Effects of systematic phase errors on optimized quantum random-walk search algorithm*

Zhang Yu-Chao(张宇超)[a)b)], Bao Wan-Su(鲍皖苏)[a)b)†], Wang Xiang(汪 翔)[a)b)], and Fu Xiang-Qun(付向群)[a)b)]

[a)] Zhengzhou Information Science and Technology Institute, Zhengzhou 450004, China
[b)] Synergetic Innovation Center of Quantum Information and Quantum Physics, University of Science and Technology of China, Hefei 230026, China

(Received 31 October 2014; revised manuscript received 2 February 2015; published online 10 April 2015)

This study investigates the effects of systematic errors in phase inversions on the success rate and number of iterations in the optimized quantum random-walk search algorithm. Using the geometric description of this algorithm, a model of the algorithm with phase errors is established, and the relationship between the success rate of the algorithm, the database size, the number of iterations, and the phase error is determined. For a given database size, we obtain both the maximum success rate of the algorithm and the required number of iterations when phase errors are present in the algorithm. Analyses and numerical simulations show that the optimized quantum random-walk search algorithm is more robust against phase errors than Grover's algorithm.

Keywords: quantum search algorithm, quantum random walk, phase errors, robustness

PACS: 03.67.Lx, 03.67.Ac **DOI:** 10.1088/1674-1056/24/6/060304

1. Introduction

Quantum computation achieves a new computing pattern for information processing with a powerful parallel computing capability by following the laws of quantum mechanics such as quantum interference, superposition, and entanglement. The Grover algorithm[1] has become one of the most remarkable quantum algorithms because it only requires $O(\sqrt{2^n})$ computational complexity to complete a 2^n-sized database search, achieving quadratic increase in speed compared with the classical exhaustive algorithm. Grover's algorithm demonstrates the superior capability of quantum algorithms. Since then, further research based on Grover's algorithm has been proposed continuously.[2–4] Grover's algorithm has also been physically implemented.[5–7] In addition to Grover's algorithm, other research on quantum algorithms is also developing rapidly.[8,9]

In practice, systematic and random errors are inevitable during the implementation of phase inversion. The error in each step may be very small; however, because the scale of quantum computation is usually exponential, small errors may accumulate into large errors and significantly affect the algorithm results. Robustness of the algorithm against errors determines the stability of the results when the algorithm is affected by error interference, and it can be measured by the difference in success rate before and after the introduction of errors. In 2000, Long *et al.* found that systematic errors in phase inversions and random errors in Hadmard–Walsh transformations have the most dominant effects on Grover's algorithm.[10] In 2003, Shenvi *et al.* studied the robustness of Grover's algorithm against a random phase error in the oracle and the error tolerability of the algorithm.[11] Studying the effects of errors plays a significant role in the determination of the robustness of quantum algorithms, which can improve the reliability of quantum algorithms.

In recent years, quantum random walks have attracted a great deal of interest. A quantum random walk is the quantum counterpart of the classical random walk, in which the random walk evolves with the quantum state, and is affected by quantum interference. In 2003, Shenvi *et al.* proposed a quantum random-walk search algorithm (SKW algorithm).[12] The SKW algorithm iterates $(\pi/4)\sqrt{2^{n+1}}$ times with a success rate of 0.5, making it the first algorithm using quantum random walks to reach quadratic increase in speed on an unstructured 2^n-sized database. In 2008, Potocek *et al.* put forward an optimized quantum random-walk search algorithm (optimized SKW algorithm)[13] on the basis of the SKW algorithm. It iterates $(\pi/4)\sqrt{2^n}$ times with a success rate of approximately 1, achieving the same calculation results of Grover's algorithm. Since then, other quantum algorithms based on quantum random walks have been proposed.[14–17]

In 2006, Li *et al.* studied the effects of errors in phase inversions on the SKW algorithm[18] and found that, in addition to eventually reducing the probability of the marked state, the error also reduces the maximum probability of the unmarked state, which is different from Grover's algorithm. In relevant experiments such as nuclear magnetic resonance, as long as the probability of the marked state is much higher than that of other states, the algorithm is successful. One

*Project supported by the National Basic Research Program of China (Grant No. 2013CB338002).
†Corresponding author. E-mail: 2010thzz@sina.com
© 2015 Chinese Physical Society and IOP Publishing Ltd

of the primary differences between the SKW algorithm and Grover's algorithm is the use of the Grover diffusion operator, i.e., $G = 2|S^{\mathrm{C}}\rangle\langle S^{\mathrm{C}}| - I$. The Hilbert space of the SKW algorithm is the tensor product $H = H^{\mathrm{C}} \otimes H^{\mathrm{S}}$, where H^{C} is the n-dimensional space associated with the quantum coin, and H^{S} is the 2^n-dimensional Hilbert space representing the vertex. The Grover diffusion operator only acts on the n-dimensional coin space as the coin operator and thus

$$|S^{\mathrm{C}}\rangle = \frac{1}{\sqrt{n}} \sum_{d=1}^{n} |d\rangle.$$

However, in Grover's algorithm, it acts on the 2^n-dimensional database, corresponding to the vertex space in the SKW algorithm and thus

$$|S^{\mathrm{C}}\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=1}^{2^n} |x\rangle.$$

The increment in the work space will increase the difficulty of quantum gate implementations and design costs. Furthermore, it may introduce more errors, causing more interference in the algorithm. From the implementary point of view, search algorithms based on random walks may have advantages over Grover's algorithm; therefore, studying quantum random-walk search algorithms has important theoretical significance and practical value. Currently, there exists no study that has determined the effects of phase errors on the optimized SKW algorithm.

This paper studies the effects of systematic phase errors produced by the Grover diffusion operator in the optimized SKW algorithm. Using theoretical analyses and numerical simulations, we establish a model of the algorithm with phase errors and depict the relationship between the success rate of the algorithm, the database size, the phase error, and the number of iterations. For a given database size, the relationship between the maximum success rate of the algorithm and the phase error is determined. The relationship between the required number of iterations and the phase error is also obtained, along with the critical value for the database scale. The algorithm will be almost unaffected by the error if the database scale is smaller than the obtained critical value. Numerical simulations show that the optimized SKW algorithm is more robust than Grover's algorithm.

The rest of this paper is organized as follows. In Section 2, we introduce the optimized SKW algorithm. The geometric description of the optimized SKW algorithm is proposed in Section 3. In Section 4, we establish a model of the algorithm with phase errors. In Section 5, we give the results of numerical simulations and make a robustness comparison with Grover's algorithm. The conclusion is summarized in Section 6.

2. Optimized SKW algorithm

The optimized quantum random-walk search is based on a quantum random walk on a hypercube. The n-dimensional hypercube is a graph with 2^n vertices, each of which can be labelled by an n-bit binary string. Two vertices are connected by an edge if their Hamming distance is 1. The data in the 2^n-sized database are expressed in binary, $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$, $x_i = \{0, 1\}$. $|\boldsymbol{x}|$ is its Hamming weight and $p(\boldsymbol{x}) = |\boldsymbol{x}| \bmod 2$ is its parity bit. The data can be mapped to a 2^{n+1}-dimensional space, $\boldsymbol{x} \to \boldsymbol{x}' = \boldsymbol{x}||p(\boldsymbol{x})$, i.e., adding the parity bit at the end. As the oracle contains information of the marked state, and $\boldsymbol{x}$ is one-to-one with $\boldsymbol{x}'$, we assume that the oracle is capable of identifying the marked state after mapping.

When $\boldsymbol{x}'$ corresponds to the vertex of an $(n+1)$-dimensional hypercube, the Hamming weights of these vertices are even. Assuming that the marked vertex is $|0\rangle$, its selection does not affect the final result because the hypercube is symmetrical. This database search problem is therefore equivalent to searching for a single marked vertex amongst the even vertices on the hypercube.

In detail, the optimized SKW algorithm can be described by the repeated application of a unitary evolution operator UU' on the Hilbert space $H = H^{\mathrm{C}} \otimes H^{\mathrm{S}}$ of an $(n+1)$-dimensional hypercube. Each state in H can be described as $|d, \boldsymbol{x}\rangle$, where d specifies the state of the coin and $\boldsymbol{x}$ specifies the position on the hypercube. The operators can be written as

$$U = S \cdot C_0, \tag{1}$$

$$U' = S \cdot C', \tag{2}$$

where the shift operator is given by

$$S = \sum_{x=0}^{2^{n+1}-1} \sum_{d=1}^{n+1} |d, \boldsymbol{x} \oplus \boldsymbol{e}_d\rangle \langle d, \boldsymbol{x}|. \tag{3}$$

The mapping of a state $|d, \boldsymbol{x}\rangle$ onto the state $|d, \boldsymbol{x} \oplus \boldsymbol{e}_d\rangle$ is dependent on the state of the coin, where $|\boldsymbol{e}_d\rangle = |0 \cdots 010 \cdots 0\rangle$ is a null vector except for a single 1 at the d-th component. The coin operator can be written as

$$C' = C_0 \otimes I + (C_1 - C_0) \otimes |0\rangle \langle 0|. \tag{4}$$

The coin operator consists of two parts: C_0 and C_1. C_0 acts on the unmarked vertices and is generally chosen as the Grover operator

$$C_0 = G = 2\left|S^{\mathrm{C}}\right\rangle \left\langle S^{\mathrm{C}}\right| - I, \tag{5}$$

where

$$|S^{\mathrm{C}}\rangle = \frac{1}{\sqrt{n+1}} \sum_{d=1}^{n+1} |d\rangle$$

is the equal superposition over all $(n+1)$ directions. C_1 acts on the marked vertex and is generally chosen to be $-I$.

The optimized SKW algorithm is implemented as follows:

(i) Initialize the quantum computer to the equal superposition over all states, $|\psi_0\rangle = |S^{\mathrm{C}}\rangle \otimes |S^{\mathrm{S}}\rangle$. Using orthogonal projection P_e, map $|\psi_0\rangle$ to the basis vectors such that the Hamming weight of the vertices are even, and then obtain the initial state $|\psi_0{}^{(e)}\rangle$.

(ii) Apply the evolution operator $UU' = S\cdot(C_0\otimes I)\cdot S\cdot C'$ to the initial state $|\psi_0{}^{(e)}\rangle$ with $(\pi/4)\sqrt{2^n}$ times. C' is used to flip the phase of the marked state, $C'|S^{\mathrm{C}}\rangle\otimes|x\rangle = (-1)^{f(x)}|S^{\mathrm{C}}\rangle\otimes|x\rangle$, when $|x\rangle = |0\rangle$, $f(x)=1$; when $|x\rangle \neq |0\rangle$, $f(x)=0$.

(iii) Measure the quantum state. The probability of the marked state $|S^{\mathrm{C}}\rangle\otimes|0\rangle$ is $1-O(1/(n+1))$.

The coin operator C' changes the phase to mark the target in the algorithm, taking on the function of an oracle. It iterates $(\pi/4)\sqrt{2^n}$ times under ideal conditions, and the success rate is approximately 1. Each iteration is equivalent to two walks on the hypercube, containing four operations with two coin-flip operations and two transfer operations. C' is only applied once; therefore, each iteration only calls the oracle once.

To establish and analyze the model of the optimized SKW algorithm with phase errors, the geometric description of the optimized SKW algorithm will be given in the following section.

3. Geometric description of optimized SKW algorithm

To simplify the problem, we will first show that the random walk on the hypercube can be collapsed onto a random walk on the line. We use the method presented in Ref. [12] to achieve this. Let P_{ij} be the permutation operator defined in Theorem 1 of Ref. [12].

Theorem 1 U and U' are the unitary operators in Section 2 and P_{ij} is the permutation operator. Then UU' and P_{ij} satisfy $P_{ij}^{+}UU'P_{ij} = UU'$.

Proof From Theorem 1 in Ref. [12], it can be proved that $P_{ij}^{+}UP_{ij} = U$ and $P_{ij}^{+}U'P_{ij} = U'$. We multiply these two equations by the left and right respectively and the theorem is proved. ■

The initial state $|\psi_0{}^{(e)}\rangle$ is an eigenvector of P_{ij} with eigenvalue 1 for all i and j. According to Theorem 1, any intermediate state $(UU')^t|\psi_0{}^{(e)}\rangle$ must also be an eigenvector of eigenvalue 1 with respect to P_{ij}. Thus $(UU')^t$ preserves the symmetry of $|\psi_0{}^{(e)}\rangle$ with respect to bit swaps. We define $2n$ basis states, $|R,0\rangle, |L,1\rangle, |R,1\rangle, \ldots, |R,n-1\rangle, |L,n\rangle$, such that their specific forms are the same as those in Eqs. (10) and (11) in Ref. [12]. Based on the above basis states, the coin operator and shift operator will be rewritten respectively by Eqs. (13) and (12) in Ref. [12]; consequently, the unitary operators also take the form of Eqs. (14) and (15) in Ref. [12].

As both U and U' are real matrices, UU and UU' are also real matrices. Their eigenvalues and eigenvectors are complex conjugate pairs. Before analyzing the eigenvalue spectrum of UU', we first analyze the eigenvalue spectrum of UU.

The eigenvalue spectrum of U is already given in Ref. [19]. Because $UU|\upsilon_{\boldsymbol{k}}\rangle = U(\mathrm{e}^{\pm \mathrm{i}\omega_k}|\upsilon_{\boldsymbol{k}}\rangle) = \mathrm{e}^{\pm \mathrm{i}2\omega_k}|\upsilon_{\boldsymbol{k}}\rangle$, we obtain the eigenvalue spectrum of UU as:

$$\begin{aligned}\mathrm{e}^{\pm \mathrm{i}2\omega_k} &= \cos 2\omega_k \pm \mathrm{i}\sin 2\omega_k \\ &= 1 + \frac{8k(k-n)}{n^2} \pm \mathrm{i}\frac{4(n-2k)}{n^2}\sqrt{k(n-k)}. \end{aligned} \tag{6}$$

Theorem 2 U and U' are the unitary operators in Section 2. Then there are two eigenvectors of UU' with eigenvalues close to 1.

Proof Assuming a database of size 2^{n-1} with n being a multiple of 4, we construct two approximate eigenvectors of UU' with $|\psi_0{}^{(e)}\rangle$ and $|\psi_1\rangle$,

$$\begin{aligned}|\psi_0{}^{(e)}\rangle &= \frac{1}{\sqrt{2^{n-1}}}|R,0\rangle + \frac{1}{\sqrt{2^{n-1}}}|L,n\rangle \\ &\quad + \sum_{x=1}^{n/2-1}\left(\sqrt{\frac{\binom{n-1}{2x}}{2^{n-1}}}|R,2x\rangle \right. \\ &\quad \left. + \sqrt{\frac{\binom{n-1}{2x-1}}{2^{n-1}}}|L,2x\rangle\right), \end{aligned} \tag{7}$$

$$\begin{aligned}|\psi_1\rangle &= \frac{1}{c}\left(\sum_{x=0}^{n/4-1}\frac{1}{\sqrt{\binom{n-1}{2x}}}|R,2x\rangle \right. \\ &\quad \left. - \frac{1}{\sqrt{\binom{n-1}{2x+1}}}|L,2x+2\rangle\right), \end{aligned} \tag{8}$$

where

$$c = \sqrt{\sum_{x=0}^{n/2-1}\frac{1}{\binom{n-1}{x}}}.$$

We evaluate

$$\begin{aligned}&UU'|\psi_0{}^{(e)}\rangle \\ &= |\psi_0{}^{(e)}\rangle - \frac{2}{\sqrt{2^{n-1}}}\left(\frac{2-n}{n}|R,0\rangle + \frac{2\sqrt{n-1}}{n}|L,2\rangle\right), \end{aligned} \tag{9}$$

obtaining

$$\langle\psi_0{}^{(e)}|UU'|\psi_0{}^{(e)}\rangle = 1 - \frac{1}{2^{n-2}}. \tag{10}$$

Therefore, when n is large enough, $|\psi_0{}^{(e)}\rangle$ is an eigenvector of UU' with eigenvalue close to 1. Then we evaluate

$$\begin{aligned} UU'|\psi_1\rangle &= |\psi_1\rangle - \frac{1}{c}\left(1-\frac{2}{n}\right)\frac{1}{\sqrt{\binom{n-1}{n/2-2}}} \\ &\quad \times (|R,n/2-2\rangle + |L,n/2+2\rangle) \\ &\quad - \frac{1}{c}\frac{2}{n}\frac{1}{\sqrt{\binom{n-1}{n/2-1}}}(|R,n/2\rangle + |L,n/2\rangle) \end{aligned} \quad (11)$$

obtaining

$$\langle\psi_1|UU'|\psi_1\rangle = 1 - \frac{1}{c^2\binom{n-1}{n/2}}. \quad (12)$$

Because $1 < c^2 < 1+2/n$, $|\psi_1\rangle$ is also an eigenvector of UU' with eigenvalue close to 1. The theorem is hence proved. ■

According to Theorems 2 and 3 in Ref. [12] and the eigenvalue spectrum of UU, there are exactly two eigenvalues $\mathrm{e}^{\mathrm{i}\omega'_0}, \mathrm{e}^{-\mathrm{i}\omega'_0}$ of UU' with their real parts larger than

$$L = 1 - \frac{8n}{3(n+1)^2}. \quad (13)$$

We then set the corresponding eigenvectors $|\omega'_0\rangle$ and $|-\omega'_0\rangle$ such that they satisfy the relation $|\omega'_0\rangle = |-\omega'_0\rangle^*$. According to the Theorem 4 in Ref. [12], it can be approximately proved that $|\omega'_0\rangle$ and $|-\omega'_0\rangle$ are composed of the linear combination of $|\psi_0{}^{(e)}\rangle$ and $|\psi_1\rangle$,

$$|\omega'_0\rangle \approx \frac{1}{\sqrt{2}}\left(\left|\psi_0{}^{(e)}\right\rangle + \mathrm{i}\,|\psi_1\rangle\right), \quad (14)$$

$$|-\omega'_0\rangle \approx \frac{1}{\sqrt{2}}\left(\left|\psi_0{}^{(e)}\right\rangle - \mathrm{i}\,|\psi_1\rangle\right). \quad (15)$$

Finally, we evaluate the range of ω'_0. We evaluate to obtain

$$\langle\psi_1|UU'|\psi_0{}^{(e)}\rangle = \langle\psi_1|\psi_0^{(e)}\rangle + \frac{1}{c}\frac{2}{\sqrt{2^n}}, \quad (16)$$

$$\langle\psi_0^{(e)}|UU'|\psi_1\rangle = \langle\psi_0^{(e)}|\psi_0^{(e)}\psi_1\rangle - \frac{1}{c}\frac{2}{\sqrt{2^n}}. \quad (17)$$

Supposing

$$|\alpha\rangle = \frac{1}{\sqrt{2}}(|\psi_0^{(e)}\rangle + \mathrm{i}|\psi_1\rangle),$$

according to the Theorem 5 in Ref. [12], we see that

$$\begin{aligned} &|\mathrm{e}^{\mathrm{i}\omega'_0} - \langle\alpha|U'|\alpha\rangle| \\ &\le 2\left(\frac{3(n+1)^2}{n\cdot 2^{n+4}} + \frac{3(n+1)^2}{16c^2\cdot n\cdot\binom{n}{n/2+1}}\right) \\ &= O\left(\frac{n^{3/2}}{2^n}\right). \end{aligned} \quad (18)$$

The imaginary part is

$$\begin{aligned} \mathrm{Im}\langle\alpha|U'|\alpha\rangle &= \frac{1}{2}(\langle\psi_0{}^{(e)}|UU'|\psi_1\rangle - \langle\psi_1|UU'|\psi_0{}^{(e)}\rangle) \\ &= -\frac{1}{c}\frac{2}{\sqrt{2^n}}. \end{aligned} \quad (19)$$

On the other hand, because of

$$\begin{aligned} |\sin\omega'_0 - \mathrm{Im}\langle\alpha|U'|\alpha\rangle| &= |\mathrm{Im}(\mathrm{e}^{\mathrm{i}\omega'_0} - \langle\alpha|U'|\alpha\rangle)| \\ &\le |\mathrm{e}^{\mathrm{i}\omega'_0} - \langle\alpha|U'|\alpha\rangle| \\ &\le O\left(\frac{n^{3/2}}{2^n}\right), \end{aligned} \quad (20)$$

we obtain

$$\left|\sin\omega'_0 + \frac{2}{c\cdot\sqrt{2^n}}\right| \le O\left(\frac{n^{3/2}}{2^n}\right). \quad (21)$$

As ω'_0 tends to 0, $\sin\omega'_0 = \omega'_0 + O({\omega'_0}^3)$. We then have

$$-\frac{2}{c\cdot\sqrt{2^n}} - O\left(\frac{n^{3/2}}{2^n}\right) \le \omega'_0 \le -\frac{2}{c\cdot\sqrt{2^n}} + O\left(\frac{n^{3/2}}{2^n}\right). \quad (22)$$

This algorithm can be geometrically described through the derived expressions above. Starting with the initial state $|\psi_0{}^{(e)}\rangle$, we successively iterate t times,

$$\begin{aligned} (UU')^t|\psi_0{}^{(e)}\rangle &= \frac{1}{\sqrt{2}}(\mathrm{e}^{\mathrm{i}\omega'_0 t}|\omega'_0\rangle + \mathrm{e}^{-\mathrm{i}\omega'_0 t}|-\omega'_0\rangle) \\ &= \cos\omega'_0 t|\psi_0{}^{(e)}\rangle - \sin\omega'_0 t|\psi_1\rangle. \end{aligned} \quad (23)$$

Each iteration is equivalent to a rotation of angle $|\omega'_0|$ in the plane composed of the initial state

$$\begin{aligned} |\psi_0{}^{(e)}\rangle &= (1/\sqrt{2})(|\omega'_0\rangle + |-\omega'_0\rangle) \\ |\psi_1\rangle &= (\mathrm{i}/\sqrt{2})(|\omega'_0\rangle - |-\omega'_0\rangle) \end{aligned}$$

(see Fig. 1). Because $\langle\psi_0^{(e)}|\psi_1\rangle = 0$, the required number of iterations is

$$\frac{\pi}{2|\omega'_0|} = \frac{\pi}{4}\sqrt{2^n}.$$

After obtaining $|\psi_1\rangle$, the marked state $|R,0\rangle$ can be measured with the probability $p = |\langle R,0|\psi_1|^2 = 1/c^2$. Therefore, the success rate of the algorithm is $1 - O(1/(n+1))$. Both the number of iterations and the success rate agree with the conclusions in Ref. [13]. The geometric description of the optimized SKW algorithm has been presented.

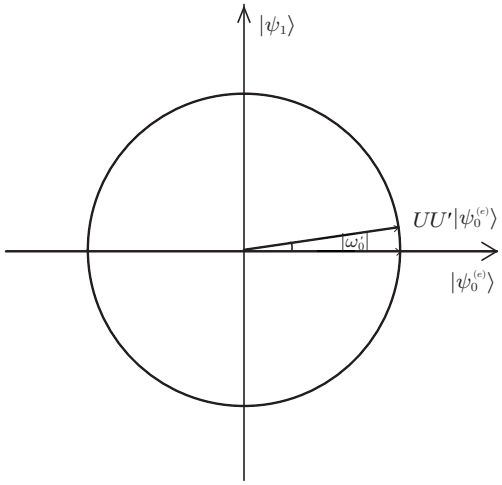


Fig. 1. Each iteration of the algorithm UU' can be viewed as a rotation in the plane composed of the initial state $|\psi_0{}^{(e)}\rangle$ and the final state $|\psi_1\rangle$. The rotation angle is $|\omega_0'|$.

4. Model of the algorithm with phase errors

4.1. Build the model

The model of the algorithm with phase errors can be built using the geometric description of the original algorithm. We assume a database size of 2^{n-1}. As the Grover operator imperfection in phase inversions of the algorithm is systematic, the explicit form of the inaccurate operator can be written as

$$\widetilde{C_0} = G = (1-\mathrm{e}^{\mathrm{i}\theta})|S^{\mathrm{C}}\rangle\langle S^{\mathrm{C}}| - I, \tag{24}$$

where $\theta = \pi + \delta$ and δ is a constant error. When $\delta = 0$, the inaccurate operator degenerates to the original operator. The coin operator with errors can then be written as

$$\widetilde{C'} = \widetilde{C_0} \otimes I + (C_1 - \widetilde{C_0}) \otimes |0\rangle\langle 0|. \tag{25}$$

The evolution operator becomes

$$\widetilde{U} = S \cdot \widetilde{C_0}, \tag{26}$$

$$\widetilde{U'} = S \cdot \widetilde{C'}. \tag{27}$$

Each iteration of the algorithm becomes $\widetilde{U}\widetilde{U'}$.

First, we analyze the eigenvalue spectrum of $\widetilde{U}\widetilde{U}$. The eigenvalue spectrum of $\widetilde{U}$ is already given in Ref. [18] with

$$\begin{aligned}\mathrm{e}^{\mathrm{i}\widetilde{\omega_k}} &= \frac{1}{2}(1-\mathrm{e}^{\mathrm{i}\theta}) - (1-\mathrm{e}^{\mathrm{i}\theta})\frac{k}{n} \\ &\mp \frac{1}{2}\sqrt{(1-\mathrm{e}^{\mathrm{i}\theta})^2\frac{(n-2k)^2}{n^2} + 4\mathrm{e}^{\mathrm{i}\theta}},\end{aligned} \tag{28}$$

$k \neq 0, k \neq n$. When $k = 0$, $\mathrm{e}^{\mathrm{i}\widetilde{\omega_0}} = -\mathrm{e}^{\mathrm{i}\theta} = \cos\delta + \mathrm{i}\sin\delta$. When $k = n$, $\mathrm{e}^{\mathrm{i}\widetilde{\omega_n}} = \mathrm{e}^{\mathrm{i}\theta}$. The introduction of errors leads to a δ-angle deviation between $\mathrm{e}^{\mathrm{i}\widetilde{\omega_0}}$ and $\mathrm{e}^{\mathrm{i}\omega_0} = 1$ in the polar coordinate plane (see Fig. 4.1 in Ref. [20]). The eigenvalue spectrum of $\widetilde{U}\widetilde{U}$ is $\mathrm{e}^{2\mathrm{i}\widetilde{\omega_k}}$. When $k = 0$, $\mathrm{e}^{2\mathrm{i}\widetilde{\omega_0}} = \mathrm{e}^{2\mathrm{i}\theta} = \cos 2\delta + \mathrm{i}\sin 2\delta$; therefore, there is a 2δ-angle deviation between $\mathrm{e}^{2\mathrm{i}\widetilde{\omega_0}}$ and $\mathrm{e}^{2\mathrm{i}\omega_0}$ in the polar coordinate plane. Eigenvalues are no longer complex conjugate pairs.

For the eigenvalue of $\widetilde{U}\widetilde{U'}$, we only need to analyze $\mathrm{e}^{\mathrm{i}\widetilde{\omega'_0}}$, $\mathrm{e}^{\mathrm{i}\widetilde{\omega'_1}}$, which originally correspond to the eigenvectors $|\omega'_0\rangle$ and $|-\omega'_0\rangle$ because they determine the initial state and the final state of the algorithm. Originally, these two eigenvalues are conjugate pairs. However, the introduction of errors leads to them being no longer conjugate. The difference between these two eigenvalues gradually increases when δ increases. $\mathrm{e}^{\mathrm{i}\widetilde{\omega'_0}}$ and $\mathrm{e}^{\mathrm{i}\widetilde{\omega'_1}}$ will gradually approach 1 and $\mathrm{e}^{\mathrm{i}2\theta}$ respectively according to Ref. [18]. If δ is very small, they are still in the vicinity of 1.

According to Theorem 1 and Ref. [18], the random walk that introduces phase errors can also be collapsed onto a random walk on the line. To redefine the operator, where

$$\widetilde{U'} = \widetilde{U} - (1-\mathrm{e}^{\mathrm{i}\theta})|L,1\rangle\langle R,0|, \tag{29}$$

the specific form of $\widetilde{U}$ has been redefined in Ref. [18].

Expanding eigenvalues of $\widetilde{U}\widetilde{U}$ to the second order with respect to δ and using Theorems 2 and 3 in Ref. [12], it can be proved that there are exactly two eigenvalues of $\widetilde{U}\widetilde{U'}$ with their real parts larger than

$$\begin{aligned}\widetilde{L} = 1 - \frac{1}{3}\Bigg[&\left(\frac{8}{n} - \frac{8}{n^2}\right) - \left(2 - \frac{4}{n}\right)\frac{\sqrt{n-1}}{n}\delta \\ &+ 3\left(\frac{1}{3} + \frac{2}{n^2} - \frac{2}{n}\right)\delta^2\Bigg].\end{aligned} \tag{30}$$

When $\delta = 0$ and n is replaced by $n+1$, equation (30) becomes equal to Eq. (13). Then, we evaluate

$$\begin{aligned}\widetilde{U}\widetilde{U'}|\psi_0{}^{(e)}\rangle = \mathrm{e}^{2\mathrm{i}\theta}|\psi_0{}^{(e)}\rangle - \frac{1-\mathrm{e}^{\mathrm{i}\theta}}{\sqrt{2^{n-1}}}\Bigg(&\frac{1-\mathrm{e}^{\mathrm{i}\theta}-n}{n}|R,0\rangle \\ &+ \frac{(1-\mathrm{e}^{\mathrm{i}\theta})\sqrt{n-1}}{n}|L,2\rangle\Bigg),\end{aligned} \tag{31}$$

obtaining

$$\begin{aligned}\langle\psi_0{}^{(e)}|\widetilde{U}\widetilde{U'}|\psi_0{}^{(e)}\rangle &= \mathrm{e}^{2\mathrm{i}\theta} - \frac{(\mathrm{e}^{\mathrm{i}\theta}-1)\mathrm{e}^{\mathrm{i}\theta}}{2^{n-1}} \\ &= \mathrm{e}^{2\mathrm{i}\delta} - \frac{(\mathrm{e}^{\mathrm{i}\delta}+1)\mathrm{e}^{\mathrm{i}\delta}}{2^{n-1}}.\end{aligned} \tag{32}$$

Similarly,

$$\begin{aligned}\widetilde{U}\widetilde{U'}|\psi_1\rangle = |\psi_1\rangle &- \frac{(1-\mathrm{e}^{\mathrm{i}\theta})^2}{c\cdot 4}\left(1-\frac{2}{n}\right)\frac{1}{\sqrt{\binom{n-1}{n/2-2}}} \\ &\times (|R,n/2-2\rangle + |L,n/2+2\rangle) \\ &- \frac{1}{c}\left(\frac{1-\mathrm{e}^{\mathrm{i}2\theta}}{4} + \frac{(1-\mathrm{e}^{\mathrm{i}\theta})^2}{2n}\right) \\ &\times \frac{1}{\sqrt{\binom{n-1}{n/2-1}}}(|R,n/2\rangle + |L,n/2\rangle),\end{aligned} \tag{33}$$

and

$$\langle\psi_1|\widetilde{U}\widetilde{U'}|\psi_1\rangle = 1-\frac{(e^{i\theta}-1)e^{i\theta}}{2c^2\binom{n-1}{n/2}} = 1-\frac{(e^{i\delta}+1)e^{i\delta}}{2c^2\binom{n-1}{n/2}}. \quad (34)$$

From Eqs. (32) and (34), it can be inferred that, when n is big enough, $|\psi_0{}^{(e)}\rangle$ is an eigenvector of $\widetilde{U}\widetilde{U'}$ with eigenvalue close to $e^{2i\delta}$, and $|\psi_1\rangle$ is an eigenvector of $\widetilde{U}\widetilde{U'}$ with eigenvalue close to 1. Because $e^{2i\delta}=\cos 2\delta+i\sin 2\delta$, its real part can be expanded by Taylor expansion to $\cos 2\delta = 1-2\delta^2$. Based on the above analysis, when the error is small, the algorithm can still be viewed as rotation in the two-dimensional plane defined by Eqs. (7) and (8).

4.2. Analysis of the algorithm

With the error increasing gradually, $|\widetilde{\omega'}_0\rangle$ and $|\widetilde{\omega'}_1\rangle$ are no longer conjugate. They cannot be expressed as the linear combination of Eqs. (14) and (15); therefore, the algorithm cannot rotate from $|\psi_0{}^{(e)}\rangle$ to $|\psi_1\rangle$ completely, and the probability to obtain $|\psi_1\rangle$ finally becomes small.

Expanding $|\psi_0{}^{(e)}\rangle$ and $|\psi_1\rangle$ with $|\widetilde{\omega'}_0\rangle$ and $|\widetilde{\omega'}_1\rangle$ respectively, we obtain

$$|\psi_0{}^{(e)}\rangle = a_0|\widetilde{\omega'}_0\rangle + a_1|\widetilde{\omega'}_1\rangle + \sqrt{1-a_0^2-a_1^2}|\gamma_0\rangle, \quad (35)$$

$$|\psi_1\rangle = b_0|\widetilde{\omega'}_0\rangle + b_1|\widetilde{\omega'}_1\rangle + \sqrt{1-b_0^2-b_1^2}|\gamma_1\rangle, \quad (36)$$

where $|\gamma_0\rangle$ and $|\gamma_1\rangle$ are the normalized vectors orthogonal to $|\widetilde{\omega'}_0\rangle$ and $|\widetilde{\omega'}_1\rangle$. a_0, a_1, b_0, b_1 are the complex numbers, determined depending on δ and n. Evaluating

$$\begin{aligned} &1-\frac{1}{2^{n-2}}-\delta^2\left(2-\frac{5}{2^n}\right)\\ &= \mathrm{Re}(\langle\psi_0{}^{(e)}|\widetilde{U}\widetilde{U'}|\psi_0{}^{(e)}\rangle)\\ &= \sum_j \cos\widetilde{\omega}_j|\langle\widetilde{\omega}_j|\psi_0^{(e)}\rangle^2|\\ &= a_0^2\cos\widetilde{\omega}_0 + a_1^2\cos\widetilde{\omega}_1 + \sum_{j\neq 0,1}\cos\widetilde{\omega}_j|\langle\widetilde{\omega}_j|\psi_0^{(e)}\rangle^2|\\ &< a_0^2+a_1^2+\widetilde{L}(1-a_0^2-a_1^2), \end{aligned} \quad (37)$$

we obtain

$$a_0^2+a_1^2 > 1-\frac{1}{1-\widetilde{L}}\left[\frac{1}{2^{n-2}}+\left(2-\frac{5}{2^n}\right)\delta^2\right]. \quad (38)$$

Similarly, evaluating $\mathrm{Re}\langle\psi_1|\widetilde{U}\widetilde{U'}|\psi_1\rangle$, we obtain

$$b_0^2+b_1^2 > 1-\frac{1}{1-\widetilde{L}}\left(\frac{4-5\delta^2}{4c^2\binom{n-1}{n/2}}\right). \quad (39)$$

Now, the algorithm with phase errors can be geometrically described using the expressions derived above. Applying the operator on the initial state $|\psi_0{}^{(e)}\rangle$ successively t times, we get

$$\begin{aligned} &(\widetilde{U}\widetilde{U'})^t|\psi_0{}^{(e)}\rangle\\ &= a_0 e^{i\widetilde{\omega'}_0 t}|\widetilde{\omega'}_0\rangle + a_1 e^{i\widetilde{\omega'}_1 t}|\widetilde{\omega'}_1\rangle + \sqrt{1-a_0^2-a_1^2}|\gamma_0'\rangle\\ &= \frac{a_0b_1 e^{i\widetilde{\omega'}_0 t}-a_1b_0 e^{i\widetilde{\omega'}_1 t}}{a_0b_1-a_1b_0}(|\psi_0{}^{(e)}\rangle-\sqrt{\varepsilon_0}|\gamma_0\rangle)\\ &\quad - \frac{a_0a_1(e^{i\widetilde{\omega'}_0 t}-e^{i\widetilde{\omega'}_1 t})}{a_0b_1-a_1b_0}(|\psi_1\rangle-\sqrt{\varepsilon_1}|\gamma_1\rangle)\\ &\quad + \sqrt{\varepsilon_0}|\gamma_0'\rangle, \end{aligned} \quad (40)$$

where

$|\gamma_0'\rangle = (\widetilde{U}\widetilde{U'})^t|\gamma_0\rangle$, $\sqrt{1-a_0^2-a_1^2}=\sqrt{\varepsilon_0}$, $\sqrt{1-b_0^2-b_1^2}=\sqrt{\varepsilon_1}$. Hence, the amplitude of $|\psi_1\rangle$ is

$$w = \frac{a_0a_1(e^{i\widetilde{\omega'}_0 t}-e^{i\widetilde{\omega'}_1 t})}{a_0b_1-a_1b_0}, \quad (41)$$

which is determined by eigenvalues $e^{i\widetilde{\omega'}_0}$, $e^{i\widetilde{\omega'}_1}$, iteration times t, and coefficients a_0, a_1, b_0, b_1.

When δ=0, $e^{i\widetilde{\omega'}_0} = e^{i\omega'_0}$, $e^{i\widetilde{\omega'}_1} = e^{-i\omega'_0}$, $a_0 = a_1 = 1/\sqrt{2}$, $b_0 = -b_1 = i/\sqrt{2}$, and $\varepsilon_0, \varepsilon_1 \to 0$. This implies that $w = \sin\omega_0 t$, which in turn indicates that $|\psi_0{}^{(e)}\rangle$ can finally finish the complete rotation to $|\psi_1\rangle$. The probability to obtain $|\psi_1\rangle$ is nearly 1. When $\delta \neq 0$, $\varepsilon_0, \varepsilon_1$ are no longer close to 0 and there is a state other than $|\psi_1\rangle$, indicating that the probability to obtain $|\psi_1\rangle$ will decrease. On the other hand, because $\langle\psi_1|\psi_0^{(e)}\rangle = 0$, $|\psi_0{}^{(e)}\rangle$ needs to rotate by an angle of $\pi/2$ in the plane defined by $|\psi_0{}^{(e)}\rangle$ and $|\psi_1\rangle$ in theory, which is equivalent to making $|\widetilde{\omega'}_0 t - \widetilde{\omega'}_1 t| = \pi$. When δ increases, the difference between eigenvalues $\widetilde{\omega'}_0 - \widetilde{\omega'}_1$ will increase. Therefore, the number of iterations t to reach maximum w will decrease. The presence of e^{it} in Eq. (41) shows that it is a periodic function; therefore, when the algorithm is in the presence of phase errors, its success rate will still periodically change depending on the number of iterations.

The above analysis shows that when the algorithm has phase errors, the success rate of the algorithm and the required number of iterations will decrease.

5. Numerical simulations and robustness comparison

The relationship between the size of the database, the phase error, the number of iterations, and the amplitude of the marked state was shown through the analysis described in the preceding section. We also analyzed the effects of the phase error on the algorithm. In this section, we will analyze the impact of phase errors on the success rate and the number of iterations by using numerical simulations, and demonstrate the

quantitative relationship between the size of the database, the phase error, the number of iterations, and the success rate.

First, we simulate the relation between the number of iterations and the success rate when $n = 8$. We compare the two cases where $\delta = 0$ and $\delta = 0.2$, as shown in Fig. 2.

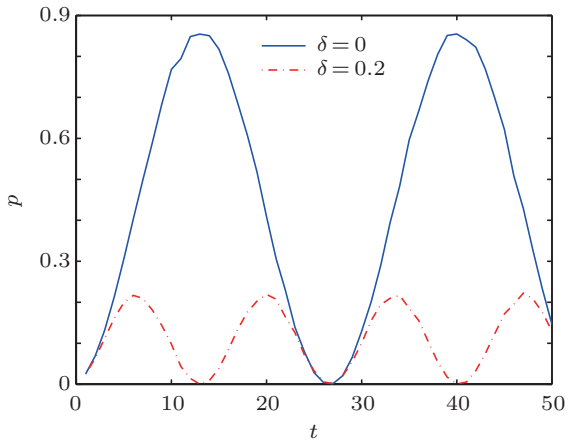


Fig. 2. (color online) Relationship between the success rate of the algorithm p and the number of iterations t when $n = 8$. The solid line represents $\delta = 0.2$ and the dotted line represents $\delta = 0$.

The solid line indicates that the algorithm without errors iterates 13 times to reach the maximum probability. The dotted line indicates that when the error is 0.2, the algorithm requires 6 iterations to reach the maximum probability, which is much smaller than the maximum probability obtained in the previous case. Both success rates change periodically with the number of iterations. The simulation results are consistent with the theoretical analysis in Subsection 4.2.

Next, we study the relationship between the maximum success rate of the algorithm, the database size, and the phase error. Each curve represents different phase errors, with $\delta = 0$, $\delta = 10^{-2}$, $\delta = 10^{-3}$, and $\delta = 10^{-4}$, as shown in Fig. 3.

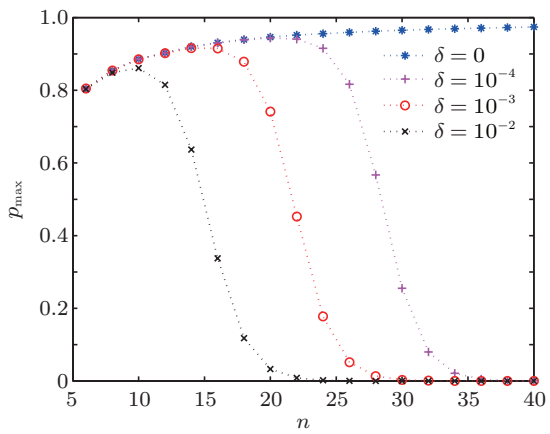


Fig. 3. (color online) Maximum success rate of the algorithm $p_{\max}$ versus the scale of the database n with different phase errors. Different lines show results for different values of δ.

Figure 3 shows that the maximum success rate increases with increasing database size. However, once it reaches a particular threshold, the value decreases exponentially. The larger the error, the smaller the database scale when the maximum success rate begins to decrease. With curve fitting, we obtain a relationship between the maximum success rate $p_{\max}$, the size of the database 2^n, and the phase error δ,

$$p_{\max} = \frac{3.8 p_0}{3.8 + \delta^2 2^n}, \tag{42}$$

p_0 is the success rate of the algorithm without errors. It can be obtained from Section 4 that

$$p_0 = \frac{1}{\sum_{x=0}^{n/2-1} \frac{1}{\binom{n-1}{x}}} = \frac{1}{c^2} \to 1. \tag{43}$$

Combining Eqs. (42) and (43), we obtain

$$p_{\max} = \frac{19}{c^2(19 + 5\delta^2 \cdot 2^n)}. \tag{44}$$

It can be seen from Eqs. (44) that, before the search, to ensure the success rate of the algorithm, we need to predict the size of the phase error to limit the size of the database. For example, if the requirement of the success rate is not less than 0.5, the database size cannot be more than $3.8/\delta^2$.

Next, we study the relationship between the iteration times required to reach the maximum success rate, the size of the database, and the phase error. The corresponding simulation results are shown in Fig. 4.

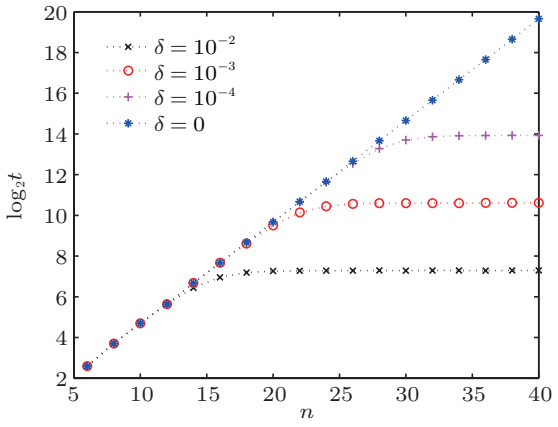


Fig. 4. (color online) Scale of the required iteration times $\log_2 t$ versus scale of database n with different phase errors. The different lines show results for different values of δ.

Figure 4 shows that the required number of iterations t_0 increases with increasing database size. Beyond a particular threshold for a given error, the required number of iterations remains the same; this threshold database scale gets smaller as the errors become larger. Using curve fitting, we obtain the relationship between iteration times t_0, the size of the database 2^n, and the phase error δ,

$$t_0 = \frac{\pi}{\sqrt{16/2^n + 4\delta^2}}. \tag{45}$$

From the above analysis, it can be inferred that, when phase errors are present, the maximum success rate of the algorithm and the required number of iterations are the same as in the original algorithm as long as the database scale does not exceed a critical value, which suggests that this algorithm has tolerance to this sort of error. Comparing Fig. 3 with Fig. 4 for a given error level, we find that the corresponding database scale when $p_{\max}$ begins to decrease is smaller than that when t_0 begins to remain constant; therefore, it is clear that $p_{\max}$ is more sensitive to the influence of the error. We define the value of n when $p_{\max}$ begins to decrease as the critical value. When $p_{\max}$ begins to decrease, we obtain the corresponding (δ, n) using numerical simulations. The expression relating the database scale n to the phase error δ is

$$n = 1.806 \cdot \log_2^{0.4642/\delta}. \tag{46}$$

Therefore, when the algorithm is in the presence of errors, the results are almost unaffected as long as the database scale satisfies the relation $n \leq 1.806 \cdot \log_2^{0.4642/\delta}$.

Now, we respectively obtain the expressions for maximum success rate $p_{\max}$ and the required iteration times t_0 as a function of the phase error δ and the database size 2^n. From the analysis of Subsection 4.2, the probability to obtain the marked state p is a periodic function, which can also be confirmed from Fig. 2. Thereby, making use of Eqs. (44) and (45), we obtain an expression for the database size 2^n, the phase error δ, the number of iterations t, and the success rate p,

$$p = \frac{19}{c^2(19+5\delta^2 \cdot 2^n)} \sin^2\left(\sqrt{\frac{4}{2^n}+\delta^2} \cdot t\right). \tag{47}$$

According to Eq. (47), for a given database size and expected error, we can calculate the number of iterations required to obtain a target success rate. This improves the efficiency of the algorithm, as shown in Fig. 5 for $n = 8$. When $\delta = 0$ or $\delta = 0.2$, the results are the same as those in Fig. 2.

Finally, we make a simple comparison of robustness against phase errors between the optimized SKW algorithm and Grover's algorithm. Probability gap is defined as the probability difference between the marked state and the maximum rest state,[21] and is given by

$$\Delta p = p_{\text{out}} - \max(p_2, p_3, \ldots, p_{N-1}). \tag{48}$$

In practice, as long as there is a sufficient probability gap, the probability of finding the marked state is much higher than other states at the end of the algorithm, and the algorithm is still successful. If the algorithmic probability gap variance is small regardless of the presence of errors, the robustness of the algorithm is strong; therefore, measuring the difference between probability gaps before and after the introduction of errors enables us to compare algorithmic robustness.

According to Ref. [20], the relationship between the database size, the phase error, and the maximum success rate is given in Grover's algorithm,

$$p_{\max} = \frac{4}{4+\delta^2 2^n}. \tag{49}$$

Therefore, the probability gap of Grover's algorithm is

$$\Delta p_2 = p_{\max} - \frac{1-p_{\max}}{2^n - 1}. \tag{50}$$

First, we simulate the probability gap for the optimized SKW algorithm Δp_1 with different phase errors. Figure 6 shows the result of this simulation. We then simulate the difference between the probability gaps $\Delta p_1 - \Delta p_2$, shown in Fig. 7.

The probability gap of the optimized SKW algorithm decreases exponentially with increase in errors, as shown in Fig. 6. However, for the same database scale, the difference between the probability gaps increases gradually, as shown in Fig. 7. These data indicate that the probability gap of Grover's algorithm decreases faster, proving that the optimized SKW algorithm is more robust.

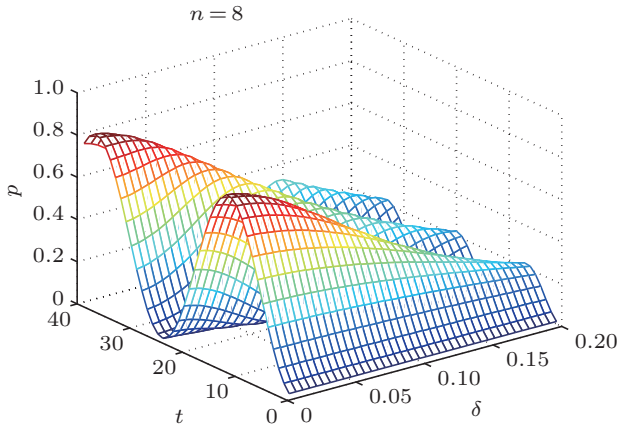


Fig. 5. (color online) Three-dimensional graph of Eq. (47) when $n = 8$. The x axis shows the number of iterations t. The y axis is the phase error δ and the z axis is the success rate of the algorithm p.

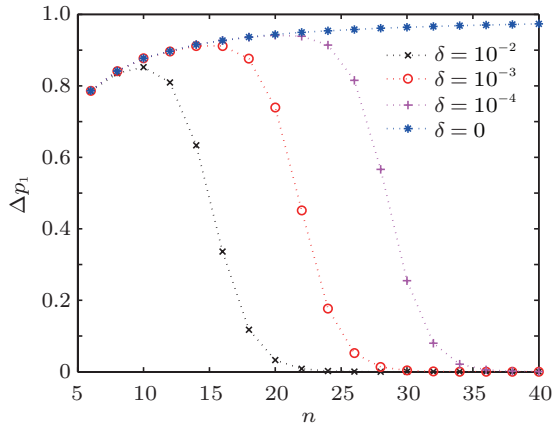


Fig. 6. (color online) Probability gap of the optimized SKW algorithm Δp_1 versus the scale of the database n with different phase errors. The different lines show results for different values of δ.

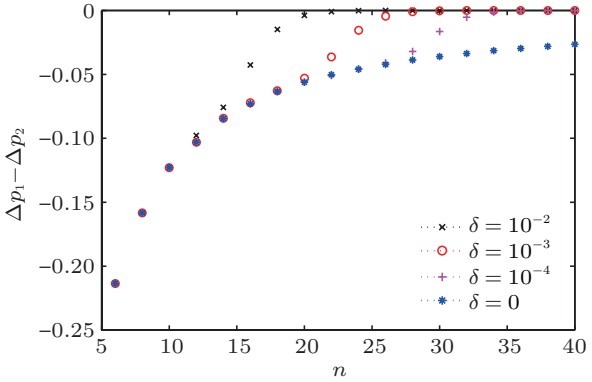


Fig. 7. (color online) Difference between the probability gaps $\Delta p_1 - \Delta p_2$ versus the scale of the database n with different phase errors. The different lines show results for different values of δ.

6. Conclusion

This study investigates the effects of systematic errors in phase inversions on the optimized SKW algorithm in detail. We first restate the optimized SKW as a geometric description, enabling the construction of the model of this algorithm with phase errors. Phase errors are shown to reduce both the number of iterations and the success rate of the algorithm through theoretical analysis and numerical simulations. Larger errors lead to a smaller maximum success rate of the algorithm and fewer number of required iterations. We derive relationships for both the maximum success rate of the algorithm and the required iteration times as a function of the phase error and the database size. Subsequently, we obtain a critical value for the database scale n. When $n \leq 1.806 \cdot \log_2^{0.4642/\delta}$, the algorithm is almost unaffected by errors. Based on the above relationships, we depict the relationship between the success rate of the algorithm, the phase error, the number of iterations, and the database size. Finally, analyses and numerical simulations show that the optimized SKW algorithm is more robust than Grover's algorithm.

References

[1] Grover L 1996 *Proceedings of the* 28*th Annual ACM Symposium on Theory of Computing* (New York: ACM Press) p. 212
[2] Long G L 2001 *Phys. Rev. A* **64** 022307
[3] Li T, Bao W S, Lin W Q, Zhang H and Fu X Q 2014 *Chin. Phys. Lett.* **31** 050301
[4] Wang X, Bao W S and Fu X Q 2011 *Chin. Sci. Bull.* **56** 484
[5] Chuang I L, Gershenfeld N and Kubinec M 1998 *Phys. Rev. Lett.* **80** 3408
[6] Zhang J F, Lu Z H, Deng Z W and Shan L 2003 *Chin. Phys.* **12** 700
[7] Zheng S B 2005 *Chin. Phys.* **14** 2222
[8] Sun J, Lu S F, Liu F and Yang L P 2012 *Chin. Phys. B* **21** 010306
[9] Zhang Y Y, Hu H P and Lu S F 2014 *Chin. Phys. B* **23** 040309
[10] Long G L, Li Y S, Zhang W L and Tu C C 2000 *Phys. Rev. A* **61** 042305
[11] Shenvi N, Brown K R and Whaley K B 2003 *Phys. Rev. A* **68** 052313
[12] Shenvi N, Kempe J and Whaley K B 2003 *Phys. Rev. A* **67** 052307
[13] Potocek V, Gábris A, Kiss T and Jex I 2009 *Phys. Rev. A* **79** 012325
[14] Childs A M and Goldstone J 2004 *Phys. Rev. A* **70** 022314
[15] Ambainis A 2008 *SIAM J. Comput.* **37** 210
[16] Tulsi A 2008 *Phys. Rev. A* **78** 012310
[17] Magniez F, Nayak A, Roland J and Santha M 2011 *SIAM J. Comput.* **40** 142
[18] Li Y, Ma L and Zhou J 2006 *J. Phys. A: Math. Gen.* **39** 9309
[19] Moore C and Russell A 2002 *Randomization and Approximation Techniques in Computer Science* (Berlin/Heidelberg: Springer) p. 164
[20] Li Y 2006 "Investigations on Quantum Random-Walk Search Algorithm", MS Thesis (Shanghai: East China Normal University) p. 34 (in Chinese)
[21] Ma L, Du J F, Li Y and L H 2006 *Chin. Phys. Lett.* **23** 779