УДК 512.6, 517.9, 519.6

MODELS OF QUANTUM SEARCH ALGORITHMS. INTRODUCTION FOR IT STUDENTS – PEDAGOGICAL EXAMPLES

Ivancova Olga[1], Ryabov Nikita[2], Korenkov Vladimir[3], Ulyanov Sergey[4]

*[1]Senior researcher;
Dubna State University,
Institute of system analysis and management;
141980, Dubna, Moscow reg., Universitetskaya str., 19;
e-mail: o_ivancova@mail.ru.*

*[2]PhD Student;
Dubna State University;
Institute of the system analysis and management;
141980, Dubna, Moscow reg., Universitetskaya str., 19;
e-mail: ryabov_nv95@mail.ru.*

*[3]Director, Doctor of Technical Science, professor;
Joint institute for nuclear researches,
Laboratory of Information Technologies;
141980, Moscow reg., Dubna, Joliot-Curie, 6;
Dubna State University,
Institute of the system analysis and management;
141980, Dubna, Moscow reg., Universitetskaya str., 19;
e-mail: korenkov@cv.jinr.ru.*

*[4]Doctor of Science in Physics and Mathematics, professor;
Dubna State University,
Institute of system analysis and management;
141980, Dubna, Moscow reg., Universitetskaya str., 19;
e-mail: ulyanovsv@mail.ru.*

This article is one of a series of articles on quantum algorithms. The article discusses quantum oracle models and Grover's computational algorithm for search problems in an unstructured database.

Keywords: quantum computing, quantum algorithms, Grover's quantum search algorithm, unstructured database.

МОДЕЛИ АЛГОРИТМОВ КВАНТОВОГО ПОИСКА. ВВЕДЕНИЕ ДЛЯ ИТ СТУДЕНТОВ – ПЕДАГОГИЧЕСКИЙ ПРИМЕР

**Иванцова Ольга Владимировна[1], Рябов Никита Владимирович[2],
Кореньков Владимир Васильевич[3], Ульянов Сергей Викторович[4]**

*[1]Старший преподаватель;
ГБОУ ВО МО «Университет «Дубна»;
Институт системного анализа и управления;
141980, Московская обл., г. Дубна, ул. Университетская, 19;
e-mail: o_ivancova@mail.ru.*

*[2]Аспирант;
ГБОУ ВО МО «Университет «Дубна»;
Институт системного анализа и управления;
141980, Московская обл., г. Дубна, ул. Университетская, 19;
e-mail: ryabov_nv95@mail.ru.*

[3]*Директор, доктор технических наук, профессор;
Объединенный институт ядерных исследований,
Лаборатория информационных технологий;
141980, Московская обл., г. Дубна, ул. Жолио-Кюри, 6;
ГБОУ ВО МО «Университет «Дубна»,
Институт системного анализа и управления;
141980, Московская обл., г. Дубна, ул. Университетская, 19;
e-mail: korenkov@cv.jinr.ru.*

[4]*Доктор физико-математических наук, профессор;
ГБОУ ВО МО «Университет Дубна»,
Институт системного анализа и управления;
141980, Московская обл., г. Дубна, ул. Университетская, 19;
e-mail: ulyanovsv@mail.ru.*

Эта статья является одной из серии статей о квантовых поисковых алгоритмах. В статье рассматриваются квантовые модели оракула и вычислительный алгоритм Гровера для задач поиска в неструктурированной базе данных.

Ключевые слова: квантовые вычисления, квантовые алгоритмы, алгоритм квантового поиска Гровера, неструктурированная база данных.

Introduction

The difference between classical and quantum algorithms (QA) is following: problem solved by QA is coded in the structure of the quantum operators. Input to QA in this case is always the same. Output of QA says which problem coded. In some sense, you give a function to QA to analyze and QA returns its property as an answer without quantitative computing. QA studies qualitative properties of the functions. The core of any QA is a set of unitary quantum operators or quantum gates. In practical representation, quantum gate is a unitary matrix with particular structure.

The size of this matrix grows exponentially with an increase in the number of inputs, which significantly limits the QA simulation on a classic computer with von Neumann architecture.

The presented article describes a practical approach to modeling one of the most famous QA on classical computers, the Grover algorithm.

Models of quantum oracles and computational algorithm

Grover's search algorithm provides an example of the speed-up that would be offered by quantum computers (if and when they are built) and has the important application in solution of *global optimization* control problems. The problem solved by Grover's algorithm is finding a sought-after («*marked*») element in an unsorted database (DB) of size N . To solve this problem, a classical computer would need $\frac{N}{2}$ database queries on average, and in the worst case it would $N-1$ queries. Using Grover's algorithm, a quantum computer can find the marked state using only $O\left(\sqrt{N}\right)$ quantum data queries. In the case of M «*marked*» elements in an unsorted *DB* of size N speed-up of quantum search process increase as $O\left(\sqrt{\frac{N}{M}}\right)$.

Related works and optimality of quantum searching

Grover discovered a *QA* for identifying a target element in an unstructured *DB* search universe of N items in approximately $\frac{\pi}{4}\sqrt{N}$ queries to a quantum oracle. For classical search using a classical oracle, the search complexity is clearly of order $\frac{N}{2}$ queries since on average half of the items must be searched. It has been proved that this square-root speed-up the best attainable performance gain by any *QA*. It work preceding Grover's, Bennett et al. (1997) had shown that no QA can solve the search problem in fewer than $O\left(\sqrt{N}\right)$ queries. Following Grover's work, Boyer et al. (1998) showed that Grover's algorithm is optimal asymptotically, and that square-root speed-up cannot be improved even if one allows, e.g., a 50% probability of error. Zalka (1999) strengthened these results to show that Grover's algorithm is an optimal algorithm exactly (not only asymptotically). Consider an information-theoretic analysis of Grover's algorithm and the optimality of Grover's algorithm from a point of view for application in design of robust intelligent control.

The Grover's algorithm has optimal order of complexity.

Search problem for an unstructured DB

Consider the problem of searching an unstructured *DB* of $N = 2^n$ records for exactly one record, which has been specifically marked. This can be rephrased in mathematical terms as *an oracle problem* as follows. Label the records of the DB with the integers $0,1,2,...,N-1$, and denote the label of the unknown marked record by x_0. We are given *an oracle*, which computes the *n*-bit binary function $f:\{0,1\}^n \to \{0,1\}$, defined by $f(x) = \begin{cases} 1, & \text{if } x = x_0 \\ 0, & \text{otherwise.} \end{cases}$.

A standard oracle no access to the internal workings of the function *f*. It operates simply as a black-box function, which we can query as many as we like. But with each such a query comes an associated computational cost.

Search problem for an unstructured DB

Find the record labeled as x_0 with the minimum amount of computational work, i.e., with the minimum number of queries of the oracle *f*.

It is known from probability theory, that if *k* records are considered, i.e., if we calculate the oracle *f* for *k* randomly chosen records, then the probability of finding the record labeled as x_0 is $\frac{k}{N}$. Hence, on a classical computer it takes $O(N) = O(2^n)$ queries to find the record labeled x_0. However, as Grover so astutely observed, on a quantum computer the search of an unstructured database can be accomplish in $O(\sqrt{N})$ steps, or more precisely, with the application of $O(\sqrt{N}\lg N)$ sufficiently local unitary transformations. Although this is not exponentially faster, it is a significant speed-up.

Main steps of Grover's search algorithm

We assume without loss of generality that $N = 2^n$, where n is an integer. The algorithm requires of n qubits carrying the computation. When we say it is in a state $|x\rangle$, we mean that its qubits are in states corresponding to the binary representation of the number x.

Example. Consider the following problem:

Input	$x_1 \in \{0,1\}, x_2 \in \{0,1\}, \dots, x_N \in \{0,1\}$ such that exactly one x_i is 1.
Output	The i such that $x_i = 1$.

Classically, one needs $\Omega(N)$ queries to solve this problem and there is no better algorithm than the locations one by one until we find $x_i = 1$. Surprisingly, there is a better algorithm in the quantum case (*Grover*, 1996): There is a QA that solves Problem with $O\left(\sqrt{N}\right)$ queries.

Qualitatively, Grover's original quantum search algorithm (QSA) consists of the following *steps*:

1) Initialize the register to $H|0\rangle$. That is, reset all qubits to 0 and apply the Hadamard transform to each of them;

2) Repeat the following operation (named the Grover iterate G) $T = \frac{\pi}{4}\sqrt{N}$ times:

(2.a) Rotate the marked state $|k\rangle$ by a phase of π radians$\left(I_k^{\pi}\right)$;

(2.b) Apply the Hadamard transform to the register;

(2.c) Rotate the $|0\rangle$ state by a phase of π radians$\left(I_0^{\pi}\right)$;

(2.d) Apply the Hadamard transform again.

Measure the resulting state.

Remark. The original Grover's iterate is $Q = -HI_0^{\pi}HI_k^{\pi}$. It has been generalized to $Q = -UI_s^{\beta}U^{\dagger}I_M^{\gamma}$ where U is an arbitrary unitary operator, s is an arbitrary state, variables β and γ are arbitrary angles, and M includes any number of marked states. We have now observed that any unitary operation Q has a unitary diagonalization. Therefore, it can be represented as $Q = -UI_{\vec{S}}^{\vec{\beta}}U^{\dagger}I_M^{\gamma}$. This is a further generalization of Grover's algorithm, where the state s is replaced by a set of states $\vec{S}$, each of which may have a different rotation angle. Thus, every iterative algorithm is a generalized Grover's algorithm.

According to abovementioned QSA in computation steps of this we must:

1) Apply a unitary transformation U mapping $|0\rangle$ to $\frac{1}{\sqrt{N}}\sum_{i=0}^{N-1}|i\rangle$;

2) Repeat for $\left\lceil \frac{\pi}{4}\sqrt{N} \right\rceil$ times:

✓ Apply the query transformation O which maps $\sum_{i=0}^{N-1} a_i |i\rangle$ to $\sum_{i=0}^{N-1} a_i (-1)^{x_i} |i\rangle$;

✓ Apply the following «diffusion operator D»

$$\begin{cases} D|1\rangle = -\frac{N-2}{N}|1\rangle + \frac{2}{N}|2\rangle + \ldots + \frac{2}{N}|N\rangle \\ D|2\rangle = \frac{2}{N}|1\rangle - \frac{N-2}{N}|2\rangle + \ldots + \frac{2}{N}|N\rangle \\ \qquad\qquad \ldots \\ D|N\rangle = \frac{2}{N}|1\rangle + \frac{2}{N}|2\rangle + \ldots - \frac{N-2}{N}|N\rangle \end{cases};$$

3) Measure the state and output the result of the measurement.

Note that Grover’s QSA is efficient not just in the number of queries but also in the running time. The reason for that is that the diffusion operator D can be implemented in $O(logN)$ time steps. Therefore, the whole algorithm can be implemented in $O\left(\sqrt{N}\, logN\right)$.

Example. Mathematical properties of quantum operations in QSA. Let $\mathcal{H}_2$ be a 2 dimensional Hilbert space with orthonormal basis $\{|0\rangle, |1\rangle\}$, and let the set $\{|0\rangle, |1\rangle, ..., |N-1\rangle\}$ denote the induced orthonormal basis in the Hilbert space $\mathcal{H} = \overset{N-1}{\underset{0}{\otimes}} \mathcal{H}_2$. From the quantum mechanical perspective, the oracle function *f* is given as a black-box transformation U_f, i.e., by

$\mathcal{H} \otimes \mathcal{H}_2 \overset{U_f}{\rightarrow} \mathcal{H} \otimes \mathcal{H}_2$
$\lvert x\rangle \otimes \lvert y\rangle \overset{U_f}{\rightarrow} \lvert x\rangle \otimes \lvert f(x) \oplus y\rangle$,
where «$\oplus$» denotes exclusive *OR* – XOR, i.e., addition modulo 2.

Remark. Instead of U_f, we will use below the computationally equivalent unitary transformation

$$I_{|x_0\rangle}(|x\rangle) = (-1)^{f(x)}|x\rangle = \begin{cases} -|x_0\rangle, & \text{if} \quad x = x_0 \\ \ \ |x\rangle, & \text{otherwise} \end{cases} .$$

That $I_{|x_0\rangle}$ is computationally equivalent to U_f follows from the easily verifiable fact that

$$U_f\left(|x\rangle \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}\right) = \left(I_{|x_0\rangle}(|x\rangle)\right) \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}},$$

and also from the fact that U_f can be constructed from a controlled $I_{|x_0\rangle}$ and two one qubit Hadamard transforms.

We’ll try to understand why this *QSA* work follows the “inverse versus average” method.

To understand the algorithm, plot the amplitudes of $|1\rangle, \ldots, |N\rangle$ at each step. After the first step, the state is $\frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} |i\rangle$ and the amplitudes are $\frac{1}{\sqrt{N}}$. Figure 1.1 (a) shows this result.

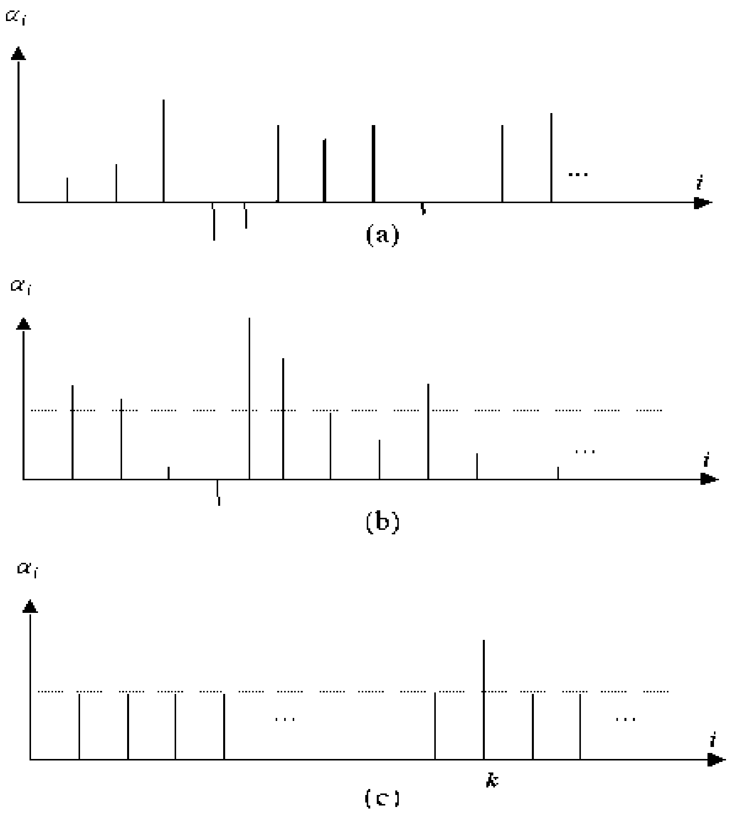

Figure 1.1. Effects of D operation: (a) States before operation; (b) States after operation; (c) result of calculations

After the first query, the amplitude of $\left|i\right\rangle$ with $x_i = 1$ becomes $\left(-\frac{1}{\sqrt{N}}\right)$.

Figure 1.1 (b) shows this result. Then the diffusion operator D is applied. Let $\left|\psi\right\rangle = \sum_{i=0}^{N-1} a_i \left|i\right\rangle$ be the state before the action of D. Then, the state after the action of D is $\left|\psi'\right\rangle = \sum_{i=0}^{N-1} a_i' \left|i\right\rangle$, where $a_i' = -\frac{N-2}{N} a_i + \sum_{i \neq j} \frac{2}{N} a_j$. We can rewrite this as $a_i' = -a_i + \sum_{j=1}^{N} \frac{2}{N} a_j$ and $a_i' + a_i = \sum_{j=1}^{N} \frac{2}{N} a_j$. Let $A = \sum_{j=1}^{N} \frac{1}{N} a_j$ be the average of probability amplitudes a_i. Thus, we have $a_i' + a_i = 2A$ and, if $a_i = A + \Delta$, then $a_i' = A - \Delta$. Therefore, the effect of the «diffusion transform» is that the every amplitude a_i is replaced by its reflection against the average of all a_i.

In particular, after the first query, the amplitude of $\left|i\right\rangle$ with $x_i = 1$ is $\left(-\frac{1}{\sqrt{N}}\right)$ and all the other amplitudes are $\left(\frac{1}{\sqrt{N}}\right)$. The average is $\left(\frac{1}{\sqrt{N}} - \frac{2}{N\sqrt{N}}\right)$ which is almost $\frac{1}{\sqrt{N}}$.

Therefore, after applying D, the amplitude of $\left|i\right\rangle$ with $x_i = 1$ becomes almost $\frac{3}{\sqrt{N}}$ and the amplitudes of all other basis states $\left|j\right\rangle$ slightly less than $\frac{1}{\sqrt{N}}$.

Figure 1.1 (c) shows this result of calculations. The next query makes the amplitude of $|i\rangle$ with $x_i = 1$ approximately $\left(-\frac{3}{\sqrt{N}}\right)$. The average of all amplitudes is slightly less than $\frac{1}{\sqrt{N}}$ and reflecting against it makes the amplitudes of $|i\rangle$ with $x_i = 1$ approximately $\left(\frac{5}{\sqrt{N}}\right)$.

Thus, each step increases the amplitude of $|i\rangle$ with $x_i = 1$ by $O\left(\frac{1}{\sqrt{N}}\right)$ and decreases the other amplitudes. A precise calculation shows that, after $\frac{\pi}{4}\sqrt{N}$ steps, the amplitude of $|i\rangle$ with $x_i = 1$ is $1-o(1)$ Therefore, the measurement gives the correct answer with probability $1-o(1)$.

Boyer et all (1998) have extended Grover's QSA to the case when there can be more than one i with $x_i = 1$. The simplest case if the number of $x_i = 1$ is known. If there are k such values, we can run the same algorithm with $\left\lceil \frac{\pi}{4}\sqrt{\frac{N}{k}} \right\rceil$ iterations instead of $\left\lceil \frac{\pi}{4}\sqrt{N} \right\rceil$. An analysis similar to one above shows that this gives a random i such that $x_i = 1$ with high probability.

A more difficult case is if k is not known in advance. The problem is that, after reaching the maximum, the amplitudes of i with $x_i = 1$ start to decrease. Therefore, if we do too many of iterations, we might not get the right answer. This problem can be handled in two ways. The *first* is running the algorithm above several times with a different number of steps. The *second* is invoking a different algorithm called «*quantum counting*» to estimate the number of $x_i = 1$ and then choose the number of steps for the search algorithm based on that. Either of those approaches gives us solution to:

Problem	$x_1 \in \{0,1\}, x_2 \in \{0,1\}, \ldots, x_N \in \{0,1\}$
Output	i with $x_i = 1$, if there is one, «none» if $x_i = 0$ for all i
Theorem (Boyer, 1998)	There is an algorithm that solves the problem with $O\left(\sqrt{N}\right)$

Many problems can be solved by reductions to both problems mentioned above. For example, consider the satisfiability, which is the canonical *NP*-compete problem. We have a Boolean formula $F(a_1,\ldots,a_n)$ and we have to find whether there exists a satisfying assignment $(a_1 \in \{0,1\}, a_2 \in \{0,1\}, \ldots, a_N \in \{0,1\})$ for which $F(a_1,\ldots,a_n) = 1$. We can reduce the satisfiability to abovementioned *Problem* by setting $N = 2^n$ and defining $(x_1,\ldots,x_N)$ to be $F(a_1,\ldots,a_n)$ for $N = 2^n$ possible assignments $(a_1 \in \{0,1\}, a_2 \in \{0,1\}, \ldots, a_N \in \{0,1\})$. This means that we construct an algorithm that takes $a_1,\ldots,a_n$ and checks if $F(a_1,\ldots,a_n) = 1$. Then, if we replace the black-box in the Grover's QSA by this algorithm, we get an algorithm that find a satisfying assignment in the time $O\left(\sqrt{2^n}\right)$ times time needed to check one assignment. A similar reduction applies to any other problem that can be solved by checking all possibilities in some search space.

Computational steps and physical interpretation of Grover's QSA

Suppose we have an unstructured *DB* with N elements. Without loss of generality, suppose that the elements are numbers from 0 to $N-1$. The elements are not ordered. Classically, we would test each element at a time, until we hit the one searched for. This takes an average of $\frac{N}{2}$ attempts and N in the worst case, therefore the complexity is $O(N)$. As we will see, using quantum mechanics only $O\left(\sqrt{N}\right)$ trials are needed. For simplicity, assume that $N = 2^n$, for some integer n. Grover's *QSA* has *two registers*: n qubits in the first and one qubit in the second.

The *first* step is to create a superposition of all 2^n is to create a superposition of all 2^n computational basis states $\left\{|0\rangle,\ldots,\left|2^n-1\right\rangle\right\}$ of the first register. This is achieved in the following way. Initialize the first register in the state $|00\ldots0\rangle$ and apply the operator $H^{\otimes n}$:

$\lvert\psi\rangle$	$=$	$H^{\otimes n}\lvert 00\ldots0\rangle$
	$=$	$\left(H\lvert 0\rangle\right)^{\otimes n}$
	$=$	$\left(\frac{\lvert 0\rangle+\lvert 1\rangle}{\sqrt{2}}\right)^{\otimes n}$
	$=$	$\frac{1}{\sqrt{N}}\sum_{i=0}^{N-1}\lvert i\rangle$

$|\psi\rangle$ is a superposition of all basis states with equal amplitudes of probability given by $\frac{1}{\sqrt{N}}$.

The *second* register can begin with a state $|1\rangle$ and, after a Hadamard gate applied, it will be in state $|-\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle-|1\rangle\right)$, now define $f:\{0,\ldots,N-1\}\to\{,1\}$ as a function, which recognizes the solution:

$$f(i)=\begin{cases}1, \textit{ if } i \textit{ is the searched element } \left(i_0\right)\\ 0, \textit{ otherwise}\end{cases}.$$

This function is used in the classical algorithm. In the *QA*, let us assume that it is possible to build a linear unitary operator also dependent on f, U_f, such that $U_f\left(|i\rangle|j\rangle\right)=|i\rangle|j\oplus f(i)\rangle$. Operator U_f is called *a quantum oracle* and its physical meaning is described below. In the above equation, $|i\rangle$ stands for a state of the first register, so i is in the set $\left\{0,\ldots,2^n-1\right\}$, $|j\rangle$ is a state of the second register, so j is in $\{0,1\}$, and the sum is modulo 2. It is easy to check that

$U_f\left(\lvert i\rangle\lvert -\rangle\right)$	$=$	$\frac{1}{\sqrt{2}}\left[U_f\left(\lvert i\rangle\lvert 0\rangle\right)-U_f\left(\lvert i\rangle\lvert 1\rangle\right)\right]$
	$=$	$\frac{1}{\sqrt{2}}\left[\lvert i\rangle\lvert f(i)\rangle-\lvert i\rangle\lvert 1\oplus f(i)\rangle\right]$
	$=$	$(-1)^{f(i)}\lvert i\rangle\lvert -\rangle$

In the last equation, we have used the fact that

$$1\oplus f(i)=\begin{cases}0, & \textit{for } i=i_0\\ 1, & \textit{for } i\neq i_0\end{cases}.$$

Now look at what happens when we apply oracle operator U_f to the superposition state coming from the first step, $|\psi\rangle|-\rangle$. The state of the second register does not change. Let us call $|\psi_1\rangle$ the resulting state of the first register:

$|\psi_1\rangle|-\rangle$	$=$	$U_f\,|\psi\rangle|-\rangle$
	$=$	$\frac{1}{\sqrt{N}}\sum_{i=0}^{N-1} U_f\left(|i\rangle|-\rangle\right)$
	$=$	$\frac{1}{\sqrt{N}}\sum_{i=0}^{N-1} (-1)^{f(i)}\,|i\rangle|-\rangle$

$|\psi_1\rangle$ is a superposition of all basis elements, but the probability amplitude of the searching element is negative while all others are positive.

The searched element has been marked with a minus sign. This result is obtained using a feature called *quantum parallelism*. At the quantum level, it is possible «to see» all DB elements simultaneously. The position of the searched element is known: it is the value of i of the term with negative amplitude in last equation. This quantum information is not fully available at the classical level. The classical information of a quantum state is obtained by practical measurements, and, at this point, it does not help if we measure the state of the first register, because it is much more likely that we obtain a non-desired element, instead of the searched one. Before we can perform a measure, the next step should be to increase the amplitude of the searched element while decreasing the amplitude of the others. This is quite general: QA's work by increasing the amplitude of the states, which carry the desired result. After that, a measurement will hit the solution with high probability.

Many QA's can be analyzed in a query (oracle) model where input is given by a block-box (that answers queries) and the complexity of the algorithm is measured by the number of queries to the black-box that it uses.

Example: *Query model.* Most QA's have operated in the so-called black-box setting (or DB– query model). In the black-box model, the input of the function f what we want to compute can only be accessed by means of queries to a black- box. This returns the $i-th$ bit of the input when queried on i. In the query model, the input $x_1,\dots,x_N$ is contained in a black-box and can be accessed by queries to the black-box. In each query, we give i to the black-box and the black-box outputs x_i. The goal is to solve the problem with the minimum number of queries. The classical version of this model is known as *decision trees*. There are two ways to define the query box in the quantum model. The *first* is an extension of the classical query.

Figure 1.2 shows quantum black-box for this case.

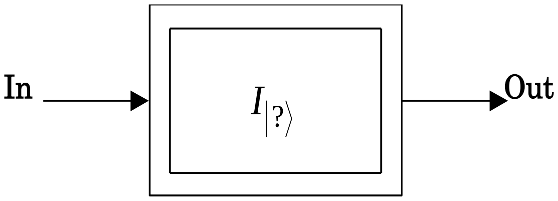

Figure 1.2. A black-box computing device

It has two inputs i, consisting of $\lceil \log N \rceil$ bits and b consisting of 1 bit. If the input to the query box is a basis state $|i\rangle|b\rangle$, the output is $|i\rangle|b \oplus x_i\rangle$. If the input is a superposition $\sum_{i,b} a_{i,b}\,|i\rangle|b\rangle$, the output is $\sum_{i,b} a_{i,b}\,|i\rangle|b \oplus x_i\rangle$. Notice that this definition applies both to the case when the values of x_i are binary and to the case when they are $k-$valued. In the $k-$valued case, we just make b consist of $\lceil \log_2 k \rceil$ bits and take $b \oplus x_i$ to be bit-wise XOR of b and x_i.

The second form of quantum query (which only applies to problem with $\{0,1\}-$valued x_i), the black-box has just one input i. If the input is a state $\sum_i a_i |i\rangle$, the output is $\sum_i a_i (-1)^{x_i} |i\rangle$. While this form is less intuitive, it is very convenient for use in QA's, including Grover's QSA.

For this case, we assume the first form as our main definition but use the second when describing Grover's QSA. This is possible to do because a query of the second type can be simulated by a query of the first type. Conversely, an oracle of the first type can be simulated by a generalization of the sign oracle, which receives $\sum_i a_{i,b} |i\rangle |b\rangle$ as an input and outputs: $\sum_i a_i (-1)^{b \cdot x_i} |i\rangle |b\rangle$. A quantum query model algorithm with T queries is just a sequence of unitary transforms $U_0 \to O \to U_1 \to O \to \ldots U_{T-1} \to O \to U_T$ on some finite-dimensional space $\mathbb{C}^k$, $U_0, U_1, \ldots U_{T-1}, U_T$ can be any unitary transformations that do not depend on the bits $x_1, \ldots, x_N$ inside the black-box. O's are query transformations that consist of applying the black-box to the first $\log N + 1$ bits of the state. That is, we represent basis states of $\mathbb{C}^k$ as $|i, b, z\rangle$. Then, O maps $|i, b, z\rangle$ to $|i, b \oplus x_i, z\rangle$. We use O_x to denote the query transformation corresponding to an input $x = (x_1, \ldots, x_N)$.

The computation starts with the state $|0\rangle$. Then, we apply $U_0, O_x, \ldots, O_x, U_T$ and measure the final state. The result of the computation is the right most bit of the state obtained by the measurement (or several bits if we are considering a problem where the answer has more than 2 values). The QA computes a function $f(x_1, \ldots, x_N)$ if, for every $x = (x_1, \ldots, x_N)$ for which f is defined, the probability that the rightmost bit of $U_T O_x U_{T-1} \ldots O_x U_0 |0\rangle$ equals $f(x_1, \ldots, x_N)$ is at least $1-\varepsilon$ for some fixed $\varepsilon < \frac{1}{2}$. The query complexity of f is the smallest number of queries used by a QA that computes f. We denote it by $Q(f)$.

Let us consider now more in detail the quantum oracle models that in quantum computation are used.

Quantum oracle model

The Grover's QSA solves the unstructured search problem, under the assumption that there exists a computational oracle that can decide whether a candidate solution is the true solution.

Types and relations between oracle models. The following oracles defined in Table 1.1 for a general function $f : \{0,1\}^m \to \{0,1\}^n$.

Table 1.1. Oracles functions

Number	Title of oracle	Type	Definition
1	The phase oracle	P_f	$|x\rangle|b\rangle \to \exp\left\{\frac{2\pi i f(x) \cdot b}{2^n}\right\}|x\rangle|b\rangle$
2	The standard oracle	S_f	$|x\rangle|b\rangle \to |x\rangle|b \oplus f(x)\rangle$
3	The minimal oracle	M_f	$|x\rangle \to |f(x)\rangle$

Here x and b are strings of m and n bits respectively, $|x\rangle$ and $|b\rangle$ the corresponding computational basis states, and $\oplus$ is addition modulo 2^n. The oracles P_f and S_f are equivalent in power: a quantum circuit containing just one copy of the other can construct each of the oracle.

If we take $m=n$ and suppose we know f is a permutation on the set $\{0,1\}^n$ then M_f is a simple invertible quantum map associated to f .

Example: *Each oracle is simulating the other.* One way round turns out to be simple. We can construct S_f from M_f and $\left(M_f\right)^{-1} = M_{f^{-1}}$ as follows: $\boxed{S_f = \left(M_{f^{-1}} \otimes I\right) \circ A \circ \left(M_f \otimes I\right)}$, where «$\circ$» represents the decomposition of operations (or concatenation of networks) and the modulo *N* adder *A* is defined by *A*: $|a\rangle \otimes |b\rangle \to |a\rangle \otimes |a \oplus b\rangle$. Thus, a standard oracle can be simulated given a minimal oracle, using just two invocations, one of M_f and one of $\left(M_f\right)^{-1}$. However, the converse is not true: simulating a minimal oracle M_f requires exponentially many uses of the standard oracle S_f. First, consider the standard oracle $S_{f^{-1}}$ which maps a bits state $|y\rangle|b\rangle$ to $|y\rangle|b \oplus f^{-1}(y)\rangle$, since $S_{f^{-1}} : |y\rangle|0\rangle \to |y\rangle| f^{-1}(y)\rangle$, simulating it allows us to solve the search problem of identifying $| f^{-1}(y)\rangle$ from a DB of *N* elements. It is known that, using Grover's search algorithm, one can simulate $S_{f^{-1}}$ with $O\left(\sqrt{N}\right)$ invocations of S_f.

Example. In the following example we explain one possible way of doing that. Prepare the state $|y\rangle|0\rangle|0\rangle|0\rangle$, where first three registers consist of *n* qubits and the last register is a single qubit. Apply Hadamard transformations on the second register to get $|\Phi_1\rangle = |y\rangle \sum_{x \in Z_n} |x\rangle|0\rangle|0\rangle$. Invoking S_f on the second and third registers new gives $|y\rangle\left[\sum_{x \in Z_N} |x\rangle| f(x)\rangle\right]|0\rangle$. Using CNOT gates, compare the first and third registers and put the result in the fourth, obtaining $\left[|y\rangle \sum_{x \in Z_N x \neq f^{-1}(y)} |x\rangle| f(x)\rangle|0\rangle\right] + \left[|y\rangle| f^{-1}(y)\rangle|y\rangle|1\rangle\right]$.

Now apply $\left(S_f\right)^{-1}$ on the second and third registers, obtaining

$$\left(|y\rangle \sum_{x \in Z_N, x \neq f^{-1}(y)} |x\rangle|0\rangle|0\rangle\right) + \left(|y\rangle| f^{-1}(y)\rangle|0\rangle|1\rangle\right).$$

Taken together, these operations leave the first and third registers unchanged, while their action on the second and fourth defines an oracle for the search problem. Applying Grover's algorithm to this oracle, we obtain the state $|y\rangle| f^{-1}(y)\rangle$ after $O\left(\sqrt{N}\right)$ invocations.

The oracle model

Suppose we are supplied with a model *oracle* – a black-box whose internal workings we discuss later, but which are not important at this stage – with the ability to recognize solutions to the search problem. This recognition is signaled by making use of an *oracle qubit*. More precisely, the oracle is a unitary operator, O, defined by its action on the computational basis: $|x\rangle|q\rangle \xrightarrow{O} |x\rangle|q \oplus f(x)\rangle$, where $|x\rangle$ is the index register, $\oplus$ denotes addition modulo 2, and the oracle qubit $|q\rangle$ is a single qubit which flipped if $f(x)=1$, and is unchanged otherwise. We can check whether x is a solution to our search problem by preparing $|x\rangle|0\rangle$, applying the oracle, and checking to see if the qubit has been flipped to $|1\rangle$. In the QSA it is useful to apply qubit initially in the state $\frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right)$, just as was done in the Deutsch – Jozsa algorithm. If x is not a solution to the search problem, applying the oracle to the state $|x\rangle \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right)$ does not change the state.

On the other hand, if x is a solution to the search problem, then $|0\rangle$ and $|1\rangle$ are interchanged by the action of the oracle, giving a final state

$$\left[-|x\rangle\frac{1}{\sqrt{2}}\big(|0\rangle-|1\rangle\big)\right].$$

The action of the oracle is thus:

$$|x\rangle\underbrace{\left(\frac{1}{\sqrt{2}}\big(|0\rangle-|1\rangle\big)\right)}_{Oracle\ qubit}\xrightarrow{O}(-1)^{f(x)}|x\rangle\underbrace{\left(\frac{1}{\sqrt{2}}\big(|0\rangle-|1\rangle\big)\right)}_{Oracle\ qubit}.$$

Notice that the state of the oracle qubit is not changed. It turns out that this remains $\frac{1}{\sqrt{2}}\big(|0\rangle-|1\rangle\big)$ throughout the QSA, and can therefore be omitted from further discussion of the algorithm, simplifying the description. With this convention, the action of the oracle may be written: $|x\rangle\xrightarrow{O}(-1)^{f(x)}|x\rangle$.

We say that the oracle *marks* the solutions to the search problem, by shifting the phase of the solution. For any N item search problem with M solutions, it turns out that we need only apply the search oracle $O\left(\sqrt{\frac{N}{M}}\right)$ times in order to obtain a solution, on a quantum computer.

It seems as though the oracle already knows the answer to the search problem. Question is what possible use could it be to have a QSA based upon such oracle consultants? The answer is that there is a distinction between knowing the solution to a search problem, and being able to recognize the solution; the crucial point is that it is possible to do the latter without necessarily being able to do the former.

When we say that one item in search space is marked it's means is given a «*black- box*» or «*oracle*» which has the ability to identify a solution to the search problem when it sees a solution. More precisely, we have in our possession *two* registers. The first register stores the index x to an element in the search space, while the second register is a single state z. Supposing s is the marked item then the oracle has the effect: $|x\rangle|z\rangle\to|x\rangle|z\oplus\delta_{sx}\rangle$.

Thus, the oracle «recognized» solutions to the search problem, in the sense that it flips the second register when it finds the solution to the problem in the first register. It's means that the oracle does *not know* the identity of the state it is searching for, but rather can recognize the solution when sees it.

Before describing the steps of the algorithm it's actually very useful to notice two things. First imagine that we prepare the first register in the state $|x\rangle$ and the second register in the superposition $|0\rangle-|1\rangle$. Then the effect of the oracle will be as follows: $|x\rangle\big(|0\rangle-|1\rangle\big)\to(-1)^{\delta_{sx}}|x\rangle\big(|0\rangle-|1\rangle\big)$.

Notice, that the state of the second register is left alone by this operation; henceforth we will ignore the state of the second register, and just write the action of the oracle as $|x\rangle\to(-1)^{\delta_{sx}}|x\rangle$.

In a similar way it's useful for us to be able to perform an operation which leaves the state of our register $|x\rangle$ alone unless it is in the all zero state, in which case a phase shift of (-1) is applied. The computational complexity of the function f is measured by the required number of queries. In this setting we want QA that use significantly fewer queries than the best classical algorithms.

Our purpose is to find the «target» y with the smallest possible number of the oracle evaluations, called the *query complexity*. Remarkable, there is a QSA, which enables this search method to be speed-up substantial, requiring only $O\big(\sqrt{N}\big)$ operations.

Elementary probability theory shows that classically if we examine *k* records then we have probability *k/N* of finding the special one, so we need *O* (*N*) such trials to find it with any constant (independent of *N*) level of probability. Grover's quantum algorithm achieves this result with only $O(\sqrt{N})$ steps (or more precisely $O(\sqrt{N})$ iterations of Grover's operator G but $O(\sqrt{N}\log N)$ steps, the $\log N$ term coming from the implementation of H). It may be shown (Zalka, 1997) that the square root speedup of Grover's algorithm is *optimal* within the context of quantum computation.

In Grover's QSA, the N inputs are mapped onto the states of n qubits. The Grover's QSA is optimal exactly, and not only asymptotically, optimal for query complexity if quantum computation consists only of unitary transformations with fixed structures and the final measurement.

The quantum problem thus becomes one of maximizing the overlap between the state of these n qubits and the target state $|y\rangle$. This is equivalent to maximizing the probability of obtaining the desired state upon measurement. The initial state of these qubits is taken to be an equal superposition of all possible bit stings. The Grover operator, which is used repeatedly in the algorithm, corresponds to a small rotation in the two-dimensional subspace spanned by the initial and target states. Each such rotation requires a single evaluation of $f(x)$. Thus, unlike a classical search, the quantum search monotonically rotates the state towards the target.

Now the circuit for Grover's QSA

We shall work out the details by introducing the circuit for Grover's QSA and analyzing it step by step. Figure 1.3 shows the circuit for Grover's QSA.

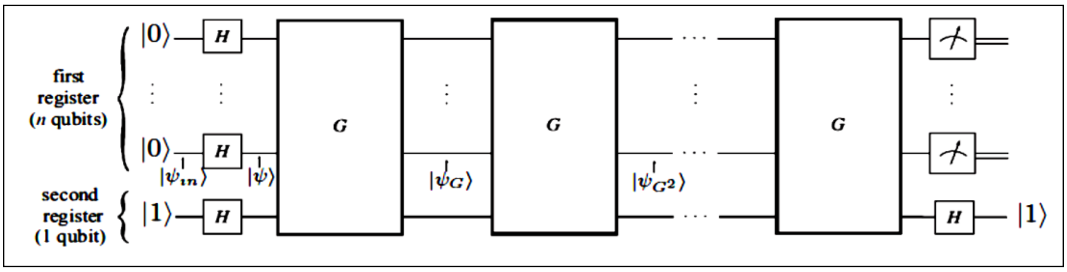

Figure 1.3. Outline of Grover's algorithm

The unitary operator G is applied $O\left(\sqrt{N}\right)$ times. The exact number will be obtained later on. The circuit for one Grover iteration G is given in Fig. 1.4.

The states $|\psi\rangle$ and $|\psi_1\rangle$ are given above. The operator $2|\psi\rangle\langle\psi| - I$ is called *inversion about the mean* for reasons that will be clear below. We will also show how each Grover operator raises the amplitude of the searching element: $|\psi_1\rangle$ can be rewritten as $|\psi_1\rangle = |\psi\rangle - \frac{2}{\sqrt{2^n}}|i_0\rangle$, where $|i_0\rangle$ is the searching element. $|i_0\rangle$ is a state of computational basis. Note that $\langle\psi|\, i_0\rangle = \frac{1}{\sqrt{2^n}}$.

Let us calculate $|\psi_G\rangle$ in Fig. 1.4. Using the abovementioned approach and two last expressions for $|\psi_1\rangle$ and $\langle\psi|\, i_0\rangle$, we obtain

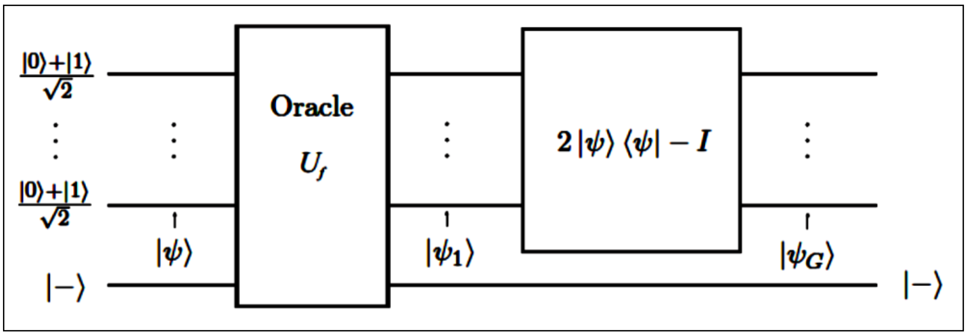
$\frac{|0\rangle+|1\rangle}{\sqrt{2}}$
$\frac{|0\rangle+|1\rangle}{\sqrt{2}}$
$|-\rangle$
$|\psi\rangle$
Oracle
U_f
$|\psi_1\rangle$
$2|\psi\rangle\langle\psi| - I$
$|\psi_G\rangle$
$|-\rangle$

Figure 1.4. One Grover iteration (G)

The states of the first register correspond to the first iteration

$\lvert\psi_G\rangle$	$=$	$(2\lvert\psi\rangle\langle\psi\rvert - I)\lvert\psi_1\rangle$
	$=$	$\frac{2^{n-2}-1}{2^{n-2}}\lvert\psi\rangle + \frac{2}{\sqrt{2^n}}\lvert i_0\rangle$

This is the state of first register after one application of G; the second register is in the state $|-\rangle$. This allows a nice geometrical representation taking $|i_0\rangle$ and $|\psi\rangle$ as base vectors (non-orthogonal basis).

Figure 1.5 shows the vectors $|i_0\rangle$ and $|\psi\rangle$.

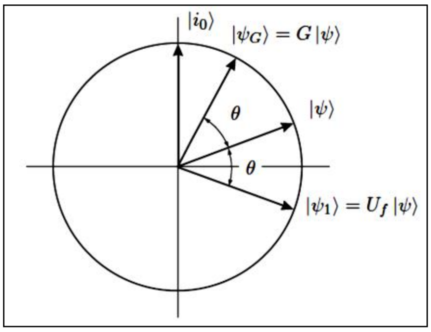
$|i_0\rangle$
$|\psi_G\rangle = G\,|\psi\rangle$
θ
$|\psi\rangle$
θ
$|\psi_1\rangle = U_f\,|\psi\rangle$

Figure 1.5. The state of the first register lives in the real vector space spanned by $|i_0\rangle$ and $|\psi\rangle$

We take these states as a basis to describe what happens in Grover's algorithm. They form an angle smaller than 90^0 as can be seen from the relation $\langle\psi|\,i_0\rangle = \frac{1}{\sqrt{2^n}}$, since $0 < \langle\psi|\,i_0\rangle < 1$. If n is large, then the angle is nearly 90^0. We can think that $|\psi\rangle$ is the initial state of the first register, and the steps of the computation are the applications of the unitary operators U_f and $2|\psi\rangle\langle\psi| - I$. Then $|\psi\rangle$ will rotate in the real plane spanned by $|\psi\rangle$ and $|i_0\rangle$, keeping the unit norm. This means that the tip of $|\psi\rangle$'s vector lies in the unit circle. From the expressions for $|\psi_1\rangle$ and $\langle\psi|\,i_0\rangle$ we see that $|\psi\rangle$ rotates θ degrees clockwise, where $\cos\theta = 1 - \frac{1}{2^{n-1}}$.

Figure 1.5 shows the position of vector $\left|\psi_1\right\rangle$ in the unit circle. From the expressions similar to $\left\langle\psi\right| i_0\rangle$ we see that the angle between $\left|\psi_G\right\rangle$ and $\left|\psi\right\rangle$ is $\cos\theta' = \left\langle\psi\middle|\psi_G\right\rangle = 1 - \dfrac{1}{2^{n-1}}$. So, $\theta' = \theta$ and $\left|\psi_1\right\rangle$ rotates 2θ degrees counterclockwise (in the direction of $\left|i_0\right\rangle$). Figure 1.6 explains also the placement of $\left|\psi_G\right\rangle$.

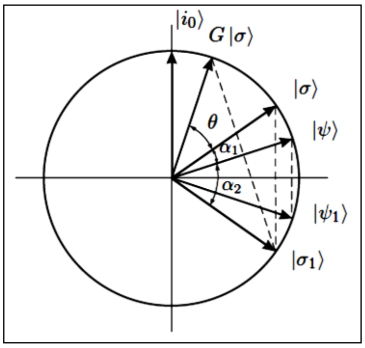

Figure 1.6. A generic vector $\left|i_0\right\rangle$ is reflected around the horizontal axis by the application of U_f, yielding $\left|\sigma_1\right\rangle$

Then, the reflection of $\left|\sigma_1\right\rangle$ about the mean $\left|\psi\right\rangle$ gives $G\left|\sigma\right\rangle$, which is θ degrees closer to $\left|i_0\right\rangle$ (vertical axis). This is remarkable result, since the resulting action of $G = \left(2\left|\psi\right\rangle\left\langle\psi\right| - I\right)U_f$ rotates $\left|\psi\right\rangle$ towards $\left|i_0\right\rangle$ by θ degrees. This means that the amplitudes of $\left|i_0\right\rangle$ in $\left|\psi_G\right\rangle$ increased and the amplitudes of $\left|i\right\rangle$, $i \neq i_0$, decreased with respect to their original values in $\left|\psi\right\rangle$. A measurement, at this point, will return $\left|i_0\right\rangle$ more likely than before. But it is not enough in general, since θ is a small angle if n □ 1 while $\cos\theta = 1 - \dfrac{1}{2^{n-1}}$. That is why we need to apply G repeatedly, ending up θ degrees closer to $\left|i_0\right\rangle$ each time, until the state of the first register be very close to $\left|i_0\right\rangle$, so we can measure.

Computation in Grover's quantum gate and geometrical interpretation of simulation results for N=8

We will describe Grover's QSA for search space of 8 elements for an unknown record with the unknown label $x_0 = 5$. If $N = 8$ then number of input qubit is $n = 3$, $2^3 = 8$. There are 3 qubits in the first register and 1 qubit in the second register. For $N = 8$, the operator G will be applied two times as we will see from estimation $\left\lfloor \dfrac{\pi}{4}\sqrt{N} \right\rfloor$. Figure 1.7 shows the circuit in this case.

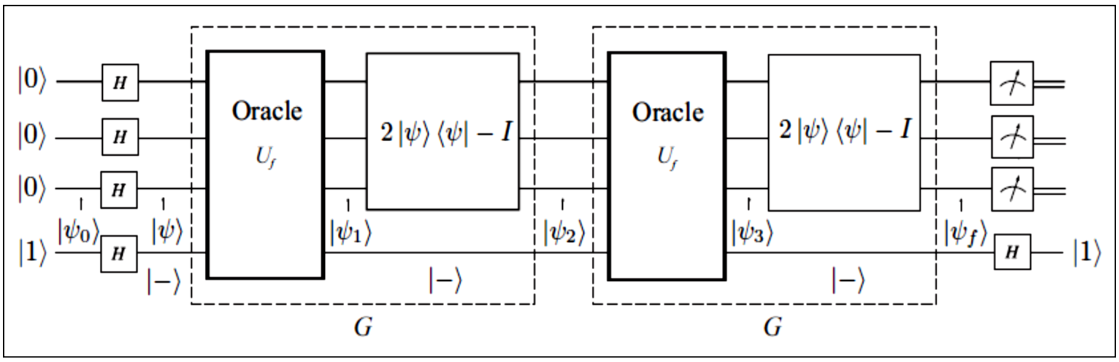

Figure 1.7. Grover's algorithm for N = 8

Classically, an average of more than 4 queries are needed in order to have a probability of success of more than $\frac{1}{2}$.

1. We are given a black-box computing device (see Fig. 1.2) that implements as an oracle the unknown unitary transformation

$$U_f = I_{|x_0\rangle} = I_{|5\rangle} = \left(\begin{array}{cccc|cccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right).$$

We cannot open the black-box in Fig. 1.2 to see what is inside. So we do not know what $I_{|x_0\rangle}$ and x_0 are. The only way that we can glean some information about x_0 is to apply some chosen state $|\psi\rangle$ as input, and then make use of the resulting output. Using of the black-box in Fig. 1.2 as a component device, we construct a computing quantum gate, which implements the unitary operator

$$Q = -HI_{|0\rangle}HI_{|x_0\rangle} = \frac{1}{4}\left(\begin{array}{cccc|cccc} -3 & 1 & 1 & 1 & -1 & 1 & 1 & 1 \\ 1 & -3 & 1 & 1 & -1 & 1 & 1 & 1 \\ 1 & 1 & -3 & 1 & -1 & 1 & 1 & 1 \\ 1 & 1 & 1 & -3 & -1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 3 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -3 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & 1 & -3 & 1 \\ 1 & 1 & 1 & 1 & -1 & 1 & 1 & -3 \end{array}\right)$$

We do not know what unitary transformation Q is implemented by the quantum gate because the black-box is one of its essential components. We can compute the state $|5\rangle$ in standard Grover's QSA as following.

STEP 0: We begin by preparing the known state (superposition)

$$|\psi_0\rangle = H|0\rangle = \frac{1}{\sqrt{8}}(1,1,1,1,1,1,1,1)^{transpose}$$

STEP 1: We proceed to loop $K = round\left(\frac{\pi}{4\sin^{-1}(1/\sqrt{8})}-\frac{1}{2}\right)-\frac{1}{2}) = 2$ times in *STEP 1.*

Iteration 1. On the first iteration, we obtain the unknown state (entanglement state)

$$|\psi_1\rangle = Q|\psi_0\rangle = \frac{1}{4\sqrt{2}}(1,1,1,1,5,1,1,1)^{transpose}$$

Iteration 2: On the second iteration, we obtain the unknown state (interference mode)

$$|\psi_2\rangle = Q|\psi_1\rangle = \frac{1}{4\sqrt{2}}(-1,-1,-1,-1,11,-1,-1,-1)^{transpose}$$

and branch to *STEP 2*.

STEP 2: We measure the unknown state $|\psi_2\rangle$ to obtain either $|5\rangle$ with probability

$$Prob_{success} = \sin^2\left[(2K+1)\beta\right] = \frac{121}{128} = 0.9453$$

or some other state with probability

$$Prob_{failure} = \cos^2\left[(2K+1)\beta\right] = \frac{7}{128} = 0.0547$$

and then exit.

2. Let us describe the quantum computation process state at each step shown in the circuit in Fig. 1.7 as following: $\left(|\psi_0\rangle \to |\psi\rangle \to |\psi_1\rangle \to |\psi_2\rangle \to |\psi_3\rangle \text{ and } |\psi_f\rangle\right)$.

(1) The *initial* state is $|\psi_0\rangle = |000\rangle$;

(2) After Hadamard gates, $|\psi\rangle = H^{\otimes 3}|000\rangle = \left(H|0\rangle\right)^{\otimes 3} = \frac{1}{2\sqrt{2}}\sum_{i=0}^{7}|i\rangle$;

Suppose that we are searching for the element with index 5.

(3) Since $|5\rangle = |101\rangle$,

$$U_f\left(|101\rangle|-\rangle\right) = -|101\rangle|-\rangle,\ for\ i = 5;$$
$$U_f\left(|i\rangle|-\rangle\right) \quad = \quad |101\rangle|-\rangle, if \quad i \neq 5\cdot$$

Define $|u\rangle$ as

$$|u\rangle = \frac{1}{\sqrt{7}}\sum_{i=0,i\neq 5}^{7}|7\rangle = \frac{|000\rangle+|001\rangle+|010\rangle+|011\rangle+|100\rangle+|110\rangle+|111\rangle}{\sqrt{7}}.$$

Then

$$|\psi\rangle = \frac{\sqrt{7}}{2\sqrt{2}}|u\rangle + \frac{1}{2\sqrt{2}}|101\rangle\cdot$$

With this result, we can see the direction of $|\psi\rangle$.

Figure 1.8 shows this direction of $|\psi\rangle$.

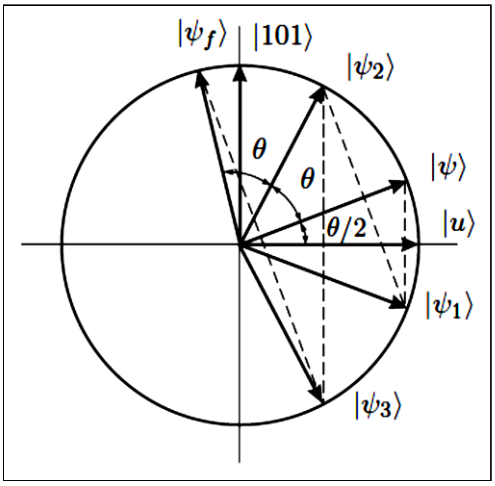

Figure 1.8. Intermediate states in Grover's algorithm for N = 8

The value of θ is

θ	$=$	$2\arccos\left(\frac{\sqrt{7}}{2\sqrt{2}}\right)$
	$=$	$\arccos\left(\frac{3}{4}\right)$
	$\approx$	$41,4^0$

Notice how close is $|\psi_f\rangle$ to $|101\rangle$, indicating a high probability that a measurement will give the searched element. The value of θ is around 41.4°.

(4) The next step is

$$|\psi_1\rangle|-\rangle \;=\; U_f\left(|\psi\rangle|-\rangle\right)= \left(\frac{|000\rangle+|001\rangle+|010\rangle+|011\rangle+|100\rangle-|101\rangle+|110\rangle+|111\rangle}{2\sqrt{2}}\right)|-\rangle.$$

Note that $|101\rangle$ is the only with a minus sign. We can write $|\psi_1\rangle$ as

$$|\psi_1\rangle = |\psi\rangle - \frac{1}{\sqrt{2}}|101\rangle \text{ or } |\psi_1\rangle = \frac{\sqrt{7}}{2\sqrt{2}}|u\rangle - \frac{1}{\sqrt{2}}|101\rangle\cdot$$

The form of last two equations is useful in the next step of calculation since we have to apply $\left(2|\psi\rangle\langle\psi|-I\right)$. The form in last equation is useful to draw the geometrical state $|\psi_1\rangle$.

Figure 1.8 shows the state $|\psi_1\rangle$. $|\psi_1\rangle$ is the reflection of $|\psi\rangle$ with respect to $|u\rangle$.

Next step is the calculation $|\psi_2\rangle = \left(2|\psi\rangle\langle\psi|-I\right)|\psi_1\rangle$. Using the last expressions for $|\psi_1\rangle$, we get $|\psi_2\rangle = \frac{1}{\sqrt{2}}|\psi\rangle + \frac{1}{\sqrt{2}}|101\rangle$ and, using the last expression for $|\psi\rangle$, $|\psi_2\rangle = \frac{\sqrt{7}}{4\sqrt{2}}|u\rangle + \frac{5}{4\sqrt{2}}|101\rangle$.

Let us conform that the angle between $|\psi\rangle$ and $|\psi_2\rangle$ is θ:

$$\cos\theta = \langle\psi|\psi_2\rangle = \frac{1}{2}\langle\psi|\psi\rangle + \frac{1}{\sqrt{2}}\langle\psi|101\rangle = \frac{3}{4},$$

which agrees with the above expression of θ. This completes one application of G.

(5) The second and last application of G is similar. $|\psi_3\rangle$ is given by

$$|\psi_3\rangle = \frac{\sqrt{7}}{2\sqrt{2}}|u\rangle - \frac{5}{4\sqrt{2}}|101\rangle .$$

Using $|\psi\rangle = \frac{\sqrt{7}}{2\sqrt{2}}|u\rangle + \frac{1}{2\sqrt{2}}|101\rangle$, we have $|\psi_3\rangle = \frac{1}{2}|\psi\rangle - \frac{3}{2\sqrt{2}}|101\rangle$.

$|\psi_3\rangle$ is the reflection of $|\psi_2\rangle$ with respect to $|u\rangle$.

(6) The last step is

$$|\psi_f\rangle = \left(2|\psi\rangle\langle\psi| - I\right)|\psi_3\rangle .$$

Using $|\psi\rangle = \frac{\sqrt{7}}{2\sqrt{2}}|u\rangle + \frac{1}{2\sqrt{2}}|101\rangle$ and $|\psi_3\rangle = \frac{1}{2}|\psi\rangle - \frac{3}{2\sqrt{2}}|101\rangle$, we have

$$|\psi_f\rangle = \frac{\sqrt{7}}{8\sqrt{2}}|u\rangle + \frac{11}{8\sqrt{2}}|101\rangle .$$

It is easy to conform that $|\psi_f\rangle$ and $|\psi_2\rangle$ form an angle θ. Note that the amplitude of the state $|101\rangle$ is much bigger than the amplitude of any other state $|i\rangle$ $(i \neq 5)$ in last expression for $|\psi_f\rangle$. This is the way most QA work. They increase the amplitude of the states that carry the desired information. A measurement of the state $|\psi_f\rangle$ in the computational basis will project it into the state $|101\rangle$ in the computational basis with probability $p = \left|\frac{11}{8\sqrt{2}}\right|^2 \approx 0.9453$. The chance of getting the result $|101\rangle$, which reads as number 5, is around $94{,}5\%$.

Generalization of computational process in QSA

The easiest way to calculate the output of Grover's QSA is to consider only the action of G instead of breaking the calculation into action of the oracle U_f and the inversion about the mean. To this end, we choose $|i_0\rangle$ and $|u\rangle$ as the basis for the subspace where $|\psi\rangle$ rotates after successive applications of G. $|i_0\rangle$ is the searched state and $|u\rangle$ is defined from the above expression in general form as

$$|u\rangle = \frac{1}{\sqrt{N-1}} \sum_{i=0, i \neq i_0}^{N-1} |i\rangle = \sqrt{\frac{N}{N-1}}|\,\psi\rangle - \frac{1}{\sqrt{N-1}}|\,i_0\rangle .$$

From the first expression above we easily see that $\langle i_0|u\rangle = 0$, i.e., $|i_0\rangle$ and $|u\rangle$ are orthogonal. From the second equation we have $|\,\psi\rangle = \sqrt{1-\frac{1}{N}}|\,u\rangle + \frac{1}{\sqrt{N}}|\,i_0\rangle$. The state of the quantum computing at each step is $G^k|\psi\rangle = \cos\left(\frac{2k+1}{2}\theta\right)|\,u\rangle + \sin\left(\frac{2k+1}{2}\theta\right)|\,i_0\rangle$, where we have dropped the state of the second register it is $|-\rangle$ all the time. Figure 1.9 shows effect of G on $|\psi\rangle$.

The above last equation is obtained after analyzing the components of $G^k\left|\psi\right\rangle$. The value of θ is obtained substituting k for 0 in last expression and comparing it above with two last equations, $\theta = 2\arccos\sqrt{1-\frac{1}{N}}$ The equation for $G^k\left|\psi\right\rangle$ expresses the fact (we proved above), that each application of G rotates the state of the first register by θ degrees towards $\left|i_0\right\rangle$. Figure 1.9 shows successive applications of G.

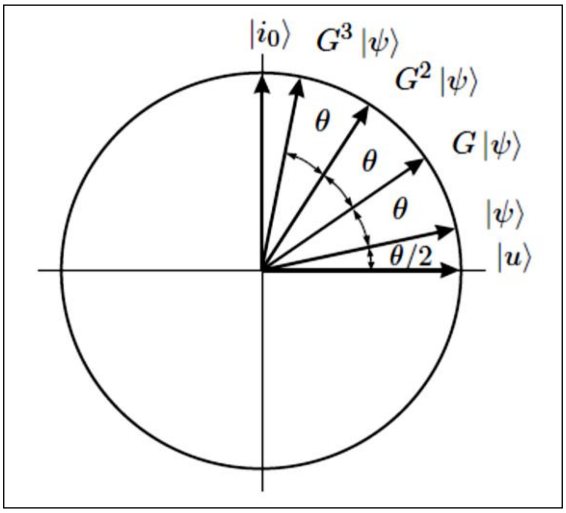

Figure 1.9. Effect of G on $\left|\psi\right\rangle$

The number of times k_0 that G must be applied obeys the equation $k_0\,\theta + \frac{\theta}{2} = \frac{\pi}{2}$. Since k_0 must be integer, we write $k_0 = round\left(\frac{\pi-\theta}{2\theta}\right)$, where θ is define from above equation $\theta = 2\arccos\sqrt{1-\frac{1}{N}}$. If N □ 1, by Tailor expanding this last equation, we get $\theta \approx \frac{2}{\sqrt{N}}$ and from the expression for $k_0 = round\left(\frac{\pi-\theta}{2\theta}\right)$, and we have $k_0 = round\left(\frac{\pi}{4}\sqrt{N}\right)$. After applying k_0 times the operator G, the probability p of finding the desired element (after a measurements) is $p = \sin^2\left(\frac{2k_0+1}{2}\theta\right)$.

Probability of successful result of quantum search

Figure 1.10 shows the evolution value of probability p of finding the desired element (after measurements) for n form 2 to 30.

Recall that, so for $n = 30$ the search space has around one billion elements. For $n = 2$ the probability of getting, the result is exactly 1. The reason for this case is that the equation for θ is $\theta = 2\arccos\sqrt{1-\frac{1}{N}}$ and yields $\theta = \frac{\pi}{3}$. And $\left|\psi\right\rangle$ makes an angle $\theta = \frac{\pi}{6}$ with $\left|u\right\rangle$.

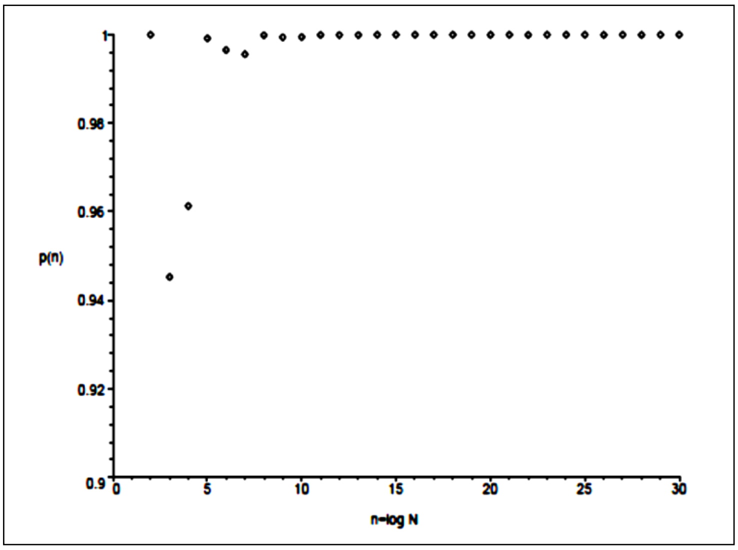

Figure 1.10. Probability of succeeding as a function of n

Applying G one times rotates $|\psi\rangle$ to $|i_0\rangle$ exactly. For $n=2$, from $p=\sin^2\left(\frac{2k_0+1}{2}\theta\right)$ yields $p\approx 0,9453$ which is the result that described above.

Simulation of Grover's Quantum Search Algorithm – Gate-based Approach

Background. We receive X_k by making a query with the index k. We call such a query a *classical query*. On quantum computers, the index associated with a query is expressed with qubits, and thus a query, in this case called a *quantum query*, will be a superposition of classical queries over all indices k ranging from 1 to N; accordingly, the answer to the quantum query will be the corresponding superposition of all X_k, where k ranges from 1 to N. With the ability to make quantum queries, the quantum search may be stated as follows (we assume for simplicity that each piece of input data is either 0 or 1, but this is not essential to the quantum search). Any classical algorithm for finding an item in a randomly ordered phone book (whether deterministic or probabilistic) requires $N/2$ steps on the average, because the only way to perform the search is to analyze each item one by one until the searched-for item is found. Recently, Grover invented a quantum algorithm that runs like $O(\sqrt{N})$.

Theorem (Quantum Search) Given N input data $X_1,\ldots, X_N \in \{0,1\}$, there exists a quantum algorithm that finds an index i with $X_i = 1$ with high probability by making approximately N data accesses (i.e., quantum queries).

To estimate the total number of steps required to solve a problem, it is necessary to count the number of steps taken to process the input data obtained via queries, as well as the number of accesses to the input data. However, we will focus only on the number of accesses to the input data, since it is a dominant factor in the search problem and the other problems dealt with in this article.

Let us review it briefly. In a phone book with $N = 2^n$ entries, each item can be represented by a binary label of length n or, equivalently, by a pure state of n spin ½ particles. The algorithm is based on constructing a coherent superposition of all these states, and applying repeatedly certain unitary transformations to it. Assume, for concreteness, that the item we are looking for is represented by the state $|\downarrow\downarrow\ldots\downarrow\rangle$, i.e. by n spin-down particles.

The algorithm works via the repeated action of the unitary steps below, starting from an initial state which we take to be the full coherent superposition of all states in the system, namely

$$\psi_0 = \frac{1}{\sqrt{N}}\begin{pmatrix}1\\1\\\vdots\\1\end{pmatrix}.$$

Of course, one could start equally well with some other initial state. The two unitary steps to be repeated are the following: Invert the phase of the looked-for state trough the unitary transformation

$$U_1 = \begin{pmatrix}-1 & 0 & \dots & 0\\ 0 & 1 & 0 & \vdots\\ \vdots & 0 & \ddots & 0\\ 0 & 0 & \dots & 1\end{pmatrix}.$$

Invert, with respect to the average, the phase of the looked-for state trough the unitary diffusion matrix

$$(U_2)_{ij} = \frac{2}{N} - \delta_{ij}.$$

These two steps are equivalent to the action of the following single unitary transformation:

$$U = U_2 U_1 = \frac{2}{N}\begin{pmatrix}-1+\frac{2}{N} & 1 & \dots & 1\\ -1 & 1-\frac{2}{N} & 1 & \vdots\\ \vdots & 1 & \ddots & 1\\ -1 & 1 & \dots & 1-\frac{2}{N}\end{pmatrix}.$$

When the unitary transformation *U* has been applied *m* times to the initial state ψ_0, the new quantum state will be

$$\psi_m = U^m\,\psi_0 = \begin{pmatrix}A_m\\\vdots\\B_m\end{pmatrix}.$$

The action of *U* on the initial state ψ_0 yields only two distinct amplitudes A_m and B_m, whereby it is possible to recast the recursion relation in just two dimensions. The restriction of *U* to this two-dimensional subspace will be denoted by *S*. Explicitly, the amplitudes A_m and B_m are given by the recursion formula

$$\begin{pmatrix}A_{m+1}\\B_{m+1}\end{pmatrix} = \begin{pmatrix}1-\frac{2}{N} & 2-\frac{2}{N}\\ \frac{-2}{N} & 1-\frac{2}{N}\end{pmatrix}\begin{pmatrix}A_m\\B_m\end{pmatrix} = S\begin{pmatrix}A_m\\B_m\end{pmatrix} = S^{m+1}\begin{pmatrix}\frac{1}{\sqrt{N}}\\ \frac{1}{\sqrt{N}}\end{pmatrix}.$$

The two-dimensional matrix *S* has eigenvalues $e^{\pm i\varphi}$, with $\cos\phi = 1-\frac{1}{N}$, whereby

$$A_m = \frac{1}{\sqrt{N}}(\cos m\phi + \sqrt{N-1}\sin m\phi),$$

$$B_m = \frac{1}{\sqrt{N}}(\cos m\phi - \frac{1}{\sqrt{N-1}}\sin m\phi).$$

The probability of finding the state we are looking for if we measure ψ_m is thus

$$P(m) = |A_m|^2 = \frac{1}{N}(\cos m\phi + \sqrt{N-1}\sin m\phi)^2.$$

With the change of variables $\phi = 2\theta$, $P(m)$ can be written as $P(m) = \sin^2(\theta(2m+1))$.

Clearly, $P(m)$ is periodic, with maxima at $\theta(2m+1) = n\pi$, where n is integer.

The first maximum for large N is approximately at $m_{\max} \cong \frac{\pi\sqrt{N}}{4}$ and $P_{\max} = P(m_{\max}) \cong 1$. The number of steps required to find the state with almost certainty scales like $\sqrt{N}$, as shown is upper.

The Grover search algorithm has four stages: initialization, oracle, amplification, and measurement, as shown in Fig. 1.11a.

On Fig. 1.11 the initialization stage creates an equal superposition of all possible input states, so the amplitude $\alpha_x = 1$ for all basis states |x⟩. The oracle stage marks the desired state, so the amplitude α_m of the marked state $|m\rangle$ becomes negative while the amplitudes α_b of the unmarked states $|b\rangle$, $b \neq m$ remains unchanged. The amplification stage performs a reflection about the mean vector $\sum_{x=0}^{N-1}|x\rangle$ which has amplitude $A = \frac{1}{N}\sum_{x=0}^{N-1}\alpha_x = \frac{1}{N}\left(-\alpha_m + (N-1)\alpha_0\right)$, to amplify the marked state. An appropriate number of repetitions of the oracle and amplification stages will maximize the amplitude of the correct answer. All qubit states are normalized by the factor $\frac{1}{\sqrt{N}}$. The algorithm can also be generalized to mark and amplify the amplitude of *t* desired states. On Fig. 1.11b general circuit diagram for a Grover search algorithm using a Boolean oracle, depicted using standard quantum circuit diagram notation. The last qubit q_a is the ancilla qubit. On Fig. 1.11c example of single-solution Boolean oracle marking the |011⟩ state. On Fig. 1.11d general circuit diagram for a Grover search algorithm using a phase oracle. On Fig. 1.11e example of two-solution phase oracle marking the |011⟩ and |101⟩ states.

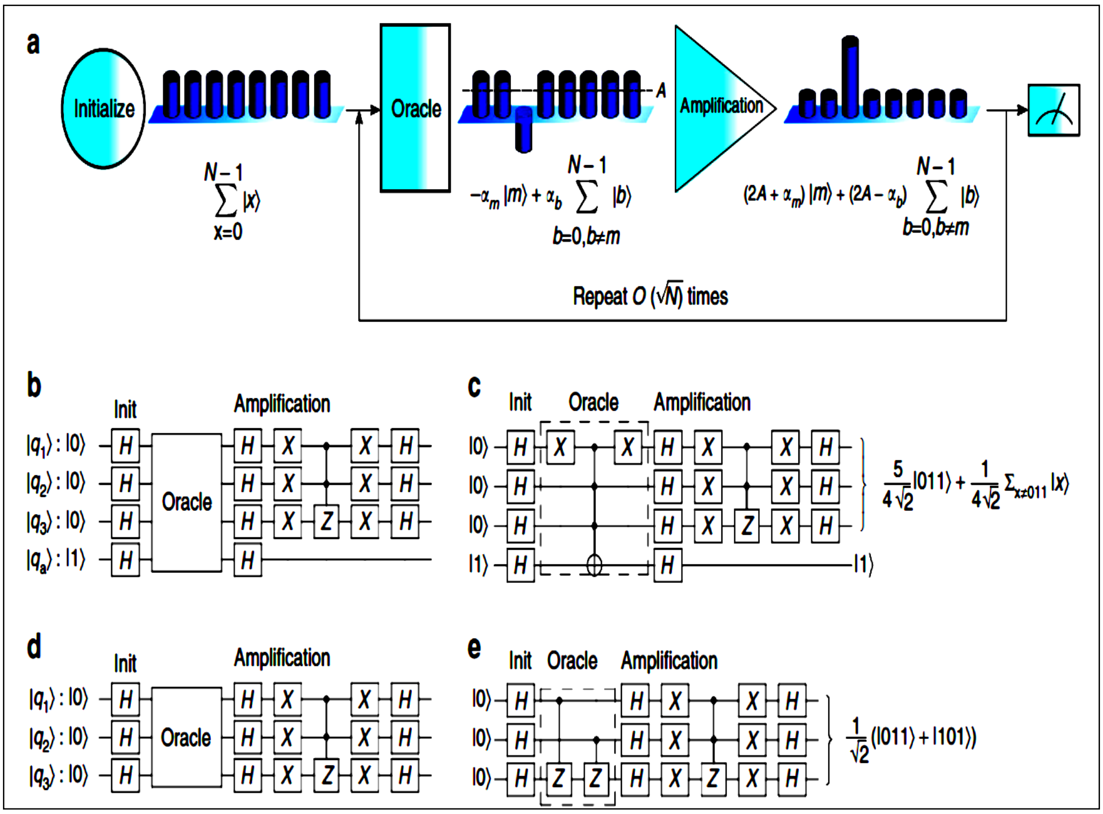

Fig. 1.11. The Grover search algorithm. ***a*** *Evolution of relative amplitudes for each state during a Grover search algorithm*

The initialization stage creates an equal superposition of all states. The oracle stage marks the solution(s) by flipping the sign of that state's amplitude. The amplification stage performs a reflection about the mean, thus increasing the amplitude of the marked state. Finally, the algorithm output is measured. For a search database of size *N*, the single-shot probability of measuring the correct answer is maximized to near-unity by repeating the oracle and amplification stages $O\left(\sqrt{N}\right)$ times. By comparison, a classical search algorithm will get the correct answer after an average of *N*/2 queries of the oracle.

For large databases, this quadratic speedup represents a significant advantage for quantum computers. All searches are performed with a single iteration. For a single-solution algorithm (*t* = 1), the algorithmic probability of measuring the correct state after one iteration is

$$t\square\left(\left[\frac{N-2t}{N}+\frac{2\left(N-t\right)}{N}\right]\frac{1}{\sqrt{N}}\right)^2=\left(\frac{5}{4\sqrt{2}}\right)=78.125\%,\ n=3, N=2^n=8,$$

compared to $\frac{t}{N}+\frac{N-t}{N}\square\frac{t}{N-1}=\frac{1}{8}+\frac{7}{8}\square\frac{1}{7}=25\%$ for the optimal classical search strategy, which consists of a single query followed by a random guess in the event the query failed. In the two-solution case (*t* = 2), where two states are marked as correct answers during the oracle stage and both states' amplitudes are amplified in the algorithm's amplification stage, the probability of measuring one of the two correct answers is 100% for the quantum case, as compared to $\frac{13}{28}\approx 46.4\%$ for the classical case. The algorithm is performed with both a phase oracle, which has been previously demonstrated on other experimental systems, and a Boolean oracle, which requires more resources but is directly comparable to a classical search. All quantum solutions are shown to outperform their classical counterparts.

The Grover search algorithm is implemented using circuits that are equivalent to those shown in Fig. 1.11b, d, but with the initialization and amplification stages optimized to minimize gate times. The circuits shown are for use with Boolean oracles; in the phase oracle case, the ancilla qubit q_a is simply omitted. To preserve the modularity of the algorithm, the initialization stage and amplification stage were each optimized without regard to the contents of the oracle, so each possible oracle can simply be inserted into the algorithm without making any changes to the other stages.

Oracles for the Grover search algorithm were constructed using a combination of reversible and classical logic synthesis techniques. For Boolean oracles, reversible logic synthesis was employed to find a set of *X*, $C^N(NOT)$ gates that marked the desired state(s) for each oracle. For phase oracles, EXOR polynomial synthesis was used to find a set of *Z*, $C^N(Z)$ gates that marked the desired state(s) for each oracle. For example, for Boolean oracles, the selection was limited to the classically available *X* (or *NOT*) and $C^N(NOT)$ gates, and a reversible circuit was constructed such that the output bit (corresponding to the ancilla qubit in the quantum oracle) would be flipped if and only if a marked state was used as the input to the circuit.

While there are many possible circuit constructions for each oracle, the oracle chosen for implementation was one that first minimized the number of two-qubit interactions required, and then minimized the number of single-qubit interactions needed. Other quantum algorithms may be implemented on this system in a similar fashion.

First, decompose the algorithm’s subroutines into high-level circuits. Second, optimize those circuits to minimize the number of two-qubit interactions required. Third, decompose the high-level circuits into physical-level *R* and *XX* gates. Finally, perform further optimizations to first minimize the number of two qubit *XX* gates required, and then to minimize the total rotation time (the sum of all rotation angles θ) across all *R* gates. However, since the optimization of quantum circuits is QMA-Hard, we anticipate that future improvements in algorithm design, circuit synthesis, and circuit optimization techniques may result in more efficient circuit implementations, facilitating increased experimental performance.

Conclusion

The article describes quantum oracle models and a computational algorithm. Is being discussed optimality of quantum search. The search problem in an unstructured database is considered and described basic computational steps, physical interpretation of the Grover algorithm and the probability of a successful quantum search result.

The mathematical model, the features of the derivation of the Grover quantum search algorithm and classical efficient modeling using this algorithm will be discussed in future articles.

References

1. Grover L.K. A fast quantum mechanical algorithm for database search // Proceedings, 28th Annual ACM Symposium on the Theory of Computing, 1996.
2. Shor P. Algorithms for Quantum Computation: Discrete Logarithms and Factoring // Foundations of Computer Science, 1994 Proceedings., 35th Annual Symposium on — IEEE, 1994. — P. 124-134.
3. 3. Schmitt I. Quantum query processing: unifying database querying and information retrieval. — Otto-von-Guericke-Universitat Magdeburg, 2006.
4. Masahito Hayashi, Satoshi Ishizaka, Akinori Kawachi, Gen Kimura, Tomohiro Ogawa. Introduction to Quantum Information Science. — Berlin: Springer-Verlag Berlin Heidelberg, 2015.
5. Coles P.J., et al. Quantum Algorithm Implementations for Beginners // arXiv:1804.03719v1 [cs.ET] 10 Apr 2018.
6. Childs A.M. Lecture Notes on Quantum Algorithms // University of Maryland. — 30 May 2017.
7. Botsinis P. et al. Quantum Search Algorithms for Wireless Communications // IEEE COMMUNICATIONS SURVEYS & TUTORIALS. — 2019. — Vol. 21. — No. 2. — Pp. 1209-1242.
8. Jairo Ernesto Castillo, Yesenia Sierra, Nelson L. CubillosClassical simulation of Grovers quantum algorithm. // Revista Brasileira de Ensino de Física. — 2020. — Vol. 42.

9. Mutibara A.B., Refianti R. Simulation of Grover algorithm Quantum search in a Classical Computer, //International Journal of computer Sconce and Information security. — 2010. — Vol. 8. — No 9.

10. Zhuang Jiaya et al. Analysis and Simulation of Grover algorithm.// International Journal of Machine Learning and Computer. 2014. — Vol. 4. — No 1.

11. Ulyanov S.V., Litvintseva L.V., Ulyanov S.S. Quantum information and quantum computational intelligence: Design & classical simulation of quantum algorithm gates. — Universita degli Studi di Milano: Polo Didattico e di Ricerca di Crema Publ. — 2005. — Vol. 80.

12. Lavor C. Grover's Algorithm: Quantum Database Search // arXiv:quant-ph / 0301079.

13. Figgatt C. et al. Complete 3-Qubit Grover search on programmable quantum computer // NATURE COMMUNICATIONS | DOI: 10.1038/s41467-017-01904-7.