Invited Paper

Quantum Biased Oracles

Kazuo Iwama,[†1,†2] Akinori Kawachi[†3] and Shigeru Yamashita[†4]

This paper reviews researches on quantum oracle computations when oracles are not perfect, i.e., they may return wrong answers. We call such oracles *biased* oracles, and discuss the formal model of them. Then we provide an intuitive explanation how quantum search with biased oracles by Høyer, et al. (2003) works. We also review the method, by Buhrman, et al. (2005), to obtain all the answers of a quantum biased oracle without any overhead compared to the perfect oracle case. Moreover, we discuss two special cases of quantum biased oracles and their interesting properties, which are not found in the classical corresponding cases. Our discussion implies that the model of quantum biased oracle adopted by the existing researches is natural.

1. Introduction

The researches on *quantum oracle computation* have intensively studied in the quantum research community. One of the main reasons may be the following: in most cases of classical oracle computations, the lower bound of computational complexity is linear to the problem size, therefore, it is relatively easy to exploit the difference between the power of classical and quantum computations in the oracle setting.

The classical oracle computation is the following scenario: we want to compute a designated Boolean function $f(x_1, x_2, \ldots, x_N)$, but the value of a_i (= 0 or 1) of each x_i can be obtained only by making a query to a black box called an *oracle*. Then, we often consider the smallest number of necessary oracle calls, which we call the *query complexity*, to obtain the value of $f(a_1, a_2, \ldots, a_N)$ with a high (say, constant) probability. Suppose we want to compute the Boolean OR of input variables, i.e., $f = x_1 \vee x_2 \vee \ldots \vee x_N$. In the case of classical computation, we need $\Theta(N)$ queries to compute the function.

By contrast, we need much fewer queries in the case of quantum computation. For instance, we need only $O(\sqrt{N})$ queries to find an index i such that $a_i = 1$ (Grover Search [18]). This is one of the major examples of quantum superiority. Therefore, *quantum query complexity* has been intensively studied as a central issue of quantum computation. Indeed, there have been a number of applications and extensions of Grover Search, e.g., [8),9),14),19),20)]. Also quite many results on efficient quantum algorithms are shown by sophisticated ways of using Grover Search. Brassard, et al. [11] showed a quantum counting algorithm that gives an approximate counting method by combining the Grover search with the quantum Fourier transformation. Quantum algorithms for the claw-finding and the element distinctness problems given by Buhrman, et al. [10] also exploited classical random and sorting methods with Grover Search. (Ambainis,[6] developed an optimal quantum algorithm with $O(N^{2/3})$ queries for element distinctness problem, which makes use of quantum walk and matches to the lower bounds shown by Shi[27]). Ambainis and Aaronson[2] constructed quantum search algorithms for spatial regions by combining Grover Search with the divided-and-conquer method. Magniez, Santha and Szegedy [24] showed efficient quantum algorithms to find a triangle in a given graph by using combinatorical techniques with Grover Search. Dürr, Heiligman, Høyer and Mhalla [15] also investigated quantum query complexity of several graph-theoretic problems. In particular, they exploited Grover Search on some data structures of graphs for their upper bounds. Ambainis, et al. [1] studied the query complexity of the most general problem what they call the oracle identification problem (OIP). An OIP is given a set S of M Boolean oracles out of 2^N ones, to determine which oracle in S is the current black box oracle. We can exploit the information that candidates of the current oracle is restricted to S. They provide almost an optimal algorithm whose query complexity is $O(\sqrt{N \log M \log N} \log\log M)$.

For oracle computation, there are several sit-

†1 Kyoto University
†2 Japan Science and Technology Corporation
†3 Tokyo Institute of Technology
†4 Nara Institute of Science and Technology

uations where we can get only a *noisy* Boolean value for each variable. Suppose again that we want to compute the Boolean OR of input variables, i.e., $f = x_1 \vee x_2 \vee \ldots \vee x_N$ by asking an input oracle. However, this time, the oracles is *noisy* in the sense that it returns us the correct a_i ($= 0$ or 1) with probability $\frac{1}{2} + \epsilon$. In this paper, we call this oracle an *ϵ-biased oracle*. For the above particular example, one simple algorithm is to call the biased oracle for each x_i many times and to guess the value of a_i by majority. It is not hard to see that we need $\Omega(\frac{1}{\epsilon^2})$ oracle calls to know the correct value of each a_i with constant probability. Thus, the query complexity obviously depends on the value of ϵ. Note that many studies assume that ϵ is a constant, which disappears in the query complexity under the big-O notation [12),28)]. Note also that we can get all the values of N bit with high probability by querying each a_i $O(\log N)$ times instead of once. Thus, we can make any algorithm *robust*, i.e., resilient against biased oracles at the cost of an $O(\log N)$-factor overhead. In some cases, this factor of $O(\log N)$ is actually needed: Feige, et al. [16)] proved that any classical *robust* algorithm to compute the parity of the N bits needs $\Omega(N \log N)$ queries. On the other hand, the same paper also gives a non-trivial classical algorithm which computes OR of the N bits with $O(N)$ queries.

Recently, two papers, by Høyer, et al. [21)] and Buhrman, et al. [13)], raised the question of how to cope with biased oracles in the quantum case. For the quantum setting, both papers [13),21)] are based on the following model: the oracle returns, for the query to bit a_i, a quantum pure state from which we can measure the correct value of a_i with a constant probability. This noise model naturally fits the motivation that a similar mechanism should apply when we use bounded-error quantum subroutines.

Based on the above biased oracle model, Høyer, et al. gave a quantum algorithm that robustly executes Grover Search with $O(\sqrt{N})$ queries, which is only a constant factor worse than the perfect oracle case [21)]. Buhrman, et al. [13)] also adopted the same model and gave a robust quantum algorithm to output all the N bits by using $O(N)$ queries. This obviously implies that $O(N)$ queries are enough to compute the parity of the N bits, which contrasts with the classical $\Omega(N \log N)$ lower bound mentioned earlier. Thus, robust quantum computation does not need a serious overhead at least for several important problems.

In this paper, we introduce the formal model of quantum biased oracles appeared in the existing researches [3),13),21),22)], and mainly review the two robust quantum algorithm mentioned above. We also discuss special cases of quantum biased oracles. This paper is organized as follows. In Section 2, we define the model of quantum biased oracles discussed in computer science community. Then we introduce two interesting researches based on the model in Sections 3 and 4. In Section 5, we discuss other models of quantum biased oracles. Finally, Section 6 concludes this paper.

2. Quantum Biased Oracle Models

2.1 Classical Biased Oracles

We start with the classical model of biased oracles.

Definition 1 A classical ϵ-biased oracle is defined to return a_i with probability $\frac{1}{2} + \epsilon$ when its input is $i (1 \leq i \leq N)$.

2.2 Quantum Biased Oracles

From Definition 1, it is natural to consider a quantum oracle to be defined as a quantum black box algorithm such that the measured value of the answer qubit produced by the algorithm is a_i with probability $\frac{1}{2} + \epsilon$ when its input is i. Therefore, the most general model of a quantum biased oracle is such that the quantum state generated by the oracle is a *mixed state* from which we can measure the correct value with probability $\frac{1}{2} + \epsilon$. This model can deal with decoherence errors. Indeed there are researches that study the effect of decoherence error on Grover Search by using this error model [25),26)].

In the computer science community, we usually consider that the quantum state generated by a quantum biased oracle is a *pure state*. In other words, a biased oracle is considered to be a unitary transformation. Recently there have been many researches [3),13),21)~23)] based on this model. If we consider a quantum subroutine as an oracle, the oracle can be considered as this model, therefore, the motivation of this model is also natural. Adcock and Cleve firstly discussed quantum biased oracles of this model [3)], and their definition can be written as follows.

Definition 2 A quantum ϵ-biased oracle is a unitary transform (denoted by O_ϵ hereafter) on $n+m+1$ qubits which satisfies the following two properties.

(1) If the last qubit of $O_\epsilon |i\rangle |0^m\rangle |0\rangle$ is measured, yielding the value $w \in \{0, 1\}$, then

$Pr[w = a_i] \geq \frac{1}{2} + \epsilon$ for any $i \in \{0,1\}^n$.

(2) For any $i \in \{0,1\}^n$ and $y \in \{0,1\}^{m+1}$, the state of the first n qubits of $O_\epsilon |i\rangle |y\rangle$ is $|i\rangle$. For simplicity, we just assume $N = 2^n$ in the rest of the paper. (Otherwise we consider an oracle whose input size is $N' = 2^n(2N > N' > N)$ by adding some dummy inputs. It is obvious that this does not change the query complexity in the big-O notation.)

The second property is for technical convenience, and any unitary operation without this property can be converted to one that has this property, by first producing a copy of the classical basis state $|i\rangle$. Note that we use the bias (ϵ in the definition) of the success probability from 1/2 to denote the parameter for a biased oracle. However, some papers use the error probability $(1/2-\epsilon)$ for the purpose. That is only the deference of notations, but for a simpler expressions in Section 4, we will use the error probability, which is denoted by δ, instead of the bias rate ϵ. In Section 4, we denote the biased oracles O_δ in stead of O_ϵ to avoid confusion.

Since O_ϵ is a unitary transform, $O_\epsilon |i\rangle |0^m\rangle |0\rangle$ must be written as

$$|i\rangle (\alpha_i |v_i\rangle |a_i\rangle + \beta_i |w_i\rangle |\overline{a_i}\rangle).$$

We consider that v_i, w_i and α_i are generally different according to i. In the classical case, we usually do not care such a condition since it seems almost impossible to utilize this kind of information usefully. However, in the quantum case, this condition should be very important. In Section 5, we will see the reason why we consider the condition important in the quantum case.

3. Quantum Search with Biased Oracles

Høyer, Mosca, and de Wolf showed a quantum search algorithm with biased oracles [21]. Their algorithm can find a *solution*, i.e., an index j such that $a_j = 1$ from N indices with high probability using $O(\sqrt{N})$ queries to a biased oracle (with a constant bias), which requires only constant overhead compared to the perfect oracle. The following quantum biased oracle is given in their algorithm:

$$O_\delta |i\rangle |0^m\rangle |0\rangle = \sqrt{p_i} |i\rangle |v_i\rangle |a_i\rangle + \sqrt{1-p_i^2} |i\rangle |w_i\rangle |\overline{a_i}\rangle,$$

where $p_i \geq 9/10$ for every i. In this section, we describe how their algorithm finds a solution with high probability using the biased oracle.

3.1 Algorithms

We first describe two ingredients to construct the main algorithm.

Lemma 1 (Amplitude Amplification [11]) Let S_0 be the unitary operator that inverts the sign of the amplitude of the all-zero state and S_1 be the unitary operator that inverts the signs of the amplitudes of all basis states whose last qubit is $|1\rangle$. Let $A|0\rangle = \sin\theta |\phi_1\rangle |1\rangle + \cos\theta |\phi_0\rangle |0\rangle$, where $0 \leq \theta \leq \pi/2$. When $G = -AS_0A^{-1}S_1$, we then have $GA|0\rangle = \sin 3\theta |\phi_1\rangle |1\rangle + \cos 3\theta |\phi_0\rangle |0\rangle$.

Lemma 2 (Error Reduction) Let $A|0\rangle = \sqrt{p}|\phi_b\rangle|b\rangle + \sqrt{1-p}|\phi_{1-b}\rangle|1-b\rangle$, where $b \in \{0,1\}$ and $p \geq 9/10$. Then, we can construct a unitary operator E such that $E|0\rangle = \sqrt{1-q}|\psi_b\rangle|b\rangle + \sqrt{q}|\psi_{1-b}\rangle|1-b\rangle$ by $O(\log(1/q))$ applications of A and majority-voting, where $|\psi_b\rangle$ and $|\psi_{1-b}\rangle$ are larger space for extra workspace than $|\phi_b\rangle$ and $|\phi_{1-b}\rangle$, respectively.

We exploit the above two procedures recursively to build the main algorithm of the robust quantum search algorithm. The main algorithm consists of a number of rounds. Assume that we have the following unitary operator A_k in the k-th round:

$$A_k|0\rangle = \alpha_k|\Gamma_k\rangle|1\rangle + \beta_k|\overline{\Gamma_k}\rangle|1\rangle + \sqrt{1-\alpha_k^2-\beta_k^2}|H_k\rangle|0\rangle,$$

where α_k and β_k are non-negative reals, $|\Gamma_k\rangle$ is a unit vector whose first register only contains $j \in \{i : a_i = 1\}$, $|\overline{\Gamma_k}\rangle$ is a unit vector whose first register only contains $j \notin \{i : a_i = 1\}$, and H_k is a unit vector. Here, A_1 is a unitary operator applying the given biased oracle O_δ to the state $(1/\sqrt{N})\sum_i |i\rangle|0\rangle|0\rangle$. Then, we build a unitary operator A_{k+1} for the next round by setting

$$A_{k+1} = E_kG_kA_k,$$

where E_k and G_k are unitary operators for the error reduction step and the amplitude amplification step, respectively. In more details, by Lemma 1, the amplitude amplification step is defined as

$$G_k = -A_kS_0A_k^{-1}S_1.$$

and by Lemma 2, the error reduction step consists of majority voting on $O(k)$ runs of the O_δ for all superposed j to decide whether $a_j = 1$ with error at most $1/2^{k+5}$:

$$E_k|j\rangle|1\rangle|0\rangle = p_{jk}|j\rangle|1\rangle|1\rangle + q_{jk}|j\rangle|1\rangle|0\rangle$$
$$E_k|j\rangle|0\rangle|0\rangle = |j\rangle|0\rangle|0\rangle,$$

where $q_{jk} = \sqrt{1-p_{jk}^2}$, $p_{jk}^2 \geq 1-1/2^{k+5}$ if $f_j = 1$ and $p_{jk}^2 \leq 1/2^{k+5}$ otherwise. (Note that we omit the second registers for workspace in the above equations.)

We now describe the main algorithm as follows.

Robust Quantum Search

(1) For $m = 0$ to $\lceil \log_9 N \rceil - 1$ do:
  (a) Run A_m 1,000 times (any large constant times is fine).
  (b) Verify the 1,000 measurement results by $O(\log N)$ times of the corresponding O_δ for each result.
  (c) If a solution has been found, then output the solution and stop this procedure.

(2) Output 'no solution'.

The following theorem holds for the above algorithm.

Theorem 1 The algorithm **Robust Quantum Search** can find a solution j such that $f_j = 1$ with a constant probability if one exists. Moreover, this outputs 'no solution' with a constant probability if no solutions exist. The query complexity is $O(\sqrt{N})$.

3.2 Intuitive Analysis

We compare the behaviors for biased and perfect oracles to understand a nature of the quantum search algorithm. Given a perfect oracle, one can see easily that the above quantum search algorithm defined recursively works similarly to Grover's quantum search algorithm. Now we focus on the k-th round of the recursion. From the definitions of G_k and A_k, we have

$$\begin{aligned} G_k A_k |0\rangle = &\sin 3\theta_k \frac{\alpha_k}{\sin\theta_k} |\Gamma_k\rangle |1\rangle \\ &+ \sin 3\theta_k \frac{\beta_k}{\sin\theta_k} |\overline{\Gamma_k}\rangle |1\rangle \\ &+ \cos 3\theta_k \frac{\sqrt{1-\alpha_k^2-\beta_k^2}}{\cos\theta_k} |H_k\rangle |0\rangle \end{aligned}$$

for a biased oracle, where $\alpha_k^2 + \beta_k^2 = \sin^2\theta_k$. Note that for the perfect oracle ($\beta_k = 0$) we have

$$\begin{aligned} G_k A_k |0\rangle = &\sin 3\theta_k \frac{\alpha_k}{\sin\theta_k} |\Gamma_k\rangle |1\rangle \\ &+ \cos 3\theta_k \frac{\sqrt{1-\alpha_k^2}}{\cos\theta_k} |H_k\rangle |0\rangle , \end{aligned}$$

where $\alpha_k = \sin\theta_k$. Comparing these two cases, if we want to amplify the success probability at each round using the biased oracle equivalently to the perfect oracle up to a constant factor, one can see that we should satisfy the following two conditions for any k: (i) $\alpha_k/\sin\theta_k$ is at least a constant and (ii) $\beta_k/\sin\theta_k$ is bounded by a small constant. (Note that $\alpha_k/\sin\theta_k = 1$ and $\beta_k/\sin\theta_k = 0$ in the perfect oracle.) If these conditions are met for any k, we can obtain a solution by the biased oracle with a constant probability similarly to the perfect oracle. To achieve this task, we perform the error reduction step to decrease the amplitude of the false positive state $|\overline{\Gamma}_k\rangle |1\rangle$. However, the error reduction step decreases not only the amplitude of the false positive state but also the amplitude of the true positive state $|\Gamma_k\rangle |1\rangle$. We therefore have to decrease the false positive amplitude as preserving the true positive one.

Intuitively, the false positive amplitude β_k becomes smaller as performing more queries in the error reduction step. We thus need the appropriate number of queries for error reduction step. Then, it should be noted that when k is small $\sin\theta_k$ is also small. This fact implies that we can save the number of the queries in the error reduction step for small k.

We now show that the above two conditions are met by the error reduction step with $O(k)$ queries. By the error reduction step, we obtain

$$\alpha_{k+1} \geq \sin 3\theta_k \frac{\alpha_k}{\sin\theta_k} \sqrt{1-2^{-(k+5)}}.$$

Since $\sin\theta_{k+1}$ is defined as $\sin 3\theta_k$,

$$\frac{\alpha_{k+1}}{\sin\theta_{k+1}} \geq \frac{\alpha_k}{\sin\theta_k} \sqrt{1-2^{-(k+5)}}.$$

We also have

$$\prod_{l=1}^{\infty} \sqrt{1-2^{-(l+5)}} = \Omega(1).$$

It follows that the condition (i) is met for any k by the error reduction with $O(k)$ queries.

On the other hand, we also have

$$\frac{\beta_{k+1}}{\sin\theta_{k+1}} \leq \frac{\beta_k}{\sin\theta_k} \sqrt{2^{-(k+5)}}.$$

One can easily verify that this value is sufficiently small to satisfy the condition (ii).

We next compare the query complexity of the two cases. Let C_k be the query complexity of A_k. If the given oracle is perfect, we obtain a recurrence $C_{k+1} = 3C_k$, which provides the total query complexity $O(\sqrt{N})$. In the case of the biased oracles, since the error reduction step just contributes an additive factor $O(k)$ queries for every k, we have $C_{k+1} = 3C_k + O(k)$, which also provides the total query complexity $O(\sqrt{N})$.

Note that the number of oracle calls for er-

ror reduction between the two oracle calls for the amplitude amplification does not increase linearly, but changes as follows: the number is increased by the number of rounds (k in the above discussion) at the end of every round in the algorithm. Since the oracle calls for the k-th round becomes three times as the $(k-1)$-th round, the number of oracle calls for error reduction is increased periodically such that the period becomes three times as long as the previous period. In other words, we do not need to increase the number of oracle calls for error reduction linearly to the number of oracle calls for the amplitude amplification, and thus we can obtain the desired query complexity.

4. Recovering All the Values of Quantum Biased Oracles

The quantum search algorithm shown in the previous section can find an index i such that $a_i = 1$ if one exists, and output 'no solutions' otherwise. Therefore, it can compute the OR function $f(a_1, ..., a_N) = a_1 \vee \cdots \vee a_N$ from biased inputs. We next review a generalization of the quantum search algorithm, shown by Burhman, et al. [13]. Their algorithm can compute any Boolean function from biased N inputs with $O(N)$ queries. Their algorithm basically *recovers* the correct inputs $(a_1, ..., a_N)$ from biased ones using the robust quantum search mentioned in the previous section.

4.1 Algorithms

We denote a unitary operator A_i, directly obtained by inputting an index i to a biased oracle, as

$$A_i |0^m\rangle |0\rangle = \alpha_i \left|\phi_i^0\right\rangle |0\rangle + \beta_i \left|\phi_i^1\right\rangle |1\rangle .$$

Let $|\alpha_i|^2 = 1 - \delta$ if $a_i = 0$ and otherwise $|\alpha_i|^2 = \delta$. That is, δ means the error probability, and we use δ (in stead of ϵ like the previous sections) in this section for easy notations. We also utilize

$$\overline{A_i} |0^m\rangle |0\rangle = \alpha_i \left|\phi_i^0\right\rangle |1\rangle + \beta_i \left|\phi_i^1\right\rangle |0\rangle$$

by applying the NOT operation to the second register. We now define a unitary operator

$$A_i(b) = \begin{cases} A_i & \text{if } b = 0, \\ \overline{A_i} & \text{if } b = 1, \end{cases}$$

Then, we can implement a unitary operation $A(x)$ for $x = (x_1, ..., x_N) \in \{0,1\}^N$ as

$$A(x) |i\rangle |0^m\rangle |0\rangle = |i\rangle A_i(x_i) |0^m\rangle |0\rangle$$

from the above modifications. Their algorithm holds $(x_1, ..., x_N)$ as temporary inputs and approaches $(x_1, ..., x_N)$ into $(a_1, ..., a_N)$ gradually. One can easily see that if $x_i = a_i$ then we get 0 from the result obtained by the application of $A_i(x_i)$. Then, their algorithm finds an index i such that $A_i(x_i)$ outputs 1 by the robust quantum search and update x by fixing x_i. Therefore, by approaching the output of $A_i(x_i)$ into 0 for all i, we can recovery all the correct inputs $(a_1, ..., a_N)$.

The heart of their algorithm is how to exploit the robust quantum search shown in the previous section. If we consider a more general biased oracle O_δ instead of a constant one like Theorem 1, the query complexity becomes $O(\sqrt{N}/(1-\delta)^2)$. Also, we can amplify the probability detecting the no-solution case by majority voting. The following lemma, stated in Ref. 13), summarizes this generalization.

Lemma 3 We can build a quantum algorithm RobustFind$(x, \beta, \gamma, \lambda)$ using $A(x)$: If $|x| \geq \beta N$, it outputs an index i such that $x_i = 1$ with at least probability $1 - \gamma$ and outputs 'no solutions' with probability at most λ. The query complexity of this algorithm is

$$O\left(\frac{1}{(1/2-\delta)^2\sqrt{\beta}} \log \frac{1}{\lambda\gamma}\right).$$

Buhrman, et al. showed the following theorem by building a quantum algorithm using the algorithm in the above lemma.

Theorem 2 Given a biased oracle O_δ, we can build a quantum algorithm that recovers $(a_1, ..., a_N)$ with probability at least 2/3 using $O(N/(1/2-\delta)^2)$ queries.

We now describe the main algorithm, which consists of two stages.

Recovering All the Values

[First Stage]

(1) For every i, run A_i and substitute the result of A_i into $\tilde{x_i}$.

(2) Run (a)-(b) from $k = 1$ to $\lceil \log(\delta(\log n)^2) \rceil$.

(a) Update $\delta' \leftarrow \delta/2^{k-1}$.

(b) Repeat (i)-(iii) $\lceil 1.7\delta' \rceil$ times.

(i) Call RobustFind$(\tilde{x}, 0.3\delta', 1/8, 1/8)$.

(ii) If an index i is found then update $\tilde{x}_i \leftarrow 1 - \tilde{x}_i$.

[Second Stage]

(1) Repeat (a)-(b) until the result of RobustFind is 'no solutions'

(a) Call RobustFind$(\tilde{x}, 1/\log^2 N, 1/10N, 1/10N)$.

(b) If an index i is found, update $\tilde{x}_i \leftarrow 1 - \tilde{x}_i$.

(2) Output $\tilde{x}$.

This algorithm roughly recovers the inputs $(a_1, ..., a_N)$ in the first stage. More precisely, this algorithm reduces the number of wrong in-

puts to at most $N/(\log N)^2$ in the first stage. In each round of the step (2), we obtain wrong inputs by the robust quantum search and fix them. Note that we may obtain correct inputs and flip them incorrectly with small probability. So, we repeat this procedure as improving the precision δ'.

The algorithm next precisely recovers the remaining wrong inputs in the second stage. Since the number of the wrong inputs is at most $N/(\log N)^2$ at the beginning of the second stage, we can consume $O(\log N)$ queries for the error reduction. Then this algorithm achieves the overall query complexity $O(N)$.

5. Quantum Biased Oracles with Special Conditions

In this section, we consider two special cases of quantum biased oracles where we relax the conditions in Definition 2. Note that our relaxations seem to be fair from the view point of classical computation, i.e., it seems almost impossible to utilize the relaxation in the classical case. However, as we will see, the query complexity changes dramatically in the quantum world.

5.1 Quantum Biased Oracles with the Same Bias Rate

The paper [23] discusses the case where the bias rate ϵ is the same for all inputs. In the classical case this condition does not alter the query complexity, however, does alter in the quantum case. Formally, we can have the following theorem.

Theorem 3 Let Q be any quantum algorithm solving some problem with probability p using a perfect oracle O with T number of queries. Then, there exists an algorithm Q' solving the same problem with probability at least $(1-1/T)^2 2p/3$ using an ϵ-biased oracle O_ϵ with $O(\frac{T}{\epsilon})$ number of queries, if $\alpha_i^2 \geq \frac{1}{2}+\epsilon$ for all i.

noindent Proof. First we construct the following oracle, $\tilde{O}_\epsilon$, by using one O_ϵ and one $O_\epsilon^\dagger$ as shown in **Fig. 1**. In the figure, X denotes a NOT gate, and Z denotes a controlled-Z gate. By simple calculation, it can be shown that

$$\tilde{O}_\epsilon |i, 0^m, 0\rangle = (-1)^{a_i} 2\epsilon |i, 0^m, 0\rangle + |i, \psi_i\rangle ,$$

where $|i, \psi_i\rangle$ is perpendicular to $|i, 0^m, 0\rangle$ and its norm is $\sqrt{1-4\epsilon^2}$.

Note that, if the oracle is perfect, i.e., $\epsilon = 1/2$, then it works as follows:

$$\tilde{O}_\epsilon |i, 0^m, 0\rangle = (-1)^{a_i} |i, 0^m, 0\rangle .$$

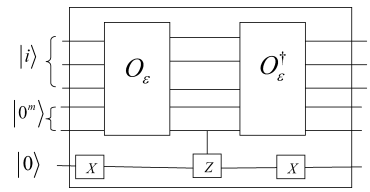

Fig. 1 The construction of $\tilde{O}_\epsilon$ oracle.

In other words, we consider that the state $(-1)^{a_i} |i, 0^m, 0\rangle$ is a *good state* in the sense that it is obtained by the perfect oracle.

The overall strategy is to construct a quantum state almost equivalent to the good state for each perfect oracle call in the original algorithm Q so that the overall success probability is only constant factor worse than the original one. To do so, first we estimate the value of θ_a such that $\sin^2\theta_a = (2\epsilon)^2$. More precisely, we find $\tilde{\theta_a}$ with probability at least $\frac{2}{3}$ s.t. $|\theta_a - \tilde{\theta_a}| \leq \frac{\theta_a}{(\pi+1)T}$ by using the biased oracle and its inverse in $O(\frac{T}{\epsilon})$. This can be done by using a similar scheme to the quantum counting method [11].

Next we perform quantum amplitude amplification method with the estimated value to amplify the amplitude of the good state. By simple calculation, we can show the following: if a $\tilde{\theta_a}$ is given s.t. $|\theta_a - \tilde{\theta_a}| \leq \frac{\theta_a}{(\pi+1)T}$, then the probability to measure the good state after quantum amplitude amplification method with $O(\frac{1}{\epsilon})$ biased oracle calls is at least $\left(1-\frac{1}{T^2}\right)$.

Now the theorem is immediate: each simulation of one query to the perfect oracle results in the success probability at least $1-\frac{1}{T^2}$. Since for all $t, n \in R$ s.t. $n \geq 1$ and $|t| \leq n$, $\left(1+\frac{t}{n}\right)^n \geq e^t\left(1-\frac{t^2}{n}\right)$, the overall success probability is $\left(1-\frac{1}{T^2}\right)^T 2p/3 \geq \left(1-\frac{1}{T}\right)^2 2p/3$ and the overall complexity is $O(\frac{T}{\epsilon})$. □

5.2 Quantum Biased Oracles with Resettable Condition

In addition to the relaxation mentioned in the previous section, if the quantum biased oracle does not have a work space, it is essentially the same as the perfect oracle. That is, although a work space does not matter in the classical case, we cannot ignore the work space in Definition 2 for the model of the quantum biased oracles.

To discuss the above matter, we introduce the following special quantum ϵ-biased oracle.

Definition 3 The following quantum ϵ-

biased oracle is called a *resettable* biased oracle.

$$O_\epsilon |i\rangle |0^m\rangle |0\rangle = |i\rangle |0^m\rangle (\alpha |a_i\rangle + \beta |\overline{a_i}\rangle),$$

where $\alpha = \sqrt{\frac{1}{2}+\epsilon}$ and $\beta = \sqrt{\frac{1}{2}-\epsilon}$.

The above oracle is essentially the same as the following one. (It is easy to verify that $\tilde{O}_\epsilon$ can be constructed by O_ϵ and two Hadamard gates.)

$$\tilde{O}_\epsilon |i\rangle |0\rangle = |i\rangle ((-1)^{a_i}\alpha |0\rangle + \beta |1\rangle), \quad (1)$$

$$\tilde{O}_\epsilon |i\rangle |1\rangle = |i\rangle (\alpha |1\rangle - (-1)^{a_i}\beta |0\rangle), \quad (2)$$

where $\alpha = \sqrt{\frac{1}{2}+\epsilon}$ and $\beta = \sqrt{\frac{1}{2}-\epsilon}$.

Let V be any perfect quantum oracle which maps $|i,b,z\rangle$ to $(-1)^{b\cdot a_i}|i,b,z\rangle$, where $i \in \{0,1\}^n$ and z be any qubit strings. Note that V is the standard definition for perfect oracles which often appears in the literature [4),5),18)].

Theorem 4 If there exists a quantum algorithm A solving some problem with probability $1-\delta$ by querying V T times, then instead of querying V, A can solve the same problem with probability $1-\delta$ by querying O_ϵ $O(T)$ times, where O_ϵ is a resettable biased oracle for V.

noindent Proof. For simplicity, we omit the description of z since it is left unchanged by the oracle transformation. Suppose that we have a quantum state $|\psi\rangle = \sum_i \gamma_i |i\rangle |0\rangle$ at some moment of the algorithm, where $\sum_i |\gamma_i|^2 = 1$. Then it follows that applying oracle O_ϵ to this $|\psi\rangle$ results in $O_\epsilon \sum_i \gamma_i |i\rangle |0\rangle = \sum_i (-1)^{a_i}\alpha\gamma_i |i\rangle |0\rangle + \sum_i \beta\gamma_i |i\rangle |1\rangle$.

Now here comes our key technique, namely, to use a measurement: if the measurement on the last qubit results in the state $|0\rangle$, we know that the quantum state after this measurement is exactly the same as the quantum state after calling V. Otherwise, if the state $|1\rangle$ is measured, we simply need to flip the last qubit to 0 and repeat querying O_ϵ since the previous state $|\psi\rangle$ is completely preserved. Note that the expected number of iteration is constant. Thus, A can query O_ϵ instead of V and the query complexity is roughly the same. □

The two relaxations discussed in this section alter the query complexities unlike the classical case. Thus, Definition 2 expresses a necessary condition for a proper model of the quantum biased oracle.

6. Concluding Remarks

In this paper we have discussed what is a quantum biased oracle, and summarized the related results known so far. It should be noted that the model of a quantum biased oracle has nothing to do with a *physical error* unlike the classical case. However, the model naturally fits probabilistic quantum algorithms; the model fits the cases where we consider to use some quantum algorithms as oracles. Of course, it should be interesting to study the properties of this model to investigate the power of quantum computation, even though having no relation to a physical noise.

As mentioned in Section 5, we should be very careful to consider some special cases of quantum biased oracles. In other words, there are much difference between classical and quantum biased oracle, i.e., some properties that seem to be natural in the classical case are not natural in the quantum case.

For the case of computing OR and retrieving all the values of oracles, the query complexity is exactly the same as the perfect oracle cases (when we consider the bias rate ϵ is a constant) as mentioned in this paper. Therefore, it is conjectured that the query complexity does not change by the error of quantum biased oracles for *all* the cases. Very recently, for the OIP problem, the query complexity of biased oracles matches to the lower bonds of the perfect oracle case for *most* cases [22)]. This result may support the above conjecture. To prove or disprove the above conjecture for all the cases may be the most interesting open problem.

References

1) Ambainis, A., Iwama, K., Kawachi, A., Masuda, H., Putra, R.H. and Yamashita, S.: Quantum identification of boolean oracles, In *Proc. 21st Annual Symposium on Theoretical Aspects of Computer Science*, LNCS 2996, pp.105–116 (2004).
2) Aaronson, S. and Ambainis, A.: Quantum search of spatial regions, *Proc. 44th Symposium on Foundations of Computer Science*, pp.200–209 (2003).
3) Adcock, M. and Cleve, R.: A quantum Goldreich-Levin theorem with cryptographic applications, *Proc. 19th Annual Symposium on Theoretical Aspects of Computer Science*, LNCS 2285, pp.323–334 (2002).
4) Ambainis, A.: Quantum lower bounds by quantum arguments, *Journal of Computer and System Sciences*, vol.64, pp.750–767 (2002).
5) Barnum, H., and Saks, M.: A lower bound on the quantum query complexity of read-once functions, *quant-ph/0201007* (2002).
6) Ambainis, A.: Quantum walk algorithm for el-

ement distinctness, *Proc. 45th Symposium on Foundations of Computer Science*, pp.22–31 (2004).
7) Bernstein, E. and Vazirani, U.: Quantum complexity theory, *SIAM Journal on Computing*, Vol.26, No.5, pp.1411–1473 (1997).
8) Biron, D., Biham, O., Biham, E., Grassl, M. and Lidar, D.A.: Generalized Grover Search Algorithm for Arbitrary Initial Amplitude Distribution, *Proc. 1st NASA International Conference on Quantum Computing and Quantum Communication*, LNCS, Vol.1509, Springer-Verlag, pp.140–147 (1998).
9) Boyer, M., Brassard, G., Høyer, P. and Tapp, A.: Tight bounds on quantum searching, *Fortschritte der Physik*, Vol.46, No.4-5, pp.493–505 (1998).
10) Buhrman, H., Dürr, C., Heiligman, M., Høyer, P., Magniez, F., Santha, M. and de Wolf, R.: Quantum Algorithms for Element Distinctness, *Proc. 16th IEEE Annual Conference on Computational Complexity* (*CCC'01*), pp.131–137 (2001).
11) Brassard, G., Høyer, P., Mosca, M. and Tapp, A.: Quantum Amplitude Amplification and Estimation, *Quantum Computation and Quantum Information: A Millennium Volume*, AMS Contemporary Mathematics Series, Vol.305 (2002).
12) Blum, A., Kalai, A. and Wasserman. H.: Noise-tolerant learning, the parity problem, and the statistical query model, *STOC*, pp.435–440 (2000).
13) Buhrman, H., Newman, I., Röhrig, H. and de Wolf, R.: Robust quantum algorithms and polynomials, *quant-ph/0309220* (2003). To appear in *STACS* (2005).
14) Chi, D.P. and Kim, J.: Quantum Database Searching by a Single Query, *Proc. 1st NASA International Conference on Quantum Computing and Quantum Communication*, LNCS, Vol.1509, Springer-Verlag, pp.148–151 (1998).
15) Dürr, C., Heiligman, M., Høyer, P. and Mhalla. M.: Quantum query complexity of some graph problems, *Proc. 31st International Colloquium on Automata, Languages and Programming*, LNCS 3142, pp.481–493 (2004).
16) Feige, U., Raghavan, P., Peleg, D. and Upfal, E.: Computing with noisy information, *SIAM Journal on Computing*, Vol.23, No.5, pp.1001–1018 (1994).
17) Goldreich, O. and Levin, L.: Hard-core predicates for any one-way function, In *STOC*, pp.25–32 (1989).
18) Grover, L.K.: A fast quantum mechanical algorithm for database search, *Proc. 28th ACM Symposium on Theory of Computing*, pp.212–218 (1996).
19) Grover, L.K.: A framework for fast quantum mechanical algorithms, *Proc. 30th ACM Symposium on Theory of Computing*, pp.53–62 (1998).
20) Grover, L.K.: Rapid sampling through quantum computing, *Proc. 32th ACM Symposium on Theory of Computing*, pp.618–626 (2000).
21) Høyer, P., Mosca, M. and de Wolf, R.: Quantum search on bounded-error inputs, *Proc. 30th International Colloquium on Automata, Languages and Programming*, LNCS 2719, pp.291–299 (2003).
22) Iwama, K., Kawachi, A., Raymond, R. and Yamashita, S.: Robust Quantum Algorithms for Oracle Identification, *quant-ph/0411204* (2004).
23) Iwama, K., Raymond, R. and Yamashita, S.: Quantum query complexity of biased oracles, *Booklet of Workshop on ERATO Quantum Information Science 2003*, pp.33–34 (2003).
24) Magniez, F., Santha, M. and Szegedy, M.: Quantum algorithms for the triangle problem, *Proc. 16th ACM-SIAM Symposium on Discrete Algorithms* (2005). To appear.
25) Pablo-Norman, B. and Ruiz-Altaba. M.: Noise in Grover's quantum search algorithm, *Physical Review A*, 61:012301 (1999). See also quant-ph/9903070.
26) Shenvi, N., Brown, K.R. and Whaley, K.B.: Effects of noisy oracle on search algorithm complexity, *Physical Review A*, 68:052313 (2003). See also quant-ph/0304138.
27) Shi, Y.: Quantum lower bounds for the collision and the element distinctness problems, *Proc. 43rd IEEE Symposium on the Foundation of Computer Science*, pp.513–519 (2002).
28) Szegedy, M. and Chen. X.: Computing boolean functions from multiple faulty copies of input bits, Rajsbaum, S., (Eds.), *LATIN 2002*, pp.539–553 (2002).

(Received February 16, 2005)
(Accepted July 4, 2005)
(Online version of this article can be found in the IPSJ Digital Courier, Vol.1, pp.461–469.)

Kazuo Iwama is a Professor of Graduate School of Informatics, Kyoto University. He received B.E., M.E. and Ph. D. degrees from Department of Electrical Engineering, Kyoto University in 1973, 1975 and 1980, respectively. His research interests are mainly algorithms and complexity theory including online, SAT and parallel algorithms beside quantum algorithms.

Akinori Kawachi is an Assistant Professor of Graduate School of Information Science and Engineering, Tokyo Institute of Technology, Japan. Received B.E., M.E. and Ph.D. degrees in Information Science from Kyoto University in 2000, 2002 and 2004, respectively. His research interests are quantum computation, foundations of cryptography, and complexity theory.

Shigeru Yamashita is an Associate Professor of Graduate School of Information Science, Nara Institute of Science and Technology. Received B.E., M.E. and Ph.D. degrees in Information Science from Kyoto University, Kyoto, Japan, in 1993, 1995 and 2001, respectively. His research interests include new types of computation including quantum computation and reconfigurable computing. He received the Best Paper Award of the 2000 IEEE Circuits and Systems Society Transactions on Computer-Aided Design of Integrated Circuits and Systems.