Quantum Information Processing manuscript No.
(will be inserted by the editor)

A Different Deutsch-Jozsa*

Debajyoti Bera

March 17, 2015

Abstract One of the early achievements of quantum computing was demonstrated by Deutsch and Jozsa in 1992 regarding classification of a particular type of Boolean functions. Their solution demonstrated an exponential speedup compared to classical approaches to the same problem; however, their solution was the only known quantum algorithm for that specific problem so far.

This paper demonstrates another quantum algorithm for the same problem, with the same exponential advantage compared to classical algorithms. The novelty of this algorithm is the use of quantum amplitude amplification, a technique that is the key component of another celebrated quantum algorithm developed by Grover in 1996. A lower bound for randomized (classical) algorithms is also presented which establishes a sound gap between the effectiveness of our quantum algorithm and that of any randomized algorithm with similar efficiency.

Keywords quantum algorithm, Deutsch-Jozsa, amplitude amplification, lower bound, randomized algorithm

1 Introduction

There are a handful of quantum algorithms which are known even outside the scientific community of *Quantum Computing*. The ones with unmatched popularity are an algorithm designed by Lok Grover in 1996 for quantum unordered search[1], and another by Peter Shor in 1997 for integer factoring[2]. Despite numerous other notable achievements in this field[3], the above two problems are still hailed as the hallmark of the superiority of quantum computing.

* The final publication is available at Springer via http://dx.doi.org/10.1007/s11128-015-0976-2

F. Author
Indraprastha Institute of Information Technology
Okhla Industrial Estate Ph-III,
New Delhi, India 110020
Tel.: +91-11-26907442
E-mail: dbera@iiitd.ac.in

Close on the heels of the above algorithms are another two well-known algorithms designed by Deutsch in 1985, and Deutsch-Jozsa in 1992 for classifying certain classes of Boolean functions[4,5]. These algorithms exhibited for the first time that it is possible to perform some operation that is beyond the scope of any classical algorithm. They showed that a quantum algorithm is able to classify a special category of Boolean functions into two classes using one function evaluation, something that no classical algorithm can ever do.

All the three algorithms mentioned above (the two latter problems, as well as their quantum algorithms are similar in nature) are strikingly dissimilar in their approaches. In this paper, we design a new algorithm for the Deutsch-Jozsa problem using concepts of the Grover's search algorithm. Currently, no other algorithm is known for the Deutsch-Jozsa problem apart from the one designed by Deutsch and Jozsa[5]. Our algorithm also has the same exponential advantage that their algorithm has over classical (deterministic) algorithms for their problem. The novelty of our approach is an interesting application of the *quantum amplitude amplification* technique to solve the Deutsch-Jozsa problem. This technique is the key ingredient of Grover's search algorithm, and has been analysed and used effectively since its formalisation by Brassard and Hoyer[6] in 1997 and independently by Grover[7] in 1998. Nevertheless, we believe that this technique has still a lot of potential left. We will show how to use this technique for the Deutsch-Jozsa problem, an approach which we believe is of independent interest.

Both the original algorithm and our algorithm determines the class of a given Boolean function using a constant number of queries without any probability of error. It is easy to see that no classical (deterministic) algorithm can do the same with non-zero probability of error. To formalize this limitation, and add to the growing list of evidences of the superiority of quantum algorithms, we will prove a lower bound on the error probability of any randomized classical algorithm for this problem.

We will discuss the background of the problem, the original quantum algorithm of the Deutsch-Jozsa problem and the amplitude amplification technique in Section 2. In Section 3, we will describe our algorithm for the same problem, and follow it up with our lower bound proofs in Section 4.

2 Background

In this section we will first discuss the Deutsch-Jozsa problem and its original quantum algorithm. We will skip background details of quantum algorithms and circuits – interested readers may refer to the book by Nielsen and Chuang[8].

The Deutsch-Jozsa problem is an extension of the Deutsch's problem [4]. Both of these algorithms were presented in the *Quantum Circuit* model of computation, and we will also present our results in the same model. It is well known that this model is equivalent to other general models of quantum computation.

These problems involve a Boolean function $F : \{0,1\}^n \to \{0,1\}$, whose actual definition is not accessible, but, algorithms can access evaluations of $F(\cdot)$ at values of their choice. Define $F^{-1}(b) = \{x \in \{0,1\}^n \mid F(x) = b\}$ for $b \in \{0,1\}$. Suppose that we are given a promise that F is either constant ($F(x) = 0$ or 1 for all $x \in \{0,1\}^n$), or bal-

anced ($|F^{-1}(1)| = 2^n/2$). The Deutsch-Jozsa problem is about determining which of these cases F satisfy, using minimal evaluations of F. Its predecessor, the Deutsch's problem, is basically the $n = 2$ case of this problem.

Classically, $1 + 2^n/2$ queries to F is sufficient and necessary to ascertain its type with certainty. Deutsch and Jozsa showed how to determine the type of F using only one query to F, an exponential speedup over classical computing. Their original algorithm used two queries, which was subsequently improved to one query by Cleve et al. in 1998[9] – this is the version which we will be presenting here.

For problems like this, where algorithms can only access their input (here F) as black-box (oracle) which they can only query (here, evaluating $F(.)$), the commonly acceptable way to design quantum circuits (equivalently, algorithms) is by allowing the use of U_F oracle gates (operators, for quantum algorithms). This gate is simply a reversible, rather unitary, way to compute $F(x)$ at arbitrary values of x in the following manner: $U_F|x\rangle|y\rangle = |x\rangle|F(x) \oplus y\rangle$ for $x \in \{0,1\}^n$ and $y \in \{0,1\}$. Note that, $U_F|x\rangle|0\rangle$ returns a state which has $|F(x)\rangle$ in its second qubit, which justifies the use of U_F analogous to evaluating $F(.)$.

2.1 Deutsch-Jozsa Circuit

The Deutsch-Jozsa problem can be solved by a circuit C described next. C operates on two registers, first with n qubits initialized to $|0^{\otimes n}\rangle$, and second with one qubit initialized to $|1\rangle$. C is constructed as: $C = (\mathscr{H}^{\otimes n} \otimes I) \cdot U_F \cdot (\mathscr{H}^{\otimes n} \otimes \mathscr{H})$. After the application of C, the first register is measured in the standard basis, which can be found to be in $|0^{\otimes n}\rangle$ if and only if F is constant. Do pay attention to the fact, that C was constructed using only one U_F gate, which is quantum equivalent of evaluating F, and answers with absolute certainty. The proof of correctness is pretty straightforward and we are omitting it in this paper.

2.2 Amplitude Amplification

Amplitude amplification[10] is the key technique behind Grover's unordered search algorithm[1]. It is the quantum analogue of repeated trials to boost sampling probability, however, with a speedup that is usually quadratic compared to classical randomized algorithms. Most of the applications of amplitude amplification, including the original formulation for unordered search, shows that, if there is a sampling gate (operator) with probability p of obtaining a "correct" sample, then after $\Theta(\sqrt{p})$ applications of this gate (along with similar number of other gates), it is possible to obtain a correct sample with probability close to 1. This technique has now been generalized and studied extensively, and tight results are known for a variety of scenarios, including when p is not known.

We will make use of the following version of amplitude amplification from [6]. Hereafter, $\imath$ will denote $\sqrt{-1}$, and N will denote 2^n.

Lemma 1 *Let $\mathscr{A}$ be a quantum algorithm that uses no measurements and that given $|0\rangle$ return $\psi = \sum_{i\in I}|i\rangle|\psi\rangle$ for some finite index set $I \subset \mathbb{Z}$. Let $\chi : I \to \{0,1\}$ be any Boolean function. Define,*

- $A = \{i \in I \mid \chi(i) = 1\}$, $|A\rangle = \sum_{i\in A}|i\rangle|\psi\rangle$, $a = \langle A|A\rangle$
- $B = \{i \in I \mid \chi(i) = 0\}$, $|B\rangle = \sum_{i\in A}|i\rangle|\psi\rangle$, $b = \langle B|B\rangle$

Then there exists a quantum algorithm Q that on input $|0\rangle$ returns

- $(\iota - 1)|A\rangle$ *if* $a = 1/2$
- $\iota|B\rangle$ *if* $a = 0$

In the lemma above, A is the basic sampling gate and $\sum_{i\in I}|i\rangle|\psi\rangle$ is the quantum analogue of a sample from I. The above lemma shows how to construct the operator Q which samples from I with stronger probability guarantees.

3 Alternative Circuit for Deutsch-Jozsa

Amplitude amplification is a technique to boost sampling probability; however, the Deutsch-Jozsa problem is not stated as a sampling problem in its natural form. In this section we will analyse this problem from a different point of view, that will allow us to use Lemma 1 for our problem.

To use the lemma effectively for our purpose, we need another function $F' : \{0,1\}^n \to \{0,1\}$ defined as $F'(x) = F(x) \oplus F(0^n)$. Note that, based on the promise of F, F' is either balanced or constant with value 0. Moreover, F is balanced if and only if F' is balanced. We will use the corresponding oracle gate $U_{F'}|x\rangle|y\rangle = |x\rangle|F'(x)\rangle$ for $x \in \{0,1\}^n$ and $y \in \{0,1\}$. We will discuss later how to construct $U_{F'}$.

Lemma 2 *There exists a quantum circuit using oracle gates $U_{F'}$ that can determine if F' is balanced or constant.*

Proof Our proof will involve a clever application of Lemma 1 acting a system of $n+1$ qubits. For $\mathscr{A}$ we will use the circuit $U_{F'} \cdot (H^{\otimes n} \otimes I)$.

$$\begin{aligned}
\mathscr{A}|0\rangle &= U_{F'} \cdot (H^{\otimes n} \otimes I)|0^{\otimes n}\rangle|0\rangle \\
&= \sum_{x\in\{0,1\}^n} \frac{1}{\sqrt{N}}|x\rangle|F'(x)\rangle \\
&= \sum_{\substack{x\in\{0,1\}^n \\ F'(x)=0}} \frac{1}{\sqrt{N}}|x\rangle|0\rangle + \sum_{\substack{x\in\{0,1\}^n \\ F'(x)=1}} \frac{1}{\sqrt{N}}|x\rangle|1\rangle
\end{aligned}$$

We will use $\chi = F'$ which gives us $A = \{x \in \{0,1\}^n \mid F'(x) = 1\}$ and $B = \{x \in \{0,1\}^n \mid F'(x) = 0\}$. Let N_1 denote $|A|$ and N_0 denote $|B|$. If F (and, so F') is balanced, $N_0 = N_1 = N/2$. If F is a constant function, F' is the constant function 0, and therefore $N_0 = N$, $N_1 = 0$.

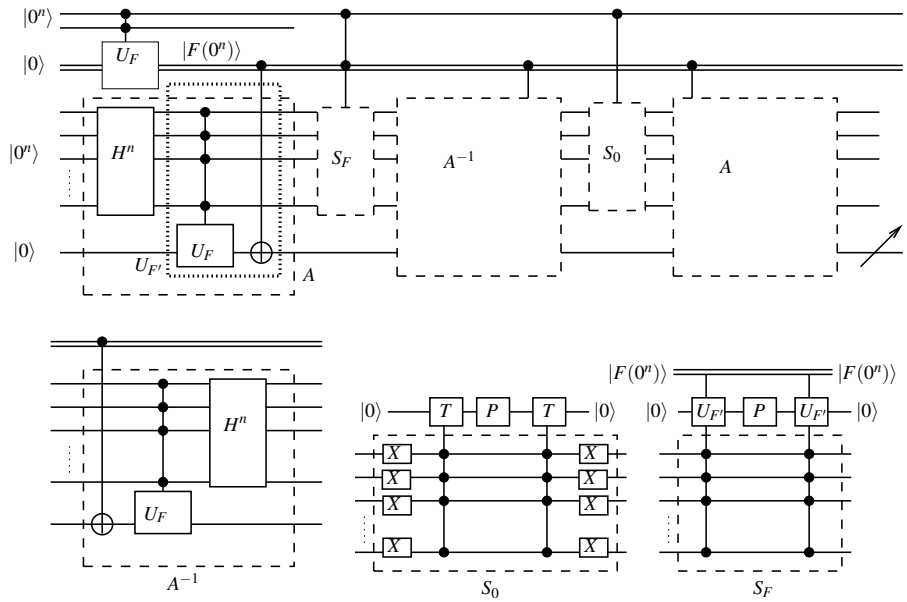

Fig. 1 Circuit for distinguishing between constant F and balanced F

The calculations above give us $|A\rangle = \sum_{x\in A} \frac{1}{\sqrt{N}}|x\rangle|1\rangle$ and similarly, $|B\rangle$, from which we obtain $a = \langle A|A\rangle = N_1/N$ and $b = \langle B|B\rangle = N_0/N$. If F is balanced, then $a = 1/2$ and if F is constant, then $a = 0$.

Now, we are ready to apply Lemma 1. As per the lemma, there exists a circuit which, on $|0^{\otimes n+1}\rangle$, creates the state $(\imath - 1)|A\rangle$ if $a = 1/2$ (which happens when F is balanced), and $\imath|B\rangle$ if $a = 0$ (which happens when F is constant). Therefore, if we measure the output of the circuit in the standard basis, then the state of the last qubit is $|0\rangle$ is F is constant, and $|1\rangle$ if F is balanced.

Theorem 1 *Given an n-bit to 1-bit Boolean function F that is either balanced or constant, there exists a quantum circuit that uses six oracle gates for F and can determine if F is a balanced or a constant function. The quantum circuit uses 6 (unbounded fanin) oracle gates, linear number of other standard gates (including unbounded-fanin generalized Toffoli gates), and has constant depth.*

Proof Lemma 2 immediately gives a proof for this theorem since it is straight forward to construct $U_{F'}$ using U_F. However, we will take this opportunity to present the circuit explicitly. The general structure of the Q operator in Lemma 2 was described in [6], which we have adopted for our purpose.

First, we will use U_F to construct $U_{F'}$. Recall that, $U_{F'}|x\rangle|y\rangle = |x\rangle|y \oplus F(x) \oplus F(0^n)\rangle$, and therefore, this operator can be implemented by first applying U_F on its qubits, and then using a CNOT gate whose control qubit carries the value $F(0^n)$. This control qubit is never changed after being initialized to $|F(0^n)\rangle$ and is reused several times in our circuit.

The complete circuit proving the theorem is $\mathscr{C} = \mathscr{A} S_0 \mathscr{A}^{-1} S_F \mathscr{A}$ (see Figure 1 for illustration), which effectively acts on an $(n+1)$-dimensional space $\mathscr{H}^{\otimes n} \otimes \mathscr{H}$, and is constructed out of the listed operators below. We need $(n+1)$ additional qubits, initialized to $|0^{\otimes n}\rangle|0\rangle$, and which are reused several times in the circuit. The controlled T gate represents the generalized Toffoli gate, the X gate represents the Pauli-X operator, and the P gate represents the phase gate $|0\rangle\langle 0| + \imath|1\rangle\langle 1|$.

- $\mathscr{A} = U_{F'} \cdot (H^{\otimes n} \otimes I)$.
  In reality, $\mathscr{A}$ and $\mathscr{A}^{-1}$ acts on $(n+2)$ qubits. However, its control qubit is always $|F(0^n)\rangle$ and therefore, can be reused. $\mathscr{A}^{-1}$ is simply $(H^{\otimes n} \otimes I) \cdot U_{F'}^{-1}$.
- S_0 and S_F require one and two additional qubits respectively, initialized to particular states. However, they leave these qubits in their initial state at the end of their operations. This allows these additional qubits to be reused and therefore, these gates can be considered to act on an n-qubit space for all practical purpose. For the sake of brevity, the additional qubits are omitted for these operators in the rest of this proof. Refer to Figure 1 for implementation of S_0 and S_F.
- $S_0 = (I - (1-\imath)|0^n\rangle\langle 0^n|)$.
  This can be created independently of the input function. On the basis states it acts as follows: $|0^n\rangle \to \imath|0^n\rangle$ and $|a\rangle \to |a\rangle$ for $a \neq 0^n$.
- $S_F = (I - (1-\imath) \sum_{x\,:\,F'(x)=1} |x\rangle\langle x|)$. Its action on the basis states can be described as follows: $|a\rangle \to \imath|a\rangle$ if $F'(a) = 1$ and $|a\rangle \to |a\rangle$ otherwise.

The input to the circuit is $|0^n\rangle|0\rangle$ and after applying all the gates, the last qubit is measured in the computational basis. As illustrated in the figure, the number of U_F gates used is 6.

Finally, we briefly describe the operation of the circuit.

$$
\begin{aligned}
\mathscr{C}|0^n\rangle|0\rangle &= \mathscr{A} S_0 \mathscr{A}^{-1} S_F \mathscr{A} |0^n\rangle|0\rangle \\
&= \mathscr{A} S_0 \mathscr{A}^{-1} S_F (|A\rangle + |B\rangle) \quad \text{(From Lemma 2)} \\
&= \mathscr{A} S_0 \mathscr{A}^{-1} (\imath|A\rangle + |B\rangle) \\
&= \mathscr{A} (I - (1-\imath)|0^n\rangle\langle 0^n|) \mathscr{A}^{-1} (\imath|A\rangle + |B\rangle) \\
&= \left(I - (1-\imath)\big(|A\rangle + |B\rangle\big)\big(\langle A| + \langle B|\big) \right) \cdot \\
&\quad (\imath|A\rangle + |B\rangle) \\
&= |A\rangle \left(\imath - (1-\imath)\frac{\imath N_1}{N} - (1-\imath)\frac{N_0}{N} \right) + \\
&\quad |B\rangle \left(1 - (1-\imath)\frac{N_0}{N} - (1-\imath)\frac{\imath N_1}{N} \right) \\
&= \begin{cases} (\imath - 1)|A\rangle & \text{if } N_0 = N_1 = N/2 \\ \imath|B\rangle & \text{if } N_1 = 0, \text{ i.e., } \langle A|A\rangle = 0 \end{cases}
\end{aligned}
$$

The measured qubit is in state $|1\rangle$ if F is balanced, and in $|0\rangle$ if F is constant.

4 Classical Lower Bound

Any deterministic classical algorithm must make at least $1+2^{n-1}$ queries to give a correct answer with certainty[5]. In contrast to deterministic computations, a simple probabilistic algorithm that queries for $f(x)$ for, say 6, randomly chosen distinct values of $x \in \{0,1\}^n$, and outputs "constant" if and only if all values of $f(x)$ are same, has an error probability at most $\frac{N}{2N-1}\frac{N-1}{2N-2}\frac{N-2}{2N-3}\frac{N-3}{2N-4}\frac{N-4}{2N-5} \sim 1/2^5$. We are interested in determining if this probability of error can be made arbitrarily low.

We will next show a negative answer to the above question by demonstrating a lower bound on the error probability of any randomized algorithm making at most 6 queries. In fact, we will prove a more general statement for any specific number of queries. Since we are interested in only query complexity, we will state and prove the required lemma in the (randomized) decision tree model of computation.

The standard way to prove lower bound in this model of computation starts with the observation that any randomized decision tree can be thought of as some distribution over a fixed family of deterministic decision trees. Our lower bound result will demonstrate that, for any probability distribution on deterministic decision trees of depth at most k, the probability of error is at least $1/2^{2k}$ when the decision tree used is chosen randomly according to this distribution. This will immediately lead us to the main result of this section.

Theorem 2 *Any randomized decision tree making at most k queries (for $k \leq 2^{n-1}$) has an error probability at least $1/2^{2k}$.*

Before the proof, we need a technical lemma. We can think of F as an N-bit string of zeroes and ones. Querying for some $F(x)$ is equivalent to querying for the x-th bit of F in this representation. A constant F implies that F is all zeroes or all ones, and a balanced F indicates an F with exactly $N/2$ ones.

We will therefore consider decision trees of depth at most k, and those, which on input F, query for certain bits of F, and try to determine if F is constant or balanced. We denote the set of such decision trees as $\mathscr{T} = \{T_1, \ldots, \}$. μ will represent any given distribution on $\mathscr{T}$.

Lemma 3 *There exists a set of $N/2$ indices D^* such that with probability at least $1/\binom{2k}{k}$ a randomly chosen T (according to μ) does not query any index outside of D^* when input is 0^N. Formally,*

$$\Pr_{T\sim\mu}\left[T(0^N) \textit{ queries indices only in } D^*\right] \geq 1 \Big/ \binom{2k}{k}$$

Proof Without loss of generality, assume N is divisible by $2k$. Divide the indices of F, namely $\langle 0,1,\ldots,N-2,N-1\rangle$, into $2k$ equal sized blocks, $\mathscr{B} = \{B_1,\ldots,B_{2k}\}$. Now, construct a set $\mathscr{D}$ of all k-element subsets of $\mathscr{B}$ and denote its elements as $\{D_1,\ldots,D_M\}$ where, $M = \binom{2k}{k}$. Note that, there is a one-one correspondence between each D_i and every set of $N/2$ distinct indices.

Order the elements of $\mathscr{D}$ (in any arbitrary order). Now, we say that a decision tree T does not query any index outside $D \in \mathscr{D}$ on input 0^N if D is the first element of $\mathscr{D}$ (according to the order) which contains all the indices that T queries upon input 0^N.

Our construction ensures that, there always exists a unique D for every T such that $T(0^N)$ queries indices only in D. Since there are M possible values of D, there must exist some $D^* \in \mathscr{D}$ such that

$$\Pr_{T\sim\mu}[T(0^N) \text{ queries only } D^*] \geq 1/M$$

This proves the lemma.

Proof (Proof of Theorem 2) Using Lemma 3, we know that there exists a set J of $N/2$ indices such that,

$$\Pr_{T\sim\mu}\left[T \text{ queries only indices in } J \text{ on input } 0^N\right] \geq 1 \Big/ \binom{2k}{k} \tag{1}$$

Now, starting with 1^N, flip the bits indicated by the indices contained in J to 0 – call the resulting string y. It is clear from (1) that,

$$\Pr_{T\sim\mu}\left[T \text{ gives same answer on both } 0^N \text{ and } y\right] \geq 1 \Big/ \binom{2k}{k} \geq 1/2^{2k}$$

However, 0^N is constant, and y is balanced. Therefore, we have that a randomly chosen T has an error probability at least $1/2^{2k}$.

Applying the theorem to $k = 1$, any randomized algorithm has an error probability at least $1/4$ if it can ask for at most one evaluation of F to determine whether F is constant or balanced; this is something that the Deutsch-Jozsa quantum algorithm is able to do without any probability of error.

5 Conclusion

In this paper we showed how to use the powerful amplitude amplification technique to obtain another solution to the Deutsch-Jozsa problem without any probability of error. While the original algorithm used two applications (subsequently improved to one) of a gate to compute the special function F at particular values, our algorithm uses six such gates, still asymptotically same as the original algorithm. Apart from these gates, our algorithm uses linear number of single qubit and multi-qubit gates, and runs in constant time; this is similar to the standard Deutsch-Jozsa algorithm which also uses linear number of single-qubit gates and runs in constant time.

It was already known that an exponential number of queries (a query is structurally equivalent to a gate) are necessary to solve the problem without any error probability. We further showed that any classical randomized algorithm making at most six queries must suffer from error probability at least $1/2^{12}$.

References

1. L.K. Grover, in *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing* (ACM Press, New York, New York, USA, 1996), pp. 212–219. DOI 10.1145/237814.237866. URL http://dl.acm.org/citation.cfm?id=237814.237866
2. P.W. Shor, SIAM Journal on Computing **26**(5), 1484 (1997). DOI 10.1137/S0097539795293172. URL http://epubs.siam.org/doi/abs/10.1137/S0097539795293172
3. S. Jordan. Quantum Algorithm Zoo. http://math.nist.gov/quantum/zoo/
4. D. Deutsch, Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences **400**(1818), 97 (1985). DOI 10.1098/rspa.1985.0070. URL http://rspa.royalsocietypublishing.org/cgi/doi/10.1098/rspa.1985.0070
5. D. Deutsch, R. Jozsa, Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences **439**(1907), 553 (1992). DOI 10.1098/rspa.1992.0167. URL http://rspa.royalsocietypublishing.org/content/439/1907/553.short
6. G. Brassard, P. Hoyer, in *Proceedings of the Fifth Israeli Symposium on Theory of Computing and Systems* (IEEE Comput. Soc, 1997), pp. 12–23. DOI 10.1109/ISTCS.1997.595153. URL http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=595153
7. L. Grover, Physical Review Letters **80**(19), 4329 (1998). DOI 10.1103/PhysRevLett.80.4329. URL http://link.aps.org/doi/10.1103/PhysRevLett.80.4329
8. M.A. Nielsen, I.L. Chuang, *Quantum Computation and Quantum Information (Cambridge Series on Information and the Natural Sciences)*, 1st edn. (Cambridge University Press, 2004). URL http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/0521635039
9. R. Cleve, A. Ekert, C. Macchiavello, M. Mosca, Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences **454**(1969), 339 (1998). DOI 10.1098/rspa.1998.0164. URL http://rspa.royalsocietypublishing.org/cgi/doi/10.1098/rspa.1998.0164
10. G. Brassard, P. Hoyer, M. Mosca, A. Tapp, in *Quantum Computation and Quantum Information: A Millennium Volume*, *AMS Contemporary Mathematics Series*, vol. 305 (American Mathematical Society, 2002)