on a quantum state. The precise result is that, for *any* quantum state $|\psi\rangle$, $\||\psi\rangle - \text{cont-}A|\psi\rangle\| \leqslant \sqrt{2}\varepsilon$. This inequality can be proven by noting that the eigenvalues of cont-A are all either 1 or $\sqrt{1-\varepsilon^2} \pm \mathrm{i}\varepsilon$. Thus, each eigenvalue is distance at most $\sqrt{2}\varepsilon$ away from 1. It follows that, if there are s cont-A and cont-$A^\dagger$ queries and t EQ queries, then, averaging over all values of a, the final state of the algorithm has distance only $s(\sqrt{2}\varepsilon) + t(2/\sqrt{2^n-1})$ from $|\phi\rangle$, from which the result follows. $\square$

Next, we observe that a cont-A query can be used to simulate an IP query. The simulation is given by the circuit in Fig. 1, denoted as C, where H denotes the Hadamard gate and S is defined as $S|b\rangle = (-\mathrm{i})^b|b\rangle$, for $b \in \{0,1\}$.

Lemma 7. *If the last output qubit in the above circuit is measured then the probability that the outcome is $a\cdot x$ is $\frac{1}{2}(1+\varepsilon)$.*

Proof. It is sufficient to show that

$$\langle x, a\cdot x|C|x,0\rangle = \frac{1+\varepsilon-\mathrm{i}(-1)^{a\cdot x}\sqrt{1-\varepsilon^2}}{2}, \tag{2}$$

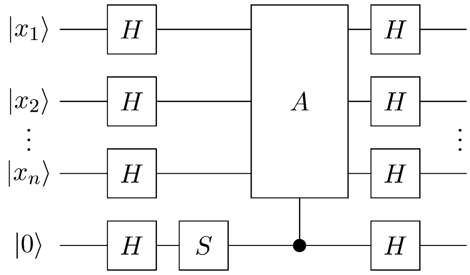

Fig. 1. Simulating an IP query using a cont-A query. The last qubit, when measured, is biased towards $a\cdot x$.

for all $x \in \{0,1\}^n$, since this implies that $|\langle x, a \cdot x|C|x,0\rangle|^2 = \frac{1}{2}(1+\varepsilon)$.

One way of establishing Eq. (2) is as follows. If circuit C is executed up to the stage of the cont-A gate on state $|x,0\rangle$, the resulting state is

$$\frac{1}{\sqrt{2}}\frac{1}{\sqrt{2^n}}\left(\sum_{y\in\{0,1\}^n}(-1)^{x\cdot y}|y\rangle\right)|0\rangle + \frac{1}{\sqrt{2}}\left(\frac{1}{\sqrt{2^n}}\sum_{y\in\{0,1\}^n}(-1)^{x\cdot y}\left(-\mathrm{i}\sqrt{1-\varepsilon^2} + (-1)^{a\cdot x}\varepsilon\right)|y\rangle\right)|1\rangle. \tag{3}$$

Also, if the last stage of circuit C is executed on state $|x, a\cdot x\rangle$, the resulting state is

$$\frac{1}{\sqrt{2}}\left(\frac{1}{\sqrt{2^n}}\sum_{y\in\{0,1\}^n}(-1)^{x\cdot y}|y\rangle\right)|0\rangle + \frac{1}{\sqrt{2}}\left(\frac{1}{\sqrt{2^n}}\sum_{y\in\{0,1\}^n}(-1)^{x\cdot y}(-1)^{a\cdot x}|y\rangle\right)|1\rangle. \tag{4}$$

Eq. (2) is obtained as the inner product between the states in Eqs. (3) and (4). □

Since Lemma 7 implies that a violation of Theorem 2 leads to a violation of Lemma 6, this completes the proof.

Acknowledgement

We are grateful to Venkatesan Guruswami and Ronald de Wolf for valuable comments. R.C. is partially supported by Canada's NSERC, MITACS, and CIAR.

References

[1] M. Adcock, R. Cleve, A quantum Goldreich–Levin theorem with cryptographic applications, in: H. Alt, A. Ferreira (Eds.), Proc. 19th Internat. Symp. on Theoretical Aspects of Computer Science (STACS 2002), in: Lecture Notes in Comput. Sci., vol. 2285, Springer-Verlag, Berlin, 2002, pp. 323–334.

[2] C. Bennett, E. Bernstein, G. Brassard, U. Vazirani, Strengths and weaknesses of quantum computing, SIAM J. Comput. 26 (5) (1997) 1510–1523.

[3] G. Brassard, P. Høyer, M. Mosca, A. Tapp, Quantum amplitude amplification and estimation, in: Quantum Computation and Quantum Information: A Millennium Volume, in: AMS Contemporary Mathematics Series, vol. 305, 2002.

[4] E. Bernstein, U. Vazirani, Quantum complexity theory, SIAM J. Comput. 26 (5) (1997) 1411–1473.

[5] O. Goldreich, L. Levin, Hard-core predicates for any one-way function, in: Proc. 21st Annual ACM Symp. on Theory of Computing (STOC 1989), 1989, pp. 25–32.

[6] O. Goldreich, Modern Cryptography, Probabilistic Proofs and Pseudorandomness, Springer, Berlin, 1999.

[7] L.K. Grover, A fast quantum mechanical algorithm for database search, in: Proc. 28th Annual ACM Symp. on Theory of Computing (STOC 1996), 1996, pp. 212–219.

[8] L.A. Levin, One-way functions and pseudorandom generators, Combinatorica 7 (4) (1987) 357–363.

[9] L.A. Levin, Randomness and non-determinism, J. Symbolic Logic 58 (3) (1993) 1102–1103.

[10] B. Terhal, J. Smolin, Single quantum querying of a database, Phys. Rev. A 58 (3) (1998) 1822–1826.