# Grover's algorithm with errors

Andris Ambainis, Artūrs Bačkurs, Nikolajs Nahimovs, Alexander Rivosh *

Faculty of Computing, University of Latvia, Raina bulv. 19, Riga, LV-1586, Latvia.

**Abstract.** Grover's algorithm is a quantum search algorithm solving the unstructured search problem of size $n$ in $O(\sqrt{n})$ queries, while any classical algorithm needs $O(n)$ queries [3].
However, if query has some small probability of failing (reporting that none of the elements are marked), then quantum speed-up disappears: no quantum algorithm can be faster than a classical exhaustive search by more than a constant factor [8].
We study the behaviour of Grover's algorithm in the model there query may report *some* marked elements as unmarked (each marked element has its own error probability, independent of other marked elements).
We analyse the limiting behaviour of Grover's algorithm for a large number of steps and prove the existence of limiting state $\rho_{lim}$. Interestingly, the limiting state is independent of error probabilities of individual marked elements. If we measure $\rho_{lim}$, the probability of getting one of the marked states $i_1, \ldots, i_k$ is $\frac{k}{k+1}$. We show that convergence time is $O(n)$.

## 1 Introduction

Grover's algorithm is a quantum search algorithm solving the unstructured search problem. The algorithm works in the following model. We have an unstructured search space of $n$ elements in which some elements have a certain property. We call these elements *marked*. We are given a procedure (*an oracle*) for checking whether an element is marked. This procedure is given as a black box that answers queries. It receives $i$ and answers whether the $i^{\text{th}}$ element is marked. In the quantum case, the algorithm is allowed to input superposition consisting of multiple $i$.

Grover's algorithm solves the unstructured search problem in $O(\sqrt{n})$ queries. It is known that any deterministic or randomized algorithm needs linear time (number of queries) to solve the above problem. Thus, Grover's algorithm provides a significant speed-up over any classical algorithm.

There has been a number of papers studying Grover's algorithm in the presence of errors of various forms. Regev and Schiff have shown [8] that if query has some small probability of failing (reporting that none of the elements are

* AA, NN and AR are supported by the European Social Fund within the project 2009/0216/1DP/1.1.1.2.0/09/APIA/VIAA/044. AB is supported by FP7 FET-Open project QCS.

marked), then quantum speed-up disappears: no quantum algorithm can be faster than a classical exhaustive search by more than a constant factor.

In this paper we study the behaviour of Grover's algorithm in the model there query may report *some* marked elements as unmarked. In our case each marked element has its own probability of failing, independent of other marked elements. We assume that faults are one-sided. That is, if the $i^{\text{th}}$ element is not marked, the black box always answers that it is not marked. If the $i^{\text{th}}$ element is marked, the black box may give the correct answer (with probability $1 - p_i$) or mistakenly answer that the element is not marked (with probability $p_i$).

Given the importance of Grover's algorithm, we think that it is interesting to find out what exactly happens if we run Grover's algorithm in this model.

Let $k$ be the number of marked elements. We show that if Grover's algorithm is run for a large number of steps, then the state of the algorithm converges to a mixed state that is a mixture of $|i\rangle$ for each marked $i$ with probability $\frac{1}{k+1}$ each and the uniform superposition of all non-marked elements with probability $\frac{1}{k+1}$. Surprisingly, the final state is independent of the error probabilities of different marked elements. Initially, the probabilities of finding the elements with higher probabilities of correct answer grow faster but, in the limit for a large number of steps the probabilities of finding all elements $i$ converge to the same value $\frac{1}{k+1}$.

We also quantify the speed of convergence: it happens in $O(n)$ steps. This matches the lower bound of [8][1].

**Related work.** The work of Regev and Schiff [8] mentioned above is the paper that is most closely related to our work.

Several authors [5, 9, 10] have studied the effect of random imperfections in either diffusion transformation or black box query on the performance of Grover's algorithm, showing that such type of noise can completely destroy the advantage of Grover's algorithm over classical exhaustive search. The difference between their work and our work is that they consider small random imperfections that occur in every step of the algorithm while we consider the case there query is performed correctly for some marked elements and not performed at all for others.

Buhrman et al. [2] have looked at a *coherent noise* model in which the algorithm has access to a set of unitary procedures $A_i$ that check whether the $i^{\text{th}}$ element is marked and have some probability of error. The algorithm is allowed to run both $A_i$ and $A_i^{-1}$ multiple times. This model is sufficiently general to enable a fault-tolerant computation and allows to simulate any noise-free quantum algorithm that makes $T$ queries by a noisy algorithm that makes $O(T \log T)$ queries. In some cases, a constant overhead instead of a logarithmic one is sufficient. The difference between coherent noise and our models is that in coherent noise model the state after query is still a pure state, while in our model query leads to a mixes state.

[1] Technically, the lower bound of [8] is for a slightly different model but the difference between the models is not important in this case.

## 2 Technical preliminaries

We use the standard notions of quantum states, density matrices, etc., as described in [6] or [7].

**Grover's algorithm**[3]

Suppose we have an unstructured search space of size $n$. Grover's algorithm starts with a starting state $|\psi_{start}\rangle = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} |i\rangle$. Each step of the algorithm consists of two transformations: $Q$ and $D$. Here, $Q$ is a query to a black box defined by

- $Q|i\rangle = -|i\rangle$ if $i$ is a marked element;
- $Q|i\rangle = |i\rangle$ if $i$ is not a marked element.

$D$ is the diffusion transformation described by the following $n \times n$ matrix:

$$D = \begin{pmatrix} -1+\frac{2}{n} & \frac{2}{n} & \dots & \frac{2}{n} \\ \frac{2}{n} & -1+\frac{2}{n} & \dots & \frac{2}{n} \\ \dots & \dots & \dots & \dots \\ \frac{2}{n} & \frac{2}{n} & \dots & -1+\frac{2}{n} \end{pmatrix}.$$

We refer to $|\psi_t\rangle = (DQ)^t |\psi_{start}\rangle$ as the state of Grover's algorithm after $t$ time steps.

Grover's algorithm has been analysed in detail and many facts about it are known [1]. If there is one marked element $i$, the probability of finding it by measuring $|\psi_t\rangle$ reaches $1 - o(1)$ for $t = O(\sqrt{n})$. If there are $k$ marked elements, the probability of finding one of them by measuring $|\psi_t\rangle$ reaches $1 - o(1)$ for $t = O(\sqrt{n/k})$.

**Frobenius norm**[4]

Let $\rho = (\rho_{ij})$ be an $n \times n$ matrix. The *Frobenius norm* (also called *Euclidean norm* or $l_2$*-norm*) of $\rho$ is defined as

$$\|\rho\|_F = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{n} |\rho_{ij}|^2}.$$

Frobenius norm is unitarily invariant: if $U$ unitary, then $\|U\rho\|_F = \|\rho\|_F = \|\rho U\|_F$ [4, chapter 5.6]. Also, $\|\rho\|_F \geq 0$ and $\|\rho_1 + \rho_2\| \leq \|\rho_1\| + \|\rho_2\|$, as for any matrix or vector norm.

## 3 Grover's algorithm with errors

We assume that a search space of size $n$ contains $k$ marked elements $i_1, i_2, \dots, i_k$. In each step, instead of the correct query $Q$, we apply a faulty query (*faulty oracle*) $Q'$ defined as follows:

- $Q'|i_j\rangle = |i_j\rangle$ with probability $p_j$;
- $Q'|i_j\rangle = -|i_j\rangle$ with probability $1 - p_j$;

– $Q'|i\rangle = |i\rangle$ if $i$ is not a marked element.

For different elements $i_j$, faults occur independently one from another. Also, for different steps faults are independent.

We show

**Theorem 1** *Let $\rho_t$ be the density matrix of state of Grover's algorithm with a faulty oracle after $t$ queries. Then, the sequence $\rho_1, \rho_2, \ldots$ converges to*

$$\rho_{lim} = \frac{1}{k+1} \sum_{j=1}^{k} |i_j\rangle\langle i_j| + \frac{1}{k+1} |\phi\rangle\langle\phi|$$

*where $|\phi\rangle = \frac{1}{\sqrt{n-k}} \sum_{i \neq i_j} |i\rangle$ is the uniform superposition over all non-marked $i$.*

If we measure $\rho_{lim}$, the probability of getting one of the marked states $i_1, \ldots, i_k$ is $\frac{k}{k+1}$. Interestingly, the final state is independent of the error probabilities $p_1, \ldots, p_k$. Initially the probabilities of finding the elements with higher probabilities of correct answer grow faster but, in the limit for a large number of steps, the probabilities of finding all elements $i_j$ converge to the same value $\frac{1}{k+1}$.

The next result quantifies the speed of convergence to the limiting state $\rho_{lim}$.

**Theorem 2** *Assume that errors occur with the same probability $p_1 = \ldots = p_k = p$ for all marked elements. Then, for every $\epsilon > 0$, there exists $t = O(n)$ such that if we run Grover's algorithm with a faulty oracle for $t$ steps and measure the result, we get one of the marked elements with probability in $[\frac{k}{k+1} - \epsilon, \frac{k}{k+1} + \epsilon]$.*

## 4 Limiting behaviour of Grover's algorithm with errors

In this section we will study limiting behaviour of Grover's algorithm with errors for large number of steps and will prove the Theorem 1.

Consider the density matrix $\rho_t$ of the quantum state of Grover's algorithm after $t$ queries. Due to symmetry, we can assume that the first $k$ basis states correspond to the marked elements. Note that Grover's algorithm acts in the same way on all unmarked elements. Therefore, the state of the algorithm is a probabilistic mixture of pure states of the form

$$\alpha_1|1\rangle + \ldots + \alpha_k|k\rangle + \sum_{i=k+1}^{n} \beta|i\rangle, \tag{1}$$

with the amplitudes of all unmarked states being equal. The density matrix $\rho_t$, then, takes the form

$$\rho_t = \begin{bmatrix} a_1 & b_{1,2} & b_{1,3} & \dots & c_1 & \dots & c_1 \\ b_{1,2} & a_2 & b_{2,3} & \dots & \vdots & \ddots & \vdots \\ b_{1,3} & b_{2,3} & a_3 & \dots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & c_k & \dots & c_k \\ c_1 & \dots & \dots & c_k & d & \dots & d \\ \vdots & \ddots & \ddots & \vdots & \vdots & \ddots & \vdots \\ c_1 & \dots & \dots & c_k & d & \dots & d \end{bmatrix}$$

because the density matrix for every pure state (1) in the mixture $\rho_t$ is of this form.

Let $p_i$ be the error probability for the $i^{\text{th}}$ marked element. The effect of the faulty query $Q'$ on the density matrix $\rho_t$ is:

$$\begin{array}{rl} a_i & \mapsto a_i \\ b_{i,j} & \mapsto (2p_i - 1)(2p_j - 1)b_{i,j} \\ c_i & \mapsto (2p_i - 1)c_i \\ d & \mapsto d \end{array}. \tag{2}$$

Let us prove $b_{i,j} \mapsto (2p_i - 1)(2p_j - 1)b_{i,j}$. Consider the corresponding entry $(Q'\rho_t Q')_{ij}$ of the density matrix, after the faulty query $Q'$ is applied. If $Q'$ changes the sign of either $|i\rangle$ or $|j\rangle$, the entry is equal to $-b_{ij}$. This happens with probability $p_i(1-p_j) + p_j(1-p_i)$. If $Q'$ changes the sign of both $|i\rangle$ and $|j\rangle$ or none of them, the entry is equal to $b_{ij}$. This happens with probability $p_i p_j + (1-p_i)(1-p_j)$. Hence,

$$(Q'\rho_t Q')_{ij} = -b_{ij}(p_i(1-p_j) + p_j(1-p_i)) + b_{ij}(p_i p_j + (1-p_i)(1-p_j)) =$$

$$= (1-2p_i)(1-2p_j)b_{ij}.$$

Similarly, we can prove that $c_i \mapsto (2p_i - 1)c_i$, $a_i \mapsto a_i$ and $d \mapsto d$.

Consider the Frobenius norm of the density matrix. If we multiply the density matrix by the unitary diffusion matrix, its Frobenius norm does not change. Since the faulty query transformation decreases the Frobenius norm (if $0 < p_i < 1$) and the Frobenius norm takes non-negative values, the $\lim_{t\to\infty} \|\rho_t\| = C$ exists.

If $\lim_{t\to\infty} b_{i,j} \neq 0$ we obtain a contradiction, because the Frobenius norm decreases infinitely. Analogously, we can prove that $\lim_{t\to\infty} c_i = 0$.

Let us prove $\lim_{t\to\infty}(a_i - a_j) = 0$ for each $i \neq j$. Assume it is not true, i.e. there exist $i \neq j$ and $\delta > 0$ so that $|a_i - a_j| > \delta$ for infinitely many $t$. Consider $t'$ so that for all $t > t'$ and all $m$, $l$ inequalities $b_{m,l} < \epsilon$ and $c_m < \epsilon$ hold. After

right multiplying the density matrix by the diffusion matrix

$$\rho_t D = \begin{bmatrix} a_1 & \dots & O(\epsilon) & O(\epsilon) & \dots & O(\epsilon) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ O(\epsilon) & \dots & a_k & O(\epsilon) & \ddots & O(\epsilon) \\ O(\epsilon) & \dots & O(\epsilon) & d & \dots & d \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ O(\epsilon) & \dots & O(\epsilon) & d & \dots & d \end{bmatrix} \begin{bmatrix} -1+\frac{2}{n} & \frac{2}{n} & \dots & \frac{2}{n} \\ \frac{2}{n} & -1+\frac{2}{n} & \dots & \frac{2}{n} \\ \dots & \dots & \dots & \dots \\ \frac{2}{n} & \frac{2}{n} & \dots & -1+\frac{2}{n} \end{bmatrix},$$

the last column contains values $\frac{2a_1}{n} + O(\epsilon)$, ..., $\frac{2a_k}{n} + O(\epsilon)$ and $\frac{d(n-2k)}{n} + O(\epsilon)$ ($n-k$ times). After left multiplying this matrix by the diffusion matrix, each of the first $k$ elements in the last column takes the value $2v - \frac{2a_i}{n} + O(\epsilon)$, where $v$ is the arithmetic mean of the last column of $\rho_t D$. We obtain a contradiction by choosing a sufficiently small $\epsilon$, because at least two of these values differ by at least $\frac{2\delta}{n} + O(\epsilon)$.

For an arbitrary $\epsilon$ we can choose $t'$ so that for every $t > t'$ the inequalities $b_{m,l} < \epsilon$, $c_m < \epsilon$ and $|a_m - a_l| < \epsilon$ hold for all $m$ and $l$. Since $a_1 + \ldots + a_k + d(n-k) = 1$ (a property of the density matrix), it follows that $a_i = \frac{1-d(n-k)}{k} + O(\epsilon)$. So, the arithmetic mean of the last column of $\rho_t D$ is

$$v = \frac{2(a_1 + \ldots + a_k) + d(n-2k)(n-k)}{n^2} + O(\epsilon) =$$

$$= \frac{2 + d(n-2k-2)(n-k)}{n^2} + O(\epsilon).$$

After left and right multiplying the density matrix by the diffusion matrix, the last column's $i$-th value is

$$2v - \frac{2a_i}{n} + O(\epsilon) = 2v - \frac{2 - 2d(n-k)}{nk} + O(\epsilon) =$$

$$= \frac{4 + 2d(n-2k-2)(n-k)}{n^2} - \frac{2 - 2d(n-k)}{nk} + O(\epsilon) =$$

$$= \frac{2(n-2k)(d(k+1)(n-k)-1)}{kn^2} + O(\epsilon).$$

Since this sum must be $O(\epsilon)$, it follows that $d(k+1)(n-k) - 1 = O(\epsilon)$, assuming $n \neq 2k$. Choosing $\epsilon$ arbitrarily small, we obtain $\lim_{t\to\infty} d = \frac{1}{(k+1)(n-k)}$ and $\lim_{t\to\infty} a_i = \frac{1}{k+1}$. □

## 5 Convergence speed of Grover's algorithm with errors

In this section we will study how fast Grover's algorithm with errors converges to its limiting state and will prove the Theorem 2.

We describe the quantum state of Grover's algorithm after $t$ queries by the density matrix

$$\rho_t = \begin{bmatrix} a_1 & b_{1,2} & b_{1,3} & \dots & c_1 & \dots & c_1 \\ b_{1,2} & a_2 & b_{2,3} & \dots & \vdots & \ddots & \vdots \\ b_{1,3} & b_{2,3} & a_3 & \dots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & c_k & \dots & c_k \\ c_1 & \dots & \dots & c_k & d & \dots & d \\ \vdots & \ddots & \ddots & \vdots & \vdots & \ddots & \vdots \\ c_1 & \dots & \dots & c_k & d & \dots & d \end{bmatrix}.$$

In this section we assume that errors occur with the same probability $p_1 = \ldots = p_k = p$ for all marked elements. Thus, the density matrix takes the much simpler form

$$\rho_t = \begin{bmatrix} a & b & b & \dots & c & \dots & c \\ b & a & b & \dots & \vdots & \ddots & \vdots \\ b & b & a & \dots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & c & \dots & c \\ c & \dots & \dots & c & d & \dots & d \\ \vdots & \ddots & \ddots & \vdots & \vdots & \ddots & \vdots \\ c & \dots & \dots & c & d & \dots & d \end{bmatrix}.$$

In the further analysis we use the square of the Frobenius norm of the density matrix:

$$\|\rho\|_F^2 = \sum_{i=1}^{n} \sum_{j=1}^{n} |\rho_{ij}|^2.$$

We will also need the function

$$S(\rho) = k(k-1)b^2 + 2k(n-k)c^2, \tag{3}$$

which gives the sum of squares of all $b$ and $c$ elements of the density matrix.

According to (2), the faulty query transformation $Q'$ decreases the square of the Frobenius norm of the density matrix by

$$\begin{aligned} k(k-1)b^2 + 2k(n-k)c^2 - k(k-1)(b(2p-1)^2)^2 - 2k(n-k)(c(2p-1))^2 = \\ = k(k-1)b^2(1-(2p-1)^4) + 2k(n-k)c^2(1-(2p-1)^2) > \\ > (k(k-1)b^2 + 2k(n-k)c^2)(1-(2p-1)^2) = S(\rho)(4p-4p^2). \end{aligned} \tag{4}$$

Before the first application of the query transformation, the Frobenius norm is 1. Each further application of the query transformation decreases the Frobenius norm. We have proved that the Frobenius norm has a limit of $\frac{1}{\sqrt{k+1}}$ (Frobenius norm of the limiting state $\rho_{lim}$). Thus, total decrease of the Frobenius norm is

$1 - \frac{1}{\sqrt{k+1}}$. Similarly, the square of the Frobenius norm decreases from 1 to $\frac{1}{k+1}$ and has the total decrease of $\frac{k}{k+1}$.

Among first $2m$ applications of the query transformation, there exist two sequential applications which decrease the square of the Frobenius norm by less than $\frac{1}{m}$. Let $\rho_1$ and $\rho_2$ be density matrices before these applications. Let $a_1, b_1, c_1, d_1$ and $a_2, b_2, c_2, d_2$ be $a, b, c, d$ values of $\rho_1$ and $\rho_2$ respectively.

From (4) we have

$$S(\rho_1) < \frac{1}{m(4p - 4p^2)} \quad \text{and} \quad S(\rho_2) < \frac{1}{m(4p - 4p^2)}. \tag{5}$$

In the further proof we use the following straightforward-to-prove lemma:

**Lemma 1** *If $S = k(k-1)b^2 + 2k(n-k)c^2 < R$ and $k \geq 2$ hold then $|c| < \sqrt{\frac{R}{n}}$ and $|b| < \sqrt{R}$ also hold.*

We also use the notation $\delta(a, b) = \{x | a - b < x < a + b\}$.

Lemma 1 and the equation (5) implies

$$c_1 \in \delta\left(0, \sqrt{\frac{R}{n}}\right),$$

$$b_1 \in \delta\left(0, \sqrt{R}\right),$$

$$c_2 \in \delta\left(0, \sqrt{\frac{R}{n}}\right),$$

$$b_2 \in \delta\left(0, \sqrt{R}\right),$$

where $R = \frac{1}{m(4p-4p^2)}$.

The diffusion matrix changes each element $a$ of a vector to $2v - a$, where $v$ is the arithmetic mean of all elements. We will call this the diffusion matrix property.

The arithmetic mean of each of the first $k$ columns of the matrix $\rho_1'$ (after the first application of the query transformation) is

$$v \in \delta\left(\frac{a_1}{n}, \sqrt{R}\frac{k-1}{n} + \sqrt{\frac{R}{n}}\frac{n-k}{n}\right) \subseteq \delta\left(\frac{a_1}{n}, \frac{k}{n}\sqrt{R} + \sqrt{\frac{R}{n}}\right).$$

Because of the diffusion matrix property, the value of the last elements of the first $k$ columns of the matrix $D\rho_1'$ is

$$c_1' = 2v - c_1 \in \delta\left(2\frac{a_1}{n}, \frac{2k}{n}\sqrt{R} + 3\sqrt{\frac{R}{n}}\right).$$

The arithmetic mean of each of the last $n-k$ columns of the matrix $\rho_1'$ is

$$v \in \delta\left(d_1\frac{n-k}{n}, \frac{k}{n}\sqrt{\frac{R}{n}}\right).$$

Hence, the value of the last elements of the last $n-k$ columns of the matrix $D\rho_1'$ is

$$d_1' = 2v - d_1 \in \delta\left(d_1\frac{n-2k}{n}, \frac{2k}{n}\sqrt{\frac{R}{n}}\right).$$

The arithmetic mean of the last row of the matrix $D\rho_1'$ is

$$v \in \delta\left(a_1\frac{2k}{n^2} + d_1\frac{(n-k)(n-2k)}{n^2}, \frac{2k^2}{n^2}\sqrt{R} + \frac{5nk-2k^2}{n^2}\sqrt{\frac{R}{n}}\right).$$

Assuming $n > 2k$ and using the definition of the diffusion matrix, we obtain

$$c_2 = 2v - c_1' \in$$

$$\in \delta\left(-2a_1\frac{n-2k}{n^2} + 2d_1\frac{(n-k)(n-2k)}{n^2}, \frac{4k^2}{n^2}\sqrt{R} + \frac{10nk-4k^2}{n^2}\sqrt{\frac{R}{n}} + \frac{2k}{n}\sqrt{R} + 3\sqrt{\frac{R}{n}}\right) \subseteq$$

$$\subseteq \delta\left(-2a_1\frac{n-2k}{n^2} + 2d_1\frac{(n-k)(n-2k)}{n^2}, \frac{4k}{n}\sqrt{R} + 13\sqrt{\frac{R}{n}}\right) =$$

$$= \delta\left(\frac{2(d_1(n-k)-a_1)(n-2k)}{n^2}, \frac{4k}{n}\sqrt{R} + 13\sqrt{\frac{R}{n}}\right).$$

As $c_2 \in \delta\left(0, \sqrt{\frac{R}{n}}\right)$, $\left|\frac{2(d_1(n-k)-a_1)(n-2k)}{n^2}\right| < \frac{4k}{n}\sqrt{R} + 14\sqrt{\frac{R}{n}}$ holds.

As $ka_1 + d_1(n-k) = 1$, it follows that $d_1(n-k) - a_1 = 1 - (k+1)a_1$. Using the inequality

$$\left|\frac{k}{k+1} - ka_1\right| < \left|1 - (k+1)a_1\right|,$$

we obtain

$$\left|\frac{k}{k+1} - ka_1\right| < \left(\frac{2k}{n} + \frac{7}{\sqrt{n}}\right)\frac{n^2}{n-2k}\sqrt{R}.$$

The left side of this inequality is the absolute value of the difference between the probability of finding any of the marked elements and $\frac{k}{k+1}$.

For an arbitrary $\epsilon$ the inequality

$$\left(\frac{2k}{n} + \frac{7}{\sqrt{n}}\right)\frac{n^2}{n-2k}\sqrt{R} < \epsilon$$

holds if

$$m > \frac{1}{4p(1-p)\epsilon^2}\left(\frac{2k}{n} + \frac{7}{\sqrt{n}}\right)^2\frac{n^4}{(n-2k)^2} = O(n)$$

(substituting $R = \frac{1}{4mp(1-p)}$). □

## References


1. A. Ambainis. Quantum search algorithms. *SIGACT News*, 35(2):22-35, 2004.
2. H. Buhrman, I. Newman, H. Roehrig, R. de Wolf. Robust Polynomials and Quantum Algorithms. *Proceedings of STACS'2005*, Lecture Notes in Computer Science, 3404:593-604, 2005. Also arXiv:quant-ph/0309220
3. L. Grover. A fast quantum mechanical algorithm for database search. *Proceedings of the 28th ACM STOC*, 212-219, Philadelphia, Pennsylvania, 1996. ACM Press.
4. R. Horn, C. Johnson, *Matrix Analysis*, Cambridge University Press, 2006.
5. G. L. Long, Y. S. Li, W. L. Zhang, C. C. Tu. An intrinsic limitation on the size of quantum database. *Physical Review A*, 61:042305, 2000. Also arXiv:quant-ph/9910076.
6. P. Kaye, R. Laflamme, M. Mosca. *An Introduction to Quantum Computing.* Cambridge University Press, 2007.
7. M. Nielsen, I. Chuang. *Quantum Computation and Quantum Information.* Cambridge University Press, 2000.
8. O. Regev, L. Schiff. Impossibility of a Quantum Speed-up with a Faulty Oracle *Proceedings of ICALP'2008*, Lecture Notes in Computer Science, 5125:773-781, 2008.
9. D. Shapira, S. Mozes, O. Biham. The effect of unitary noise on Grover's quantum search algorithm. *Physical Review A*, 67:042301, 2003. Also arXiv:quant-ph/0307142
10. N. Shenvi, K. R. Brown, K. B. Whaley. Effects of Noisy Oracle on Search Algorithm Complexity. *Physical Review A*, 68:052313, 2003. Also quant-ph/0304138.