SCIENTIFIC PAPERS, UNIVERSITY OF LATVIA, 2010. Vol. 756
COMPUTER SCIENCE AND INFORMATION TECHNOLOGIES 221–225 P.

# A Note on the Optimality of the Grover's Algorithm

**Nikolajs Nahimovs, Alexander Rivosh**[⋆]
Faculty of Computing, University of Latvia
*nikolajs.nahimovs@lais.lv, aleksandrs.rivoss@lais.lv*

**Abstract.** The Grover's algorithm is a quantum search algorithm solving the unstructured search problem in about $\frac{\pi}{4}\sqrt{N}$ queries. It is known to be optimal - no quantum algorithm can solve the problem in less than the number of steps proportional to $\sqrt{N}$ [3]. Moreover, for any number of queries up to about $\frac{\pi}{4}\sqrt{N}$, the Grover's algorithm ensures the maximal possible probability of finding the desired element [2].
However, it is still possible to reduce the average number of steps required to find the desired element by ending the computation earlier and repeating the algorithm if necessary. This fact was mentioned by Christof Zalka as a short remark on analysis of the Grover's algorithm [2]. Our article gives a detailed description of this simple fact.

## 1 Unstructured Search

Suppose we have a function

$$f : \{0,1\}^n \rightarrow \{0,1\}$$

given by a black box. The unstructured search problem is to find a string $x \in \{0,1\}^n$ such that $f(x) = 1$, or to conclude that no such $x$ exists if $f$ is identical to 0.

It is easy to see that a deterministic algorithm would need to make $N = 2^n$ queries to the blackbox in the worst case (to distinguish the case where $f$ is identical to 0 from any of the cases where there is a single $x$ for which $f(x) = 1$). It can be shown [4] that probabilistically we also need $\Omega(N)$ queries to solve the problem. In contrast, a quantum computer can solve the problem using $O(\sqrt{N})$ queries.

## 2 The Grover's Quantum Search Algorithm

The Grover's algorithm is a quantum search algorithm that solves the unstructured search problem in about $\frac{\pi}{4}\sqrt{N}$ queries. We will give a brief description of the algorithm. For a detailed description, see [1] or [4].

[⋆] Supported by Unversity of Latvia grant ZB01-100 and ESF.

### The Grover's algorithm

1. Let $X$ be an $n$-qubit quantum register with initial state $|0^n\rangle$. Perform $H^{\otimes n}$ on $X$.
2. For $k$ times ($k$ will be specified later), apply to the register $X$ the transformation $G = DQ$, where $D$ is a rotation about average [1] and $Q$ is a query transformation $Q\,|x\rangle = (-1)^{f(x)}\,|x\rangle$.
3. Measure $X$ and output the result.

For the purposes of analysis define two sets of strings:

$$A = \{x \in \{0,1\}^n : f(x) = 1\}$$

$$B = \{x \in \{0,1\}^n : f(x) = 0\}\,.$$

We will think of the set $A$ as the set of "good" strings; the goal of the algorithm is to find one of these strings. The set $B$ contains all the "bad" strings that do not satisfy the search criterion.

Let $a = |A|$ and $b = |B|$. Define states

$$|A\rangle = \frac{1}{\sqrt{a}} \sum_{x\in A} |x\rangle \quad \text{and} \quad |B\rangle = \frac{1}{\sqrt{b}} \sum_{x\in B} |x\rangle$$

which are both unit vectors and are orthogonal to one another.

The initial state of the register $X$ is

$$|X\rangle = H^{\otimes n}\,|0^n\rangle = \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle = \sqrt{\frac{a}{N}}\,|A\rangle + \sqrt{\frac{b}{N}}\,|B\rangle\,.$$

Calculations show that the transformation $G$ changes states $|A\rangle$ and $|B\rangle$ as follows:

$$G\,|A\rangle = \left(1 - \frac{2a}{N}\right)|A\rangle - \frac{2\sqrt{ab}}{N}\,|B\rangle$$

$$G\,|B\rangle = \frac{2\sqrt{ab}}{N}\,|A\rangle - \left(1 - \frac{2b}{N}\right)|B\rangle\,.$$

As $\sqrt{\frac{a}{N}}^2 + \sqrt{\frac{b}{N}}^2 = 1$ there exists an angle $\theta$ that satisfies

$$\sin\theta = \sqrt{\frac{a}{N}} \quad \text{and} \quad \cos\theta = \sqrt{\frac{b}{N}}\,.$$

Using this notation, the initial register $X$ state can be written as

$$|X\rangle = \sin\theta\;|A\rangle + \cos\theta\;|B\rangle$$

and the transformation $G$ as

$$G\,|A\rangle = \cos 2\theta\ |A\rangle - \sin 2\theta\ |B\rangle$$

$$G\,|B\rangle = \sin 2\theta\ |A\rangle + \cos 2\theta\ |B\rangle$$

which is simply a rotation by angle $2\theta$ in the space spanned by $|A\rangle$ and $|B\rangle$. T his implies that after $k$ iterations of $G$, the state of $X$ will be

$$\sin((2k+1)\theta)\ |A\rangle + \cos((2k+1)\theta)\ |B\rangle$$

The goal of the algorithm is to measure some element $x \in A$, so we would like the state of $X$ to be as close to $|A\rangle$ as possible. If we want

$$\sin((2k+1)\theta) \approx 1$$

then

$$(2k+1)\theta \approx \frac{\pi}{2}$$

will suffice, so we should choose

$$k \approx \frac{\pi}{4\theta} - \frac{1}{2}\,.$$

Suppose $a = 1$. Then

$$\theta = \sin^{-1}\sqrt{\frac{1}{N}} \approx \frac{1}{\sqrt{N}}$$

so

$$k = \lfloor \frac{\pi}{4}\sqrt{N} \rfloor$$

is a reasonable choice for the algorithm.

In the general case the situation is more challenging. However, it can be shown that $O(\sqrt{\frac{N}{a}})$ queries are still enough to find an $x \in A$

[4].

## 3 Ending the Computation Earlier

Suppose we have an algorithm which gives a correct answer with some probability $p$. To obtain the correct answer, we need to repeat it $\frac{1}{p}$ times on the average. If the algorithm's running time is $k$, repeating it gives the average running time of $\frac{k}{p}$.

In the previous section we have shown that after $k$ steps, the state of the Grover's algorithm is

$$\sin((2k+1)\theta)\ |A\rangle + \cos((2k+1)\theta)\ |B\rangle\,.$$

The amplitude of the correct answer grows proportionally to $\sin(2k\theta) \approx \sin(\frac{2k}{\sqrt{N}})$, therefore, the probability to obtain the correct answer grows proportionally to $\sin^2(\frac{2k}{\sqrt{N}})$. To get rid of $N$, let us scale $k$ from $[0, \frac{\pi}{4}\sqrt{N}]$ to $[0, 1]$. That is, let the running time of the original algorithm be 1 and let $k$ represent the fraction of steps completed by the algorithm. The probability to obtain the correct answer becomes $p = \sin^2(\frac{\pi k}{2})$.

Now, if we stop the computation at the moment $k$, the average running time of the algorithm will be

$$\frac{k}{p} = \frac{k}{\sin^2\left(\frac{\pi k}{2}\right)}.$$

If $k \in [0, 0.5)$ then

$$\sin^2\left(\frac{\pi k}{2}\right) < k$$

and

$$\frac{k}{p} = \frac{k}{\sin^2\left(\frac{\pi k}{2}\right)} > 1.$$

Therefore, the average running time is greater than in the original algorithm. If $k = 0.5$, then

$$\sin^2\left(\frac{\pi k}{2}\right) = 0.5$$

and

$$\frac{k}{p} = \frac{k}{\sin^2\left(\frac{\pi k}{2}\right)} = 1.$$

The average running time is the same as in the original algorithm. If $k \in (0.5, 1]$, then

$$\sin^2\left(\frac{\pi k}{2}\right) > k$$

and

$$\frac{k}{p} = \frac{k}{\sin^2\left(\frac{\pi k}{2}\right)} < 1.$$

Therefore, the average running time is less than in the original algorithm.

The optimal moment to end the computation is the minimum of the $\frac{k}{p}$ function.

$$\left(\frac{k}{p}\right)' = \left(\frac{k}{\sin^2\left(\frac{\pi k}{2}\right)}\right)' = \frac{\sin^2(\frac{\pi k}{2}) - k \cdot 2 \cdot \sin(\frac{\pi k}{2}) \cdot \cos(\frac{\pi k}{2}) \cdot \frac{\pi}{2}}{\sin^4(\frac{\pi k}{2})}$$

As $\sin(\frac{\pi k}{2}) \neq 0$,

$$\sin^2\left(\frac{\pi k}{2}\right) = 2 \cdot \sin\left(\frac{\pi k}{2}\right) \cdot \cos\left(\frac{\pi k}{2}\right) \cdot \frac{\pi k}{2}$$

or

$$\pi k = \tan\left(\frac{\pi k}{2}\right).$$

The equation has an infinite number of solutions. We are interested in a solution with $k \in (0.5, 1)$. Numeric calculation gives $k \approx 0.74202$ and the average running time $\frac{k}{p} \approx 0.87857$. i.e., is the average number of steps can be reduced by approximately 12.14%.

## 4 Conclusions

We have shown how to reduce the average number of the steps of the Grover's algorithm by approximately 12.14%. The same argument can be applied to a wide range of other quantum query algorithms, such as amplitude amplification, some variants of quantum walks, and NAND formula evaluation, etc: namely is to all algorithms that can be analysed similarly based on rotation from a "bad" to "good state".

## References

1. Lov Grover
   *A fast quantum mechanical algorithm for database search.*
   Proceedings, 28th Annual ACM Symposium on the Theory of Computing (STOC), May 1996, pages 212-219
   also quant-ph/9605043
2. Christof Zalka
   *Grovers quantum searching algorithm is optimal.*
   quant-ph/9711070
3. C. Bennett et al.
   *Strengths and Weaknesses of Quantum Computing.*
   SIAM Journal on Computing (special issue on quantum computing) volume 26, number 5, pages 1510-1523.
   also quant-ph/9701001
4. John Watrous
   *Quantum Computation.*
   Lecture course "CPSC 519/619", University of Calgary, 2006.
   http://www.cs.uwaterloo.ca/ watrous/lecture-notes.html