Quantum Inf Process (2011) 10:177–188
DOI 10.1007/s11128-010-0187-9

Quantum algorithm for the asymmetric weight decision problem and its generalization to multiple weights

Byung-Soo Choi · Samuel L. Braunstein

Received: 2 March 2010 / Accepted: 30 June 2010 / Published online: 13 July 2010

Abstract As one of the applications of Grover search, an exact quantum algorithm for the symmetric weight decision problem of a Boolean function has been proposed recently. Although the proposed method shows a quadratic speedup over the classical approach, it only applies to the symmetric case of a Boolean function whose weight is one of the pair $\{0 < w_1 < w_2 < 1, w_1 + w_2 = 1\}$. In this article, we generalize this algorithm in two ways. Firstly, we propose a quantum algorithm for the more general asymmetric case where $\{0 < w_1 < w_2 < 1\}$. This algorithm is exact and computationally optimal. Secondly, we build on this to exactly solve the multiple weight decision problem for a Boolean function whose weight as one of $\{0 < w_1 < w_2 < \cdots < w_m < 1\}$. This extended algorithm continues to show a quantum advantage over classical methods. Thirdly, we compare the proposed algorithm with the quantum counting method. For the case with two weights, the proposed algorithm shows slightly lower complexity. For the multiple weight case, the two approaches show different performance depending on the number of weights and the number of solutions. For smaller number of weights and larger number of solutions, the weight decision algorithm can show better performance than the quantum counting method. Finally, we discuss the relationship between the weight decision problem and the quantum state discrimination problem.

B.-S. Choi (✉)
Department of Electronics Engineering, Ewha Womans University,
Seoul, Republic of Korea (South Korea)
e-mail: bschoi3@gmail.com

S. L. Braunstein
Department of Computer Science, University of York, York, YO10 5DD, United Kingdom
e-mail: schmuel@cs.york.ac.uk

Keywords Quantum algorithm · Generalized weight decision problem · Query complexity

1 Introduction

Quantum database search shows a quadratic speedup over any classical approach [1,2]. Although the speedup of quantum search is modest, the application areas of the quantum search are significantly wider. For this reason, many studies have investigated the properties of quantum search [3–9] or extended it to other areas [10–18]. Our work here provides one such extension.

For a binary function, its weight is the ratio of inputs which evaluate to one over all possible inputs. Weight analysis has been investigated for cryptanalysis [19], coding theory [20], fault-tolerant circuit design [21], and for the built-in self-testing of circuits [22].

For a general Boolean function, or oracle, which can be evaluated on N inputs, the query complexity for weight analysis is $O(N)$ for any classical algorithm [23]. By contrast, quantum algorithms showed promise in the weight analysis of Boolean functions from the start. The Deutsch algorithm can determine whether a one-bit input function is constant or balanced with a single oracle evaluation [24]. This was extended to n-bit input functions in the Deutsch-Jozsa algorithm [25] where it shows an exponential advantage to any exact classical algorithm. Indeed, the Deutsch-Jozsa algorithm has been analyzed in the context of cryptanalysis [26]. This suggests that more general weight analysis algorithms may play a useful role for cryptanalysis.

In this work, we focus on the weight decision problem: the task is to determine which of a promised pair of weights, $\{w_1, w_2\}$, a Boolean function actually possesses. In its initial setting this problem was proposed with the symmetric constraint that $w_1 + w_2 = 1$ and it was found that an efficient quantum algorithm provides a quadratic speedup over any classical approach [27].

In this study, we extend it to the asymmetric and the multiple weight cases, and compare it with the quantum counting and the quantum state discrimination. Firstly, we consider the asymmetric weight decision problem where $\{0 < w_1 < w_2 < 1\}$. We reduce this problem to the symmetric case by adding two more qubits with modification of the Boolean function. This algorithm yields exact results which it inherits from the exact solution to the symmetric weight decision algorithm and also shows a quadratic speedup over classical approaches. Secondly, we consider the multiple weight case where $\{0 < w_1 < w_2 < \cdots < w_m < 1\}$. For this problem, we propose a knock-out approach with an overall complexity $O(m\sqrt{N})$. Hence when $m < \sqrt{N}$, the proposed algorithm shows a speedup over classical approaches. Thirdly, we compare the proposed weight decision algorithm and the quantum counting method. For the case of two weights, both algorithms are computationally optimal, however the proposed weight decision algorithm shows slightly lower complexity. For the multiple weight case, the relative advantage of the two approaches depends on the number of weights and the number of solutions. In this case, exact quantum counting needs $O(\sqrt{tN})$ queries, where t is the number of solutions. Hence when $m < \sqrt{t}$, the weight decision algorithm shows better performance. Lastly, we compare quantum state discrimination and the weight decision algorithm; quantum state discrimination cannot perfectly distinguish two quantum states unless they are mutually orthogonal. On the other hand, in the weight decision problem, two (generally non-orthogonal) quantum states representing Boolean functions can be exactly distinguished with unit probability using multiple invocations of the oracle. Therefore we find that access to the oracle is more powerful than access solely to the quantum states that represent them.

This paper is organized as follows. Details of the weight decision problems for the symmetric, asymmetric, and multiple-weight cases are discussed in Sect. 2. In Sect. 3, we propose a quantum algorithm for the asymmetric case and show that this is exact and computationally optimal. Section 4 extends the problem to the multiple weight case and obtains the condition for retaining a quantum advantage. The comparison between quantum counting and the proposed algorithm is shown in Sect. 5. Some relationships between the weight decision problem and quantum state discrimination are considered in Sect. 6. Section 7 concludes with some discussions of possible future work.

2 Weight decision problems

Definition 1 (*Weight w of a Boolean Function f*)[27] The weight w of a Boolean function f is defined as the ratio of the number of inputs whose outputs are one over the number of all possible inputs of f.

Definition 2 (*Multiple Weight Decision Problem*) Given a Boolean function f, decide exactly the weight w of f where $w \in \{0 < w_1 < w_2 \cdots < w_m < 1\}$.

For two weights, the most general form of the problem is called the asymmetric weight decision problem. Additionally, if these weights satisfy $w_1 + w_2 = 1$, we call it the symmetric weight decision problem.

3 Quantum algorithm for the asymmetric weight decision problem

3.1 Brief description of the symmetric weight decision algorithm

3.1.1 Properties of the Grover operator

Consider a Boolean function f with a weight $w = \sin^2 \frac{\beta_w}{2}$, where $0 \leq \beta_w \leq \pi$. The uniform superposition of all states is used as the initial state for the algorithm and may be expressed as

$$|\psi_{w,0}\rangle = \sin \frac{\beta_w}{2} |s\rangle + \cos \frac{\beta_w}{2} |ns\rangle \ , \tag{1}$$

where $|s\rangle$ and $|ns\rangle$ denote the uniform superpositions of solution [i.e., where $f(x) = 1$] and non-solution [i.e., where $f(x) = 0$] basis states, respectively.

Now the generalized Grover operator consists of two inversion operators as

$$G = -I_{|\psi_{w,0}\rangle}(\theta) I_{|s\rangle}(\phi) \ , \tag{2}$$

where the inversion operator is defined as

$$I_{|\psi\rangle}(\theta) \equiv I - (1 - e^{i\theta})|\psi\rangle\langle\psi| \ . \tag{3}$$

Since $\theta = \phi = \pi$ in the standard Grover search algorithm, the standard form of the Grover operator is

$$\begin{aligned} G &= -I_{|\psi_{w,0}\rangle}(\pi) I_{|s\rangle}(\pi) \\ &= (2|\psi_{w,0}\rangle\langle\psi_{w,0}| - I)(I - 2|s\rangle\langle s|), \end{aligned} \tag{4}$$

where $-I_{|\psi_{w,0}\rangle}(\pi)$ inverts all states about the average, and $I_{|s\rangle}(\pi)$ flips the sign of all solution states.

After applying the standard Grover operator k times, the initial state becomes

$$|\psi_{w,k}\rangle = \sin(2k+1)\frac{\beta_w}{2}|s\rangle + \cos(2k+1)\frac{\beta_w}{2}|ns\rangle \ . \tag{5}$$

Measuring this state in the computational basis, we can find one of the solutions with a success probability of $\sin^2(2k+1)\frac{\beta_w}{2}$.

3.1.2 Simple application of the Grover operator for weight decision

As the simple case, we study which weights can be decided exactly after k iterations of the Grover operator. Consider a situation where the weight is $w_1 = \sin^2\left(\frac{k}{2k+1}\frac{\pi}{2}\right)$, so β_{w_1} should be $\frac{k}{2k+1}\pi$. Note that if the weight is $w_2 = \cos^2\left(\frac{k}{2k+1}\frac{\pi}{2}\right)$ we may reformulate it as

$$\begin{aligned} \cos^2\left(\frac{k}{2k+1}\frac{\pi}{2}\right) &= \cos^2\left(\frac{2k+1-k-1}{2k+1}\frac{\pi}{2}\right) \\ &= \sin^2\left(\frac{k+1}{2k+1}\frac{\pi}{2}\right) \ , \end{aligned} \tag{6}$$

to conclude that β_{w_2} should be $\frac{k+1}{2k+1}\pi$.

After applying k iterations of the Grover operator, the final states for the two cases $w = w_1$ and $w = w_2$ are

$$|\psi_{w_1,k}\rangle = \sin^2\left(\frac{k\pi}{2}\right)|s\rangle + \cos^2\left(\frac{k\pi}{2}\right)|ns\rangle \tag{7}$$

and

$$|\psi_{w_2,k}\rangle = \sin^2\left(\frac{(k+1)\pi}{2}\right)|s\rangle + \cos^2\left(\frac{(k+1)\pi}{2}\right)|ns\rangle \tag{8}$$

$$= \cos^2\left(\frac{k\pi}{2}\right)|s\rangle + \sin^2\left(\frac{k\pi}{2}\right)|ns\rangle \tag{9}$$

respectively.

Note that if k is even, $|\psi_{w_1,k}\rangle = |ns\rangle$ which denotes the normalized non-solution state; similarly, $|\psi_{w_2,k}\rangle = |s\rangle$ which denotes the normalized solution state. As a result, if k is even, and the measured value $\hat{x}$ is one of solutions, then we can conclude that $w = w_2$ otherwise $w = w_1$. Likewise if k is odd, the two final states are exchanged. Therefore, based on the value of k and the function output of the measured value, we can decide the exact weight.

3.1.3 Phase matching for the symmetric weight decision problem

Unfortunately, the above algorithm can find the exact weight *only* when $w_1 + w_2 = 1$ and $w_1 = \sin^2\left(\frac{k}{2k+1}\frac{\pi}{2}\right)$ and $w_2 = \cos^2\left(\frac{k}{2k+1}\frac{\pi}{2}\right)$, and hence cannot be used even for the symmetric weight decision problem. To resolve this problem, the symmetric weight decision algorithm was proposed as shown in Algorithm 1. The basic idea of this approach is to use the same idea of the above algorithm for the first to $k-2$ Grover operators, and apply some phase matching for the last two Grover operators as shown in the third and fourth step. Hence the number of Grover iterations is k, which can be evaluated from line 1.b in Algorithm 1. Basically this approach is based on the phase matching condition for the sure success quantum search algorithm [3–9]. In this case, the algorithm applies two phase matching conditions for the last two phases because there are two weights. The phases for the last two steps are as follows.

$$\cos\theta_1 = \frac{(-1)^k \cos\beta_{w_1} - \cos 2\beta_{w_1}\cos(2k-2)\beta_{w_1}}{\sin 2\beta_{w_1}\sin(2k-2)\beta_{w_1}}. \tag{10}$$

$$\cos\theta_2 = \frac{(-1)^k \sin 2\beta_{w_1}(y\sin\theta_2 - (-1)^k \sin\beta_{w_1})}{\cos\beta_{w_1}\cos 2\beta_{w_1} - (-1)^k\cos(2k-2)\beta_{w_1}}. \tag{11}$$

The interpretation of the measurement output and hence the decision is the same as with the simple application case.

3.2 Asymmetric weight decision algorithm

The key idea for the symmetric weight decision algorithm is based on the symmetry between the given weights where the sum of two weights must be one. On the Bloch sphere, the two initial states are located in symmetric positions in the XY-plane, and hence the Grover operator would rotate the two states in a symmetric fashion. This is

Algorithm 1 Quantum Algorithm for the Symmetric Weight Decision Problem [27]

```
1.a   $|\psi_{w,0}\rangle = (H|0\rangle)^{\otimes n}|1\rangle, i = 0.$
1.b   If $w_1 \leq \sin^2 \frac{\pi}{5}$, $k$ is 2, otherwise $k$ satisfies $\sin^2\left(\frac{k-1}{2k-1}\frac{\pi}{2}\right) < w_1 \leq \sin^2\left(\frac{k}{2k+1}\frac{\pi}{2}\right)$.
1.c   $\frac{k-1}{2k-1}\pi < \beta_{w_1} \leq \frac{k}{2k+1}\pi. \beta_{w_1} + \beta_{w_2} = \pi.$
2.    While($i < (k-2)$) do
      {
          $|\psi_{w,i+1}\rangle = -I_{|\psi_{w,0}\rangle}(\pi)I_{|s\rangle}(\pi)|\psi_{w,i}\rangle,$
          $i = i + 1$
      }
3.    $|\psi_{w,k-1}\rangle = -I_{|\psi_{w,0}\rangle}(-\theta_1)I_{|s\rangle}(\pi)|\psi_{w,k-2}\rangle$
4.    $|\psi_{w,k}\rangle = -I_{|\psi_{w,0}\rangle}(-\theta_2)I_{|s\rangle}(\pi)|\psi_{w,k-1}\rangle$
5.    Measure $|\psi_{w,k}\rangle$ in the computational basis. Let the result be $\hat{x}$.
6-1.  If $k$ is odd and if $f(\hat{x}) = 1$ then $w = w_1$ else $w = w_2$.
6-2.  If $k$ is even and if $f(\hat{x}) = 1$ then $w = w_2$ else $w = w_1$.
```

the reason why we call this algorithm the symmetric weight decision algorithm in this article, not the limited case as it is called in Ref. [27].

Unfortunately, since the asymmetric case has no such symmetry between the two weights in the Bloch sphere it appears to be relatively difficult to design a suitable algorithm. However, we can reduce the asymmetric case to the symmetric case by modifying the Boolean function by adding extra inputs in such a way as to regain the symmetry.

Here we shall write $w_1 = \frac{n_1}{N}$ and $w_2 = \frac{n_2}{N}$ for a given Boolean function f. Now we consider extra inputs from N to $4N - 1$. Hence the total input space is $4N$. Based on this, we can formulate the modified Boolean function f' as follows.

$$f'(x) \equiv \begin{cases} f(x), & 0 \leq x < N, \\ f(x-N), & N \leq x < 2N, \\ 1, & 2N \leq x < 2N + l, \\ 0, & 2N + l \leq x < 4N. \end{cases} \tag{12}$$

Hence the modified weight for f' is $w_1{}' = \frac{n_1+n_1+l}{4N} = \frac{2n_1+l}{4N}$ and $w_2{}' = \frac{n_2+n_2+l}{4N} = \frac{2n_2+l}{4N}$. To make it satisfy the symmetric condition, the value l should be $2N-(n_1+n_2)$.

Since the Boolean function is modified, the corresponding oracle operation must be changed. In addition, this change requires just two more qubits, in order to represent inputs from N to $4N - 1$. Having made these changes, the asymmetric weight decision algorithm has been reduced to the symmetric weight decision algorithm on this modified oracle and modified input space. This allows us to use Algorithm 1 with our modified oracle.

3.3 Analysis

First, the proposed algorithm is one of the exact or sure success algorithms. In this case, the additional value l is an integer number, and hence Algorithm 1 can be used.

Second, the proposed algorithm is computationally optimal. Since the asymmetric weight decision problem can be reduced to the symmetric weight decision problem,

Algorithm 2 Multiple Weight Decision Algorithm

```
Let      $S = \{w_1, w_2, \ldots, w_m\}$.
WHILE    $(|S| \neq 1)$
{
         $w_{\min}$ = smallest weight from $S$.
         $w_{\max}$ = largest weight from $S$.
         Asymmetric Weight Decision Algorithm($w_{\min}, w_{\max}$).
         $S = S -$ non_selected weight.
}
         Return the exact weight as $S$.
```

the computational complexity for the symmetric weight decision problem is the same for the asymmetric weight decision problem. For the symmetric weight decision problem, let us assume w_1 and w_2 are $\sin^2\left(\frac{k}{2k+1}\frac{\pi}{2}\right)$ and $\cos^2\left(\frac{k}{2k+1}\frac{\pi}{2}\right)$, respectively. In that case, the symmetric weight decision algorithm can decide the exact weight w with $O(k)$ Grover operations [27]. However, for the same problem, any classical probabilistic algorithm needs $\Omega(k^2)$ query complexity [27]. Hence the symmetric weight decision algorithm shows a quadratic speedup over the classical approach, which means the computational complexity is $O(\sqrt{N})$ since the classical bound is $O(N)$. Meanwhile, the computational lower bound for estimating the weight of a given Boolean functions is $\Omega\left(\sqrt{\frac{N}{\lceil\epsilon(t+1)\rceil}} + \frac{\sqrt{t(N-t)}}{\lceil\epsilon(t+1)\rceil}\right)$, where t is the number of solutions and ϵ is an error bound, and hence is $\Omega(\sqrt{N})$ as shown in Theorem 1.13 of Ref. [28]. Therefore, the proposed quantum algorithm is one of the optimal algorithms.

4 Extension to the multiple weight decision problem

4.1 Algorithm

Our method can be extended to a more general situation where multiple weights $0 < w_1 < w_2 < \cdots < w_m < 1$ are given. Algorithm 2 shows a way of exploiting the asymmetric weight decision algorithm for two weights. The set S contains all candidate weights w_i. Until the set S contains only one element, we apply the asymmetric weight decision algorithm with two candidate weights $w_{\min}$ and $w_{\max}$ which are the smallest and the largest weights from S. After applying the asymmetric weight decision algorithm, a candidate weight is selected. In the next step, the weight which is not selected is eliminated from S. With the smaller sized S, the same round applies again until S contains only one element. When S contains only one element, this element represents the actual weight of the Boolean function.

4.2 Correctness

The correctness of Algorithm 2 is based on the following two properties.

First, because of the sure success property, the asymmetric weight decision algorithm can choose the exact weight when one of two candidate weights is the actual weight. Because of this property, if the exact weight is checked in some round, the exact weight will survive until the last round.

Second, since the proposed algorithm eliminates only one element from S each round, each weight will be checked at least once. Therefore, the exact weight is also checked at least once in some round.

Therefore, the exact weight will be checked at least once by the second property and will survive until the last round by the first property.

4.3 Analysis

The computational complexity of the classical approach for the case of multiple weights is $O(N)$. Similarly, the computational complexity of the asymmetric weight decision problem is $O(\sqrt{N})$. Based on the proposed method, we need $m - 1$ repetitions of the asymmetric weight decision algorithm, and hence the total computational complexity is $O(m\sqrt{N})$. Therefore, provided $m \leq \sqrt{N}$, our approach is advantageous over classical approaches.

Meanwhile two candidate weights (the smallest and the largest weights from S) are chosen considering the smallest number of Grover iterations. In the symmetric weight decision algorithm, the number of Grover iterations depends on the value of w_1 as shown in $1.b$ of Algorithm 1. Hence if w_1 is smaller, the number of Grover iterations is smaller as well. It also implies that when the difference between two weights is large, the number of Grover iterations is small. Therefore, to reduce the number of Grover iterations in each while loop, we select two weights with the largest weight difference. Therefore, in the algorithm the smallest and the largest weights are chosen for the candidate weights, which needs the smallest number of Grover iterations. By making this selection for the loops that follow the total number of Grover iterations can be minimized.

5 Comparison with the quantum counting algorithm

Since quantum counting can count the number of solutions of a given function [29], it can be applied to the weight decision problem.

5.1 Quantum counting

Since repeated Grover operations show a periodic pattern of the solution and the non-solution basis, it contains the weight information. To retrieve such period information, the quantum Fourier transform is used as shown in Algorithm 3 [29]. In the algorithm, $\mathbf{G}_f = \mathbf{Q}(\mathbf{W}, f, -1, -1)$ denotes the Grover operator with the notation of Refs. [1,2], where $\mathbf{W}$ denotes the Walsh-Hadamard transform on n qubits, f is the given Boolean function, the first and second -1 denote $\theta = \pi$ and $\phi = \pi$, respectively. Also $\mathbf{f}_P$ represents the quantum Fourier transform over P, which determines the time taken by the algorithm, and consequently, the precision of the estimation. In this work, the value P represents the number of oracle queries for the Boolean function.

Algorithm 3 Quantum Counting [29]

```
Let   $\mathbf{C}_f : |x\rangle \otimes |\Psi\rangle \mapsto |x\rangle \otimes (\mathbf{G}_f)^x |\Psi\rangle$
Let   $\mathbf{f}_P : |k\rangle \mapsto \frac{1}{\sqrt{P}} \sum_{l=0}^{P-1} e^{2\pi \imath kl/P} \, |l\rangle$
      $|\Psi_0\rangle = \mathbf{W} \otimes \mathbf{W}|0\rangle|0\rangle$
      $|\Psi_1\rangle = \mathbf{C}_f \, |\Psi_0\rangle$
      $|\Psi_2\rangle = |\Psi_1\rangle$ after the second register is measured (optional)
      $|\Psi_3\rangle = \mathbf{f}_P \otimes \mathbf{I}|\Psi_2\rangle$
      $\hat{x}$ = measured value of $|\Psi_3\rangle$
      $\left(\text{if } \hat{x} > \frac{P}{2} \text{ then } \hat{x} = (P - \hat{x})\right)$
      output: $N \sin^2\left(\frac{\hat{x}\pi}{P}\right)$ (and $\hat{x}$ if needed)
```

5.2 Comparison of query complexity for the asymmetric weight decision problem

Since the asymmetric weight decision problem can be reduced to the symmetric weight decision problem as shown in the previous section, it is sufficient to compare the query complexity for two weights such as $w_1 = \sin^2\left(\frac{k\pi}{4k+2}\right)$ and $w_2 = \cos^2\left(\frac{k\pi}{4k+2}\right)$.

For this problem, the weight decision algorithm can decide the actual weight with sure success by calling k queries.

With the same condition, we need to analyze the number of queries for quantum counting. First, we must understand the relation between P, $\hat{x}$, and the given weight. If we are asked to check whether or not the given weight is correct, we should know the exact value of P and the expected value of $\hat{x}$. Note that P is the number of queries for quantum counting, and hence we have to find the exact value of P for the symmetric case. For the chosen weights, if the actual weight is $w_1 = \sin^2\left(\frac{k\pi}{4k+2}\right)$, then we can confirm it by using $P = 4k + 2$ and by checking whether or not $\hat{x}$ is k. Likewise, if the actual weight is $w_2 = \cos^2\left(\frac{k\pi}{4k+2}\right) = \sin^2\left(\frac{(k+1)\pi}{4k+2}\right)$, then we can verify it by using $P = 4k + 2$ and by checking whether or not $\hat{x}$ is $k + 1$. Therefore, the quantum counting method can be exploited by assuming $P = 4k + 2$ and by checking $\hat{x}$. More explicitly, after P queries, if $\hat{x}$ is k, we can conclude that the exact weight is w_1; if $\hat{x}$ is $k + 1$, the exact weight is w_2. Note that since k and $k + 1$ are integer, these two cases can be distinguished with sure success by the same number of queries. In addition to that, since the difference between k and $k + 1$ is just one, which is the smallest integer, we cannot reduce P. Therefore, for sure success quantum counting, the value P should be $4k + 2$, which means the number of queries is $4k + 2$.

Therefore, quantum counting for this problem takes around $4k + 2$ queries, but our method requires only k queries, and hence our method requires four times fewer queries than the quantum counting method. As an aside we note that the quantum counting method is based on the quantum Fourier transform, which might be difficult to implement depending on what kind of quantum computer is being used.

5.3 Comparison of query complexity for the multiple weight decision problem

In general, regardless of information about the number and value of the weights, quantum counting can estimate the number of solutions, and hence can infer the actual weight. Therefore, it is sufficient to find the query complexity for the general case. The query complexity of exact quantum counting for the general case is $O(\sqrt{tN})$, where t is the number of solutions, shown in Corollary 4 of Ref. [29].

On the other hand, the proposed knock-out algorithm exploits information about the number of weights. Unfortunately, this information is not so much use since the query complexity increases linearly with the number of weights. Hence the total query complexity for the proposed algorithm is $O(m\sqrt{N})$, where m is the number of weights.

Now quantum counting and the weight decision algorithm show different performances depending on the values of t and m. When $m > \sqrt{t}$, which means the number of weights is much larger than the number of solutions, quantum counting shows better performance. Otherwise, the proposed weight decision algorithm is better. Unfortunately, since we do not know the value of t in advance, we cannot select which method is best for the given problem. However, for the case when the number of weights is small and w_1 is large (hence t is large also), the proposed weight decision algorithm would show better performance.

6 Relation with quantum state discrimination

In quantum state discrimination, we are given a single quantum state, where the state is either $|\psi_1\rangle$ or $|\psi_2\rangle$. The goal is to find the exact quantum state with high probability. Since the discrimination of the quantum states is very important, many works have investigated this question. Details about quantum state discrimination and its potential applications may be found in Ref. [30].

For this problem, unfortunately, the success probability is given by the Helstrom bound to be $\frac{1}{2}(1+\sqrt{1-|\langle\psi_1|\psi_2\rangle|^2})$. Therefore, unless two candidate states are orthogonal, we cannot guarantee the exact quantum state.

On the other hand, we can regard the two states as the quantum states for the given two weights. In other words, we can say that the weight decision problem is an oracle discrimination problem. Contrary to quantum state discrimination, in the oracle discrimination problem, we have additional information, namely the oracle. Hence, by applying the oracle several times, we can convert the two quantum states for each candidate oracle into orthogonal states. Therefore, we can increase the success probability to certainty. Meanwhile, depending on the given weights, we can determine the lower bound on oracle queries k by

$$\cos\left(\frac{(k-1)\pi}{2k-1}\right) > |w_2 - w_1| \geq \cos\left(\frac{k\pi}{2k+1}\right). \tag{13}$$

Note that the above equation is for the symmetric weight condition, but it can be generalized to the asymmetric and the multiple weight cases using methods from the previous sections. From this equation we may determine that the number of oracle queries depends on the given weights. By this analysis, we can conclude that the additional information, the oracle, is very useful and makes a sharp difference from the difficult problem of quantum state discrimination.

7 Conclusion and open problems

In this work, we have shown two generalizations of the symmetric weight decision algorithm for the case of asymmetric weights and its extension for the case of multiple weights. However, we have the following questions as well.

In this study we have considered the case where two weights are exact. However, we can consider the case where two weights are not given exactly, but only roughly, for example, $w_1^{\text{low}} \leq w_1 \leq w_1^{\text{high}}$ and $w_2^{\text{low}} \leq w_2 \leq w_2^{\text{high}}$. In this case, since we do not know the exact values of w_1 and w_2 in advance, we cannot know the required minimum value of k_1 and k_2, and hence only an approximate approach will be possible. Now the question is approximately how many queries are sufficient to resolve this situation with a success probability higher than a certain value. Or what is the query lower bound for this problem. And what is an optimal quantum algorithm?

Finally, it would be worthwhile to investigate in more detail the kind of classical cryptanalysis algorithms that can be improved using the weight analysis available through the current algorithm with the aid of a quantum computer. Since most classical cryptanalysis problems rely heavily on large numbers of (parallel) computations, this area would be one candidate where quantum computers should show real advantage.

Acknowledgements We thank Masahito Hayashi for suggesting how to extend the asymmetric weight decision algorithm to the case of multiple weights.

References

1. Grover, L.K.: A fast quantum mechanical algorithm for database search. In: Proceedings of Symposium on the Theory of Computing, pp. 212–219 (1996)
2. Grover, L.K.: Quantum mechanics helps in searching for a needle in a haystack. Phys. Rev. Lett. **79**(2), 325–328 (1997)
3. Zalka, C.: Grover's quantum searching algorithm is optimal. Phys. Rev. A **60**(4), 2746–2751 (1999)
4. Høyer, P.: Arbitrary phases in quantum amplitude amplification. Phys. Rev. A **62**, 052304 (2000)
5. Long, G.-L.: Grover algorithm with zero theoretical failure rate. Phys. Rev. A **64**, 022307 (2001)
6. Long, G.-L., Lia, X., Sune, Y.: Phase matching condition for quantum search with a generalized initial state. Phys. Lett. A **294**(3–4), 143–152 (2002)
7. Biham, O., Shapira, D., Shimoni, Y.: Analysis of Grover's quantum search algorithm as a dynamical system. Phys. Rev. A **68**, 022326 (2003)
8. Grover, L.K.: Fixed-point quantum search. Phys. Rev. Lett. **95**, 150501 (2005)
9. Li, D., Li, X., Huang, H., Li, X.: Fixed-point quantum search for different phase shifts. Phys. Lett. A **362**(4), 260–264 (2007)
10. Protopopescu, V., Barhen, J.: Solving a class of continuous global optimization problems using quantum algorithms. Phys. Lett. A **296**(1), 9–14 (2002)
11. Baritompa, W.P., Bulger, D.W., Wood, G.R.: Grover's quantum algorithm applied to global optimization. SIAM J. Optim. **15**(4), 1170–1184 (2005)
12. Korepin, V.E.: Optimization of partial search. J. Phys. A: Math. Gen. **38**(44), L731–L738(1) (2005)
13. Korepin, V.E., Grover, L.K.: Simple algorithm for partial quantum search. Quantum Inf. Process. **5**(1), 5–10 (2006)
14. Choi, B.-S., Walker, T.A., Braunstein, S.L.: Sure succss partial search. Quantum Inf. Process. **6**(1), 1–8 (2007)
15. Choi, B.-S., Korepin, V.E.: Quantum partial search of a database with several target items. Quantum Inf. Process. **6**(4), 243–254 (2007)
16. Dua, J.-Z., Qin, S.-J., Wen, Q.-Y., Zhu, F.-C.: Threshold quantum cryptograph based on Grover's algorithm. Phys. Lett. A **363**(5–6), 361–368 (2007)

17. Fang, Y., Kaszlikowski, D., Chin, C., Tay, K., Kwek, L.C., Oh, C.H.: Entanglement in the grover search algorithm. Phys. Lett. A **345**(4–6), 265–272 (2005)
18. Shimoni, Y., Biham, O.: Groverian entanglement measure of pure quantum states with arbitrary partitions. Phys. Rev. A **75**, 022308 (2007)
19. Filiol, E., Fontaine, C.: Highly nonlinear balanced Boolean functions with a good correlation-immunity. In: Proceedings of Advances in Cryptology-EUROCRYPT'98, International Conference on the Theory and Application of Cryptographic Technique. Lecture Notes in Computer Science, vol.1403, pp. 475–488 (1998)
20. MacWilliams, F.J., Sloane, N.J.A.: The theory of error-correcting codes. Publisher, North Holland (1996)
21. Chakrabarty, K., Hayes, J.P.: Balance testing and balance-testable design of logic circuits. J. Electron. Testing **8**(1), 71–86 (1996)
22. Chakrabarty, K., Hayes, J.P.: Cumulative balance testing of logic circuits. IEEE Trans. VLSI Syst. **3**(1), 72–83 (1995)
23. Filiol, E.: Designs, intersecting families, and weight of Boolean functions. In: Proceedings of the 7th IMA International Conference on Cryptography and Coding. Lecture Notes In Computer Science, vol. 1746, pp. 70–80 (1999)
24. Deutsch, D.E.: Quantum theory, the Church-Turing principle and the universal quantum computer. Proc. R. Soc. A **400**(1818), 97–117 (1985)
25. Deutsch, D., Jozsa, R.: Rapid solution of problems by quantum computation. Proc. R. Soc. A. **439** (1907), 553–558 (1992)
26. Maitra, S., Mukhopadhyay, P.: Deutsch-Jozsa algorithm revisited in the domain of cryptographically significant Boolean functions. Int. J. Quantum Inf. **3**(2), 359–370 (2005)
27. Braunstein, S.L., Choi, B.-S., Ghosh, S., Maitra, S.: Exact quantum algorithm to distinguish Boolean functions of different weights. J. Phys. A: Math. Theor. **40**, 8441–8454 (2007)
28. Nayak, A., Wu, F.: The quantum query complexity of approximating the median and related statistics. In: Proceedings of the Annual ACM Symposium on Theory of Computing, Atlanta, Georgia, United States, pp. 384–393 (1999)
29. Brassard, G., Høyer, P., Tapp, A.: In: Proceedings of the 25th International Colloquium on Automata, Languages and Programming (ICALP'98). Lecture Notes In Computer Science. vol. 1443, pp. 820–831 (1998)
30. Bergou, J.A.: Quantum state discrimination and selected applications. J. Phys. Conf. Ser. **84**, 012001 (2007)